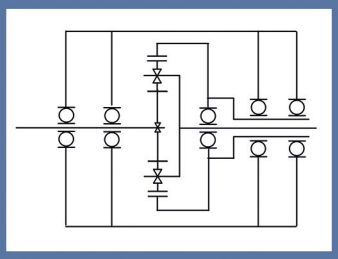
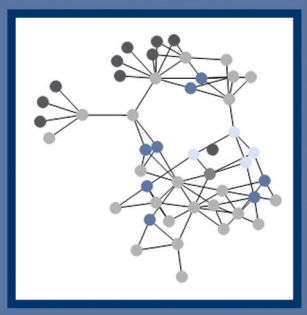
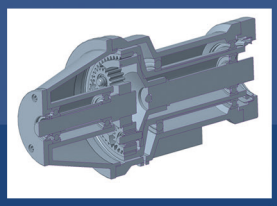
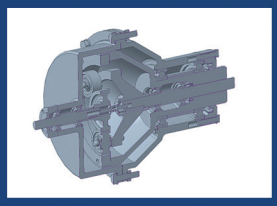
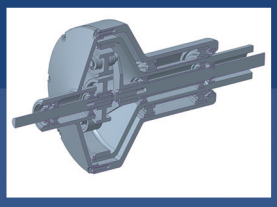










FAU Studien aus dem Maschinenbau 466

Sebastian Bickel

Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung

FAU University Press

Sebastian Bickel

Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung

FAU Studien aus dem Maschinenbau

Band 466

Herausgeber der Reihe:

Prof. Dr.-Ing. Jörg Franke
Prof. Dr.-Ing. Nico Hanenkamp
Prof. Dr.-Ing. habil. Tino Hausotte
Prof. Dr.-Ing. habil. Marion Merklein
Prof. Dr.-Ing. Sebastian Müller
Prof. Dr.-Ing. Michael Schmidt
Prof. Dr.-Ing. Sandro Wartzack

Sebastian Bickel

Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung

Dissertation aus dem Lehrstuhl für Konstruktionstechnik (KTmfk)
Prof. Dr.-Ing. Sandro Wartzack

Erlangen
FAU University Press
2025

Bibliografische Information der Deutschen Nationalbibliothek:
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Kontakt: Sebastian Bickel, Friedrich-Alexander-Universität Erlangen-Nürnberg, (ROR https://ror.org/00f7hpc57), ORCID: 0000-0002-9558-3506

Bitte zitieren als
Bickel, Sebastian. 2025. *Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung.* FAU Studien aus dem Maschinenbau Band 466. Erlangen: FAU University Press. DOI: 10.25593/978-3-96147-833-0

Das Werk, einschließlich seiner Teile, ist urheberrechtlich geschützt.
Die Rechte an allen Inhalten liegen bei ihren jeweiligen Autoren.
Sie sind nutzbar unter der Creative-Commons-Lizenz BY-NC.

Der vollständige Inhalt des Buchs ist als PDF über OPEN FAU
der Friedrich-Alexander-Universität Erlangen-Nürnberg abrufbar:
https://open.fau.de/home

Verlag und Auslieferung:
FAU University Press, Universitätsstraße 4, 91054 Erlangen

Druck: docupoint GmbH

ISBN: 978-3-96147-832-3 (Druckausgabe)
eISBN: 978-3-96147-833-0 (Online-Ausgabe)
ISSN: 2625-9974
DOI: 10.25593/978-3-96147-833-0

Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur
Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von

Sebastian Bickel, M.Sc.

aus Bamberg

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung:	07.05.2025
Gutachter:	Prof. Dr.-Ing. Sandro Wartzack
	Prof. Dr.-Ing. Roland Lachmayer
	Leibniz Universität Hannover

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Konstruktionstechnik (KTmfk) der Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU). Wesentliche Teile dieser Arbeit wurden im Rahmen der Forschungsprojekte EU EFRE E|Asy-Opt (Kompetenz- und Analyseprojekt für die Prozess- und Produktionsoptimierung mittels Data Mining) und WA 2913/47-1 erarbeitet, sodass ich dem EFRE und der DFG für die Unterstützung danke. Ich möchte diese Gelegenheit nutzen, um mich bei allen zu bedanken, die auf fachlicher und zwischenmenschlicher Ebene in einem erhöhten Maße zum Gelingen dieser Arbeit beigetragen haben.

Mein besonderer Dank gilt dabei zunächst meinem Doktorvater Prof. Dr.-Ing. Sandro Wartzack für seine wertvolle wissenschaftliche wie persönliche Betreuung. Die ermöglichten Freiheiten und die zahlreichen fachlichen Diskussionen haben entscheidend zum Gelingen der Arbeit beigetragen.

Daneben möchte ich mich bei Prof. Dr.-Ing Roland Lachmayer für die Übernahme des Zweitgutachtens und Interesse an meiner Arbeit bedanken. Zusätzlich gilt mein Dank Prof. Dr.-Ing. Tobias Günther für das fachfremde Gutachten sowie Prof. Dr.-Ing. Sebastian Müller für den Vorsitz während der mündlichen Prüfung.

Zudem bedanke ich mich herzlichst bei allen ehemaligen und aktiven Kollegen für die gute Zusammenarbeit, die gegenseitige Unterstützung und die schöne gemeinsame Zeit. Hervorheben muss ich hier vor allem meine beiden Oberingenieure aus der Fachgruppe Digital Engineering. Prof. Dr.-Ing. habil. Benjamin Schleich danke ich für intensive Betreuung und ausführliches Feedback zu Beginn meiner Lehrstuhlzeit. Für die herausragende fachliche und freundschaftliche Zusammenarbeit mit Dr.-Ing. Stefan Götz möchte ich mich auch bedanken, vor allem für die ausführlichen Diskussion und konstruktiven Rückmeldungen über meine Forschungsthemen und diese Arbeit. Ein weiterer besonderer Dank gilt meinen Kollegen und Kolleginnen aus der Fachgruppe Digital Engineering. Besonders hervorheben möchte ich hier meine Bürokollegen Tobias Sprügel, Jakob Harlan, Benjamin Geschütz, Leonie Walter, Marc Behringer und Yannik Utz. Vielen Dank für das stets sehr gute Arbeitsklima und den intensiven fachlichen Austausch.

Darüber hinaus gilt mein Dank auch allen Studierenden, die mich durch wissenschaftliche Arbeiten und Tätigkeiten am Lehrstuhl in der Forschung unterstützt haben. Besonders sind hier Adrian Baron, Julian Bittner, Leonard

Lieg, Markus Johannes Seitz, Mostafa Frastaoui, Louis Lang und Florian Schäfer zu nennen.

Der größte Dank gehört meiner Familie und Freunden, die mich in der gesamten Zeit stets unterstützt, motiviert und auch besonders Verständnis aufgebracht haben. Hervorzuheben sind hier insbesondere meine Großeltern, meine Eltern und mein Bruder, die mich stets gefördert und unterstützt haben. Am Ende möchte ich mich noch ganz herzlich bei meiner Freundin Theresa für ihre Unterstützung, Geduld und Verständnis während meiner Lehrstuhlzeit und bei der Entstehung dieser Arbeit bedanken.

Erlangen, 29.05.2025 Sebastian Bickel

Inhaltsverzeichnis

Formelzeichen- und Abkürzungsverzeichnis . . . . . **ix**

Bildverzeichnis . . . . . **xiii**

Tabellenverzeichnis . . . . . **xix**

1 Einleitung **1**

1.1 Motivation und Problemstellung 2

1.2 Aufgabenstellung und Aufbau der Arbeit 3

2 Stand der Forschung **5**

2.1 Produktentwicklungsprozess 5

2.1.1 VDI 2221/23 7

2.1.2 Formate 10

2.1.3 Datengetriebene Produktentwicklung 16

2.2 Datengetriebene Methoden 21

2.2.1 Übersicht Machine-Learning 22

2.2.2 Objekterkennung (Bilder) 28

2.2.3 Geometrie und Machine-Learning 41

2.2.4 Verknüpfung von Skizzen und Geometrie 60

3 Handlungsbedarf und Vorgehensweise **63**

3.1 Handlungsbedarf 63

3.2 Hypothese und Forschungsfragen 65

3.3 Vorgehensweise 66

4 Erkennung und Kontextualisierung technischer Skizzen . . **69**

4.1 Modulübersicht 69

4.2 Symbolerkennung 69

4.2.1 Trainingsdatenerzeugung 70

4.2.2 Objekterkennungsmodelle 74

4.3 Anwendung der Symbolerkennung 74

4.3.1 Implementierung der Objekterkennungsmodelle . 74

4.3.2 Auswertemetriken und Testdatensätze 75

4.3.3 Ergebnisse 77

4.3.4 Auswertung 79

4.4 Iteration Symbolerkennung 82

4.5 Kontextualisierung . . . 83
4.5.1 Transformation in Graphen . . . 84
4.5.2 Ähnlichkeitssuche . . . 86
4.5.3 SysML Überführung . . . 89
4.5.4 Weitere Anwendungsmöglichkeiten . . . 90
4.6 Diskussion . . . 91

5 Ähnlichkeitssuche von Baugruppen mittels Skizzen . . . 93

5.1 Modulübersicht . . . 94
5.1.1 Kontaktgraph . . . 95
5.1.2 Bauteilerkennung . . . 97
5.1.3 Graphaufbereitung und -matching . . . 101
5.2 Anwendung an Demonstratordatensätzen . . . 107
5.2.1 Auswertemetriken . . . 107
5.2.2 Getriebestufen-Datensatz . . . 108
5.2.3 Baugruppen-Datensatz . . . 115
5.3 Diskussion . . . 120

6 Ähnlichkeitssuche für Bauteile . . . 121

6.1 Modulübersicht . . . 122
6.1.1 Datenaufbereitung . . . 123
6.1.2 Objektabstraktion . . . 126
6.1.3 Shape-Deskriptor-Erzeugung . . . 126
6.2 Untersuchung der Einflussfaktoren auf die Retrievalgenauigkeit . . . 128
6.2.1 Bauteilausrichtung . . . 129
6.2.2 Featureextraktion . . . 131
6.2.3 Projektionsart und Distanzmetrik . . . 132
6.2.4 Dimensionsreduktion . . . 134
6.3 Vergleich mit Stand der Technik Verfahren . . . 136
6.3.1 Vergleich mit eigenen Vorarbeiten . . . 136
6.3.2 Vergleich mit etablierten Verfahren . . . 137
6.4 Diskussion . . . 139

7 Anwendungsszenario . . . 141

7.1 Szenario Vorstellung . . . 141
7.2 Konzeptphase . . . 142
7.3 Entwurfsphase . . . 144
7.4 Diskussion . . . 146

8 Zusammenfassung und Ausblick **149**

9 Summary and Outlook **153**

Anhang **157**

A Ergänzungen zu Kapitel 2 . . . . . **157**

A.1 Auswertung Objekterkennung - Beispiel 157
A.2 Auswertung Ähnlichkeitssuche - Beispiel 158
A.3 Convolutional Neural Network - Erklärung 159
A.4 Datensätze - Übersicht 161
A.4.1 MCB 161
A.4.2 ESB 162
A.5 Auswertemetriken Klassifikation - Übersicht 163
A.6 Auswertemetriken Klassifikation - ROC Diagramm 164
A.7 MCB Datensatz - Vergleichsergebnisse 164
A.8 Ähnlichkeitssuche - Literaturübersicht 165
A.9 Ähnlichkeitssuche - Stand der Technik-Verfahren 174
A.10 Symbolerkennung - Literaturübersicht 176

B Ergänzungen zu Kapitel 4 . . . . . **187**

B.1 Datenerzeugung - Algorithmen 187
B.1.1 Pillow 187
B.1.2 SketchGraph 188
B.2 Datensätze - Übersicht 189
B.2.1 Getriebestufen 189
B.2.2 Übersicht aller Bilder 190
B.2.3 Unbekannte Baugruppen 195
B.2.4 Übersicht aller Bilder 196
B.2.5 Übersicht der Bewertungsmetriken für beide Test-Datensätze 200
B.2.6 Häufigkeit der Symbole für die Iteration der Sketch-Graph Datenerzeugung 201
B.2.7 Parameter und Ergebnisse für GED-Skizzen - Vergleich 202

C Ergänzungen zu Kapitel 5 . . . . . **203**

C.1 Formel für Projektion 203
C.2 PointNet++ 203
C.3 Demonstrator-Datensatz Skizze - Baugruppe 205

C.4 Pseudocode Graphaufbereitung 210
C.5 Demonstrator-Datensatz - Baugruppe 215
C.6 Vergleich Bauteilerkennung 226
C.7 Vergleich Projektion - Stand der Technik 227
C.8 Ergebnis automatisierte Bauteilerkennung 228
C.9 Regel für Baugruppen 229
C.10 Scores für Baugruppen-Datensatz 232

D Ergänzungen zu Kapitel 6 237

D.1 Ähnlichkeitssuche Bauteile 237
D.1.1 Einfluss der Anzahl an Punkten pro Sampling . . . 239
D.1.2 Übersicht Studie Dimensionreduktionsgröße 240
D.1.3 Siamese Network 245

E Ergänzungen zu Kapitel 7 247

E.1 Datensätze 247
E.2 Suchergebnisse GED-Distanz - Prinzipskizzen 248

Literaturverzeichnis 249

Formelzeichen- und Abkürzungsverzeichnis

Symbol	*Einheit*	*Beschreibung*
AAD		Actual Artifacts Dataset
ANN		Artificial Neural Networks
AR		Augmented Reality
AP		Average Precision
BRep		Boundary-Representation
$\vec{C}$		Koordinaten-Nullpunt [0,0,0]
$c_{Operation}$		cost
c		Mittelpunkt-Matrix
CAD		Computer-Aided-Design
CAE		Computer-Aided-Engineering
CAM		Computer-Aided-Manufacturing
CAP		Computer-Aided-Planning
CAQ		Computer-Aided-Quality
CNN		Convolutional Neural Networks
CSG		Constructive Solid Geometry
dc		Distanz Mittelpunkt-Matrix
DCG		Discounted Cumulated Gain
dis_{EW}		Eigenwert-Score
DLAN		Deep Local feature Aggregation Network
DM		Data Mining
dn		Distanz Normalen-Matrix
DPM		Deformable Part-based Model
EAM		Enriched Assembly Model
ECGM		error-correcting graph machting
ESB		Engineering Shape Benchmark
F		F-Maß
F_α		F_α Maß
FEM		Finite-Element-Methode

Symbol	*Einheit*	*Beschreibung*
FP		False positives
FPR		False-Positive-Rate
FPN		Pyramid Networks
FPS		Frames per Second
FN		False negatives
FR		Fehlerrate
FT		First Tier
FAST		Functional Analysis System Technique
G		Genauigkeit
G_n		Graph
G		Geometric
GED		Graph Edit Distance
GLOH		Gradient location-orientation histogram
H		Hybrid
HOG		Histogram of Oriented Gradients
IGES		Initial Graphics Exchange Specification
IoT		Internet of Things
IoU		Intersection over Union
J		Jaccard-Ähnlichkeit
KBE		Knowledge Based Engineering
KI		Künstliche Intelligenz
KNN		k-nearest neighbors Algorithmus
LIDAR		Light Detection and Ranging
LLM		Large Language Models
LR		Label required
M_A, M_B		Menge A, B
mAP		mean Average Precision
MCB		Mechanical Component Benchmark
MCS		Most Common Subgraph
ML		Machine Learning
MOPS		multi-scale oriented patches

Symbol	***Einheit***	***Beschreibung***
MVCNN		Multi-View CNN
n		Normalen-Matrix
$\vec{n}$		Normals Vertices
N		Negativität
NLP		Natural Language Processing
NN		Nearest Neighbor
NT		no training
NURBS		non uniform rational B-Splines
OCR		Object Character Recognition
OMG		Object Management Group
P		Positivität
$\vec{PC}$		PointCloud Vertices
PCA		Hauptkomponenten-Analyse
PDM		Produktdaten-Management
P&ID		Piping and Instrument Diagrams
PPS		Produktionsplanung und -steuerung
PR-		Precision-Recall-Kurve
PR		Präzision
R-CNN		region-based CNN
RBM		Restricted Boltzmann Machine
RGB		Red-Green-Blue
ROC		Receiver-Operating-Characteristic
RoI		Region of Interest
RPN		Region Proposal Network
SE		Spezifität
SHREC		SHape REtrieval Challenge
SIFT		Scale Invariant Feature Transform
sim_J		Jaccard-Ähnlichkeit
SOM		Self-Organizing Maps
SSD		Single Shot Detector
ST		Second Tier

Symbol	***Einheit***	***Beschreibung***
STL		Stereolithografie
STEP		Standard for the Exchange of Product model data
SVM		Support Vektor Machine
SWOT		Strengths, Weaknesses, Opportunities and Threats
SysML		oder Systems Modeling Language
t-SNE		t-distributed stochastic neighbor embedding
TP		True positives
TPR		True-Positive-Rate
TN		True negatives
UML		Unified Modeling Language
USL		Unsupervised Learning
VB		View-Based
VDI		Verein Deutscher Ingenieure
VR		Virtuelle Realität
VRN		Voxception-ResNet
XML		Extensible Markup Language
YOLO		You Only Look Once
λ		Eigenwerte der Laplace-Matrix

Bildverzeichnis

1 Ist- und Soll-Zustand für die Nutzung vorhandener Daten in der Produktentwicklung 1
2 Übersicht des inhaltlichen Aufbaus der Arbeit 4
3 Formate 8
4 Beispiele für die unterschiedliche Dimensionalität von Formaten in der Produktentwicklung am Beispiel eines Scheibenwischergestänges 10
5 Beispiele für verschiedene Skizzen aus der Ingenieurdomäne 11
6 Beispiel für eine Prinzipskizze und die daraus resultierende technische Zeichnung eines entworfenen Bauteils nach [36] 12
7 Darstellung der verschiedenen Geometrieformate in der Produktentwicklung nach [82] 13
8 Übersicht der zeitlichen Einordnung verschiedener Produktentwicklungsphasen nach [99] 17
9 Übersicht eines Einsatzmodells für die datengetriebenen Produktentwicklung nach [102] 18
10 Verknüpfung von datengetriebenen Methoden und den Schritten des Produktentwicklungsprozesses nach [10] . . . 20
11 Grobe Einordnung der verschiedenen Begriffe nach [113] . . 22
12 Beispiel für das Ergebnis einer Klassifikation mit zwei Merkmalen, wobei ein neuer Datenpunkt (schwarz) in die dunkelblaue Kategorie einsortiert wird 24
13 Beispielhafte Anwendung von Clusteringalgorithmen zur Einordnung von Datenpunkten in vier Klassen 25
14 Übersicht für den Ablauf von verstärkendem Lernen nach [127] 26
15 Prozessvergleich zwischen Machine und Deep-Learning nach [136, 138] 27
16 Darstellung der unterschiedlichen Aufgaben von Bilderkennungsverfahren nach [157] 28
17 Zeitliche Einordnung der Meilensteine für Verfahren der Objekterkennung nach [160] 31
18 Modellübersicht für den YOLO-Detector nach [174] 32
19 Aufbau eines R-CNN-Modells nach [179] 33
20 Ablaufübersicht für das MASK-R-CNN-Modell nach [183] . 34
21 Bestimmung des IoU Wertes für die Auswertung von Objekterkennungsverfahren 35
22 Beispiele für unterschiedliche IoU Werte 36
23 Ablauf und Einordung des Digitalisierungsvorgehen für Zeichnungen im Ingenieurwesen nach [187] 37

24 Vorstellung der Ergebnisse der Literaturrecherche mittels einer Domänenverteilung und Darstellung des zeitlichen Verlaufs 38
25 Einordnung der Verfahren anhand der verwendeten Datengrundlage 39
26 Beispiel für den Netlist-Ansatz zur Überführung der Symbole für einfache Simulationen nach [191] 40
27 Übersicht der verschiedenen Geometrieformate und Anwendungsfelder nach [83, 84, 205, 207, 208] 42
28 Zufällige Auswahl an Bauteilen des MCB- und ESB-Datensatzes 44
29 Allgemeine Einteilung der Bauteilerkennungs-Verfahren nach dem Input-Format 45
30 Wahrheitsmatrix für Klassifikation nach [233, 235] 48
31 Ähnlichkeit in unterschiedlichen Formen nach [248] 50
32 Darstellung der verschiedenen Ähnlichkeitsarten für Baugruppen nach [249] 51
33 Allgemeines Vorgehen zur Geometrie-Ähnlichkeitssuche nach [252] 52
34 Beispiel-Framework für eine Ähnlichkeitssuchmethode nach [253] 53
35 Überblick über die Anzahl der Veröffentlichungen im Laufe der Zeit, sortiert nach Kategorien 56
36 Ein Ausschnitt eines Baugruppenaufbaus für ein Scheibenwischergestänge 59
37 Übersicht der Anwendungsfälle für 2D-Skizzen nach [332] 61
38 Darstellung der Forschungslücke mit den Bereichen Ähnlichkeitssuche und der fehlenden Unterstützung bei der Nutzung vorhandener Daten 65
39 Überblick und Einordnung der unterschiedlichen Methoden in den PEP inkl. Kapitelzuweisung und Forschungsfragen . 66
40 Gesamtübersicht des Sub-Vorgehensmodells nach [P4–P6] 70
41 Beispiel für Graph zur Erzeugung eines Lagersymbols mittels SketchGraphs nach [S5] 72
42 Beispielbilder für die Datenerzeugungsmethoden Pillow und SketchGraph 73
43 Sammlung der implementierten Symbole nach [S5] 76
44 Beispiel-Darstellung der unterschiedlichen Getriebestufen 77
45 Beispiel für unterschiedliche Baugruppenklassen und die dazugehörigen Skizzen 78
46 Gute und schlechte Erkennungsbeispiele für die trainierten Modelle und verschiedene Trainingsdatensätze 81

47 Vergleich des neuen Datenerzeugungsvorgehens mit den vorherigen Varianten 82
48 Exemplarisches Testbild mit detektierten Symbolen des mit Datensatz 2 trainierten Modells 84
49 Vorgehen zur Umwandlung der erkannten Symbole in Graphen nach [P5] 85
50 Screenshot des Suchergebnisse für eine exemplarische Text-Eingabe 87
51 Beispiel für die Suche mittels Graph-Input und handgezeichneter Skizze 88
52 Auszug der Ergebnisse für das Subgraph-Matching nach [P5] 89
53 Beispiel für die Schnittstelle zwischen Skizzen und SysML Modellen 90
54 Einordnung des Vorgehens zum Finden von Baugruppen auf Basis von Zeichnungen in die VDI 2221 [26] 93
55 Übersicht des entwickelten Moduls zur Verknüpfung von Skizzen und Baugruppen 94
56 Beispiel für die Umwandlung einer STEP Baugruppe in einen Kontaktgraphen 96
57 Überblick des Verfahrens zur Bauteilerkennung nach [P7] . 98
58 Vergleich der verschiedenen Projektionsarten, welche für die Bauteilerkennung verwendet werden nach [P9] 99
59 Ergebnisse der Benchmark-Untersuchung für die Bauteilerkennung nach [P7] 100
60 Beispiele für Baugruppen- und Prinzipskizzen-Graphen unterschiedlicher Komplexität 102
61 Ablauf des Preprocessings für das Graphmatching 102
62 Strukturelle Ähnlichkeiten von Graphen in der Graphentheorie nach [S9, 422] 103
63 Ablauf des Ähnlichkeitsvergleichs und des Matchings der Graphen 104
64 Beispiele aus dem genutzten Datensatz der Getriebebaugruppen 108
65 Beispiele aus dem genutzten Datensatz der Getriebebaugruppen 110
66 Ablauf Modell für die Bauteilerkennung am Getriebestufen-Datensatz 112
67 Ergebnis der Baugruppensuche mit Skizzen als Input und Deep-Learning Bauteilerkennung 115
68 Beispiele aus dem genutzten Datensatz der Getriebebaugruppen nach [276] 116
69 Ergebnis der angepassten Entfernung der Bounding Boxen 117

70 Retrievalergebnisse für den Baugruppen-Datensatz [276] . 118
71 Übersicht des gesamten Vorgehens zur Ähnlichkeitssuche von Bauteilen, angelehnt an die Prozessabschnitte von Bustos et al. [252] nach [P10] 122
72 Vorgehen zur Ausrichtung der Bauteile nach dem Prinzip des CAD-Designs nach [P10] 124
73 Beispiele für die unterschiedlichen Bauteilausrichtungsverfahren nach [P10] 125
74 Optischer Vergleich der drei Methoden zur Bauteilorientierung und der entsprechenden Projektionsmatrizen nach [P10] 129
75 Vergleich der unterschiedlichen Bauteilausrichtungen nach [P10] 130
76 Ergebnis Heatmap für den Vergleich zwischen den Distanzmaßen und verschiedenen Projektionsarten und -zusammensetzungen 133
77 Verbesserung der Suchergebnisse durch Dimensionsreduktionsverfahren 135
78 Vergleich des CNN-Shape-Deskriptor-Ansatzes mit den direkten Matrix-Vergleichsalgorithmen nach [P9, P10] mit den Verfahren Korrelations-Koeffizient (corr2), der structural similarity Index (ssim), peak signal-to-noise-ratio (psnr) und der euklidische Distanz (l1) 137
79 Vergleich mit Stand der Technik Verfahren, Auszug aus Tabelle 13 138
80 Auflistung der Aufgabe und deren Randbedingungen für das Entwicklungsteam im präsentierten Szenario 141
81 Entwickelte Konzepte für die gestellte Szenario-Aufgabe . . 142
82 Ergebnis der Prinzipskizzensuche 143
83 Ergebnis der Baugruppensuche, mit Prinzipskizzen als Sucheingabe 145
84 Ergebnis der Bauteil-Ähnlichkeitssuche, mit dem Planetenträger als Sucheingabe 146
85 Beispiel für die Auswertung der AP für einen IoU von 0.75 . 157
86 Beispiel für die Auswertung der Ähnlichkeitssuche 158
87 Beispieldarstellung für die Anwendung von CNNs 159
88 Beispielberechnung der Convolutional Schicht mit einem 3x3 Filter 159
89 Ergebnis einer Max-Pooling-Operation für einer 4x4 Matrix 160
90 Das ROC-Diagramm zur Beurteilung der Klassifikationsergebnisse nach [129, 233, 234, 481] 164
100 Häufigkeit der Symbole im alten und neuen Ansatz zur Datenerzeugung nach [S5] 201

101 Überblick des Pointnet++ zur Bauteilerkennung nach [225] 204
102 Wahrheitsmatrix für den MSC-Score 211
103 Wahrheitsmatrix für den Jaccard-Score 212
104 Wahrheitsmatrix für den Eigenwert-Score 213
105 Wahrheitsmatrix für den GED-Score 214
106 Ergebnis für den Vergleich zwischen neuer Methode und bestehendem Verfahren [231] nach [P7] 226
107 Vergleich der Methode mit anderen aus dem MCB-Datensatz nach [P7] 227
108 Wahrheitsmatrix für den GED-Score 228
109 Darstellung der Regeln für die Reduktion der Planetenräder im Baugruppen-Graph nach [S9] 229
110 Wahrheitsmatrix für den MCS-Score 232
111 Wahrheitsmatrix für die Jaccard-Ähnlichkeit 233
112 Wahrheitsmatrix für den Eigenvektor-Score 234
113 Wahrheitsmatrix für den GED-Score 235
114 Beispiele für unterschiedlichen Sampling-Punkte und deren Auswirkung auf die erzeugten Matrizen nach [P10] 239
115 Zeitbedarf für die unterschiedlichen Sampling-Punkte für Punktewolken nach [P10] 239
116 Vorgehen zur Nutzung der ermittelten Label für das Training eines Siamese Networks 245

Tabellenverzeichnis

1 mAP50 Ergebniswerte für alle Modelle und Trainingsdatensätze mit unbekannten Daten mit den besten Ergebnissen fett und unterstrichen hervorgehoben nach [P6] 79
2 Ergebnismetriken für die neuen Trainingsdatensätze auf handgezeichneten Daten 83
3 Beispiel für die Kostendefinitionen der GED nach [429] . . 105
4 Out-of-Sample Genauigkeit der beiden trainierten Modelle 114
5 Ergebnisse für verschiedene vortrainierte Netze und euklidischer Distanz 131
6 Gleichungen für alle verschiedenen Feature-Vektor-Kombinationen 132
7 Übersicht der besten erzielten Suchergebnisse über alle getesteten Reduktionsverfahren 134
8 Sammlung aller Bauteilklassen des MCB-Datensatzes und der jeweiligen Bauteilanzahl nach [218] 161
9 Sammlung aller Bauteilklassen des ESB-Datensatzes und der jeweiligen Bauteilanzahl nach [220] 162
10 Übersicht der Auswertemetriken für Klassifikationsverfahren 163
11 Genauigkeit und F1-Score für den MCB-A-Datensatz nach [218] 164
12 Liste aller eingeordneten Methoden zur Ähnlichkeitssuche von Bauteilen 165
13 Liste aller Ergebnisse mittels des ESB-Datensatzes 174
15 Übersicht der Symbol-Instanzen im Datensatz der unbekannten Getriebestufen ohne angewendete Skalierung 189
16 Übersicht über die Symbolinstanzen im unbekannten Baugruppendatensatz 195
17 Bewertungsmetriken für alle Modelle und Trainingsdatensätze auf unbekannten Daten, wobei die besten Ergebnisse fett hervorgehoben und unterstrichen sind 200
18 Ergebnisse der Suche mit einer Handskizze als Input 202
19 Übersicht aller Baugruppen des Demonstratordatensatzes mit ihrer Nummer bestehend aus *Bauteilnummer - Getriebestufen - Bauteilanzahl* 205
20 Übersicht aller Baugruppen des Assembly-Datensatzes . . 215
21 Übersicht der angewendeten Regel für den Baugruppen-Datensatz 229
22 Ergebnisse der Dimensionsreduktionsstudie für unterschiedliche Verfahren 240

23 Übersicht der besten erzielten Suchergebnisse über alle getesteten Reduktionsverfahren 248

1 Einleitung

Über 80 % der verfügbaren Daten in Unternehmen liegen nach aktuellen Studien [1, 2] unstrukturiert vor und bieten dementsprechend ein hohes Nutzungspotential für die Betriebe. Allerdings werden davon nur 26 % automatisiert nutzbar gemacht und nur 58 % mehr als einmal nach der Erstellung wiederverwendet [1]. Zusätzlich geben 18 % der Unternehmen an, überhaupt Vorteile aus der Nutzung der vorhandenen Daten zu ziehen [3], wobei das Interesse daran zeitgleich zunimmt [4].

Diese generellen Aussagen belegen, dass das Potential zum Nutzen und Wiederverwenden vorhandener Daten hoch ist, auch in der Produktentwicklung [5], was vereinfacht in Bild 1 dargestellt ist. Allerdings zeigt die Studie [6], dass eine erfolgreiche Umsetzung in der Produktentwicklung noch nicht stattgefunden hat, da 32 % der Arbeitszeit von Ingenieuren mit „non-value work“ verbracht wird. Dabei sind 63 % dieser „wertlosen“ Arbeit direkt mit der Suche, Nacherzeugung oder Sammlung vorhandener Daten verknüpft. Dementsprechend sehen auch 29 % der befragten Unternehmen aus der Studie [7] eine Herausforderung bei der Wiederverwendung vorhandener Daten in einem Produktdaten-Management (PDM) System. Folglich haben auch PDM-Betreiber wie Autodesk einen eigenen Leitfaden [8] für die Wiederverwendung existierender Konstruktionen entwickelt. Aber selbst aktuelle Beispiele aus der Formel 1 [9] zeigen, dass es noch Defizite in der Umsetzung gibt, da in diesen Fällen die Verwaltung der Fahrzeug-Bauteile über eine Excel-Liste erfolgte.

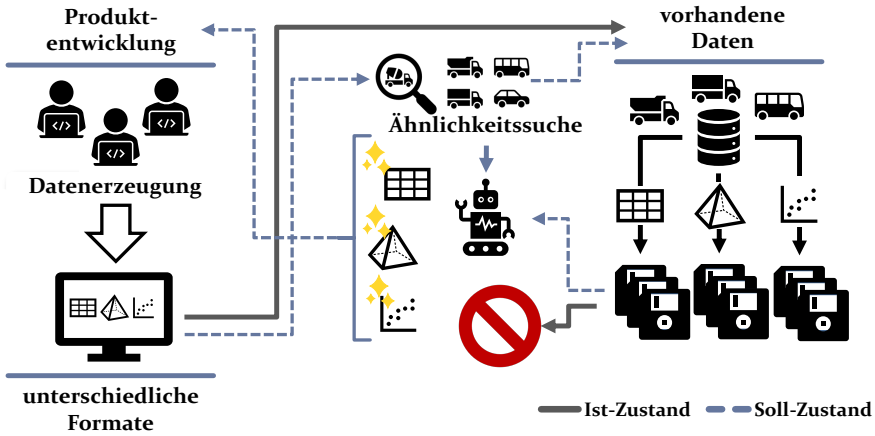


Bild 1: Ist- und Soll-Zustand für die Nutzung vorhandener Daten in der Produktentwicklung

Die Herausforderungen entstehen vor allem durch die erhöhte Komplexität der Produkte [10] und somit großen Datenmengen und unterschiedlichen Formaten, die im Laufe der Produktentwicklung anfallen [11]. Die Vielzahl an Formaten und Datentypen aus dem Produktentwicklungsprozess, wie beispielsweise Lasten- oder Pflichtenheft, Bewertungen, Analysen, Bilder, Zeichnungen, Skizzen, Versuchsergebnisse, 3D-Geometrien, Oberflächenscans, Simulationen, Videos oder Audio-Aufnahmen, macht die Umsetzung einer automatisierten Suche vorhandener Daten besonders schwierig.

1.1 Motivation und Problemstellung

Damit diese Herausforderungen adressiert und gelöst werden können, müssen neue Vorgehensmodelle zur automatisierten Nutzung der vorhandenen Daten entwickelt werden. Der Einfluss auf die Produktqualität und -kosten ist in den frühen Phasen besonders hoch [12, 13], weshalb in diesen ein Zugriff auf die bereits bestehenden Daten besonders gewinnbringend ist. Zudem kommen die Methoden und Werkzeuge der virtuellen Produktentwicklung stark in den frühen Phasen zum Einsatz [14], wodurch der Fokus dieser Arbeit auf deren Ergebnissen liegt. Durch eine weitere Verknüpfung der Entwicklungsergebnisse, beispielsweise einer Baugruppe und der zugehörigen Betriebsanleitung, können die Erzeugnisse der späteren Phasen bereits früher verwendet werden und somit Kosten und Aufwand einsparen.

Allerdings sind der notwendige manuelle Input und die Datenaufbereitung eine Hürde für die Anwendbarkeit und produktiven Einsatz der Ähnlichkeitssuche in der Produktentwicklung. Dementsprechend ergibt sich das Ziel, die vorhandenen Daten möglichst automatisiert und ohne weitere händische Aufbereitung zu finden und weiterzuverwenden. Ein elementarer Bestandteil der Suche ist die Eingabe, wobei zunächst das Format des Suchinputs definiert werden muss. Im speziellen Kontext der Produktentwicklungs-Formate (z. B. Geometrie) sind vor allem textbasierte Lösungen zu fehleranfällig [15], weshalb in dieser Arbeit zunächst unimodale Lösungen betrachtet werden, die mit dem gleichen Such- und Ergebnisformat arbeiten.

Dadurch wird die Zuverlässigkeit der jeweiligen Entwicklungsphase erhöht und auch beschleunigt, da bereits bewährte und ausgearbeitete Lösungen, zusammen mit dazugehörigen weiteren Dokumenten, direkt bereitgestellt werden [16]. Allgemein setzen sich neue Produkte meist aus bekannten Geometrieelementen oder Standardbauteilen zusammen, weshalb die Wiederverwendung in der Produktentwicklung besonders vorteilhaft ist. Weiterhin kann der Problemlösungsprozess in der Produktentwicklung verbessert werden, da schnell viele Varianten gefunden und so spezifische Eigenschaften für das neue Produkt übernommen werden können.

Diese Vorteile bestätigt auch die Aussage eines Ingenieurs (John Burrill, Transpo Electronics) aus der Studie [17], welcher eine 30 %ige Einsparung der Entwicklungszeit für neue Produkte bestimmt hat, wenn vorhandene Entwicklungsdaten verwendet werden können. Die Ersparnis kann sogar auf 80 % anwachsen, wenn eine starke Wiederverwendung der Daten möglich ist. Unabhängig davon ist die Suche nach ähnlichen Daten immer das Fundament für eine erfolgreiche Umsetzung [18]. Im Kontext des gesamten Produktlebenszyklus wird diese Art der Wiederverwendung als Typ II bezeichnet [19], in Abgrenzung an das Recycling und Wiederverwenden von gefertigten Produkten oder Materialien (Typ I).

Somit sind die Vorteile und Potentiale für die Ähnlichkeitssuche deutlich, weshalb auch viele Methoden im Laufe der Zeit entwickelt wurden. Allerdings basieren diese häufig auf Regeln und starren Vorgaben, weshalb sie nicht gut adaptierbar sind oder gewisse Datenarten gar nicht verwenden können. Das Ziel dieser Arbeit ist deshalb, das Konzept der Ähnlichkeitssuche mit datengetriebenen Methoden zu verknüpfen, damit neue Arten der Suche möglich oder bestehende Prinzipien verbessert werden. Die Nutzung von datengetriebenen Methoden im Rahmen der Produktentwicklung [10, 20, 21] ist ein breites Themenfeld, das häufig auch als Digital Engineering bezeichnet wird. Eine Studie [22] aus dem Jahr 2019 legt offen, dass 58 % der befragten Produktentwicklungs-Unternehmen keine Künstliche Intelligenz (KI) Methoden einsetzen, und nur 5 % gaben an, diese Verfahren umfassend zu nutzen. Der Einsatz dieser neuen Methoden ist auf vielen Ebenen herausfordernd [10], beispielsweise durch die Art der Produktentwicklung [23], die Formate und Datentypen [11, 24] oder die Komplexität der Produkte [25]. Im Rahmen dieser Arbeit sollen die neuen Algorithmen vor allem für die Umsetzbarkeit und eine möglichst hohe Automatisierung eingesetzt werden, damit die vorhandenen Daten möglichst gut nutzbar werden.

1.2 Aufgabenstellung und Aufbau der Arbeit

Die Aufgabenstellung dieser Arbeit ist daher die Entwicklung eines Vorgehensmodells zur automatisierten Nutzbarmachung vorhandener Daten in den frühen Phasen der Produktentwicklung. Dabei sollen datengetriebenen Methoden in die neuen Ansätze integriert werden, damit weniger händischer Input für die Ähnlichkeitssuchen notwendig ist. Der Fokus liegt auf den frühen Phasen, weshalb Konzept- und Entwurfsphase mit den jeweiligen Ergebnissen, Prinzipskizze und Bauteilgeometrie, besondere Beachtung finden.

Für die Bearbeitung der Aufgabenstellung soll zunächst der vorhandene Stand der Forschung für den Produktentwicklungsprozess und datengetriebene

Methoden erarbeitet und ausgewertet werden. Anschließend wird in Anlehnung an den chronologischen Ablauf des Produktenwicklungsprozesses [26, 27] zunächst eine Methode für die Ähnlichkeitssuche von Prinzipskizzen entwickelt und mittels eines Demonstratordatensatzes bewertet. Auf Basis dieser Ergebnisse soll im anschließenden Kapitel die Verknüpfung zu 3D-Baugruppen erfolgen, indem vorhandene Baugruppen automatisiert transformiert und nachfolgend mit der Prinzipskizze als Input gesucht werden können. Danach soll im Sinne der steigenden Granularität im Entwicklungsprozess der Feinentwurf von Bauteilen unterstützt werden, indem ein Ansatz für die Suche von Bauteilen präsentiert wird. Abschließend soll ein Anwendungsszenario die Module nacheinander durchlaufen, um die Vorteile der Anwendung im Entwicklungsprozess darzulegen. Ein gesamter Überblick der Kapitel und zugehörigen Seitenzahlen ist in Bild 2 dargestellt.

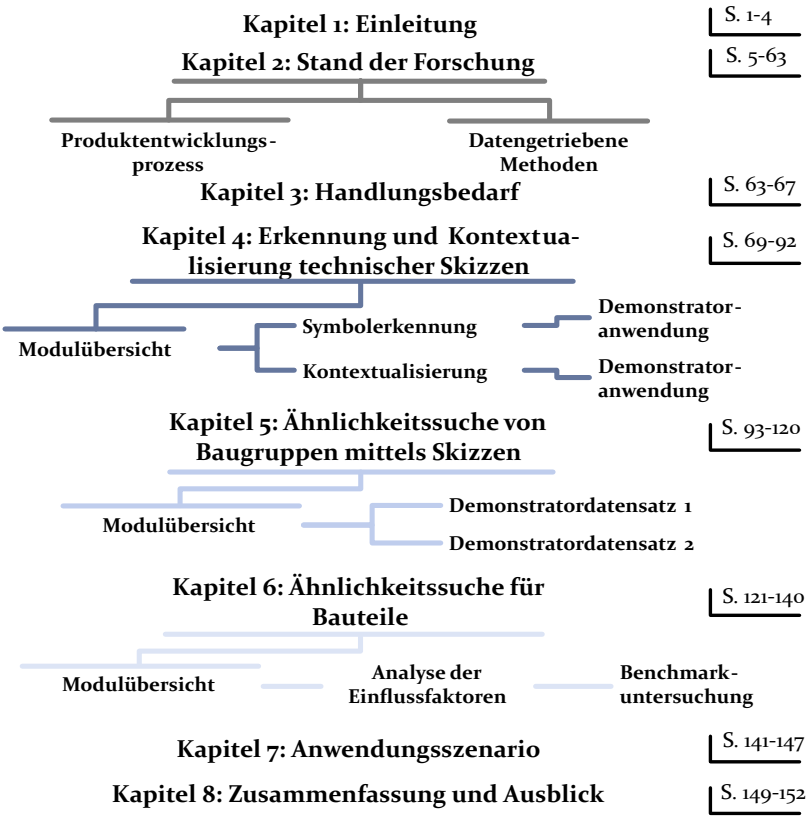


Bild 2: Übersicht des inhaltlichen Aufbaus der Arbeit

2 Stand der Forschung

Der Stand der Forschung zu den Themen Produktentwicklung und datengetriebenen Methoden wird zunächst in diesem Kapitel genauer präsentiert. Im ersten Schritt wird der Produktentwicklungsprozess mit speziellem Fokus auf die vorhandenen Datenarten vorgestellt. Darauf aufbauend werden für die in den frühen Phasen relevanten Produktdarstellungsformate (Bilder und Geometrie) die bereits vorhandenen Algorithmen und Methoden dargelegt. Besonders die aktuell schon umgesetzten Ansätze der diversen datengetriebenen Methoden für die Produktentwicklung werden jeweils am Ende der Unterkapitel präsentiert.

2.1 Produktentwicklungsprozess

Die Entwicklung technischer Produkte ist sehr herausfordernd und ohne strukturiertes Vorgehen nicht praktikabel umsetzbar. Daher sind über viele Jahrzehnte Vorgehensmodelle entwickelt worden, welche den aufwendigen Vorgang in logische Elemente unterteilen. Die Notwendigkeit für diese Modelle ergeben sich aus der steigenden Produktkomplexität, den globalen Trends und Entwicklungen sowie der schwierigen Planbarkeit [28]. Diese Vorgehensmodelle sind als grober Leitfaden zu verstehen, der eine Struktur für den Entwicklungsprozess an die Hand gibt. Dementsprechend ist eine scharfe Trennung der Phasen nicht immer möglich und eine iterative Bearbeitung mit Rückgriffen die Regel [13, 26]. Eines der bekanntesten Modelle ist das Vorgehen von Pahl/Beitz nach [13, 29]. Dieses gliedert sich in die vier Bereiche *Planen*, *Konzipieren*, *Entwerfen* und *Ausarbeiten*.

Im ersten Schritt, dem Planen und Klären der Aufgabe, müssen die Anforderungen über das zu entstehende Produkt gesammelt und festgelegt werden. Diese bilden die Grundlage für alle nachfolgenden Bewertungen der Ergebniserfüllung. Die Informationen für die Anforderungsliste können über verschiedene Kanäle bezogen werden, beispielsweise durch direkten Kundenkontakt, Marketingumfragen oder vorherige Produktgenerationen [13, 30–32].

Die Anforderungen sind die Grundlage für die nachfolgende Konzeptphase, in welcher die entsprechenden Funktionen für das Produkt abgeleitet und in erste prinzipielle Lösungskonzepte überführt werden. Dazu wird die Aufgabenstellung in Funktionsstrukturen umgewandelt, welche im Anschluss auf Basis von natürlichen Effekten (z. B. physikalische, chemische oder elektrische) in die Wirkstrukturen überführt werden. Am Ende des Prozessschrittes muss eine Auswahl von weiterzuverfolgenden Lösungskonzepten ermittelt

werden, aufbauend auf Bewertungsergebnissen, welche durch verschiedene Methoden erstellt werden können.

In der anschließenden Entwurfsphase wird das ermittelte Lösungskonzept in eine konkrete Produktgestalt umgewandelt. Dabei werden virtuelle Werkzeuge wie Computer-Aided-Design (CAD)-Modelle und Simulationsmethoden verwendet. Zum Abschluss soll wiederum eine technische Bewertung der erstellten Varianten erfolgen, mit dem Ziel, einen endgültigen Gesamtentwurf zu definieren.

Im letzten Schritt wird der Entwurf soweit ausgearbeitet, dass eine Herstellung des Produkts möglich ist. Dies umfasst u. a. die Definition der Form, Festlegung aller Einzelteile, Materialien, Toleranzen und Fertigungsarten.

Neben dem kurz dargelegten Vorgehen nach Pahl/Beitz sind im Laufe der Zeit weitere Vorgehensweisen entwickelt und veröffentlicht worden. Etablierte Beispiele sind hier die VDI 2206 (V-Modell) [33], die VDI 2221/23 [26, 27], Pahl/Beitz [13], das Axiomatic Design von Suh [34], die Konstruktionskataloge von Roth [35–37], sowie die Modelle von Rodenacker [38] und Koller [39].

Bei der Untersuchung der diversen Verfahren fällt auf, dass die allgemeine Gliederung in verschiedene Phasen in allen Modellen wiedererkennbar ist. Das grundsätzliche Vorgehen, die gegebenen Anforderungen in Funktionen zu überführen, anschließend auf die Grundstrukturen zu übertragen und danach wieder in ein Gesamtsystem zu überführen, ist in allen Modellen vorhanden. Das V-Modell der VDI 2206 nimmt dabei noch eine Sonderstellung ein, da es sich um ein Modell für die Erzeugung von mechatronischen Produkten handelt.

Die genannten Abläufe beschreiben alle präskriptive Modelle, also Vorgehen, die auf systematischen Vorgehensschritten mit Phasen und Aktivitäten aufbauen [40]. Das Gegenstück dazu stellen deskriptive Modelle dar, welche sich auf Erfahrungen aus dem Entwicklungsprozess stützen und teilweise als Vorstufe zu präskriptiven Modelle fungieren [41]. Andere Arten zur Unterscheidung von Designprozessen sind der Fokus auf Phasen oder Aktivitäten, Problem oder Lösungsorientierung und das Abstraktionslevel der Betrachtung [42].

Als Konsensmodell wird dabei die VDI 2221/23 [26, 27] bezeichnet, da diese auf den vorherigen Modelle (u. a. Pahl/Beitz [13], Hubka [43], Hall [44] und Pugh [45]) aufbaut und in ein Modell vereint [46]. Daher wird für die nachfolgende genaue Betrachtung des Entwicklungsprozesses die VDI 2221/23 als Vorgehensmodell herangezogen.

2.1.1 VDI 2221/23

Die Grundlagen der heutigen VDI-Norm für die Produktentwicklung sind in den 1970er-Jahren gelegt worden und stetig an den aktuellen Stand der Technologien und Forschung angepasst worden. Die aktuellste Version aus dem Jahr 2019 versucht die unterschiedlichen Vorgehensmodelle u. a. [47–51] aus dem Forschungsumfeld zu vereinen und richtet daher den Fokus verstärkt auf die Prozesse in der Produktentwicklung. Dabei wird die VDI 2221 [26] als Kernrichtlinie verstanden mit zusätzlichen Details und Erläuterung in den weiterführenden Richtlinien VDI 2222 [52], 2223 [27] und 2206 [33]. Eine Gesamtübersicht des Vorgehensmodells nach der VDI 2221 ist in Bild 3 dargestellt mit dem schematischen Prozessplan auf der linken Seite. Auf der gegenüberliegenden Seite sind den jeweiligen Entwicklungsaktivitäten die passenden Datenformate zugeordnet [24, 53, 54]. Diese sind generell anhand ihrer Dimension unterteilt, wobei eindimensionale Erzeugnisse Texte, Bewertungen und Berechnungen umfassen. Im Gegensatz dazu sind unter den zweidimensionalen Formaten primär Bilder und Strukturdiagramme eingeordnet. Die letzte Klassen bilden die 3D-Formate, was in dem Umfeld hauptsächlich 3D-CAD-Geometrie und FE-Netze sind. Andere mögliche Arten wie Videos oder Audioformate sind nicht weiter betrachtet worden.

Ablauf
Der initiale Prozessschritt im Ablauf ist das *Klären und Präzisieren der Aufgabenstellung*. In dieser Aktivität gilt es die gestellten Ziele und Anforderungen zu spezifizieren und gegebenenfalls nachzubessern. Dazu können weitere Informationen eingeholt werden, um die Aufgabe präziser zu formulieren und somit die Planung der nachfolgenden Schritte zu vereinfachen. Die Anforderungen können in diesem Schritt auf die neue Informationslage angepasst werden, was auch für alle nachfolgenden Aktivitäten gilt. Dies gewährleistet einen immer aktuellen Vergleich zu den gestellten Zielen.

Der nächste wichtige Schritt ist das *Ermitteln von Funktionen und deren Strukturen*. In Abhängigkeit der Produktkomplexität ist eine Beschreibung der Funktionen über eine einfache Objekt-Verb-Form möglich, bei komplexeren Produkten ist dagegen eine Übertragung in Funktionsgraphen oder Matrizen notwendig. Für die Umsetzung können Unified Modeling Language (UML) oder Systems Modeling Language (SysML)-Diagramme [55], Funktionsbäume [56], IDEF0 (Integration Definition for Process Modeling) [57] oder FAST (Functional Analysis System Technique)-Diagramme [58, 59] verwendet werden.

Die ermittelten Funktionen werden danach mit jeweils passenden Lösungsprinzipien kombiniert, was in dem Ablaufpunkt *Suchen nach Lösungsprinzipien und deren Strukturen* zusammengefasst wird. Dabei werden den Teilfunk-

tionen Wirkprinzipien zugeordnet, was zu einem Gesamtlösungskonzept verknüpft werden kann. In der Umsetzung können diskursive Methoden und Leitfäden verwendet werden, welche bei der Zuordnung und Findung des Lösungsprinzips unterstützen sollen. Beispiele sind der morphologische Kasten [60], die Nutzung von Konstruktionskatalogen [35–37, 61] oder die Analyse der physikalischen Zusammenhänge [62]. Aber auch Kreativitätstechniken können für die Lösungsfindung genutzt werden, beispielsweise das Brainstroming bzw. Mind Map [63], die 6-3-5 Methode [64] oder das Delphi-Verfahren [65]. Unabhängig von dem Vorgehen zur Lösungsidentifikation sollten nach dem Durchlauf der Phase mehrere prinzipielle Lösungen für die anschließende Bewertung entwickelt worden sein. Die Dokumention dieser Lösungen kann als Prinzipskizze, Diagramm, verbale Beschreibung oder Zeichnung vorliegen.

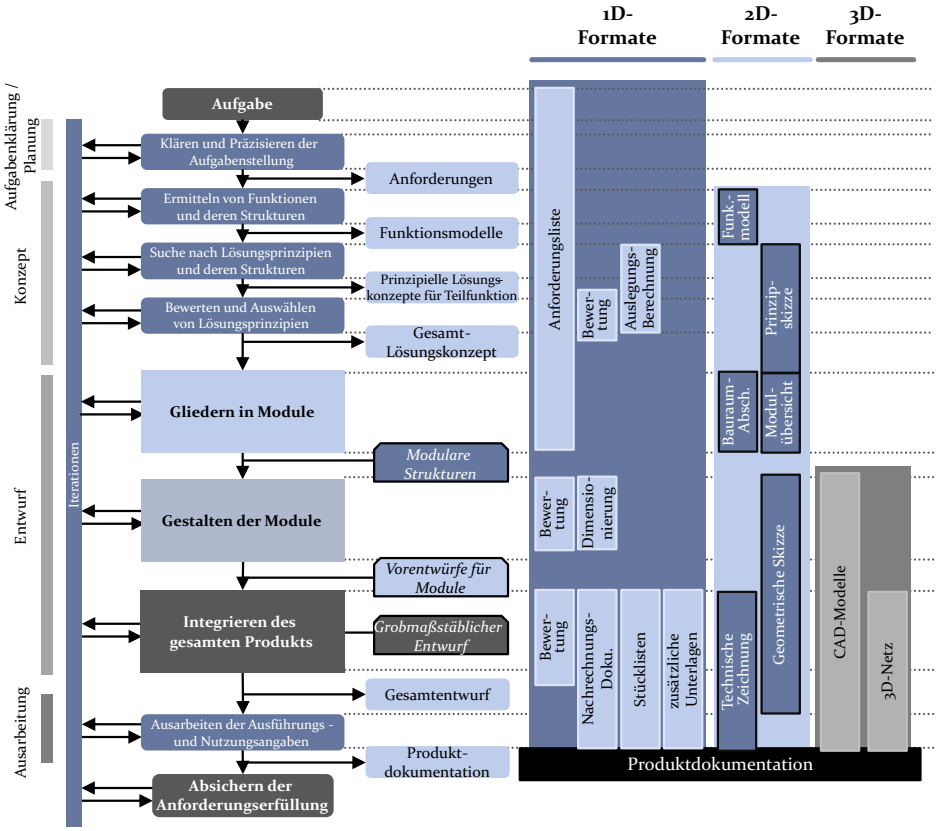


Bild 3: Übersicht der unterschiedlichen Formate im Produktentwicklungsprozess nach [24, 26, 27, 66, P1]

Im darauffolgenden Schritt *Bewerten und Auswählen des Lösungskonzepts* müssen die erzeugten Konzepte anhand der Anforderungen bewertet werden. Das Ziel ist das Lösungskonzept mit dem höchsten Potential zu ermitteln und nachfolgend weiter auszuarbeiten, da eine parallele Bearbeitung mehrerer Konzepte meist nicht umsetzbar ist. Dazu müssen geeignete Bewertungskriterien definiert und mittels verschiedener Methoden angewendet werden. Zur Anwendung können, abhängig vom Ziel und der Komplexität der Bewertung, beispielsweise eine Beurteilung der technischen Wertigkeit [67], ABC-Analyse [68], SWOT-Analyse [69] oder Nutzwertanalyse [70] kommen.

Das ausgewählte Konzept muss im nächsten Prozessschritt *Gliedern in Module - Schnittstellendefinition* in verarbeitbare Abschnitte unterteilt werden. Die Einordnung kann anhand der Ingenieurdomäne (Mechanik, Elektronik, Software, Verfahrenstechnik usw.) getroffen werden, wobei die Schnittstellen zwischen den Modulen klar definiert sein müssen. Durch die Aufgliederung in Gruppen sind die nachfolgenden Schritte parallel für die verschiedenen Abteilungen durchführbar. Dieser und die nachfolgenden Ablaufpunkte sind ausführlich in [27] erläutert.

Nach der Einordnung in Module werden diese durch den Schritt *Gestalten der Module* weiter konkretisiert und in virtuelle Modelle überführt. Dies umfasst vor allem die Erzeugung von CAD-Modellen, welche die Geometrie des Produktes repräsentieren.

Die Zusammenführung der unterschiedlichen Module erfolgt im Schritt *Integrieren des gesamten Produkts*. Dabei werden die gestalterischen Freiheiten immer weiter reduziert, was in einem Gesamtentwurf für das Produkt resultiert. Dieser kann CAD-Modelle, Stücklisten, Zeichnungen oder Schaltdiagramme umfassen.

Die Ergebnisse bilden auch die Grundlage für den nächsten Schritt *Ausarbeiten der Ausführungs- und Nutzungsangaben*. Der Fokus liegt vor allem auf der Bereitstellung von Produktdokumentionen für die Fertigung und Nutzung des Produktes. Dies beinhaltet beispielsweise Anleitungen, Fertigungstoleranzen, Code-Dokumention, Montagevorschriften oder Recycling-Informationen. Nachdem alle Schritte durchgeführt wurden und das fertige Produkt entwickelt wurde, erfolgt im letzten Abschnitt das *Absichern der Anforderungserfüllung*. Dieser Schritt ist natürlich in allen vorherigen Phasen auch Teil des Entwicklungsprozesses, wird aber am Ende nochmals zur finalen Absicherung des Produktes angewendet. Dabei können virtuelle Methoden, wie die Finite Element- oder Mehrkörpersimulation, oder reale Versuche zur Anwendung kommen.

Für den gesamten Entwicklungsprozess gilt auch, dass ein so strukturiertes Vorgehen nur selten in der Realität umsetzbar ist. Daher ist im Prozess vorgesehen, dass Iterationen durchlaufen werden oder spezifische Schritte mehrfach wiederholt werden können. Zudem können Querschnittsthemen in der Produktentwicklung, die Auswirkungen auf alle Schritten haben allgegenwärtig, beispielsweise das Design for X, neue Technologien oder Vorgaben (z. B. Manufacturing X [71]), Qualitätsmanagement oder Varianten bzw. Innovationsmanagement [26].

Nach der Vorstellung des methodischen Vorgehens werden im nächsten Kapitel die verwendeten Formate der einzelnen Schritte näher analysiert und vorgestellt.

2.1.2 Formate

Die Auflistung der Datenformate im Entwicklungsprozess aus Bild 3 stellt dabei eine generelle Übersicht der jeweiligen Kernformate der Entwicklungsphase dar. Natürlich können viele Formate in allen Phasen angewendet werden, allerdings liegt bei der Darstellung im Bild der Fokus auf dem Hauptbeitrag zur Produktentwicklung. Die Einordnung der Formate basiert auf der Dimension und setzt sich aus drei Hauptkategorien zusammen, abgeleitet aus den Klassifizierungen von [24, 53, 66, 72, 73]. Die generelle Einordnung mit jeweils einem Beispielformat ist in Bild 4 illustriert.

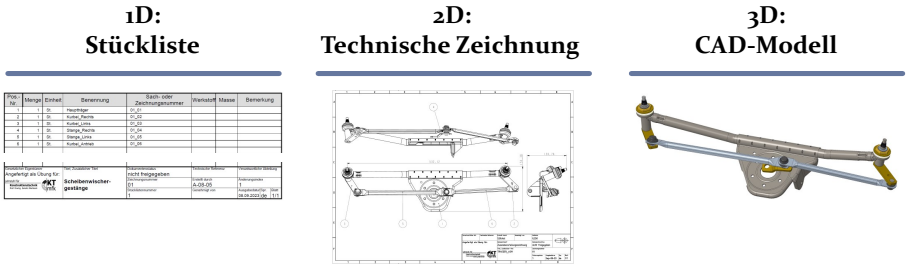


Bild 4: Beispiele für die unterschiedliche Dimensionalität von Formaten in der Produktentwicklung am Beispiel eines Scheibenwischergestänges

Weitere Klassifizierungen von Produktmodellen im Verlauf des Entwicklungszyklus sind in [72] aufgeführt. Die oben genannten Kategorien werden in den nächsten Kapitel ausführlich vorgestellt.

2.1.2.1 Eindimensionale Formate

Unter die Kategorie *Text* fallen in den frühen Phasen vor allem die Anforderungsliste, Kundenfeedback, Pflichten- und Lastenhefte. Weiterhin sind auch

alle Berechnungen hierunter zusammengefasst, entweder für die Dimensionierung oder die Nachrechnung bei der finalen Absicherung. In den späteren Phasen sind Stücklisten, Betriebsanleitungen, Handbücher, Fertigungs- bzw. Montageinformationen dieser Kategorie zuzuordnen. Ebenfalls sind die stetigen Bewertungen unterschiedlicher Konzepte und Lösungen mittels Tabellen oder Matrizen in dieser Klasse verortet. Viele Text-Dateien sind auch durchgängig im Entwicklungsprozess notwendig und können zu keiner spezifischen Phase eingegliedert werden, beispielsweise Protokolle, Formulare, Checklisten, Emails oder Konstruktionsrichtlinien.

2.1.2.2 Zweidimensionale Formate

Eine weitere wichtige Formatklasse sind die *Bilder* im Entwicklungsablauf. Dabei treten neben technischen Zeichnungen und Skizzen auch Graphen, Fotos oder CAD-Ansichten auf. In der Konzept- und Entwurfphase sind vor allem einfache Abbildungen des groben Produktentwurfs mit Hilfe von Bildern dargestellt. Später unterstützen Bilder die Erzeugung der 3D-Geometrie und sind bei der Produktdokumention relevant. In vielen Domänen sind technische Zeichnungen immer noch das rechtlich bindende Dokument für ein Produkt, beispielsweise bei der Auftragsvergabe an externe Dienstleister. Die technische Zeichnung beinhaltet viele eigene Darstellungsregeln und Zusatzinformationen, welche in der DIN-Norm 199 [74] festgehalten werden.

Im Rahmen der Arbeit liegt der Fokus vor allem auf den Prinzipskizzen aus den frühen Phasen, weshalb diese genauer vorgestellt werden.

Skizzen
Nahezu jede technische Disziplin hat eine eigene Art, wichtige Informationen und Wissen durch Grafiken, Diagramme oder Skizzen darzustellen [75]. In Bild 5 ist ein kurzer Überblick der verschiedene Skizzen dargestellt, sortiert nach der jeweiligen technischen Domäne. Diese Zeichnungen und Skizzen können sich in Format, Größe, Komplexität und in der Art ihrer Erstellung unterscheiden.

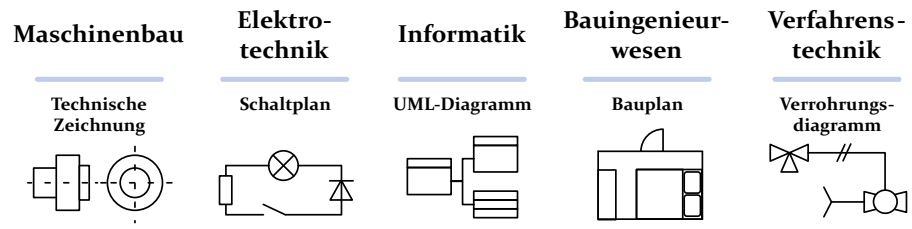


Bild 5: Beispiele für verschiedene Skizzen aus der Ingenieurdomäne

Eine der frühsten bildlichen Darstellungsformen im Entwicklungsprozess ist die Prinzipskizze, welche eine Repräsentation der prinzipiellen Lösung darstellt. Das Ziel ist die Grundfunktionen grafisch abzubilden, um somit einen guten Überblick über das erzeugte Konzept zu erhalten [76, 77]. Die genaue Art der Umsetzung ist dabei nicht definiert. So können weitgehend genormte symbolische Skizzen, freie Strichskizzen als Abstraktion realer Strukturen, 3D-Freihandskizzen oder nicht-maßstäbliche Grobentwürfe zur Erstellung genutzt werden [52]. Eine Sammlung von Symbolen für diverse Prinzipskizzen ist in [77–80] aufgeführt.

Bild 6 zeigt ein Beispiel für eine Prinzipskizze, welche zudem den Unterschied zwischen einer Prinzipskizze und einer technischen Zeichnung verdeutlicht. Die Skizze veranschaulicht einen Mechanismus, der eine Translationsbewegung in eine Rotation umwandelt, wogegen die technische Zeichnung die detaillierte Konstruktion des Bauteils zeigt. Es besteht aus einem Kolben (c) und einem passenden Gehäuse, das als Führung für den Kolben dient (d). Der Kolben ist über ein Pleuel (a) mit einem Lager (b) verbunden. In Bild 6 ist neben der Prinzipskizze das fertige Pleuel dargestellt. Dieser Vergleich zeigt, wie sehr die Bauteile in Prinzipskizzen vereinfacht sind. Durch die reduzierte Darstellung von Zusammenhängen bieten diese Repräsentationsform große Vorteile, beispielsweise durch die schnelle Erstellung von Varianten mit unterschiedlichen Lösungsprinzipien.

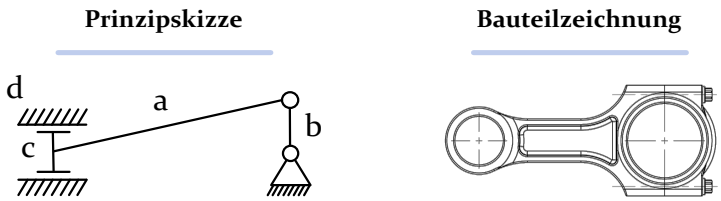


Bild 6: Beispiel für eine Prinzipskizze und die daraus resultierende technische Zeichnung eines entworfenen Bauteils nach [36]

2.1.2.3 Dreidimensionale Formate

Ein Hauptergebnis der Produktentwicklung ist das virtuelle Modell der Produktgeometrie. Durch den großen Einfluss ist es sinnvoll, die Geometrieerzeugung in ein separates System, das CAD-Programm, zu verlagern. Diese Software-Tools bieten viele spezifische Vorteile für die Produktentwicklung, zum Beispiel parametrische Modelle oder die automatische Integration von wissensbasierten Konstruktionselementen.

Das erzeugte 3D-Modell ist die Grundlage für alle nachfolgenden Analysen und Simulationen z. B. Finite-Elemente-, Mehrkörper-, Montage- bzw. Produktionssimulationen oder Wartungs- bzw. Betriebsanalysen [81]. CAD-Modelle werden für diesen Zweck als Volumenmodelle aufgebaut, um das eigentliche Bauteil möglichst genau abzubilden und bestmögliche Simulationsergebnisse zu erzielen. Neben der reinen Geometrie sind weitere Informationen über das Produkt im Modell hinterlegt, zum Beispiel Material, Fertigungs- oder Toleranzangaben.

Generell lassen sich 3D-Modelle in unterschiedliche Repräsentationsformate unterteilen: Kanten-, Flächen-, Volumen- und Zellenmodell [80, 82]. Eine grobe Übersicht mit Beispielbildern ist in Bild 7 dargestellt und wird im Anschluss genauer erläutert.

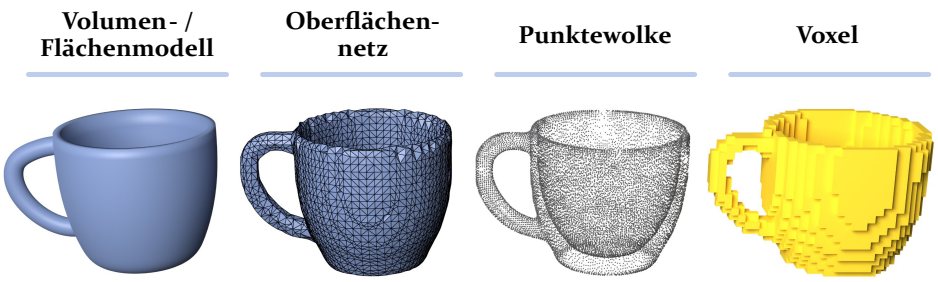


Bild 7: Darstellung der verschiedenen Geometrieformate in der Produktentwicklung nach [82]

Punktewolke
Eine einfache und effiziente Art der Geometrie-Repräsentation sind Punktewolken, welche sich aus mehreren Punkten im 3D-Raum zusammensetzen. Sie kommen vor allem beim 3D-Scannen von Objekten durch Reverse Engineering oder in Kombination mit LIDAR-Sensoren im Rahmen des autonomen Fahrens vor [83, 84]. Auch im Bereich der Metrologie werden Geometrieobjekte häufig durch Punktewolken dargestellt, wobei diese auf geometrische Abweichungen gegenüber der Idealform untersucht werden können. Punktewolken zeichnen sich durch ihren ungeordneten Charakter und ihre einfache Beschreibung aus, da nur die Knotennummer und die drei Koordinatenwerte notwendig sind, um die Geometrie hinreichend zu definieren [85].

Oberflächennetze
Im Vergleich zu Punktewolken sind Oberflächennetze der ursprünglichen CAD-Geometrie sehr ähnlich, da sie die Form mit Polygonen reproduzieren und dabei teilweise vereinfachen. Diese Art von Netzen wird häufig im Reverse Engineering oder für den 3D-Druck verwendet [81]. Im Vergleich

zu Flächen- oder Volumenmodellen, benötigen Netze weniger Speicher und Grafikleistung und eignen sich daher für Visualisierungsaufgaben. Diese Eigenschaft wird zusätzlich bei Zulieferteilen in einem großen Baugruppenmodell oder bei der Präsentation eines Produkts in der Virtuellen Realität (VR) genutzt. Die einfachsten und am häufigsten eingesetzten Netze sind aus Dreiecken zusammengesetzte Oberflächennetze, welche im Preprocessing für den 3D-Druck eine wichtige Rolle spielen, da sich diese Modelle relativ einfach in Schichten umwandeln lassen.

Flächenmodell
Eine Erweiterung der Oberflächennetze sind Flächenmodelle, welche ebenfalls nur aus Flächenbeschreibungen bestehen, allerdings nicht über eine approximierende Tesselierung. Sie bestehen meist aus 2D-Konturen oder 3D-Skizzen, welche über verschiedene Operationen (Rotation, Zug oder Extrusion) in Flächen übertragen werden [81, 86]. Grundsätzlich ist keine Unterscheidung bei der erzeugten Geometrie möglich, welche Flächen „innen" oder „außen" liegen. Die hauptsächliche Anwendung von Flächenmodellen sind Bauteile, deren Form nicht analytisch beschreibbar ist. Beispiele sind das Exterieur einer Fahrzeug-Karosserie, Design-Objekte, aerodynamisch optimierte Produkte oder wenn ein Volumenmodell nicht sinnvoll ist, z. B. bei Blechbauteilen [81]. Für die Modellierung können Freiformflächen zum Einsatz kommen, welche häufig mit speziellen Kurvenalgorithmen erzeugt werden. Bekannte Vertreter sind die Bézier- [87], Hermite-, B-Spline [88] oder NURBS (non uniform rational B-Splines) Kurven [89].

Volumenmodell
Im Gegensatz zu Flächenmodellen bzw. Oberflächennetzen haben Volumenmodelle einen geschlossenen Flächenverbund [81]. Dem Namen entsprechend besitzen die Modelle ein Volumen und sind nur die „Hülle" einer Geometrie. Volumenmodelle werden in unterschiedliche Arten eingeordnet [89]: Boundary-Representation (BRep) [90], Constructive Solid Geometry (CSG) und Hybrid-Modelle.

CSG-Modelle erzeugen 3D-Modelle unter Verwendung boolescher Operationen (Addition, Ausschneiden, Schnittmenge usw.) zur Manipulation des Basismodells. Im Gegensatz dazu beschreiben BRep-Modelle das Bauteil durch die Begrenzungsflächen und angrenzenden Kanten. Volumen CAD-Modelle werden in der Regel in ihrem proprietären Programmformat oder in allgemeinen Austauschformaten wie STEP [91], IGES [92] oder JT [93] gespeichert [94]. Neben den Geometriedaten können noch viele weitere Produktinformationen an dem Modell gespeichert werden, beispielsweise Material- oder Fertigungsinformationen [81].

Voxel
Die letzte Geometriedarstellung wird Voxel genannt und der Name setzt sich aus dem Akronym „vo“ für Volumen und „el“ für Element zusammen. Es handelt sich um ein räumliches 3D-Gitter, das die Geometrie in einem definierten Raum abbildet. Ein Punkt in diesem Raster wird als Voxel bezeichnet, vergleichbar mit einem Pixel in einem Bild. Das Format bietet den Vorteil, dass die Geometrie gleichmäßig in die Gitterdarstellung übertragen wird. Dadurch sind Voxel immer in einer einheitlichen Größe und daher sehr strukturiert, allerdings leidet folglich die Präzision der geometrischen Darstellung. Voxel werden häufig für Topologie-Optimierung in der Produktentwicklung eingesetzt oder in medizinischen Anwendungen, zum Beispiel bei MRT/CT-Scans.

Diese vorgestellten Darstellungsarten können im Kontext der Produktentwicklung grob in zwei Kategorien eingeteilt werden, den einfacheren und allgemeinen Geometrieformaten (Punktewolke, Voxel, Oberflächenetze) und CAD-spezifischen Formaten (Volumen- und Flächenmodelle). Die erste Kategorie kann aus den CAD-Modellen abgeleitet werden und hat einige Vor- und Nachteile gegenüber dem proprietären CAD-Format. Vor allem die Editierbarkeit der Geometrie leidet stark unter der Transformation. Eine Änderung der Geometrie ist zwar möglich, aber nur mit hohem Aufwand und nicht produktiv umsetzbar, darüber hinaus gehen Informationen verloren, die mit dem ursprünglichen CAD-Format verbunden sind. Dazu gehören geometrische Details, die durch die Konvertierung verloren gehen, und der Modellbaum, der alle parametrischen Beziehungen des Modells enthält. Zusätzlich werden meist alle semantischen Informationen des Modells gelöscht, z. B. Fertigungs-, Toleranz- oder Materialspezifikationen. Ein erster Ansatz zur Kompensation dieses Problems wurde von Hallmann et al. [95] entwickelt. Dieses Vorgehen übertragt die Toleranzinformationen aus CAD-Bauteilen auf Flächenmodelle.

Allerdings bieten die einfacheren Geometrieformate auch Vorteile, wie z. B. einen geringeren Speicher- und Grafikbedarf, was nicht nur die Visualisierung verbessert, sondern auch die Anwendung und Analyse der Geometrien erleichtert. Aufgrund ihrer einfachen Struktur sind diese Formate gut für die Verarbeitung durch Algorithmen geeignet. Der Verzicht auf semantische Informationen dient dem Schutz sensibler interner Daten und ermöglicht somit einen sicheren Austausch der Geometrie, weshalb Lieferanten diese Darstellungsart häufig verwenden. Die Konvertierung in die genannte Geometriedarstellung ist ein einseitiger Prozess, d. h. eine Konvertierung aus einer CAD-Geometrie ist immer möglich, die Umkehrung des Prozesses ist jedoch sehr schwierig. Diese Art der Umwandlung stellt derzeit noch eine

große Herausforderung dar und viele Forschungsprojekte befassen sich mit diesem Problem [96, 97].

Die CAD-spezifischen Formate dagegen beinhalten die Vorteile der CAD-Software und repräsentieren das Produkt vollumfänglicher. Weiterhin können (teil-) automatisiert zusätzlich relevante Dokumente, beispielsweise Stücklisten, technische Zeichnungen oder Betriebsanleitungen, von der 3D-Geometrie abgeleitet werden. Durch die genaue Darstellung der Produktgeometrie können zudem Fehler frühzeitig erkannt und somit Optimierungszyklen reduziert werden. Neben der Erzeugung von weiteren Dokumenten kann die Erstellung der Modelle auch automatisiert werden, indem parametrische und feature-basierte Modelle genutzt werden. Diese können für die wissensbasierte Konstruktion weiterverwendet werden, indem Regeln oder Berechnungen die Modellerzeugung automatisiert unterstützen [33]. Der Einsatz verbessert zudem die Qualität der 3D-Modelle, da menschliche Modellierungsfehler vermieden werden können [81].

Neben den vielen Vorteilen können CAD-Programme allerdings auch zu Problemen im Ablauf führen. Der Aufwand für die Erzeugung von Modellen erhöht sich deutlich, vor allem da die Vorteile des Einsatzes sich erst in den späteren Phasen der Produktentwicklung zeigen. Auch ergeben sich die Vorteile erst durch die verknüpfte Verwendung mit anderen Software-Lösungen, was eine Änderung der Abläufe und IT-Infrastruktur mit sich bringt. Die parameterische und feature-basierte Modellierung der Bauteile ist grundsätzlich aufwendiger und benötigt daher initial mehr Ressourcen für die Umsetzung.

2.1.3 Datengetriebene Produktentwicklung

Der vorher beschriebene Produktentwicklungsprozess hat im Laufe der Zeit viele verschiedene Entwicklungszyklen durchlebt, ein Überblick ist in Bild 8 aufgeführt. Das vorgestellte methodische Vorgehen zum Entwickeln von Produkten wurde grob ab 1950 entwickelt und eingesetzt. Die nächste größere Änderung war die Nutzung von Computern, was zunächst zur rechnerunterstützten und dann zur rechnerorientierten Produktentwicklung geführt hat. Die immer weitere Nutzung von virtuellen Methoden und Algorithmen hat die Entstehung der virtuellen Produktentwicklung vorangetrieben. Diese setzt auf die digitale Beschreibung des Produkt und die Nutzung möglichst vieler digitaler Ansätze zur Modellierung.

Dementsprechend sind auch viele neue Begriffe in der Produktentwicklung etabliert worden. Ein großer Aspekt umfasst dabei die Computer Unterstützung (*engl. Computer Aided*). Diese wird über den Begriff „CAX“ beschrieben,

wobei das „X“ für den Anwendungsbereich der virtuellen Verfahren steht. Beispiele sind CAD (Design), CAE (Engineering), CAM (Manufacturing), CAP (Planning) oder CAQ (Quality) [98].

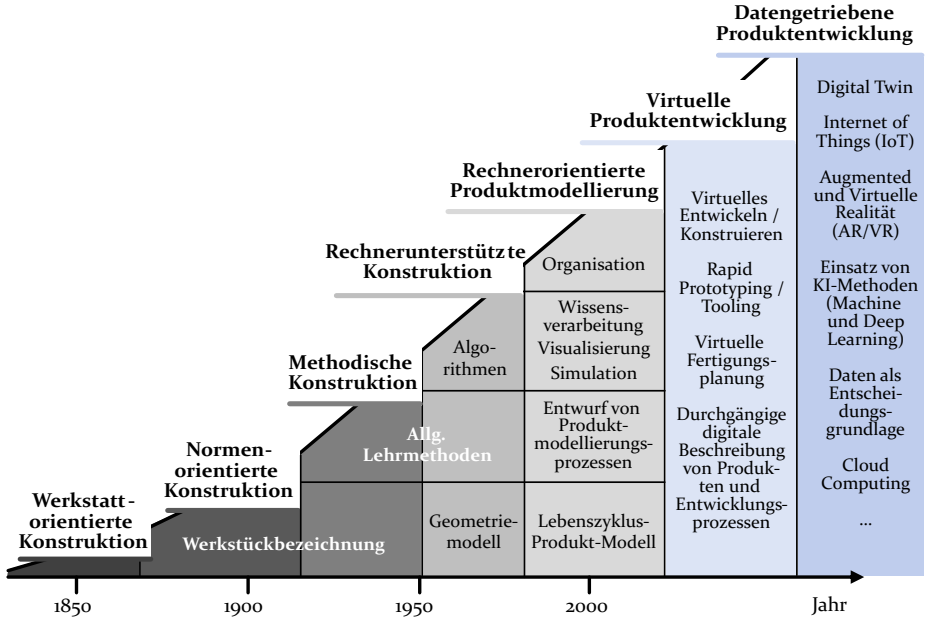


Bild 8: Übersicht der zeitlichen Einordnung verschiedener Produktentwicklungsphasen nach [99]

Daneben sind auch Tools für das gesamte Produktdatenmanagement (PDM) in den Produktentwicklungsprozess integriert worden. Das Ziel dieser Systeme ist die vollständige Rechnerunterstützung in der Produktentwicklung zu ermöglichen, was über die Verknüpfung und Bereitstellung der vorhandenen Daten sowie Schnittstellendefinition umgesetzt wird [98]. Ähnliche Softwarelösungen sind auch für die Fertigung entwickelt worden, diese werden unter der Bezeichnung Produktionsplanungs- und steuerungssystem (PPS) zusammengefasst.

Die neueste Stufe in den Evolutionsphasen der Produktentwicklung ist die datengetriebene Produktentwicklung, bei der die Daten eine große Rolle für die Unterstützung bei Entscheidungsprozessen spielen [10, 100–102]. Ein theoretischer Einsatz der datengetriebenen Produktentwicklung ist in Bild 9 aufgeführt. Die Nutzung kann im gesamten Produktentwicklungsprozess erfolgen und dementsprechend ist die Zielsetzung anpassbar. Nach der Festlegung des Anwendungsbereichs erfolgt die Umsetzung, wobei die Grundlagen dafür die historischen Daten sind. Neben den Daten sind auch ausreichende

Rechenleistung und passende Algorithmen für die erfolgreiche Umsetzung notwendig.

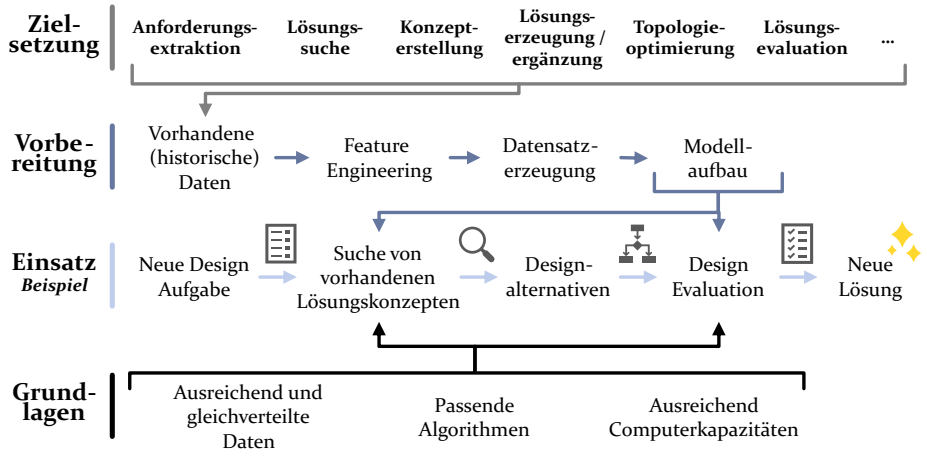


Bild 9: Übersicht eines Einsatzmodells für die datengetriebenen Produktentwicklung nach [102]

Durch die konsequente Nutzung vorhandener Datenquellen konnte die Entwicklungszeit verkürzt und neue kreative Ideen entwickelt werden [54]. Dabei können viele Datenursprünge herangezogen werden, beispielsweise Social Media [103], Simulationsmodelle [P2, P3], Sensoren [104] oder Kundenfeedback [105–107]. Zur bestmöglichen Nutzung der Daten werden zudem neue Technologien in der datengetriebenen Produktentwicklung eingesetzt [16, 108]:

- Internet of Things (IoT)
- Augmented (AR) und Virtual Reality (VR)
- Cloud Computing
- Data Mining (DM)
- Machine-Learning (ML)

Die dargelegten Technologie überschneiden sich auch mit dem Feld der Big Data Verfahren, eine Einordnung dieser Ansätze ist in [109] zu finden. Da für diese Arbeit die Machine-Learning Verfahren von besonderer Relevanz sind, werden diese im Kapitel 2.2 nochmals genauer erläutert.

Einordnung in Produktentwicklungsphasen
Diese neue Methoden und Algorithmen können entsprechend den verschiedenen Abschnitten des Produktentwicklungsprozesses zugeordnet werden. Den allgemeinen Rahmen zur Verortung bietet die VDI 2221/23 aus Kapitel 2.1.1. Diese Zuweisung ist anhand mehrerer Literaturrecherchen analysiert

worden, die erste Untersuchung von Plappert et al. [110] fokussiert sich auf Knowledge-Based Engineering (KBE) und Methoden der Künstlichen Intelligenz (KI). Die generelle Einordnung erfolgte dabei neben den Stufen des Entwicklungsprozesses auch in die Klassen „Methodik", „Anwendung" und „Algorithmus". Das Ergebnis der Recherche war, dass der Hauptfokus der Arbeiten im Bereich KBE und KI auf dem Entwurf des Produktes liegt, da 70 % der relevanten Literatur dieses Themenfeld bearbeitet. Auf die Konzeptphase entfallen ca. 12 % der Beiträge, wobei der Schritt *Suche und Definition der prinzipiellen Lösungen* mit lediglich 3 % am wenigsten repräsentiert ist.

Eine weitere Analyse von Feng et al. [16] hat sich auf die Kombination von datengetriebener Produktentwicklung und intelligenten Fertigungsverfahren spezialisiert. Dabei werden die einzelnen Schritte des Entwicklungsprozesses aufgezeigt und vorhandene Verfahren und Potentiale für datengetriebene Methoden vorgestellt. In dieser Recherche sind mehrere zentrale Themen und Herausforderungen für den Entwicklungsprozess herausgearbeitet worden:

- Automatische Nutzung von Kundendaten und Überführung in Anforderungen mittels Text Mining und Natural Language Processing (NLP)
- Verwendung der vorhandenen Entwicklungsdaten und -historie für datengetriebene Methoden, um ähnliche Design Aufgaben zu lösen
- Förderung kollaborierender Konzeptentwicklung über die Produktentwicklung hinweg durch Integration von Kunden und Produktion
- Kombination von datengetriebenen Algorithmen und Unsicherheitsbetrachtungen zur Produktoptimierung

Ähnliche zentrale Aspekte sind von Vlah et al. [10] erarbeitet worden, wobei der Hauptfokus auf der datengetriebenen Produktentwicklung liegt. In dieser Recherche wurden 249 Veröffentlichungen für eine ausführliche Auswertung untersucht. Im zeitlichen Verlauf konnte deutlich gezeigt werden, dass ab ca. 2017 die Anzahl an Paper in diesem Themenfeld stark anstieg. Neben dem Veröffentlichungsdatum wurde auch die örtliche Verteilung der Beiträge analysiert. Die meisten Publikationen sind in den USA entstanden, gefolgt von China und Europa. Eine ausführliche Einordnung der gefundenen Veröffentlichungen in die Phasen der Produktentwicklung ist in Bild 10 dargestellt.

Die Übersicht zeigt die Phasen der Produktentwicklung auf der X-Achse und die unterschiedlichen datengetriebenen Methoden auf der Y-Achse. Die Anzahl an entwickelten Verfahren ist an den jeweiligen Schnittstellen eingetragen. Bei der Auswertung fällt zunächst auf, dass die meisten Methoden im

Bereich des Kunden[1] bzw. Anwender-Analyse angesiedelt sind. Die meisten Veröffentlichungen verwenden zudem Verfahren aus der Textverarbeitung, entweder durch NLP oder Text Mining. Die Schnittmenge der Kundenanalyse und der Text-Verfahren hat demzufolge die größte Sammlung an Beiträgen und deckt sich daher mit den Erkenntnissen aus den beiden vorherigen Literaturstudien.

	Marktrecherche	Kundenfeedback	Anforderungsdefintion	Funktionsanalyse	Entwicklung Prinziplösung	Entwurfsphase	CAD und CAE	Produkt Plattform Zuordnung	Ausarbeitungsphase	Versuch und Iteration	Fertigung	Produktanwendung
Regression	2	4	1	0	0	0	0	0	1	0	3	0
Klassifikation	0	15	6	3	1	10	2	3	3	3	6	2
Clustering	0	10	1	4	4	2	2	4	1	0	2	0
Dimensions-reduktion	0	1	0	0	1	1	0	2	0	0	0	0
Deep-Learning	0	1	1	0	0	1	1	1	0	0	4	0
NLP	5	16	4	0	6	0	0	1	1	1	1	1

Bild 10: Verknüpfung von datengetriebenen Methoden und den Schritten des Produktentwicklungsprozesses nach [10]

Die zweit meisten Publikationen sind in der Produktion eingeordnet, mit dem größten Anteil an Klassifikationsverfahren. Dementsprechend entfallen auf die restlichen Schritte des Entwicklungsprozesses weniger verfügbare Methoden. Die Darstellung zeigt ebenso deutlich die vorhandenen Lücken aus der Kombination von datengetriebenen Verfahren und Entwicklungsschritten. Auffällig sind dabei zwei größere Bereiche ohne aktuell vorhandene Verfahren. Der erste umfasst die Verknüpfung von Deep-Learning und Dimensionsreduzierung in den frühen Phasen des Entwicklungsprozesses, insgesamt sind nur vier Veröffentlichungen zur Entwurfsphase eingeordnet, wobei zwei die Hauptkomponentenanalyse (PCA) und zwei weitere die Restricted Boltzmann Machine (RBM) anwenden. Ein weiteres wenig beachtetes Feld ist

[1] In dieser Arbeit wird aus Gründen der besseren Lesbarkeit das generische Maskulinum verwendet, welches ausdrücklich alle Geschlechtsidentitäten einschließt.

die Nutzung von Regressionsverfahren über den gesamten Produktentwicklungsprozess, wobei in der Recherche nur zwei Publikationen diesem Bereich zugeordnet wurden.

Eine weitere Literaturrecherche von Bertoni [111] zu diesem Themengebiet umfasst die Nutzung von datengetriebenen Methoden in der Konzeptphase. Insgesamt wurden 127 Veröffentlichungen analysiert, wobei viele der vorher ermittelten Erkenntnisse ebenfalls bestätigt wurden. Die Recherche zeigt, dass ein großer Anstieg an Veröffentlichungen ab 2017 erkennbar ist, und die meisten Verfahren in der Konzeptphase verortet sind. Allerdings sind im Gegensatz zur vorherigen Analyse keine Verfahren für die Fertigung ermittelt worden. Parallelen zeigen sich zudem bei den genutzten Methoden, da wiederum das Text-Mining für die Ermittlung der Kundenansprüche am häufigsten Einsatz fand, mit 32 aus 52 Veröffentlichungen. Die restlichen Publikationen teilen sich auf die übrigen Schritte der Konzeptphase auf, wobei die „Erzeugung des Konzepts“ mit sechs gefolgt von der „Erkundung des Design Space“ mit fünf noch die größten Ansammlungen aufweisen.

Zusammenfassend zeigt die Auswertung der vorhandenen Literaturstudien folgende Ergebnisse:

- hohes Potential zur Verbesserung des Entwicklungsprozesse durch die neuen datengetriebenen Methoden und Algorithmen
- steigendes Interesse und vielfältige Einsatzmöglichkeit im Kontext der Produktentwicklung
- aktuell großer Fokus auf textbasierten Methoden vor allem zur Definition der Kunden/Nutzer-Anforderungen

Damit allerdings der Wandel von der virtuellen zur datengetriebenen Produktentwicklung vollzogen werden kann, sind vor allem neue Methoden und Algorithmen notwendig. Diese können unter dem Begriff „datengetriebene Methoden“ zusammengefasst werden und sollen im nächsten Kapitel genauer eingeordnet und erklärt werden.

2.2 Datengetriebene Methoden

Unter dem Begriff datengetriebene Methoden werden allgemeine Verfahren aus der Domäne der Data Science zusammengefasst. Dieses Forschungsfeld hat sich aus dem Bereich der Statistik weiterentwickelt [112, 113] und eine der ersten Beschreibung erfolgte 1962 von Tukey [114]. Dabei änderte sich der Fokus von der statistischen Inferenz zu anderen Aspekten, wie Datenbereitstellung, -reinigung, -visualisierung, -sammlung und Analyseverfahren [115]. Ebenso hat [114] auch neue Verfahren zur Datenauswertung entwickelt, wie beispielsweise Boxplots oder Streudiagramme. Mit steigender Leistungsfähig-

keit der Computer, mehr verfügbaren Daten und besseren Softwarelösungen hat sich die Datenanalyse immer weiter zu einem interdisziplinären Feld, der heutigen Data Science, entwickelt [112, 116, 117]. Im Venn-Diagramm in Bild 11 sind die verschiedenen Disziplinen und deren Schnittmengen dargestellt.

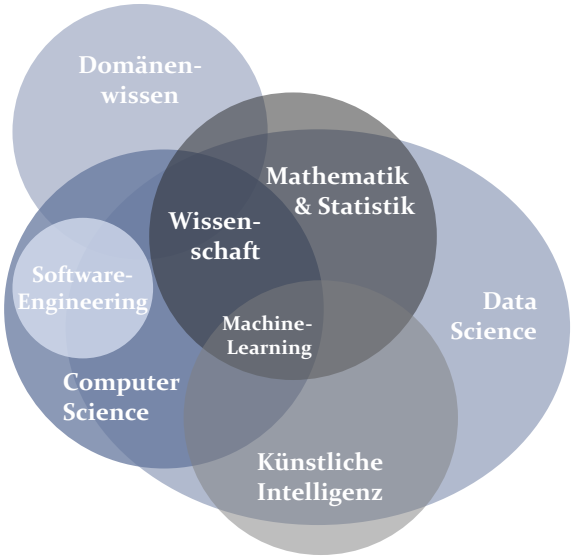


Bild 11: Grobe Einordnung der verschiedenen Begriffe nach [113]

Dabei beinhaltet Data Science Elemente der Fachbereiche Mathematik bzw. Statistik, Künstliche Intelligenz, Machine-Learning, Computer Science, Software Entwicklung, Domänen-Expertenwissen und Forschungsergebnisse. Aktuelle Einschätzungen bezeichnen den Job des Data Scientist sogar als den ansprechendsten des 21. Jahrhunderts [118]. In den folgenden Kapiteln soll vor allem auf die Machine- und Deep-Learning-Grundlagen ausführlich eingegangen werden, da diese die größte Relevanz für diese Arbeit aufweisen.

2.2.1 Übersicht Machine-Learning

Das Forschungsfeld beschreibt Verfahren, die einem Computerprogramm die Fähigkeit zu Lernen ermöglichen, ohne dabei explizit darauf programmiert zu sein [119–121]. Der Begriff Machine-Learning wurde zuerst von Samuel [119] zur Erläuterung eines neuen Algorithmus eingeführt, welcher einem Computer das Schach spielen beibringt [122]. Diese ursprüngliche Definition wird teilweise noch durch das Finden von unbekannten Mustern aus Daten ergänzt [115, 123, 124]. Das generelle Vorgehen für ML-Systeme besteht aus den drei Bereichen Daten, Features und Modell. Die Daten sind der essentielle Bestandteil des gesamten Vorgehens und müssen dementsprechend zuerst

gesammelt, gereinigt und für die Verwendung aufbereitet werden. Abhängig vom Lernverfahren muss in diesem Prozessschritt auch ein Labeling der Daten erfolgen, also ein manuelles Klassifizieren des Datensatzes. Eine Aufteilung der gesamten vorhandenen Daten in Trainings- und Testdatensätze ist ebenfalls sinnvoll, damit die Performance des trainierten Modells an unbekannten Daten ermittelt wird. Der Trainingsdatensatz wird dem Namen nach für das Trainieren des Modells verwendet, er ist somit dem Modell bekannt. Dagegen wird der Testdatensatz erst nach dem Abschluss des Modelltrainings für die Auswertung relevant. Die in ihm enthaltenen Daten sind dem Modell komplett unbekannt, wodurch die Übertragbarkeit auf unbekannte Daten mit dieser Aufteilung des Gesamtdatensatzes geprüft werden kann. Der zweite Schritt beschäftigt sich mit der Feature-Definition, die dem Modell übergeben wird. Zur Erstellung sollte domänenspezifisches Wissen eingesetzt werden, damit die Features möglichst kompakt sind, aber trotzdem eine hohe Aussagekraft besitzen [122]. Die Auswahl des passenden Modells ist der letzte Schritt im Ablauf, wobei viele Randbedingungen berücksichtigt werden müssen, u. a. die Aufgabenstellung, vorhandene Daten, Zielgrößen und erreichte Vorhersagegenauigkeit.

Der Aufbau von neuen Modellen zur Vorhersage von unbekannten Daten wird als „Training“ oder „Lernen“ bezeichnet [124, 125]. Die Bezeichnung ist angelehnt an das menschliche Lernen, was zu großen Teilen auf Erfahrung basiert. Beim Lernprozess von Machine-Learning-Modellen wird diese Erfahrung durch Daten substituiert, die einem Trainingsalgorithmus übergeben werden. Dieser erzeugt nach dem erfolgreichen Durchlaufen des Training ein anwendbares Modell, welches für neuen Daten verwendet werden kann [125]. Dieser Trainingsprozess konnte das Problem früherer Expertensysteme lösen, indem die aufwendige händische Wissensrepräsentation automatisiert wird [122]. Die unterschiedlichen Aufgaben von Machine-Learning Algorithmen können grob anhand der Trainings- bzw. Lernverfahren untergliedert werden: [116, 122, 125–128]

- Überwachtes Lernen (*engl.: supervised learning*)
- Unüberwachtes Lernen (*engl.: unsupervised learning*)
- Verstärkendes Lernen (*engl.: reinforcement learning*)

2.2.1.1 Überwachtes Lernen

Das überwachte Lernen unterscheidet sich vom unüberwachten hauptsächlich durch den Einsatz von gelabelten Daten für die Modellbildung. Demzufolge ist jedem Datenpunkt eine entsprechende Klasse oder ein Zahlenwert zugeordnet, welche vorher definiert bzw. ermittelt werden müssen. Somit

ist die Eingabe und Ausgabe für das Modell bekannt und das Ziel ist, diese Verknüpfung möglichst gut über ein trainiertes Modell abzubilden.

Die beiden Hauptanwendungsgebiete für überwachtes Lernen sind Regression und Klassifikation. Die Regressionsanalyse hat das Ziel, die funktionalen Abhängigkeiten zwischen Merkmalen zu ermitteln, um danach Zusammenhänge aus den Daten abzuleiten. Aus diesem Grund wird sie häufig zur numerischen Vorhersage eingesetzt. So kann beispielsweise die Produktqualität durch gezielte Steuerung einzelner Produktionsmerkmale verbessert werden, indem die notwendigen Parameterwerte zur Steigerung der Qualität mittels Regression ermittelt werden. Generell werden die Verfahren zwischen linearen und nicht-linearen Regressionsalgorithmen unterschieden, wobei mehr Informationen unter [129] zu finden sind.

Im Gegensatz dazu ist das Ziel der Klassifizierung das Zuordnen von Daten zu vorher definierten Klassen. Schematisch ist die Klassenzuordnung in Bild 12 für zwei Merkmale dargestellt, wobei ein neues Objekt (schwarzer Datenpunkt) in die bereits vorhandenen Klassen (dunkelblaue Klasse) eingeordnet werden soll. Die Klassen müssen vor der Analyse definiert werden und jedem Trainingsdatenpunkt zugeordnet sein. Die einsetzbaren Algorithmen zur Klassifizierung sind ebenfalls sehr vielfältig, Beispiele sind Regelbasierte Systeme, Neuronale Netze (ANN), k-nearest neighbors Algorithmus (KNN), Entscheidungsbäume, Support Vektor Machine (SVM) oder Naive Bayes Klassifikation [116, 122, 126, 129].

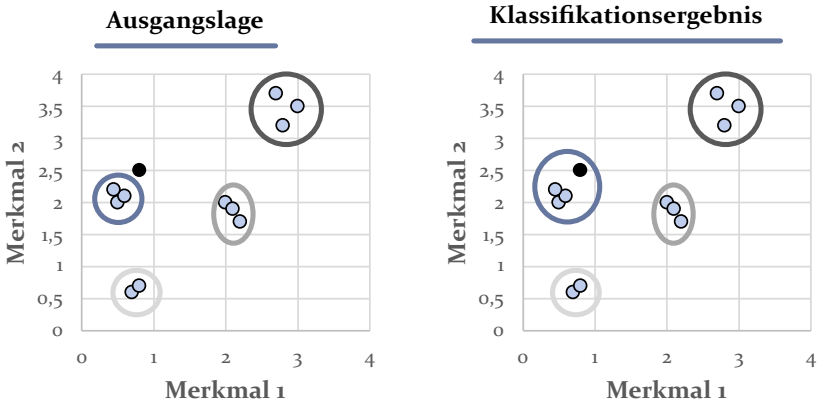


Bild 12: Beispiel für das Ergebnis einer Klassifikation mit zwei Merkmalen, wobei ein neuer Datenpunkt (schwarz) in die dunkelblaue Kategorie einsortiert wird

2.2.1.2 Unüberwachtes Lernen

Bei den unüberwachten Verfahren werden neben den Daten keine weiteren Informationen benötigt, wodurch diese die einzige Eingabe für das Modell sind. Angewendet werden die Modelle häufig für das Clustering, also dem Einordnen von Datenpunkten in eine vordefinierte Anzahl an Gruppen bzw. Klassen. Im Gegensatz zur Klassifizierung ist nur die Anzahl an Klassen bekannt und nicht die genaue Zuordnung. Ein einfaches Beispiel für das Clustering ist in Bild 13 abgebildet. Auf der linken Seite ist der Input für den Clusteralgorithmus dargestellt und auf der rechten das Ergebnis, veranschaulicht mit den unterschiedlich gefärbten Kreisen. Es gibt wiederum eine große Anzahl an Methoden zur Durchführung der Clusteranalyse z. B. der k-Means-Algorithmus, die Fuzzy-Variante, der Ansatz der Erwartungsmaximierung oder Neuronale Netze [129–132].

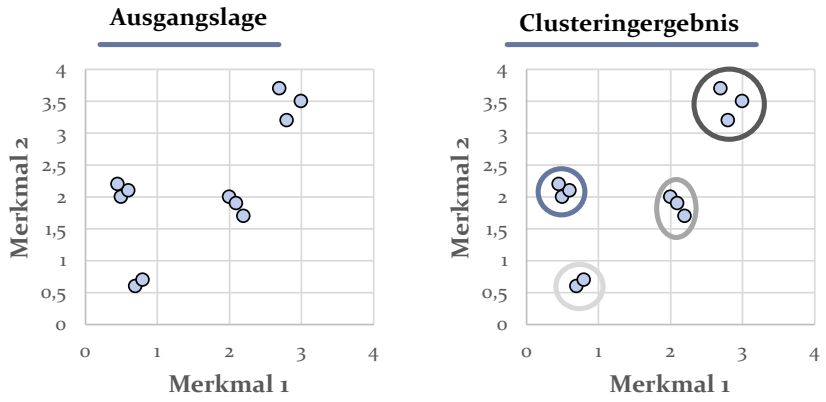


Bild 13: Beispielhafte Anwendung von Clusteringalgorithmen zur Einordnung von Datenpunkten in vier Klassen

Daneben können unüberwachte Algorithmen auch zur Dimensionsreduktion von großen Datenmengen verwendet werden, indem die Anzahl an relevanten Features verringert wird. Dies wird häufig als Vorbereitung für überwachte Lernverfahren verwendet. Zusätzlich können unüberwachte Methoden auch zur Ableitung von Regeln aus Datenbeständen herangezogen werden [126, 133, 134].

2.2.1.3 Verstärkendes Lernen

Die dritte Lernart unterscheidet sich von den vorherigen Ansätzen, da das Training über eine Belohnung auf getroffene Handlungen erfolgt. Bei den bisher vorgestellten Verfahren wird über einen Abgleich zur wahren Lösung aus dem Datensatz eine stetige Bewertung des Modells während des Trainings

durchgeführt. Dagegen führt beim verstärkenden Lernen ein Agent eine Handlung aus, welche die Umwelt beeinflusst. Diese Änderung führt zu einer Belohnung bzw. Bestrafung des Agenten und die Änderung der Umwelt wird wiederum an den Agenten übergeben, auf Basis derer er eine neue Aktion ausführt. Der gesamte Ablauf ist in Bild 14 schematisch dargestellt [135].

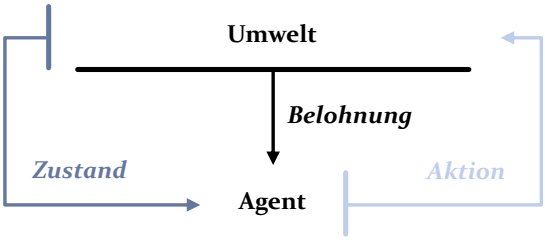


Bild 14: Übersicht für den Ablauf von verstärkendem Lernen nach [127]

Diese Art des Lernens wird häufig eingesetzt, wenn in einer definierten Umgebung eine Sequenz an Handlung notwendig ist, beispielsweise bei der Steuerung von Robotern oder Gegenspielern in Videospielen [126, 135].

2.2.1.4 Deep-Learning

Eine besondere Kategorie im Bereich des Maschinellen Lernens ist das Deep-Learning, welches häufig mit der Verwendung neuronaler Netze verknüpft ist [136–139]. Es unterscheidet sich vom klassischen Machine-Learning durch die Anzahl der Schichten und die Erzeugung von Merkmalen, da bei einem ML-Modell der Entwickler die Modellmerkmale selbständig festlegt. Im Gegensatz dazu werden diese in einem Deep-Learning-Modell automatisiert während der Lernphase erzeugt, was auch Bild 15 zeigt. In [136] wird dieser Prozess als automatische Zerlegung komplizierter Konzepte in einfachere beschrieben.

Das Ergebnis der Überführung des Deep-Learning-Prinzips auf Bilder sind Convolutional Neural Networks (CNN). Diese Neuronalen Netze haben als Vorbild das menschliche Sehen und zielen darauf ab, den visuellen Kortex des Auges nachzubilden. Zu diesem Zweck werden verschiedene Neuronen, Faltungs- und Pooling-Schichten verwendet, um mehr Informationen aus den Bilddaten zu filtern. Der Begriff „Convolutional“ wurde erstmals 1989 von LeCun et al. [140] eingeführt und wird seitdem mit dieser Art von Netzwerken in Verbindung gebracht. Ein generelles Beispiel für die Erkennung von Verkehrsschildern ist im Anhang in Bild 87 aufgeführt, inkl. dem Aufbau des Netzwerks. Anhand des Bildes sind die verschiedenen Bestandteile gut zu erkennen, welche im Anhang unter Kapitel A.3 genauer erläutert werden.

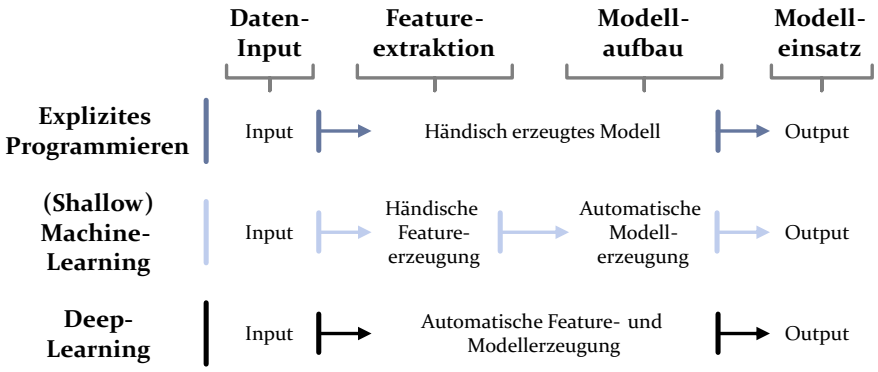


Bild 15: Prozessvergleich zwischen Machine und Deep-Learning nach [136, 138]

Entstehung und Entwicklung von CNNs

Eines der ersten und dadurch sehr bekanntes CNN ist das 1998 veröffentlicht LeNet von [141], welches in der Lage ist, handgeschriebene Buchstaben zu erkennen. Mit der Hilfe von immer leistungsfähigeren Computersystemen und größeren, öffentlich zugänglichen Datensätzen (z. B. MNIST [141], CIFAR [142], PASCAL VOC [143] oder ImageNet [144]) war es möglich, CNNs praktisch und nutzbringend einzusetzen. Im Rahmen eines Bilderkennungswettbewerbs im Jahr 2012 erregte das AlexNet aus [145] mit seinen Ergebnissen besondere Aufmerksamkeit. Im Vergleich zum LeNet-5 hat das AlexNet mehr Schichten und ist für höher aufgelöste Bilder geeignet. Zwei Jahre später wurde von Simonyan und Zisserman [146] ein noch tieferes Netz mit 16 Schichten vorgestellt, welches vgg16 genannt wurde. Alle vorgenannten CNN-Modelle sind seriell aufgebaut, was es dem Netz ermöglicht, alle Schichten sequentiell zu durchlaufen. Nach [147] sollten sich theoretisch tiefere Netze besser an eine gestellte Aufgabe anpassen, da sich der Parameterraum für die Adaption vergrößert. Allerdings konnte gezeigt werden, dass die erreichte Genauigkeit zur Sättigung neigt und danach abnimmt. Dieses beschriebene Problem kann durch mehrere Pfade im CNN-Aufbau gelöst werden, was von He et al. [148] durch Residualblöcke im CNN implementiert und ResNets genannt wurde.

Durch den frühen erfolgreichen Einsatz von CNN-Modellen für die Erkennung von Bilder haben diese den Weg für weitere Deep-Learning-Methoden geebnet [136]. CNNs sind in der Lage große Mengen an einheitlichen Daten zu verarbeiten und diese einzuordnen. Demzufolge werden diese vor allem für die Erkennung von Schriften oder Bildern eingesetzt und bilden die Grundlage für das nächste Kapitel: die Objekterkennung in Bildern.

2.2.2 Objekterkennung (Bilder)

Die Aufgabe das menschliche Sehen in den Computer zu übertragen ist eine lange existierende Problemstellung, welche sich unter dem Überbegriff Computer / Machine Vision etabliert hat. Das grundsätzliche Ziel ist das Abbilden der realen Welt über Bilder mittels verschiedener Modelle. Da Bilder immer eine zweidimensionale Projektion der 3D-Welt sind, ist das Finden und Bereitstellen der Informationen in Bildern herausfordernd [149].

Weiterhin ist der Begriff nicht eindeutig definiert und umfasst eine Vielzahl an Verfahren und Methoden [150, 151]. Das Vorbild ist dabei immer das menschliche Sehen, welches der Computer zu imitieren versucht. Dabei werden beispielsweise Algorithmen aus der Bildverarbeitung, Mustererkennung, Kalibrierung, Kantenerkennung, Computergrafik und dem Maschinellen Lernen angewendet [152].

2.2.2.1 Aufgabenübersicht

Das Aufgabenspektrum bei der Erkennung von Objekten lässt sich nach [151, 153–156] in drei Bereiche spezifizieren: Bildklassifizierung, Objekterkennung und Bildsegmentierung. Diese werden im folgenden Kapitel genauer vorgestellt. In Bild 16 sind die unterschiedlichen Ergebnisse für die verschiedenen Aufgaben dargestellt.

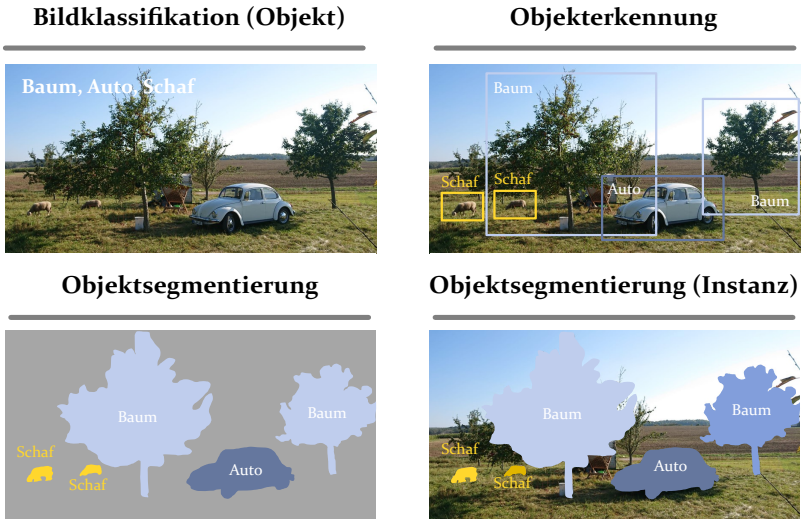


Bild 16: Darstellung der unterschiedlichen Aufgaben von Bilderkennungsverfahren nach [157]

Klassifizierung
Das Ziel der Bildklassifikation ist die Zuordnung eines Bildes zu bestimmten

Klassen, welche vor der Untersuchung festgelegt sein müssen. Entweder wird die Klasse mit der höchsten Punktzahl dem Bild zugeordnet, wobei das Klassenlabel den Hauptinhalt des Bildes beschreibt, oder mehrere passende Label werden als Ergebnis präsentiert [151, 153, 157, 158].

Objekterkennung
Dagegen können durch die Objekterkennung die Position und Klasse von Objekten in den Bildern erkannt werden. Hierzu werden Bounding Boxen verwendet, welche über ihre Größe und Position beliebig auf einem Bild angeordnet und auch mehrmals pro Bild platziert werden können. Dadurch kann mehr semantische Informationen aus den Bildern extrahiert werden. Allerdings ist in die Datengenerierung aufwendiger, da neben einer Klasseninformation auch die Position der Bounding Box als Eingabe definiert werden muss [151, 154–156, 159].

Segmentierung
Bei der Segmentierung erfolgt die Klassifizierung auf Pixelebene. Durch diese genaue Zuordnung jedes Pixels zu einer Klasse oder dem Hintergrund können die Umrisse verschiedener Objekte in einem Bild erkannt und kategorisiert werden. Die Segmentierung ist herausfordernder als die Klassifizierung oder Objekterkennung und dementsprechend auch zeitaufwendiger. Die Segmentierung ist beispielsweise in der Robotik notwendig, um ein präzises Greifen von Objekten zu ermöglichen [151, 153, 158].

Für die Anwendung in der Produktentwicklung sind vor allem die Objekterkennungsverfahren relevant. Diese haben in den letzten Jahren einen enormen Entwicklungsschritt gemacht und sind besonders gut geeignet die unterschiedlichen Bestandteile der technischen Skizzen aus den frühen Phasen der Produktentwicklung zu erkennen (siehe Kapitel 2.1.2). Dementsprechend werden im nächsten Kapitel die relevanten und etablierten Algorithmen genauer vorgestellt und erklärt.

2.2.2.2 Objekterkennungsalgorithmen

Die Erkennung von Objekten in Bildern ist ein Teilbereich der Computer Vision und beschäftigt sich mit dieser für Menschen verhältnismäßig einfachen, aber für Algorithmen sehr aufwendig umzusetzenden Aufgabe [149]. Die Herausforderung besteht darin, ein inverses Problem zu lösen, da die Bilder eine Projektion der realen 3D-Welt darstellen und aus den Bildern Eigenschaften über die reale Welt ermittelt werden sollen, z. B. die Erkennung von Formen oder Farbverteilungen. Ziel ist dabei über statistische Verfahren oder Maschinelles Lernen unbekannte Variablen zu ermitteln, damit die Lösung vollständig spezifiziert werden kann [151].

Die Anzahl an Veröffentlichungen im Bereich der Objekterkennungsalgorithmen nahm nach [160] innerhalb der letzten 20 Jahren deutlich zu. Die Anfänge basierten häufig auf dem Abgleich von vorhandenen Elementen über Regeln oder Matchingverfahren [161]. Dabei lag der Fokus auf dem Erkennen von spezifischen Objekten, die sich durch ihre Eigenschaften beschreiben lassen, wobei u. a. auch Klassifizierer aus dem Maschinellen Lernen zum Einsatz kamen [162].

Die nächste große Entwicklung war der Einsatz von bag-of-feature-Verfahren, welche Objekte und Bildern als eine ungeordnete Sammlung von Merkmals-Descriptoren beschreiben [151, 163]. Eine Weiterentwicklung war der Fokus auf die lokalen Merkmale in Bildern [157]. Diese haben den Vorteil, dass sie unabhängig von Bildänderungen, wie Translation, Rotation oder Beleuchtung, sind. Ein wichtiger Beitrag war die Vorstellung der Scale Invariant Feature Transform (SIFT) Merkmale [164]. Andere Beispiele sind multi-scale oriented patches (MOPS) [165] oder Gradient location-orientation histogram (GLOH) [166]. Ein sehr bekannter Vertreter für ein von der Skalierung und Translation unabhängiges Verfahren ist das Histogram of Oriented Gradients (HOG) von [167]. In diesem Vorgehen werden einzelne Bildausschnitte in HOG-Features transformiert und anschließend über eine SVM eingeordnet. Eine Weiterentwicklung des HOG-Ansatzes ist das Deformable Part-based Model (DPM) von [168, 169]. Die Idee ist, die Bilder in möglichst viele Bestandteile zu unterteilen, danach zusammenzusetzen und dadurch die Objekte zu erkennen. Beispielsweise besteht ein Auto aus Fenstern, Türen, Reifen und Lichtern.

Der nächste große Meilenstein war die Entwicklung und Umsetzung von Convolutional Neural Networks. Eine ausführliche Erläuterung zu CNNs ist bereits in Kapitel 2.2.1.4 beschrieben. Unterstützt durch die steigende Leistungsfähigkeit von Computern und der vermehrten Verfügbarkeit von gelabelten Datensätzen sind immer neue Verfahren entwickelt worden, welche sich mittlerweile in ein- und zweistufige Verfahren untergliedern lassen. Die durchgeführten Wettbewerbe und öffentlich zur Verfügung gestellten Datensätze haben einen großen Einfluss auf die Entwicklung der neuen Verfahren. Erst durch diese, welche mehr als eine Million Bilder [144] enthalten, war es möglich, CNNs sinnvoll zu trainieren und zu vergleichen. Populäre Wettbewerbs- bzw. Benchmark-Datensätze sind das ImageNet [154], Pascal VOC [153] und der COCO-Datensatz [170].

Dabei kann der zeitliche Ablauf anhand der Einführung von CNNs ab 2012 untergliedert werden, wie in Bild 17 dargestellt [145]. Die vorher entwickelten Verfahren basieren stark auf Featureerzeugung und -erkennung in Bildern

oder auf das Finden von spezifischen Objekten, beispielsweise nur Fußgänger oder Autos [151, 171].

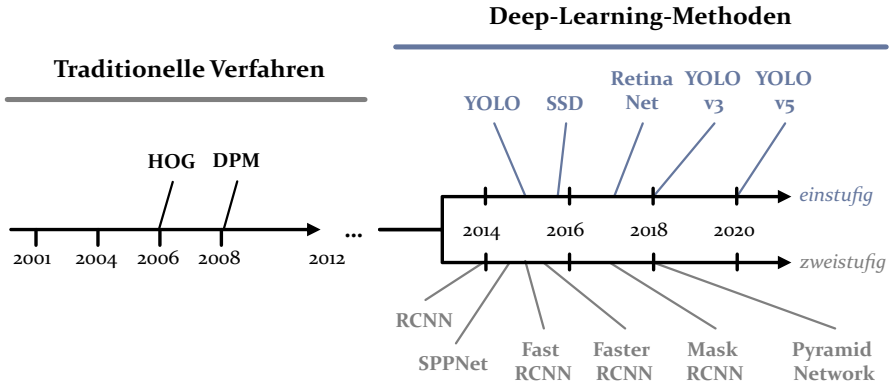


Bild 17: Zeitliche Einordnung der Meilensteine für Verfahren der Objekterkennung nach [160]

Die ersten Verfahren, die CNNs für die Objekterkennung genutzt haben, waren zweistufige Verfahren. Dabei wird zuerst ein CNN verwendet, um mögliche Regionen zu ermitteln. Diese werden im nächsten Schritt mit einem anderen Klassifizierer eingeordnet. Dagegen basieren einstufige Algorithmen auf einem CNN, welches die Lokalisation und Klassifizierung der Bounding Boxen in einem Schritt durchführt [151, 160, 172, 173].

Einstufige Verfahren

Eines der bekanntesten einstufigen Verfahren ist das YOLO (You Only Look Once) System von Redmon et al. [174], dessen grundsätzlicher Aufbau in Bild 18 illustriert ist. Dabei wird, wie der Name schon sagt, nur einmal auf das Bild geschaut und auf dieser Basis mehrere Bounding Boxen inklusive Klassen durch ein CNN vorgeschlagen. Das ganze Vorgehen wird dabei als Regression-Problem beschrieben, wodurch der gesamte Ansatz sehr schnell ist und auch direkt auf Videos angewendet werden kann. Das Verfahren wurde stetig weiterentwickelt, wobei die aktuelle Version das YOLOv8 ist [175].

Ein andere Ansatz wurde von Liu et al. [176] mit dem SSD (Single Shot Detector) Modell verfolgt. Dieser kombinierte den Einsatz von Multi-Referenz- und Multi-Auflösungs-Erkennungstechniken mit einem einstufigen Vorgehen, wodurch besonders die Erkennungsgenauigkeit kleiner Objekte weiter gesteigert werden konnte [160].

Das RetinaNet wurde von Lin et al. [177] vorgestellt und ist eine Verknüpfung von ein- und zweistufigen Verfahren. Dabei wird die Focal-Loss Funktion genutzt, welche das Problem der ungleichen Verteilung zwischen Vorder-

und Hintergrund bei den Trainingsbildern lösen soll. Dadurch sollen die schwierig zu klassifizierenden Bilder stärker gewichtet werden und somit die gesamte Erkennung verbessert werden. Somit konnten die Vorteile der schnellen Erkennung mit einer (für einstufige Verfahren) hohen Erkennungsgenauigkeit kombiniert werden [160, 172].

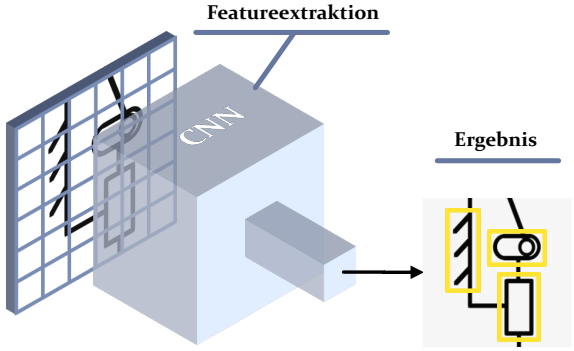


Bild 18: Modellübersicht für den YOLO-Detector nach [174]

Eine weitere Umsetzung ist von Zhao et al. [178], welche mittels Multi-Level-Feature Pyramiden Netzwerken (MLFPN) eine verbesserte Skalierungsvariation über die verschiedenen Objekte erzeugt. Die dreistufigen Feature Pyramiden Netzwerke werden mit der SSD Architektur verknüpft und können somit eine leichte Verbesserung gegenüber dem RetinaNet in der Erkennungsgenauigkeit erzielen, wobei zeitgleich die Erkennungsgeschwindigkeit langsamer wurde [172].

Zweistufige Verfahren

Eine der ersten erfolgreichen Adaption von CNNs für die Objekterkennung war das region-based CNN (R-CNN) Modell von Girshick et al. [179]. Dieser Ansatz umgeht das Problem, dass viele mögliche Regionen für eine Objektposition durch das starre sliding window Vorgehen ermittelt werden, durch einen Suchalgorithmus, der die möglichen Bounding Box Regionen auf ca. 2.000 Vorschläge eingrenzt. Als CNN-Architektur für die Featureerzeugung der verschiedenen Bild-Regionen wird das AlexNet von [145] verwendet. Anschließend erfolgt die Klassifizierung der Vektoren mittels eines SVM Klassifizierers. Die gesamte Struktur ist in Bild 19 dargestellt. Der Nachteil des Verfahrens ist der hohe Trainingsaufwand, die langsame Erkennungsgeschwindigkeit und die fehlende Fähigkeit den Suchalgorithmus über das Modell Training zu verbessern [160].

Eine Verbesserung konnte von He et al. [180] erzielt werden, indem spatial pyramid pooling (SPP) Schichten in die Architektur integriert wurden. Diese

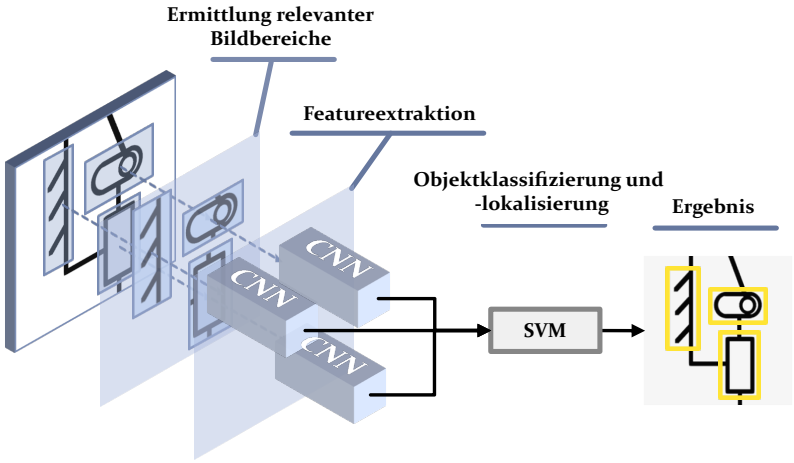


Bild 19: Aufbau eines R-CNN-Modells nach [179]

ermöglichen die Berechnung eines fixen Feature Vektors unabhängig von der Input-Bildgröße. Damit können für ein Bild die Feature-Maps einmal bestimmt und nachfolgend gleichlange Feature-Vektoren für beliebige Regionen im Trainingsprozess generiert werden. Dadurch ist die Erkennungszeit 20-mal schneller als beim R-CNN [160].

Eine weitere Iteration durch den ursprünglichen Autor wurde ein Jahr nach der Veröffentlichung des R-CNN präsentiert. Das von Girshick [181] als Fast R-CNN bezeichnete Modell verknüpft den R-CNN mit dem SPP-Ansatz. Dabei wird das Eingangsbild mittels Convolutional und Max Pooling Schichten in eine Feature-Map umgewandelt. Danach extrahiert eine Region of Interest (RoI) Schicht einen Merkmalsvektor aus der Feature Map, welcher nachfolgend von zwei Neuronalen Netzen genutzt wird, um die Klasse und den Ort der Bounding Box zu bestimmen. Durch diese Maßnahmen konnte die Trainings- und Erkennungszeit weiter verkürzt und die Genauigkeit erhöht werden [151].

Im nächsten Schritt wurde der Suchalgorithmus für die Bounding Boxen überarbeitet. Hierzu präsentierte [182] das Faster R-CNN, welches einen lernenden Suchalgorithmus enthält, das Region Proposal Network (RPN). Das Faster R-CNN besteht aus zwei Modulen, dem RPN und dem Detektor, welcher vom Fast R-CNN übernommen wurde. Das RPN basiert auf einem fully connected Neuronalen Netz, welches Vorschläge für die relevanten Regionen liefert und einen Score, zu welcher Wahrscheinlichkeit sich ein Objekt in dieser Region befindet. Durch diese neue Integration konnten bessere Erkennungsgenauigkeiten erzielt werden [160].

In der nächsten Erweiterung wurde mit der Veröffentlichung des MASK R-CNN die Objekt Segmentierung in das Aufgabenspektrum durch [183] integriert. Dieses basiert wiederum auf dem Faster R-CNN, allerdings mit der Implementierung eines Backbone Netzwerks, welches für die Erzeugung der Feature Map verantwortlich ist. Als Netzwerke kommen hier ResNet Architekturen [184] in einer möglichen Verbindung mit Feature Pyramid Networks (FPN) [185] zum Einsatz. Die damit generierte Feature-Map wird dann an die, aus dem Faster R-CNN bekannten, RPN und den RoI-Algorithmus übergeben. Das Ergebnis sind dann individuelle RoI Feature-Maps, welche von voll verknüpften Neuronalen Netzen zur Ermittlung der Klasse, Bounding Box Position und der Masken-Erzeugung verwendet werden, wobei der gesamte Ablauf des Modells in Bild 20 dargestellt ist. Durch diese erneute Verbesserung konnte die Erkennungsgenauigkeit weiter gesteigert und neue Funktionen in das Modell integriert werden [172, 173].

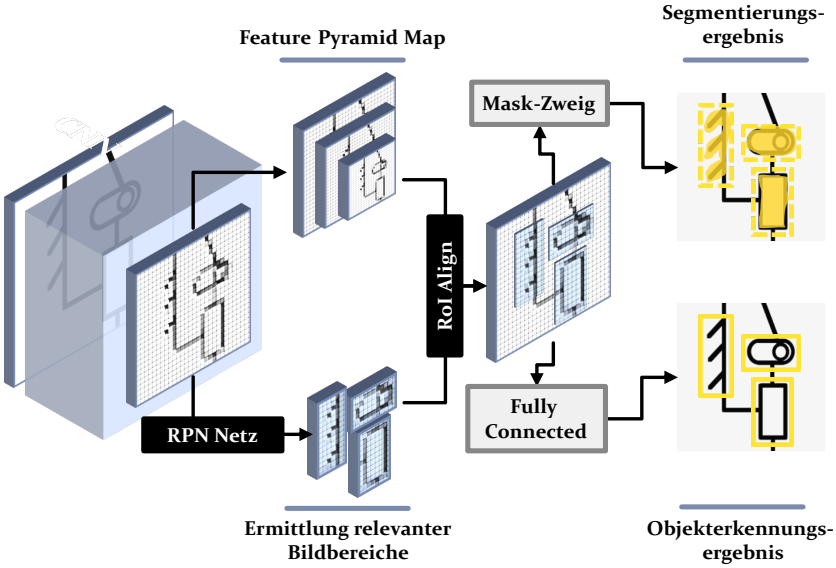


Bild 20: Ablaufübersicht für das MASK-R-CNN-Modell nach [183]

2.2.2.3 Auswertemetriken

Damit die verschiedenen Verfahren aus dem vorherigen Kapitel auch untereinander verglichen werden können, sind über die Jahre verschiedene Vergleichsmetriken entwickelt worden. Die meisten sind im Zusammenhang mit neuen Datensätzen oder Wettbewerben etabliert worden. Grundsätzlich können drei Kriterien für die Bewertung genutzt werden, die Geschwindigkeit in Frames per Second (FPS), die Precision und der Recall. Dabei hat sich

die Average Precision (AP) als wichtiger Ergebniswert herauskristallisiert, da dieser Precision und Recall verbindet [143, 153, 157]. Die drei Metriken sind mit den nachfolgenden Formeln erklärt.

$$Precision = \frac{True\ positives}{True\ positives + False\ positives} \tag{1}$$

$$Recall = \frac{True\ positives}{True\ positives + False\ negatives} \tag{2}$$

$$AP = \int_0^1 p(r)dr \quad \text{mit } p(r) = \max_{\tilde{r}>r} p(\tilde{r}) \tag{3}$$

Damit repräsentiert der AP Wert das Precision/Recall (P/R) Diagramm (siehe Bild 85 im Anhang), wobei die Precision an verschiedenen Recall Werten interpoliert wird. Hierbei können sich die AP unterscheiden, ob an 11 [143], 101 [170] oder über alle Werte interpoliert wird [171]. Der AP Wert wird für eine Objektklasse bestimmt, um eine Aussage über den gesamten Datensatz zu treffen. Dazu muss im Anschluss der Mittelwert aller AP Werte bestimmt werden, was als mean Average Precision (mAP) bezeichnet wird, siehe Formel 4.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{4}$$

Im Gegensatz zur Bildklassifizierung fließt bei der Objekterkennung neben der Klasse auch die Position der Bounding Box in die Bewertung mit ein. Dazu ist der Intersection over Union (IoU) Wert relevant, welcher die Überlappung der idealen Bounding Box mit der detektierenden Box bestimmt. In Bild 21 und Formel 5 ist die Erzeugung des Wertes dargestellt.

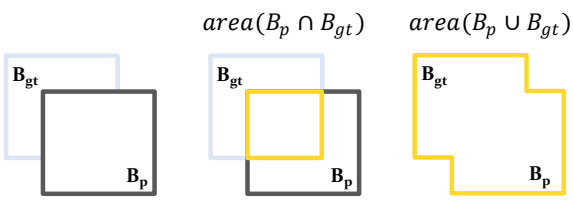


Bild 21: Bestimmung des IoU Wertes für die Auswertung von Objekterkennungsverfahren

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}$$
$$\text{mit } B_p = \text{ Bounding Box predicted} \qquad (5)$$
$$\text{mit } B_{gt} = \text{ Bounding Box ground truth}$$

Bei einer ideal vorhergesagten Bounding Box beträgt der IoU 1, bei keiner Überlappung 0. In der Auswertung entscheidet der IoU-Schwellwert, ob eine erkannte Bounding Box als richtig oder falsch eingeordnet wird. Eine Anwendung des Wertes ist in Bild 22 dargestellt.

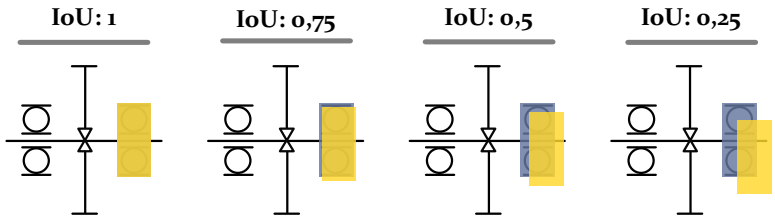


Bild 22: Beispiele für unterschiedliche IoU Werte

Falls ein kritischer IoU Wert unterschritten wird, gilt die erkannte Box als falsch, unabhängig von der zugeordneten Klasse. Der Schwellwert wird dann in die Auswertemetrik übernommen, z. B. mAP50, was der mAP bei einem IoU von 0,50 entspricht.

Daher kann der AP Wert in Kontext verschiedener Wettbewerbe unterschiedlich definiert sein. Für den Pascal VOC Datensatz ist der IoU für die AP auf 0,5 festgelegt, wogegen für den COCO-Datensatz der Mittelwert über zehn IoUs gebildet wird (0,5-0,95 in 0,05 Schritten) [157].

Die genauen Unterschiede bei der Bestimmung der mAP Werte sowie weitere weniger relevante Auswertegrößen sind in [171] ausführlich aufgeführt und beschrieben. Eine Beispielberechnung des mAP Wert für ein erkanntes Bild ist im Anhang unter Kapitel A.1 genauer erläutert und mit Ergebnissen dargestellt.

2.2.2.4 Anwendung in der Produktentwicklung

Nach der Vorstellung verschiedener Verfahren, Datensätze und Auswertungsmöglichkeiten liegt in diesem Kapitel der Fokus auf der Anwendung diverser Verfahren im Umfeld der Produktentwicklung. Grundsätzlich sind die Anwendungsmöglichkeiten sehr breit gefächert. In [151, 159, 172] wird eine Vielzahl

an allgemeinen Möglichkeiten aufgeführt: Schrifterkennung, Autonomes Fahren, Schilderkennung, Tracking Systeme, Gesichtserkennung, Bewegungsanalyse, Fußgängererkennung, Auswertung medizinischer Bilder, Bildersuche, Fertigungstechnik, Robotik oder Lebensmittelerkennung.

Viele Methoden der Objekterkennung finden in der Produktionstechnik und Montage Anwendung, vor allem zur Planung, Steuerung und Qualitätssicherung von Fertigungsprozessen [186]. Durch die vielen Vorarbeiten zur Erkennung von Schrift und Zeichen (u. a. Handschrifterkennung auf Briefen oder Schildererkennung) konnten diese auch für die Produktentwicklung adaptiert werden. Dabei werden hauptsächlich Algorithmen aus der Objekterkennung für die Nutzbarmachung von Skizzen und Zeichnungen eingesetzt. Über dieses Themenfeld erfolgt in dem folgenden Kapitel eine genauere Vorstellung.

Symbolerkennung
Die Grundmotivation der Symbol- und Zeichnungserkennung in den Ingenieurwissenschaften ist schon lange existent, weshalb die ersten Vorgehen bereits zu Beginn der 1980er Jahre veröffentlicht wurden. Diese Methoden basieren meist auf Regeln zur Erkennung der jeweiligen Symbole. In den neueren Veröffentlichungen nach [187] wird der gesamte Digitalisierungsablauf grundsätzlich in zwei größere Abschnitte gegliedert, siehe Bild 23. Dabei wird zwischen der Symbolerkennung und der Kontextualisierung von Erkennungsergebnissen unterschieden. Die Erkennung umfasst dabei das Lokalisieren und Klassifizieren der verschiedenen Objekte einer Zeichnung. Hierbei unterscheiden sich je nach Aufgaben und Umfang die Algorithmen deutlich voneinander. Bei der Kontextualisierung ist das Ziel, die erkannten Symbole möglichst automatisiert weiterzuverwenden, beispielsweise um nach ähnlichen Zeichnungen zu suchen, verknüpfte Dokumente zu finden oder die Zeichnung auf Fehler zu überprüfen.

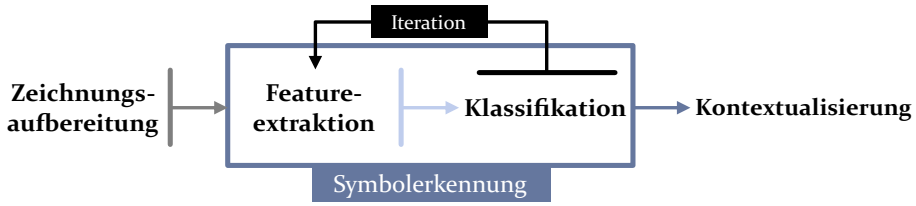


Bild 23: Ablauf und Einordung des Digitalisierungsvorgehen für Zeichnungen im Ingenieurwesen nach [187]

Ein allgemeiner Überblick der entwickelten Verfahren wird in Bild 24 gegeben. In diesen zwei Grafiken sind die Ergebnisse einer Literaturrecherche von ca.

200 Veröffentlichungen aus dem Bereich der Symbolerkennung im Ingenieurwesen aufgeführt. Dabei sind alle Verfahren anhand ihrer Einsatzdomäne, der Nutzung von Machine oder Deep-Learning sowie der Art des Datensatzes eingeordnet worden, die gesamte Tabelle mit allen Referenzen ist im Anhang in Tabelle 14 zu finden.

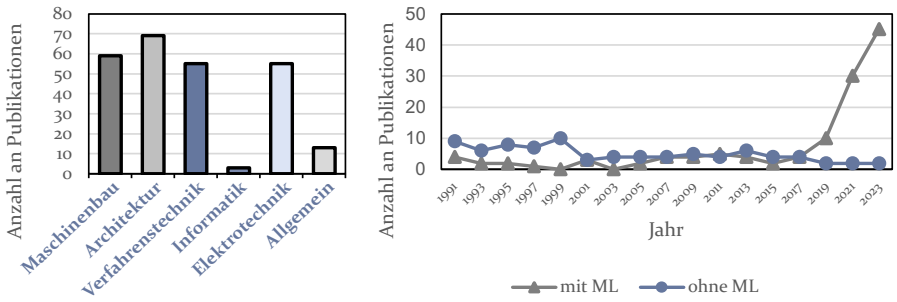


Bild 24: Vorstellung der Ergebnisse der Literaturrecherche mittels einer Domänenverteilung und Darstellung des zeitlichen Verlaufs

Zuerst ist die Verteilung anhand der Ingenieurdomäne dargestellt. Das Balkendiagramm zeigt deutlich, dass viele Verfahren für den Maschinenbau, Architektur, Elektrotechnik und Chemie bzw. Verfahrenstechnik (P&ID) entwickelt wurden. Allgemein gültige Methoden und der Bereich Computer Science haben deutlich weniger Beiträge.

Neben der Domänenverteilung ist die Anzahl an Veröffentlichungen über den zeitlichen Verlauf in zwei Jahresschritten dargestellt. Zusätzlich ist zwischen Verfahren mit Machine- oder Deep-Learning (mit ML) und ohne (ohne ML) differenziert worden. Dabei fällt deutlich auf, dass der grundsätzliche Ansatz über sehr viele Jahre der Einsatz von regelbasierten Algorithmen und Verfahren war. Erst ab der Mitte der 2010er Jahre ist ein deutlicher Anstieg der Learningverfahren zu verzeichnen. Die Steigerung an Veröffentlichungen im zeitlichen Verlauf lässt sich mit der Entwicklung der CNN-Verfahren und der zunehmenden Rechnerleistung erklären, die bereits in Kapitel 2.2.1.4 ausführlich beschrieben wurden.

Neben der zeitlichen Betrachtung ist ein weiterer Vergleich an Veröffentlichungen in dem Sunburst-Diagramm in Bild 25 illustriert. Dieses zeigt eine Gegenüberstellung der Anzahl an Veröffentlichungen für die verschiedenen Kategorien. Die Analyse zeigt, dass die meisten Verfahren sich mit der Symbolerkennung beschäftigen und die Kontextualisierung weniger im Fokus steht. Zudem ist über die gesamte Recherche der Anteil an Verfahren mit und ohne ML-Einsatz relativ ähnlich verteilt.

Allerdings haben die Verfahren mit Machine- oder Deep-Learning den Nachteil eines großen Bedarfs an gelabelten Daten für das Training, weshalb in Bild 25 eine Aufschlüsslung der verschiedenen Lösungsansätze im äußersten Kreis gelistet ist. Bei genauer Betrachtung der Verfahren mit Machine-Learning fällt auf, dass viele auf einen eigenen Datensatz zurückgreifen. Dabei können Industriepartner mit Zeichnungen unterstützen oder mittels Datenbanken und Scraping Verfahren ein eigener Datensatz erzeugt werden. Dieser muss allerdings im Anschluss noch händisch gelabelt werden, damit die Daten für das Training nutzbar werden.

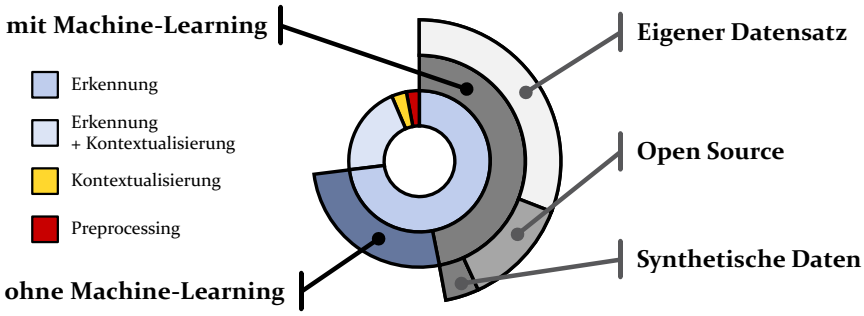


Bild 25: Einordnung der Verfahren anhand der verwendeten Datengrundlage

Eine weitere Möglichkeit ist die Nutzung von Open-Source Datensätzen. Hier sind beispielsweise für die Elektrotechnik der GREC'03 Datensatz [188] und für die Architektur der BRIDGE [189] und SESYD [190] zu nennen. Mit Hilfe dieser gelabelten und dementsprechend aufbereiteten Datensätze ist es möglich, ein Modell vergleichbar zu trainieren. Allerdings gibt es hier im Bereich des Maschinenbaus noch keinen veröffentlichten Datensatz, der den Vergleich von entwickelten Algorithmen ermöglicht.

Ein kleiner Teil der vorgestellten Arbeiten verwendet einen synthetischen Datensatz zum Training des Modells. Dieses Vorgehen bietet den Vorteil, dass die Datenerzeugung beeinflusst werden kann und somit das Modell dementsprechend adaptierbar ist. Neben dem hohen Aufwand für den initialen Aufbau eines synthetischen Datensatzes ist auch die Überprüfung des trainierten Modells schwieriger. Hierfür wird jeweils ein kleiner, von der synthetischen Erzeugung unabhängiger Datensatz benötigt, mit dem die Übertragbarkeit des Modells nachweisbar ist.

Kontextualisierung

Dieser Prozessschritt baut auf den erkannten Symbolen auf und nutzt diese für weitere Operationen. Laut [187] umfasst die Kontextualisierung das Design und die Umsetzung eines Systems oder Methode, welches die Infor-

mation einer oder mehrerer technischer Skizzen in ein Programm oder Hilfstool überführt. Die Anwendungsmöglichkeiten sind dementsprechend stark aufgaben- und domänenabhängig.

Nach [187] ist eine häufige Methode zur Weiterverarbeitung der erkannten Symbole eine Netlist. Diese beschreibt einen Graph, bei dem die Symbole als Knoten repräsentiert werden und die Kanten die Verbindungen zwischen diesen darstellen, wie in Bild 26 vereinfacht dargestellt. Beispiel-Anwendungen, die eine Netlist verwenden, sind u. a. [191–194].

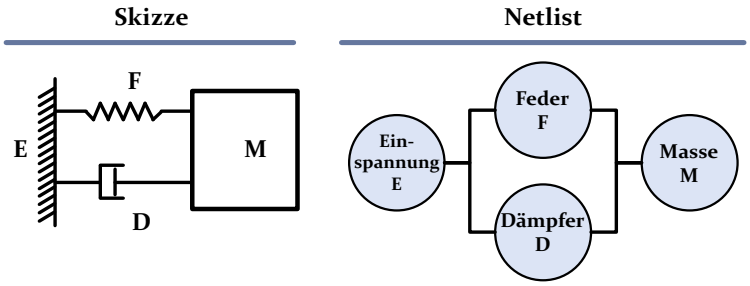


Bild 26: Beispiel für den Netlist-Ansatz zur Überführung der Symbole für einfache Simulationen nach [191]

Diese Umwandlung ermöglicht weitere Ansätze, welche Zeichnungen auf Fehler analysieren [195] oder mittels Graph-Matching nach ähnlichen Skizzen suchen [196]. Zudem beschäftigen sich Verfahren mit der Kopplung von technischen Zeichnungen und 3D-Modellen. Eines der ersten Systeme ist das Celesstin [197], welches gescannte Zeichnungen mittels Vektorisierung in eine Vektorgrafik überführt und anschließend in CAD-Bauteile. Ein aktueller Beitrag zur Vektorisierung und Rekonstruktion von Zeichnungen ist die Methode von Alwan [198]. Ein anderes Beispiel für die Kontextualisierung ist das von Banerjee et al. [199] entwickelte Vorgehen zur automatischen Verknüpfung verschiedener Architekturzeichnungen. Für das System wurde eine Kombination aus Object Character Recognition (OCR) und der Hough-Kreis-Erkennung verwendet. [200] hat ein Verfahren entwickelt, welches in der Lage ist, einen vorhandenen Möbelbestand in eine neue Wohnung zu übertragen und mögliche Raumausstattungen vorzuschlagen. Die Umsetzung erfolgt mittels Heuristiken gekoppelt mit verschiedenen Kosten-Funktionen.

Weitere Objekterkennungs-Anwendungsfälle
Neben der Symbolerkennung und Kontextualisierung sind zusätzliche Methoden zum Einsatz von Objekterkennungsalgorithmen erstellt worden. Beispielsweise das System von Müller et al. [201], welches durch einen Objekterkennungsalgorithmus fehlerhafte Bauteilauslegungen in 2D-Schnittbildern

erkennt. Im Fokus standen dabei eine gießgerechte Konstruktion ohne Lunkerbildung. Die Schnittbilder sind an einen YOLO-Algorithmus übergeben worden, welcher die ungünstigen Materialanhäufungen erkennen und dementsprechend markieren soll. Eine Weiterentwicklung von CNN für 3D-Objekte setzt [202] in seiner Methode zur Erkennung von Bauteilfunktionen ein. Dabei wird die CNN-Architektur so adaptiert, dass Voxel als Input verwendet werden können. Diese Beispiele zeigen, dass die Methoden für Bilder auch auf 3D-Objekte und deren eigene Formate übertragen werden können. Dieser Forschungsbereich soll im nachfolgenden Kapitel ausführlicher dargelegt werden.

2.2.3 Geometrie und Machine-Learning

Der Einsatz von Machine-Learning-Techniken in Kombination mit geometrische Daten ist in den letzten Jahren sehr stark angestiegen. Die einfachere Verfügbarkeit an 3D-Daten durch neue Technologien wie 3D-Scannern, LIDAR oder RGB-D Kameras (z. B. Microsoft Kinect) [83, 84] und der öffentlichen Bereitstellung mittels Datenbanken (thingiverse, traceparts oder ModelNet) [203, 204] ermöglichten eine erfolgreiche Anwendung der neuen Algorithmen [205]. Die Verfahren werden unter dem Begriff datengetriebene Geometrieanalyse zusammengefasst und haben sich aus dem Feld der mathematischen Geometrieanalyse [206] entwickelt [207, 208].

Im Gegensatz zu tabellarischen oder vektoriellen Daten sind Geometrien deutlich schwieriger mit datengetriebenen Algorithmen zu verknüpfen. Dies liegt zum einen an der Unregelmäßigkeit von 3D-Daten und zum anderen an den vielen Darstellungsarten [203]. Neben einer Einordnung der Methoden anhand der verarbeitbaren Formaten kann auch nach dem Ziel der Anwendung unterschieden werden. In [207] werden als Anwendungsfelder die Ähnlichkeitssuche, Segmentierung, Rekonstruktion, Synthese, Matching, Ausrichtung und Erkennung aufgeführt. Bild 27 stellt die verschiedenen Anwendungsmöglichkeiten und -formate für geometrische Daten vor.

Die prinzipielle Pipeline zur Anwendung der Methoden gliedert sich in die Bereiche Datensammlung, Feature Extraktion, Training und Inferenz. Allerdings sind die genauen Abläufe stark von den Formaten und Anwendungen geprägt. Die Featureextraktion umfasst dabei die Transformation von 3D-Daten in einheitliche Formate, welche nachfolgend von Algorithmen genutzt werden können. Das Ergebnis der Umwandlung wird als Shape Deskriptor bezeichnet und kann auf vielfältige Weise erzeugt werden. Ausführliche Erklärungen und Beispiele sind in [208] dargelegt.

Im Rahmen dieser Arbeit sind vor allem zwei Bereiche interessant, die Erkennung von Bauteilen und die Ähnlichkeitssuche von Geometrien. Die beiden Aufgaben können nach dem oben gezeigten Grundprinzip durchgeführt werden und bieten für die Produktentwicklung ein hohes Verbesserungspotential. Zunächst wird die Erkennung von Bauteilen genauer vorgestellt, gefolgt von der Suche nach ähnlichen Geometrien.

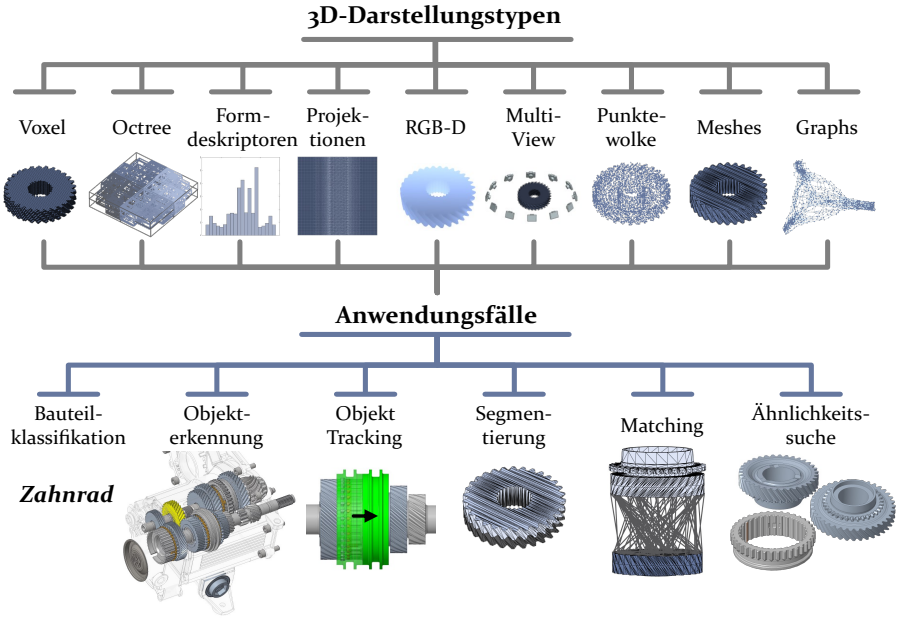


Bild 27: Übersicht der verschiedenen Geometrieformate und Anwendungsfelder nach [83, 84, 205, 207, 208]

2.2.3.1 Datensätze

Damit Methoden für die Klassifizierung oder Suche von Bauteilen angewendet werden können, sind Beispieldatensätze notwendig. Diese repräsentieren eine mögliche Sammlung an Geometrien, die z. B. in einer Datenbank oder PDM-System vereint sind. Sie sind notwendig, um die entwickelten Verfahren und Algorithmen unabhängig zu vergleichen. Sehr bekannte allgemeine Datensätze sind der Princeton Shape Benchmark [209], das ModelNet [210], PartNet [211] , ShapeNet [212] oder 3D WordNet [213].

Neben den allgemein öffentlich zugänglichen Datensammlungen existieren auch zeitlich begrenzte Wettbewerbe, für die ebenfalls Daten zur Verfügung gestellt werden. Ein sehr bekannter, jährlicher Wettbewerb ist die SHape REtrieval Challenge (SHREC), welche im Rahmen der Eurographics Konfe-

renz abgehalten wird. Weitere bekannte Datensätze mit spezifischen Aufgabenstellung sind: verformbare Geometrie [214], Skizzen-basierte Suche [215] oder die partielle Suche von deformierbaren Bauteilen [216].

Anwendung in der Produktentwicklung
Aber diese vorgestellten Datensätze sind nicht einfach für die Anwendung in der Produktentwicklung adaptierbar. Die Probleme der allgemeinen Datensätze sind [217, 218]:

- Bauteile aus dem Maschinenbau haben ein physisches Vorbild und unterscheiden sich durch ihre Form deutlich von den allgemeinen Objekten
- Die Einordnung in Bauteilklassen ist in der Produktentwicklung durch Randbedingungen vorgegeben und nicht subjektiv
- Kleine Details in der Geometrie können großen Einfluss auf die Bauteilklassen haben

Daher sind einige eigene Datensätze für die Produktentwicklung erstellt worden. Eine der ersten spezifischen Maschinenbau-Datensammlungen ist das „Actual Artifacts Dataset“ (AAD) von Bespalov et al. [217], welches sich aus vier verschiedenen Verzeichnissen zusammensetzt. Insgesamt ergeben sich daher 347 Bauteile in 16 Klassen für den AAD-Datensatz, unterteilt nach Fertigungsverfahren, Funktion, Lego-Bauteilen und Datenqualität. Ein moderner Vertreter ist der ABC-Datensatz von Koch et al. [219], welcher eine Million CAD-Bauteile enthält, allerdings sind diese nicht in Klassen eingeordnet. Dies schränkt die Einsatzmöglichkeiten stark ein, da kein überwachtes Training und auch keine Auswertung für z. B. Ähnlichkeitssuchen möglich ist. Dagegen enthalten die beiden nachfolgenden Datensätze gelabelte Geometriedaten, weshalb sie genauer vorgestellt werden.

Mechanical Component Benchmark
Ein sehr großer und aktueller Datensatz mit verschiedenen Bauteilen aus dem Maschinenbau ist der von Kim et al. [218] entwickelte „Mechanical Component Benchmark“ (MCB) Datensatz. Er besteht aus 58.696 verschiedenen Geometrien, die in 68 Klassen eingeteilt sind. Für eine bessere Vergleichbarkeit ist der Datensatz vorunterteilt in Test- und Trainingsdaten.

Der Datensatz ist in zwei verschiedene Sätze gegliedert: A und B. Dabei umfasst der MCB-A alle Geometrien, allerdings in einer relativ einheitlichen Orientierung, wogegen im MCB-B 18.038 Bauteile aus 25 Klassen zusammengefasst werden. Im B-Datensatz sind die Bauteile weniger gleichmäßig ausgerichtet, da die Bauteile aus dem A-Datensatz teilweise auf parametrischen Modellen basieren und somit gleich ausgerichtet sind. Eine kurze Übersicht der Bauteilklassen ist im Anhang in Tabelle 8 aufgeführt. Ein zufälliger Auszug der verschiedenen Bauteile ist in Bild 28 dargestellt.

Engineering Shape Benchmark

Ein weiterer Datensatz aus der Maschinenbaudomäne ist der „Engineering Shape Benchmark“ (ESB) von Jayanti et al. [220]. Die Geometrie-Sammlung umfasst insgesamt 866 überwiegend technische Bauteile oder Maschinenelemente. Die Teile werden nach ihrem allgemeinen Erscheinungsbild in drei Hauptkategorien sortiert: flach-dünnwandig, rechteckig-kubisch oder rotatorisch. Darüber hinaus sind die Bauteile in 45 Kategorien einsortiert, mit mindestens 4 und höchstens 58 Bauteilen pro Klasse. Im Anhang in der Tabelle 9 sind die Klassen und die Anzahl der Komponenten genau aufgelistet.

Die Teile werden als Netze im Datensatz gespeichert und in den Formaten obj und STL abgelegt. Die Bauteile sind nicht vororientiert, sondern werden in ungeordneter Form gesammelt. Bild 28 zeigt zufällig ausgewählte Bauteile aus dem MCB- und dem ESB-Datensatz. Diese Datensammlung wurde nicht nur wegen ihrer Spezialisierung auf mechanische Bauteile ausgewählt, sondern auch, weil sie seit 2005 verfügbar ist. Daher wurden viele Methoden mit dem ESB-Datensatz untersucht, was die Aussagekraft der Bewertung erhöht.

Wie bereits erwähnt kann die Einordnung der mechanischen Bauteile in definierte Gruppen eine Herausforderung darstellen. In diesem Zusammenhang wurde von Jagadeesan et al. [221] eine Studie durchgeführt, in der mittels Crowdsourcing die Klassentrennung für flache und dünnwandige Bauteile im ESB-Datensatz untersucht wurde. In dieser Umfrage wurden die Teilnehmer anonym aufgefordert, alle Komponenten selbständig in Gruppen zu ordnen. Die Ergebnisse der Studie zeigten, dass die Klassen des Datensatzes gut mit der Einteilung der Teilnehmer übereinstimmen. Es wurden einige Komponenten unterschiedlich zugeordnet, aber das Gesamtfazit war trotzdem positiv gegenüber der definierten Klasseneinteilung. Die Einordnung ist essentiell für überwachte Lernverfahren, damit diese die entsprechende Bauteilklasse für unbekannte Geometrien ermitteln können. Dieses Forschungsfeld wird unter dem Begriff Bauteilerkennung zusammengefasst.

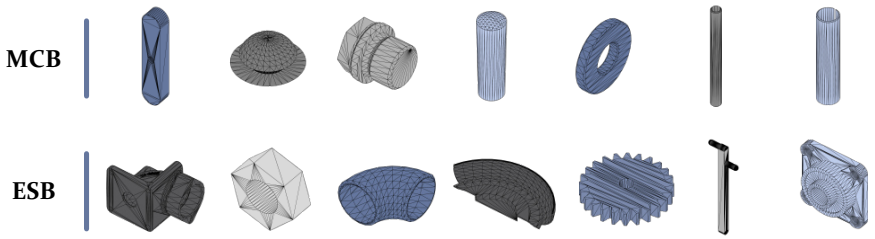


Bild 28: Zufällige Auswahl an Bauteilen des MCB- und ESB-Datensatzes

2.2.3.2 Bauteilerkennung

Die Idee der Klassifizierung besteht darin, einem gegebenen Input ein vordefiniertes Klassenlabel zuzuweisen. Folglich handelt es sich um eine überwachte maschinelle Lernmethode, die ein breites Anwendungsspektrum hat, von der Bilderklassifizierung, über Zeitsequenzen oder Textdaten. Dieses Funktionsprinzip lässt sich auch auf 3D-Geometrien übertragen, wobei der Input die 3D-Geometrie ist und als Ergebnis die Kategorie des Bauteils bestimmt wird.

Die verschiedenen Methoden unterscheiden sich durch das Input-Geometrieformat. Eine kurze Übersicht bekannter Formate ist in Bild 29 dargestellt, mit den Formatkategorien Voxeln, Punktewolken, Bilder oder Oberflächennetze. Dabei bieten die Oberflächennetze den Vorteil, dass diese sich in alle genannten Formate umwandeln lassen. Für alle Formatarten werden nachfolgend kurz die Eigenschaften und entwickelte Verfahren zur Erkennung von Objekten vorgestellt.

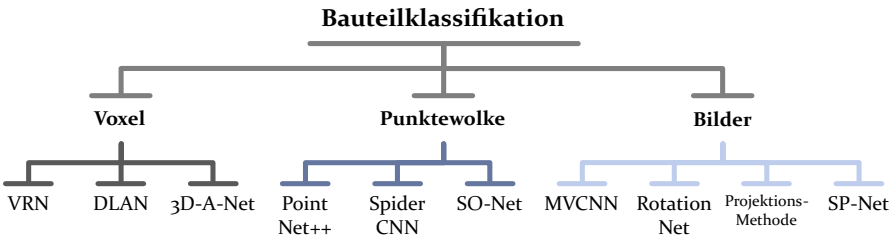


Bild 29: Allgemeine Einteilung der Bauteilerkennungs-Verfahren nach dem Input-Format

Voxel

Für Voxel-Geometrien hat [222] einen neuen Ansatz entwickelt, der auf der VoxNet-Architektur basiert und diesen um Inception-Module und ein relativ flaches Netzwerk erweitert. Dieses neue Verfahren wurde aufgrund der Integration der Inception-Module als Voxception-ResNet (VRN) bezeichnet. Der Ansatz von Furuya et al. [223] kombiniert die Beschreibung von 3D-Objekten über lokale Merkmale mit rotationsnormalisierten Rastern zu einer neuen Methode namens „Deep Local feature Aggregation Network“ (DLAN). Eine weitere Möglichkeit Bauteile in Voxel-Repräsentation zu erkennen ist das 3D-A-Net, welches von Ren et al. [224] veröffentlicht wurde. Dieser Ansatz verwendet eine schichtweise Unterteilung der Geometrie als Eingabe für den Feature Generator. Die Übertragung in Features erfolgt für jedes Schichtbild mittels einer Kombination aus CNN- und RNN-Netzwerken. Nach der Umwandlung wird das Ergebnis an ein weiteres Netzwerk übermittelt, welches die Klassifizierung des Feature Vektors übernimmt.

Punktewolke
Im Gegensatz zu Voxeln sind Punktewolken unregelmäßig und ungeordnet im Raum verteilt, was die Anwendung von normalen CNNs unmöglich macht. Dieses Problem wurde vom PointNet adressiert, einer von Qi et al. [85] entwickelten Deep-Learning-Architektur, die Merkmale aus Punktewolken extrahiert. Die beschriebene Methode ist nicht nur in der Lage, Bauteile zu erkennen, sondern auch zu segmentieren. Die Bauteilerkennung basiert auf zwei Schritten: der Eingabetransformation und der Merkmalserzeugung sowie der Klassifizierung der globalen Merkmale. Für die Eingangstransformation wird eine Kombination aus Transformationsnetzen und Matrixmultiplikatoren verwendet, die über Max-Pooling in einen globalen Merkmalsvektor überführt werden. Dieser Vektor wird dann von einem Neuronalen Netz klassifiziert und das entsprechende Label ermittelt. Das beschriebene Verfahren wurde zum PointNet++ [225] weiterentwickelt, mit dem Ziel lokale Feinheiten zu erkennen. Im Gegensatz zum Pointnet ist bei der Weiterentwicklung ein hierarchischer Ansatz gewählt worden, welcher die Pointnet Architektur wiederholt anwendet. Ein weiterer Ansatz für Punktewolken wurde von Xu et al. [226] erzeugt, das SpiderCNN. Diese Architektur besteht aus sogenannten SpiderConv-Einheiten, die die Faltungsoperationen auf unregelmäßige Punktmengen erweitern, indem sie eine Gruppe von Faltungsfiltern parametrisieren. Diese Filter sind definiert als das Produkt aus einer einfachen Stufenfunktion zur Erkennung lokaler Merkmale und einem Taylor-Polynom, das die Gewichte approximiert. Eine Adaption von Self-Organizing Maps (SOM) auf Punktewolken hat [227] umgesetzt und unter dem Namen SO-Net veröffentlicht. In dieser Architektur wird die Punktewolke über eine k-nearest SOM in eine Knotenmatrix überführt, welche über eine zusätzliche Max-Pooling Operation in einen globalen Feature Vektor vereinfacht wird. Mit Hilfe dieses Vektors kann danach entweder eine Segmentierung oder Erkennung über Neuronale Netze erfolgen.

Bilder
Neben der Verwendung von Voxeln und Punktewolken als Eingabe können auch Bilder ausgewählt werden. Die vorhandenen Methoden rendern üblicherweise mehrere Bilder aus verschiedenen Ansichten, die dann mithilfe von mehreren CNNs klassifiziert werden. Eine bekannte Methode aus dieser Kategorie ist das Multi-View-CNN (MVCNN) von Su et al. [228]. Bei diesem Verfahren werden die erstellten Ansichten des 3D-Objekts durch die erste CNN-Schicht geleitet, um ansichtsbezogene Merkmale zu extrahieren. In der anschließenden View-Pooling-Schicht werden die zuvor extrahierten Features aus allen Ansichten kombiniert und durch ein zweites CNN geschleust, das für die Teileklassifizierung zuständig ist. Ein anderer, bildbasierter Ansatz beruht nicht nur auf Bauteilen, sondern auch auf Orientierungsvorhersa-

gen, das von Kanezaki et al. [229] beschriebene RotationNet. Mit Hilfe des unüberwachten Lernens werden aus zwölf festen Kameraperspektiven die drei besten Objektpositionen ermittelt, um die Klassifizierungsgenauigkeit zu optimieren. Im Gegensatz zu den vorhergehenden Techniken sind die auf Bilddaten basierenden Methoden nicht in der Lage, 3D-Geometrien zu segmentieren und haben daher weniger universelle Funktionen.

Ein weiteres Vorgehen nach diesem Prinzip der Bildtransformation ist das SP-Net von Yavartanoo et al. [230]. Hierbei wird über stereographische Projektionen die Input Geometrie in ein Bild umgewandelt. Anschließend erfolgt über ein adaptiertes CNN die Klassifizierung des Inputs. Zur Verbesserung der Erkennungsgenauigkeit wurde noch die Ausrichtung der Bauteile über Ensemble- und Selektionsverfahren verbessert. Ein weiteres Beispiel für die Nutzung von Projektionen ist die Methode von Spruegel et al. [231], die sich auf die Identifizierung von FE-Simulationsbauteilen fokussiert. In dieser Methode wird ein Bild nicht in ein CNN, sondern in ein einfaches neuronales Netz übergeben. Dazu wurde die Geometrie mittels Projektionsmethode in eine Matrix übertragen, die dann in einen Vektor umgewandelt wurde, welcher als Input für das neuronale Netz dient. Die Kombinationen verschiedener Inputformate zur Einordnung von Geometrien ist von You et al. [232] präsentiert worden. In diesem Verfahren werden Punktewolken und Multi-View miteinander zu einem Netzwerk kombiniert. Zur Umsetzung werden die beiden Inputs durch eigene Netzwerke aufbereitet und im Anschluss über ein Embedding Attention Fusion Modul miteinander zu einem Ergebnis verknüpft.

Auswertemetriken

Neben den Datensätzen und Methoden sind für die Analyse und Bewertung der verschiedenen Verfahren die Auswertemetriken entscheidend. Für Klassifikationsaufgaben hat sich als Standard die Wahrheitsmatrix etabliert, siehe Bild 30.

Die relevanten Kennwerte der Wahrheitsmatrix, die das Ergebnis einer Klassifikation für eine ausgewählte Klasse beschreiben, umfassen [129, 233, 234]:

- **True positives** (TP): Anzahl an positiven Objekten, die korrekt klassifiziert wurden.
- **True negatives** (TN): Anzahl an negativen Objekten, die korrekt klassifiziert wurden.
- **False positives** (FP): Anzahl an positiven Objekten, die falsch klassifiziert wurden.
- **False negatives** (FN): Anzahl an negativen Objekten, die falsch klassifiziert wurden.

		vorhergesagte Klasse		
		positiv	negativ	Gesamt
tatsächliche Klasse	positiv	*TP*	*FN*	*P*
	negativ	*FP*	*TN*	*N*
	Gesamt	*P´*	*N´*	*P+N*

Bild 30: Wahrheitsmatrix für Klassifikation nach [233, 235]

Aus diesen vier Werten können verschiedene Kennzahlen für eine Klassifikation berechnet werden, wobei die Genauigkeit (Formel 6) und der F1-Score (Formel 7) häufig eingesetzt werden [235].

$$Genauigkeit = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$F1 - Score = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{7}$$

Abhängig von Datensatz und Anwendungsfall sind weitere Metriken im Anhang in Tabelle 10 aufgelistet. Somit ermöglichen diese Metriken einen Vergleich der vorher präsentierten Methoden über die Erkennungsgenauigkeit. Eine aktuelle Untersuchung [218] zeigt dabei, dass das RotationNet aktuell die beste Erkennungsgenauigkeit von 97,35 % für Bauteile aus dem Maschinenbauumfeld liefert. Die genauen Werte für das RotationNet und andere Methoden sind im Anhang unter Tabelle 11 aufgeführt.

Anwendung in der Produktentwicklung

Das Spektrum für die Verwendung von Bauteilerkennung in der Produktentwicklung ist sehr breit gefächert, lässt sich aber grob in zwei Bereiche gliedern. Der erste umfasst das „fertigungsgerechte Konstruieren", hier werden häufig Methoden adaptiert, um das Fertigungsverfahren oder die generelle Fertigbarkeit von Produkten zu ermitteln. Dagegen ist der zweite Aspekt mit der Unterstützung im Entwicklungsprozess beschäftigt, indem Standardaufgaben automatisiert und das verknüpfte Wissen zu spezifischen Bauteilen genutzt werden kann.

Fertigungsgerechtes Konstruieren

Im Themenfeld des fertigungsgerechten Konstruierens ist ein Unterstüt-

zungswerkzeug von Peddireddy et al. [236] entwickelt worden, welches die Fertigbarkeit von Bauteilen erkennt. Dazu wird ein 3D-CNN trainiert, welches das erzeugte Bauteil den Fertigungsoperationen Drehen, Fräsen oder nicht-fertigbar zuordnet. In der gleichen Domäne hat [237] ein Vorgehen entwickelt, welches die Designparameter für Punktschweißverbindungen im Fahrzeug vorhersagt. Dazu mussten zunächst die Bauteile erkannt werden, damit die passenden Parameter teilespezifisch vorhergesagt werden können. Das Vorgehen wurde anhand von zwei Fahrzeugdatensätzen mit Toyota- und Audi-Bauteilen getestet. Ein ähnliches Ziel wurde von Ip und Regli [238] verfolgt, allerdings wurde das Fertigungsverfahren mittels der Oberflächenkrümmung ermittelt.

Neben der Fertigung wird auch die Montage von Baugruppen mittels Bauteilerkennung durch eine System von Neb et al. [239] automatisch erkannt und zugeordnet. Dabei wird versucht, dass Automatisierungspotential einer Baugruppe zu ermitteln, indem beispielsweise die Separierung der Bauteile abhängig von der Geometrie vorhergesagt wurde.

Unterstützungswerkzeug im Entwicklungsprozess

Die entwickelten Verfahren zur Erkennung von Bauteilen werden auch für das Ausarbeiten der Geometrie in CAD-Software zur Unterstützung eingesetzt. Das Vorgehen von Hyvärinen [240] erkennt automatisch die notwendigen Zusammenbaubedingungen in der CAD-Umgebung. Die Umsetzung dazu erfolgt über eine Adaption von Graph-CNN-Netzwerken, welche die entsprechende Randbedingung als Ergebnis liefern. Auch die Methode von Dworschak et al. [241] setzt auf die Bauteilerkennung, um bereits vorhandene Bauteile zu erkennen und dementsprechend das Vorgehen zu adaptieren. Das Ziel ist die Überführung von CAD-Geometrie in eine Ontologie, auf deren Basis weitere Automatisierungen des Produktdesigns erfolgen können. Bei der Tolerierung von Bauteilen soll die Methode von Qie et al. [242] unterstützen, indem automatisiert die Einordnung der einzelnen Bauteilflächen in die ISO-Invarianzklassen erfolgt. Diese zusätzliche Information kann nachfolgend für die Tolerierung des Bauteils verwendet werden. Das Framework von Jin [243] zielt auf die automatische Ableitung von Produktfamilien von Bildern ab. Ein Teilaspekt ist dabei die Erkennung der Unterbaugruppen bzw. Bauteilen, um diese im Anschluss mit anderen Varianten zu ersetzen und so verschiedene Produktkombinationen zu erzeugen. Die Anwendung der Bauteilerkennung auf Baugruppen ist von Lupinetti et al. [244] mittels einer Web Applikation umgesetzt worden. In diesem Tool können Baugruppen unabhängig von der CAD-Software analysiert und verglichen werden. Ein Teilschritt ist dabei die Erkennung der einzelnen Bestandteile zur Ableitung der allgemeinen Funktion der Bauteile bzw. Baugruppen.

Auch für die Baugruppensuche oder die schnelle Erzeugung neuer Ideen ist die Detektierung von Bauteilen hilfreich. Ein Vorgehen ist von Lupinetti et al. [245] vorgestellt worden, bei dem die Bauteilerkennung zur Unterstützung und Verbesserung der Suche eingesetzt wird. Auch für die Arbeitserleichterung beim Aufbau von FE-Simulationen kann die Bauteilerkennung genutzt werden. Die Methode von Spruegel et al. [231] erkennt Standard-Bauteile in FE-Simulationen mit dem Ziel, vordefinierte Lastfälle oder Vernetzungen direkt auf die erkannten Bauteile zu applizieren.

2.2.3.3 Bauteilsuche

Ein verwandtes Anwendungsfeld der Bauteilerkennung von Geometriedaten ist die Ähnlichkeitssuche (*engl. shape retrieval*). Das Ziel ist, abhängig vom gegebenen Input, ein möglichst ähnliches Bauteil zu finden, wodurch sich viele Vorteile im Produktentwicklungsprozess ergeben. So können vorhandene Daten viel besser genutzt und bereits existierende Geometriemodelle übernommen werden oder dienen als Grundlage für eine Überarbeitung. Somit wird das komplette Neu-Konstruieren von bereits bestehenden CAD-Bauteilen vermieden und dadurch Zeit und Geld eingespart. Zusätzlich kann auch die Ideen-Findung bzw. das Problemlösen von aktuellen Entwicklungen optimiert werden. Durch die schnelle Suche von ähnlichen Ergebnissen können neue Ideen und Impulse das aktuell existierende Design verbessern.

Problemstellung
Der allgemeine Begriff der Ähnlichkeit ist subjektiv und lässt sich auf vielfältige Weise definieren, je nach erworbenem Wissen, Entscheidungen oder dem Umfeld. Die Bedeutung hängt zudem von der jeweiligen Domäne ab, wie z. B. Filme, Bilder, Texte oder Musik [246]. Alle Sektoren vereint jedoch, dass Ähnlichkeit in den verschiedenen Bereichen immer mit vergleichbaren Merkmalen verbunden ist [247]. Da die Geometrie die Hauptkomponente der virtuellen Produktentwicklung ist, wird in dieser Arbeit primär die geometrische Ähnlichkeit betrachtet. Ein Beispiel für die Probleme, die sich aus der geometrischen Ähnlichkeit ergeben, wird von Berchtold [248] gegeben und in Bild 31 anhand von 2D-Geometrien dargestellt.

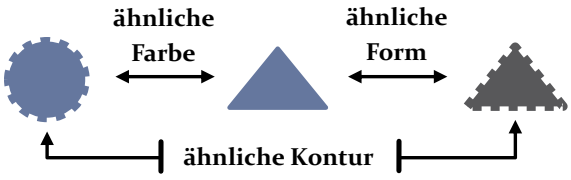


Bild 31: Ähnlichkeit in unterschiedlichen Formen nach [248]

In dieser Darstellung ist die Ähnlichkeit abhängig vom angestrebten Suchziel, sei es die Farbe, die allgemeine Form oder die Kontur der Objekte. Dieses Beispiel zeigt, dass es keine allgemeingültige Lösung gibt, sondern das Ergebnis vielmehr je nach Anwendung variiert. Damit ein Vergleich durchgeführt werden kann, muss aus dem ursprünglichen Format eine Ähnlichkeitsmetrik abgeleitet werden, mit der die Ähnlichkeit oder Unähnlichkeit zwischen zwei Produkten bestimmt wird. Viele Suchvorgänge, wie z. B. Internet-Suchmaschinen, arbeiten nach diesem Prinzip.

Für die Anwendung auf Geometrien wird diese Problemstellung nochmals aufwendiger, da hier weitere Unterscheidungen möglich sind. Ein Beispiel ist in Bild 32 aufgeführt, indem zwischen drei Arten der Ähnlichkeit unterschieden werden: die globale, partielle und lokale Ähnlichkeit. In der Abbildung sind die verschiedenen Arten für Baugruppen dargestellt, allerdings können diese genauso auf Bauteile übertragen werden.

Bild 32: Darstellung der verschiedenen Ähnlichkeitsarten für Baugruppen nach [249]

Bei der globalen Ähnlichkeit zwischen Geometrien wird das gesamte Bauteil für den Vergleich herangezogen. Die Suche von Teilbereichen in einer Gesamtgeometrie wird als partielle Ähnlichkeitssuche bezeichnet und basiert auf der Vorstellung, dass jede Form aus mehreren Teilgeometrien besteht. Betrachtet man den menschlichen Körper als Ganzes, so sind die Hände ein Beispiel für eine Teilgeometrie. Eine Hand kann jedoch weiter in einzelne Finger unterteilt werden. Diese genaue Definition der Unterkategorien verdeutlicht die Problematik der partiellen Geometriesuche. Die partielle Ähnlichkeit wird weiter unterschieden zwischen der part-in-whole und whole-to-whole Bauteilsuche. Für beide Arten sind in Bild 32 zwei Suchanfragen aufgeführt. Bei der ersten Unterkategorie wird nach dem Suchbauteil in einer gesamten Geometrie gesucht, wogegen die zweite Art die beiden Geometrien vergleicht, allerdings auf Basis der partiellen Ähnlichkeit. Die zweite Kategorie wird von Lupinetti et al. [245] auch als lokale Ähnlichkeit bezeichnet. Da der Fokus dieser Arbeit auf dem Vergleich der globalen Ähnlichkeit von Bauteilen liegt, wird dieses Prinzip in den folgenden Kapitel genauer erläutert.

Ähnlichkeitssuche mit Bauteilen

Im Zusammenhang mit der geometrischen Ähnlichkeit werden die Begriffe

Formbeschreibung (*engl. shape description*) und Formrepräsentation (*engl. shape representation*) allgemein als geometrische Beschreibung für einen Ähnlichkeitsvergleich verwendet. In der Literatur existieren unterschiedliche Definitionen für beide Begriffe. Im Rahmen dieser Arbeit wird die Definition von Loncaric [250] verwendet, welche die Formrepräsentation als eine nichtnumerische Repräsentation der ursprünglichen Geometrie definiert und die Formbeschreibung als eine numerische Beschreibung der Geometrie, auch Formdeskriptor (*engl. shape descriptor*) genannt. Dieser Formdeskriptor bildet die Grundlage für eine geometrische Ähnlichkeitssuche, da er die Geometrie in eine einheitliche und vergleichbare Struktur transformiert. Die Verfahren zur Erzeugung dieses Formdeskriptors sind vielfältig und unterscheiden sich in ihrer Funktion erheblich. Daher wird im folgenden Abschnitt der Ablauf einer Ähnlichkeitssuche allgemein erläutert und die verschiedenen Möglichkeiten zur Erzeugung des Shape-Deskriptors erklärt.

Generelles Vorgehen
Das Ziel der geometrischen Ähnlichkeitssuche ist die ähnlichste Geometrie als Antwort auf eine vorhandene Eingabe zu liefern. In [217] wird bei der Suche zwischen Volumen- und Oberflächenmodellen unterschieden. Aufgrund der in Kapitel 2.1.2.3 genannten Vorteile werden hauptsächlich Methoden für Oberflächennetze genauer betrachtet. Andere Einteilungsarten der Ähnlichkeitssuche können das Format des Suchinputs (z. B. Bilder oder Geometrie) oder die Verformbarkeit der Objekte (z. B. medizinische Anwendung oder Animationsfilme) sein [251].

Aufgrund der Unregelmäßigkeit von Oberflächennetzen wird in [252] das allgemeine Vorgehen bei der Ähnlichkeitssuche wie folgt beschrieben: Zunächst wird die Geometrie aufbereitet, dann abstrahiert und transformiert. Als Ergebnis wird schließlich ein Formdeskriptor erzeugt, der in einer mit der Originalgeometrie verknüpften Datenbank gespeichert wird und die Grundlage für die anschließende Ähnlichkeitssuche bildet. Der gesamte Prozess ist in Bild 33 dargestellt.

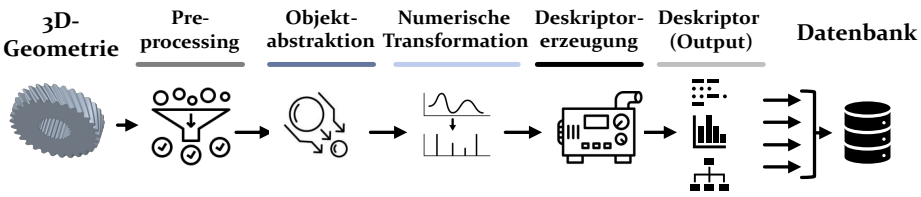


Bild 33: Allgemeines Vorgehen zur Geometrie-Ähnlichkeitssuche nach [252]

Verschiedene Verfahren und Methoden können für die einzelnen Teilschritte eingesetzt werden. In [252] werden die Möglichkeiten für die einzelnen

Schritte erläutert, so können z. B. in der Phase der Vorverarbeitung eine Translation, eine Skalierung, eine Rotation oder eine Entrauschung der Geometrie durchgeführt werden und auch Verfahren wie Remeshing sind an dieser Stelle denkbar. Im Anschluss kann die Ähnlichkeit zwischen den verschiedenen Formdeskriptoren durch die Berechnung des Abstands zwischen ihnen bestimmt werden. Dieser Vergleich ermöglicht es, den ähnlichsten Vektor und damit die passende Geometrie zu finden.

Zur Implementierung einer Ähnlichkeitssuche wurden bereits mehrere Frameworks entwickelt, beispielsweise von [253] und [254]. Die Vorgehensweise und das Prinzip der Ähnlichkeitssuche können anhand der Prozessarchitektur in Bild 34 nachvollzogen werden. Der passive Teil des Frameworks wird in der Regel im Vorfeld durchgeführt und enthält die vorhandene Datenbasis einschließlich der Bauteile, die in ihre Formdeskriptoren umgewandelt werden. Die aktive Suche anhand der Eingabegeometrie und die Ergebnisermittlung bilden den unteren Teil der Prozessarchitektur. Ein Vergleich zwischen der Geometrie und den Objekten in der Datenbank wird über die Formsignaturen oder Merkmale durchgeführt. Die Ergebnisse werden anschließend in geordneter Form angezeigt. [254] erweitert diesen Ansatz um eine webbasierte Suche und die Möglichkeit, mehrere Eingaben für die Suche zuzulassen. So sind neben Geometrie auch Schlüsselwörter oder Bilder möglich. Für den Vergleich von Geometrien sind verschiedene Methoden zur Berechnung der Formdeskriptoren entwickelt worden, die im Folgenden vorgestellt werden.

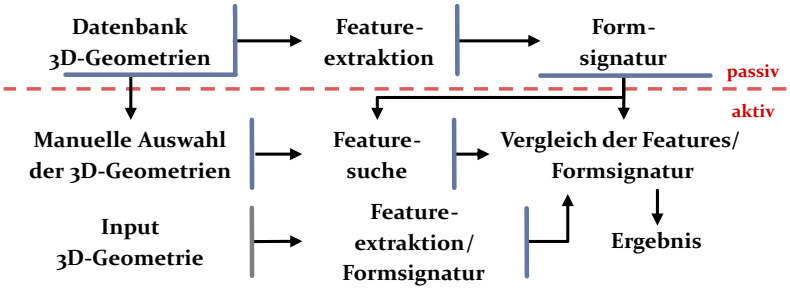


Bild 34: Beispiel-Framework für eine Ähnlichkeitssuchmethode nach [253]

Einordnung der Verfahren

In den letzten zwei Jahrzehnten wurden verschiedene Kategorisierungen der Formdeskriptorerzeugung vorgestellt, bekannte Beispiele finden sich in [15, 252, 254, 255]. Die Unterscheidung in dieser Arbeit basiert auf einer neueren Einteilung von Li et al. [251], wobei in vier Kategorien unterschieden wird: geometriebasierte, graphenbasierte, bildbasierte oder hybride Techniken. Im

Kontext der Produktentwicklung sind auch merkmalsbasierte und gruppierungsbasierte Verfahren (GT) anwendbar [256]. Diese benötigen jedoch in der Regel BRep-Daten als Input und sind daher weniger universell einsetzbar.

Geometrie Methoden
Geometriebasierte Verfahren nutzen die Verteilung geometrischer Elemente auf einem Bauteil zur Identifikation. Die Merkmale können global oder lokal für jede Geometrie erzeugt werden. Beispiele für globale Methoden sind Gestalt-Histogramme [257] und für lokale Methoden erweiterte Gaußsche Abbildungen [258].

Graph Methoden
Graphbasierte Methoden übertragen die Geometrie in verschiedene Graphstrukturen. Diese Methode wird häufig für BRep-Geometrien gewählt, z. B. verwendet [259] dieses Prinzip für STEP-Dateien. Nach [251] kann die Anwendung auf Oberflächennetze in Skelettgraphen und topologische Graphen unterteilt werden. Die Verfahren für Skelettgraphen abstrahieren die Geometrie in Abhängigkeit ihrer Attribute in einen Graphen, wie z. B. das Verfahren aus [260]. Topologische Graphen nutzen die Unterschiede in der topologischen Struktur der Geometrie zum Vergleich aus. Ein bekanntes Beispiel ist der Reeb-Graph aus [261], der auf der Morse-Theorie basiert.

Bild Methoden
Eine andere Form der Umwandlung stützt sich auf gerenderte Bilder der Geometrie als Grundlage für die Erzeugung des Formdeskriptors. Anschließend wird die Ähnlichkeit zwischen den erzeugten Bildern berechnet. Zu diesem Zweck wurden mehrere Methoden entwickelt, von denen einige durch Alignment- oder Multiview-Ansätze erweitert wurden. Ein sehr populärer Vertreter dieser Art ist der Light-Field-Descriptor [262]. Häufig eingesetzt wird der Bag-Feature-Ansatz [263], der mehrfach weiterentwickelt wurde, zum Beispiel durch KAZE [264] oder SIFT [265]. Eine detailliertere Zusammenfassung der ansichtsbasierten Methoden findet sich in [266].

Hybrid Methoden
Die letzte Kategorie umfasst die Kombination von zwei Ansätzen aus den drei vorher genannten Typen. Diese Gruppe wird als hybrides Verfahren bezeichnet und ist entsprechend breit gefächert. Ein Beispiel für ein hybrides Verfahren ist die PANORAMA-Methode von Papadakis et al. [267]. Dieser Ansatz verwendet Panoramabilder und daraus abgeleitete Parameter für die Geometriesuche. Ein anderes Verfahren ist DBNAA DERE [268], das verschiedene Methoden wie Formverteilung, Bounding Boxes und depth buffer kombiniert.

Machine-Learning Einsatz
Darüber hinaus haben neue Entwicklungen im Bereich des Maschinellen Lernens zu innovativen Ansätzen für die Geometriesuche geführt. Insbesondere die Entwicklung von Convolutional Neural Networks, einer wichtigen Variante des Deep-Learning (vgl. Kapitel 2.2.1.4), hat viele neue Methoden hervorgebracht.

Bei der Anwendung von Deep-Learning-Methoden auf die Geometrie werden Klassifikation und Retrieval oft gemeinsam behandelt. Dies ist möglich, indem der aus der Geometrie erzeugte Feature-Vektor an ein Klassifikationsmodell übergeben werden kann oder mit diesem eine Distanzberechnung für die Ähnlichkeitssuche initiiert wird. Im Allgemeinen ist es immer noch möglich, Deep-Learning-Methoden auf die gleiche Weise zu kategorisieren wie bereits dargestellt, wobei die Kategorie den Input für das Deep-Learning-Modell spezifiziert. So gilt beispielsweise das Multi-View CNN (MVCNN) [228] als ansichtsbasiertes Verfahren, da mehrere Bilder an ein Machine-Learning-Modell übergeben werden. Im Gegensatz dazu wird das PointNet [85] als geometrisches Verfahren charakterisiert, da die Punktewolke direkt als Input für das neuronale Netz dient. Anhand dieser beiden Kriterien wurde die Unterscheidung zwischen der Eingabe eines Deskriptors und der Art des Trainings (Label erforderlich, unüberwacht, kein Training erforderlich) für mehrere veröffentlichte Ansätze untersucht und kategorisiert.

Ergebnis der explorativen Literaturrecherche
Anhand der vorgestellten Kategorien Input-Format und Machine-Learning-Unterstützung ist eine explorative Literaturrecherche für Methoden zur Ähnlichkeitssuche durchgeführt worden. Die Auflistung umfasst insgesamt ca. 200 Publikationen, den Autor, Titel des Verfahrens, das notwendige Input-Format und die Trainingskategorie. Alle analysierten Verfahren und ihre jeweiligen Kategorien sind im Anhang in Tabelle 12 zu sehen. Eine Zusammenfassung der Literatur findet sich in dem Diagramm in Abbildung 35.

In diesem Diagramm ist die zeitliche Entwicklung auf der X-Achse aufgetragen und in vier Fünfjahreszeiträume unterteilt, die vor 2006 beginnen und 2022 enden. Die Anzahl der veröffentlichten Verfahren ist auf der Y-Achse aufgelistet. Die vier Kategorien des Arbeitsprinzips sind als gestapelte Balken dargestellt, wobei die graphbasierten Verfahren in gelb, die hybriden Verfahren hellblau, die bildbasierten in blau und die geometrischen dunkelblau eingefärbt sind.

Die Anzahl der durch die Lernverfahren getrennten Ansätze sind als Linien mit der farblichen Unterscheidung aufgetragen: „Label notwendig“ schwarz, „Unüberwachtes Lernen“ dunkelblau und „kein Training“ grau. Der Begriff „Label notwendig“ umfasst alle Methoden, die mehr Input als nur den Daten-

satz der Geometrien für die Ähnlichkeitssuche benötigen, dazu gehören User Feedback, überwachte, semi und weakly überwachte Methoden.

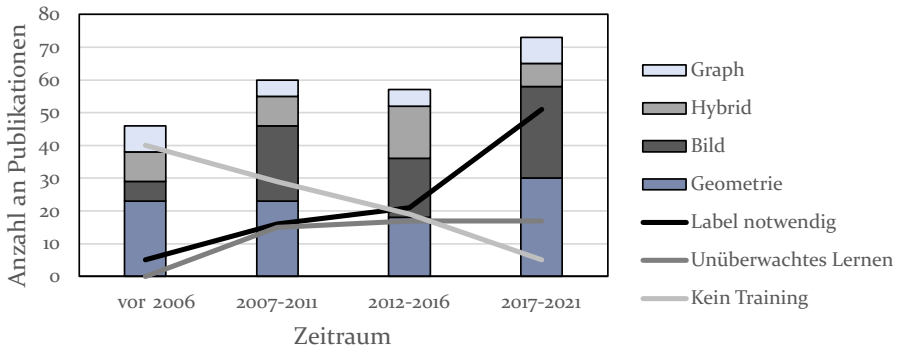


Bild 35: Überblick über die Anzahl der Veröffentlichungen im Laufe der Zeit, sortiert nach Kategorien

Das Diagramm veranschaulicht, dass die Zahl der Veröffentlichungen in diesem Bereich allgemein zunimmt, was auf ein anhaltendes und wachsendes Interesse an diesem Thema hinweist. In den Liniendiagrammen der verschiedenen Lernmethoden sind zwei Trends deutlich erkennbar: Zu Beginn der 2000er Jahre waren nur Methoden ohne Training etabliert und diese nahmen daher im Laufe der Zeit zu, bis ihre Anzahl mit dem Aufkommen des Maschinellen Lernens zurückging. Dies wird im Allgemeinen auf die höhere Leistung der neuen Computergenerationen und die bessere Zugänglichkeit der Algorithmen des Maschinellen Lernens zurückgeführt. Die Möglichkeit, Formklassifizierung und Datenextraktion in einem Ansatz zu integrieren, erklärt den Aufschwung des überwachten Lernens. Bei der Verteilung der verschiedenen Funktionsprinzipien zeigt sich, dass die geometriebasierten Methoden den größten Anteil haben. Die ansichtsbasierten Verfahren haben durch die steigende Popularität von CNNs begonnen, zusammen mit den Methoden des Maschinellen Lernens zu wachsen und übertreffen im letzten Zeitraum die geometriebasierten Methoden. Der Anteil der hybriden Ansätze ist leicht zurückgegangen, was darauf zurückzuführen sein könnte, dass die Methoden des Maschinellen Lernens häufig nur ein Eingabeformat verwenden. Die graphbasierten Verfahren spielen in der Verteilung eine untergeordnete Rolle.

Genauso wie bei der Bauteilerkennung ist neben den Daten und Methoden vor allem die Auswertemetrik für den Vergleich von Algorithmen relevant. Diese unterscheiden sich von den vorher präsentierten Metriken und werden daher nachfolgend ausführlich vorgestellt.

Auswertemetriken

Im Laufe der Jahre wurden viele verschiedene Metriken für die Bewertung von Ähnlichkeitsergebnissen definiert, die jeweils unterschiedliche Ziele verfolgen. Die häufig verwendeten und relevanten Metriken für die Geometriesuche nach [251] und [209] werden in diesem Abschnitt vorgestellt.

Die erste Art der Bewertung umfasst insgesamt drei Scores, die im Allgemeinen gleichzeitig angewendet werden: Nearest Neighbor (NN), First Tier (FT) und Second Tier (ST). Der erste Wert, Nearest Neighbor, beschreibt die Präzision des ersten und damit ähnlichsten Retrievalergebnisses. FT enthält die Recall-Ergebnisse der besten *C-1*-Ergebnisse, wobei C die Anzahl der relevanten Modelle für die Retrieval-Aufgabe ist. Das Second-Tier-Ergebnis wird nach demselben Prinzip berechnet, allerdings für die doppelte Anzahl an Ergebnissen. In [269] sind die Gleichungen für Präzision und Recall angegeben und für die Bewertungsmetriken angepasst. Im Anhang ist eine kurze Beispielberechnung für die Metriken in Bild 86 dargestellt.

$$Präzision = \frac{|retrieved@R \cap relevant|}{|retrieved@R|} \quad \text{für NN: } R = 1 \tag{8}$$

$$Recall = \frac{|retrieved@R \cap relevant|}{|relevant|} \quad \text{für FT: } R = C - 1 \quad \text{für ST: } R = 2 * (C - 1) \tag{9}$$

Darüber hinaus wird der Discounted Cumulated Gain (DCG) ebenfalls als Vergleichsmetrik verwendet. Dieser Wert gewichtet relevante Ergebnisse entsprechend ihrer Position in der Abrufliste. DCG bewertet also frühere Ergebnisse höher, indem der normalisierte, summierte und gewichtete Wert für die Position der relevanten Modelle verwendet wird. Nach [209, 251] wird der DCG anhand des Ergebnisvektors G des Retrieval-Ergebnisses R berechnet, dabei gilt $G = 1$, wenn R_i ein relevantes Ergebnis ist und $G = 0$, wenn nicht.

$$DCG_i \begin{cases} G_1 & \text{i=1} \\ DCG_{i-1+\frac{G_i}{\lg_2 i}} & \text{sonst} \end{cases} \tag{10}$$

$$DCG = \frac{DCG_n}{1 + \sum_{j=2}^{C} \frac{1}{lg_2 j}} \tag{11}$$

Andere Bewertungsmetriken sind die R-Präzision, die Precision-Recall-Kurve (PR) oder die Average Precision (AP), wobei weitere Informationen in [209, 251, 270] zu finden sind. Für alle in diesem Abschnitt aufgeführten Metriken gilt: je höher der Score-Wert, desto besser das Retrievalergebnis.

Anwendung in der Produktentwicklung
Für die konkrete Anwendung der Geometriesuche in der Produktentwicklung sind verschiedene Szenarien denkbar. Zuerst ist die Suchebene interessant, wobei zwischen Bauteil und Baugruppen unterschieden wird, wie am Anfang des Kapitels bereits erklärt. Zusätzlich ist noch das Format der Geometrie relevant, wobei primär zwischen einfachen Oberflächennetzen und BRep Bauteilen differenziert wird.

Baugruppensuche
Für den Vergleich von ganzen Baugruppen ist meist der hierarchische Aufbau, umgewandelt in einen Graph, das Grundfeature des Vergleichs. Ein einfaches Beispiel hierfür ist in Bild 36 dargestellt. In der Umsetzung von Chen et al. [271] sind zusätzlich noch Bewegungsinformationen der verschiedenen Bauteile in den Graphen eingeflossen. Dagegen nutzt [272] die Umwandlung der Baugruppen Bauteile in Shape Deskriptor, kombiniert diese und vergleicht die Ähnlichkeit auf Basis dieses neuen Vektors. Der von Lupinetti et al. [273, 274] vorgestellte EAM (Enriched Assembly Model) Descriptor ist eine zusätzliche Erweiterung des hierarchischen Graphen. Dabei werden neue Aspekte auf verschiedenen Ebenen in den Descriptor integriert, wie z. B. die Bauteilklassen, Funktionen, Kontaktflächen oder allgemeine Statistiken zur Baugruppe. Das Vorgehen wurde stetig erweitert, um auch andere Such-Inputs als Baugruppen nutzen zu können [275] oder auch Unterbaugruppen bzw. partielle Ähnlichkeit in Baugruppen zu suchen [276]. Eine ausführliche Einordnung der verschiedenen Baugruppen-Suchverfahren ist in [245] gegeben, inklusive einer genaueren Verortung anhand des Input-Formats und Funktionsumfangs. Neben den vorgestellten allgemeinen Verfahren zur Suche von Baugruppen wurden auch spezifische Lösungen für ausgewählte Anwendungsfälle entwickelt. Ein Beispiel ist das von Li et al. [277] präsentierte Vorgehen zum Retrieval von ähnlichen Fahrzeuginnenraumverkleidungen oder die Suche von Komponenten aus dem Anlagenbau [278].

Bauteilsuche
Grundsätzlich kann bei der Nutzung der Bauteilsuche im Kontext der Produktentwicklung zwischen den Frameworks zur Anwendung und der Metho-

de unterschieden werden. Die erste Gruppe umfasst dabei Strukturen, die die Umsetzung der Geometriesuche in die Praxis beschreiben. Oftmals werden die präsentierten Vorgehensmodelle auch als Suchmaschinen für 3D-Bauteile bezeichnet. Bei den Verfahren zur Suche von Bauteilen werden häufig spezifische Geometrieformate aus der Produktentwicklung eingesetzt. Nachfolgend werden beide Aspekte der Bauteilsuche genauer vorgestellt.

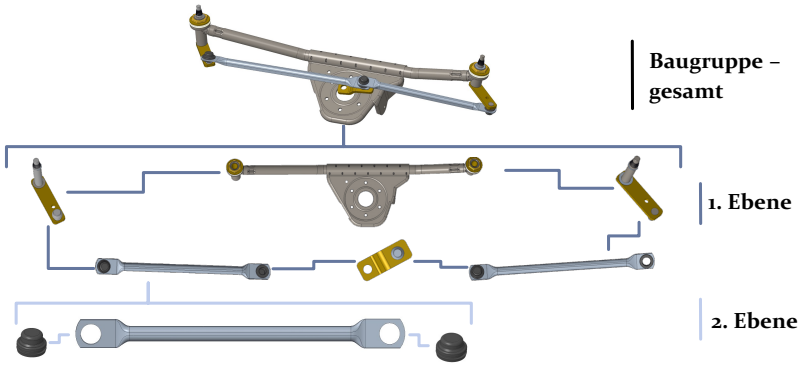


Bild 36: Ein Ausschnitt eines Baugruppenaufbaus für ein Scheibenwischergestänge

Frameworks zur Anwendung
Das allgemeine Vorgehen zum Einsatz von Bauteil-Retrieval wurde bereits vorher im Kapitel erklärt. Allerdings haben sich für die Problemstellungen in der Produktentwicklung eigene Lösungen etabliert. Die erste Gruppe beschreibt dabei Geometriesuchmaschinen und erklärt deren Aufbau [279–286]. Diese berücksichtigen zahlreiche neue Aspekte wie Datenbanken, User-Feedback oder Suchanfrage-Formate. Diese Idee wurde dann auf das gesamte Product Lifecycle Management erweitert [287–289]. Auch andere Aspekte können adressiert werden, z. B. die Nachhaltigkeit des Produkt [290] oder Fertigungsinformationen [291].

Methoden
Bei der Bauteilsuche mit höherwertigen Geometrieformaten aus der Produktentwicklung kann wiederum nach dem Input der Suche unterschieden werden. Viele Verfahren nutzen die Eigenschaften der CSG- oder BRep-Geometrie, indem sie diese in einen Graphen überführen [292–299]. Die Verwendung von Histogrammen [300], Vektoren [301] oder CSG-Bäumen [302] ist ebenfalls möglich. Die neueren Methoden nutzen dagegen entweder eine Kombination verschiedener Eingabemöglichkeiten (*engl. Multi-Mode*) [303–307] oder Ontologie [308] bzw. integrierte Produktmodelle [309].

Neben den Geometrieinformationen können auch Fertigungsfeatures bzw. Prozessinformationen für die Suche verwendet werden. Die meisten Vorgehen

basieren auf der NC-Steuerung für spanende Fertigungsverfahren, wobei auch hier Graphen [310–313], Multi-Mode [314–317] und Vektoren [318, 319] als Vergleichsfeatures angewendet werden. Einzelne Methoden sind auch für umformende Fertigungstechnologien entwickelt worden [320, 321].

Auch eine direkte Verwendung der CAD-Konstruktionselemente für die Suche ist möglich, wobei wiederum eine Umwandlung in Graphen [322–324], Vektoren [325] oder Ontologien [326] notwendig ist. Zudem können auch FE-Ergebnisse für verschiedene Geometrien für eine Suche adaptiert werden [327]. Bei Volumenmodellen können mit dem Solid verknüpfte Merkmale für die Bauteilsuche verwendet werden, z. B. Fertigungsinformationen, Flächenarten oder Materialinformationen. Bekannte Verfahren sind in [328], [329] und [330] beschrieben.

Neben Methoden für Solid-Geometrie sind auch neue Verfahren für Oberflächennetze und den Einsatz in der Produktentwicklung beschrieben worden. Der Unterschied zu den Methoden aus der Literaturrecherche ist, dass hier nur Ansätze betrachtet werden, welche mit einem Datensatz aus dem Maschinenbau getestet wurden. Als Grundlage dient der ESB-Datensatz, da dieser sehr lange verfügbar ist und somit viele Methoden an diesem getestet wurden. Weiterhin ist für Verfahren im Einsatzgebiet der Produktentwicklung wichtig, dass die Suche nach ähnlichen Bauteilen oder Komponenten ohne das Labelling der vorhandenen Daten durchführbar sein muss. Dies begründet sich vor allem durch den enormen Aufwand für das händische Einordnen der Geometrien in bestehende Kategorien. Die Methoden, die alle Anforderungen erfüllen, sind nach Funktionskategorie, Trainingsart und Bewertungsmetriken in der Tabelle 13 im Anhang eingeordnet worden. Eine ausführliche Erläuterung der Tabelle ist ebenfalls im Anhang zu finden, zusammen mit den insgesamt 24 ermittelten Verfahren. Die Auflistung legt deutlich dar, dass eine Vielzahl an Verfahren für die Suche von Geometrien auf Basis von Oberflächennetzen ermittelt wurde und dabei viele unterschiedliche methodische Ansätze verfolgt wurden. Nach der Betrachtung der Bauteilsuche soll abschließend noch ein spezieller Teilbereich der Multi-Modal-Suche analysiert werden, die Verwendung von 2D-Skizzen.

2.2.4 Verknüpfung von Skizzen und Geometrie

Die allgemeine Grundidee 2D-Skizzen mit den datengetriebenen Methoden zu kombinieren ist schon länger Bestandteil der Forschungslandschaft. Nach [331, 332] kann die Skizze als Grundlage für eine Vielzahl an Aufgaben dienen, beispielsweise zur Erkennung, Generierung oder zur Ähnlichkeitssuche. Neben der Anwendung in derselben Modalität sind auch übergreifende Ver-

fahren möglich, wobei beispielsweise von der Skizze eine Geometrie oder ein Video als Suchergebnis identifiziert wird.

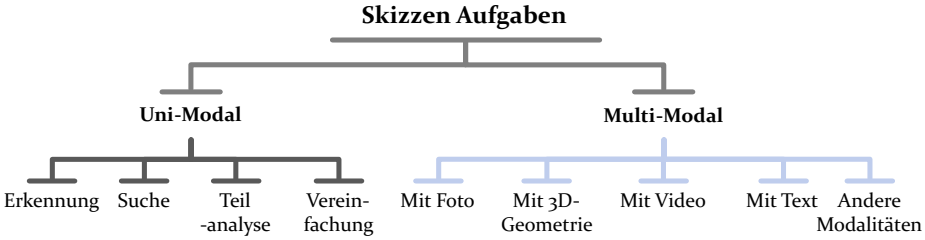


Bild 37: Übersicht der Anwendungsfälle für 2D-Skizzen nach [332]

Im Kontext der geometrischen Ähnlichkeitssuche sind Methoden, die Skizzen und Geometrie verknüpfen eher relevant, weshalb diese im Folgenden näher betrachtet werden. Ein essentieller Baustein bei der Entwicklung neuer Verfahren ist der verfügbare Datenbestand. Er ist bei dieser Suchart besonders herausfordernd, da für beide Format-Domänen hochwertige Daten gesammelt werden müssen. Erste Datensätze wurden ab dem Ende der 2000 Jahre veröffentlicht, beispielsweise von [333, 334]. Großen Einfluss auf das Forschungsfeld hatte die Bereitstellung des SHREC'12- und '13-Datensatzes von [335]. Diese basieren auf zwei großen Datensätzen der jeweiligen Domänen: dem PSB [209] für Geometrie und der Skizzen-Sammlung von [336].

Die konkreten Lösungsansätze zur Suche von Geometrie sind vielfältig und nutzen häufig gerenderte Ansichten der Bauteile als Vergleichskriterien. Der klassische Ansatz beinhaltet die Definition unterschiedlicher Ansichten des Bauteils und anschließend eine Überführung in skizzenähnliche Bilder mittels Projektion- oder Renderalgorithmen. Im Anschluss werden die abgeleiteten Bilder mit den Skizzen anhand von Matchingalgorithmen verglichen um die ähnlichsten Bauteile zu ermitteln [337–342]. Andere Methoden nutzen beispielsweise Semantiken [343] oder Graphen [344, 345] für den Vergleich.

Mit dem Aufkommen der verschiedenen Machine- und Deep-Learning-Techniken sind diese auch auf die Problemstellung der Skizzen zur Geometriesuche angewendet worden. Nach [332] werden wiederum diverse Techniken zur Funktionsumsetzung genutzt, unter anderem das end-to-end Deep-Learning-Paradigma [346–350], Deep-Metric-Learning [351] oder Triplet-Ranking [352, 353].

Anwendung in der Produktentwicklung
Zwei Bereiche stehen besonders im Fokus für den Einsatz in der Produktentwicklung: die Suche von Bauteilen und die Erzeugung von 3D-Geometrie,

jeweils ausgehend von einer Handskizze. Da die Verwendung von Ähnlichkeiten der Hauptbestandteil der Arbeit ist, werden nachfolgend primär die Suchalgorithmen genauer betrachtet.

Geometriesuche
Bedingt durch die unterschiedlichen Regeln aus dem Maschinenbau für technische Skizzen sind auch die Vorgehensweisen zum Finden ähnlicher Bauteile daran angepasst. So verwenden erste Veröffentlichungen die orthogonale Projektion als initialen Schritt für den nachfolgenden Vergleich [354–358]. Diese Aufbereitungsart ist vergleichbar mit der Klapp-Regel beim Erstellen von technischen Zeichnungen. Andere Verfahren orientieren sich an dem klassischen Vorgehen, präsentiert im vorherigen Kapitel, indem sie weitere Ansichts- bzw. Bild-Projektionsarten für die Erstellung der Vergleichsbilder verwenden. Dabei kommen isometrische und orthogonale Projektionen [359] oder Multi-View-Verfahren [360, 361] zum Einsatz.

Neben dem klassischem Vorgehen sind noch andere Methoden entwickelt worden, welche sich weitere Eigenschaften der technischen Skizze zu Nutzen machen. Beispielsweise basiert die Suche von Li et al. [362] auf 3D-Skizzen, die mittels Handcontroller in einer virtuellen Umgebung erzeugt werden. Wang et al. [363] verwendet einen Skelettierungs-Algorithmus zum Ableiten von Vergleichsmöglichkeiten zwischen den Skizzen und der 3D-Geometrie. Auch die Umwandlung der Skizzen in Graphen ist eine Art den Ähnlichkeitsvergleich durchzuführen, indem eine vorherige Segmentierung für die Grapherzeugung verwendet wird [364, 365].

In den bisherigen Beispielen aus dem Bereich der Produktentwicklung sind hauptsächlich die geometrischen Merkmale der Bauteile bzw. Baugruppen ausgenutzt worden. Es wurden aber keine anderen Informationen verwendet und Baugruppen und Bauteile immer gleich gesetzt, bzw. die globale Ähnlichkeit als Vergleichskriterium herangezogen. Dagegen entwickelte Wen et al. [366] für das Umfeld der Verfahrenstechnik ein neues Vorgehen, welches ausgehend von einem Verrohrungsdiagramm 3D-Prozess-Anlagen sucht.

Neben der Suche beschäftigt sich ein anderer Zweig mit der automatischen Generation von 3D-Bauteilen, ausgehend von technischen Handskizzen. Dabei werden primär Verfahren der Computer Vision verwendet, um die durchgängigen Linien der Handskizzen zu ermitteln. Im Anschluss erfolgt eine Zuordnung der Schnittpunkte und eine Übertragung ins BRep-Format [198, 367–369]. Andere Ansätze [370, 371] nutzen Skizzen für die automatische Ableitung von Simulationsmodellen.

3 Handlungsbedarf und Vorgehensweise

Nach der ausführlichen Beschreibung des aktuellen Stands der Forschung in den beiden Hauptbereichen Produktentwicklung und datengetriebene Methoden soll in diesem Kapitel zunächst der Handlungsbedarf ermittelt und im Anschluss daran die Forschungsfragen abgeleitet werden. Abschließend wird die Vorgehensweise zur Beantwortung der Fragen präsentiert und kurz erläutert.

3.1 Handlungsbedarf

Der Handlungsbedarf für diese Arbeit motiviert sich aus dem **Potential und Nutzen**, welche **vorhandene Daten** aus dem Produktentwicklungsprozess für die Entwicklung neuer Produkte bieten. Die Vorteile sind in mehreren Studien [17, 372, 373] und Analysen [374, 375] ausgiebig dargelegt worden. Demzufolge führt der Einsatz in der Produktentwicklung zu weniger Risiken und Fehlern, kürzeren Produktversuchen, erhöhter Qualität und somit geringeren Kosten und Zeitbedarf für die Produktentwicklung [18, 288, 326, 374, 376, 377]. Die Einsparung von Zeit und Ressourcen im Entwicklungsprozess durch die Wiederverwendung bestehender Daten führt demnach auch zu einer nachhaltigeren Produktentwicklung [373, 378]. Zudem ist eine erfolgreiche Nutzbarmachung der vorhandenen Daten die Grundlage für die Evolution von Produkten [374] und kann bei der Problemlösung durch eine schnelle Generation von Varianten helfen. In mehreren Beispiele aus unterschiedlichen Anwendungsdomänen, wie Open-Source-Hardware [379], Luftfahrt mit Fokus auf die Evolution [375, 380, 381], Werkzeugbau oder Automobilindustrie [372], konnte der Mehrwert der Wiederverwendung bestätigt werden.

Dabei ist die Ähnlichkeitssuche immer die Grundlage für eine erfolgreiche Umsetzung eines Wiederverwendungssystems [17, 18, 378], wobei viele Arbeiten die Suche nach Wissen adressieren, aber nicht die vorliegenden Dateiformate [379]. Diese erfolgreiche Suche ist demzufolge die Basis, um verknüpfte Informationen oder Wissen zu erhalten [17, 18]. Zudem ist der Verwendungszeitpunkt im Entwicklungsprozess relevant, da eine Umsetzung in den frühen Phasen bedingt durch den hohen Einfluss auf die Produktqualität und -kosten [12, 13] besonders vorteilhaft ist [382]. Allerdings ist der flächendeckende Einsatz in der Produktentwicklung immer noch nicht vollzogen, da die **Heterogenität** der verschiedenen **Datenformate** (siehe Kapitel 2.1.2) im Entwicklungsprozess eine **große Herausforderung** darstellen. Einzelne Insellösungen für spezifische Dateiformate sind vorhanden, wie beispiels-

weise die Nutzung von CAD-Modellierungs-Strategien [383] zur besseren Wiederverwendung der Geometrien. Aber die hohe Vielfalt und Varianz der möglichen Daten führt zu einem großen Wartungs- und Verwaltungsaufwand, da bestehende Daten aufbereitet, kategorisiert, abgeändert oder umbenannt werden müssen, damit bestehende Ansätze diese verarbeiten können. Hinzu kommen weitere Probleme [288] bei der Ähnlichkeitssuche, wie die fehlende Verknüpfung von Modellen zu weiteren Inhalten und unbekannten oder ungenauen Bezeichnungen. Auch uneindeutige Keywords und sich über den zeitlichen Verlauf ändernde Bezeichnungen oder Kontext für gewisse Produkte oder Bauteile sind Herausforderungen für die Suche.

Die Darlegung des Produktentwicklungsprozesses im Stand der Technik (siehe Kapitel 2.1.1) zeigt deutlich den iterativen Ablauf und die unterschiedlichen Entwicklungsphasen. Jeder Abschnitt hat seine Charakteristika und dementsprechend eigene Daten, die währenddessen erzeugt oder geändert werden. Nach der Definition der Anforderungen sind Skizzen im Produktentwicklungsprozess eine der ersten Repräsentationen des zu entwickelnden Produkts. Sie treten folglich besonders in den frühen Phasen der Produktentwicklung auf und bieten die Möglichkeit durch den Einsatz einheitlicher Symbole schnell Ergebnisse mit hoher Aussagekraft zu erzeugen. Allerdings sind die generierten Skizzen schwierig automatisiert auszulesen, da viele Variablen und Randbedingungen den Inhalt beeinflussen können. Der aktuelle Stand der Forschung zur Symbolerkennung in verschiedenen Ingenieursdomänen ist ausführlich in Kapitel 2.2.2.4 beschrieben und analysiert worden. Dabei wurde festgestellt, dass viele **Algorithmen** und Verfahren aus dem Machine bzw. Deep-Learning zur Anwendung auf Skizzen **vorhanden** sind, allerdings **fehlen** die zum Training notwendigen **Datensätze**. Weiterhin liegt der **Hauptfokus** auf der **Erkennung** von Symbolen, wobei die **Kontextualisierung** der erkannten Skizzen vor allem im Maschinenbau aktuell kaum verfolgt wird.

Neben den Skizzen ist vor allem die Produkt-Geometrie ein essentieller Bestandteil im Entwicklungsprozess. Die Geometrie wird auch als eines der Hauptergebnisse der virtuellen Produktentwicklung bezeichnet [81, 384]. Allerdings werden viele unterschiedliche Geometrieformate im Entwicklungsprozess eingesetzt, abhängig von der verwendeten Software, Aufgabe und Zielsetzung (siehe Kapitel 2.1.2.3). Daher ist eine ausführliche Analyse des aktuellen Standes der Forschung im Bereich der Geometrie-Ähnlichkeitssuche in Kapitel 2.2.3.3 durchgeführt worden. Im Gegensatz zu den Skizzen zeigt sich, dass **Datensätze vorhanden** sind, allerdings **wenig** Ansätze und **Vorgehensmodelle**, welche die spezifischen Anforderungen der Produktentwicklung berücksichtigen. Demnach ist eine möglichst breite Anwendbarkeit der Ähnlichkeitssuche anzustreben, weshalb sich **Oberflächennetze** besonders

gut als Grundlage eignen, da diese aus beinahe allen Geometrieformaten im Entwicklungsprozess erstellt werden können.

3.2 Hypothese und Forschungsfragen

Das vorherige Kapitel hat den aktuellen Handlungsbedarf für die Entwicklung von Vorgehensmodellen für die Ähnlichkeitssuche in der Produktentwicklung ausführlich dargelegt. Dementsprechend liegt folgende Hypothese der Forschung zu Grunde:

Forschungshypothese
Datengetriebene Methoden in der Konzept- und Entwurfsphase der Produktentwicklung erlauben ohne händische Aufbereitungsoperationen das virtuelle Finden vorhandener Prinzipskizzen und deren Verknüpfung zu Baugruppen, sowie die Verbesserung der Ähnlichkeitssuche von Bauteilen.

Hieraus lässt sich die Forschungslücke ableiten, welche grafisch in Bild 38 dargestellt ist. Dabei zeigt sich das Problem, dass die Ähnlichkeitssuche für unterschiedliche Formate in der Produktentwicklung eine hohe Relevanz besitzt, bedingt durch die Vorteile der Anwendung. Allerdings ist die hohe Vielfalt an unterschiedlichen Formaten, bedingt durch den Aufwand für die Datenaufbereitung, ein aktuelles Hindernis für den erfolgreichen Einsatz.

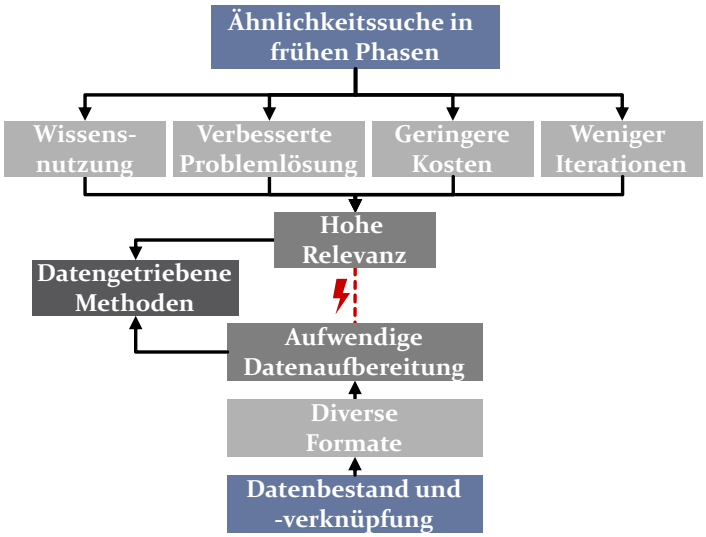


Bild 38: Darstellung der Forschungslücke mit den Bereichen Ähnlichkeitssuche und der fehlenden Unterstützung bei der Nutzung vorhandener Daten

Diese präsentierte Forschungslücke soll durch die nachfolgenden Fragen adressiert werden:

F1: Wie können die Zeichnungselemente auf technischen Skizzen mit Algorithmen aus dem Bereich der Computer Vision erkannt und nachfolgend für die Ähnlichkeitssuche oder die weitere Anwendung nutzbar gemacht werden?

F2: Wie kann eine automatisierte Verknüpfung von Skizzen und Baugruppen umgesetzt werden, um den Nutzen der automatisierten Zeichnungserkennung und Kontextualisierung weiter zu steigern?

F3: Wie kann das Vorgehen der geometrische Bauteilsuche an die spezifischen Herausforderungen der Produktentwicklung angepasst und umgesetzt werden?

3.3 Vorgehensweise

Zur Beantwortung der vorgestellten Fragestellungen ist ein modulares Vorgehensmodell entwickelt worden. Im Rahmen der Arbeit wird jede der Fragen deshalb in einem eigenständigen Kapitel adressiert und diskutiert, was in Bild 39 illustriert ist.

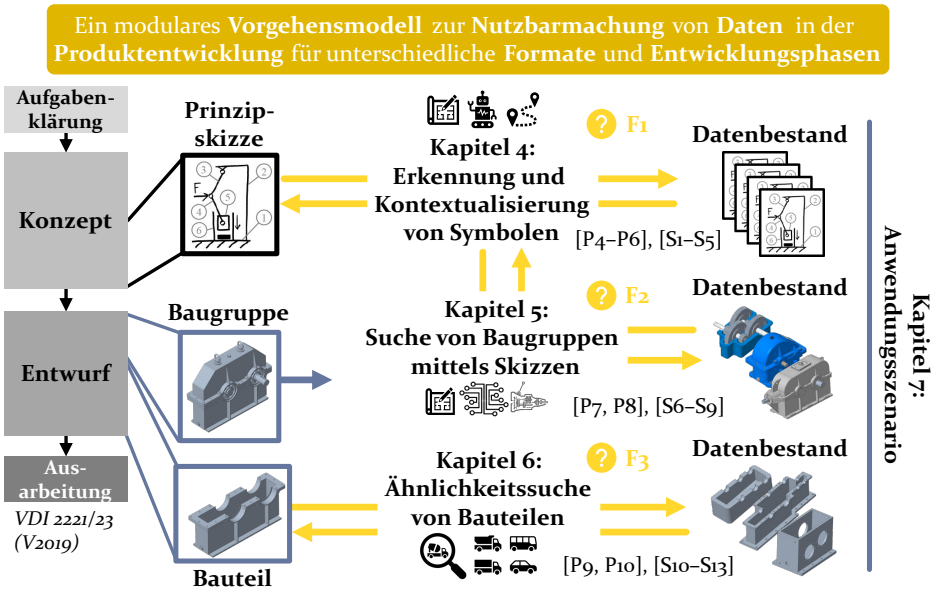


Bild 39: Überblick und Einordnung der unterschiedlichen Methoden in den PEP inkl. Kapitelzuweisung und Forschungsfragen

Neben der Einordnung der Forschungsfragen ist in diesem Bild auch eine Verortung im Produktentwicklungsprozess vorgenommen worden, wodurch die breite Anwendbarkeit und der Nutzen ersichtlich wird. Abschließend wird das gesamte Vorgehensmodell anhand eines Anwendungsszenarios in Kapitel 7 durchlaufen. Allgemein können die entwickelten Module des Vorgehensmodells aus Kapitel 4 bis 6 unabhängig voneinander eingesetzt werden, abhängig von der gestellten Aufgabe. Wenn beispielsweise schon eine 3D-Geometrie vorhanden ist, kann direkt das Modul zur Bauteil-Ähnlichkeitssuche aus Kapitel 6 angewendet werden, ohne vorher eine Skizze erstellen zu müssen.

Die erste Forschungsfrage wird ausgiebig in Kapitel 4 bearbeitet, wobei der Fokus auf der Nutzbarmachung von Prinzipskizzen liegt, weshalb diese zunächst mittels datengetriebenen Methoden erkannt werden müssen. Danach können diese für eine Suche oder weitere Umwandlungen verwendet werden, was ausgiebig im zweiten Teil des Kapitels erläutert wird. Anknüpfend daran wird im nachfolgenden Kapitel 5 ein neues Vorgehen vorgestellt, welches über eine bislang unbekannte Skizze als Input vorhandene ähnliche Baugruppen aus einem bestehenden Datenbestand suchen kann. Auf der Geometrieebene wird zusätzlich ein Lösungsansatz für Bauteile in Kapitel 6 vorgestellt werden. Abschließend wird am Ende der Arbeit das gesamte Vorgehen anhand eines Anwendungsbeispiels durchlaufen.

Im Folgenden wird chronologisch im Produktentwicklungsprozess vorgegangen, weshalb die Erkennung und Nutzbarmachung von Prinzipskizzen zuerst präsentiert wird.

4 Erkennung und Kontextualisierung technischer Skizzen

Der erste Forschungsfrage über die Erkennung und Kontextualisierung von Prinzipskizzen in der Produktentwicklung wird in diesem Kapitel bearbeitet und die zugehörigen Ergebnisse vorgestellt. Dafür wird zunächst ein Modul präsentiert, welches die Herausforderungen einer automatisierten Skizzenerkennung und -auswertung lösen soll. Danach werden unterschiedliche Arten der Datenerzeugung erläutert und zusammen mit verschiedenartigen Deep-Learning-Modellen untersucht. Neben der Symbolerkennung werden im Anschluss die Kontextualisierungsvarianten genauer vorgestellt und mittels eines Demonstratordatensatzes angewendet. Das Vorgehen zur Erkennung von Symbolen wurde teilweise in [S1–S3, S5] bearbeitet und in [P4, P6] veröffentlicht. Der Ansatz für die Kontextualisierung ist in [P5] erschienen.

4.1 Modulübersicht

Aufbauend auf der ausführlichen Literaturrecherche in Kapitel 2.2.2.4 und des daraus abgeleiteten Handlungsbedarfs (vgl. Kapitel 3.1) wird deutlich, dass die Datenerzeugung das größte Problem für die Symbolerkennung darstellt. Zur Lösung dieser Problemstellung ist ein Vorgehen für dieses Modul entwickelt worden, welches auf synthetischer Datenerzeugung und trainierten Deep-Learning-Modellen basiert, und gesamtheitlich in Bild 40 dargestellt ist. Dementsprechend werden die Trainingsdaten durch eine synthetische Datenerzeugung generiert, was der Subprozess im unteren Abschnitt des Schaubilds zeigt.

Nach der erfolgreichen Erkennung der Symbole in der Input-Skizze können die gefundenen Bounding Boxen verwendet werden, um die Skizze in einen Graphen zu überführen. Dieser kann im Anschluss für diverse Anwendungen genutzt werden, beispielsweise zur Ähnlichkeitssuche, Designsynthese oder für ein verbessertes Produktdesign. Die verschiedenen Teilschritte werden nachfolgend genauer vorgestellt und die möglichen Parameter der Verfahren erläutert.

4.2 Symbolerkennung

Die Erkennung der Symbole in Prinzipskizzen gliedert sich grob in die Bereiche Datenerzeugung und Objekterkennungsmodelle. Im ersten Abschnitt müssen die Trainingsbilder, Bounding Boxen und Symbol-Klassen definiert werden, damit die Modelle ausreichend Trainingsdaten zur Verfügung haben.

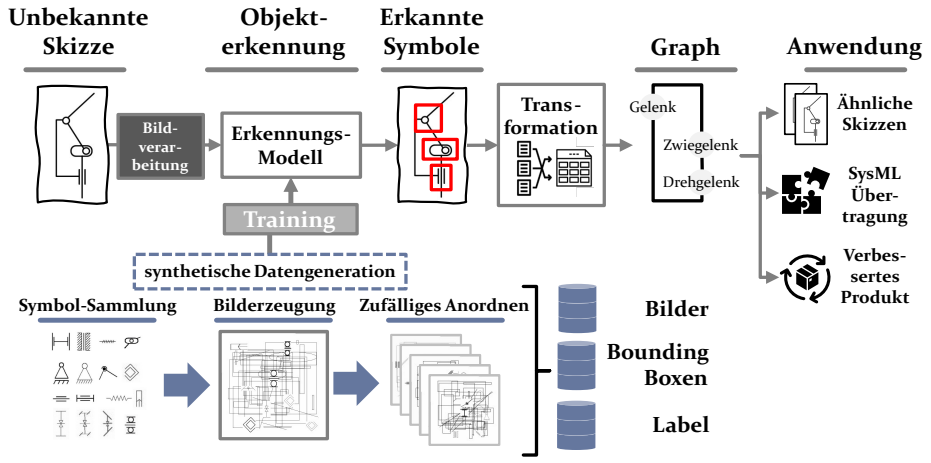


Bild 40: Gesamtübersicht des Sub-Vorgehensmodells nach [P4–P6]

Dabei sind allgemein drei Arten der Datensatzgenerierung möglich, wobei die erste das händische Labeln vorhandener Bilder umfasst. Dieser Vorgang ist sehr zeitintensiv und setzt einen ausreichenden Bestand an Skizzen voraus. Die zweite Art ist die Nutzung von vorhandenen, öffentlich zugänglichen Datenbeständen. Die synthetische Datenerzeugung beschreibt die letzte Möglichkeit, wobei in diesem Fall künstlich über Regeln oder Algorithmen Bilder und Bounding Boxen erzeugt werden. Bedingt durch die Einschränkungen der Maschinenbaudomäne soll die synthetische Datenerzeugung in diesem Vorgehensmodell umgesetzt werden. Die so generierten Daten können danach einem Objekterkennungsmodell übergeben werden, welches die neu erzeugten Skizzen lokalisiert und einordnet. In den nachfolgenden Sektionen sollen somit die Datenerzeugungsansätze in Kombination mit dem Objekterkennungsmodellen genauer vorgestellt werden.

4.2.1 Trainingsdatenerzeugung

Das grundlegende Ziel der synthetischen Datengenerierung ist, die Symbole der Prinzipskizzen so genau wie möglich zu reproduzieren, damit diese auf realen Skizzen mit hoher Genauigkeit erkannt werden. Da es jedoch nicht genügend vorhandene Daten für diesen Skizzentyp für eine künstliche Datenvervielfältigung (*engl. Data Augmentation*) gibt, müssen die Symbole zufällig generiert werden. Der schematische Ablauf der Trainingsdatenerzeugung ist im unteren Teil von Bild 40 dargestellt. Dazu müssen die Symbole der Skizze vorher definiert und parametrisiert werden. Anschließend können diese zufällig ausgewählt und wiederum mit zufälligen Parameterwerten in

eine leere Skizze eingefügt werden. Dieser Prozess wird mehrfach wiederholt, bis eine beliebige Anzahl an Symbolen auf dem Bild platziert wurde. Das generelle Vorgehen bietet den Vorteil, dass neben den Bilder simultan die Bounding Boxen und Label erzeugt werden, wodurch kein weiteres Aufbereiten der erstellten Daten notwendig ist. Zudem sind die Anzahl und Variation der Symbole frei wählbar, ebenso wie die Gesamtzahl der Trainingsbilder. Diese hängt in erster Linie von der Rechenleistung und der zur Verfügung stehenden Zeit ab.

Im Rahmen dieser Arbeit wurden zwei verschiedene Verfahren zur Generation der Trainingsdaten mit unterschiedlichen Python-Paketen (Pillow [385] und SketchGraph [386]) entwickelt, die im Folgenden vorgestellt, näher untersucht und untereinander verglichen werden. Für beide Generierungsmethoden gelten die gleichen Randbedingungen, angelehnt an dem bereits beschriebenen grundlegenden Aufbau.

4.2.1.1 Pillow

Der erste Ansatz basiert auf der weit verbreiteten Python-Bibliothek „Pillow“, die in erster Linie für die Bildverarbeitung und -manipulation eingesetzt wird. Die einfache Bedienung und die Möglichkeit, schnell Bilder mit neuen Inhalten zu erzeugen, sind die Hauptargumente für die Auswahl des Pakets. Außerdem ist das Verfahren parallelisierbar, was eine schnelle Datengenerierung ermöglicht. Das allgemeine Vorgehen ist mittels Pseudo-Code im Anhang unter Algorithmus 1 beschrieben und beginnt mit der Erstellung eines weißen Bildes in einer definierten Größe und wurde bereits teilweise in [P4] veröffentlicht.

Im Anschluss wird eine zufällige Anzahl an Symbolen auf das Bild gezeichnet, wozu zunächst eine beliebige Position des neuen Symbols bestimmt und geprüft wird, ob diese mit einer bestehenden Bounding Box kollidiert. Wenn nicht, wird die Position an die nächste Funktion *drawSymbol* übergeben. Diese zeichnet das ausgewählte Symbol nach vorher definierten Regeln mit Hilfe des Pillow-Pakets, wobei die Werte für Rotation, Liniendicke, allgemeine Größe, Verhältnis von Höhe und Breite zufällig für jedes Symbol definiert werden. Darüber hinaus wird ein Variationsfaktor gewählt, der die Abweichungen für jede Symbolform erzeugt. Anschließend werden je nach Symboltyp die notwendigen Punkte generiert und dann durch die Klasse *ImageDraw* gezeichnet. Die Hintergrundgenerierung ist ein Parameter, um die verschiedenen Arten der Datengenerierung zu untersuchen. Somit werden zwei Datensätze für das Training betrachtet, einer mit und einer ohne zufälligen Hintergrund. Der theoretische Hintergedanke besteht darin, mögliche Verbindungslinien zu imitieren und somit ein robusteres Erkennungsmodell zu trainieren,

indem ein zufällig generierter Hintergrund verwendet wird. Die Ergebnisse der erzeugten synthetischen Skizzen sind in der oberen Reihe in Bild 42 mit einem blauen Rahmen markiert.

4.2.1.2 SketchGraph

Im Gegensatz zum Pillow-Ansatz verwendet die SketchGraph-Version eine andere Art der Symbolerstellung, ähnlich wie die Skizzierumgebung in einem CAD-Programm. Zunächst wird eine Befehlsfolge erstellt, die alle relevanten Informationen für die Erstellung der Skizze enthält. SketchGraph unterscheidet dabei zwischen Knoten- und Kantenoperationen, sowie den damit verbundenen Eigenschaften. Ein Beispiel für die Umwandlung des Graphen in eine Skizze ist in Bild 41 illustriert. Die Sequenz wird dann in ein Skizzenobjekt überführt, das mit der API von PTC Onshape interagiert. Diese Verknüpfung ist notwendig, da Onshape als geometrischer Constraint Solver arbeitet und somit die endgültige Skizze erzeugt. Die resultierende gelöste Skizze kann dann mit verschiedenen Einstellungen gerendert werden. Es besteht die Möglichkeit, einen „handgezeichneten" Modus auszuwählen, der die Linien der Skizze verändert, um eine handgezeichnete Skizze zu imitieren. Der allgemeine Programmablauf ist wieder im Anhang in Algorithmus 2 detailliert beschrieben und wurde in [P6] veröffentlicht.

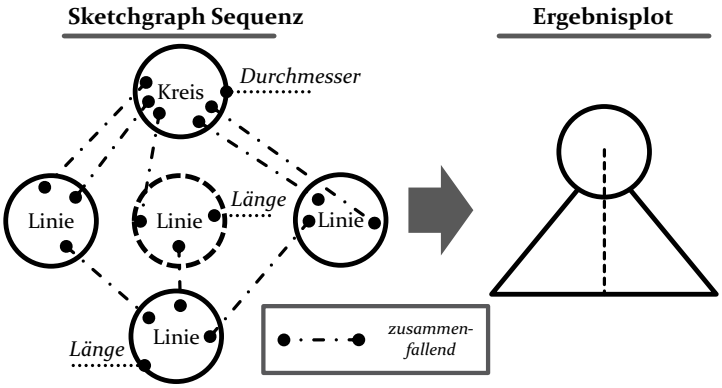


Bild 41: Beispiel für Graph zur Erzeugung eines Lagersymbols mittels SketchGraphs nach [S5]

Ein weiterer Unterschied zur Pillow-Variante ist die Möglichkeit, die Darstellung der neuen Skizzen realistischer umzusetzen. Ausgehend von den realen Prinzipskizzen werden Symbolpaare nach vorher definierten Regeln gebildet, z. B. werden Zahnradsymbole direkt nebeneinander platziert. Darüber hinaus werden auch Verbindungslinien zwischen ähnlichen Domänensymbolen erzeugt, z. B. zwischen einem Drehgelenk und einem Festlager. Die Auswahl

der Kontaktpartner und zugehörigen Symbole erfolgt nach dem Zufallsprinzip, wobei die Position der vorhandenen Symbole für die Verbindungslinien berücksichtigt wird, um Überschneidungen zu vermeiden. Die übrigen Symbole werden zufällig gewählt, positioniert und variiert, wie beim vorherigen Ansatz. Allerdings erfolgt in diesem Vorgehen die Kollisionsprüfung in der jeweiligen Objektklasse. Für den Aufbau der einzelnen Symbolklassen gelten ebenfalls allgemein festgelegte Regeln, außer für das Planetengetriebe, da dieses Symbol deutlich mehr Platz benötigt. Bei der Erzeugung der Skizze wird daher die Anzahl der Planetenräder überprüft und auf ein Symbol pro Skizze reduziert, um genügend Raum für die anderen Symbole zu lassen. Neben den vielen kleinen Faktoren können zwei große Parameter bei der Datensatzerstellung eingestellt werden: Handgezeichnete Symbole und realistischere Skizzendarstellung. Die grau umrahmten Beispielbilder in Bild 42 veranschaulichen die Ergebnisse für die verschiedenen Kombinationen.

Der Nachteil dieser Methode ist, dass der Zeitaufwand für die Generierung der Daten erheblich steigt. Obwohl Multiprocessing und mehrere Onshape-Dokumente zur Erstellung der Skizzen verwendet werden, ist der Zeitaufwand immer noch höher. Im Durchschnitt dauert die Erstellung einer Skizze mit der SketchGraph-Methode etwa 0,7 bis 1 Sekunde, während der Pillow-Ansatz nur 0,001 Sekunden benötigt.

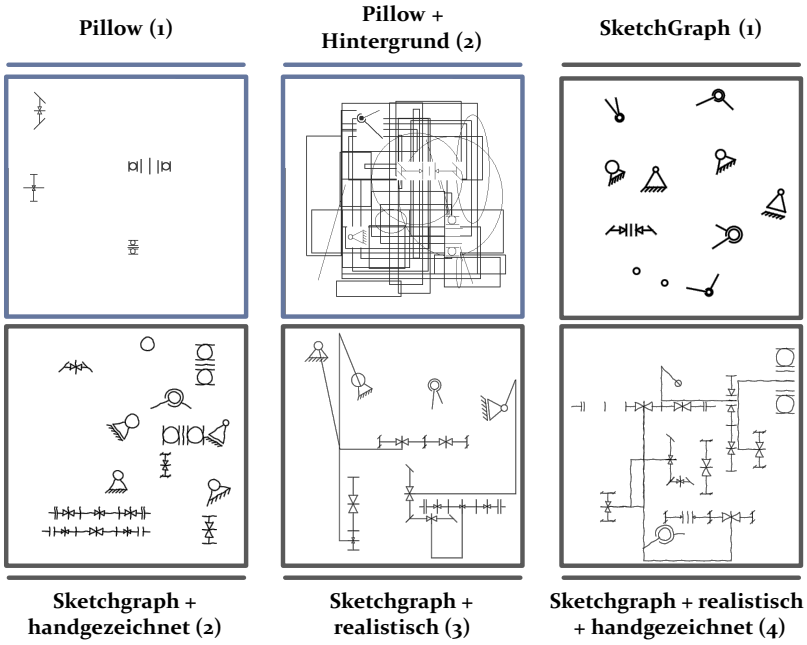


Bild 42: Beispielbilder für die Datenerzeugungsmethoden Pillow und SketchGraph

4.2.2 Objekterkennungsmodelle

Nach der Datenerzeugung können die generierten Datensätze, bestehend aus Bilder, Label und Bounding Boxen, einem Objekterkennungsmodell übergeben werden. Der vorhandene Stand der Forschung in diesem Bereich ist in Kapitel 2.2.2 ausführlich beschrieben. Die Algorithmen werden grundsätzlich zwischen ein- und zweistufigen Verfahren untergliedert, welche verschiedene Eigenschaften mit sich bringen. Im Rahmen dieser Arbeit sollen somit für einen besseren Vergleich beide Arten an Objekterkennungsmodellen untersucht werden, weshalb das YOLOv5 und MASK- bzw. Faster-RCNN (siehe für beide Modelle Kapitel 2.2.2.2) ausgewählt wurden.

Fazit Kapitel 4.2

Insgesamt ist für die Aufgabe, Symbole in Prinzipskizzen zu erkennen, ein Vorgehen mit mehreren Subprozessen entwickelt worden. Abgeleitet aus dem Stand der Technik liegt der Forschungsbedarf auf der Datenerzeugung, weshalb zwei synthetische Arten zur Datensatzgenerierung mit verschiedenen Grundlagen entwickelt und umgesetzt wurden. Die Pillow- und SketchGraph-Varianten besitzen dabei unterschiedliche Vor- und Nachteile, welche in Kombination mit dem Objekterkennungsmodellen nachfolgend analysiert werden.

4.3 Anwendung der Symbolerkennung

Zur Beantwortung der ersten Forschungsfrage wird in nachfolgenden Analysen untersucht, ob die Daten oder das Objekterkennungsmodell den größeren Einfluss auf die Erkennungsgenauigkeit unbekannter Skizzen haben. Zu diesem Zweck wurden insgesamt sechs Trainingsdatensätze erzeugt, zwei mit Pillow- und vier mit dem SketchGraph-Ansatz. Die trainierten Modelle werden mit zwei unbekannten Test-Datensätzen ausgewertet, um die Fähigkeit der Symbolerkennung zu überprüfen. Zunächst werden daher die Objekterkennungsmodelle und die unbekannten Datensätze genauer vorgestellt und anschließend die Ergebnisse diskutiert.

4.3.1 Implementierung der Objekterkennungsmodelle

Damit die verschiedenen Arten der Datenerzeugung objektiv miteinander verglichen werden können, sollen verschiedene Objekterkennungsmodelle mit den jeweiligen Datensätzen trainiert werden. Zur Abdeckung vieler Verfahren sind deshalb ein- und zweistufige Modelle ausgewählt worden, was für die Durchführung der Studie in folgenden Implementierungen der

Objekterkennungsmodelle mündete: MASK-RCNN [387], YOLOv5 [388], Faster-RCNN [389]. Alle Modelle wurden mit Transfer Learning trainiert, wobei die Gewichte des COCO-Datensatzes als Ausgangswerte dienten. Die Modelle YOLO und Faster-RCNN wurden auf einem Server mit zwei AMD EPYC 7643 Prozessoren, 256 GB RAM und zwei Nvidia A40 (46 GB) trainiert, während das MASK-RCNN auf einem Workstation-PC mit einem Intel Xeon W-2125, 32 GB RAM und einer Nvidia Titan V trainiert wurde.

Für das YOLO-Modell gibt es verschiedene Architekturen, die in der Breite und Tiefe der Modelle variieren. In dieser Studie wurden die S-, M- und L-Modelle getestet, um den Einfluss der Modellarchitektur genauer zu analysieren. Für die MASK-RCNN-Modelle werden die Backbone-Netze resnet50 und resnet101 verwendet und für das Faster-RCNN ein resnet50.

4.3.2 Auswertemetriken und Testdatensätze

Für beide Datenerzeugungs-Methoden sind etwa 100.000 Trainingsbilder pro Datensatz mit einer Auflösung von 600 x 600 Pixeln erzeugt worden. Die SketchGraph-Methode ist bei der Erzeugung von Bildern fehleranfälliger, da aufgrund von Laufzeitbeschränkungen die Verbindung zu Onshape teilweise fehlschlug, wodurch bis zu 100 Bilder weniger im Datensatz enthalten sein können. Beide Verfahren erzeugen die gleichen 14 Symboltypen: Festlager, Loslager, Drehgelenk, Kugelgelenk, geradverzahntes Stirnrad, schrägverzahntes Stirnrad, Kegelrad, Planetengetriebe, allgemeines Lager. Die letzten sechs Symbole werden in zwei Varianten erstellt: mit und ohne Hohlwelle. Die Darstellung der Symbole ist abgeleitet von bestehenden Sammlungen, welche in Kapitel 2.1.2.2 genauer beschrieben sind. Das Bild 43 zeigt die unterschiedlichen Symbole in allen Varianten.

Zum Vergleich der Erkennungsfähigkeiten werden die Bewertungsmetriken des COCO-Datensatzes [170], bestehend aus der mean Average Precison für einen IoU von 0,50 (mAP50) und der mean Average Precison (mAP), verwendet (siehe Kapitel 2.2.2.1). Beide Metriken werden für die Bewertung herangezogen, allerdings ist der mAP50 für den vorliegenden Anwendungsfall relevanter, da für die spätere Kontextualisierung die richtige Symbolklasse wichtiger ist als die exakte Position.

4.3.2.1 Datensatz - Getriebestufen

Neben den Metriken haben die unbekannten Testdatensätze eine entscheidende Rolle bei der Beurteilung der trainierten Objekterkennungsmodelle. Demzufolge ist der erste unbekannte Datensatz eine Sammlung von Prinzipskizzen verschiedener Getriebekonfigurationen. Insgesamt wurden 30 Getrie-

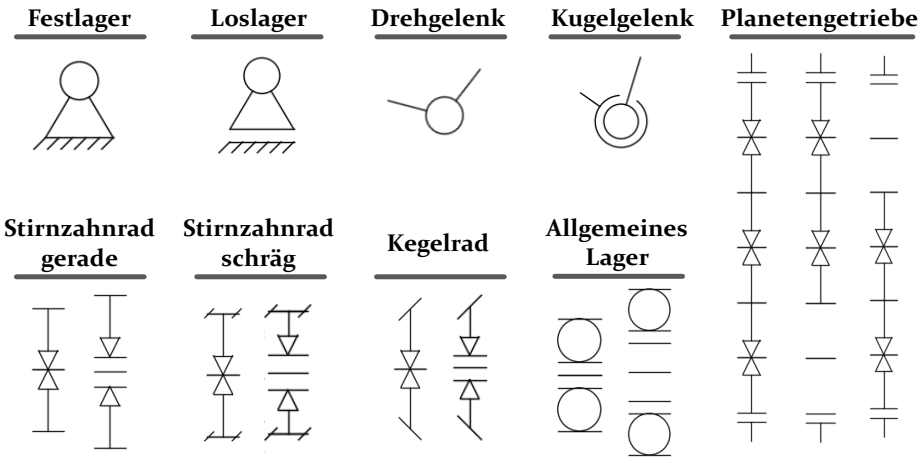


Bild 43: Sammlung der implementierten Symbole nach [S5]

bevarianten gezeichnet, die sich grob anhand der Getriebestufe unterscheiden lassen. Die Skizzen bestehen aus den Symbolen: Allgemeines Lager, gerad- und schrägverzahntes Stirnrad und Kegelrad sowie dem Gehäuse des jeweiligen Getriebes. Die genaue Anzahl der Instanzen pro Symbolklasse ist im Anhang in Tabelle 15 aufgeführt. Der Datensatz wurde manuell erstellt und gelabelt, wobei die einfachste Zeichnung sechs und die aufwendigste 22 Symbole enthält. Anschließend wurde der Datensatz in zwei Schritten skaliert, was zu insgesamt 90 Skizzen führte. Dazu wurde die größte Skizze als Maximum definiert und anschließend noch Zeichnungen mit 80 % und 66 % skalierten Symbolen erzeugt. Die verschiedenen Skalierungsmaßstäbe sollen eine zusätzliche Herausforderung für die Modelle darstellen und ihre Fähigkeiten bei der Erkennung verschiedener Symbolgrößen testen. Ein Beispiel für jede Stufe der Getriebe ist in Bild 44 dargestellt. Alle Zeichnungen sind im Anhang unter Kapitel B.2.2 aufgelistet.

4.3.2.2 Datensatz - Baugruppen

Der zweite unbekannte Datensatz besteht aus händisch abgeleiteten Skizzen von 3D-Baugruppen, welche aus dem „3D Assembly Repository“ von Lupinetti et al. [276] stammen. Dieser beinhaltet mehrere Baugruppenkategorien, in der jeweils verschiedene STEP-Baugruppen einsortiert wurden. Aus diesen Baugruppen werden komplexe Zeichnungen abgeleitet, wobei Skizzen für sechs verschiedene Kategorien erstellt werden. Bei den verwendeten Baugruppen handelt es sich um: Differential, Windturbinengetriebe, Flugzeug-Fahrwerk, Doppelrotor Turbine, hydraulische Reduktion und einen

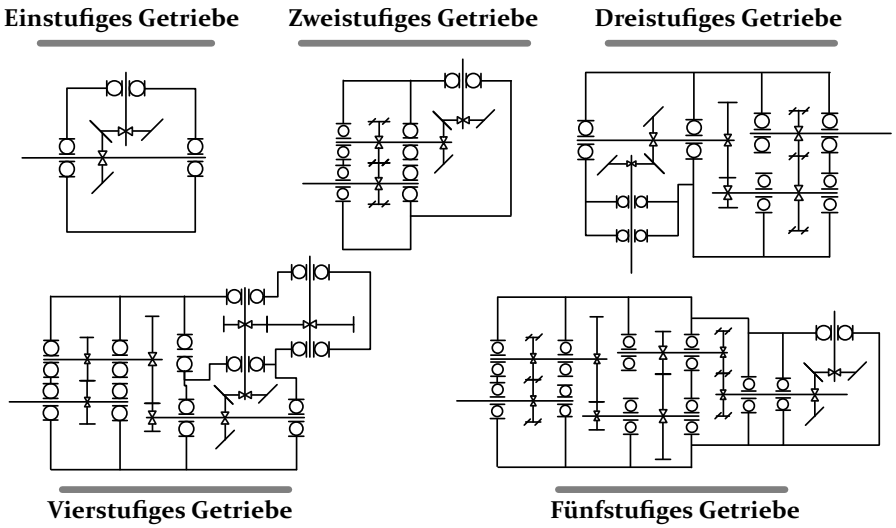


Bild 44: Beispiel-Darstellung der unterschiedlichen Getriebestufen

hydraulischen Rotor. Bild 45 zeigt jeweils eine Beispielskizze für die entsprechende Kategorie. Die exakte Instanzenanzahl der Symbole ist wiederum in Tabelle 16 im Anhang hinterlegt, zusammen mit allen Bildern des Datensatzes in Kapitel B.2.4.

4.3.3 Ergebnisse

Nach der Vorstellung der unterschiedlichen Arten für die synthetische Datenerzeugung, der verwendeten Modelle und Auswertedatensätze werden in diesem Abschnitt die erzielten Ergebnisse präsentiert und analysiert. Dazu sind die mAP50 Werte für alle Kombinationen aus Datensätzen und Modellen in der Tabelle 1 aufgelistet. Die Tabelle ist zweigeteilt aufgebaut, mit den Ergebniswerten für den Getriebenstufen Datensatz in der oberen und für den Baugruppen in der unteren Hälfte. Die zusätzlich erzielten mAP Werte sind im Anhang in Tabelle 17 zu finden.

Die Auswertemetriken für den Getriebestufen-Datensatz legen nahe, dass eine gute Erkennung relevanter Symbole erreicht wurde. Darüber hinaus zeigen die Metriken, dass verschiedene Datensätze zu deutlich unterschiedlichen Ergebnissen führen, wobei der niedrigste mAP50-Wert bei 0,042 und der höchste bei 0,994 liegt. Insgesamt erzielten die Modelle, die auf dem SketchGraph-Datensatz trainiert wurden, deutlich bessere Ergebnisse, insbesondere wenn eine realistische Datengenerierung gewählt wurde. Die Ausnahme bildet dabei der SketchGraph 2 Datensatz, der mit keiner Deep-Learning-

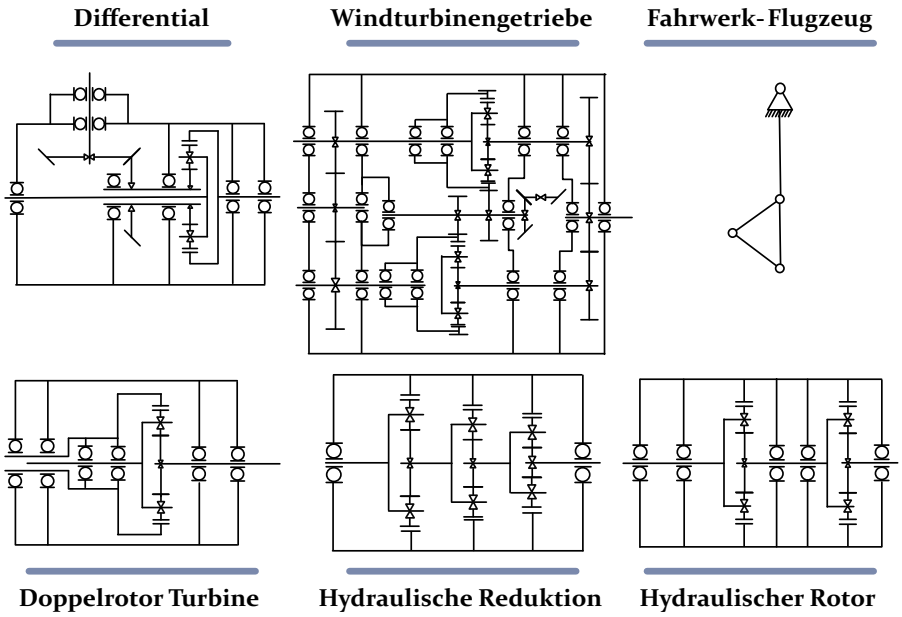


Bild 45: Beispiel für unterschiedliche Baugruppenklassen und die dazugehörigen Skizzen

Architektur funktioniert, weder mit RCNNs noch YOLO. Ein Vergleich der Modelle ergab eine deutliche Tendenz zugunsten der YOLO-Modelle, dies zeigt auch der durchschnittliche Unterschied zwischen beiden Modellen (0,302 mAP50).

Die Ergebnisse für den zweiten Datensatz belegen, dass das Erkennungsniveau im Vergleich zum ersten Datensatz sinkt. Dies ist vor allem auf die komplexeren Zeichnungen und die größere Vielfalt der gezeichneten Symbole zurückzuführen. Außerdem müssen in diesem Datensatz feinere Unterschiede zwischen Symbolen erkannt werden, z. B. das Kegelrad mit und ohne Hohlwelle. Die Erkenntnisse aus dem Getriebestufen Datensatz sind auch in der Auswertung zu finden, da die YOLO-Modelle wiederum besser abschneiden als die RCNN-Architektur und der SketchGraph 2 Datensatz die schlechtesten Ergebnisse liefert. Die Kombination aus Pillow-Datengenerierung und YOLO-Modellen erreicht ebenfalls gute mAP50-Werte, mit dem YOLOv5 L als beste Architektur. Die höchste Genauigkeit bietet ebenfalls das YOLOv5 L mit dem SketchGraph 3 Datensatz, mit einem mAP50 von 0,906 und mAP von 0,562.

Tabelle 1: mAP50 Ergebniswerte für alle Modelle und Trainingsdatensätze mit unbekannten Daten mit den besten Ergebnissen fett und unterstrichen hervorgehoben nach [P6]

Dataset	Sketch Graph 1	Sketch Graph 2	Sketch Graph 3	Sketch Graph 4	Pillow 1	Pillow 2
Testdatensatz Getriebestufen						
Yolov5 L	0,742	0,084	0,993	0,994	0,817	0,832
Yolov5 S	0,563	0,180	0,993	0,992	0,388	0,716
Yolov5 M	0,805	0,076	**0,994**	0,994	0,625	0,868
MASK-RCNN r50	0,149	0,080	0,741	0,812	0,439	0,235
MASK-RCNN r101	0,105	0,135	0,949	0,834	0,127	0,351
Faster-RCNN r50	0,286	0,241	0,736	0,757	0,042	0,197
Testdatensatz Baugruppen						
Yolov5 L	0,688	0,200	**0,906**	0,891	0,597	0,723
Yolov5 S	0,659	0,274	0,904	0,893	0,492	0,559
Yolov5 M	0,717	0,193	0,904	0,868	0,485	0,561
MASK-RCNN r50	0,111	0,080	0,501	0,451	0,189	0,011
MASK-RCNN r101	0,110	0,140	0,642	0,575	0,142	0,061
Faster-RCNN r50	0,352	0,365	0,628	0,664	0,144	0,392

4.3.4 Auswertung

Die Auswertung der verschiedenen Modelle und Datensätze hilft bei der Beantwortung der ersten Forschungsfrage. Dabei konnte eindeutig gezeigt werden, dass eine Erkennung von Symbolen in Prinzipskizzen mit dem vorgestellten Vorgehen möglich ist. Im Allgemeinen zeigen die Ergebnisse, dass dabei der Datensatz einen größeren Einfluss auf die Erkennungsgenauigkeit hat als das Modell. Für den dritten SketchGraph-Datensatz waren sowohl die YOLO- als auch RCNN-Modellen in der Lage, unbekannten Skizzen mit

hoher Genauigkeit zu erkennen, was auch in der oberen Zeile in Bild 46 dargestellt ist.

Ebenfalls zeigt der zweite SketchGraph-Datensatz, dass ein weniger repräsentativer Datensatz zu Problemen für alle Modelle führt, da aus den synthetischen Skizzen falsche Merkmale gelernt werden. Ein visuelles Beispiel für die Erkennungsprobleme mit dem SketchGraph 2 Datensatz ist in der unteren rechten Ecke von Bild 46 illustriert.

Der Einfluss der Modell-Architekturen auf die Erkennungsergebnisse wird bei den Pillow-Datensätzen deutlich, da die einstufigen Methoden besser funktioniert haben als die zweistufigen. Dagegen variieren die Resultate für die RCNN-Modelle stärker, weshalb kein Modell eindeutig als das beste identifiziert werden kann. Im Gegensatz dazu zeigen die YOLO-Modelle, dass die M- und L-Versionen sehr gut für die Erkennung geeignet sind, da sie auch für den Pillow-Datensatz höhere Erkennungsgenauigkeiten liefern als die S-Version.

Auch die Zusammensetzung des Testdatensatzes beeinflusst die Auswertung, zumal der Fokus bei der Erstellung auf realistischen Zeichnungen und weniger auf Quantität lag. Folglich sind Symbole im Trainingsdatensatz enthalten, die im Testdatensatz nicht vorhanden sind, z. B. das Kugelgelenk. Dieser Fakt wurde jedoch in Kauf genommen, da das Ziel darin bestand ein Modell zu trainieren, das möglichst breit anwendbar ist und nicht eine spezifische Lösung darstellt. Diese Anforderung führt auch zu Problemen bei der Symbolerkennung, beispielsweise kam es häufig zu Verwechslungen zwischen Drehgelenken und allgemeinen Lagern. Diese falsche Klassifizierung ist auch unten links in Bild 46 zu sehen. Da das Lagersymbol praktisch aus zwei Drehgelenkssymbolen besteht, ist die Verwechslung nachvollziehbar.

Eine weitere Herausforderung bei der Erkennung war die Skalierung der Zeichnungs-Symbole. Auffallend war, dass das RCNN-Modell vor allem bei kleineren Symbolen deutlich mehr Fehler bei der Erkennung hatte. Eine Anpassung der Datengenerierung mit einer größeren Varianz der Symbolgröße könnte hier eventuell zu einer allgemeinen Verbesserung führen. Zuletzt konzentrierte sich die Auswahl der Objekterkennungsmodelle auf die etablierten Methoden, weshalb auch andere Modelle berücksichtigt werden können, z. B. Ansätze mit visuellen Transformatoren [390, 391] oder graphbasierten Neuronalen Netzen [392].

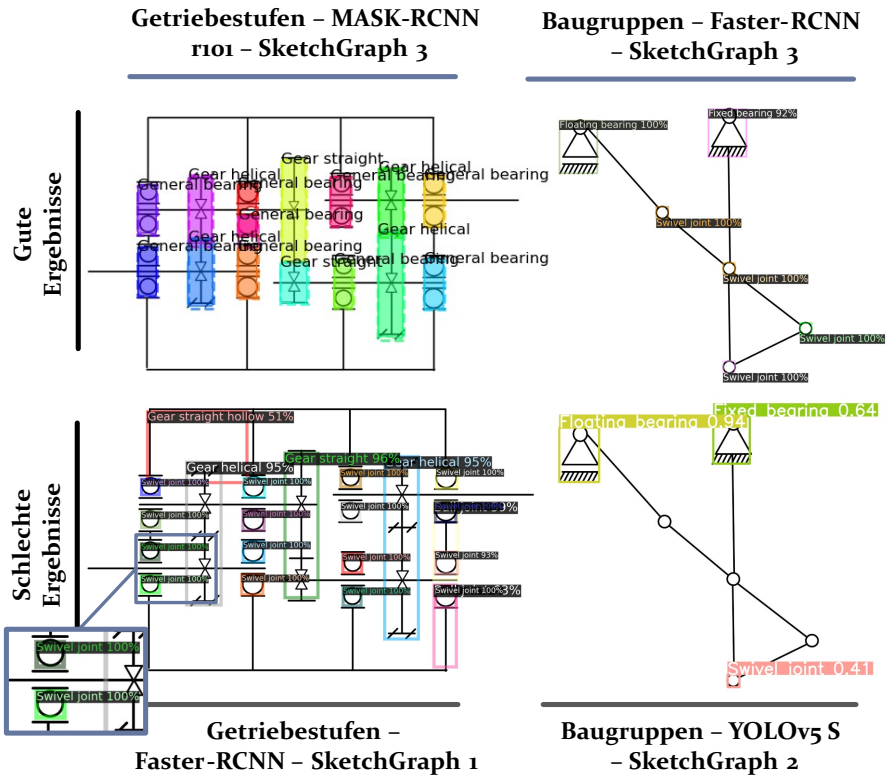


Bild 46: Gute und schlechte Erkennungsbeispiele für die trainierten Modelle und verschiedene Trainingsdatensätze

Fazit Kapitel 4.3

Für die Untersuchung der Symbolerkennung sind jeweils sechs unterschiedliche Objekterkennungsmodelle mit sechs Trainingsdatensätzen mit jeweils 100.000 Bildern trainiert und mit zwei unbekannten Datensätzen ausgewertet worden. Zusammenfassend konnte gezeigt werden, dass Objekterkennungsmodelle in der Lage sind Symbole in technischen Skizzen zu erkennen. Dabei zeigte die Auswertung, dass manche Kombinationen aus Datenerzeugung und Deep-Learning-Modell (RCNN in Kombination mit Pillow) eine schlechte Erkennung erzielten, allerdings die Erzeugung mittels SketchGraph und YOLOv5-Modellen sehr gute Ergebnisse lieferte.

4.4 Iteration Symbolerkennung

Die Erkenntnisse aus den beiden vorgestellten Datensätzen sollen im Anschluss zur Verbesserung der Datenerzeugung genutzt werden. Da der SketchGraph-Ansatz besser als die Pillow Version funktioniert hat, wird dieser als Grundlage verwendet. Das Ziel der Iteration ist neben der Erhöhung der Erkennungsgenauigkeit auch eine Potentialanalyse zur Erkennung von Handskizzen. Zunächst werden drei Aspekte für den neuen Datensatz angepasst.

Als Erstes wird der Faktor für den Handskizzen Modus modifiziert und auf 0,5 gesetzt. Damit soll die Verteilung zwischen den Renderarten der Symbole ausgeglichener und beide Darstellungsformen im Datensatz repräsentiert werden. Weiterhin wurde die Darstellung des Drehgelenksymbols überarbeitet, da dieses häufig mit dem allgemeinen Lager verwechselt bzw. übereinander angeordnet wurde. In der neuen Version werden die Verbindungslinien immer in Kombination mit dem Drehgelenk gezeichnet und nur ausgeblendet, wenn eine Verbindung mit einem anderen Symbol vorgegeben wird. Zuletzt wurde die Gesamtverteilung der Symbole adaptiert, da bisher vor allem die Hohlwellen-Varianten weniger häufig im Datensatz eingebaut waren. Die genaue Auflistung der Häufigkeit pro Symbolklasse ist im Anhang in Bild 100 grafisch aufbereitet.

Das Diagramm in Bild 47 zeigt den Vergleich des Yolo V5 L Modells mit der neuen Datenerzeugung zu den besten Kombinationen aus der vorherigen Studie. Dabei liegen die erreichten Genauigkeiten mit dem neuen Vorgehen auf einem gleichen Niveau, die Metriken ändern sich nur im dreistelligen Nachkommabereich.

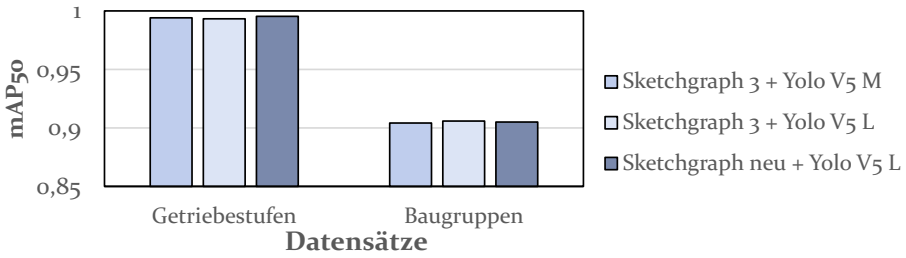


Bild 47: Vergleich des neuen Datenerzeugungsvorgehens mit den vorherigen Varianten

Abschließend soll noch die Fähigkeit der Modelle überprüft werden, handgezeichnete Skizzen zu erkennen. Dazu ist im Rahmen einer Masterarbeit [S5] ein Datensatz aus 26 handgezeichneten Skizzen erstellt worden, welche die Getriebestufenbilder als Vorbilder hatten. Die Prinzipskizzen wurden mit

Bleistift gezeichnet und anschließend eingescannt und gelabelt. Es wurde keine zusätzliche Aufbereitung mittels Filter oder anderer Methoden durchgeführt.

Tabelle 2: Ergebnismetriken für die neuen Trainingsdatensätze auf handgezeichneten Daten

Testdatensatz Handskizzen				
Dataset	SketchGraph 3		SketchGraph neu	
Metric	mAP50	mAP	mAP50	mAP
YOLOv5 M	0,953	0,500	0,981	0,574
YOLOv5 L	**0,985**	0,529	0,983	**0,579**

In der Tabelle 2 sind die Ergebnismetriken für den Handskizzen Datensatz aufgeführt. Die Werte zeigen zuerst, dass alle Modelle sehr hohe Genauigkeiten erzielen und wenig voreinander abweichen, vor allem beim Vergleich des mAP50. Allerdings erreichte das Modell mit dem neu erzeugten Trainingsdatensatz eine Verbesserung des mAP von 0,05 gegenüber den anderen beiden Modellen. Diese Verbesserung lässt darauf schließen, dass die Iteration der Datenerzeugung zu einem robusteren Modell führt, welches zwar die vorhandenen Testdatensätze nicht besser als die anderen Modelle erkennen kann, aber bei stärkeren Abweichungen vom Trainingsdatensatz bessere Ergebnisse erzielt. Die Resultate der Handskizzenerkennung zeigt Bild 48, indem zwei Beispielskizzen und die dazu ermittelten Boundingboxen dargestellt sind. Die durchgeführte Symbolerkennung wurde mit der neuen Datenerzeugungsiteration und einem Yolo V5 L durchgeführt.

Fazit Kapitel 4.4

Eine weitere Verbesserung des Ansatzes konnte durch die Überarbeitung der Datenerzeugung anhand der Erkenntnisse aus der Untersuchung in Kapitel 4.3 erreicht werden. Dabei ist die SketchGraph Datenerzeugung überarbeitet worden, indem einzelne Symbole klarer in den Trainingsdaten erzeugt wurden. Zudem ist das Potential zur Erkennung von Handskizzen mittels eines Testdatensatzes überprüft und positiv evaluiert worden.

4.5 Kontextualisierung

Nach der Präsentation der Symbolerkennung in technischen Skizzen soll in diesem Kapitel die Kontextualisierung der erkannten Symbole vorgestellt

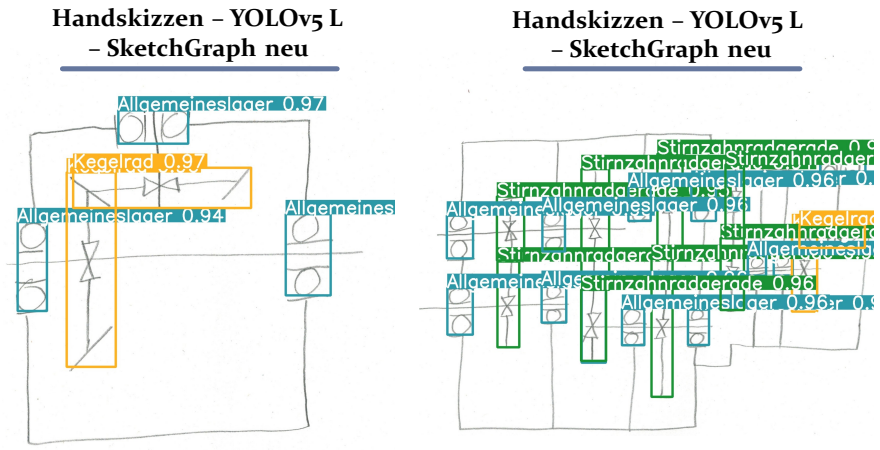


Bild 48: Exemplarisches Testbild mit detektierten Symbolen des mit Datensatz 2 trainierten Modells

werden. Das Vorgehen und die Demonstratoranwendung sind größtenteils in [P5] veröffentlicht. Die Anwendungsfelder sind sehr vielfältig und können daher nach Domäne und Zielsetzung stark variieren. Damit die erkannten Symbole aber möglichst allgemein genutzt werden können, müssen diese zunächst in ein unabhängiges Format übertragen werden. Dazu bietet sich der Netlist-Ansatz aus Kapitel 2.2.2.4 an, der eine Überführung der Bounding Boxen in einen Graphen ermöglicht. Dieser aus der Skizze abgeleitete Graph ist die Grundlage für die nachfolgenden Anwendungsmöglichkeiten, beispielsweise eine Ähnlichkeitssuche oder eine Umwandlung in andere Produkt- bzw. Systemmodelle. Nach dieser Ablauffolge ist auch das Kapitel aufgebaut, indem zunächst die Transformation der Skizzen in Graphen gezeigt wird und nachfolgend unterschiedliche Nutzbarmachungen mittels verschiedener Demonstratordatensätze.

4.5.1 Transformation in Graphen

Die grundlegende Idee der Umwandlung von Skizzen in Graphen ist, den Inhalt in bestmöglicher Qualität zu erfassen. Dazu werden die Verbindungslinien der Zeichnung benötigt, da diese die Symbole miteinander verknüpfen. Dementsprechend müssen diese zunächst extrahiert und identifiziert werden, damit aus den Bounding Boxen ein Graph erzeugt werden kann. Der gesamte Ablauf zur Überführung der Skizze in Graphen ist in Bild 49 dargestellt. Das entwickelte Vorgehen orientiert sich an dem Netlist-Ansatz von Fu und Kara [191] und übernimmt den ersten Schritt der Methode.

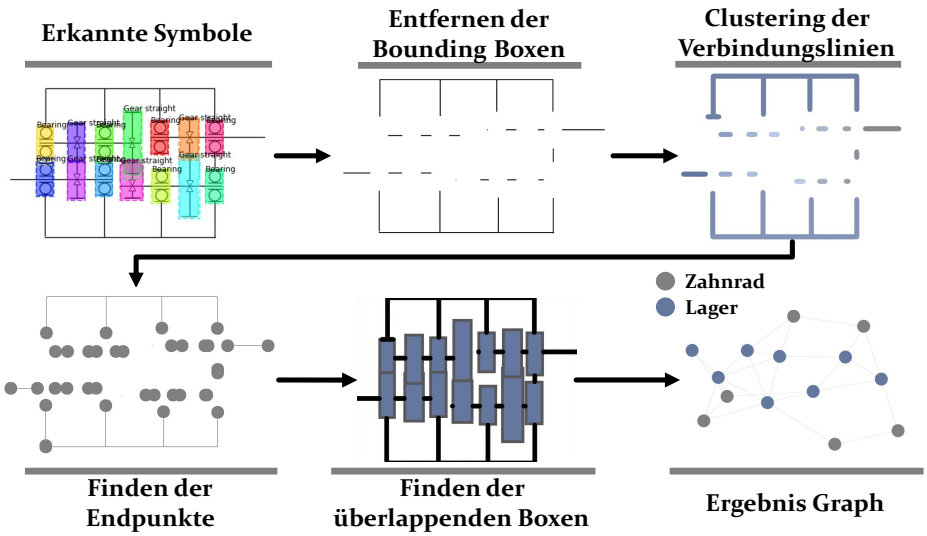


Bild 49: Vorgehen zur Umwandlung der erkannten Symbole in Graphen nach [P5]

Zunächst werden die Bounding Boxen aus der Skizze entfernt, so dass nur die Verbindungslinien zwischen den Symbolen übrig bleiben. Das dadurch entstandene Bild wird einem dbscan-Cluster-Algorithmus [393] übergeben, welcher die einzelnen Linien separiert. Damit die Zuordnung besser funktioniert, werden die Linien vorab skelettiert. Der Einsatz des dbscan Clusterings ermöglicht es, die Linien über die Hyperparameter zu filtern und erlaubt somit, abhängig vom Anwendungsfall, sehr kleine Elemente zu berücksichtigen oder zu ignorieren. Anschließend werden über eine Hit-or-Miss-Transformation [394] für jedes Cluster die offenen Endpunkte ermittelt. Diese Transformation ist in der Lage mehrere offene Enden einer Linie zu erkennen, was eine notwendige Eigenschaft für den Einsatz mit Prinzipskizzen ist. Die offenen Endpunkte und Bounding Boxen der Symbolerkennung werden mittels einer Kollisionserkennung geprüft und anschließend verknüpft, wodurch eine Verbindung im Graph zwischen den Symbolen ermöglicht wird. Dabei ist die Größe der Endpunkte wählbar, größere Punkte führen dabei zu einer besseren Kollisionserkennung, können aber auch zu falschen Verknüpfungen führen. Die letzte fehlende Information für den Graphen sind Verbindungen zwischen den Bounding Boxen ohne eine zusätzliche Verbindungslinie. Dazu werden die Bounding Boxen auf eine Überschneidung untereinander geprüft und gegebenenfalls miteinander verknüpft. Mit diesen beiden Informationen als Grundlage kann aus der Skizze ein Graph erstellt werden, welcher als Knoten die jeweilige Symbolklasse hat und über die Kanten miteinander ver-

bunden ist. Ein beispielhaftes Ergebnis ist rechts unten in Bild 49 dargestellt, wobei die blauen Knoten die Lager und die grauen die Zahnräder darstellen.

4.5.2 Ähnlichkeitssuche

Eine erste Nutzungsmöglichkeit für die abgeleiteten Graphen ist die Ähnlichkeitssuche, mit der Zielsetzung auf Basis unterschiedlicher Input-Modalitäten die ähnlichste Skizze aus einem Datenbestand vorhandener Skizzen zu finden. Zunächst werden deshalb Ansätze für eine textbasierte Suche präsentiert, gefolgt von Bildern bzw. Graphen als Sucheingabe. Durch diese automatisierte Nutzung vorhandener Daten können Problemstellungen durch einen Überblick der vorhandenen Lösungen schneller bewältigt werden. Die vielfältigen Input-Formate gewährleisten zusätzlich eine möglichst breite Anwendung des Vorgehens.

4.5.2.1 Demonstratordatensatz

Für die Umsetzung der unterschiedlichen Suchprozesse ist eine existierende Sammlung an Prinzipskizzen notwendig. Diese wird durch die Sammlung an Getriebevarianten aus Kapitel 4.3.2.1 imitiert, welche sich nach der Getriebestufe unterscheiden lassen. Insgesamt sind 30 Getriebevarianten für den Demonstratordatensatz ausgewählt worden, wobei nur eine einheitliche Skalierung für die Skizze verwendet wurde. Die Anzahl und Symbolarten sowie Beispielbilder sind ausführlich in Kapitel 4.3.2.1 beschrieben.

Alle Zeichnungen werden einem, mit dem SketchGraph 5 trainierten, YOLO V5 L übergeben und nach der erfolgreichen Erkennung automatisiert in Graphen umgewandelt. Der Pool an vorhandenen Daten besteht somit aus 30 Graphen, welche für die Suche genutzt werden. Für die Bearbeitung und Darstellung der Graphen wird das networkX-Paket [395] verwendet.

4.5.2.2 Text

Die erste Option ist die Verwendung von Text als Sucheingabe, wozu die Datenbank an Graphen zunächst in eine Tabelle überführt werden muss. Die Unterscheidungsmerkmale sind generell frei wählbar, wurden aber diesen Fall auf die Anzahl an Symbolarten pro Zeichnung festgelegt. Diese Umwandlung ermöglicht das Suchen oder Filtern der Tabelle nach bestimmten Einträgen. Die konkreten Kategorien für den Demonstratordatensatz sind die Anzahl an Lagern, die Getriebestufen und die Anzahl an den spezifischen Zahnrädern.

Zudem bietet diese Umwandlung eine schnelle Überprüfung der Erkennung und Transformation. Für diesen Datensatz ist die Plausibilitätsprüfung über einen Vergleich der Zahnrad-Knoten im Kontakt umgesetzt, welcher testet,

ob nur gleiche Zahnradtypen im Graphen verknüpft sind. Ist dies nicht der Fall, wird der Graph nicht in den Datenbestand übernommen und der Graph oder die Skizze müssen nochmals händisch durch einen zuständigen Ingenieur korrigiert oder angepasst werden. Eine exemplarische Suchfrage für die Textsuche aus der Datenbank könnte lauten:

„*Getriebestufe fünf* und Ergebnisse nach Anzahl an *schrägverzahnten Zahnrädern* und *Kegelrädern* sortiert"

Das Bild 50 zeigt die Ergebnisse dieser Suchanfrage in Form eines pandas dataframes [396, 397]. Das beste Ergebnis wird als erster Eintrag aufgelistet, gefolgt von den weniger ähnlichen Ergebnissen. Die Nummer links in dem dataframe beschreibt die Zeichnungsnummer der Skizze, welche im Anhang unter Kapitel B.2.1 abgebildet ist.

	Anzahl Lager	Getriebestufen	Anzahl Zahnräder gesamt	Anzahl Zahnräder gerade	Anzahl Zahnräder schräg	Anzahl Kegelräder	Zahnradpaare korrekt
28	11	5	10	0	8	2	True
29	10	5	10	0	8	2	True
26	11	5	10	4	4	2	True
24	11	5	10	8	0	2	True
25	10	5	10	8	0	2	True

Bild 50: Screenshot des Suchergebnisse für eine exemplarische Text-Eingabe

4.5.2.3 Graph

Die Verwendung von Graphen als Eingabe für die Abfrage ist eine weitere Möglichkeit, die Ähnlichkeitssuche zu nutzen. Der Input-Graph kann auf verschiedene Arten erstellt werden, beispielsweise indem die Knoten und Kanten entsprechend des aktuellen Entwurf erzeugt werden. Zudem kann eine computergenerierte oder handgezeichnete Prinzipskizze genutzt werden, die nach dem im Abschnitt 4.5.1 beschriebenen Ablauf umgewandelt wird.

Unabhängig von der Erstellung des Suchgraphen sind viele generelle Algorithmen zum Vergleich von Graphen entwickelt worden. Das besondere Merkmal von Graphen, die aus Prinzipskizzen abgeleitet werden, sind die Attribute der einzelnen Knoten. Diese entsprechen der Symbolbezeichnung und müssen bei der Suche berücksichtigt werden, wodurch ein reiner Topologievergleich der Graphen ungeeignet ist. Aus diesem Grund wurde die Graph Edit Distance (GED) als Ähnlichkeitsmaß gewählt [398], da diese Metrik die Attribute des Graphen berücksichtigt. Das Distanzmaß beschreibt den kleinsten Änderungspfad, damit zwei Graphen isomorph sind, und basiert auf der Levenshtein-Distanz [399], welche die Anzahl der Änderungsoperationen für Wörter beschreibt. Die Kosten für die verschiedenen Graph-Operationen

(Knoten oder Kante, hinzufügen oder entfernen) können frei gewählt werden, im Rahmen dieser Untersuchung wurden allerdings alle Änderungskosten gleich gewichtet.

Nachfolgend soll eine handgezeichnete Skizze als Beispiel für eine Suche mit Graphen dienen, indem die sechste Skizze aus dem Datensatz als Handzeichnung erstellt wurde. Die abfotografierte Zeichnung ist lediglich mittels Image Thresholding (50 %) in ein Graustufenbild umgewandelt und dann an das trainierte Modell übergeben worden. Die Transformation erfolgte nach dem vorgestellten Prinzip und das Ergebnis der Suche ist in Bild 51 dargestellt. Das Bild zeigt, dass die gefundenen Ergebnisse mit der Anfrage übereinstimmen, da alle Ergebnisse die gleichen Getriebestufen aufweisen. Das zweite und dritte Ergebnis haben jeweils den gleichen Ähnlichkeitswert und liegen daher beide auf dem zweiten Platz. Die gesamten Ergebniswerte für die GED-Distanz zwischen der Inputskizze und dem Datensatz sind in der Tabelle 18 im Anhang angefügt, zusammen mit den relevanten Parametern der Umwandlung.

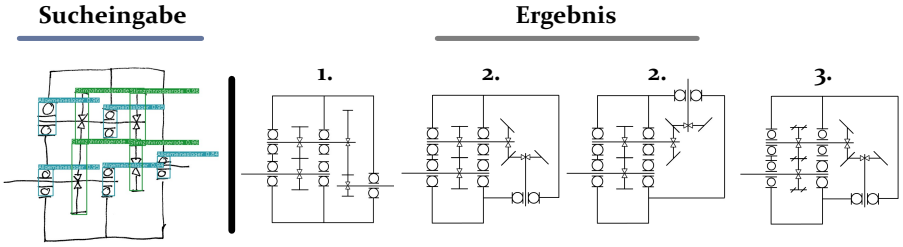


Bild 51: Beispiel für die Suche mittels Graph-Input und handgezeichneter Skizze

4.5.2.4 Subgraph

Ein weiterer Vorteil der graphenbasierten Suche ist, dass Teilgraphen für die Suche verwendet werden können. Die Algorithmen hierfür werden oft als Subgraphen-Matching bezeichnet und prüfen, ob der Suchgraph in einem der Datenbankgraphen vorkommt. Für den Demonstrator wurde ein VF2-Algorithmus [400, 401] genutzt, der ebenfalls in der Lage ist, die Knotenattribute zu berücksichtigen. Das Prinzip hinter dem Algorithmus ist ein Entscheidungsbaum, bei dem in jedem Schritt der jeweilige Knoten des Suchgraphs mit dem Vergleichsgraphs überprüft wird. Im Vergleich zur GED-Distanz ist das Resultat der Suche keine Rangliste, sondern immer ein boolesches Ergebnis. Dies zeigt auch die Anwendung auf den Demonstratordatensatz in Bild 52. In diesem Beispiel besteht die Eingabesequenz aus einem Lager, zwei geraden Zahnrädern, einem Lager, zwei Kegelrädern und einem Lager.

Das Ergebnisbild zeigt die jeweilige Abfolge hervorgehoben in der originalen Prinzipskizze.

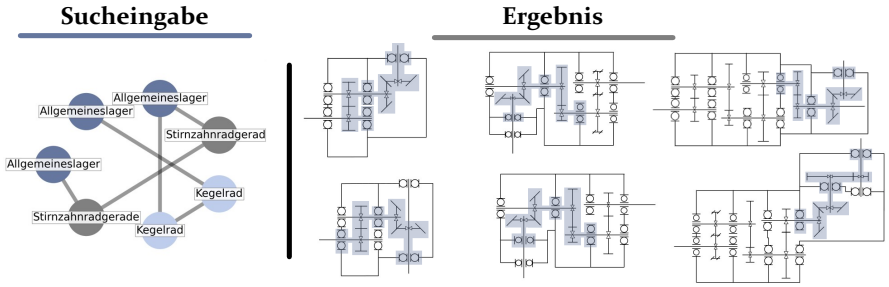


Bild 52: Auszug der Ergebnisse für das Subgraph-Matching nach [P5]

4.5.3 SysML Überführung

Nach der erfolgreichen Präsentation unterschiedlicher Möglichkeiten der Kontextualisierung im Rahmen von Ähnlichkeitssuchen soll eine weitere Art der Anwendung gezeigt werden. Die Idee ist, eine Schnittstelle zu bestehenden Produkt- bzw. Systementwicklungsumgebungen herzustellen, damit Prinzipskizzen einen breiteren Einsatz finden können. Dementsprechend soll eine Verknüpfung der transformierten Graphen in System Modeling Language (SysML) Modelle erfolgen, welche eine etablierte Systemrepräsentation im Systems-Engineering darstellen.

Die SysML Sprache basiert auf der Unified Modeling Language, die zur Modellierung und Darstellung von Software entwickelt wurde [402]. Es wurde von der Object Management Group (OMG) zur Unterstützung des Entwicklungsprozesses komplexer Systeme geschaffen. Dabei soll die Anwendbarkeit sehr allgemein sein, weshalb sowohl Anforderungen, Verhalten oder Struktur damit modelliert werden können [403, 404]. Die SysML stellt grundsätzlich neun Diagrammtypen zur Verfügung, von denen Blockdefinitionsdiagramme sehr häufig zur Beschreibung der Systemhierarchie verwendet werden. Das Verhalten oder die Eigenschaften von Systemelementen kann durch diese Diagramme repräsentiert werden, wobei sich ein Diagramm immer aus Blöcken und Verbindungen zwischen diesen zusammensetzt [405].

Das Ziel war, eine Schnittstelle zu entwickeln, welche den umgewandelten Graphen als Input bekommt und anschließend ein xml-File für den Import in Enterprise Architect erstellt. Die ganze Umwandlung läuft automatisch ab, indem die Knoten und Kanten in passende Blöcke und Verbinder übertragen werden. In Bild 53 ist ein Beispiel für die erste Skizze aus dem Getriebestufen-

Datensatz gezeigt. Auf der rechten Seite des Bildes ist das generierte Blockdiagramm in Enterprise Architect als Ergebnis präsentiert.

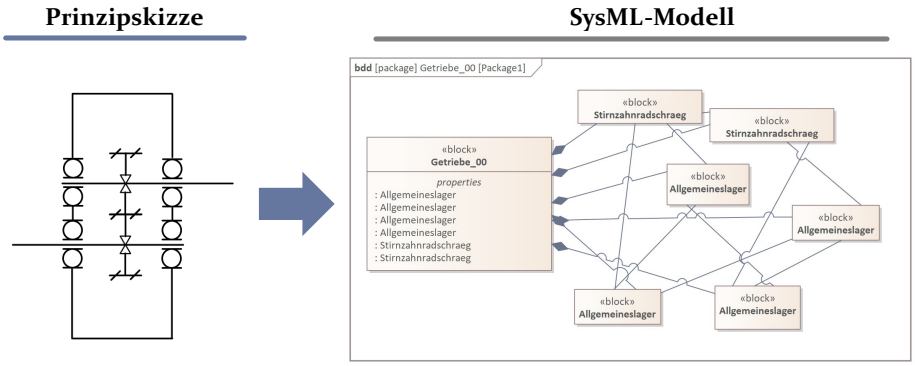


Bild 53: Beispiel für die Schnittstelle zwischen Skizzen und SysML Modellen

Der Vorteil dieser Verknüpfung ist die Möglichkeit aus Skizzen automatisch ein Systemmodell zu generieren, welches den Startpunkt für die weitere System- oder Subsystem-Entwicklung darstellt. Die händische Erzeugung von Verbindungen und Blöcken wird somit vermieden und Skizzen können weiterhin zur schnellen Ideenerzeugung und -iteration verwendet werden. Insgesamt ermöglicht das Vorgehen eine frühzeitige Überführung von Produkten in ein SysML Modell, ganz im Sinne des Frontloadings. Darüber hinaus bildet die XML-Verknüpfung die Grundlage für eine bidirektionale Schnittstelle. Dies würde die automatische Generierung von Skizzen aus den Diagrammen ermöglichen, was bei der Visualisierung und Erklärung von Systemen helfen kann.

4.5.4 Weitere Anwendungsmöglichkeiten

Neben den bisher genannten Beispielen wie der Ähnlichkeitssuche oder der Verknüpfung von Skizzen und SysML, sind auch weitere Anwendungsfelder für die umgewandelten Graphen sinnvoll. So kann der konvertierte Graph zum Beispiel mit Hilfe von Ontologien automatisiert mit Informationen erweitert werden. Dazu werden produkt- oder domänenspezifische Ontologien benötigt, die Zusammenhänge erkennen und diese in den Graphen übertragen. Ein Beispiel ist das von Goetz et al. [406] entwickelte System, das Robust Design und Toleranzmanagement kombiniert. Für diese Methode kann der vorgestellte Ansatz als Vorbereitungsschritt genutzt werden, um eine höhere Automatisierung zu erreichen.

Ein weiteres vielversprechendes Anwendungsfeld ist die Übertragung anderer Ergebnisse des Produktentwicklungsprozesses in Graphen, was eine for-

matübergreifende Ähnlichkeitssuche und damit Zeit- und Kosteneinsparungen ermöglicht. Die Nutzung von Anforderungen aus der Produktentwicklung wäre hier denkbar, indem mittels Natural-Language-Processing (NLP)-Verfahren relevante Zusammenhänge aus den Anforderungen extrahiert und in eine graphbasierte Form überführt werden. Erste Ansätze hierzu wurden von Kestel et al. [407] primär mit Fokus auf FEM-Simulationsberichte durchgeführt. Ebenfalls geeignet sind hierfür Geometrien in Form von Baugruppen, welche mittels Ähnlichkeitsmetriken verglichen werden, was eine skizzenbasierte Suche nach Geometrien ermöglichen würde. Diese Form der formatübergreifenden Suche soll im folgenden Kapitel genauer analysiert und untersucht werden.

Fazit Kapitel 4.5

In diesem Kapitel konnten verschiedene Anwendungsbeispiele für die Nutzung und Weiterverwendung von erkannten Prinzipskizzen dargelegt werden. Dazu sind zunächst unterschiedliche Arten der Ähnlichkeitssuche analysiert, implementiert und getestet worden, mit dem Getriebestufendatensatz aus Kapitel 4.3.2.1 als Datengrundlage. Als Input für die Suche sind Text-Eingaben, Graphen und Subgraphen umgesetzt und mittels des Datensatzes getestet worden. Neben der Suche ist auch die Umwandlung in SysML Diagramme präsentiert worden, somit ermöglichen die vorgestellten Anwendungsmöglichkeiten eine durchgängige Datennutzung und breitere Anwendung von Prinzipskizzen im Produktentwicklungsprozess.

4.6 Diskussion

Das präsentierte Modul zur Erkennung und Kontextualisierung von Skizzen ist in der Lage automatisch Skizzen in Graphen zu überführen und anschließend eine Suche auf Basis der erzeugten Datenbank umzusetzen. Damit ist direkt die erste Forschungsfrage beantwortet, indem die Kombination aus einer synthetischen Datenerzeugung mittels des SketchGraph Packages und dem YOLOv5 L Netzwerk eine Erkennung von Zeichnungselementen in Prinzipskizzen möglich wird. Zudem konnten die verschiedenen Anwendung für die transformierten Skizzen gezeigt und angewendet werden, mit dem Fokus auf der Ähnlichkeitssuche und der Umwandlung in SysML Modelle.

Ein Nachteil ist der hohe Aufwand für die Datenerzeugung und das Training der Deep-Learning-Modelle. Diese benötigen durch ihren vorgegebenen Aufbau und Architektur eine große Datenmenge, damit die verschiedenen Layer ausreichend trainiert werden können. Demzufolge ist im Rahmen dieser

Umsetzung auf Transfer-Learning zurückgegriffen worden, um den initialen Aufwand des Trainings zu reduzieren. Dieser bleibt aber im Vergleich zu nicht Deep-Learning-Vorgehen immer noch relativ hoch.

Die Stärken des präsentierten Vorgehens liegen in der hohen Automatisierung und fehlenden händischen Aufbereitung. So ist einzig die Definition der Symbole für den erfolgreichen Einsatz des Vorgehens notwendig. Dabei müssen die Regeln für das Zeichnen der Symbole für die Datenerzeugung implementiert werden, was allerdings auch den Vorteil bietet, eigenständige Symbole in das Vorgehen zu integrieren. Der einzige sonstige Input für das Vorgehen sind die Auswahl-Features beim Übertragen der Graphen in eine Tabelle, wobei hier die Anzahl der Symbole als Standard-Ansatz immer verwendet werden kann. Zudem können neue Algorithmen zur Objekterkennung jederzeit aufgenommen werden, um die Erkennungsleistung weiter zu steigern. Somit bietet das Modul auch die Möglichkeit weitere Zeichnungen aus dem Bereich der Produktentwicklung nutzbar zu machen, beispielsweise technische Zeichnungen. Eine Erweiterung auf andere Zeichnungsarten ist auch denkbar, da das Vorgehen allgemeingültig einsetzbar ist. Anhand von technischen Zeichnungen ist dies in der Arbeit [S4] prototypenhaft umgesetzt worden.

5 Ähnlichkeitssuche von Baugruppen mittels Skizzen

Nach der Präsentation der automatischen Erkennung und Nutzung von Prinzipskizzen wird in diesem Kapitel ein Vorgehen präsentiert, welches die Verknüpfung von Prinzipskizzen und Baugruppen ermöglicht. Zunächst wird die Motivation und das Potential für den Produktentwicklungsprozess dargelegt, gefolgt von der Vorstellung des Moduls. Dieses wird im Anschluss ausführlich erklärt und die einzelnen Schritte genau beschrieben. Nach der Erläuterung der Kontakterkennung, Bauteilerkennung und dem Matching der Graphen wird das Vorgehen mit zwei Demonstratordatensätzen getestet und evaluiert. Das Vorgehen zur Überführung der Baugruppen in Graphen ist teilweise in [S8] bearbeitet worden und das Aufbereiten und Matching in [S9].

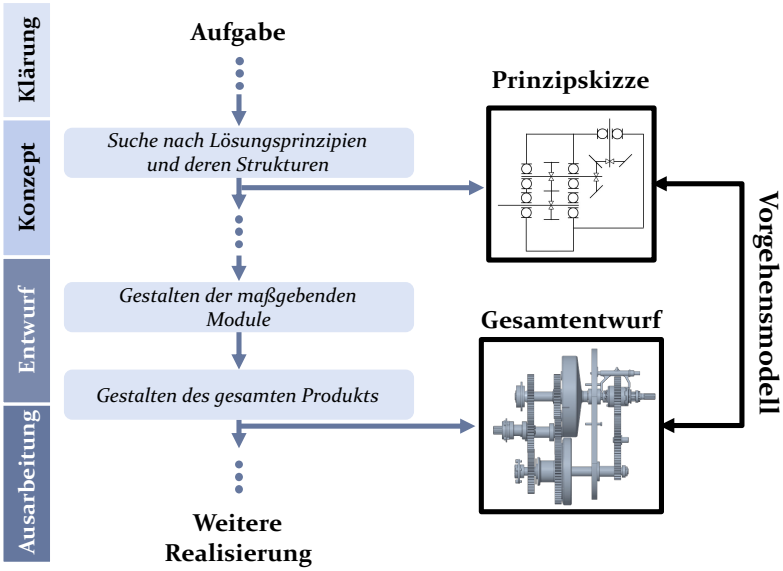


Bild 54: Einordnung des Vorgehens zum Finden von Baugruppen auf Basis von Zeichnungen in die VDI 2221 [26]

In Bild 54 ist eine gestaffelte Übersicht der VDI 2221 abgebildet, sortiert anhand der unterschiedlichen Entwicklungsphasen. Die Grundidee des neuen Vorgehens ist, die Ergebnisse verschiedener Entwicklungsphasen zu verknüpfen, trotz der unterschiedlichen Modalität der Ergebnisse. So soll es nach der erfolgreichen Durchführung des Ansatzes möglich sein, mittels einer Skizze eine Baugruppe aus der Entwurfs- oder Ausarbeitungsphase zu

finden und direkt zu Beginn des Entwicklungsprozesses diese zusätzlichen Informationen zur Verfügung zu haben. Somit werden vorhandene Daten deutlich effizienter genutzt, da keine Geometrie als Suchinput erstellt werden muss. Die gefundenen Baugruppen können entweder als Startpunkt für die Konstruktion dienen, indem sie nur noch auf die neue Problemstellung adaptiert werden müssen oder als Inspiration zum Lösen des gestellten Problems. Dabei soll das Vorgehen möglichst automatisiert ablaufen und keinen zusätzlichen Input benötigen, weshalb für die Baugruppen das universelle STEP-Format [408] verwendet wird.

5.1 Modulübersicht

Das entwickelte Sub-Vorgehensmodell für das Modul ist in einer Prozessübersicht in Bild 55 dargestellt und lässt sich in mehrere Teilschritte untergliedern. Die Transformation einer Prinzipskizze in einen Graphen ist der erste Teilschritt und bereits detailliert in Kapitel 4 erläutert worden. Die Baugruppen müssen ebenfalls in einen Graphen übertragen werden, damit ein Abgleich über ein gemeinsames Format möglich ist. Der aktuelle Stand der Forschung dazu ist in Kapitel 2.2.3.3 erklärt, zusammen mit Beispielen aus der Verfahrenstechnik für das Matching von Skizzen zu Geometrien [366] oder über das Konzept Bewegungsfunktionen technischer Systeme mit Bauteilen oder Baugruppen zu verknüpfen [271].

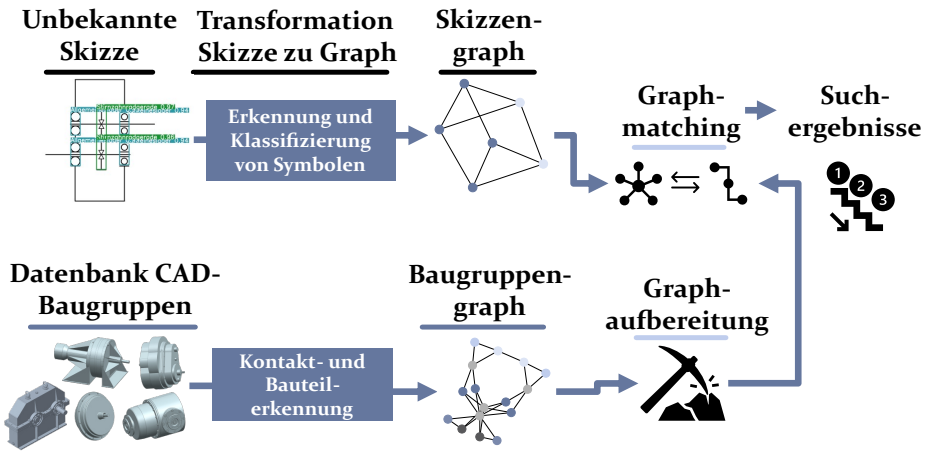


Bild 55: Übersicht des entwickelten Moduls zur Verknüpfung von Skizzen und Baugruppen

Die Transformation der Baugruppe in einen Graphen ist in mehrere Teilschritte gegliedert und orientiert sich in den ersten Schritten an dem Prozess von Lupinetti et al. [245]. Zunächst muss die Baugruppe in ihren einzelnen

Bauteile aufgeteilt und anschließend der Kontakt zwischen diesen ermittelt werden. Zusätzlich ist die Bauteilart bzw. -klasse in der Baugruppe relevant, welche über die vergebenen Bauteilnamen in der Baugruppe oder über ein trainiertes Deep-Learning-Modell ermittelt werden sollen. Anschließend können die beiden Informationen zu einem Graphen verknüpft werden.

Dieser Graph könnte dann für eine Suche auf der Baugruppen-Ebene verwendet werden, muss allerdings für die Suche mittels Skizzen noch weiter aufbereitet werden. Der Grund ist der höhere Detailgrad auskonstruierter Baugruppen, welche mehr Bauteile und Informationen enthalten als eine Prinzipskizze. Daher muss der Baugruppen-Graph vereinfacht werden, was einen Vergleich mit dem Prinzipskizzen-Graph zulässt. Über verschiedene Matchingverfahren kann nachfolgend der ähnlichste Graph aus dem Datenpool an Baugruppen-Graphen ermittelt und somit eine fertig entwickelte Baugruppe mit einem Skizzen-Suchinput verknüpft werden. Die einzelnen Prozessabschnitte werden in den nächsten Kapiteln vorgestellt und abschließend anhand zweier Demonstratordatensätze angewendet.

5.1.1 Kontaktgraph

Der erste Schritt ist die Erzeugung eines Kontaktgraphen, abgeleitet aus der Baugruppe. Der generelle Ablauf beginnt mit dem Einlesen der neutralen STEP-Datei der Baugruppe. Nachfolgend kann die Topologie der Baugruppe ermittelt und dementsprechend die einzelnen Bauteile oder Unterbaugruppen abgespeichert werden. In diesem Schritt werden STEP- und STL-Files für jeden Bestandteil der hierarchischen Struktur gespeichert, um eine nachfolgende Kontakt- und Bauteilerkennung zu erleichtern. Anschließend erfolgt die Erkennung der Kontakte zwischen den Bauteilen über eine Kollisionserkennung auf Basis von Oberflächennetzen.

Einlesen der STEP-Daten
Das Einlesen der STEP-Daten erfolgt mittels pythonOCC [409, 410], was eine Python Implementierung von OpenCascade [411] ist. OpenCascade stellt einen CAD-Kernel zur Verfügung, wodurch viele Funktionen bereitgestellt werden, wie 3D-Modellierung, Fertigungs- oder Belastungssimulationen. PythonOCC bietet die Möglichkeit, auf dieses umfangreiche C++ Framework zuzugreifen und einen großen Bestandteil der implementierten Funktionen zu nutzen.

Pfaderkennung
Nach dem Import und der Überführung in die OpenCascade Struktur kann die Baugruppe nach ihrer hierarchischen Struktur aufgefächert und gespeichert werden. Dabei wird der Aufbau der Baugruppe einmal komplett durch-

laufen, bis nur noch Bauteile und keine Baugruppen mehr vorhanden sind. Bei jedem Schritt in einer tieferen Ebene wird ein neues Verzeichnis angelegt und die aktuellen STEP- und STL-Dateien für das jeweilige Objekt erzeugt.

Kontakterkennung
Dieser Aufbau an Verzeichnissen und Geometrien ist die Grundlage für die nachfolgende Kontakterkennung zwischen den unterschiedlichen Baugruppen. Für die Aufgabe wird das python package trimesh [412] verwendet, welches für die Handhabung und Bearbeitung von Oberflächennetzen entwickelt wurde. Zunächst werden alle Objekte in der vorher definierten Baugruppen-Ebene gegeneinander auf Kollision überprüft. Dies erfolgt mit den vorher umgewandelten Oberflächennetzen und einem Collision-Checker, welcher einen Kontakt zwischen zwei Objekten erkennt.

Grapherzeugung
Abschließend wird diese Kontaktinformation genutzt, um aus der Topologie der Baugruppe einen Kontaktgraph zu erstellen. Der Vorteil dieses Vorgehens ist, dass es universell auf STEP Baugruppen anwendbar ist und daher keinen zusätzlichen Input aus dem CAD-System (z. B. Montage oder Komponentenplatzierunginformationen) benötigt. Ein Beispiel für eine überführte Baugruppe, abgeleitet von einem Scheibenwischergestänge, ist in Bild 56 zu erkennen. Zusätzlich ist noch die Topologie der zweiten Baugruppenebene dargestellt, welche durch ein Baumdiagramm in der Mitte des Bildes zu erkennen ist.

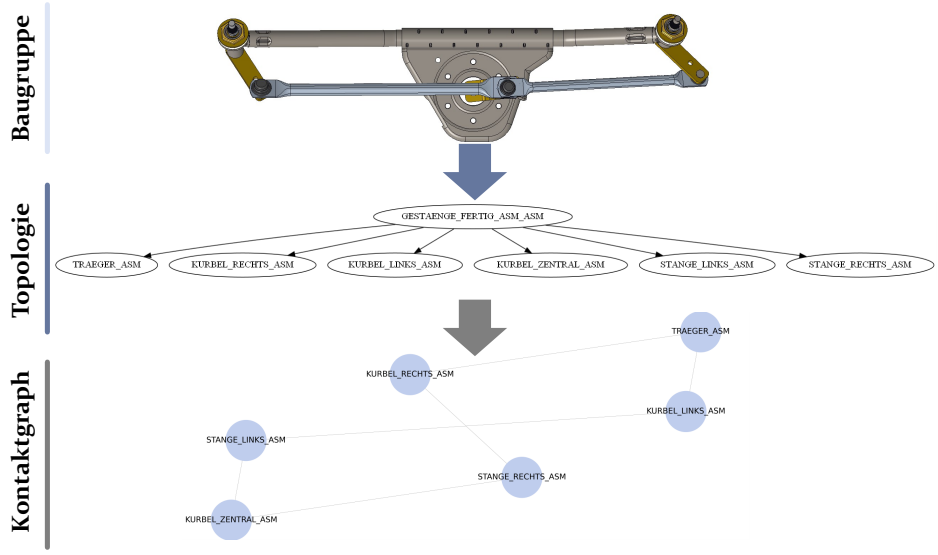


Bild 56: Beispiel für die Umwandlung einer STEP Baugruppe in einen Kontaktgraphen

5.1.2 Bauteilerkennung

Damit ein aussagekräftiger Vergleich zwischen den Prinzipskizzen und Baugruppen erfolgen kann, ist als zusätzliche Information die Bauteilart notwendig. Diese kann dem jeweiligen Knoten als Attribute angefügt werden und somit die Suche mit Skizzen nach Baugruppen potentiell verbessern. Die Ermittlung der Bauteilbezeichnung kann auf unterschiedliche Arten durchgeführt werden, wobei die einfachste Möglichkeit das Auslesen der Bauteilbezeichnung aus dem STEP-file ist. In der Umsetzung muss natürlich eine konsistente Benennung der Bauteile erfolgen und die Zuordnung muss korrekt sein. Daneben besteht noch die Möglichkeit, die vorhandenen Daten selbständig zu labeln, und somit einen durchgängig eingeordneten Datenbestand sicherzustellen. Der initiale Aufwand ist hierfür entsprechend hoch, allerdings gewährleistet es eine sehr genaue und fachlich richtige Zuordnung der Bauteilklassen. Weiterhin können Deep-Learning-Modelle zur Klassifikation von Bauteilen verwendet werden, wobei die Grundlagen der Bauteilerkennung ausführlich in Kapitel 2.2.3.2 beschrieben wurden. In den folgenden Unterkapiteln soll eines neues Vorgehen zur Erkennung von Bauteilen vorgestellt und mit den etablierten Verfahren verglichen werden.

5.1.2.1 Projektionsmethode

Die neue Idee des Erkennungsalgorithmus besteht darin, eine Punktewolke zunächst auf eine Kugel zu projizieren und diese nachfolgend in eine Matrix zu überführen. Der Ursprung der Projektion ist in [231] beschrieben, allerdings ist dieses Vorgehen deutlich überarbeitet worden, da der ursprüngliche Ansatz sich auf Volumennetze spezialisierte und zudem sehr spezifische Bauteilklassen (z. B. Schraube ISO 4762 - M6x20) detektieren konnte und somit nicht für die allgemeine Erkennung von Bauteilklassen (z. B. Schraube, Zahnrad oder Passstift) geeignet ist. Zudem nutzte das vorherige Vorgehensmodell Volumennetze als Input, wogegen die neue Methode Oberflächennetze für die Erkennung von Klassen einsetzt. Ein 3D-Oberflächennetz ist der allgemeine Ausgangspunkt für die Anwendung, wobei auch Punktewolken direkt als Eingabe verwendet werden können, wenn diese ähnlich verteilte Punkte besitzen, um eine vergleichbare Matrix zu erhalten. Die so erzeugten Matrizen werden danach einem CNN übergeben, welches die Einordnung der Klasse durchführt. Das gesamte Vorgehen zur Erkennung von Bauteilen ist in Bild 57 dargestellt und in zwei größere Bereiche untergliedert, die Geometrieaufbereitung und -umwandlung und das Modell-Training. Das Vorgehen wurde in [P7, P8] veröffentlicht und Teilaspekte der Bauteilerkennung wurden zudem in den Arbeiten [S6, S7] betrachtet.

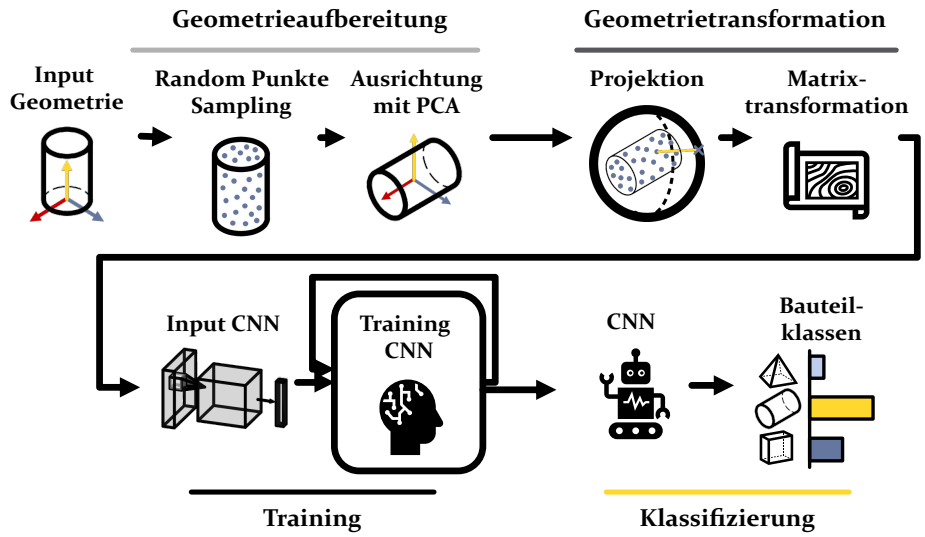


Bild 57: Überblick des Verfahrens zur Bauteilerkennung nach [P7]

Aufbereitung und Transformation

Der erste Schritt besteht darin, die Geometriedatei einzulesen und in eine Punktewolke umzuwandeln, indem eine bestimmte Anzahl an Punkten auf der Oberfläche des jeweiligen Bauteils zufällig erzeugt wird. Die Anzahl an Punkten sollte immer ähnlich groß sein, damit die umgewandelten Matrizen vergleichbar bleiben. Im nächsten Schritt wird die Punktewolke mittels der Hauptkomponenten Analyse (*engl. Principal Component Analysis (PCA)* [413, 414]) ausgerichtet, da die Projektion rotationsabhängig ist und somit die Bauteilorientierung Einfluss auf das Projektionsergebnis hat.

Die so ausgerichtete Punktewolke wird nachfolgend in eine Matrix transformiert. Dazu werden die einzelnen Punkte genutzt und auf eine Detektorkugel projiziert, welche in mehrere Bereiche, auch Pixel genannt, unterteilt ist. Die mathematische Beschreibung des Detektorkugelaufbaus ist in [P2] ausführlich dargelegt. Schließlich werden die projizierten Punkte pro Pixel gezählt und die Kugel wie ein Globus zu einer Karte aufgefaltet. Diese Matrix dient als Eingabe für das CNN und ist primär von zwei Parametern abhängig: der Projektionsart und der Matrix-Auflösung.

Die Auflösung der Projektionsmatrix beschreibt die Anzahl der Pixel auf der Detektorkugel, wobei eine Auflösung von 100 × 100 bedeutet, dass die Kugel 10.000 Pixel hat. Zusätzlich kann auch die Art der Projektion verändert werden. Die beiden entwickelten Möglichkeiten sind in Bild 58 dargestellt. Die erste Art ist die Mittelpunkt-Projektion, in diesem Fall wird vom Mittelpunkt des

Bauteils über die Punkte der Punktewolke eine Gerade aufgespannt, welche dann die Detektorkugel in einem Pixel schneidet. Diese Projektionsart wurde zuerst von Spruegel et al. [231] vorgestellt und unterscheidet sich von der zweiten Art durch die Richtung der Projektion. Der Neuheitsgrad bei der zweiten Projektionsart ist, dass für die Projektion die Normalenrichtung eines jeden Punktes verwendet wird. Die Richtung kann entweder über die Flächen oder anhand der Umgebungspunkte berechnet werden. Die genauen Formeln zur Berechnung beider Projektionsmöglichkeiten ist im Anhang in Formel 19 aufgeführt. Die Matrizen werden anschließend über vorhandene GreyScale-zu-RGB-Algorithmen in RGB-Bilder überführt, um den Anpassungsbedarf der CNN-Modelle möglichst gering zu halten.

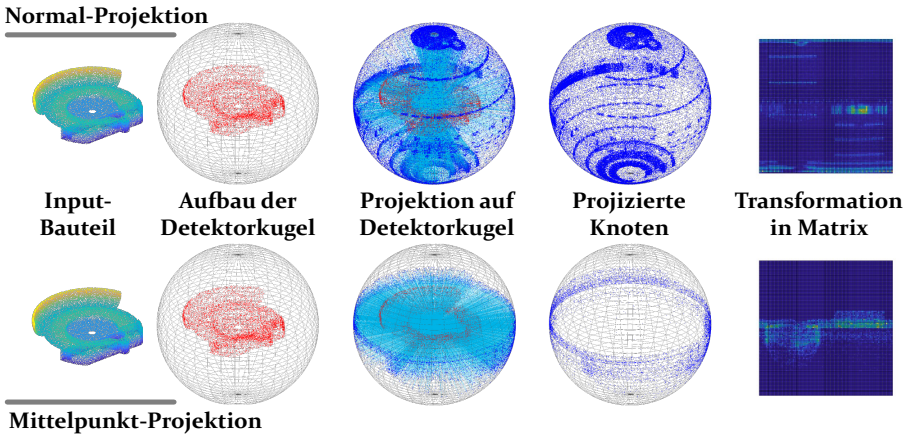


Bild 58: Vergleich der verschiedenen Projektionsarten, welche für die Bauteilerkennung verwendet werden nach [P9]

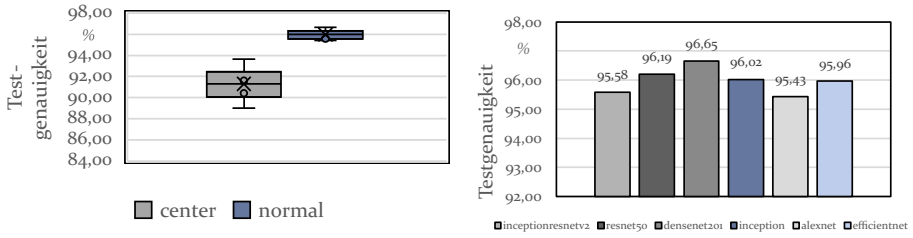
Klassifikationsmodell

Das Vorgehen zur Erstellung der CNN-Modelle orientiert sich an den Standardprozess für das Maschinelle Lernen, indem die Matrizen den jeweiligen CNN-Modellen übergeben werden. Zudem wird für diese Problemstellung Transfer-Learning eingesetzt und somit vortrainierte Netze als Startpunkt herangezogen und nur spezifische Schichten für die jeweilige Aufgabe ausgetauscht. Dadurch wird das Modell nur fein abgestimmt, was in der Regel zu schneller trainierten und stabileren Modellen führt. Die verwendeten Modelle sind aus der Matlab-Umgebung entnommen und alle mit dem ImageNet [144] vortrainiert. Nach der Vorstellung des Vorgehens soll mittels eines Benchmark-Datensatzes der neue Ansatz im Vergleich zu etablierten Modellen getestet werden.

Testen am MCB-Datensatz
Als Bauteildatenbank dient der MCB-Datensatz, welcher bereits in Kapitel 2.2.3.1 vorgestellt wurde. Für diese Studie wird eine feste Auflösung der Matrizen von 256 × 256 definiert, bedingt durch die effiziente Erzeugung und die Standard-Eingangsgröße der verfügbaren Netzarchitekturen von 230-260 Pixeln. Weiterhin ist eine feste Punktewolkenanzahl von 25.000 pro Komponente gewählt worden, da dies einen guten Kompromiss zwischen Laufzeit und Genauigkeit der Projektion darstellt. Insgesamt sind fünf verschiedene CNN-Architekturen zur Erkennung der Bauteile ausgewählt und in [P7] veröffentlicht worden: resnet50 [148], inceptionresnetv2 [415], densenet201 [416], alexnet [145], inceptionv3 [417] und efficientnet [418].

Im ersten Vergleich wird das Vorgehen von Spruegel et al. [231] gegenüber dem neu entwickelten Ansatz getestet. Dazu ist ein Resnet 50 Modell und die Mittelpunkts-Projektion verwendet worden. Die Ergebnisse sind im Anhang in Bild 106 dargestellt und zeigen deutlich, dass das neue Vorgehen eine ca. 25 % höhere Genauigkeit bei der Erkennung der MCB-Geometrien erzielt als der vorherige Ansatz.

Nach diesem erfolgreichen ersten Abgleich sollen die unterschiedlichen Parameter analysiert werden. Dazu ist das Ergebnis für die beiden Projektionsarten in Bild 59a geplottet. Der Boxplot zeigt jeweils die Resultate für alle fünf Modelle, wobei deutlich erkennbar ist, dass die normalen Projektion bessere Erkennungsgenauigkeiten erzielt als die Center-Variante. Im nächsten Schritt sollte daher der Einfluss der unterschiedlichen Modelle untersucht werden, mit den Ergebnissen als Balkendiagramm in Bild 59b illustriert. Alle verwendeten Architekturen haben eine sehr hohe Genauigkeit bei der Untersuchung erzielt mit jeweils über 95 % Testgenauigkeit. Im absoluten Vergleich konnte allerdings das densenet 201 am besten abschneiden mit 96,65 % Genauigkeit.



(a) unterschiedliche Projektionsarten (b) verschiedene Modelle mit normaler Projektion

Bild 59: Ergebnisse der Benchmark-Untersuchung für die Bauteilerkennung nach [P7]

5.1.2.2 Vergleich mit Stand der Technik

Abschließend erfolgt der Abgleich zu erzielten Ergebnissen aus der MCB-Publikation [218], welche sieben State-of-the-Art-Verfahren miteinander vergleicht. Es handelt sich dabei um die Verfahren: PointCNN [419], PointNet++ [225], SpiderCNN [420], MVCNN [228], RotationNet [229], DLAN [223] und VRN [421]. Das Gesamtergebnis inklusive der Einordnung des neuen Vorgehens ist im Anhang in Bild 107 zu finden und in [P7] veröffentlicht. Einzig das RotationNet konnte leicht bessere Ergebnisse als das neu entwickelte Verfahren liefern, allerdings handelt es sich dabei um einen Multi-View-Ansatz, weshalb die Rechenleistungsanforderungen deutlich höher sind. Alle anderen Verfahren erzielten niedrige Genauigkeitswerte, weshalb das neue Projektionsverfahren auch für die Anwendung auf die Prinzipskizzen-Suche genutzt werden soll. Damit die Vergleichbarkeit gewährleistet ist, soll zudem ein PointNet++ Modell trainiert werden, um die Leistungsfähigkeit objektiv nachzuprüfen.

5.1.3 Graphaufbereitung und -matching

Nachdem die Kontakt- und Bauteilinformationen vorliegen, können die Graphen der Skizze und der Baugruppe miteinander verglichen werden, was häufig als Graphmatching bezeichnet wird. Damit dieser Abgleich die am besten geeignete Baugruppe der Skizze zuweist, müssen die Baugruppen-Graphen vorab aufbereitet werden. Dies ist notwendig, da die Baugruppen viel detallierter auskonstruiert sind und somit aus deutlich mehr Bauteilen bestehen als die Prinzipskizzen. Beispielsweise werden in einer Prinzipskizze keine Schrauben oder Dichtringe dargestellt, in dem fertigen CAD-Assembly aber schon. Dieses Phänomen zeigt Bild 60 mit zwei Beispielgraphen für die Skizze und Baugruppe.

Dieser Detail-Unterschied soll durch verschiedene Aufbereitungs- und Matchingalgorithmen überwunden werden, was in den nächsten Kapiteln beschrieben wird.

5.1.3.1 Graphaufbereitung

Die Aufbereitung der Graphen hat zuerst die Aufgabe, die Baugruppen-Graphen zu verkleinern, also unnötige Knoten oder Kanten zu entfernen. Die Idee ist dabei, die komplexeren und verzweigten Baugruppen-Graphen auf die reduzierten Prinzipskizzen-Graphen anzugleichen. Das hat den Vorteil, dass die Laufzeit der Matchingalgorithmen verringert wird, da diese eine hohe Komplexitätsklasse (NP hard oder complete) haben können. Die einzelnen Schritte des Preprocessing sind in Bild 61 aufgeführt, wobei zuerst die isolier-

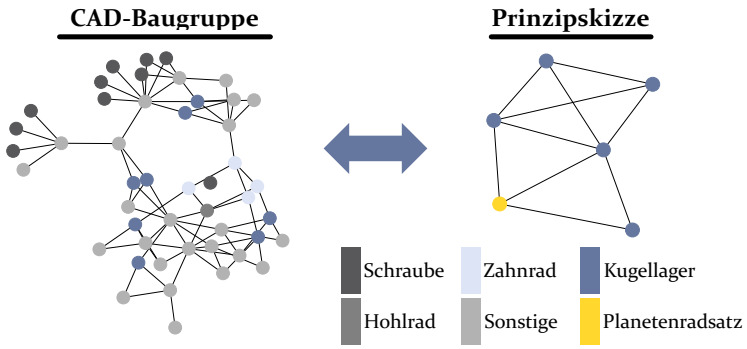


Bild 60: Beispiele für Baugruppen- und Prinzipskizzen-Graphen unterschiedlicher Komplexität

ten Knoten erkannt und aus dem Graphen entfernt werden. Danach können bestehende Knotenmuster zu einem Knoten zusammengefasst werden. Im letzten Schritten werden spezifische Bauteilklassen von dem Graphen ausgeschlossen, abhängig von der Baugruppe und Suchaufgabe. Diese Schritte dienen dazu, das Matching der beiden Graphen aus unterschiedlichen Domänen zu verbessern, indem unnötige oder zusätzliche Informationen aus dem Graph ausgeschlossen werden.

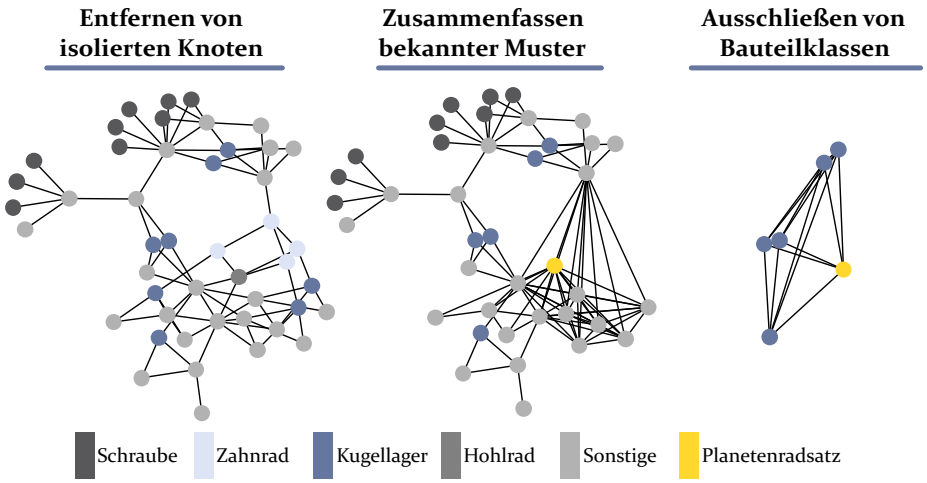


Bild 61: Ablauf des Preprocessings für das Graphmatching

Das Entfernen der isolierten Knoten ist durch eigene Funktionen des python Packages schnell umsetzbar. Eine nachfolgende Art der Vereinfachung ist das Kombinieren von bestimmten Sequenzen zu einem neuen Knoten. Das Vorge-

hen ist regelbasiert und abhängig von den Symbolen und der angewendeten Domäne, weshalb die Regeln für das Knoten-Ersetzen im Vorfeld definiert werden müssen. Das grundlegende Prinzip ist im Anhang bei Quellcode C.1 zu finden. Das Ausschließen von Knoten mit spezifischen Attributen ist der letzte Aufbereitungsschritt und aufwendiger als das Löschen der isolierten Knoten, da durch ein simples Löschen der Knoten die Kontaktinformationen zwischen den Bauteilen verloren gehen würde. Das genaue Vorgehen ist wiederum im Anhang mittels Pseudo Code in Quellcode C.2 beschrieben.

5.1.3.2 Graphmatching

Nach der erfolgreichen Aufbereitung und Vereinfachung der Baugruppen-Graphen können diese für eine Ähnlichkeitssuche verwendet werden. Dazu sollen unterschiedliche Metriken getestet werden, um festzustellen, welche sich am besten für die Verknüpfung von Skizzen und Baugruppen eignen.

Generell lassen sich nach Conte et al. [422] Matching-Algorithmen von Graphen in exakte und inexakte Verfahren unterteilen. Exakte Verfahren sind dabei kantenerhaltend, was bedeutet, dass wenn im ersten Graphen zwei über eine Kanten verbundene Knoten sind, muss dies auch für den zweiten Graphen gelten. Zudem sind weitere Ähnlichkeitsmetriken für Graphen entwickelt worden, welche in Bild 62 dargestellt sind. Nach [422] sind die häufig verwendeten Vergleichsstrukturen, sortiert nach abnehmender Vergleichsgenauigkeit: isomorph, subgraph-isomorph, monomorph und homomorph. Bedingt durch strenge Anforderung der isomorphen Bedingung werden in der Praxis häufiger subgraph-isomorph oder monomorph als Kriterium eingesetzt. Generell sind bekannte Lösungen für die Gleichheitsbedingungen (außer für isomorph) NP-complete Probleme und somit schwierig effizient lösbar. Ein bekannter Algorithmus für das exakte Matching von Graphen ist der Ullmann-Algorithmus [423].

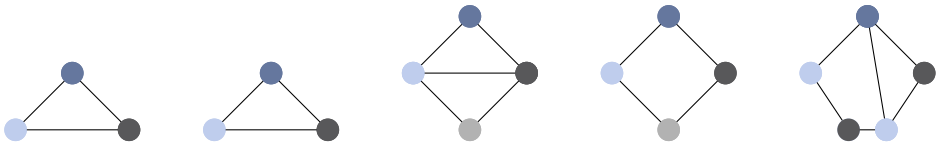


Bild 62: Strukturelle Ähnlichkeiten von Graphen in der Graphentheorie nach [S9, 422]

Die inexakten Vergleichsalgorithmen sind bei nicht einheitlichen oder abweichenden Graphen sinnvoll, da diese weniger streng gegenüber Veränderungen

der Graphenstruktur sind. Die Verfahren sind daher toleranter in der Ähnlichkeitsdefinition und damit auch weniger zeitintensiv als exakte Verfahren. Ein bekannter Vertreter dieser Kategorie nutzt das Prinzip von Kosten für den Vergleich zweier Graphen, die Graph-Edit-Distanz von Bunke [424].

In dem Übersichtsbild 63 sind die vier selektierten Algorithmen aufgelistet. Dabei sind sowohl exakte als auch inexakte Algorithmen ausgewählt worden, welche zudem in der Lage sind die zusätzlichen Attribute der Knoten mit zu berücksichtigen. Allerdings sollte auch ein Verfahren ohne die Verwendung der Knoten-Attribute getestet werden, falls die vorgelagerte Bauteilteilerkennung nicht ausreichende Ergebnisse liefert. Deshalb wurde die Eigenwert-Distanz mit in den Pool an Vergleichsalgorithmen aufgenommen.

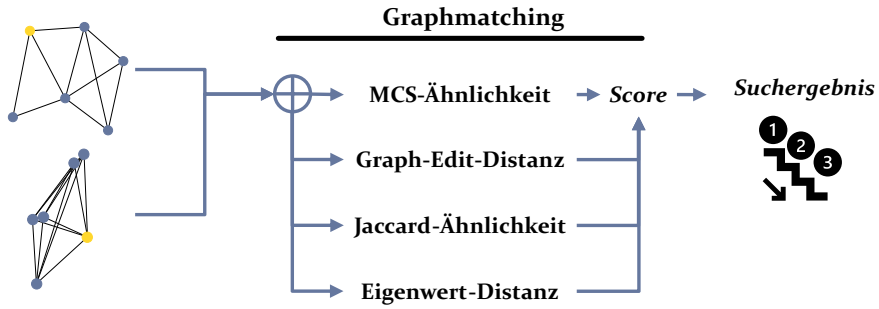


Bild 63: Ablauf des Ähnlichkeitsvergleichs und des Matchings der Graphen

Most-Common-Subgraph (MCS)-Ähnlichkeit
Das Vorgehen zur Bestimmung des MCS-Ähnlichkeits-Scores ist zweigeteilt. Zunächst muss der größte gemeinsame Subgraph zwischen den beiden Suchgraphen ermittelt werden, basierend auf der Subgraph-isomorph Bedingung aus dem vorherigen Kapitel. Für die konkrete Implementierung ist auf den Algorithmus von Houbraken [425] zurückgegriffen worden, da dieser eine kurze Laufzeit verspricht und im networkx-package bereits implementiert ist. Alternativ wäre auch der Einsatz des Ullmann-Algorithmus [423] für diese Aufgabe denkbar. Der Houbraken-Algorithmus verwendet die Symmetrie in Graphen, um den größten Subgraph zu ermitteln und berücksichtigt dabei die Attribute der Graphen mit.

Zur Berechnung des Ähnlichkeitwertes für zwei Graphen wird die Metrik von Bunke und Shearer [426] genutzt, welche in Formel 12 vorgestellt wird. Im Kontext dieser Gleichung wird die Anzahl an Knoten eines Graphes $G = (V, E)$ mit $|G|$ angegeben. Somit definiert sich der Ähnlichkeitswert über die Knotenanzahl des längsten gemeinsamen Graphen durch die Maximalanzahl an Knoten der beiden Graphen.

$$sim_{\text{MCS}}(G_1, G_2) = 1 - \frac{|MCS(G_1, G_2)|}{max(|G_1|, |G_2|)} \tag{12}$$

G_1, G_2	Graph
MCS	Maximum Common Subgraph

Graph-Edit-Distanz (GED)
Das nächste Vorgehen ist ein inexaktes Matching-Verfahren, welches die Kosten für die Annäherung eines Graphen an den Vergleichsgraphen aufsummiert. Das Ziel ist die minimalen Änderungskosten zu ermitteln und anhand derer die Ähnlichkeit der Graphen zu bestimmen. Da die Berechnung der exakten Kosten sehr rechenintensiv (maximal $O(n^3)$) ist, können bei komplexeren Graphen auch Approximationen für die Kosten verwendet werden [427, 428]. Das Ziel des Algorithmus ist, eine error-correcting graph matching (ECGM) Funktion ($m : G_1 \longrightarrow G_2$) zu bestimmen, welche den Eingabegraphen an den Vergleichgraphen anpasst. Die möglichen Operationen und deren Kosten zur Graphanpassung umfassen nach [424, 429]:

Tabelle 3: Beispiel für die Kostendefinitionen der GED nach [429]

Typ	Editiervorgang	G_1	→	G_2	Variable	Kosten
Knoten	Ersetzen	i	→	k	c_{ns}	0
Knoten	Löschen	i	→		c_{nd}	1
Knoten	Einfügen		→	k	c_{ni}	1
Kante	Ersetzen	ij	→	kl	c_{es}	0
Kante	Löschen	ij	→		c_{ed}	1
Kante	Einfügen		→	kl	c_{ei}	1

Zuletzt werden die einzelnen Kosten aufsummiert und ergeben damit die finalen Kosten (vgl. Formel 13). Dieser Wert dient nachfolgend als Vergleich zwischen unterschiedlichen Graphen, wobei ein kleinerer Wert eine höhere Ähnlichkeit der Graphen repräsentiert.

$$c(m) = \sum c_{ns} + \sum c_{nd} + \sum c_{ni} + \sum c_{ed} + \sum c_{ei} + \sum c_{es} \tag{13}$$

Eigenwert-Distanz
Die Eigenwert-Distanz von Koutra et al. [430] vergleicht die Topologien der Graphen untereinander ohne zusätzliche Berücksichtigung der Knoten-Attribute. Dazu werden die Eigenwerte $\lambda_{i,j}$ für die Laplace-Matrizen $L_i =$

$D_i - A_i$ berechnet, welche sich aus D der Diagonalmatrix und A der Adjazenzmatrix der jeweiligen Graphen G_i zusammensetzt. Eine Adjazenzmatrix ist eine andere Darstellungart eines Graphen, indem die Knoten und Kanten in eine Matrix überführt werden. Die Matrix ist nach dem lateinischen Begriff für Nachbarschaft („Adjazenz“) benannt, da diese die Zusammenhänge zwischen allen Knoten darlegt. Danach kann die Eigenwerte-Distanz über die Formel 14 bestimmt werden. Diese besteht aus der Summe der quadrierten Differenzen zwischen den Vergleichsgraphen. Dabei sollen nur die top k Eigenwerte für den Vergleich verwendet werden, weshalb eine Zusatzbedingung definiert wird. Diese stellt sicher, dass nur die Eigenwerte behalten werden, deren Energie $> 0,9$ ist. Im Gegensatz zu den restlichen Metriken berücksichtigt dieses Vorgehen nicht die Attribute der Knoten und ist somit unabhängiger von der Erkennung der Bauteile.

$$dis_{\mathrm{EW}} = \sum_{i=1}^{k} (\lambda_{1\,i} - \lambda_{2\,i})^2 \text{ mit } k = \min_{j=\{1,2\}} \left(\frac{\sum_{i=1}^{k} \lambda_{j\,i}}{\sum_{i=1}^{n} \lambda_{j\,i}} \geq 0,9 \right) \tag{14}$$

λ Eigenwerte der Laplace-Matrix

Jaccard-Ähnlichkeit
Die Jaccard-Ähnlichkeit (oder -Index) [431] ist eine allgemeine Kennzahl aus der Mengenlehre, welche die Überschneidung von Mengen bewertet. Die allgemeine Form ist in Gleichung 15 aufgeführt, welche auch für die Bewertung der Objekterkennungsmodelle genutzt wird (vgl. Formel 5).

$$J(M_A, M_B) = \frac{|M_A \cap M_B|}{|M_A \cup M_B|} \tag{15}$$

M_A, M_B Menge A, B
J Jaccard-Ähnlichkeit

Die Adaption der Ähnlichkeitsmetrik ist somit auf unterschiedliche Arten möglich, wobei die Attribute der Knoten in diesem Fall mit berücksichtigt werden. Neben dem reinen Vergleich der Knotenmenge ist auch ein Abgleich über die Kanten der Graphen möglich. Da Kontaktinformationen der CAD-Baugruppen durch die Kanten repräsentiert werden, sollen diese für den Vergleich mitverwendet werden. Durch die Anfangs- und Endknoten der Kanten können ebenfalls die Attribute der Knoten in den Vergleich aufgenommen werden. Die konkrete Umsetzung zeigt Formel 16.

$$sim_J(G_1, G_2) = \frac{|E_{G_1} \cap E_{G_2}|}{|E_{G_1} \cup E_{G_2}|} \text{ mit } E \subseteq \{\{x, y\} \mid x, y \in V \text{ und } x \neq y\} \tag{16}$$

Nach der Präsentation soll im nächsten Kapitel das gesamte Vorgehen mit zwei Demonstratordatensätzen veranschaulicht und analysiert werden.

Fazit Kapitel 5.1

Ein Vorgehen zur Übertragung von Baugruppen in Graphen ist in diesem Kapitel vorgestellt worden. Dabei sind die besonderen Anforderungen an die Umwandlungsprozesse in jedem Schritt des Vorgehens berücksichtigt worden. So basiert das Vorgehen auf dem universellen STEP-Dateiformat, welches zunächst eingelesen und hierarchisch aufgefächert wird. Danach werden die Bauteile erkannt, entweder über ein Deep-Learning-Modell oder über die Beschreibungen im STEP-file. In Kombination einer Kollisionserkennung und daraus abgeleiteten Kontakt-Informationen kann nachfolgend ein Baugruppen-Graph erzeugt werden. In den letzten Schritten, die Aufbereitung und das Matching der Graphen, muss vor allem der unterschiedliche Detailgrad der Prinzipskizzen- und Baugruppen-Graphen im Ansatz berücksichtigt werden. Dazu sind unterschiedliche Matchingalgorithmen abgeleitet worden, welche in den nächsten beiden Kapiteln ausführlich an Demonstratordatensätzen untersucht werden.

5.2 Anwendung an Demonstratordatensätzen

Nach der theoretischen Vorstellung des Moduls soll in diesem Abschnitt eine Analyse durch zwei Demonstratordatensätze erfolgen. Für jede Demonstrator-Anwendung sind zwei Datensätze erforderlich, einmal die Prinzipskizzen als Sucheingabe und die vorhandenen Baugruppen als Suchergebnis. Bei den Skizzen wird auf den Bestand an Prinzipskizzen zur Symbolerkennung aus Kapitel 4.3.3 zurückgegriffen. Allerdings sind zusätzlich noch fertig konstruierte Baugruppen im STEP-Format erforderlich für die Baugruppen-Datensätze. Dazu müssen neue Datensätze erstellt werden, welche im Detail präsentiert werden.

5.2.1 Auswertemetriken

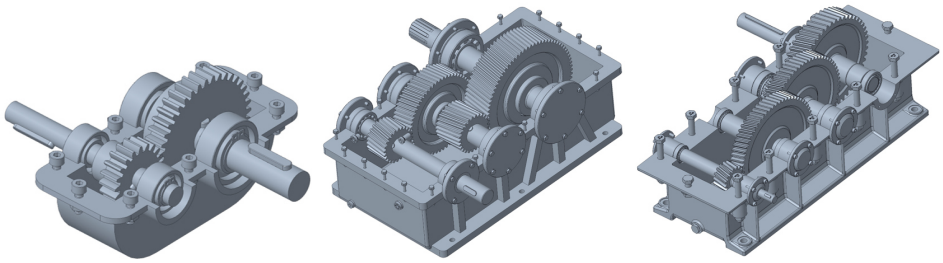
Zur Auswertung der Suchergebnisse sind zwei Informationen relevant: Einerseits die Klasseneinordnung, welche ein richtiges und falsches Retrievalergebnis unterscheidet, andererseits die Metriken zur Beurteilung. Die Einordnung

der Baugruppen erfolgt dabei nach den Getriebestufen. Die Auswertemetriken für Ähnlichkeitssuche können ebenfalls mit einer leichten Adaption für diese Aufgabe genutzt werden. Die Grundlage ist Formel 9, welche den First und Second Tier Wert berechnet. Allerdings ist bei dieser Suchart der Input nicht im Vergleichsdatensatz enthalten, daher werden die Auswertebereiche entsprechend angepasst (siehe Formel 17).

$$\begin{aligned} &\text{für } \mathrm{FT}_{Skizze-Baugruppe} : R = C \\ &\text{für } \mathrm{ST}_{Skizze-Baugruppe} : R = 2 * C \end{aligned} \tag{17}$$

5.2.2 Getriebestufen-Datensatz

Der erste Datensatz an Getriebebaugruppen setzt sich aus 21 verschiedenen Baugruppen zusammen, welche von grabcad.com heruntergeladen wurden. Die gesamte Übersicht der Baugruppe und deren genaue Download-Links sind im Anhang unter Tabelle 19 aufgeführt. Ein kurzer Auszug für die Getriebestufen eins bis drei ist in Bild 64 abgebildet mit einem Vertreter pro Stufe.



(a) 1-stufiges Getriebe (b) 2-stufiges Getriebe (c) 3-stufiges Getriebe

Bild 64: Beispiele aus dem genutzten Datensatz der Getriebebaugruppen

Alle Baugruppen wurden direkt als STEP-file heruntergeladen und danach händisch aufbereitet. Dieser Schritt ist notwendig, da die CAD-Qualität der Baugruppen stark variiert, was bei Firmendaten nicht der Fall wäre. Bei manchen Baugruppen fehlte der Kontakt zwischen Bauteilen, die allgemeine Zuordnung der Bauteile war nicht vorhanden, leere Bauteile wurden verwendet oder auch die Topologie der Baugruppe war schlecht strukturiert. Alle diese Fehler mussten händisch ausgebessert werden, damit die automatische Kontakterkennung für die Assembly-Modelle funktioniert. Jedoch war die Bezeichnung der Bauteile und Unterbaugruppen in den heruntergeladenen STEP-Daten relativ konsistent, weshalb die erste Untersuchung des Vorgehens mit einer Bauteilerkennung über die vergebenen Namen abläuft.

5.2.2.1 Ergebnisse

Für die Bewertung der Retrievalergebnisse dienen die Prinzipskizzen als Suchinput, welche ausführlich in Kapitel 4.3.2.1 beschrieben sind. Insgesamt sind 30 Zeichnungen für den Vergleich herangezogen werden, welche von einer bis fünf Getriebestufen variieren.

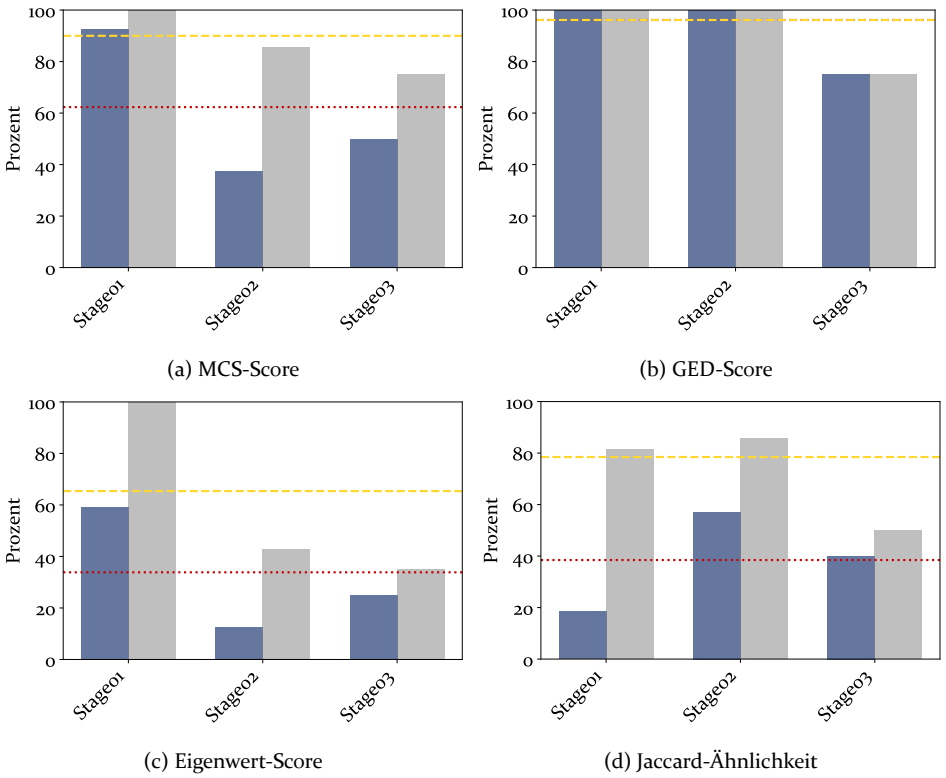
Das Testen des entwickelten Vorgehens ist zweigeteilt. Zunächst wird eine ideale Bauteilerkennung angenommen, welche auf den Bezeichnungen der Bauteile und Unterbaugruppen aus dem STEP-file basiert. Dies soll zuerst überprüfen, ob eine Verknüpfung von Skizzen und Baugruppen möglich ist. Danach wird die Situation von unbekannten Bauteil-Kategorien getestet, weshalb ein trainiertes Modell zum Klassifizieren der Bauteile genutzt wird.

5.2.2.2 Ideale Bauteilerkennung

Die erreichten Ergebniswerte für die vier vorgestellten Vergleichsmetriken sind mittels Balkendiagrammen in Bild 65 dargestellt. Dabei ist der First-(dunkelbau) und Second-Tier (hellgrau)-Score für drei Getriebestufen ausgewertet und geplottet. Daher werden auch nur Skizzen für diese Auswertung berücksichtigt, die maximal drei Getriebestufen beinhalten. Der Mittelwert ist als gestrichelte Linie für die beiden Tier-Metriken zusätzlich eingefügt (Rot für First und Gelb für Second Tier). Für diesen spezifischen Vergleich ist vor allem der First-Tier-Wert relevant, da aufgrund der ungleichen Klassenverteilung die Aussagekraft des Second-Tier-Wertes für die ersten beiden Getriebestufen deutlich reduziert ist.

Für die Durchführung sind die vorher präsentierten Matchingalgorithmen verwendet worden. Die Kosten bei der GED sind allerdings noch angepasst worden, da bekannt ist, dass die CAD-Baugruppen mehr Informationen und daher auch mehr Kontakte als eine Prinzipskizze aufweisen. Dieses Wissen soll bei der Suche berücksichtigt werden, weshalb das Einfügen von Kanten ohne Kosten erfolgt. Zudem sind die Knoten-Attribute für den Vergleich relevant, daher wurden die Kosten für das Ersetzen von Knoten auf 1 angehoben und somit die Änderungskosten an die übrigen Knotenoperationen angeglichen wurden. Daher sind im Vergleich zur Tabelle 3 folgende Änderungen umgesetzt: $c_{ns} = 1 | c_{ei} = 0$

Die vier Diagramme zeigen deutlich, dass die Erkennungsgenauigkeit stark von der Vergleichsmethode abhängt. So erzielen der MCS- (Bild 65a) und GED-Ansatz (Bild 65b) sehr gute Ergebnisse mit jeweils FT-Werten über 80 %. Dagegen zeigt der Eigenwert-Score (Bild 65c) schlechte Suchergebnisse mit bis zu 10 % Erkennungsgenauigkeit bei der Getrieben der zweiten Stufe. Die Jaccard-Metrik (Bild 65d) positioniert sich zwischen dem MCS- und



(a) MCS-Score

(b) GED-Score

(c) Eigenwert-Score

(d) Jaccard-Ähnlichkeit

Bild 65: Beispiele aus dem genutzten Datensatz der Getriebebaugruppen

Eigenwert-Score, wobei vor allem die FT-Ergebnisse der zweiten Stufe positiv hervorzuheben sind. Mit Abstand die besten Suchergebnisse liefert die GED-Methode mit den angepassten Gewichten. Dabei konnten alle Skizzen der ersten und zweiten Stufen korrekt den jeweiligen Baugruppen zugeordnet werden. Einzig bei der dritten Getriebestufe ist eine Baugruppe nicht korrekt gefunden worden. Dies ist allerdings auf die CAD-Datei (Nummer 12-3-7 vgl. Tabelle 19) zurückzuführen, da in dieser Baugruppe alle Lager fehlen und somit nur die Zahnräder, Wellen und Gehäuse vorhanden sind.

Zusätzlich zu den Balkendiagrammen sind für alle Vergleichsmetriken noch die genauen Scores mit ihrer jeweiligen Wahrheitsmatrix für alle Suchanfragen im Anhang unter Kapitel C.4.0.1 aufgelistet. Interessant ist dabei noch die Auswertung der Ergebnisse für die GED-Metrik in Bild 105. Bei der Auswertung ist deutlich erkennbar, dass selbst die Skizzen mit Getriebestufe vier und fünf als Ergebnis die Baugruppen der dritten Stufen erhalten haben. Diese Fähigkeit ist nicht in der vorherigen Auswertung enthalten, da dort nur die ersten drei Stufen berücksichtigt wurden. Zusammenfassend hat

diese Untersuchung gezeigt, dass die Metriken einen großen Einfluss auf die Erkennungsgenauigkeit haben. Die erzielten Ergebnisse variieren stark, wobei die besten Ergebnisse die angepasste Version der Graph-Edit-Distanz erreicht hat.

5.2.2.3 Deep-Learning-basierte Bauteilerkennung

Der Nachweis über die allgemeine Funktionserfüllung der Baugruppensuche mit Prinzipskizzen konnte im vorherigen Kapitel erbracht werden, allerdings mit der Einschränkung, dass die Bauteilklassen aus dem Namen der Baugruppe ermittelt wurden. Für eine noch stärker automatisierte Suche ist es notwendig, dass die Bauteile über trainierte Modelle erkannt werden und somit kein Input oder keine Umbenennung der Teile notwendig ist. Ein grundsätzliches Problem bei der Erkennung ist, dass unbekannte Bauteilklassen, welche nicht im Trainingsdatensatz vorhanden sind, Bestandteile der unterschiedlichen Baugruppen sein können. Für diese Herausforderung ist ein neuer Ansatz entwickelt worden, welcher nachfolgend kurz präsentiert wird. Danach sollen verschiedene Algorithmen zur Erkennung der Bauteile untersucht und final das gesamte Vorgehen der Baugruppensuche automatisiert durchlaufen werden. Als Trainingsdatensatz dient dabei wiederum der MCB-Benchmarkdatensatz, welcher einen vorhandenen Datenbestand eines Unternehmens realistisch nachbildet. Daher können nur die Bauteilklassen aus diesem Datensatz in den vorhandenen Baugruppen bestimmt werden. Zusätzlich wird die Erkennung der Bauteile erschwert, da alle Bauteile der Baugruppen komplett unbekannt für die trainierten Modelle sind, wodurch vor allem die Übertragbarkeit auf unbekannte Bauteile getestet wird. Im Gegensatz zum klassischen Hold-Out [132] Split handelt es sich hierbei um ein out-of-sample Problem [432, 433], was laut [434] eine sehr relevante Metrik für die Bewertung von trainierten Modellen darstellt.

Vorgehen Bauteilerkennung
Bei der realen Anwendung trainierter Modelle sind viele unvorhergesehene Herausforderungen vorhanden, vor allem bestimmt durch die Unberechenbarkeit der unbekannten Eingabedaten. Deshalb wurde von Geng et al. [435] eine Einordnung unterschiedlicher Anwendungsprinzipien unternommen, welche die Bekanntheit der Trainings- und Testdatensets (von Known-Known bis Unknown-Unknown) und das generelle Ziel des Vorhabens berücksichtigt.

Nach dieser Einordnung ist die einfachste Lösung eine normale Klassifikation mit einer Reject-Option, abhängig vom Confidence-Schwellwert. Sollte also der ermittelte Wert des Modells darunter liegen, wird das Bauteil als unbekannt oder nicht-relevant eingestuft, liegt er darüber, wird dem Bauteil das

entsprechende Label zugewiesen. Das Vorgehen ist für unterschiedliche Klassifizierungsalgorithmen [436] und Anwendungsfälle weiterentwickelt worden [437], hat aber das Problem, dass die Ergebnisgüte stark von dem gewählten oder trainierten Schwellwert abhängig ist. Ein allgemeiner Ansatz dafür ist die One-Class-Classification [438–440] oder auch Ausreißer-Erkennung. Diese wird angewendet, wenn nur positive Ereignisse vorhanden sind oder die negativen Beispiele keine einheitlichen Strukturen aufweisen. Das Ziel dabei ist, durch das trainierte Modell die positiven Beispiele deutlich von den negativen zu trennen.

Dieses Grundprinzip ist auch in dem neu entwickelten Ansatz übertragen worden, indem ein zweistufiges Vorgehen definiert wurde, welches in Bild 66 illustriert ist. Zuerst soll ein Modell die relevanten von den nicht-relevanten Geometrien unterscheiden und nachfolgend den relevanten Bauteilen eine genaue Klasse zuweisen. Für den Anwendungsfall dienen als Trainingsdaten jeweils der MCB-Datensatz, mit dem Ziel die drei, für die Funktion entscheidenden Bauteilklassen Kegelrad, Zahnrad und Lager, zu erkennen. Somit wird der MCB-Datensatz für das Relevanz-Unterscheidungsmodell in zwei Klassen unterteilt. Die relevanten Bauteile setzen sich aus den MCB-Klassen „Radial contact ball bearings“, „Right angular gearings“ und „Right spur gears“ zusammen. Für die nicht-relevanten Bauteile werden zufällig aus allen anderen bestehenden Klassen Bauteile ausgewählt, wobei darauf geachtet wurde, dass die Anzahl an Sample pro Klasse nicht zu weit auseinander liegt (ca. 1700 zu 2600). Wenn das trainierte Modell entscheidet, dass es sich um ein relevantes Bauteil handelt, wird diese Geometrie dem Modell zur Bestimmung der spezifischen Bauteilklasse übergeben. Dabei dienen die drei vorherigen Klassen wiederum als Trainingsdaten, allerdings diesmal als Multi-Class-Klassifikationsproblem, weshalb alle Klassen eigenständig sind.

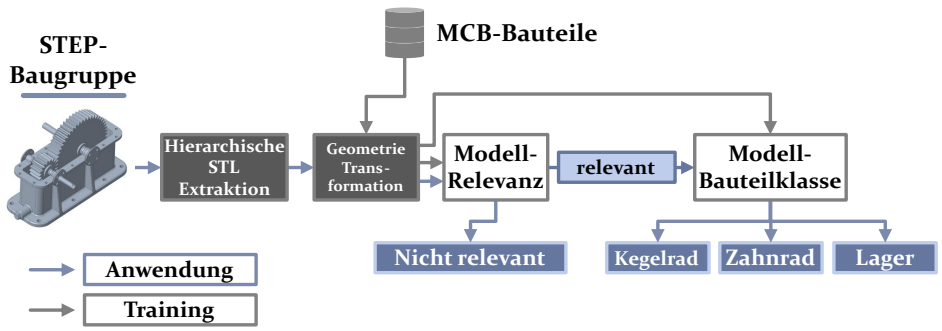


Bild 66: Ablauf Modell für die Bauteilerkennung am Getriebestufen-Datensatz

Nachdem beide Erkennungsmodelle trainiert wurden, kann eine STEP-Baugruppe an diese übergeben werden. Dazu muss zunächst die hierarchische Struktur der Baugruppe durchlaufen werden und die relevanten Bauteile als Oberflächennetz (z. B. STL) abgespeichert werden. Diese müssen dann für den jeweiligen Erkennungsalgorithmus aufbereitet und den beiden trainierten Modellen übermittelt werden. Nach der Einordnung liefert das Vorgehen als Ergebnis eine der vier möglichen Klassen, welche im Bild dunkelblau hervorgehoben sind.

Erkennungsergebnisse
Das Testen des entwickelten Vorgehens erfolgt mit den out-of-sample Daten aus dem Getriebestufen-Datensatz, welche anhand der Bauteilbezeichnung vorab in die entsprechenden Klassen eingeordnet wurden. Als Metrik wird die Genauigkeit nach Formel 6 aus Kapitel 2.2.3.2 verwendet. Dabei soll das neue Projektionsverfahren mit einem etablierten Bauteilerkennungsalgorithmus verglichen werden, was in diesem Fall das PointNet++ darstellt, welches ausführlich im Anhang in Kapitel C.2 erläutert wird.

Die Ergebnisse der Untersuchung listet die Tabelle 4 auf, indem diese das Vorgehen, Modelle und die Parameter für beide Datensätze aufführt. Beim Projektionsansatz wird zwischen der vortrainierten Deep-Learning-Architekturen (DenseNet und InceptionResNetv2) und der Projektionsart (Center und Normal) unterschieden. Zudem wurde noch die Möglichkeit ergänzt, beide Ergebnisse der Projektionsarten zu vereinen und somit das Ergebnis von zwei trainierten CNNs zu einem Gesamtergebnis zu kombinieren, ähnlich dem Ensemble-Prinzip aus dem Machine-Learning. Als Auflösung wurde eine 128×128 Matrix verwendet, da dies zu schnellen Laufzeiten für die Anwendung führt. Das PointNet-Modell dagegen wurde mit unterschiedlichen Auflösungen (2048 und 8192 Punkten) der Punktewolke untersucht, was den beiden Parametern Low- und HighRes(olution) aus der Tabelle entspricht.

Das beste Ergebnis wurde für jeden Datensatz in der Tabelle fett und unterstrichen hervorgehoben. Dabei fällt auf, dass das Projektionsverfahren sehr gute Ergebnisse an den unbekannten Geometrien erreichen kann mit über 90 % Genauigkeit bei beiden Datensätzen. Die Erkenntnisse aus der Untersuchung des MCB-Datensatzes lassen sich teilweise übertragen, da für die Entscheidung, ob ein Bauteil relevant für die Suche ist, die normalen Projektion in Verbindung mit dem densenet eine sehr hohe Genauigkeit erzielen konnte. Dagegen hat das PointNet++ in beiden Fällen weniger hohe Ergebnismetriken bewerkstelligen können, was auf das fehlende Transfer-Learning zurückzuführen ist. Dadurch wurde das Modell sehr genau auf die vorhandenen Bauteile optimiert, war aber nicht mehr in der Lage die Erkennung

auf die unbekannten und abweichenden Bauteile des Test-Datensatzes zu übertragen. Auch die Verknüpfung von Ergebnissen der beiden Projektion zeigte sehr positive Ergebnisse, vor allem für den ersten Datensatz. Durch die Feature-Kombination konnte die Erkennungsgenauigkeit nochmals um fast 2 % auf 93,5 % gesteigert werden. Nach dieser erfolgreichen Anwendung der Bauteilerkennung soll nachfolgend das in Bild 66 vorgestellte Vorgehen in die Erkennung integriert und die Auswirkung der automatisierten Erkennung untersucht werden.

Tabelle 4: Out-of-Sample Genauigkeit der beiden trainierten Modelle

Erkennung Bauteil-Relevanz			
Vorgehen	Modell	Parameter	Out-of-Sample Genauigkeit
Projektion	DenseNet	Center	0,849
Projektion	DenseNet	Normal	0,917
Projektion	DenseNet	Features-Kombiniert	**0,931**
Projektion	InceptionResNetV2	Center	0,836
Projektion	InceptionResNetV2	Normal	0,871
Projektion	InceptionResNetV2	Features-Kombiniert	0,901
PointNet++	Zwei Abstraktion Layer	LowRes	0,635
PointNet++	Zwei Abstraktion Layer	HighRes	0,754
Erkennung spezifische Bauteilklasse			
Projektion	DenseNet	Center	**0,901**
Projektion	DenseNet	Normal	0,547
Projektion	DenseNet	Features-Kombiniert	0,863
Projektion	InceptionResNetV2	Center	0,870
Projektion	InceptionResNetV2	Normal	0,588
Projektion	InceptionResNetV2	Features-Kombiniert	0,740
PointNet++	Zwei Abstraktion Layer	LowRes	0,504
PointNet++	Zwei Abstraktion Layer	HighRes	0,473

Retrievalergebnisse Sketch-Assembly

Die Baugruppen, Skizzen und Auswertemetriken sind alle der vorherigen Untersuchung aus Kapitel 5.2.2.2 entnommen, wobei in diesem Fall nur die GED als Vergleichsalgorithmus verwendet wird, da diese in der vorherige Untersuchung am besten abgeschnitten hat. Das Suchergebnis ist in Bild 67 mittels eines Balkendiagramms und den Metriken First- und Second-Tier abgebildet.

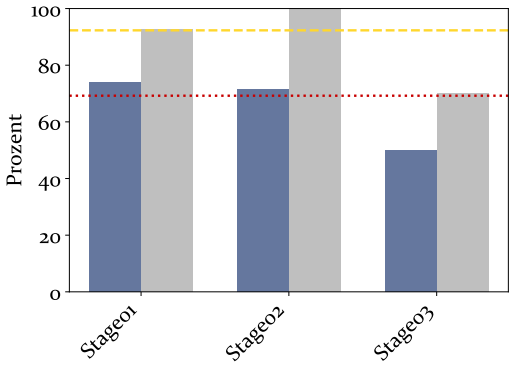
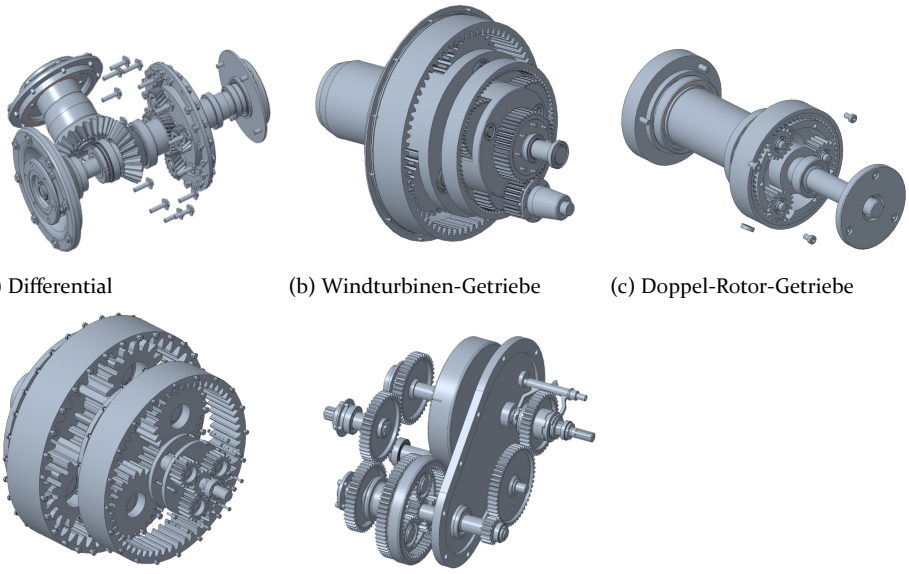


Bild 67: Ergebnis der Baugruppensuche mit Skizzen als Input und Deep-Learning Bauteilerkennung

Dabei zeigen die Metriken für alle drei Getriebestufen eine leicht schlechtere Retrievalgüte im Vergleich zur idealen Bauteilerkennung. Somit beeinflusst die Bauteilerkennung deutlich die Ergebnisse, da Fehler in beiden Erkennungsmodellen auftreten können und sich am Ende daher aufsummieren. Allerdings sind die First-Tier Werte immer noch sehr hoch, da die Ergebnisse für die ersten beiden Stufen über 70 % liegen und demnach fast alle Baugruppen in den ersten Suchergebnissen platziert sind. Alle erzielten GED-Scores für die Vergleichskombinationen sind wiederum im Anhang in Bild 108 aufgeführt. Bei der genauen Auswertung wird deutlich, welche Bauteile in den jeweiligen Baugruppen nicht perfekt erkannt wurden (z. B. Stage 01-3, Stage 01-24 oder Stage 02-21), da dort höhere GED-Kosten als in der vorherigen Untersuchung anfallen. Zusammenfassend konnte demonstriert werden, dass eine automatisierte Übertragung unbekannter STEP-Baugruppen in einen annotierten Graphen durchführbar ist und somit ein Abgleich mit ebenfalls transformierten Prinzipskizzen ermöglicht wird.

5.2.3 Baugruppen-Datensatz

Nach dem vielversprechenden Testen des Vorgehens an den ähnlichen Getrieben, welche sich primär durch die Stufen unterscheiden, sollen in diesem Kapitel aufwendigere Baugruppen verwendet werden. Die Baugruppen sind dem Datensatz „3D Assembly Repository“ [276] entnommen. Für die Untersuchung werden folgende Baugruppen-Kategorien genutzt: Differential, Windturbinen-Getriebe, Doppel-Rotor-Turbine, hydraulisches Reduktionsgetriebe und hydraulischer Rotor. Jeweils ein Beispielbild der verschiedenen Baugruppenarten ist in Bild 68 dargestellt.



(a) Differential (b) Windturbinen-Getriebe (c) Doppel-Rotor-Getriebe

(d) Reduktionsgetriebe (e) Hydraulik-Rotor

Bild 68: Beispiele aus dem genutzten Datensatz der Getriebebaugruppen nach [276]

5.2.3.1 Baugruppenaufbereitung

Die Baugruppen unterscheiden sich durch ihre Komplexität, Größe, Bauteilanzahl, Funktion und ihrem Detailgrad. So sind teilweise die Baugruppen vollständig fertig konstruiert, inklusive Details wie Sicherungsringe, Unterlegscheiben oder Nutmuttern, gleichzeitig sind Baugruppen mit nur angedeuteten Lagern und Zahnrädern ebenso vorhanden. Daher erfolgt für diese Untersuchung die Bauteilerkennung rein über den Bauteilnamen aus der Baugruppe, da eine Klassifikation von vereinfachten Maschinenelementen nicht möglich ist und auch Bauteile erkannt werden müssten, die nicht im MCB-Datensatz vorhanden sind (z. B. Hohlrad).

Zusätzlich wurden pro Baugruppenkategorie nur gewisse Baugruppen selektiert und für die Suche herangezogen, da einige Baugruppen unvollständig oder fehlerhaft waren. Die genaue Auswahl und Übersicht aller Baugruppen sind im Anhang in Tabelle 20 abgebildet. Da in den STEP-Baugruppen teilweise Wellen oder Zahnräder falsch positioniert sind, zu stark vereinfacht wurden oder nicht im Kontakt stehen, ist ein Set an Regeln erzeugt worden, welches diese Beziehungen mit in den Graphen integriert. Alle verwendeten Regeln für einen Kontaktaufbau zwischen zwei Bauteilen sind im Anhang unter Kapitel C.9 aufgelistet. Weiterhin sind in den Baugruppen auch Planetengetriebe vorhanden, welche sich aus verschiedenen Kombinationen aus

Hohl-, Sonnenzahnrädern und Lagern zusammensetzen. Da in den Zeichnungen aber auch eigene Symbole für Planetengetriebe verwendet werden, sollen in den Baugruppen diese ebenfalls zu einem Knoten verknüpft werden. In Bild 109 im Anhang sind die dafür angewendeten Regeln grafisch aufbereitet, welche die unterschiedlichen Kombinationen der Knotenattribute in einen Knoten überführen.

5.2.3.2 Skizzenaufbereitung

Als Input-Skizzen dienen wieder die vorher erkannten Zeichnungen aus Kapitel 4.3.2.2 und das dort vorgestellte YOLO-Modell. Diese durchlaufen den präsentierten Prozess zur Graph-Umwandlung und dienen danach als Suchinput für die vorhandenen Baugruppen. Die neuen Skizzen weisen allerdings eine Besonderheit im Vergleich zu den vorherigen Getriebestufen auf, da in diesen Hohlwellen zum Einsatz kommen. Diese neuartigen Symbole werden automatisch als eigene Klasse erkannt, allerdings ist für die Transformation in Graphen die Zusatzinformation relevant, da der mittlere Bereich der erkannten Bounding Box nicht weiß ausgefüllt wird. Dies stellt sicher, dass die innere Welle nicht unterbrochen wird und somit die Zuordnung der Symbole in dem Graphen dem Aufbau der Baugruppe entspricht. Der Vorgang mit den jeweiligen Ergebnisse ist zur besseren Veranschaulichung in Bild 69 dargestellt. Dabei werden die Bounding Boxen der Klassen mit Hohlwelle automatisch aufgeteilt, damit die mittige Welle nicht geschnitten wird.

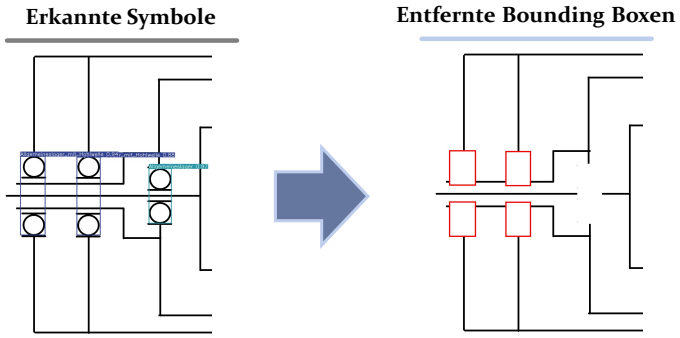
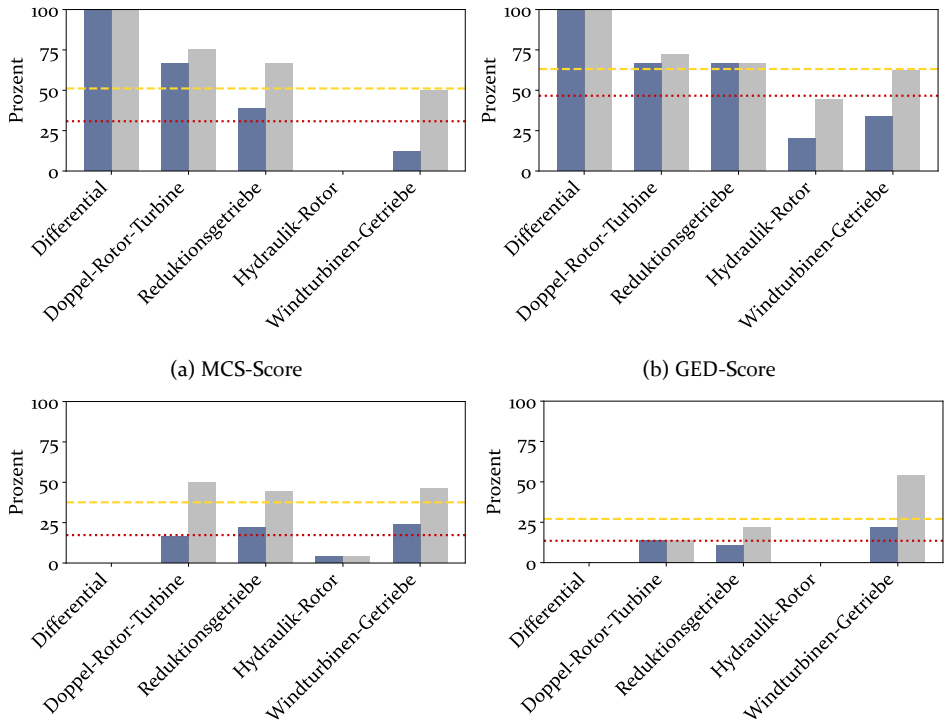


Bild 69: Ergebnis der angepassten Entfernung der Bounding Boxen

5.2.3.3 Ergebnisse

Für die Untersuchung des großen Baugruppen-Datensatzes sind wiederum die Matchingalgorithmen aus der vorherigen Untersuchung herangezogen worden. Für die beiden Verfahren MCS und GED ist eine Laufzeitbeschränkung pro Suchvorgang von jeweils 30 Sekunden festgelegt worden. Die Kosten

bei der GED sind von der ersten Untersuchung übernommen worden, mit einer Ergänzung für den Fall, dass ein Knoten mit dem Attribut „Planetengetriebe“ angepasst werden muss. Für diesen Fall wurden die Kosten auf 10 erhöht, da dieser mehrere Bauteile zusammenfasst und somit höher gewichtet wird. Die Retrievalergebnisse sind in Bild 70 für alle vier Matchingverfahren als Balkendiagramm geplottet. Im Anhang unter Kapitel C.10 sind alle Suchergebnisse inklusive der jeweiligen Ähnlichkeitsscores aufgeführt. Bedingt durch die ähnliche Baugruppen-Anzahl pro Kategorie sind in dieser Auswertung auch beide Metriken (FT und ST) für die Beurteilung der Ergebnisse relevant.



(a) MCS-Score

(b) GED-Score

(c) Eigenwert-Score

(d) Jaccard-Ähnlichkeit

Bild 70: Retrievalergebnisse für den Baugruppen-Datensatz [276]

Die Ergebnissen zeigen dabei Parallelen zur vorherigen Analyse der Getriebestufen-Baugruppen. Der GED-Ansatz erzielt wiederum die besten Retrievalergebnisse, gefolgt vom MCS-Verfahren. Allerdings ist diesmal der Unterschied zwischen MCS- und Eigenwert-Score deutlich geringer als bei der vorangegangen Auswertung. Die Jaccard-Ähnlichkeit hat die schwächsten Ergebnisse geliefert, mit einem ST-Wert von nur knapp über 25 %.

Generell ist festzustellen, dass die Erkennungsgenauigkeit von Baugruppe zu Baugruppe teilweise stark pro Matchingverfahren schwankt. Das liegt zum einen an der Anzahl an Input-Skizzen, beispielsweise ist nur eine Skizze für das Differenzial vorhanden. Zum anderen an den Baugruppen selbst, diese haben teilweise einen ähnlichen Aufbau, obwohl sie in verschiedenen Kategorien eingeordnet sind, etwa die Reduktionsgetriebe- und Windturbinen-Getriebe-Modelle. Beide Baugruppen nutzen ein zweistufiges Planentengetriebe zur Umwandlung der Antriebskräfte. Eine weitere Herausforderung ist die Hydraulik-Rotor-Klasse, da dort die verschiedenen Baugruppen sehr heterogen aufgebaut und somit schwierig zuzuordnen sind.

Einen zusätzlichen Faktor stellen die erkannten Skizzen dar. Hierbei wiesen wieder die Hydraulik-Rotor-Skizzen die meisten Fehler bei der Symboldetektion auf. Das Hauptproblem waren zu große Abstände zwischen den Hohlwellen, wodurch beispielsweise ein Hohlwellen-Lager als zwei einzelne Lager erkannt wurde, dieser Fehler trat z. B. in Skizze HydraulicRotor-4 und -5 auf. Ebenfalls kann bei der Umwandlung der Skizzen in Graphen ein Fehler auftreten, wenn die Bounding Box der Symbole durch das Deep-Learning-Modell zu groß gesetzt wurde. Dadurch können Verbindungslinien geschnitten und eigentlich in Kontakt stehende Symbole nicht miteinander verknüpft werden. Dieser Fehlertyp konnte bei den Skizzen HydraulikReduction-2 und WindTurbine-3 jeweils für das Planetengetriebe-Symbol festgestellt werden.

Fazit Kapitel 5.2

Das entwickelte Vorgehen zur Suche von Baugruppen mittels Skizzen wurde in diesem Abschnitt anhand von zwei Datensätzen ausgiebig getestet. Dazu mussten zunächst die Datensätze erzeugt und aufbereitet werden. Anschließend konnten mit diesen die verschiedenen Matchingverfahren getestet und analysiert werden. Für den Vergleich zwischen Skizzen und Baugruppen stellte sich die Graph-Edit-Distanz für den Getriebestufen- und Baugruppen-Datensatz als bester Ansatz heraus. Den ersten, einfacheren Datensatz konnten auch andere Matching-Verfahren noch ausreichend zuordnen, bei dem deutlich aufwendigeren zweiten Datensatz lieferte allerdings nur die GED ein zufriedenstellendes Ergebnis. Zudem wurde noch die Deep-Learning-basierte Erkennung der Bauteile untersucht, mit einem automatisierten Vorgehen als Ergebnis. Die Gesamt-Erkennungsleistung nimmt zwar durch den Einsatz etwas ab, allerdings ist keine händische Aufbereitung der Bauteile mehr notwendig.

5.3 Diskussion

In dem vorherigen Kapitel ist ein Vorgehen zum Matching von Skizzen zu Graphen vorgestellt und erfolgreich an zwei Datensätzen angewendet worden. Dabei steht die automatisierte Umwandlung von Baugruppen zu Graphen im Vordergrund. Solange korrekte und richtig konstruierte Baugruppen verwendet werden, benötigt dieser Prozessschritt vor allem die domänenspezifischen Regeln für die Reduktion von Graph-Knoten und den gelabelten Datensatz für die Bauteilerkennung als zusätzlichen Input. Falls die Baugruppe nicht immer korrekt aufgebaut wurde, muss diese noch händisch aufbereitet werden, was vor allem die Kontakte, Bauteilnamen und die Anordnungs-Ebene der Bauteile betrifft. Alle weiteren Operationen laufen automatisch ohne weitere Eingaben ab.

Diese Bedingung trifft auch auf die Matchingalgorithmen der Graphen zu, einzig die Hyperparameter für die GED können angepasst werden, falls speziell zusammengesetzte Knoten zum Einsatz kommen. In Kombination mit dem ersten Datensatz zeigt vor allem die GED eine sehr gute Performanz, indem alle korrekten Baugruppen den jeweiligen Skizzen zugeordnet werden konnten. Der Einsatz von Deep-Learning-Modellen zur Erkennung senkte zwar die Retrieval-Genauigkeit, allerdings nur auf ca. 70 %, was immer noch einem guten Ergebnis entspricht. Somit ist die zweite Forschungsfrage positiv mit dem präsentierten Vorgehen beantwortet, welches die Umwandlung und Matching von Prinzipskizzen zu Baugruppen mit einer hohen Retrieval-Genauigkeit ermöglicht.

Abschließend hat die zweite Untersuchung des entwickelten Vorgehens ebenfalls die vorherigen Erkenntnisse bestätigt, indem diese zeigte, dass der Ansatz in der Lage ist, hohe Genauigkeiten für die Suche mit einer Skizze als Input zu erreichen. Im zweiten, aufwendigeren Datensatz konnte ein ST von knapp unter 70 % über alle Suchanfragen erreicht werden, trotz der im vorherigen Kapitel beschriebenen Herausforderungen für die Erkennung und die Matchingalgorithmen. Dabei legte die Auswertung dar, dass sich die Fehler aus der Skizzenerkennung ebenfalls in den Suchergebnissen niederschlagen, da falsche verknüpfte oder fehlende Symbole zusätzliche Probleme darstellen.

Generell ist bei der Auswertung anzumerken, dass die Einordnung der Baugruppenklassen auch einen Einfluss auf das Ergebnis hat, da teilweise ähnliche Getriebevarianten in verschiedene Kategorien eingeordnet wurden (Wind-Turbine und Reduktionsgetriebe). Ein neues Zuweisen der Klassen, beispielsweise nach der primären Funktion, könnte hier einen positiven Einfluss auf die Erkennungsgenauigkeit liefern.

6 Ähnlichkeitssuche für Bauteile

Nach der Analyse der Baugruppensuche mittels Skizzen als Suchinput soll in diesem Abschnitt die Ähnlichkeit von Bauteilen untersucht werden. Das Vorgehen und die Ergebnisse sind bereits in [P9, P10] veröffentlicht worden und Teilaspekte in [S10–S13] bearbeitet. Zunächst soll das entwickelte Vorgehen präsentiert und ausführlich erläutert werden, danach der Einfluss der verschiedenen Parameter ermittelt und die erzielten Ergebnisse mit bereits vorhandenen Verfahren verglichen werden. Der erste Schritt ist dabei festzulegen, welche Anforderungen der Suchprozess von Bauteilen im Kontext der Produktentwicklung erfüllen muss.

Das übergeordnete Ziel des Vorgehen ist, eine möglichst hohe Erkennungsgenauigkeit von Bauteilen aus dem Maschinenbau-Umfeld zu erreichen. Dabei soll die globale geometrische Ähnlichkeit zwischen Bauteilen als Vergleichselement herangezogen werden, wodurch kein weiterer Suchinput, wie beispielsweise Material, Funktion oder Fertigungsverfahren, für eine erfolgreiche Anwendung notwendig ist. Diese Bedingung setzt auch fest, dass kein Labeling der Bauteile erfolgen darf, da dies den Arbeitsaufwand für einen Einsatz drastisch erhöht. Somit sind alle Verfahren, die auf (weakly- oder semi-) supervised Learning setzen, nicht nutzbar für die gestellte Aufgabe. Weiterhin soll eine Suchanfrage auch in einem gewissen zeitlichen Rahmen durchführbar sein, damit die Anwendung flüssig in den bestehenden CAD-Entwurfsprozess integriert werden kann. Die Laufzeit für eine Suchanfrage ist immer abhängig von der eingesetzten Hardware, weshalb keine konkreten Zeiten definiert werden. Allerdings sollte eine Suche auf einem normalen Workstation PC nicht länger als 5 Minuten benötigen. Zuletzt soll das Vorgehen möglichst Geometrieformat-unabhängig einsetzbar sein, weshalb CAD-proprietäre Formate nicht in Fragen kommen. Dagegen bieten Oberflächennetze eine gute Repräsentationsqualität der Geometrie bei geringerem Speicherbedarf (siehe Kapitel 2.1.2.3). Zudem lassen sich diese einfach handhaben und manipulieren und können immer ausgehend von einem CAD-Modell generiert werden.

In Verknüpfung mit der ausführlichen Literaturrecherche aus Kapitel 2.2.3.3 und der daraus entstanden Grafik 35 sind vor allem die ansichtsbasierten Verfahren als zielerfüllend identifiziert worden. Diese werden häufig eingesetzt und erzielen dabei in Benchmark-Vergleichen hohe Genauigkeitswerte. Allerdings wird dabei oft ein Label für das Training benötigt. Somit ist die Idee, die Fähigkeit der guten Bilderkennung zu nutzen, jedoch vortrainierte Modelle zu verwenden, welche kein neues Training erfordern und damit in der Lage sind, den Feature-Vektor für die jeweilige Geometrie zu erzeugen. Dieses Prinzip ist in ein Gesamtvorgehen überführt worden, welches in den

kommenden Kapiteln genauer vorgestellt und anhand eines Benchmarkdatensatzes bewertet wird.

6.1 Modulübersicht

Das grundlegende Vorgehen zur Ähnlichkeitssuche orientiert sich an dem allgemeinen Ansatz, welcher im Stand der Technik in Bild 33 dargestellt ist. Das Ziel des gesamten Ansatzes ist dabei, die 3D-Geometrie in einen Formdeskriptor zu überführen, welcher für einen Abgleich unter den Deskriptoren genutzt werden kann. In Bild 71 sind die Hauptabschnitte aufgeführt inklusive der jeweiligen Subprozesse.

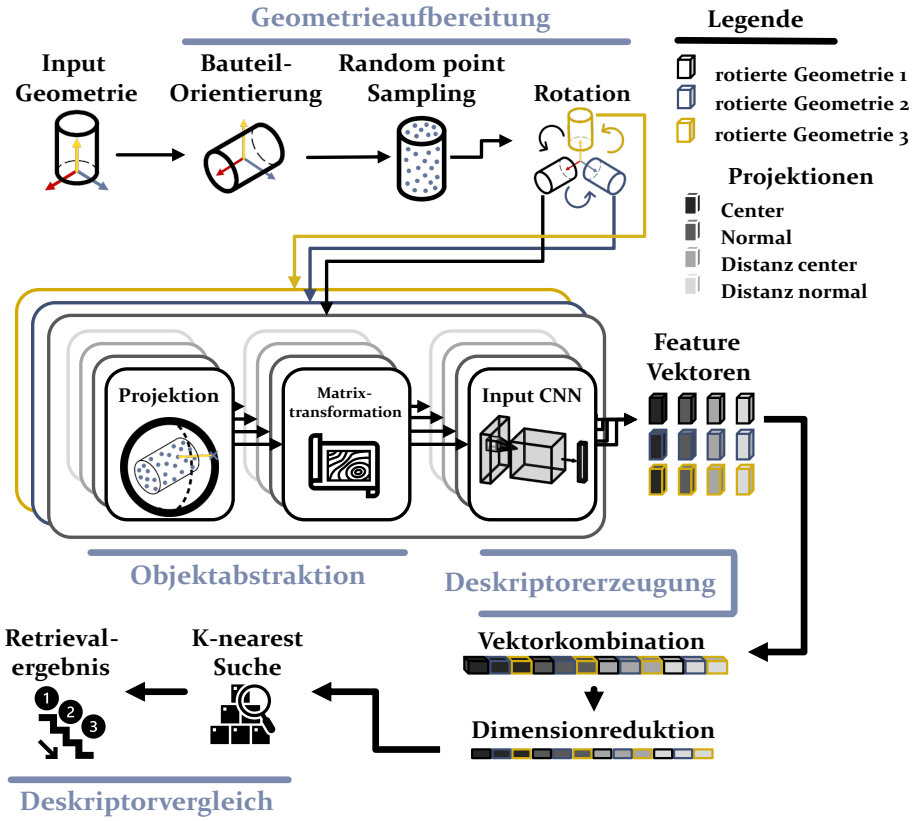


Bild 71: Übersicht des gesamten Vorgehens zur Ähnlichkeitssuche von Bauteilen, angelehnt an die Prozessabschnitte von Bustos et al. [252] nach [P10]

Der erste Schritt in diesem Prozess ist das Aufbereiten der Geometrie, wobei der Ablauf der Bauteilerkennung aus Kapitel 5.1.2 ähnelt. Zunächst wird das 3D-Modell gleichmäßig ausgerichtet, in eine Punktewolke umgewandelt und

diese um 90° in allen drei Hauptachsen rotiert. Danach erfolgt die Objektabstraktion mittels der Projektion auf eine Detektorkugel und anschließender Umwandlung in eine Matrix. Diese Matrizen können über verschiedene Algorithmen verglichen werden, beispielsweise über Korrelations-Koeffizienten oder die euklidische Distanz. Eine weitere Alternative ist das Verwenden von vortrainierten CNN-Modellen, welche ebenfalls die Matrizen als Input verwenden können und der Feature-Vektor als Shape-Deskriptor genutzt werden kann. Der erzeugte Vektor kann anschließend für eine k-nearest-neighbors-Suche [441] mit den Vektoren der Datenbank verglichen werden, um so die ähnlichste Geometrie zum Input-3D-Modell zu finden.

Das Grundprinzip dabei ist, über die Projektion möglichst viele geometrische Informationen aus einem Oberflächennetz zu extrahieren, damit ein möglichst passendes Bauteil gefunden werden kann. Weiterhin erfüllt das Vorgehen die gestellten Anforderungen, da kein zusätzliches Labeling der Datenbasis notwendig ist, die Transformation schnell abläuft, da kein Training notwendig ist, und Oberflächennetze als Input verwendet werden können. Deshalb sollen nachfolgend die einzelnen Teilschritte im Detail präsentiert und erläutert werden.

6.1.1 Datenaufbereitung

Im ersten Schritt der Datenaufbereitung muss die Geometrie als Oberflächennetz importiert werden, gefolgt von einem wichtigen Schritt im Gesamtprozess, der Bauteilausrichtung. In der realen Anwendung sind CAD-Bauteile nie einheitlich ausgerichtet, da die Orientierung der Bauteile von vielen Variablen abhängig ist, wie z. B. der späteren Montageposition, den Design-Randbedingungen oder dem Konstrukteur. Da für die Objektabstraktion ein Projektionsverfahren angewendet wird, ist das gesamte Vorhaben ansichtsbasiert und somit rotationsabhängig, wodurch eine einheitliche Ausrichtung der Bauteile essentiell für die Funktionserfüllung ist. Insgesamt sollen im Rahmen dieser Arbeit zwei unterschiedliche Methoden zur Bauteilausrichtung untersucht werden. Die erste ist eine neu entwickelte Orientierungsmethode, welche auf dem Konstruktionsprozess von Bauteilen in einem CAD-Programm basiert und die Orientierung dabei nachstellen soll. Als Vergleichsansatz dient die Hauptkomponentenanlyse, welche bereits in Kapitel 5.1.2 erläutert und eingesetzt wurde. Nach der einheitlichen Orientierung der Oberflächennetze können diese über eine zufällige Verteilung von Punkten auf der Oberfläche in eine Punktewolke transformiert werden. Dazu ist im Anhang in den Bildern 114 und 115 ein Vergleich der Anzahlen an Punkten einer Punktewolke über die Projektionsqualität und Laufzeit durchgeführt worden. Dabei erwies sich die Menge von 50.000 Punkten pro Bauteilmodell

als guter Kompromiss zwischen Genauigkeit und Transformationszeit. Die so erzeugten Punktewolken werden danach noch um die X-, Y- und Z-Achse in 90°-Schritten rotiert, was zu insgesamt 24 unterschiedlichen Ausrichtungen des Inputbauteils führt.

6.1.1.1 Bauteilausrichtung

Der Einsatz der PCA zur Bauteilausrichtung ist etabliert und liefert allgemein gute Ergebnisse. Allerdings ist die Anwendung schwierig bei Bauteilen aus dem Maschinenbau-Umfeld, da diese mehrere Löcher oder Ausschnitte aufweisen können, was zu uneinheitlichen Ausrichtungsergebnissen führt. Ein weiteres Problem der Hauptkomponentenanalyse ist, dass viele Bauteile nicht symmetrisch sind und somit die Ausrichtung selten flach auf einer Ebene des globalen Koordinatensystems liegt. Dagegen werden in der Produktentwicklung Bauteile normalerweise nach einem definierten Muster konstruiert, das in der PCA nicht repräsentiert wird. Dies soll durch das neue Verfahren zur Bauteilausrichtung berücksichtigt werden, welches allgemeine Konstruktionsprinzipien mechanischer Bauteile beachtet. Die Grundidee ist, dass CAD-Modelle in Z-Richtung extrudiert und somit in der X-Y-Ebene skizziert werden. Weiterhin soll die längste Kante der Geometrie in Richtung der X-Achse orientiert sein. In Bild 72 ist das entwickelte Vorgehen aufgeführt, welches bereits vorhandene Bauteile so gut wie möglich in diese Struktur zurückführen soll. Zudem ist im Anhang der prinzipielle Code für das Ausrichtungsverfahren in den Algorithmen 3 und 4 aufgelistet.

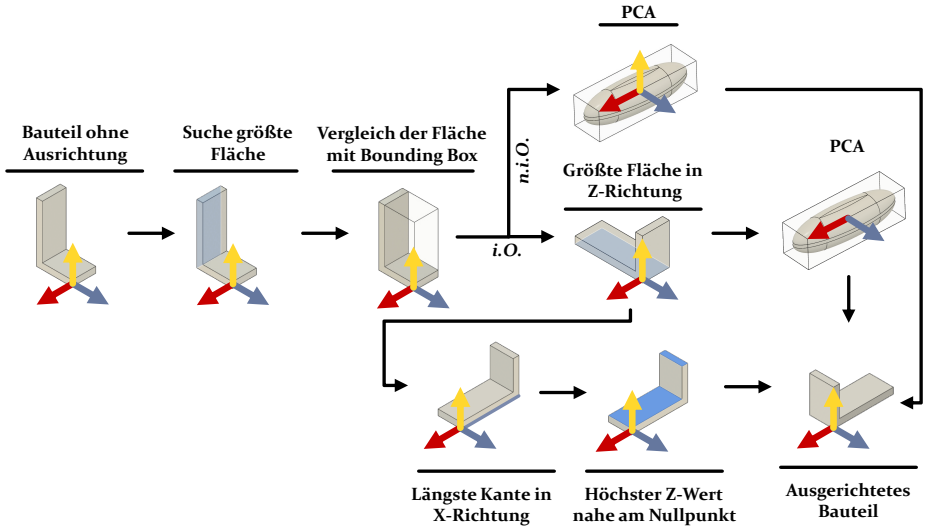


Bild 72: Vorgehen zur Ausrichtung der Bauteile nach dem Prinzip des CAD-Designs nach [P10]

Im ersten Schritt des Prozesses müssen die größten benachbarten Flächen der Geometrie mit der gleichen Normalenrichtung identifiziert werden. Wenn allerdings die größte Fläche zu klein im Vergleich zur größten Fläche der Bauteil Bounding Box ist, wird das Oberflächennetz durch die PCA ausgerichtet und alle nachfolgenden Schritte übersprungen. Der Vergleich dient dazu, um Bauteile mit einem hohen Freiform-Anteil zu erkennen und diese über die PCA auszurichten, anstatt über ihre Kanten, da dies zu uneinheitlichen Ergebnissen führen würde. Tritt der Fall nicht ein, wird das 3D-Modell an der Z-Achse ausgerichtet. In der zweiten Überprüfung wird getestet, ob die größte Fläche rund oder symmetrisch ist. Falls dies zutrifft wird eine PCA nur in der X-Y-Ebene durchgeführt und das Bauteil so final orientiert. Bei Nichterfüllung der Bedingung wird im Anschluss das Modell über die längste Kante der Fläche parallel zur X-Achse ausgerichtet mit Hilfe einer Rotation um die Z-Achse. Schließlich kann die Geometrie noch um 180° gedreht werden, wobei die Seite mit dem größeren Z-Wert näher am Ursprung liegen sollte.

Nach der theoretischen und algorithmischen Beschreibung der unterschiedlichen Ausrichtungsverfahren, werden in Bild 73 die Ergebnisse der Ansätze für verschiedene Bauteile illustriert. Insgesamt sind die originale Ausrichtung der Bauteile aus dem Datensatz, die PCA-Variante und das neue Verfahren für fünf 3D-Modelle aus dem ESB-Datensatz (vgl. Kapitel 2.2.3.1) gegenübergestellt.

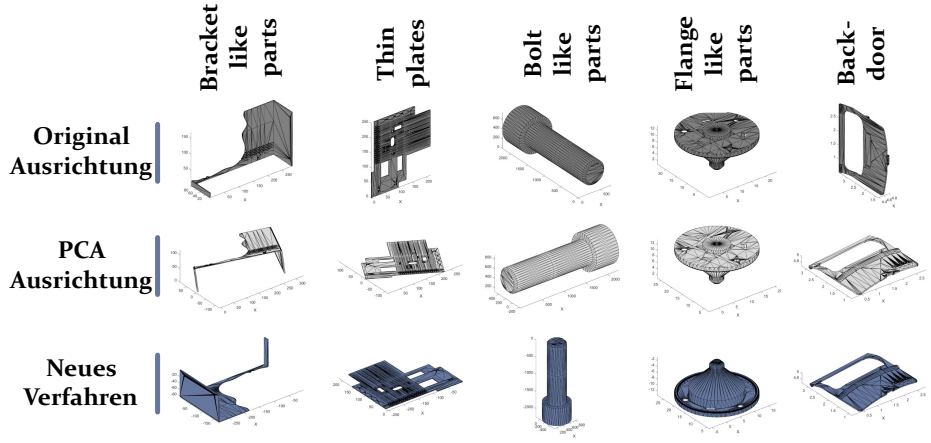


Bild 73: Beispiele für die unterschiedlichen Bauteilausrichtungsverfahren nach [P10]

Die originale Ausrichtung ist dunkelgrau dargestellt und zeigt wie uneinheitlich die 3D-Modelle vorliegen. Darunter sind die durch die PCA-orientierten

Bauteile in hellgrau und durch das neue Vorgehen ausgerichteten Oberflächennetze in blau geplottet. Beim Vergleich der Verfahren veranschaulicht die Abbildung deutlich, dass die neue Methode bei ungleichmäßigen Teilen, wie dem ersten und zweiten Netz, konsistentere Ausrichtungen liefert als die PCA-Variante. Dagegen ist der Unterschied bei rotatorischen Bauteilen weniger eindeutig, da das Unterscheidungsmerkmal hauptsächlich die Ausrichtung der Rotationsachse darstellt. Im konkreten Beispiel orientiert das PCA-Verfahren diese nach der X-Achse, wogegen der neue Ansatz die Z-Achse wählt. Das letzte Bauteil ist ein Beispiel für den Fall, dass der Bereich der größten Fläche zu klein ist und somit eine Ausrichtung durch die PCA erfolgt, was zu identischen Ausrichtungsergebnissen mit der PCA-Variante führt. Die Demonstration des Vorgehens mittels Beispielbauteilen soll die Vorzüge des neuen Ansatzes zeigen und durch das einheitliche Ausrichten die Genauigkeit bei der Ähnlichkeitssuche verbessern.

6.1.2 Objektabstraktion

Nach der erfolgreichen Ausrichtung und anschließenden Rotation der Punktewolken müssen diese im Schritt der „Objektabstraktion“ in ein einheitliches Format überführt werden. Dazu werden erneut die Projektionsverfahren verwenden, welche bereits in Kapitel 5.1.2.1 erklärt wurden.

Bei der Transformation werden wiederum die beiden Projektionsarten „Mittelpunkt“ und „Normalen-Richtung“ eingesetzt. Beim Einsatz für die Bauteilerkennung werden dabei die Punkte pro Pixel gezählt und mit diesem Wert die Matrizen befüllt. Für die Ähnlichkeitssuche ist noch eine neue Variante entwickelt worden, die Distanz-Projektion. Diese verwendet den Abstand vom Mittelpunkt zum Punkt auf die Detektorkugel als Pixeleintrag. Anschließend wird das arithmetische Mittel pro Pixel berechnet und in die Matrix überführt. Damit stehen insgesamt vier unterschiedlichen Arten der Projektion zur Verfügung, welche für die Suche nach ähnlichen Geometrien eingesetzt werden können.

6.1.3 Shape-Deskriptor-Erzeugung

Die so erzeugten Matrizen müssen im nächsten Schritt in einen Vergleichsvektor, auch Shape-Deskriptor genannt, übertragen werden. Dazu sollen verschiedene Verfahren getestet und auf ihrer Leistungsfähigkeit untersucht werden. Das Unterscheidungskriterium ist dabei, ob die Matrix direkt für den Vergleich verwendet wird, oder nochmals in einen Vektor umgewandelt und dieser dann für den Ähnlichkeitsvergleich herangezogen wird. Beide Varianten werden im Folgenden mit den zugehörigen Algorithmen vorgestellt.

6.1.3.1 Matrizen-Vergleich

Die ersten Vorgehen nutzen direkt zwei Matrizen als Input, liefern den Ähnlichkeitswert als Ergebnis und wurden in [P9] erstmals veröffentlicht. Die Algorithmen stammen häufig aus der Bildverarbeitung und untersuchen die Ähnlichkeiten zwischen Eingabebildern. Dabei sind insgesamt vier verschiedene Algorithmen in Betracht gezogen worden: Der Korrelations-Koeffizient (corr2), der structural similarity Index (ssim) [442], peak signal-to-noise-ratio (psnr) und die euklidische Distanz (l_1). Für die konkrete Umsetzung sind folgende Implementierungen [443–445] verwendet worden.

Der Korrelations Koeffizient variiert zwischen -1 und 1, wobei 1 den höchsten Ähnlichkeitswert beschreibt. Ähnlich verhält sich der ssim-Wert, welcher zwischen 0 und 1 liegt und auch hier 1 den besten Ähnlichkeitsscore darstellt. Die peak signal-to-noise-ratio ist immer ein positiver Ergebniswert, wobei ein höherer Wert einer größeren Ähnlichkeit entspricht. Auch die euklidische Distanz kann nur positive Werte oder 0 annehmen, allerdings repräsentiert hier ein kleinerer Ergebniswert eine höhere Ähnlichkeit.

6.1.3.2 pretrained CNNs

Eine weitere Möglichkeit für die Erzeugung eines Vergleichsvektors ist die Verwendung von vortrainierten CNNs (vgl. Kapitel A.3). Diesen Netzen werden die Matrizen als Eingabe übergeben und anschließend werden alle Convolutional- und Pooling-Operationen bis zur letzten Faltungsschicht bzw. der letzten Schicht vor der voll verbundenen Schicht durchlaufen. Dadurch wird die Matrix der Größe $256 \times 256 \times 3$ in einen Vektor der Dimension $1 \times 2048 \times 1$ übertragen (Beispiel resnet50). Zum Einsatz kommen dabei Modelle, welche mit dem ImageNet trainiert wurden, welches aus fast 1,3 Millionen Bildern und 1.000 Klassen besteht. Auf diese Weise werden die trainierten Merkmalsextraktions-Eigenschaften des CNN auf die Matrizen angewendet, ohne dass ein Netz mit gelabelten Daten trainiert werden muss.

Ein genereller Überblick über die Nutzung von pretrained Features in CNNs liefert [446]. Zusätzliche Beispiele für die Anwendung zur Bildersuche sind in [447–449] aufgeführt. In dieser Umsetzung für die Ähnlichkeitssuche von Bauteilen wird für jede Geometrie und Rotation im gleichen Verfahren ein Vektor erzeugt, welcher im Anschluss über Dimensionsreduktionsalgorithmen verkleinert werden kann. Dadurch sollen die relevanten Merkmale heraus kristallisiert und somit der Vergleich zwischen Vektoren verbessert werden. Allgemein sind die Methoden zur Dimensionsreduktion sehr vielfältig, wobei in [450–453] ein Überblick der möglichen Methoden gegeben wird. Die Bandbreite erstreckt sich von der Hauptkomponentenanalyse, welche

bereits vorgestellt und bei der Bauteilerkennung eingesetzt wurde, bis zu Methoden des unüberwachten Lernens, wie beispielsweise Autoencoder.

Nach der Reduktion können die aufbereiteten und transformierten Vektoren auf ihre Ähnlichkeit untereinander untersucht werden. Hierzu wird eine k-nearest-neigbhour-Suche verwendet, welche den ähnlichsten Vektor mittels verschiedener Distanzmaße ermittelt. Im Rahmen der Arbeit wurden insgesamt vier verschiedene Distanzmaße untersucht, der euklidische Abstand, Cityblock-, Korrelation- und Kosinus-Distanz, welche ausführlich in [454–456] beschrieben und erläutert werden.

Fazit Kapitel 6.1

In diesem Kapitel erfolgte die Vorstellung des Ansatzes zur Suche von Bauteilen aus dem Umfeld der Produktentwicklung. Dabei lag der Fokus auf der Berücksichtigung der Anforderungen (Kein Label, Zeitbedarf und hohe Anwendbarkeit) aus dem Maschinenbau-Umfeld bei der Entwicklung des neuen Vorgehens. Demzufolge ist ein Vorgehen aus vortrainierten CNN-Modellen in Kombination mit einem Multi-View-Projektionsansatz entwickelt und die jeweiligen Schritte ausführlich theoretisch erläutert worden. Die so erzeugten Vektoren können noch durch zusätzliche Dimensionreduktionsverfahren verkleinert werden, damit der Suchprozess effizienter abläuft.

6.2 Untersuchung der Einflussfaktoren auf die Retrievalgenauigkeit

Nach der theoretischen Darlegung der Vorgehensweise werden in diesem Abschnitt die Einflüsse der unterschiedlichen Prozessschritte analysiert. Neben dem vorgestellten Vorgehen sind der Datensatz und die Auswertemetriken ebenfalls notwendig, damit eine Bewertung möglich wird. Die Daten entstammen dem „Engineering Shape Benchmark“ (vgl. Kapitel 2.2.3.1), welcher sich besonders gut für die Ähnlichkeitssuche eignet, da der Datensatz auf mechanische Komponenten spezialisiert und bereits seit 2005 verfügbar ist. Dadurch wird die Vergleichbarkeit der Ergebnisse erhöht, da viele andere Methoden den ESB-Datensatz ebenfalls verwendet haben. Als Metriken dienen die etablierten Verfahren für die Informationssuche, der Nearest Neigbhour (NN), First Tier (FT), Second Tier (ST) und Discounted Cumulated Gain (DCG). Alle Variablen wurden ebenfalls im Stand der Technik in Kapitel 2.2.3.3 beschrieben.

Mit diesen Rahmenbedingungen werden nachfolgend die Einflüsse auf die Ähnlichkeitsgenauigkeit untersucht. Dazu soll chronologisch im Rahmen der entwickelten Methoden vorgegangen werden, indem zuerst die Bauteilausrichtung, danach die Featureextraktion, gefolgt von der Projektionsart und Distanzmetrik und zuletzt die Dimensionsreduktion analysiert und verglichen werden.

6.2.1 Bauteilausrichtung

Damit unterschiedliche Bauteilorientierungen getestet werden können, müssen die übrigen Parameter des Vorgehens fest definiert werden. Daher ist die Punkteanzahl pro Komponente auf 50.000 festgesetzt, die Projektionsauflösung beträgt 256 × 256 und die Summe aller Projektionen wird als Input verwendet. Das vortrainierte Modell ist ein efficientnet [418], als Distanzmetrik wird der euklidische Abstand genutzt und es findet keine Dimensionsreduktion statt. Mit diesen Parametern sollen die drei Bauteilausrichtungen getestet werden: die originale Orientierung aus dem Datensatz, die PCA Ausrichtung und das neue CAD-inspirierte Vorgehen. Zunächst ist ein visueller Vergleich der Ergebnismatrizen für zwei Bauteile aus der gleichen ESB-Kategorie durchgeführt worden, um die Effektivität des neuen Verfahrens optisch zu prüfen, die Ergebnisse sind in Bild 74 dargestellt.

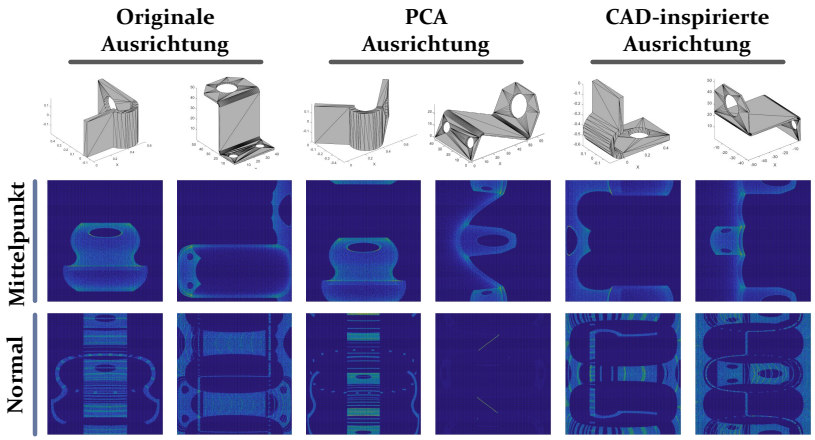
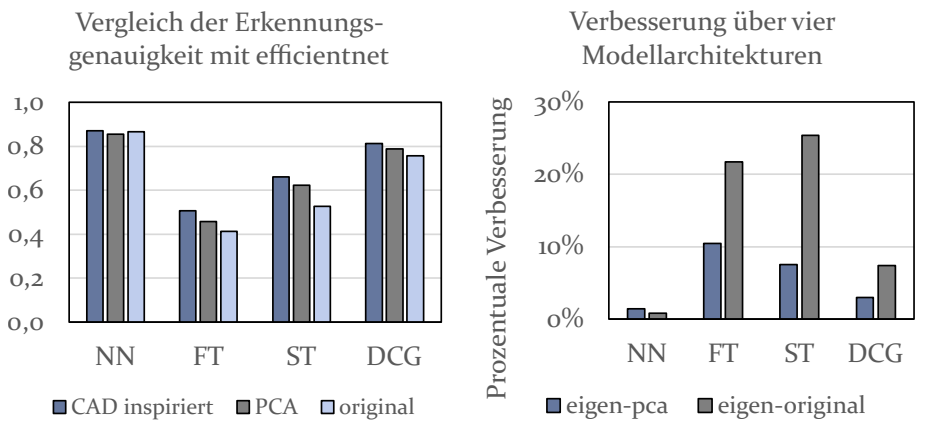


Bild 74: Optischer Vergleich der drei Methoden zur Bauteilorientierung und der entsprechenden Projektionsmatrizen nach [P10]

Der visuelle Vergleich zeigt klar, dass die CAD-inspirierte Ausrichtung die Matrizenerzeugung deutlich verbessert, indem die Merkmale der Geometrie gleichmäßig in der Projektionsmatrix dargestellt werden. Nach dem

vielversprechenden Vergleich sind die Retrieval-Ergebnisse in zwei Balkendiagrammen in Bild 75 geplottet. Im Diagramm auf der linken Seite ist das CAD-inspirierte Vorgehen in dunkelblau, die PCA in grau und die originale Ausrichtung in hellblau dargestellt. Der Vergleich über alle vier Metriken lässt die Schlussfolgerung zu, dass die CAD-inspirierte, neue Ausrichtungsmethode bessere Suchergebnisse liefert als die beiden anderen Vorgehen. Insbesondere die, für die Produktentwicklung relevanten, Metriken First- und Second-Tier werden deutlich verbessert. Auf der rechten Seite des Bildes ist zusätzlich noch die prozentuale Steigerung für die vier Modell-Architekturen (efficientnet, resnet50 [148], googlenet [457] und densenet [416]) aufgeführt. Diese Untersuchung soll ausschließen, dass die erste Analyse nur eine zufällig gut gewählte Kombination der vielen Parameter darstellt. Auch hier zeigt sich jeweils eine Verbesserung der Ergebniswerte, vor allem für die FT und ST. Der NN-Wert weist dabei die geringste Änderung auf. Insgesamt lässt diese Untersuchung den Schluss zu, dass die neue Bauteilausrichtung eine deutliche Verbesserung gegenüber keiner Ausrichtung und auch der PCA darstellt. Dies stimmt mit den vorherigen Annahmen über das gesamte Vorgehen überein, da aufgrund der Rotationsabhängigkeit der ansichtsbasierten Methoden die gleichmäßige Ausrichtung der Bauteile einen Einfluss auf die Suche hat.



(a) Ausrichtungsvergleich mit efficientnet

(b) Prozentuale Verbesserung über vier CNN-Modelle

Bild 75: Vergleich der unterschiedlichen Bauteilausrichtungen nach [P10]

6.2.2 Featureextraktion

Für die Untersuchung der Featureextraktion sind die gleichen Parameter für die Ähnlichkeitssuche aus der ersten Analyse übernommen worden. Insgesamt sollen zwölf verschiedene mit dem ImageNet vortrainierte CNN-Architekturen auf ihre Tauglichkeit zur Bauteilsuche untersucht werden, wobei folgende Modelle ausgewählt wurden: efficientnet [418], densenet201, squeezenet [458], inceptionv3 [417], googlenet, resnet18, resnet50, resnet101, shufflenet [459], darknet19, darknet53 (beide [460]), vgg16, vgg19 (beide [146]) und nasnet mobile [461]. Der Feature-Vektor wird jeweils vor der ersten fully connected Schicht aus dem Modell extrahiert und die Input-Matrizen werden jeweils auf die geforderte Input-Größe des jeweiligen Modells angepasst, welche in der Regel bei 256×256 liegt. Eine ausführliche Übersicht der erzielten Ergebnisse für die verschiedenen Architekturen liefert die Tabelle 5. In dieser sind die Evaluationsmetriken mit den jeweiligen Modellen aufgelistet, sortiert nach dem First-Tier-Wert.

Tabelle 5: Ergebnisse für verschiedene vortrainierte Netze und euklidischer Distanz

Modell-Architektur	**NN**	**FT**	**ST**	**DCG**
efficientnet b0	0,871	**0,507**	**0,662**	**0,813**
densenet201	0,870	0,498	0,653	0,807
resnet50	**0,873**	0,496	0,654	0,807
resnet101	0,870	0,492	0,647	0,806
googlenet	0,863	0,485	0,648	0,803
resnet18	0,870	0,484	0,647	0,804
shufflenet	0,865	0,482	0,636	0,802
squeezenet	0,857	0,475	0,626	0,795
vgg16	0,867	0,468	0,612	0,789
vgg19	0,855	0,460	0,599	0,784
inceptionv3	0,820	0,406	0,526	0,744
darknet19	0,813	0,381	0,488	0,729
darknet53	0,801	0,354	0,460	0,716
nasnetmobile	0,804	0,353	0,458	0,709

Grundsätzlich zeigt die Tabelle, dass viele Modelle ein sehr gutes Ergebnis für die Ähnlichkeitssuche erzielen können. Da dieser Trend über mehrere Modelle erkennbar ist, bestätigt dies eine gewisse Robustheit des Vorgehens und eine zufällige Kombination aus Modell und Vorgehensparametern ist eher auszuschließen. Die besten Werte sind jeweils fett und unterstrichen, wobei für den NN-Wert auffällt, dass vor allem Architekturen mit weniger Schichten

hier gute Ergebnisse erzielen. Am besten performte das efficientnet mit einem FT-Ergebnis von 0,507 und einem ST von 0,662. Die weiteren Modelle wie das densnet, resnet50/101 oder googlenet waren nur sehr knapp dahinter, alle mit sehr guten Retrieval-Ergebnissen. Da allerdings das efficientnet insgesamt am besten in dieser Kombination funktioniert hat, wird diese Architektur für die nächsten Untersuchungen verwendet.

Neben den vortrainierten Modellen sind im vorherigen Kapitel die Algorithmen zum Vergleich der Matrizen vorgestellt worden. Allerdings liefern diese als Ergebnis direkt einen Ähnlichkeitsscore, weshalb die nächsten Kapitel für diese Vorgehen nicht relevant sind. Daher wird erst in dem Abschnitt 6.3.1 eine Gegenüberstellung beider Vorgehen durchgeführt.

6.2.3 Projektionsart und Distanzmetrik

Neben den CNN-Architekturen können auch die Input-Matrizen unterschiedlich zusammengesetzt werden, bedingt durch die verschiedenen Projektionsarten aus Kapitel 5.1.2.1. Dementsprechend sollen unterschiedliche Kombinationen der Matrizen in Verbindung mit verschiedenen Distanzmetriken in dieser Sektion genauer analysiert werden. Insgesamt sind vier Projektionsarten möglich, welche miteinander verknüpft werden können, was in der Tabelle 6 mit den entsprechenden Formeln aufgeführt ist. Zum Einsatz kommt dabei die Summe aus allen Matrizen, nur die Mittelpunkts- bzw. Normalen-Projektion, nur die Distanz- bzw. alle Nicht-Distanz Matrizen.

Tabelle 6: Gleichungen für alle verschiedenen Feature-Vektor-Kombinationen

Summe	**Center Alle**	**Normale Alle**	**Distanz Alle**	**Nicht-Distanz Alle**
$c + n + dc + dn$	$c + dc$	$n + dn$	$dc + dn$	$c + n$
mit c = Mittelpunkt Matrix; n = Normalen Matrix;				
dc = Distanz Mittelpunkt Matrix; dn = Distanz Normalen Matrix				

Der Vergleichsalgorithmus ist mit der k-nearest-neigbhour-Suche vorgegebenen, allerdings kann die Berechnung der Distanz auf diverse Arten erfolgen. Im Rahmen der Analyse der Bauteilähnlichkeitssuche sind vier Algorithmen zur Berechnung der Distanz untersucht worden: korrelation, cityblock, cosine und euclidean [462]. Zur Auswertung sind die verschiedenen Matrizen-Zusammensetzungen und Distanz-Maße für jede der vier Evaluationsmetriken in einem Heatmap-Plot in Bild 76 dargestellt.

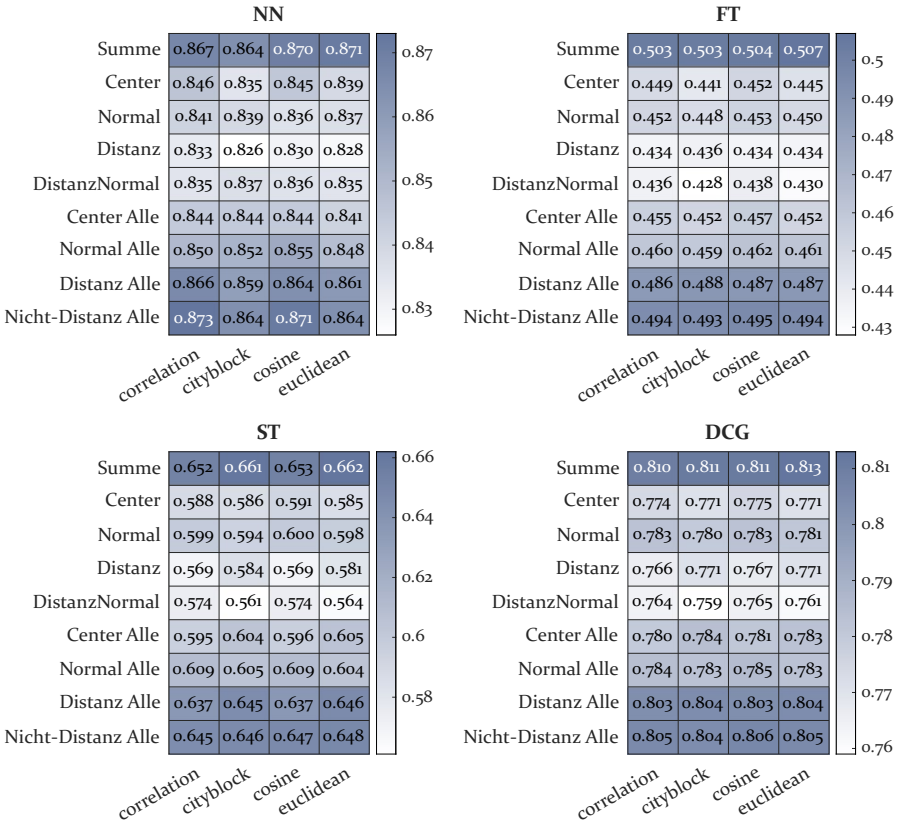


Bild 76: Ergebnis Heatmap für den Vergleich zwischen den Distanzmaßen und verschiedenen Projektionsarten und -zusammensetzungen

Die Grafiken demonstrieren dabei deutlich den Einfluss der Inputmatrizen-Zusammensetzung auf die Suchgenauigkeit des Gesamtvorgehens. Auffallend ist, dass eine Kombination aus Mittelpunkt- und Normalen-Projektion bei allen vier Bewertungsmetriken eine eindeutige Verbesserung im Vergleich zu nur einer Projektionsart erzielt. Eine direkte Gegenüberstellung der Matrizen-Konfigurationen zeigt, dass die Gesamtsumme bessere Ergebnisse erreicht als die anderen Kombinationen.

Dagegen ist bei der Analyse der Distanzmaße zu erkennen, dass diese weniger Relevanz für die Suchergebnisse haben. Allerdings kristallisieren sich die Abstandsmaße Korrelation und Kosinus unter allen vier Maßen heraus, da diese zu einer kleinen Steigerung der Evaluationsmetriken führen, allerdings nur für die ausgewählten Matrizen-Kombinationen. Für die Summe aller Matrizen ist dagegen die euklidische Distanz am besten geeignet, was

deutlich in den vier Heatmaps zu erkennen ist. Zusammenfassend führen die Untersuchungen zu der Erkenntnis, dass die Summe aller generierten Matrizen in Kombination mit der Euklidischen-Distanzmetrik am besten geeignet für eine Bauteilsuche im Maschinenbau-Kontext ist. Die erzielten Auswertemetriken sind: NN: 0,871, FT: 0,507, ST: 0,662, DCG: 0,813

6.2.4 Dimensionsreduktion

Der letzte Analyseparameter ist die Dimensionreduktion des umgewandelten Feature-Vektors. Dazu ist eine Vielzahl an Verfahren einsetzbar, alle mit dem Ziel, die relevanten Information herauszufiltern und dabei den Vektor zu verkleinern. Dadurch wird, unabhängig von einer Verbesserung der Suchergebnisse, ein schnellerer Suchprozess aufgrund der reduzierten Vektorgröße für den Ähnlichkeitsabgleich ermöglicht.

Aus dem verfügbaren Verfahrenspool sind drei Methoden ausgewählt worden, welche in der Tabelle 7 aufgelistet sind. Der erste Wert neben dem Reduktionsverfahren beschreibt die Reduktionsgröße für einen gegebenen Input-Vektor. Der Wert für die Ansichtskompression gibt an, wieviele Ansichten gleichzeitig dem Reduktionsalgorithmus übergeben werden. Dabei bedeutet beispielsweise ein Wert von 24, dass alle Ansichten gemeinsam übergeben und auf die entsprechende Dimension reduziert werden. Dieser Wert ist in vier Abstufungen (1, 6, 12, 24) untersucht worden. Somit wird der Merkmalsvektor jeder transformierten Matrix konzentriert und im Anschluss wieder zu einem großen Vektor zusammengesetzt. Insgesamt sind für alle Verfahren mehrere Reduktionsdimensionen und Ansichtskompressionen untersucht worden, wobei in der Tabelle jeweils das beste Ergebnis aufgeführt ist.

Tabelle 7: Übersicht der besten erzielten Suchergebnisse über alle getesteten Reduktionsverfahren

Reduktions-verfahren	Reduktions-dimension	Ansichts-kompression	**NN**	**FT**	**ST**	**DCG**
Baseline	-	-	0,871	0,507	0,662	0,813
Auto-encoder	25	3	0,871	0,530	0,689	0,827
PCA	25	6	0,868	0,517	0,681	0,820
t-SNE	100	3	0,860	0,517	0,670	0,814

Insgesamt sind drei Verfahren zur Reduktion der Dimension angewendet worden, Autoencoder [463], Hauptkomponentenanalyse und t-distributed stochastic neighbor embedding (t-SNE) [464]. Als Input wird wiederum die

Summe aller Matrizen herangezogen, welche durch das efficientnet umgewandelt und mittels der euklidischen Distanz auf die finale Ähnlichkeit verglichen wurden. Die ausführlichen Ergebnisse der Studie sind im Anhang für alle Dimensionsreduktionsalgorithmen in Tabelle 22 aufgelistet.

Die Ergebnisse in der Tabelle bestätigen die Annahme, dass durch eine Dimensionsreduktion des Eingabevektors eine Verbesserung der Suchergebnisse erzielt werden kann. Alle getesteten Verfahren erhöhen vor allem die relevanten FT- und ST-Auswertemetriken. Die beiden besten Vorgehen sind nochmals extra in Bild 77 geplottet, wobei die Verbesserung im Vergleich zu den Ausgangswerten dargestellt ist. Die größte prozentuale Verbesserung konnte dabei der Autoencoder erreichen, welcher den First-Tier-Wert um ca. 4,5 % erhöht. Die Hauptkomponentenanalyse erreichte mit einer Aufbesserung von ca. 2 % leicht geringere Werte. Allerdings ist diese aufgrund der einfachen und schnellen Umsetzung eine sehr geeignete Variante für eine industrielle Anwendung, da sie die Genauigkeit erhöht und gleichzeitig die Suchzeit reduziert. Insgesamt zeigt diese Auswertung, dass eine Optimierung der Ergebnisse und in einigen Fällen sogar eine Beschleunigung des Prozesses durch eine Reduktion der Eingabedimension erreicht werden kann. Die erzielten Ergebnisse des neuen Ansatzes sollen daher im nächsten Kapitel mit dem aktuellen Stand der Technik verglichen und eingeordnet werden.

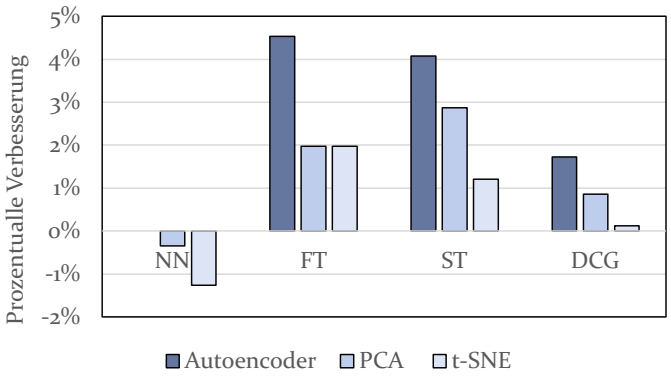


Bild 77: Verbesserung der Suchergebnisse durch Dimensionsreduktionsverfahren

Fazit Kapitel 6.2

Nach der ausführlichen Analyse der Einflussfaktoren auf die Ähnlichkeitssuche stellt sich heraus, dass vor allem die Bauteilausrichtung, die Dimensionsreduktion und das CNN-Modell großen Einfluss auf die Ergebnisse haben. Dagegen ist der Effekt der Matrizen-Zusammensetzung auf die Geometriesuche etwas geringer. Die Wahl der Distanzmetrik ist in dieser Rangfolge am ehesten zu vernachlässigen, da diese nur sehr kleine Auswirkungen hat. Als Fazit lässt sich daher festhalten, dass das präsentierte Vorgehen funktioniert und durch spezifische Stellschrauben systematisch verbessert werden kann.

6.3 Vergleich mit Stand der Technik Verfahren

Die erzielten Genauigkeitswerte für die Ähnlichkeitssuche sollen in diesem Kapitel mit den bestehenden Vorgehen abgeglichen werden, um den neuen Ansatz zu bewerten. Als Datenbasis wird wiederum der ESB-Datensatz verwendet. In einer ersten Untersuchung wird ein Vergleich zu dem Vorgehen aus [P9] durchgeführt. Im nächsten Schritt erfolgt die Einordnung zu anderen etablierten Vorgehen aus der Literatur. Somit soll am Ende dieses Kapitels eine Aussage über die Einordnung der Leistungsfähigkeit des neuen Ansatzes im Vergleich zu vorhandenen Vorgehen möglich sein.

6.3.1 Vergleich mit eigenen Vorarbeiten

Im ersten Abgleich der Verfahren werden die Algorithmen zum Matrizenvergleich aus Kapitel 6.1.3.1 gegenüber den pretrained CNNs getestet. Dazu sind die Aufbereitung der Daten und die Umwandlung in die Matrizen identisch für beide Vorgehen (50.000 Punkte pro Bauteile und 256 × 256 Auflösung).

Die Ergebnisse der verschiedenen Ansätze sind in Bild 78 mittels Balkendiagramm dargestellt. Dabei ist deutlich zu erkennen, dass unter den vier Methoden zum direkten Matrizenvergleich vor allem der Korrelationskoeffizient und der ssim gute Suchergebnisse erzielen, während der euklidische Abstand die schlechtesten Ergebnisse liefert. Allerdings ist das Experiment eine eindeutige Bestätigung, dass die neue Methode deutlich bessere Ergebnisse liefert. Das belegen vor allem die FT- und ST-Metriken, welche teilweise eine Verbesserung von über die Hälfte erfahren haben. Einzig bei dem nearest-neighbor-Wert ist der Unterschied weniger ausgeprägt, aber trotzdem noch erkennbar. Somit ist festzuhalten, dass die Verwendung von vortrainierten

CNNs eine klare Verbesserung der Suchergebnisse von 3D-Geometrie darstellt, verglichen mit den direkten Algorithmen für Matrizen.

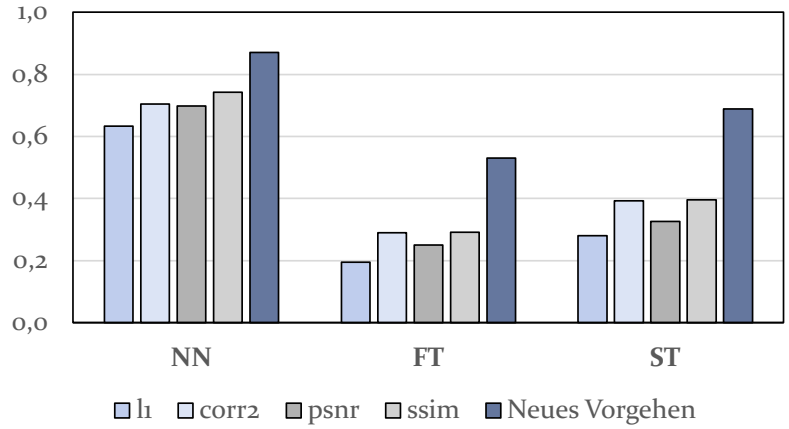


Bild 78: Vergleich des CNN-Shape-Deskriptor-Ansatzes mit den direkten Matrix-Vergleichsalgorithmen nach [P9, P10] mit den Verfahren Korrelations-Koeffizient (corr2), der structural similarity Index (ssim), peak signal-to-noise-ratio (psnr) und der euklidische Distanz (l1)

6.3.2 Vergleich mit etablierten Verfahren

Nach dem Abgleich mit den Matrizenalgorithmen soll im letzten Schritt der Untersuchung der Vergleich zu anderen Verfahren zur Ähnlichkeitssuche von Geometrien erfolgen. Dazu ist im Anhang unter Kapitel A.9 eine ausführliche Erläuterung aufgeführt, welche zusammen mit der Tabelle 13 die aktuellen Verfahren auflistet. Dabei werden nur Vorgehen berücksichtigt, die auch den ESB-Datensatz (siehe Kapitel 2.2.3.1) als Grundlage für eine Suchanfrage verwenden.

Damit die erzielten Ergebniswerte besser untereinander verglichen werden können, wurde ein Auszug aus der Tabelle 13 in ein Balkendiagramm umgewandelt. Hier werden aus Gründen der Übersichtlichkeit nur die Vergleichswerte der sechs relevanten Verfahren herausgegriffen und in Bild 79 dargestellt. Die unterschiedlichen Varianten des neu entwickelten Ansatzes sind mit jeweils unterschiedlichen gelben Schraffuren im Diagramm hervorgehoben. Als relevant werden alle Vorgehen eingestuft, die kein Bauteil-Label oder Datensatz-spezifische Parameter (z. B. abhängig von der Anzahl oder Geometrie der Bauteile im Datensatz) für die Anwendung benötigen. Eine Adaption auf Datensatz-spezifische Parameter und der Vergleich mit den entsprechenden Verfahren sind im Anhang unter Kapitel D.1.3 dargestellt.

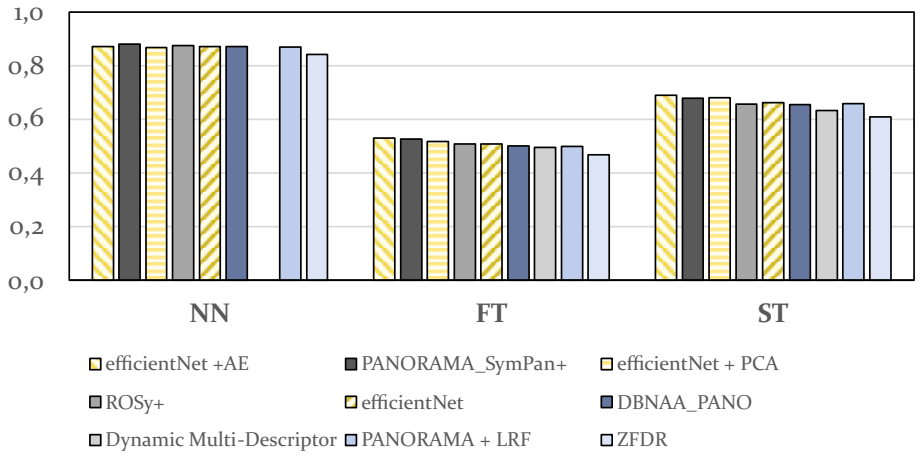


Bild 79: Vergleich mit Stand der Technik Verfahren, Auszug aus Tabelle 13

Dieses Diagramm verdeutlicht, dass für den nearest-neighbour-Wert alle Algorithmen ein relativ hohes Niveau aufweisen und der Ergebniswert nur in einem kleinen Bereich schwankt. Bei der Analyse der für die Anwendung im Maschinenbau relevanten Ergebniswerten FT und ST schneidet das vorgestellte Verfahren (efficientNet + Autoencoder) sehr gut ab. Keine der Methoden konnte bei diesen beiden Bewertungsmetriken bessere Werte erzielen. Auch bei einer Einordnung des neuen Vorgehens ohne die vorher erläuterten Randbedingungen sind nur drei Verfahren mit besseren FT-Werten vorhanden. Auffällig ist auch, dass durch die Anwendung der Dimensionsreduktion eine deutliche Verbesserung im Vergleich zu den etablierten Verfahren erreicht werden konnte.

Zusammenfassend lässt sich sagen, dass die neue Technik im Vergleich zu früher entwickelten Methoden sehr gute Ergebnisse erzielt. Das neue Vorgehen ist definitiv konkurrenzfähig zu den anderen Ansätzen und bietet dennoch Möglichkeiten zur weiteren Verbesserung der Abfrageergebnisse, abhängig von der gestellten Aufgabe.

6.3.2.1 Verbesserung Ähnlichkeitssuche

Die positiven Ergebnisse bilden die Grundlage für mögliche weitere Arbeiten an dieser Methode, da es aufgrund des Methodenumfangs noch viele Stellschrauben gibt. Ein Ansatzpunkt wäre die Verwendung anderer vortrainierter Modelle für die Extraktion von Merkmalsvektoren, welche nicht mit dem ImageNet Datensatz trainiert wurden. Zudem könnten auch andere Deep-Learning-Architekturen, wie Visual Transformer [465–467], für die Featureerzeugung genutzt werden. Darüber hinaus ist es auch notwendig,

die Methoden der Dimensionsreduktion kontinuierlich weiterzuentwickeln und mit neuen Ansätzen zu experimentieren [S10–S12].

Neben der Dimensionsreduktion bieten Verfahren des (Deep)-Metric-Learnings [468] weitere Anknüpfungspunkte für Verbesserungen [S13]. Diese Deep-Learning-Kategorie benötigt zum Training der Modelle keine Klassen-Label, sondern nur die Information, ob die jeweiligen Input-Daten ähnlich oder unähnlich zueinander sind.

Fazit Kapitel 6.3

Der Abgleich der Ergebnisse des neuen Vorgehensmodells mit dem Stand der Technik liefert für die definierten Randbedingungen sehr gute Ergebnisse. Die relevanten Vergleichsgrößen für die Ähnlichkeitssuche sind höher als bei vergleichbaren Methoden und die gesteckten Ziele für das Maschinenbauumfeld konnten eingehalten werden. Zudem ist eine deutliche Verbesserung zur ersten Iteration des Ansatzes erreicht worden.

6.4 Diskussion

Die präsentierten Ergebnisse zeigen, dass die neue Methode im Vergleich zum aktuellen Stand der Technik konkurrenzfähig ist, da unter den vorgestellten Rahmenbedingungen keine Methode bessere Suchergebnisse erzielen konnte. Die Anforderungen der Produktentwicklung an eine Bauteilsuche werden ebenfalls erfüllt, da kein zusätzlicher Input für eine gegebene Bauteilsammlung notwendig ist, weder als Datensatz-Parameter, für das Vorgehen, noch als Bauteil-Label. Somit wird mit Hilfe des Vorgehensmodell die dritte Forschungsfrage erfolgreich beantwortet, da alle gestellten Anforderungen an die Bauteil-Ähnlichkeitssuche durch das entwickelte Vorgehen abgedeckt werden.

Die Auswertung offenbart aber auch, dass die Methode Herausforderungen aufweist, die es noch zu berücksichtigen gilt. Ein Aspekt ist die Abhängigkeit der Ergebnisse von der Komponenten-Ausrichtung, weshalb allerdings auch eine Verbesserung der Ähnlichkeitssuche durch eine einheitliche Ausrichtung möglich ist. Die vorgestellte Ausrichtungsmethode orientiert sich am CAD-Konstruktionsprozess, wodurch das domänenspezifisches Wissen in den gesamten Suchprozess integriert werden konnte. Die Analysen zeigen allerdings auch auf, dass das uneinheitliches Ausrichten der Bauteile zu potentiellen Fehlern oder schlechten Suchergebnissen führt.

Ein weiterer Vorteil des gesamten Vorgehen ist, dass keine Bauteil- bzw. Datensatz-spezifischen Parameter definiert werden müssen. Die notwendigen Parameter für einen erfolgreichen Einsatz sind entweder generell an die Geometrieart (Oberflächennetz) geknüpft, beispielsweise die Punkte-Sampling Zahl oder die Auflösung der Projektionsmethode. Weiterhin können noch die CNN-Modell-Auswahl und der Dimensionsreduktion-Wert einen Einfluss auf die Erkennungsgenauigkeit haben, aber die beiden sind wiederum nicht abhängig von den verwendeten Bauteilen bzw. dem Datensatz.

Die Möglichkeit das Vorgehensmodell auf einen gegebenen Einsatzzweck zu adaptierten, ist ebenfalls ein wichtiger Vorteil der Methode, beispielsweise durch die verfügbaren Algorithmen der Dimensionsreduktion. Ist primär die Suchdauer relevant, beispielsweise wenn im CAD direkt das ähnlichste Bauteil angezeigt werden soll, kann auf die Hauptkomponentenanalyse zurückgegriffen werden. Wenn dagegen die Suchgenauigkeit wichtiger ist, liefert der Autoencoder noch bessere Retrievalergebnisse. Als möglicher Einsatz wäre die Überarbeitung einer bestehenden Baugruppe denkbar, wobei ein Bauteil durch bereits entwickelte Komponenten ersetzt werden soll. Somit sollen mehrere geometrisch ähnliche Bauteile aus dem Datenbestand als Ergebnis aufgeführt werden.

Ein möglicher Nachteil in der Anwendung könnte die fehlende Nachvollziehbarkeit des Vorgehens sein, da CNNs eher als unbekannte Blackboxen wahrgenommen werden und somit die Grundlage für den Vergleich nicht sofort verständlich sein könnte. Allerdings arbeiten viele Verfahren zur geometrischen Ähnlichkeitssuche mit abstrakten Überführungen der Geometrie, weshalb dieses Problem schwierig zu lösen ist. Eben dieser Einsatz der Netzwerke gekoppelt an den modularen Aufbau der Ähnlichkeitssuche ermöglicht eine einfache Aktualisierung des Vorgehens, da neue vortrainierte Modelle ohne großen Aufwand in den bestehenden Ansatz integriert werden können.

Weiterhin könnten noch Methoden der Shape Segmentation [469] für eine zweistufige Analyse der Bauteile genutzt werden. Somit wird zunächst eine globale Ähnlichkeit zwischen Bauteilen ermittelt und im nächsten Schritt die Geometrie anhand ihrer relevanten Bereiche untergliedert. Diese separierten Geometrie-Teilbereiche werden anschließend wiederum auf ihre Ähnlichkeit geprüft. Dadurch ist eine genauere Suche mit spezifischer Zielsetzung möglich, beispielsweise wenn spezielle fertigungsrelevante Bereiche für die Suche mit betrachtet werden sollen.

7 Anwendungsszenario

Nach der ausführlichen Untersuchung, Analyse und Vergleich der einzelnen Module (Erkennung und Suche von Prinzipskizzen, Suche nach Baugruppen mittels Skizzen und Ähnlichkeitssuche von Bauteilen) soll abschließend eine mögliche Anwendung in der Produktentwicklung beispielhaft durchlaufen werden. Dazu werden die in den vorherigen Kapiteln vorgestellten Datensätze, Modelle und Vorgehensweise als Grundlage verwendet und im Kontext einer Konstruktions-Abteilung angewendet. Zuerst werden dementsprechend das Szenario und die Konstruktionsaufgabe kurz vorgestellt und danach der Einsatz der jeweiligen neuen Ansätze und deren Auswirkung gezeigt. Alle nachfolgend präsentierten Ergebnisse sind mit den Modellen und Methoden tatsächlich erzeugt worden, wobei die zu Grunde gelegten Datensätze in den jeweiligen Kapitel erläutert wurden. Im Rahmen dieser Betrachtung wird der Entwicklungsprozess nach der VDI 2221/23 durchlaufen.

7.1 Szenario Vorstellung

Das Szenario basiert darauf, dass ein Kollege in einer Konstruktionsabteilung eines großen Energiekonzerns neu anfängt. Die gestellte Entwicklungsaufgabe für die gesamte Abteilung ist, die Effizienz von Windkraftanlagen weiter zu steigern und somit die Energiewende leichter zu erreichen. Insgesamt gibt es mehrere Unterabteilungen und Fachexperten, grob aufgeteilt in die Bereiche Statik und Auslegung, Elektronik, Aerodynamik und Kraftübertragung. Der neue Kollege wurde für den Fachbereich Kraftübertragung / Getriebe eingestellt und soll sich dort um die Auslegung kümmern. Zunächst erhält das gesamte Team weitere Konkretisierungen durch das Management, welche in Bild 80 aufgelistet sind.

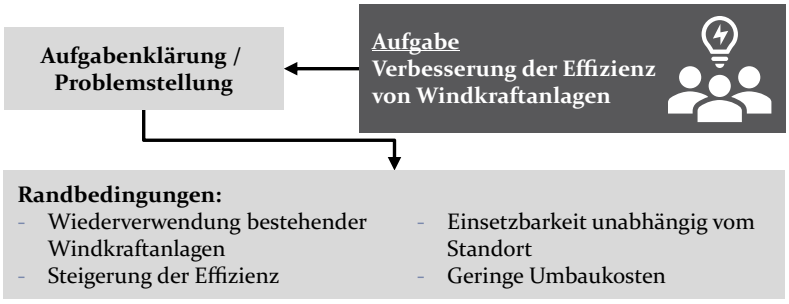


Bild 80: Auflistung der Aufgabe und deren Randbedingungen für das Entwicklungsteam im präsentierten Szenario

Im Fokus steht dabei, die bestehenden Anlagen zu aktualisieren, damit weniger Kosten und Bürokratie entstehen. Die Steigerung der Effizienz sollte so hoch wie möglich sein bei gleichzeitig möglichst geringen Kosten. Zuletzt sollte die entwickelte Lösung sowohl auf Off-Shore-Anlagen als auch bei einzelnen Windrädern auf dem Land umsetzbar sein. Nach der Definition und Erklärung der genauen Aufgabenstellung wird in der gesamten Abteilung ein Brainstorming über mögliche Konzepte durchführt.

7.2 Konzeptphase

In dem offenen Brainstorming sind viele Ideen und Konzepte entwickelt, diskutiert und bewertet worden. Dabei kamen einerseits konventionelle Ansätze hervor, wie beispielsweise größere Rotoren. Aber auch ausgefallenere Lösungen, wie Doppel-Rotoren [470, 471] oder die Archimedes-Schraube [472], sind als mögliche Startpunkte ermittelt worden. Weiterhin dienten auch andere Bereiche wie Turboprob-Triebwerke [473, 474] aus der Luftfahrt als Inspirationsquelle für neue Ideen. Nach ausführlicher Bewertung und Diskussion sind drei Konzepte für die weitere Ausarbeitung ausgewählt worden, welche in Bild 81 illustriert sind.

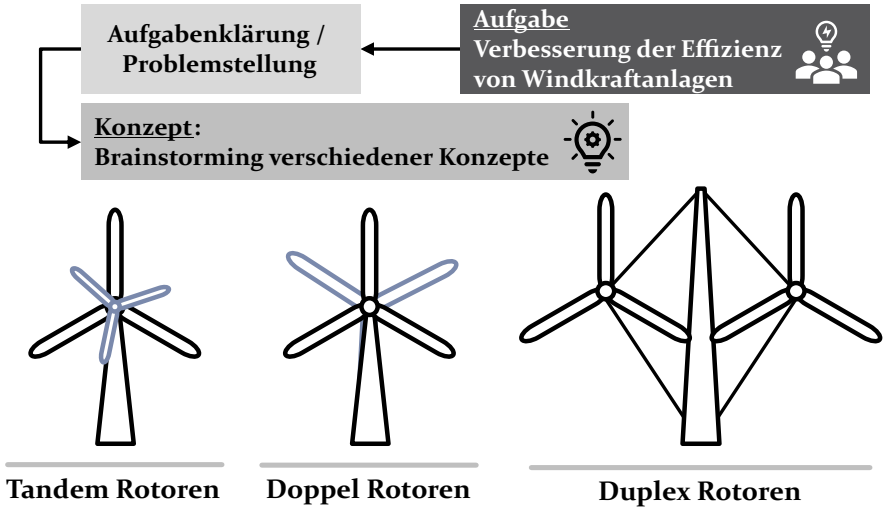


Bild 81: Entwickelte Konzepte für die gestellte Szenario-Aufgabe

Das erste Konzept verwendet zwei voreinander angeordnete gegenläufige Rotoren, abgeleitet von Turboprop-Triebwerken. Der Vorteil davon ist die hohe Wiederverwendung bestehender Strukturen, wodurch Kosten und Umbauzeit reduziert werden. Der Aufwand für das zweite Konzept ist höher

einzuschätzen, da hier ein zweiter Rotor gespiegelt ans Ende der vorhandenen Gondel montiert wird. Im Vergleich zur ersten Lösung könnte die erreichte Effizienz damit verbessert werden, allerdings ist die Auslegung durch die auftretenden Wechselwirkungen deutlich aufwendiger. Der dritte Ansatz platziert zwei komplette Rotoren nebeneinander, was den meisten Umbauaufwand erzeugt, gleichzeitig aber die höchste Leistung für eine vorhandene Stützstruktur verspricht. Ein Nachteil ist der notwendige Platzbedarf, da dieser beispielsweise bei Off-Shore Anlagen nicht zwangsläufig vorhanden ist.

Zum Ende des Termins wurde festgelegt, dass die drei Konzepte im Maßstab 1:100 als erste Prototypen umgesetzt werden sollen, damit diese anschließend im Windkanal auf ihre Leistungsfähigkeit getestet werden können. Dementsprechend erhält der neue Mitarbeiter für die Getriebeabteilung die Aufgabe, diese drei Varianten erst virtuell und anschließend physisch zu entwerfen.

Er beginnt mit dem Konzept der gegenläufigen Tandem-Rotoren, indem er eine erste schnelle Skizze per Hand zeichnet, die die geforderten Funktionalität erfüllt, wie oben links in Bild 82 dargestellt. Dazu müssen zwei gegenläufig drehende Welle zu einer Ausgangswelle kombiniert werden, wodurch die Wahl auf ein Planetengetriebe fällt.

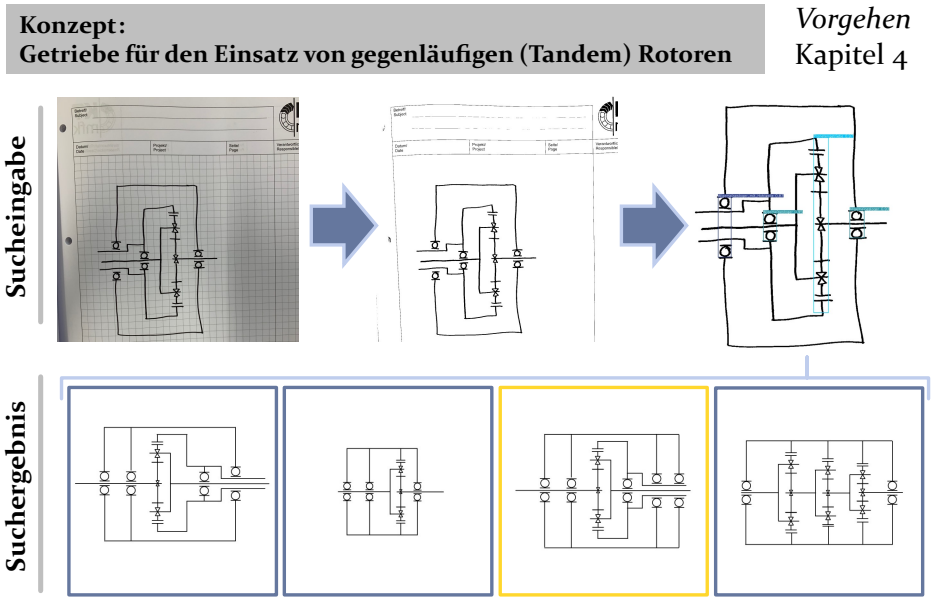


Bild 82: Ergebnis der Prinzipskizzensuche

Nachdem die Skizze gezeichnet ist, nimmt er mit seinem Firmenhandy über eine eigene App ein Bild davon auf und sucht danach ähnliche vorhandene Skizzen im Bestand der Firma. Diese App bereitet das Bild automatisch mit einem Filter (Helligkeitsschwellwert 25 %) und Zuschneide-Operationen auf und übergibt dieses einem Rechenserver mit entsprechendem Symbolerkennungs-Modell, welches automatisch die vorhandenen Symbole erkennt und das Ergebnis in einen Graphen überführt.

Da der Anwender ausgewählt hat, dass er nach ähnlichen Skizzen im Datenbestand suchen möchte, wird der Graph mit den vorhandenen Skizzen bzw. deren Graphen auf ihre Ähnlichkeit abgeglichen. Die vier Ergebnisse werden dem Anwender als Ergebnis am PC angezeigt, wobei er sich für die dritte Variante entscheidet, da diese mit der zweifachen Doppellagerung besser für den gegebenen Anwendungsfall geeignet scheint. Die genau erzielten Ähnlichkeitswerte für die Suche ist im Anhang in Tabelle 23 aufgeführt.

Nach der Skizzenauswahl muss der Bearbeiter aber feststellen, dass keine weiteren Daten oder Geometrien mit dieser Skizze verknüpft sind, da diese aus einem älteren Brainstorming ohne konkrete Umsetzung stammt.

7.3 Entwurfsphase

Allerdings öffnet er jetzt an seinem Rechner das Skizzen-Baugruppen-Suchprogramm, welches auf Grundlage der ausgewählten Skizze passende CAD-Baugruppen aus dem Datenbestand der Firma findet. Er übergibt dieser Anwendung die ausgewählte Skizze und erhält als Ergebnis vier STEP-Geometrien, die er direkt in seinem CAD-Programm anschauen kann. Dieser Vorgang, inklusive der jeweiligen Ergebnisse, zeigt Bild 83.

Der neue Mitarbeiter wählt die vierte Baugruppe aus, da diese seinen Anforderungen mit einer doppelt gelagerten Ausgangs- und Eingangswelle entspricht. Zudem ist diese Baugruppe, im Vergleich zu den anderen drei, sehr detailreich konstruiert, da fast alle Zahnräder komplett dargestellt sind. Außerdem sind weitere Elemente wie Verschraubung oder Sicherungsringe bereits in der Baugruppe inkludiert und müssen nicht manuell nachträglich integriert werden.

Bei der genaueren Analyse fällt dem Produktentwickler auf, dass der Träger der Planeten-Zahnräder in Kollision mit der äußeren Hohlwelle steht. Anstatt händisch nach ähnlichen Lösungen in älteren Projekten zu suchen, verwendet der neue Kollege seine im CAD-System integrierte Bauteil-Ähnlichkeitssuche. Er wählt einfach das relevante Bauteil aus und übergibt die Geometrie dem entsprechenden CAD-Addin. Nach kurzer Suche erscheinen fünf Bauteile

als Suchergebnis in einem extra Fenster im CAD-Programm, veranschaulicht in Bild 84.

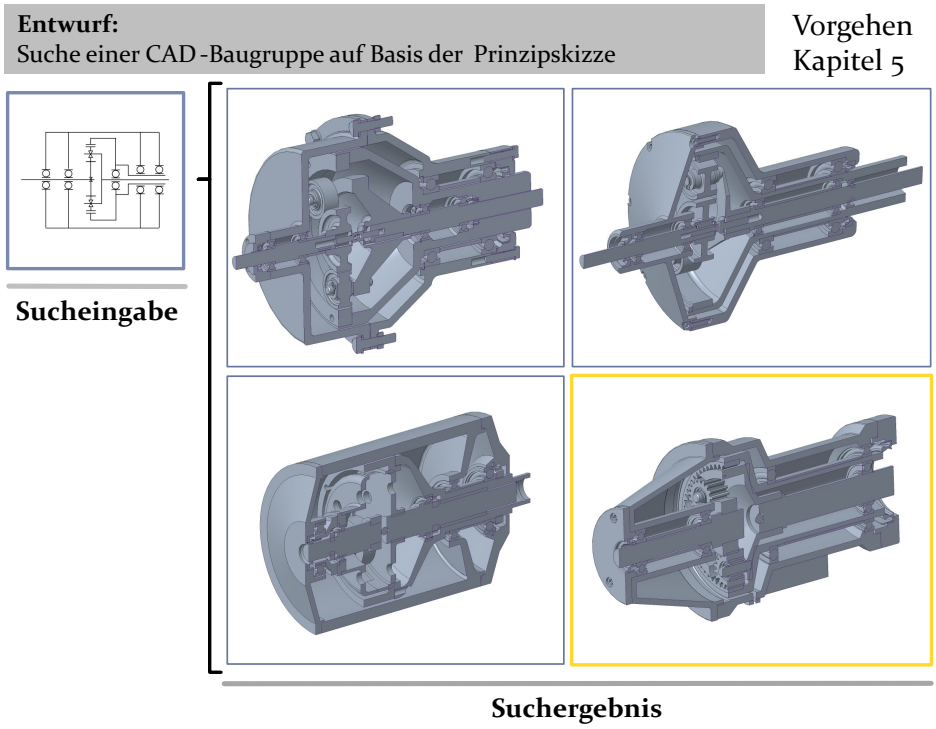


Bild 83: Ergebnis der Baugruppensuche, mit Prinzipskizzen als Sucheingabe

Die gefundenen Bauteile kann der Entwickler zwar nicht direkt übernehmen, allerdings erkennt er, dass in älteren Projekten bereits Planetenträger ohne konisch zulaufende Welle verwendet wurden. Durch diese Inspiration passt er die Geometrie des Trägers und der äußeren Welle händisch an, damit alle Bauteile kollisionsfrei in der Baugruppe angeordnet sind. Das entstandene Ergebnis ist unten in Bild 84 dargestellt.

Zuletzt erkennt er noch den fehlerhaften Lagerdeckel an der Antriebsseite, welchen er aber durch sein erlangtes Wissen im Studium selbstständig in kurzer Zeit überarbeiten kann. Nach der erfolgreichen Anpassung und finalen Überprüfung der Baugruppe übergibt der Konstrukteur die Daten direkt an die additive Fertigung im Prototypenbau und kann sich zügig um die Bearbeitung des zweiten Konzepts kümmern.

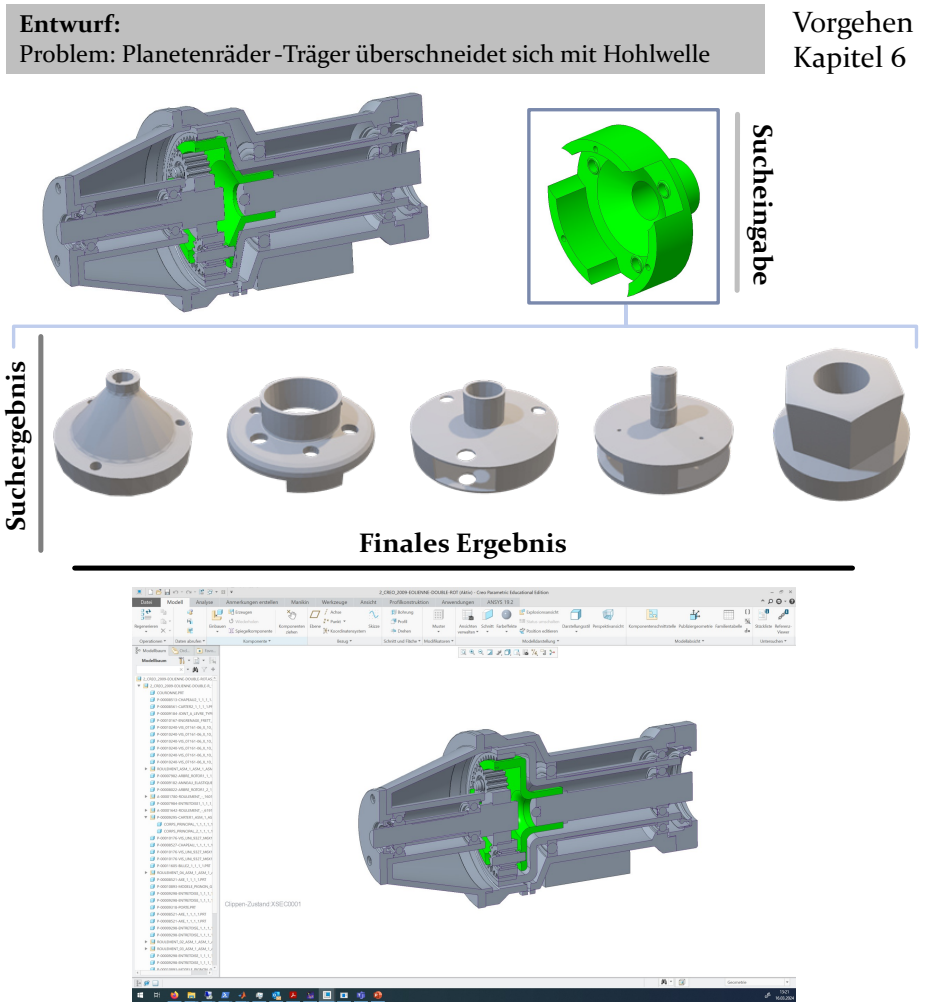


Bild 84: Ergebnis der Bauteil-Ähnlichkeitssuche, mit dem Planetenträger als Sucheingabe

7.4 Diskussion

Das vorgestellte Szenario sollte zeigen, wie der Einsatz der neu entwickelten Vorgehensweise im Produktentwicklungsprozess erfolgen kann. Alle gezeigte Ergebnisse wurden mit den Datensätzen aus den jeweiligen Kapiteln erzielt und demonstrieren mit dieser Stichprobe somit nochmals die Leistungsfähigkeit der neuen Methode. Im beschriebenen Beispiel erfolgt der Einsatz sequentiell und aufeinander aufbauend, allerdings ist durch die modulare Entwicklung keine stringente Kopplung der Verfahren notwendig. Das Sze-

nario könnte auch nur die Bauteilsuche oder Skizzensuche verwenden und die weitere Schritte nicht durchführen.

Die Anwendung der datengetriebenen Methoden in der Produktentwicklung können somit dazu führen, dass der Anwender entlastet wird, wobei er immer noch die Entscheidungsgewalt behält. Somit ist er in der Lage selbständig Wissen aufzubauen und sich weiterzuentwickeln und gleichzeitig auf die vorherigen Ergebnisse schnell und einfach zuzugreifen. Die vorhandenen Daten bleiben somit nutzbar, auch wenn Mitarbeiter das Unternehmen verlassen oder andere Abteilungen an ähnlichen Problemen arbeiten. Der Einsatz dieser neuen Vorgehensmodelle würde somit zu einer schnellen und qualitativ besseren Produktentwicklung, vor allem in den frühen Phasen, führen.

8 Zusammenfassung und Ausblick

In der präsentierten Arbeit ist ein modulares Vorgehensmodell zur Nutzung vorhandener Daten in den frühen Phasen des Produktentwicklungsprozesses vorgestellt worden. Diese Modell mit seinen Modulen ermöglicht die Nutzung vorhandener Skizzen und 3D-Geometrien, ohne zusätzlichen manuellen Input und Aufbereitung. Zur Umsetzung sind datengetriebene Methoden und vor allem Machine- bzw. Deep-Learning-Modelle verwendet worden, welche die Berücksichtigung der Randbedingungen aus der Produktentwicklung ermöglichen.

Die Motivation für den Einsatz liegt in dem Potential der vorhandenen Daten. Die Nutzung dieser bestehenden Daten verspricht eine erhöhte Produktqualität, weniger Risiken in der Entwicklung, sowie geringeren Zeit- und Kostenbedarf. Somit ebnet der Einsatz der vorgestellten Methoden den Weg für eine nachhaltigere Produktentwicklung, da weniger Ressourcen für die Entstehung neuer Produkte aufgebracht werden müssen. Neben dem direkten Wiederverwenden der bestehenden Daten ist auch die Verbesserung des Problemlösungsprozesses ein wichtiger Vorteil der präsentierten Vorgehensmodelle. So müssen nicht direkt die Daten übernommen werden, sondern dienen eher als Inspiration für die aktuelle Problemstellung. Speziell in den frühen Phasen des Entwicklungsprozesses ist der Einsatz besonders vorteilhaft, da hier ein großer Teil der späteren Kosten definiert wird und somit eine höhere Qualität der entstandenen Produkte sich direkt auf die restliche Entwicklung auswirkt.

Allerdings sind die verschiedenen Datenformate in den Konzept- und Entwurfsphasen eine Herausforderung für den Einsatz der Ähnlichkeitssuche. Dementsprechend sind für die, in den frühen Phasen relevanten, Produktrepräsentationen ausführliche Literaturrecherchen durchgeführt worden. Dabei zeigte sich für die Nutzung von Prinzipskizzen ein Problem mit der Datenbereitstellung und weniger mit den verfügbaren Algorithmen, welche bereits ausführlich in der Computer Vision-Domäne erforscht wurden. Dagegen präsentiert die Ähnlichkeitssuche von Geometrien ein entgegengesetztes Bild, da hier Benchmark-Datensätze verfügbar sind, allerdings die generellen Methoden nicht die spezifischen Anforderungen der Produktentwicklung erfüllen. Folglich mussten eigene Vorgehensweisen entwickelt werden, um die Spezifikationen einzuhalten. Für beide Vorgehensweisen gilt, dass die Aufbereitung möglichst gering gehalten werden soll, damit die Einsetzbarkeit so hoch wie möglich ist.

Die somit aufgezeigte Forschungslücke ist mit mehreren Modulen bearbeitet worden. Der Ansatz für die Prinzipskizze ist dabei zweigeteilt und umfasst zunächst die Erkennung der einzelnen Symbole und nachfolgend die Weiterverwendung. Die Herausforderung in der Umsetzung der Symbolerkennung war vor allem die Bereitstellung von ausreichend Daten für das Training eines Deep-Learning-Modells, weshalb eine synthetische Datenerzeugung entwickelt und ausführlich mit unterschiedlichen Ansätzen getestet wurde. Die ermittelten Symbole sind anschließend in einen Graphen transformiert worden, welcher die Grundlage für weitere Anwendungen bildet. Verwirklichte Implementierungen in der Arbeit sind Ähnlichkeitssuchen mit diversen Sucheingaben (Text, Skizzen (Graphen) oder Subgraphen), sowie die Umwandlung in ein SysML-Modell.

Neben der uni-modalen Verwendung der Prinzipskizzen ist im darauffolgenden Kapitel ein Vorgehen präsentiert worden, welches das Finden von Baugruppen auf Basis von Prinzipskizzen ermöglicht. Dazu muss ebenfalls eine Baugruppe in einen Graphen überführt werden, gefolgt von einer Aufbereitung und einem Vergleich der Baugruppen- und Skizzen-Graphen. In den einzelnen Subprozessen des Moduls ist wiederum der Fokus auf geringen zusätzlichen Input gelegt worden, weshalb für die Durchführung der Skizze-zu-Baugruppen-Suche nur die Anwendungsdomänen-spezifischen Regeln für die Aufbereitung notwendig sind. Falls die Bauteilklassen nicht aus den Bauteilnamen der STEP-Datei ausgelesen werden können, ist zusätzlich ein Datensatz für die Erkennung der Bauteile über Deep-Learning-Modelle unerlässlich. Dabei kann auf öffentliche Bestände zurückgegriffen werden oder ein eigener gelabelter Datensatz erzeugt werden. Das gesamte Vorgehen ist mit zwei unterschiedlich aufwendigen Datensätzen an Skizzen und Baugruppen getestet worden, wobei für beide sehr gute Retrieval-Genauigkeiten erzielt werden konnten.

Nach der Verknüpfung über Dateiformate hinweg beschäftigte sich das letzte Kapitel mit der Suche von 3D-Geometrie, wieder als uni-modaler Input. Diese Methode verfolgt das Prinzip der feineren Granularität im Entwicklungsprozess, wodurch der Fein-Entwurf von Bauteilen später angesiedelt ist. Das Ziel war das Finden von ähnlichen Bauteilen aus einem vorhandenen Datenbestand, ohne zusätzliche Aufbereitung oder Daten-Labeling. Die Umsetzung erfolgte über eine Transformation der Bauteile in Matrizen und einen nachgelagerten Vergleich über vortrainierte CNN-Modelle. Zusätzlich sind verschiedene Iterationen zur Verbesserung der Suche getestet und analysiert worden, welche abhängig von der Suchaufgabe eingesetzt werden können. Final ist das Modul über einen Benchmark-Datensatz mit dem aktuell verfügbaren Vorgehen verglichen worden, wobei das neue Vorgehen im Stande ist, bessere Retrieval-Ergebnisse zu erreichen.

Der vorgestellte Ansatz schließt somit die Forschungslücke für die Nutzung vorhandener Prinzipskizzen und Bauteile in den frühen Phasen der Produktentwicklung. Die unterschiedlichen modularen Vorgehensweisen ermöglichen das Finden und somit Wiederverwenden vorhandener Daten, wodurch Zeit, Ressourcen und final Kosten eingespart werden. Alle präsentierten Ansätze verfolgen das Ziel, möglichst breit anwendbar zu sein und ohne zusätzlichen händischen Input auszukommen.

Trotz der vielversprechenden Ergebnisse für die einzelnen Module sind noch weitere Entwicklungen und Verbesserungen möglich. Die Symbolerkennung könnte noch weiter auf die Erkennung von handgezeichneten Symbolen getestet werden, indem zunächst ein ausführlicher Datensatz mit mehreren Probanden erzeugt wird. Dieser bildet danach die Grundlage für weitere Anpassungen am Vorgehen, damit die Handskizzen besser umgewandelt und nachfolgend erkannt werden können. Weiterhin kann der erzeugte Graph aus den Baugruppen bzw. Skizzen weiterverwendet und zusätzlich mit Informationen über die Baugruppen (z.B. Zahnrad-Zähnezahl oder Bauteil-Material) angereichert werden, um weitere Suchmöglichkeiten zu generieren. Zusätzlich wäre eine Verknüpfung mit dem Vorgehen von Lupinetti et al. [276] denkbar, wodurch die Suche von Baugruppen als Eingabe ebenfalls in das Vorgehen integriert ist. Weiterhin bietet der Graph als vereinendes Element vielseitige Anknüpfungsoptionen, beispielsweise durch Large Language Models (LLM) [475–477]. Dadurch wäre eine Art Konstruktions-Bot denkbar, der die Textbefehle direkt verarbeitet und so bei der Entwicklung neuer Produkte unterstützen kann. Darüber hinaus könnten Ansätze die Design-Synthese von neuen Produkten umfassen, das Erkennen von Fehlern im Aufbau oder fehlenden Elementen in der Baugruppe, jeweils mit der Graph-Beschreibung als Grundbaustein. Im Bezug auf die Auswirkung der entwickelten Vorgehen auf die industrielle Anwendung sollte eine ausgiebige Nutzerstudie durchgeführt werden, welche die Auswirkungen und Änderungen des Arbeitsalltags durch die neuen Ansätze untersucht.

9 Summary and Outlook

This thesis presents a modular procedure for utilising existing data in the early phases of the product development process. These methods enable the utilization of existing sketches and 3D geometries without additional manual input and preparation. Data-driven methods and, above all, machine and deep learning models were used for implementation, which allow the constraints from product development to be considered.

The motivation for its application is the potential of existing data. The usage of this available data promises increased product quality, fewer risks in development, and lower time and cost requirements. The application of the presented methods thus facilitates more sustainable product development, as fewer resources are required for the creation of new products. In addition to the direct reuse of existing data, the improvement of the problem-solving process is also an important advantage of the presented method. The existing data does not have to be adopted directly, but rather serves as inspiration for the current problem. Their deployment is particularly advantageous in the early phases of the development process, as this is where a large proportion of the subsequent costs are defined and the higher quality of the resulting products has a direct impact on the rest of the development process.

However, the different data formats in the concept and design phase are a challenge for the implementation of a similarity search. Accordingly, extensive literature research was carried out for the product representations relevant in the early phases. This revealed a problem with the availability of data for the utilization of principle sketches and less with the available algorithms, which were already extensively researched in the computer vision domain. In contrast, the similarity search of geometry presents the opposite picture, as benchmark data sets are available here, but the general methods do not fulfill the specific requirements of product development. As a result, separate procedures must be developed in order to comply with the specifications. For both approaches, the preparation should be kept to a minimum to maximize usability.

The resulting research gap was addressed using several process models. The approach for the principle sketch is divided into two parts and initially comprises the recognition of the individual symbols and then their further use. The main challenge in the implementation of symbol recognition was the provision of sufficient data for training a deep learning model, for which reason synthetic data generation was developed and extensively tested with different approaches. The identified symbols were then transformed into a

graph, which forms the basis for further applications. Realized implementations in the thesis are similarity searches with various search inputs (text, sketches (graphs) or subgraphs), as well as the transformation into a SysML model.

In addition to the uni-modal utilization of principle sketches, the fifth chapter presented a procedure that enables assemblies to be found on the basis of principle sketches. For this purpose, an assembly must be converted into a graph, followed by preparation and matching of the assembly and sketch graphs. In the individual sub-processes of the method, the focus was placed on low additional input, which is why only the application-domain-specific rules for preparation are required to perform the sketch-to-assembly search. If the component classes cannot be read from the component names of the STEP files, an additional data set is essential for recognizing the components using deep learning models. This can be based on public data sets or a separate labeled data set must be created. The entire process was tested with two differently complex data sets of sketches and assemblies, with very good retrieval accuracies being achieved for both.

After linking across file formats, the last chapter dealt with the search for 3D geometry, again as uni-modal input. This method follows the principle of finer granularity in the development process, which places the detailed design of components at a later stage. The aim was to find similar components from an existing database without additional preparation or data labeling. The implementation was carried out by transforming the components into matrices and a downstream comparison using pre-trained CNN models. In addition, various iterations to improve the search were tested and analyzed, which can be applied depending on the search task. Finally, the procedure model was compared with the currently available procedure using a benchmark data set, whereby the new procedure is able to achieve better retrieval results.

The approach presented therefore closes the research gap for the use of existing principle sketches and components in the early phases of product development. The different modular approaches enable finding and reusing existing data, saving time, resources and ultimately costs. All of the presented approaches aim to be as widely applicable as possible and to function without additional manual input.

Despite the promising results for the individual modules, further development and improvements are still feasible. Symbol recognition could be further tested for the recognition of hand-drawn symbols by first generating a detailed data set with several test participants. This then forms the basis for further adjustments to the procedure so that the hand-drawn sketches can be better converted and subsequently recognized. Furthermore, the graph

generated from the assemblies or sketches can be reused and additionally enriched with information about the assemblies (e.g. number of gear teeth or component material) to generate further search options. In addition, a link with the procedure of Lupinetti et al. [276] would be conceivable, whereby the search of assemblies as input is also integrated in the procedure. Moreover, the graph as a unifying element offers versatile linking options, for example through Large Language Models (LLM) [475–477]. This would make a kind of design bot imaginable that processes the text commands directly and can thus support the development of new products. Beyond this, approaches could include the design synthesis of new products or the detection of errors in the structure or missing elements in the assembly, in each case with the graph description as the basic building block. With regard to the impact of the developed approaches on industrial application, an extensive user study should be carried out to investigate the effects and changes to everyday working life as a result of the new approaches.

Anhang

A Ergänzungen zu Kapitel 2

A.1 Auswertung Objekterkennung - Beispiel

Beispiel Auswertung

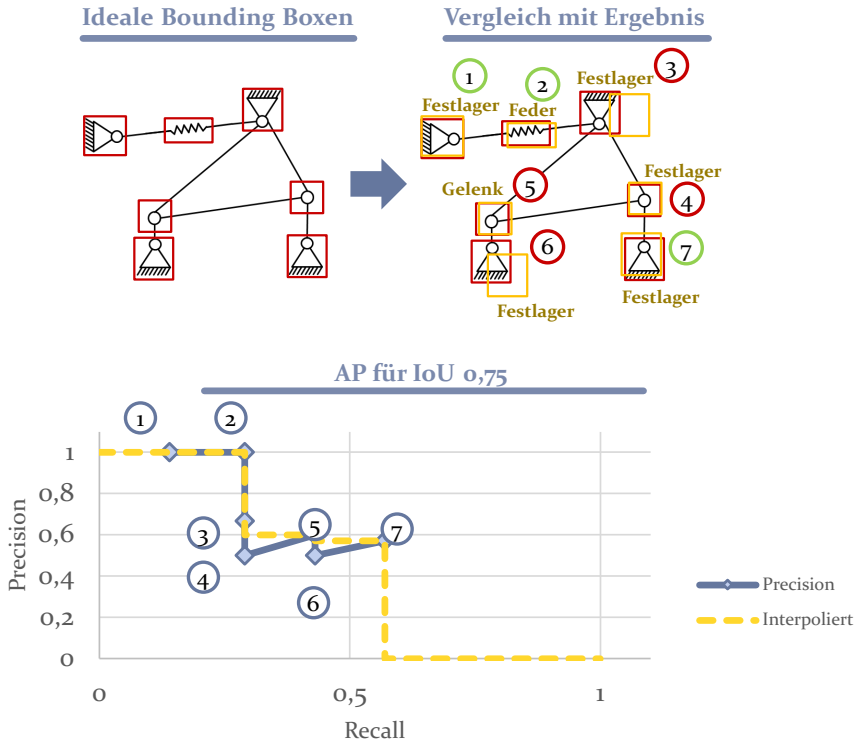


Bild 85: Beispiel für die Auswertung der AP für einen IoU von 0.75

In Bild 85 ist ein einfaches Auswertungsbeispiel dargestellt. Oben links befinden sich die idealen Bounding Boxen und rechts daneben das Ergebnis der Objekterkennung. Die Symbole können dabei richtig erkannt und innerhalb der IoU Grenze liegen, z.B. 1, 2, 5 und 7. Falls sich das detektierte Ergebnis außerhalb der Grenze befindet, wird es als fehlerhaft eingestuft, beispielsweise an Stelle 3 und 6. Natürlich werden falsch erkannte Symbole als nicht

korrekt in die Auswertung übertragen, in diesem Beispiel gilt dies für die vierte Auswerteposition. Die Ergebnisse sind in das P/R-Diagramm im unteren Teil des Bildes übertragen worden, wobei die zugehörigen Symbole und Punkte im Diagramm mittels Nummer gekennzeichnet sind. Da der Ablauf der Symbolauswertung einen Einfluss auf das Ergebnis hat, erfolgt dieser auf Basis der Sicherheit der Symbolerkennung. Dieser Wert wird vom Modell ermittelt und gibt die Wahrscheinlichkeit an, wie sicher sich das Modell mit der zugeordneten Klasse ist. Für dieses einfache Beispiel kann auch die AP (mit 11-Interpolationsschritten) berechnet werden, durchgeführt in Formel 18:

$$mAP75_{11} = \frac{1}{11}(1 + 1 + 1 + 0,6 + 0,6 + 0,57) = 0,434 \tag{18}$$

A.2 Auswertung Ähnlichkeitssuche - Beispiel

Beispiel Auswertung

Das Bild zeigt dabei ein mögliches Retrievalergebnis, bei dem die blauen Daten ein richtiges bzw. passendes Ergebnis zeigen und die weißen ein nicht relevantes. Die Reihenfolge ist anhand des Ähnlichkeitswertes sortiert, wodurch sich die Reihenfolge ergibt. Mittels dieser Einordnungen können die entsprechenden Auswertemetriken NN, FT und ST berechnet werden. Die zugehörigen Ergebnisse sind direkt im Bild mit angegeben. Dabei gilt die Annahme, dass insgesamt fünf relevante Dokumente im gesamten Datensatz vorliegen.

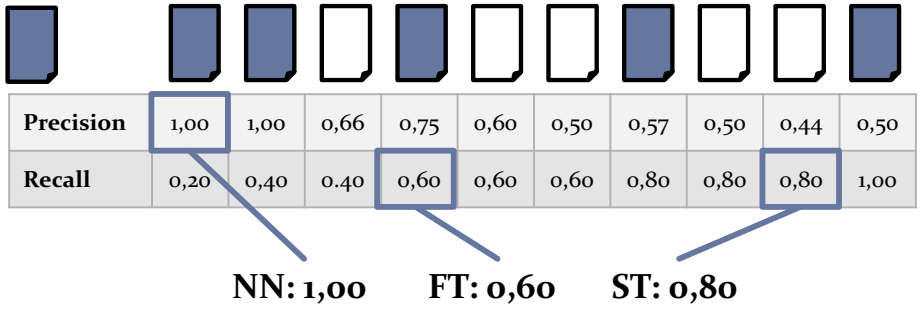


Bild 86: Beispiel für die Auswertung der Ähnlichkeitssuche

A.3 Convolutional Neural Network - Erklärung

Der allgemeine Aufbau für ein Convolutional Neural Network (CNN) ist in Bild 87 dargestellt, mit den relevanten Convolutional- und Pooling-Schichten.

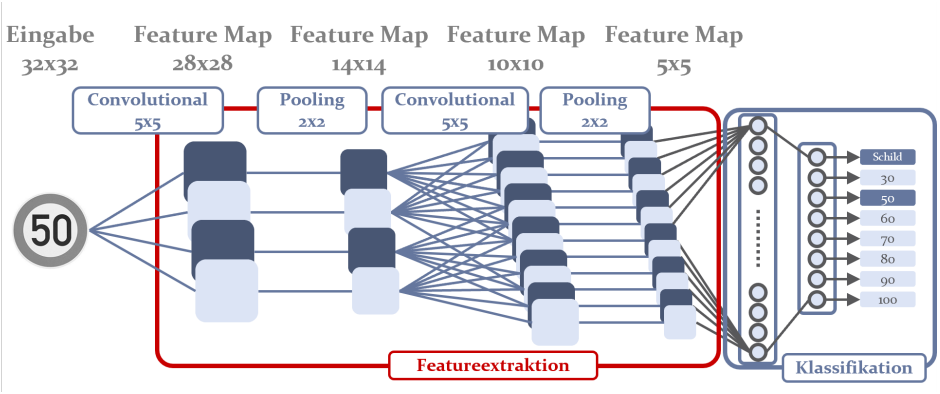


Bild 87: Beispieldarstellung für die Anwendung von CNNs

Convolutional

Die Convolutional-Schicht ist die wichtigste Operation des CNNs und deshalb auch für den Namen des Verfahrens verantwortlich. Der generelle Ablauf ist in Bild 88 abgebildet, wobei durch mehrere Filter eine Erweiterung der Eingabe-Matrix erfolgt. Der Filter ist ebenfalls eine dreidimensionale Matrix , die in der Größe und Anzahl an Kanälen variieren kann. Die Gewichte der Filter werden durch das Trainieren des Modells mittels eines Backpropagation-Algorithmus erzeugt. Das Endergebnis der Convolutional-Schicht wird als Feature Map bezeichnet und den nachfolgenden Operationen übergeben [136, 478, 479].

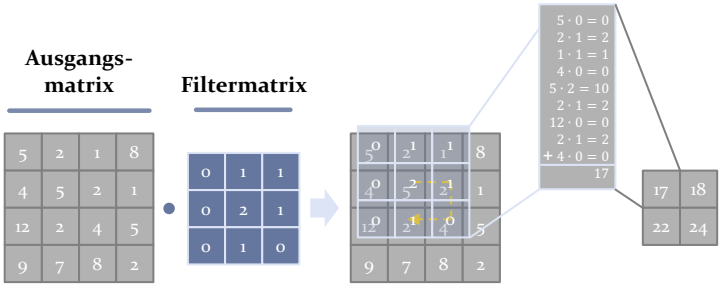


Bild 88: Beispielberechnung der Convolutional Schicht mit einem 3x3 Filter

Pooling

Nach dem Durchlaufen der Convolutional-Schicht ist häufig eine Pooling-

Schicht angeordnet, welche die Aufgabe hat, die Matrizen zu verkleinern und somit überflüssige Informationen zu eliminieren. Für die Umsetzung gibt es zahlreiche Verfahren, wobei die Variante des Max-Pooling sehr häufig verwendet wird. Bei dieser Art des Poolings wird aus einem definierten Bereich der Feature Map nur der Maximalwert für weitere Berechnungsschritte beibehalten. Dieses Vorgehen ist in Bild 89 exemplarisch dargestellt. Eine Alternative zum Max-Pooling stellt das Mean-Pooling dar, bei dem das Pooling-Ergebnis über das arithmetische Mittel bestimmt wird, anstatt nur den Maximalwert zu verwenden [478–480].

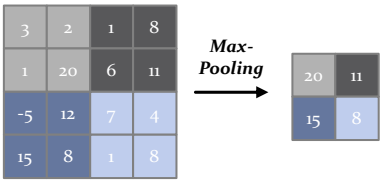


Bild 89: Ergebnis einer Max-Pooling-Operation für einer 4x4 Matrix

Die Kombination der beiden Schichten (Convolutional und Pooling) wird als Featureextraktion bezeichnet und kann mehrmals hintereinander wiederholt werden.

Klassifikation

Die vorbereiteten Feature Maps werden dann einem Neuronalen Netz übergeben. Dieses wird im Rahmen der CNNs oftmals als Fully-Connected-Schicht bezeichnet und ist für die finale Klassifizierung der Feature Map zuständig. Der entsprechende Abschnitt im CNN ist in Bild 87 blau umrandet. Dabei sind die jeweiligen Schichten vollständig mit den Aktivierungen der vorherigen Schichten vernetzt [478]. Bei einer Klassifikation entspricht im Regelfall die Anzahl der finalen Neuronen der Anzahl an verfügbaren Objekt-Klassen. Um eine bessere Aussage über die Klassifikationswahrscheinlichkeit für jede Objektklasse treffen zu können, wird die letzte Schicht in der Regel mit einer Softmax-Funktion normiert.

A.4 Datensätze - Übersicht

A.4.1 MCB

Tabelle 8: Sammlung aller Bauteilklassen des MCB-Datensatzes und der jeweiligen Bauteilanzahl nach [218]

Klasse	Nr.	Klasse	Nr.	Klasse	Nr.
Articulations eyelets & joints	1.632	Impeller	145	Socket	858
Bearing accessories	107	Keys and keyways, splines	4.936	Spacers	113
Bushes	764	Knob	644	Split pins	472
Cap nuts	225	Lever	1.032	Spring washers	55
Castle nuts	226	Locating pins	55	Springs	328
Castor	99	Locknuts	254	Square	72
Chain drives	100	Lockwashers	434	Square nuts	53
Clamps	155	Nozzle	154	Standard fittings	764
Collars	52	Plain guidings	49	Studs	4.089
Conventional rivets	3.806	Plates circulate plates	365	Switch	173
Convex washers	91	Plugs	169	T-nut	101
Cylindrical pins	1.895	Pulleys	121	T-shape fitting	338
Elbow fitting	383	Radial contact ball bearings	1.199	Taper pins	1.795
Eye screws	1.131	Right angular gearings	60	Tapping screws	2.182
Fan	213	Right spur gears	430	Threaded rods	1.022
Flange nut	53	Rivet nut	51	Thrust washers	2.333
Flanged block bearing	404	Roll pins	1.597	Toothed	47
Flanged plain bearings	110	Screws and bolts with countersunk head	2.452	Turbine	85
Grooved pins	2.245	Screws and bolts with cylindrical head	3.656	Valve	94
Helical geared motors	732	Screws and bolts with hexagonal head	7.058	Washer bolt	912
Hexagonal nuts	1.039	Setscrew	1.334	Wheel	243
Hinge	54	Slotted nuts	78	Wingnuts	50
Hook	119	Snap rings	609	**Gesamt**	**58.696**

A.4.2 ESB

Tabelle 9: Sammlung aller Bauteilklassen des ESB-Datensatzes und der jeweiligen Bauteilanzahl nach [220]

Flat-thin components	Nr.	Rectangular-cubic prism	Nr.	Solids of revolution	Nr.
Backdoors	7	Bearing blocks	7	90 degree elbows	41
Bracket like parts	18	Contoured surfaces	5	Bearing like parts	20
Clips	4	Handles	18	Bolt like parts	53
Contact switches	8	L Blocks	7	Container like parts	10
Curved housings	9	Long machine elements	15	Cylindrical parts	43
Miscellaneous	12	Machined blocks	9	Discs	51
Rectangular housings	14	Machined plates	49	Flange like parts	15
Slender thin plates	12	Miscellaneous	21	Gear like parts	36
Thin plates	23	Motor bodies	7	Intersecting pipes	9
-	-	Prismatic stock	36	Long pins	58
-	-	Rocker arms	10	Miscellaneous	33
-	-	Slender links	13	Non-90 degree elbows	8
-	-	Small machined blocks	12	Nuts	19
-	-	T shaped parts	15	Oil Pans	8
-	-	Thick plates	12	Posts	11
-	-	Thick slotted plates	20	Pulley like parts	12
-	-	U shaped parts	25	Round change at end	21
-	-	-	-	Simple pipes	16
-	-	-	-	Spoked wheels	15

A.5 Auswertemetriken Klassifikation - Übersicht

Tabelle 10: Übersicht der Auswertemetriken für Klassifikationsverfahren

Name	**Formel**	**Erklärung**
Positivität	$P = TP + FN$	Anzahl der positiven Objekte
Negativität	$N = TN + FP$	Anzahl der negativen Objekte
Genauigkeit	$G = \frac{TP+TN}{P+N}$	Prozentsatz der richtig klassifizierten Objekte
Fehlerrate	$FR = \frac{FP+FN}{P+N}$	Prozentsatz der falsch klassifizierten Objekte
Sensitivität, True-Positive-Rate (TPR)	$TPR = \frac{TP}{TP+FN}$	Prozentsatz der positiven Objekte, die richtig klassifiziert wurden
Ausfallrate, False-Positive-Rate (FPR)	$TPR = \frac{FP}{FP+TN}$	Prozentsatz der negativen Objekte, die falsch klassifiziert wurden
Spezifität	$SE = \frac{TN}{N}$	Prozentsatz der negativen Objekte, die richtig klassifiziert wurden
Präzision	$PR = \frac{TP}{TP+FP}$	Prozentsatz der positiv klassifizierten Objekte, die positiv sind
F-Maß	$F = 2 \cdot \frac{PR \cdot TPR}{PR+TPR}$	Kombiniertes Maß aus Präzision und Sensitivität mittels des gewichteten harmonischen Mittels
F_α Maß	$F_\alpha = \frac{(1+\alpha)^2 \cdot (PR \cdot TPR)}{\alpha^2 \cdot PR+TPR}$	Basiert auf dem F-Maß, allerdings mit anderer Gewichtung

A.6 Auswertemetriken Klassifikation - ROC Diagramm

Ein weiteres Hilfsmittel, um die Qualität eines Klassifikators besser einschätzen zu können ist die Receiver-Operating-Characteristic-Curve (ROC). In diesem Diagramm werden die True-Positive-Rate über die False-Positive-Rate aufgetragen. Ein Beispiel hierfür ist in Bild 90 abgebildet [481, 482].

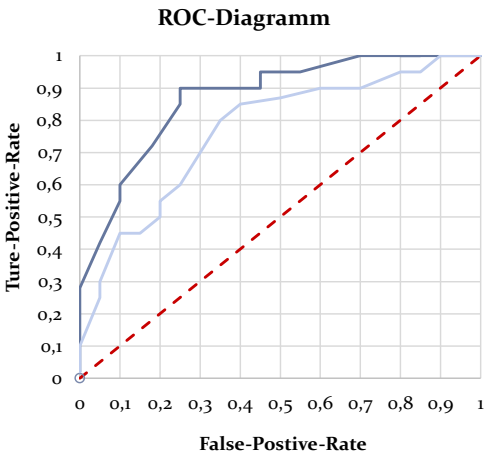


Bild 90: Das ROC-Diagramm zur Beurteilung der Klassifikationsergebnisse nach [129, 233, 234, 481]

Zum Vergleich zwischen verschiedenen ROC-Kurven wird häufig die AUC (Area under ROC Curve) herangezogen [483]. Dieser Wert variiert zwischen 0 und 1 und liefert beim zufälligen Raten das Ergebnis 0,5.

A.7 MCB Datensatz - Vergleichsergebnisse

Tabelle 11: Genauigkeit und F1-Score für den MCB-A-Datensatz nach [218]

Vorgehen	Genauigkeit	F1-Score
PointCNN	93,89	83,86
PointNet++	87,45	74,59
SpiderCNN	87,45	81,30
MVCNN	64,67	69,69
RotationNet	97,35	92,29
DLAN	93,53	83,81
VRN	93,17	81,48

A.8 Ähnlichkeitssuche - Literaturübersicht

Tabelle 12: Liste aller eingeordneten Methoden zur Ähnlichkeitssuche von Bauteilen

Jahr	Autor	Titel	Input Kategorie	Learning Art	DOI
1989	[484]	Recognition and shape synthesis of 3D objects based on attributed hypergraphs	Graph	Kein Training	/10.1109/34.21797
1989	[258]	Extended gaussian images	Geometrie	Kein Training	/10.1109/PROC.1984.13073
1992	[485]	Measuring shapes by size functions	Graph	Kein Training	/10.1117/12.57059
1996	[486]	On 3D shape similarity	Geometrie	Kein Training	/10.1109/cvpr.1996.517122
1998	[487]	3D object retrieval based on subjective measures	Geometrie	Label notwendig	10.1109/dexa.1998.707504
1999	[488]	Using spin images for efficient object recognition in cluttered 3d scenes	Geometrie	Kein Training	/10.1109/34.765655
1999	[257]	3D shape histograms for similarity search and classification in spatial databases	Geometrie	Kein Training	/10.1007/3-540-48482-5_14
2000	[489]	Topological persistence and simplification	Graph	Kein Training	/10.1007/s00454-002-2885-2
2000	[490]	Description of shape information for 2-D and 3-D objects	Geometrie	Kein Training	/10.1016/S0923-5965(00)00020-5
2000	[491]	Nefertiti: A tool for 3-d shape databases management	Hybrid	Kein Training	/10.4271/1999-01-1891
2001	[492]	Tools for 3d-object retrieval: Karhunen-loeve transform and spherical harmonics	Geometrie	Kein Training	/10.1109/MMSP.2001.962749
2001	[493]	3D-shape-based retrieval within the MPEG-7 framework	Geometrie	Kein Training	/10.1117/12.424969
2001	[261]	Topology matching for fully automatic similarity estimation of 3D shapes	Graph	Kein Training	/10.1145/383259.383282
2001	[494]	A geometric approach to 3D object comparison	Geometrie	Kein Training	/10.1109/sma.2001.923387
2002	[495]	Shape distributions	Geometrie	Kein Training	/10.1145/571647.571648
2002	[300]	Using shape distributions to compare solid models	Geometrie	Kein Training	/10.1145/566282.566322
2002	[496]	Shape-based retrieval of 3D mesh models	Geometrie	Kein Training	/10.1109/icme.2002.1035812
2002	[497]	Content Based Retrieval of VRML Objects — An Iterative and Interactive Approach	Geometrie	Label notwendig	/10.1007/978-3-7091-6103-6_12
2002	[498]	3d models retrieval by using characteristic views	Bild	Kein Training	/10.1109/ICPR.2002.1048337
2003	[499]	Rotation invariant spherical harmonic representation of 3D shape descriptors	Geometrie	Kein Training	/10.2312/SGP/SGP03/156-165
2003	[262]	On visual similarity based 3D model retrieval	Bild	Kein Training	/10.1111/1467-8659.00669
2003	[500]	3D shape matching with 3D shape contexts	Geometrie	Kein Training	-
2003	[260]	Skeleton based shape matching and retrieval	Graph	Kein Training	/10.1109/SMI.2003.1199609
2003	[501]	Analyzing appearance and contour based methods for object categorization	Hybrid	Kein Training	/10.1109/CVPR.2003.1211497
2003	[502]	An improvement of rotation invariant 3D-shape descriptor based on functions on concentric spheres	Geometrie	Kein Training	/10.1109/ICIP.2003.1247355
2003	[503]	Polyhedral model retrieval using weighted point sets	Geometrie	Kein Training	/10.1142/S021946780300097X
2003	[504]	A search engine for 3-D models	Geometrie	Kein Training	/10.1145/588272.588279
2003	[505]	Retrieving 3D shapes based on their appearance	Geometrie	Kein Training	/10.1145/973264.973272

2003	[506]	Reeb graph based shape retrieval for CAD	Graph	Kein Training	/10.1115/DETC2003/CIE-48194
2004	[507]	Recognizing objects in range data using regional point descriptors	Geometrie	Kein Training	/10.1007/978-3-540-24672-5_18
2004	[268]	Model Retrieval	Hybrid	Kein Training	-
2004	[508]	Using entropy impurity for improved 3D object similarity search	Hybrid	Label notwendig	/10.1109/ICME.2004.1394465
2004	[509]	A multi-scale hierarchical 3D shape representation for similar shape retrieval	Graph	Kein Training	/10.1109/TIP.2021.3082765
2005	[510]	Shape-Similarity Search of 3D Models by using Enhanced Shape Functions	Geometrie	Kein Training	/10.1109/tpcg.2003.1206936
2005	[511]	Desire: a composite 3D-shape descriptor	Hybrid	Kein Training	/10.1109/ICME.2005.1521584
2005	[512]	3D shape context based gesture analysis integrated with tracking using omni video array	Geometrie	Kein Training	/10.1109/CVPR.2005.382
2005	[513]	Semantic-oriented 3D shape retrieval using relevance feedback	Hybrid	Label notwendig	/10.1007/s00371-005-0341-z
2005	[514]	SVM-based semantic clustering and retrieval of a 3D model database	Hybrid	Label notwendig	/10.1080/16864360.2005.10738363
2006	[515]	Local feature extraction and matching partial objects	Graph	Kein Training	/10.1016/j.cad.2006.07.005
2006	[516]	Enhanced 2D/3D Approaches Based on Relevance Index for 3D-Shape Retrieval	Bild	Kein Training	/10.1109/smi.2006.11
2006	[517]	Spherical Wavelet Descriptors for Content-based 3D Model Retrieval	Hybrid	Kein Training	/10.1109/smi.2006.39
2006	[518]	PTK: A novel depth buffer-based shape descriptor for three-dimensional object retrieval	Bild	Kein Training	/10.1007/s00371-006-0037-z
2006	[519]	A new 3d model retrieval approach based on the elevation descriptor	Bild	Kein Training	/10.1016/j.patcog.2006.04.034
2007	[520]	3D model retrieval using principal plane analysis and dynamic programming	Geometrie	Kein Training	/10.1016/j.patcog.2006.06.006
2007	[521]	Using Enhanced Shape Distributions to Compare CAD Models	Geometrie	Kein Training	/10.1007/978-3-540-77255-2_41
2007	[522]	Laplace-beltrami eigenfunctions for deformation invariant shape representation	Geometrie	Kein Training	/10.2312/SGP/SGP07/225-233
2007	[523]	A spectral approach to shape-based retrieval of articulated 3D model	Geometrie	Kein Training	/10.1016/j.cad.2007.02.009
2007	[524]	A new descriptor for 2D depth image indexing and 3D model retrieval	Bild	Kein Training	/10.1109/ICIP.2007.4379599
2007	[525]	3D shape retrieval integrated with classification information	Geometrie	Label notwendig	/10.1109/ICIG.2007.13
2007	[526]	Learning semantic categories for 3D model retrieval	Geometrie	Label notwendig	/10.1145/1290082.1290090
2007	[527]	A boosting approach to content-based 3D model retrieval	Bild	Label notwendig	/10.1145/1321261.1321301
2007	[528]	3D classification via structural prototypes	Graph	Label notwendig	/10.1007/978-3-540-77051-0_16
2007	[529]	A Bayesian 3-D search engine using adaptive views clustering	Bild	Unüberwachtes Lernen	/10.1109/TMM.2006.886359
2007	[530]	Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation	Geometrie	Kein Training	/10.1016/j.patcog.2006.12.026
2007	[531]	Mesh Retrieval by Components	Graph	Kein Training	/10.1007/978-3-540-75274-5_3
2008	[532]	Retrieving articulated 3-D models using medial surfaces	Graph	Kein Training	/10.1007/s00138-007-0097-8
2008	[533]	3D Object Retrieval using an Efficient and Compact Hybrid Shape Descriptor	Hybrid	Kein Training	/10.2312/3DOR/3DOR08/009-016
2008	[534]	Characterizing shape using conformal factors	Geometrie	Kein Training	/10.2312/3DOR/3DOR08/001-008
2008	[535]	Salient local visual features for shape-based 3D model retrieval	Bild	Kein Training	/10.1109/SMI.2008.4547955

2008	[536]	Supervised learning of salient 2D views of 3D models	Bild	Label notwendig	/10.3756/artsci.7.124
2008	[537]	Supervised learning of similarity measures for content-based 3D model retrieval	Geometrie	Label notwendig	/10.1007/978-3-540-78159-2_20
2008	[538]	SHREC'08 entry: semi-supervised learning for semantic 3D model retrieval	Geometrie	Label notwendig	/10.1109/SMI.2008.4547987
2008	[539]	A powerful relevance feedback mechanism for content-based 3-D model retrieval	Geometrie	Label notwendig	/10.1007/s11042-007-0188-6
2008	[540]	Accelerating Bag-of-Features SIFT Algorithm for 3D Model Retrieval	Bild	Unüberwachtes Lernen	-
2008	[541]	Ranking on Semantic Manifold for Semantic 3D Model Retrieval	Bild	Label notwendig	/10.1145/1460096.1460163
2008	[542]	SHREC'08 Entry: Local 2D Visual Features for CAD Model Retrieval	Bild	Unüberwachtes Lernen	/10.1109/SMI.2008.4547985
2008	[543]	Relevance Feedback in Content-based 3D Object Retrieval A Comparative Study	Geometrie	Label notwendig	/10.3722/cadaps.2008.753-763
2008	[544]	Similarity Score Fusion by Ranking Risk Minimization for 3D Object Retrieval	Geometrie	Label notwendig	-
2008	[545]	Spatially Enhanced Bags of Words for 3D Shape Retrieval	Hybrid	Unüberwachtes Lernen	/10.1007/978-3-540-89639-5_34
2009	[546]	Visual vocabulary signature for 3D object retrieval and partial matching	Geometrie	Unüberwachtes Lernen	-
2009	[547]	Multi-Fourier spectra descriptor and augmentation with spectral clustering for 3D shape retrieval	Geometrie	Unüberwachtes Lernen	/10.1007/s00371-008-0304-2
2009	[548]	3D Mechanical Models Retrieval Based on Combined Histograms for Rapid Product Design	Geometrie	Kein Training	/10.4028/www.scientific.net/AMM.16-19.65
2009	[549]	Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features	Bild	Unüberwachtes Lernen	/10.1145/1646396.1646430
2009	[550]	3D Model Retrieval Using Probability Density-Based Shape Descriptors	Geometrie	Kein Training	/10.1109/TPAMI.2009.25
2009	[551]	A concise and provably informative multi-scale signature based on heat diffusion	Hybrid	Kein Training	/10.1111/j.1467-8659.2009.01515.x
2009	[552]	Topology and data	Graph	Kein Training	/10.1090/S0273-0979-09-01249-X
2009	[553]	3d object retrieval using the 3d shape impact descriptor	Geometrie	Kein Training	/10.1016/j.patcog.2009.04.024
2009	[554]	Weighted subspace distance and its applications to object recognition and retrieval with image sets	Bild	Kein Training	/10.1109/LSP.2008.2010819
2009	[555]	Scale-Weighted Dense Bag of Visual Features for 3D Model Retrieval from a Partial View 3D Model	Bild	Unüberwachtes Lernen	/10.1109/ICCVW.2009.5457716
2009	[556]	A diffusion wavelet approach for 3-D model matching	Geometrie	Label notwendig	/10.1016/j.cad.2008.11.007
2010	[338]	A 3D shape retrieval framework supporting multimodal queries	Bild	Kein Training	/10.1007/s11263-009-0277-2
2010	[557]	Visual similarity based 3D shape retrieval using Bag-of-Features	Bild	Unüberwachtes Lernen	/10.1109/SMI.2010.20
2010	[558]	Aggregating local descriptors into a compact image representation	Bild	Kein Training	/10.1109/CVPR.2010.5540039
2010	[267]	PANORAMA: a 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval	Hybrid	Unüberwachtes Lernen	/10.1007/s11263-009-0281-6
2010	[559]	Semantics-driven approach for automatic selection of best views of 3D shapes	Bild	Label notwendig	/10.2312/3DOR/3DOR10/015-022
2010	[560]	Learning the compositional structure of man-made objects for 3D shape retrieval	Hybrid	Label notwendig	/10.2312/3DOR/3DOR10/039-046
2010	[561]	Squeezing bag-of-features for scalable and semantic 3D model retrieval	Bild	Label notwendig	/10.1109/CBMI.2010.5529890

2010	[562]	Spherical correlation of visual representations for 3d model retrieval	Bild	Kein Training	/10.1007/s11263-009-0280-7
2010	[563]	Representative Views Re-Ranking for 3D Model Retrieval with Multi-Bipartite Graph Reinforcement Model	Bild	Unüberwachtes Lernen	/10.1145/1873951.1874120
2010	[564]	3D Model Comparison using Spatial Structure Circular Descriptor	Hybrid	Kein Training	/10.1016/j.patcog.2009.07.012
2010	[565]	Distance Metric Learning and Feature Combination for Shape-Based 3D Model Retrieval	Bild	Unüberwachtes Lernen	/10.1145/1877808.1877822
2010	[564]	3D Object Retrieval with Bag-of-Region-Words	Bild	Unüberwachtes Lernen	/10.1145/1873951.1874122
2010	[566]	Spectral-Driven Isometry-Invariant Matching of 3D Shapes	Geometrie	Kein Training	/ 10.1007/s11263-009-0250-0
2011	[567]	Rosy+: 3d object pose normalization based on pca and reflective object symmetry with application in 3d object retrieval	Hybrid	Kein Training	/10.1007/s11263-010-0395-x
2011	[568]	The wave kernel signature: A quantum mechanical approach to shape analysis.	Hybrid	Kein Training	/10.1109/ICCVW.2011.6130444
2011	[569]	3D shape retrieval by poisson histogram	Geometrie	Kein Training	/10.1016/j.patrec.2011.01.003
2011	[570]	View context based 2D sketch-3D model alignment	Bild	Kein Training	/10.1007/s11263-009-0277-2
2011	[571]	3D model retrieval using accurate pose estimation and view-based similarity	Bild	Kein Training	/10.1145/1991996.1992037
2011	[572]	A new framework for composing vectorial semantic labels in 3D model retrieval	Hybrid	Unüberwachtes Lernen	/10.1117/12.910564
2011	[573]	Efficient 3D Object Retrieval with Query View Selection	Bild	Label notwendig	/10.1109/TMM.2011.2160619
2011	[574]	Shape google: Geometrie words and expressions for invariant shape retrieval	Geometrie	Unüberwachtes Lernen	/10.1145/1899404.1899405
2012	[264]	KAZE Features	Bild	Kein Training	/10.1007/978-3-642-33783-3_16
2012	[575]	K-Partite Graph Reinforcement and Its Application in Multimedia Information Retrieval	Bild	Unüberwachtes Lernen	/10.1016/j.ins.2012.01.003
2012	[576]	View-based 3D Object Retrieval by Bipartite Graph Matching	Bild	Unüberwachtes Lernen	/10.1145/2393347.2396341
2012	[577]	Camera Constraint-Free View-Based 3D Object Retrieval	Geometrie	Label notwendig	/10.1109/TIP.2011.2170081
2012	[578]	3D Object Retrieval and Recognition with Hypergraph Analysis	Geometrie	Label notwendig	/10.1109/TIP.2012.2199502
2012	[579]	Efficient manifold learning for 3D model retrieval by using clustering-based training sample reduction	Geometrie	Unüberwachtes Lernen	/10.1109/ICASSP.2012.6288385
2012	[580]	Local Geometrical Feature with Positional Context for Shape-based 3D Model Retrieval	Geometrie	Unüberwachtes Lernen	-
2012	[581]	Improving 3D similarity search by enhancing and combining 3D descriptors	Hybrid	Label notwendig	/10.1007/s11042-010-0689-6
2012	[582]	Exploring collections of 3D models using fuzzy correspondences	Hybrid	Label notwendig	/10.1145/2185520.2185550
2012	[583]	Scale-invariant features for 3-D mesh models	Geometrie	Unüberwachtes Lernen	/10.1109/tip.2012.2183142
2012	[584]	Embedding retrieval of articulated geometry models	Geometrie	Unüberwachtes Lernen	/10.1109/tpami.2012.17
2011	[585]	3D CAD model retrieval with perturbed Laplacian spectra	Graph	Kein Training	/10.1016/j.compind.2011.09.003
2012	[586]	Investigating the Effects of Multiple Factors Towards More Accurate 3-D Object Retrieval	Hybrid	Unüberwachtes Lernen	/10.1109/tmm.2011.2176111
2013	[265]	Retrieving 3D Model Using Compound-Eye Visual Representation	Hybrid	Unüberwachtes Lernen	/10.1109/CADGraphics.2013.30

2013	[587]	3D model retrieval using hybrid features and class information	Hybrid	Label notwendig	/10.1007/s11042-011-0873-3
2013	[588]	A New Shape Descriptor based on Local and Global Feature for 3D Shape-Retrieval	Geometrie	Kein Training	/10.11485/itetr.37.56.0_19
2013	[589]	Content-based 3D model retrieval using a single depth image from a low-cost 3D camera	Hybrid	Kein Training	/10.1007/s00371-013-0819-z
2013	[590]	3D shape retrieval focused on holes and surface roughness	Hybrid	Kein Training	/10.1109/APSIPA.2013.6694132
2013	[591]	Data-aware 3D partitioning for generic shape retrieval	Geometrie	Kein Training	/10.1016/j.cag.2013.04.002
2014	[592]	Symmetry discovery and retrieval of nonrigid 3D shapes using geodesic skeleton paths	Graph	Kein Training	/10.1007/s11042-013-1417-9
2013	[593]	3D shape retrieval using kernels on extended Reeb graphs	Graph	Kein Training	/10.1016/j.patcog.2013.03.019
2013	[594]	PHOG: photometric and geometric functions for textured shape retrieval	Graph	Kein Training	/10.1111/cgf.12168
2015	[595]	3D shape retrieval using viewpoint information-theoretic measures	Bild	Kein Training	/10.1002/cav.1566
2013	[596]	3D object retrieval via range image queries in a bag-of-visual-words context	Bild	Unüberwachtes Lernen	/10.1007/s00371-013-0876-3
2013	[597]	Combining topological and view-based features for 3D model retrieval	Hybrid	Unüberwachtes Lernen	/10.1007/s11042-012-1000-9
2013	[598]	Cm-bof: visual similarity-based 3d shape retrieval using clock matching and bag-of-features	Bild	Unüberwachtes Lernen	/10.1007/s00138-013-0501-5
2014	[599]	A novel 3D model retrieval approach using combined shape distribution	Geometrie	Kein Training	/10.1007/s11042-012-1130-0
2016	[600]	Deep Learning Representation using Autoencoder for 3D Shape Retrieval	Bild	Unüberwachtes Lernen	/10.1109/spac.2014.6982699
2014	[601]	Pose normalization of 3D models via reflective symmetry on panoramic views	Hybrid	Kein Training	/10.1007/s00371-014-0935-4
2014	[602]	Fusing Multiple Features for Shape-based 3D Model Retrieval	Bild	Unüberwachtes Lernen	/10.5244/C.28.16
2014	[603]	Shape vocabulary: A robust and efficient shape representation for shape matching	Hybrid	Label notwendig	/10.1109/TIP.2014.2336542
2014	[604]	Supervised learning of bag-of-features shape descriptors using sparse coding	Hybrid	Label notwendig	/10.1111/cgf.12438
2014	[605]	Persistence-based structural recognition	Graph	Kein Training	/10.1109/CVPR.2014.257
2014	[606]	Sphere image for 3-D model retrieval	Bild	Kein Training	/10.1109/TMM.2014.2314073
2014	[607]	Sparse patch coding for 3D model retrieval	Hybrid	Kein Training	/10.1007/978-3-319-04117-9_11
2014	[608]	Covariance descriptors for 3D shape matching and retrieval	Hybrid	Kein Training	/10.1109/CVPR.2014.533
2015	[609]	3D object retrieval based on histogram of local orientation using one-shot score support vector machine	Geometrie	Label notwendig	/10.1007/s11704-015-4291-y
2015	[610]	3D CAD model retrieval based on the combination of features	Hybrid	Kein Training	/10.1007/s11042-013-1850-9
2015	[611]	3D Shape Matching via Two Layer Coding	Bild	Kein Training	/10.1109/TPAMI.2015.2424863
2015	[210]	3D ShapeNets: A Deep Representation for Volumetric Shapes	Geometrie	Label notwendig	/10.1109/CVPR.2015.7298801
2015	[612]	VoxNet: A 3D convolutional neural network for real-time object recognition	Geometrie	Label notwendig	/10.1109/IROS.2015.7353481
2015	[228]	Multi-view convolutional neural networks for 3D shape recognition (MVCNN)	Bild	Label notwendig	/10.48550/arXiv.1505.00880
2015	[613]	3D deep shape descriptor	Hybrid	Unüberwachtes Lernen	/10.1109/CVPR.2015.7298845
2015	[614]	Geodesic convolutional neural networks on Riemannian manifolds	Geometrie	Label notwendig	/10.48550/arXiv.1501.06297

2015	[615]	Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks	Geometrie	Label notwendig	/10.1111/cgf.12693
2015	[616]	DeepPano: Deep panoramic representation for 3-D shape recognition	Geometrie	Label notwendig	/10.1109/LSP.2015.2480802
2015	[617]	Diffusion-on-Manifold Aggregation of Local Features for Shape-based 3D Model Retrieval	Graph	Unüberwachtes Lernen	/10.1145/2671188.2749380
2015	[618]	3D object understanding with 3D Convolutional Neural Networks	Bild	Label notwendig	/10.1016/j.ins.2015.08.007
2016	[619]	Accurate aggregation of local features by using K-sparse autoencoder for 3D model retrieval	Bild	Unüberwachtes Lernen	/10.1145/2911996.2912054
2016	[620]	Unsupervised 3D local feature learning by circle convolutional restricted Boltzmann machine	Geometrie	Unüberwachtes Lernen	10.1109/TIP.2016.2605920
2016	[621]	Volumetric and multi-view CNNs for object classification on 3D data	Hybrid	Label notwendig	/10.48550/arXiv.1604.03265
2016	[622]	PointNet: A 3D convolutional neural network for real-time object class recognition	Geometrie	Label notwendig	/10.1109/IJCNN.2016.7727386
2016	[623]	GIFT: A real-time and scalable 3D shape search engine	Bild	Label notwendig	/10.48550/arXiv.1604.01879
2016	[624]	Anisotropic diffusion descriptors	Geometrie	Label notwendig	/10.1111/cgf.12844
2016	[421]	Generative and Discriminative Voxel Modeling with Convolutional Neural Networks	Geometrie	Label notwendig	/10.48550/arXiv.1608.04236
2016	[625]	Multi-View 3D Object Retrieval With Deep Embedding Network	Bild	Label notwendig	10.1109/TIP.2016.2609814
2016	[626]	Anisotropic diffusion map based spectral embedding for 3D CAD model retrieval	Geometrie	Kein Training	/10.1109/indin.2016.7819325
2017	[627]	A Novel 3D CAD Model Retrieval Method Based on Vertices Classification and Weights Combination Optimization	Hybrid	Label notwendig	/10.1155/2017/6049750
2017	[628]	Three-Dimensional Model Retrieval Using Dynamic Multi-Descriptor Fusion	Hybrid	Unüberwachtes Lernen	/10.11989/JEST.1674-862X.6071527
2017	[629]	Unsupervised 3D object retrieval with parameter-free hierarchical clustering	Hybrid	Unüberwachtes Lernen	/10.1145/3095140.3095147
2016	[630]	Orientation-boosted Voxel Nets for 3D Object Recognition	Geometrie	Label notwendig	/10.48550/arXiv.1604.03351
2016	[631]	Mesh convolutional restricted Boltzmann machines for Unüberwachtes Lernen of features with structure preservation on 3-D meshes	Geometrie	Unüberwachtes Lernen	/10.1109/TNNLS.2016.2582532
2017	[632]	Escape From Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models	Geometrie	Label notwendig	/10.1109/ICCV.2017.99
2017	[633]	Progressive shape-distribution-encoder for learning 3D shape representation	Geometrie	Unüberwachtes Lernen	10.1109/TIP.2017.2651408
2017	[634]	DeepShape: Deep-learned shape descriptor for 3D shape retrieval	Geometrie	Unüberwachtes Lernen	/10.1109/TPAMI.2016.2596722
2017	[85]	PointNet: Deep learning on point sets for 3D classification and segmentation	Geometrie	Label notwendig	/10.1109/CVPR.2017.16
2017	[635]	Unüberwachtes Lernen of 3-D local features from raw voxels based on a novel permutation voxelization strategy	Geometrie	Unüberwachtes Lernen	/10.1109/TCYB.2017.2778764
2018	[636]	PVNet: A Joint Convolutional Network of Point Cloud and Multi-View for 3D Shape Recognition	Bild	Label notwendig	/10.1145/3240508.3240702
2018	[637]	Hierarchical Graph Structure Learning for Multi-View 3D Model Retrieval.	Graph	Unüberwachtes Lernen	/10.24963/ijcai.2018/127
2018	[638]	Multi-Dimensional Network Embedding with Hierarchical Structure	Graph	Unüberwachtes Lernen	/10.1145/3159652.3159680

2018	[639]	General-purpose deep point cloud feature extractor	Geometrie	Label notwendig	/10.1109/WACV.2018.00218
2018	[640]	SO-Net: Self-organizing network for point cloud analysis	Geometrie	Label notwendig	/10.48550/arXiv.1803.04249
2018	[641]	FoldingNet: Point cloud auto-encoder via deep grid deformation	Geometrie	Label notwendig	/10.1109/CVPR.2018.00029
2018	[420]	SpiderCNN: Deep Learning on Point Sets with Parameterized Convolutional Filters	Geometrie	Label notwendig	/10.48550/arXiv.1803.11527
2018	[419]	PointCNN: Convolution On X-Transformed Points	Geometrie	Label notwendig	/10.48550/arXiv.1801.07791
2018	[642]	Triplet-Center Loss for Multi-View 3D Object Retrieval	Bild	Label notwendig	/10.48550/arXiv.1803.06189
2018	[643]	Cross-domain image-based 3d shape retrieval by view sequence learning	Bild	Unüberwachtes Lernen	/10.1109/3DV.2018.00038
2018	[644]	Emphasizing 3d properties in recurrent multi-view aggregation for 3d shape retrieval	Bild	Label notwendig	-
2018	[645]	Learning a disentangled embedding for monocular 3d shape retrieval and pose estimation	Hybrid	Label notwendig	/10.48550/arXiv.1812.09899
2018	[646]	Scale-invariant wave kernel signature for non-rigid 3D shape retrieval	Geometrie	Unüberwachtes Lernen	/10.1109/BigComp.2018.00072
2018	[647]	Geodesic-based 3D Shape Retrieval Using Sparse Autoencoders	Geometrie	Unüberwachtes Lernen	/10.2312/3dor.20181049
2018	[648]	3D shape retrieval basing on representatives of classes	Bild	Label notwendig	/10.48550/arXiv.1810.09008
2018	[649]	Content-based 3D shape retrieval using deep learning approach	Bild	Label notwendig	/10.1145/3230905.3230957
2018	[650]	3D Shape Retrieval using Volumetric and Image CNNs: A Meta-Algorithmic Approach	Bild	Label notwendig	/10.2352/ISSN.2470-1173.2018.10.IMAWM-419
2018	[651]	A methodology for part classification with supervised machine learning	Hybrid	Label notwendig	/10.1017/S0890060418000197
2018	[652]	SeqViews2SeqLabels: Learning 3D Global Features via Aggregating Sequential Views by RNN With Attention	Bild	Label notwendig	/10.1109/tip.2018.2868426
2018	[653]	Three-Dimensional CAD Model Matching With Anisotropic Diffusion Maps	Graph	Kein Training	/10.1109/tii.2017.2696042
2019	[654]	CAD models clustering with machine learning	Geometrie	Label notwendig	/10.24425/ame.2019.128441
2019	[655]	MeshNet: Mesh neural network for 3D shape representation	Geometrie	Label notwendig	/10.1609/aaai.v33i01.33018279
2019	[656]	Relation-Shape Convolutional Neural Network for Point Cloud Analysis	Geometrie	Label notwendig	/10.1109/CVPR.2019.00910
2019	[657]	Cognitive-inspired class-statistic matching with triple-constrain for camera free 3D object retrieval	Bild	Label notwendig	/10.1016/j.future.2018.12.039
2019	[658]	MLVCNN: Multi-Loop-View Convolutional Neural Network for 3D Shape Retrieval	Bild	Label notwendig	/10.1609/aaai.v33i01.33018513
2019	[659]	Learning Cross-Domain Representation with Multi-Graph Neural Network	Graph	Unüberwachtes Lernen	/10.48550/arXiv.1905.10095
2019	[660]	MeshCNN: A network with an edge	Geometrie	Label notwendig	/10.1145/3306346.3322959
2019	[661]	Point-voxel CNN for efficient 3D deep learning	Geometrie	Label notwendig	/10.48550/arXiv.1907.03739
2019	[662]	3DCapsule: Extending the capsule architecture to classify 3D point clouds	Geometrie	Label notwendig	/10.1109/WACV.2019.00132
2019	[663]	3D point cloud retrieval with bidirectional feature match	Geometrie	Kein Training	/10.1109/ACCESS.2019.2952157
2019	[664]	Pvrnet: Point-view relation neural network for 3D shape recognition	Bild	Label notwendig	/10.1609/aaai.v33i01.33019119
2019	[665]	3D2SeqViews: Aggregating Sequential Views for 3D Global Feature Learning by CNN With Hierarchical Attention Aggregation	Geometrie	Label notwendig	/10.1109/tip.2019.2904460

2019	[666]	View n-gram network for 3D object retrieval	Bild	Kein Training	/10.48550/arXiv.1908.01958
2019	[667]	Enhancing 2D representation via adjacent views for 3D shape retrieval	Bild	Label notwendig	/10.1109/ICCV.2019.00383
2019	[668]	Web3D learning framework for 3D shape retrieval based on hybrid convolutional neural networks	Bild	Label notwendig	/10.26599/TST.2018.9010113
2019	[669]	Learning domain-invariant feature for robust depth-image-based 3D shape retrieval	Bild	Label notwendig	/10.1016/j.patrec.2017.09.041
2019	[670]	Rethinking loss design for large-scale 3D shape retrieval	Bild	Label notwendig	/10.48550/arXiv.1906.00546
2019	[671]	A dimensional reduction guiding deep learning architecture for 3D shape retrieval	Geometrie	Label notwendig	/10.1016/j.cag.2019.04.002
2019	[672]	Multi-feature distance metric learning for non-rigid 3D shape retrieval	Geometrie	Label notwendig	/10.1007/s11042-019-7670-9
2019	[673]	Training convolutional neural network from multi-domain contour images for 3D shape retrieval	Bild	Label notwendig	/10.1016/j.patrec.2017.08.028
2019	[674]	Learning discriminative and generative shape embeddings for three-dimensional shape retrieval	Bild	Label notwendig	/10.1109/TMM.2019.2957933
2019	[675]	Non-rigid 3D shape retrieval based on multi-view metric learning	Geometrie	Label notwendig	/10.48550/arXiv.1904.00765
2019	[676]	3D Shape Retrieval through Multilayer RBF Neural Network	Geometrie	Unüberwachtes Lernen	/10.1109/ICIP.2019.8803384
2019	[677]	Skeleton Tree based Non-rigid 3D Shape Retrieval	Graph	Kein Training	/10.1145/3356422.3356436
2019	[678]	Anisotropic Laplace-Beltrami Operators for Non-Rigid 3D Shape Retrieval	Geometrie	Label notwendig	/10.1109/ITME.2019.00141
2020	[679]	View-Based 3-D CAD Model Retrieval With Deep Residual Networks	Bild	Label notwendig	/10.1109/tii.2019.2943195
2020	[680]	Multiple Discrimination and Pairwise CNN for view-based 3D object retrieval	Bild	Label notwendig	/10.1016/j.neunet.2020.02.017
2020	[681]	Multi-graph convolutional network for unsupervised 3D shape retrieval	Bild	Unüberwachtes Lernen	/10.1145/3394171.3413987
2020	[682]	MPAN: Multi-Part Attention Network for Point Cloud Based 3D Shape Retrieval	Geometrie	Label notwendig	/10.1109/ACCESS.2020.3018696
2020	[683]	RISA-Net: Rotation-invariant structure-aware network for fine-grained 3D shape retrieval	Geometrie	Unüberwachtes Lernen	/10.48550/arXiv.2010.00973
2020	[684]	Extending DeepSDF for automatic 3D shape retrieval and similarity transform estimation	Bild	Label notwendig	/10.48550/arXiv.2004.09048
2020	[685]	Gram Regularization for Multi-view 3D Shape Retrieval	Bild	Label notwendig	/10.48550/arXiv.2011.07733
2021	[686]	Multi-Scale Representation Learning on Hypergraph for 3D Shape Retrieval and Recognition	Graph	Label notwendig	10.1109/TIP.2021.3082765
2021	[687]	Joint learning of 3d shape retrieval and deformation	Hybrid	Label notwendig	/10.48550/arXiv.2101.07889
2021	[688]	Single Image 3D Shape Retrieval via Cross-Modal Instance and Category Contrastive Learning	Bild	Unüberwachtes Lernen	/10.1109/ICCV48922.2021.01121
2021	[689]	Multi-Modality Learning for Non-Rigid 3D Shape Retrieval via Structured Sparsity Regularizations	Geometrie	Label notwendig	/10.1109/JSEN.2021.3094122
2021	[690]	CLN: Cross-domain Learning Network for 2D Image-based 3D Shape Retrieval	Bild	Label notwendig	/10.1109/TCSVT.2021.3070969
2021	[691]	PREMA: Part-based REcurrent Multi-view Aggregation Network for 3D Shape Retrieval	Bild	Label notwendig	/10.48550/arXiv.2111.04945
2021	[692]	A method of generating depth images for view-based shape retrieval of 3D CAD models from partial point clouds	Bild	Unüberwachtes Lernen	/10.1007/s11042-020-10283-z

2021	[693]	Graph-Based Shape Analysis for Heterogeneous Geometric Datasets: Similarity, Retrieval and Substructure Matching	Graph	Kein Training	/10.1016/j.cad.2021.103125
2022	[694]	Geometric Digital Twinning of Industrial Facilities: Retrieval of Industrial Shapes	Hybrid	Label notwendig	/10.48550/arXiv.2202.04834
2022	[695]	AI based Geometrie similarity search supporting component reuse in engineering design	Graph	Label notwendig	/10.1016/j.procir.2022.05.249
2023	[696]	Center-push loss for joint view-based 3D model classification and retrieval feature learning	Bild	Label notwendig	/10.1007/s11760-021-01923-4

A.9 Ähnlichkeitssuche - Stand der Technik-Verfahren

Die gefundenen Verfahren sind nach Funktionskategorie, Trainingsart und Bewertungsmetriken unterteilt und in Tabelle 13 aufgeführt. Für die Funktion und den Trainingstyp wurden die gleichen Kategorien verwendet wie in Kapitel 2.2.3.3. In der Tabelle steht die Abkürzung NT für „No Training", LR für „Label Required" und USL für „Unsupervised Learning". Alle Verfahren, die ein Label für das Training benötigen, sind in hellblau hinterlegt. Eine Besonderheit stellt dabei das Verfahren von [611] dar. Dieses benötigt zwar keine Label für das Training, allerdings müssen Parameter abhängig vom angewendeten Datensatz definiert werden, deshalb ist dieses Vorgehen mit dunkelblau hervorgehoben worden. Nearest Neighbour (NN), First Tier (FT) und Second Tier (ST) wurden als Bewertungsmetriken für die Retrieval-Ergebnisse ausgewählt, da diese in den meisten Arbeiten verwendet wurden. In einigen Fällen wurden Methoden mit anderen Bewertungsmaßstäben vorgestellt, die aber in diesem Vergleich nicht berücksichtigt wurden. Außerdem wurden die Ergebnisse teilweise mit einem reduzierten ESB-Datensatz ermittelt. Diese sind grau in der Tabelle markiert.

In einigen Fällen wurden die Ergebniswerte nicht direkt in der ursprünglichen Veröffentlichung mit dem ESB-Datensatz erzeugt, sondern in einer späteren Publikation. Dies ist in der Tabelle in der Spalte „Quelle" durch die Angabe einer weiteren Referenz nach der ersten gekennzeichnet, welche die ursprüngliche Quelle des Ergebniswertes darstellt. Wurden mehrere Varianten einer Methode in einer Publikation vorgestellt und ausgewertet, so sind diese ebenfalls mehrfach in der Tabelle aufgeführt, jedoch mit der gleichen Quelle. Die Übersicht legt deutlich dar, dass eine Vielzahl an Verfahren für die Suche von Geometrien auf Basis von Oberflächennetzen ermittelt wurde.

Tabelle 13: Liste aller Ergebnisse mittels des ESB-Datensatzes

Akronyme	**NN [%]**	**FT [%]**	**ST [%]**	**Input-Kategorie**	**Semantik**	**Quelle**
TLC + I-Pair	88,20	56,80	72,00	VB	USL	[611]
PAN + HC	90,38	55,09	66,48	H	USL	[629]
HLO-SVM-OSS	88,70	53,70	71,20	G	LR	[609]
HSR- DE	90,20	52,80	69,70	H	NT	[590]
PANORAMA SymPan+	88,00	52,60	67,80	H	NT	[601]
VSC WCO	84,00	51,00	58,00	H	LR	[627]

MFSD+SC	87,48	50,99	71,18	G	USL	[547]
ROSy+	87,40	50,80	65,70	H	NT	[567]
DBNAA PANO	87,10	50,00	65,40	H	NT	[610]
PANORAMA + LRF	87,00	49,90	65,80	H	USL	[267]
Dynamic Multi-Descriptor	-	49,49	63,25	H	USL	[628]
PANORAMA	86,50	49,40	64,10	H	NT	[267]
Local and global features	88,50	49,21	64,02	G	NT	[588]
ShapeVocabulary	84,60	48,70	64,00	H	LR	[603]/[609]
Autoencoder	85,70	47,90	63,10	VB	USL	[600]
Spectral clustering	95,12	47,63	63,58	G	USL	[547]
ZFDR	84,10	46,80	60,90	H	LR	[587]
3D Hybrid	82,90	46,50	60,50	H	NT	[533]
PH	83,30	46,40	61,80	G	NT	[569]/[609]
CSD	82,20	46,10	61,90	G	NT	[599]/[609]
DESIRE	82,30	41,70	55,00	H	NT	[511]/[267]
Single Depth Image	79,70	41,40	54,30	H	NT	[589]/[609]
D-IA	52,00	41,00	30,00	G	NT	[521]
LF	82,00	40,40	53,90	VB	NT	[262]/[267]
SH-GEDT	80,30	40,10	53,60	G	NT	[499]/[267]
PPA-DP	72,30	32,20	47,30	G	NT	[520]/[609]
D2	40,00	29,00	18,00	G	NT	[495]
AAD	68,70	26,50	44,60	G	NT	[510]/[609]

A.10 Symbolerkennung - Literaturübersicht

Quelle	Jahr	Titel	Ingenieur-Domäne	Prozessschritt	Vorgehen (mit ML / ohne ML)	Datensatz
[697]	1979	A Threshold Selection Method from Gray-Level Histograms	Allgemein	Preprocessing	ohne ML	eigene
[698]	1990	A system for interpretation of line drawings	Allgemein	Erkennung	ohne ML	eigene
[699]	1994	Precise line detection from an engineering drawing using a figure fitting method based on contours and skeletons	Allgemein	Preprocessing	ohne ML	eigene
[700]	1994	Skeleton generation of engineering drawings via contour matching	Allgemein	Preprocessing	ohne ML	-
[701]	1994	Finding arrows in utility maps using a neural network	Allgemein	Erkennung	mit ML	eigene
[702]	1996	Automatic learning and recognition of graphical symbols in engineering drawings	Allgemein	Erkennung	ohne ML	eigene
[703]	1998	A NEW ALGORITHM FOR LINE IMAGE VECTORIZATION	Allgemein	Preprocessing	ohne ML	-
[704]	2000	Adaptive document image binarization	Allgemein	Preprocessing	ohne ML	eigene
[705]	2006	Robust and accurate vectorization of line drawings	Allgemein	Preprocessing	ohne ML	eigene
[706]	2012	Multi-Level Block Information Extraction in Engineering Drawings Based on Depth-First Algorithm	Allgemein	Erkennung + Kontextualisierung	ohne ML	eigene
[707]	2017	Adaptive document image skew estimation	Allgemein	Preprocessing	mit ML	eigene
[708]	2019	Anchor Point based Hough Transformation for Automatic Line Detection of Engineering Drawings	Allgemein	Preprocessing	ohne ML	eigene
[709]	2020	Deep Vectorization of Technical Drawings	Allgemein	Preprocessing	mit ML	Open-Source
[710]	1993	Management of graphical symbols in a CAD environment: A neural network approach	Architektur	Erkennung	mit ML	eigene
[711]	1994	SYMBOL RECOGNITION IN A CAD ENVIRONMENT USING A NEURAL NETWORK	Architektur	Erkennung	mit ML	eigene
[712]	1997	A system to understand hand-drawn floor plans using subgraph isomorphism and Hough transform	Architektur	Erkennung	ohne ML	eigene
[713]	1998	A constraint network for symbol detection in architectural drawings	Architektur	Erkennung	ohne ML	eigene
[714]	2000	A complete system for the analysis of architectural drawings	Architektur	Erkennung + Kontextualisierung	ohne ML	eigene
[715]	2001	Architectural symbol recognition using a network of constraints	Architektur	Erkennung	ohne ML	eigene
[716]	2001	Symbol recognition by error-tolerant subgraph matching between region adjacency graphs	Architektur	Erkennung	ohne ML	eigene
[717]	2002	An object-oriented progressive-simplification-based vectorization system for engineering drawings: model, algorithm, and performance	Architektur	Preprocessing + Erkennung	ohne ML	eigene
[718]	2003	A model for image generation and symbol recognition through the deformation of lineal shapes	Architektur	Erkennung	ohne ML	synthetische
[719]	2005	Using engineering drawing interpretation for automatic detection of version information in CADD engineering drawing	Architektur	Erkennung	ohne ML	eigene

[720]	2007	Automatic analysis and integration of architectural drawings	Architektur	Erkennung + Kontextualisierung	ohne ML	eigene
[721]	2008	Knowledge Extraction from Structured Engineering Drawings	Architektur	Erkennung	ohne ML	eigene
[722]	2009	Symbol Detection Using Region Adjacency Graphs and Integer Linear Programming	Architektur	Erkennung	ohne ML	Open-Source
[190]	2010	Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems	Architektur	Erkennung	mit ML	synthetische
[723]	2010	Symbol spotting in vectorized technical drawings through a lookup table of region strings	Architektur	Erkennung	ohne ML	eigene
[724]	2011	Statistical Grouping for Segmenting Symbols Parts from Line Drawings, with Application to Symbol Spotting	Architektur	Erkennung	ohne ML	Open-Source
[725]	2012	Object recognition in floor plans by graphs of white connected components	Architektur	Erkennung	ohne ML	eigene + Open-Source
[726]	2013	Geometric-based symbol spotting and retrieval in technical line drawings	Architektur	Erkennung	mit ML	eigene + Open-Source
[727]	2013	Building a Symbol Library from Technical Drawings by Identifying Repeating Patterns	Architektur	Erkennung	mit ML	Open-Source
[728]	2013	Combining geometric matching with SVM to improve symbol spotting	Architektur	Erkennung	mit ML	Open-Source
[729]	2013	Efficient symbol retrieval by building a symbol index from a collection of line drawings	Architektur	Erkennung	mit ML	Open-Source
[730]	2013	A symbol spotting approach in graphical documents by hashing serialized graphs	Architektur	Erkennung	ohne ML	Open-Source
[731]	2014	Data Extraction from DXF File and Visual Display	Architektur	Erkennung + Kontextualisierung	ohne ML	eigene
[199]	2016	Automatic Hyperlinking of Engineering Drawing Documents	Architektur	Erkennung	ohne ML	eigene
[732]	2017	Graph-Based Deep Learning for Graphics Classification	Architektur	Erkennung	mit ML	Open-Source
[733]	2018	Extraction of Ancient Map Contents Using Trees of Connected Components	Architektur	Erkennung	ohne ML	eigene
[734]	2018	Object Detection in Floor Plan Images	Architektur	Erkennung	mit ML	eigene
[735]	2019	Graph Neural Network for Symbol Detection on Document Images	Architektur	Erkennung	mit ML	Open-Source
[736]	2019	Symbol spotting for architectural drawings: state-of-the-art and new industry-driven developments	Architektur	Erkennung	mit ML	eigene
[189]	2019	BRIDGE: Building Plan Repository for Image Description Generation, and Evaluation	Architektur	Erkennung	mit ML	Open-Source
[737]	2020	Symbol Spotting on Digital Architectural Floor Plans Using a Deep Learning-based Framework	Architektur	Erkennung	mit ML	eigene / Open Source
[738]	2020	Floor Plan Recognition and Vectorization Using Combination UNet, Faster-RCNN, Statistical Component Analysis and Ramer-Douglas-Peucker	Architektur	Erkennung	mit ML	eigene
[739]	2021	Knowledge-driven description synthesis for floor plan interpretation	Architektur	Erkennung + Kontextualisierung	mit ML	Open-Source
[740]	2021	Fine Grained Feature Representation Using Computer Vision Techniques for Understanding Indoor Space	Architektur	Erkennung	mit ML	Open-Source
[741]	2021	FloorPlanCAD: A Large-Scale CAD Drawing Dataset for Panoptic Symbol Spotting	Architektur	Erkennung	mit ML	Open-Source
[742]	2021	Towards Robust Object Detection in Floor Plan Images: A Data Augmentation Approach	Architektur	Erkennung	mit ML	Open-Source
[743]	2021	PU learning-based recognition of structural elements in architectural floor plans	Architektur	Erkennung	mit ML	eigene + Open-Source

[744]	2021	3DPlanNet: Generating 3D Models from 2D Floor Plan Images Using Ensemble Methods	Architektur	Erkennung + Kontextualisierung	mit ML	Open-Source
[745]	2022	Automatic Detection and Classification of Symbols in Engineering Drawings	Architektur	Erkennung	mit ML	eigene
[746]	2022	CADTransformer: Panoptic Symbol Spotting Transformer for CAD Drawings	Architektur	Erkennung	mit ML	Open-Source
[747]	2022	GAT-CADNet: Graph Attention Network for Panoptic Symbol Spotting in CAD Drawings	Architektur	Erkennung	mit ML	Open-Source
[748]	2022	Mask-Aware Semi-Supervised Object Detection in Floor Plans	Architektur	Erkennung	mit ML	Open-Source
[749]	2022	Designing a Human-in-the-Loop System for Object Detection in Floor Plans	Architektur	Erkennung	mit ML	synthetisches Dataset
[750]	2023	Digitalization of 2D Bridge Drawings Using Deep Learning Models	Architektur	Erkennung + Kontextualisierung	mit ML	eigene + sythetische Erweitertung
[751]	2023	Improving Symbol Detection on Engineering Drawings Using a Keypoint-Based Deep Learning Approach	Architektur	Erkennung	mit ML	synthetisches Dataset
[752]	2023	You Only Look for a Symbol Once : An Object Detector for symbols and Regions in Documents.	Architektur	Erkennung	mit ML	Open-Source
[753]	2023	Towards Automatic Digitalization of Railway Engineering Schematics	Architektur	Erkennung	mit ML	eigene
[754]	2023	Leveraging Deep Convolutional Neural Network for Point Symbol Recognition in Scanned Topographic Maps	Architektur	Erkennung	mit ML	eigene
[755]	2024	Deep learning-based text detection and recognition on architectural floor plans	Architektur	Erkennung	mit ML	Open-Source + sythetische Erweitertung
[756]	1997	Adaptive Vectorization of Line Drawing Images	Architektur + Elektrotechnik	Preprocessing	ohne ML	eigene
[757]	2005	Symbol recognition via statistical integration of pixel-level constraint histograms: a new descriptor	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[758]	2006	Symbol recognition with kernel density matching	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[759]	2006	Symbol Spotting in Technical Drawings Using Vectorial Signatures	Architektur + Elektrotechnik	Erkennung	ohne ML	Open-Source
[760]	2007	A Bayesian classifier for symbol recognition	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[761]	2007	A New Syntactic Approach to Graphic Symbol Recognition	Architektur + Elektrotechnik	Erkennung	ohne ML	Open-Source
[762]	2008	On the Combination of Ridgelets Descriptors for Symbol Recognition	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[763]	2009	Graphic Symbol Recognition Using Graph Based Signature and Bayesian Network Classifier	Architektur + Elektrotechnik	Erkennung	mit ML	eigene
[764]	2010	A Bayesian network for combining descriptors: application to symbol recognition	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[765]	2011	A New Adaptive Structural Signature for Symbol Recognition by Using a Galois Lattice as a Classifier	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source

[766]	2019	GSD-Net: Compact Network for Pixel-Level Graphical Symbol Detection	Architektur + Elektrotechnik	Erkennung	mit ML	Open-Source
[767]	2002	TIF2VEC, An Algorithm for Arc Segmentation in Engineering Drawings	Architektur + Elektrotechnik + Maschinenbau	Preprocessing	ohne ML	Open-Source
[768]	2014	BoR: BAG-OF-RELATIONS FOR SYMBOL RETRIEVAL	Architektur + Elektrotechnik + P&ID	Erkennung	ohne ML	Open-Source
[769]	2013	Img2UML: A System for Extracting UML Models from Images	Computer Science	Erkennung + Kontextualisierung	ohne ML	eigene
[770]	2014	Automatic Classification of UML Class Diagrams from Images	Computer Science	Erkennung	mit ML	eigene
[771]	2021	Multiclass Classification of UML Diagrams from Images Using Deep Learning	Computer Science	Erkennung	mit ML	eigene
[772]	1982	Automatic Interpretation of Lines and Text in Circuit Diagrams	Elektrotechnik	Erkennung + Kontextualisierung	ohne ML	-
[773]	1985	Symbol recognition in electrical diagrams using probabilistic graph matching	Elektrotechnik	Erkennung	mit ML	eigene
[774]	1988	An automatic circuit diagram reader with loop-structure-based symbol recognition	Elektrotechnik	Erkennung	mit ML	Open-Source
[775]	1988	A topology-based component extractor for understanding electronic circuit diagrams	Elektrotechnik	Erkennung	mit ML	eigene
[776]	1990	TRANSLATION-, ROTATION- AND SCALE-INVARIANT RECOGNITION OF HAND-DRAWN SYMBOLS IN SCHEMATIC DIAGRAMS	Elektrotechnik	Erkennung	mit ML	eigene
[777]	1993	Recognition of logic diagrams by identifying loops and rectilinear polylines	Elektrotechnik	Erkennung	ohne ML	eigene
[778]	1993	A symbol recognition system	Elektrotechnik	Erkennung	mit ML	eigene
[779]	1993	A new system for the analysis of schematic diagrams	Elektrotechnik	Erkennung	ohne ML	eigene
[780]	1995	Automatic understanding of symbol-connected diagrams	Elektrotechnik	Kontextualisierung	ohne ML	eigene
[781]	1995	Electronic Schematic Recognition	Elektrotechnik	Erkennung + Kontextualisierung	ohne ML	eigene
[782]	2003	Engineering drawings recognition using a case-based approach	Elektrotechnik	Erkennung	ohne ML	eigene
[783]	2003	Symbol recognition in electronic diagrams using decision tree	Elektrotechnik	Erkennung	ohne ML	eigene
[784]	2004	Extracting System-Level Understanding from Wiring Diagram Manuals	Elektrotechnik	Erkennung	ohne ML	eigene
[785]	2009	A visual approach to sketched symbol recognition	Elektrotechnik	Erkennung	ohne ML	eigene
[786]	2009	On-line hand-drawn electric circuit diagram recognition using 2D dynamic programming	Elektrotechnik	Erkennung	mit ML	eigene
[787]	2011	Recognition of electrical symbols in document images using morphology and geometric analysis	Elektrotechnik	Erkennung	ohne ML	eigene
[788]	2012	Symbol recognition using spatial relations	Elektrotechnik	Erkennung	ohne ML	eigene
[789]	2015	Detection and identification of logic gates from document images using mathematical morphology	Elektrotechnik	Erkennung	mit ML	eigene
[790]	2016	Hand Drawn Optical Circuit Recognition	Elektrotechnik	Erkennung	mit ML	eigene

[791]	2017	Recognizing Electronic Circuits to Enrich Web Documents for Electronic Simulation	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	eigene
[792]	2018	Analysis of methods for automated symbol recognition in technical drawings	Elektrotechnik	Erkennung	mit ML	eigene
[793]	2019	Automatic Abstraction of Combinational Logic Circuit from Scanned Document Page Images	Elektrotechnik	Kontextualisierung	mit ML	eigene
[794]	2020	CIM/G graphics automatic generation in substation primary wiring diagram based on image recognition	Elektrotechnik	Erkennung	mit ML	eigene
[795]	2021	Graph-Based Object Detection Enhancement for Symbolic Engineering Drawings	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	eigene
[796]	2021	A public ground-truth dataset for handwritten circuit diagram images	Elektrotechnik	Erkennung	mit ML	Open-Source
[797]	2022	Symbol Spotting in Electronic Images Using Morphological Filters and Hough Transform	Elektrotechnik	Erkennung	ohne ML	Open-Source
[798]	2022	Substation One-Line Diagram Automatic Generation Based On Image Recongnition	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	eigene
[799]	2023	Instance segmentation based graph extraction for handwritten circuit diagram images	Elektrotechnik	Erkennung	mit ML	Open-Source
[800]	2023	ElectroNet: An Enhanced Model for Small-Scale Object Detection in Electrical Schematic Diagrams	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	eigene
[801]	2023	Single Line Electrical Drawings (SLED): A Multiclass Dataset Benchmarked by Deep Neural Networks	Elektrotechnik	Erkennung	mit ML	Open-Source
[802]	2023	Intelligent Digitization of Substation One-Line Diagrams Based on Computer Vision	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	synthetisches Dataset
[803]	2023	Song, Aibo and Kun, Huang and Peng, Bowen and Chen, Rui and Zhao, Kun and Qiu, Jingyi and Wang, Kaixuan	Elektrotechnik	Erkennung + Kontextualisierung	mit ML	eigene
[804]	2007	An interactive example-driven approach to graphics recognition in engineering drawings	Elektrotechnik + Architektur	Erkennung	ohne ML	eigene
[805]	2008	Spotting Symbols in Line Drawing Images Using Graph Representations	Elektrotechnik + Architektur	Erkennung	ohne ML	eigene
[806]	1994	Isolating symbols from connection lines in a class of engineering drawings	Elektrotechnik + P&ID	Erkennung	ohne ML	eigene
[193]	1997	A System for Recognizing a Large Class of Engineering Drawings	Elektrotechnik + P&ID	Erkennung	ohne ML	eigene
[807]	2014	Accurate junction detection and characterization in line-drawing images	Elektrotechnik, Architektur	Erkennung	ohne ML	Open-Source
[808]	1996	Vector-based segmentation of text connected to graphics in engineering drawings	Elektrotechnik, Maschinenbau, Architektur	Erkennung	ohne ML	eigene
[809]	1989	Processing of engineering line drawings for automatic input to CAD	Maschinenbau	Preprocessing	ohne ML	-
[810]	1989	Automatic Scanning and Interpretation of Engineering Drawings for CAD-Processes	Maschinenbau	Erkennung	ohne ML	eigene
[811]	1990	Engineering drawing processing and vectorization system	Maschinenbau	Preprocessing	ohne ML	-
[812]	1990	Interpretation of line drawings with multiple views	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[813]	1990	Randomized Hough Transform (RHT) in Engineering Drawing Vectorization System	Maschinenbau	Erkennung	ohne ML	eigene

[814]	1991	Detection of dashed lines in engineering drawings and maps	Maschinenbau	Erkennung	ohne ML	eigene
[197]	1992	Celesstin: CAD conversion of mechanical drawings	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[815]	1992	Dimensioning analysis	Maschinenbau	Erkennung	ohne ML	eigene
[816]	1992	Knowledge-directed interpretation of mechanical engineering drawings	Maschinenbau	Erkennung	ohne ML	eigene
[817]	1993	Recognition of dimensions in engineering drawings based on arrowhead	Maschinenbau	Erkennung	ohne ML	eigene
[818]	1994	Detection of dimension sets in engineering drawings	Maschinenbau	Erkennung	ohne ML	eigene
[819]	1994	Knowledge organization and interpretation process in engineering drawing interpretation	Maschinenbau	Erkennung	ohne ML	eigene
[820]	1994	SYNTACTIC ANALYSIS OF TECHNICAL DRAWING DIMENSIONS	Maschinenbau	Erkennung	ohne ML	eigene
[821]	1995	Recognition of dimension sets and integration with vectorized engineering drawings	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[822]	1995	Vector-based arc segmentation in the machine drawing understanding system environment	Maschinenbau	Erkennung	ohne ML	eigene
[823]	1996	Functional parts detection in engineering drawings: Looking for the screws	Maschinenbau	Erkennung	ohne ML	eigene
[824]	1996	A clustering-based approach to the separation of text strings from mixed text/graphics documents	Maschinenbau	Erkennung	mit ML	eigene
[825]	1996	Perfecting Vectorized Mechanical Drawings	Maschinenbau	Preprocessing	ohne ML	eigene
[826]	1996	Arrowhead recognition during automated data capture	Maschinenbau	Erkennung	ohne ML	eigene
[827]	1997	Orthogonal Zig-Zag: An algorithm for vectorizing engineering drawings compared with Hough Transform	Maschinenbau	Erkennung	ohne ML	eigene
[828]	1998	Detection of text regions from digital engineering drawings	Maschinenbau	Erkennung	ohne ML	eigene
[829]	1998	Generating multiple new designs from a sketch	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[830]	1998	Segmentation and Recognition of Dimensioning Text from Engineering Drawings	Maschinenbau	Erkennung	ohne ML	eigene
[831]	1998	A system for automatic recognition of engineering drawing entities	Maschinenbau	Erkennung + Preprocessing	ohne ML	eigene
[832]	1999	Automated CAD conversion with the Machine Drawing Understanding System: concepts, algorithms, and performance	Maschinenbau	Erkennung	ohne ML	eigene
[833]	1999	Automatic extraction of manufacturable features from CADD models using syntactic pattern recognition techniques	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[834]	1999	Dimension sets detection in technical drawings	Maschinenbau	Erkennung	ohne ML	eigene
[835]	1999	A complete system for the intelligent interpretation of engineering drawings	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[836]	2000	Symbol and character recognition: application to engineering drawings	Maschinenbau	Erkennung	mit ML	eigene
[837]	2000	Engineering Drawing Database Retrieval Using Statistical Pattern Spotting Techniques	Maschinenbau	Erkennung	mit ML	eigene
[838]	2001	Intelligent system for extraction of product data from CADD models	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene

[839]	2004	Strategy for Line Drawing Understanding	Maschinenbau	Erkennung	ohne ML	eigene
[840]	2004	A new way to detect arrows in line drawings	Maschinenbau	Erkennung	ohne ML	eigene
[841]	2009	Information extraction from scanned engineering drawings	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[842]	2010	An information extraction of title panel in engineering drawings and automatic generation system of three statistical tables	Maschinenbau	Erkennung + Kontextualisierung	ohne ML	eigene
[191]	2011	From engineering diagrams to engineering models: Visual recognition and applications	Maschinenbau	Erkennung + Kontextualisierung	mit ML	synthetisches Dataset
[843]	2016	Dimensional Arrow Detection from CAD Drawings	Maschinenbau	Erkennung	ohne ML	eigene
[844]	2017	ConvNet-Based Optical Recognition for Engineering Drawings	Maschinenbau	Erkennung	mit ML	eigene
[845]	2019	Detection of Primitives in Engineering Drawing using Genetic Algorithm	Maschinenbau	Preprocessing	ohne ML	Open-Source
[846]	2021	Extraction of dimension requirements from engineering drawings for supporting quality control in production processes	Maschinenbau	Erkennung	mit ML	eigene
[847]	2021	An Automated Engineering Assistant: Learning Parsers for Technical Drawings	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[848]	2022	Data Augmentation of Engineering Drawings for Data-Driven Component Segmentation	Maschinenbau	Erkennung	mit ML	synthetische
[849]	2022	AI-Based Engineering and Production Drawing Information Extraction	Maschinenbau	Erkennung	mit ML	synthetische
[198]	2022	Ünsupervised and hybrid vectorization techniques for 3D reconstruction of engineering drawings"	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[850]	2023	Graph neural network-enabled manufacturing method classification from engineering drawings	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[851]	2023	An Approach to Engineering Drawing Organization: Title Block Detection and Processing	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[852]	2023	AI-Based Method for Frame Detection in Engineering Drawings	Maschinenbau	Erkennung	mit ML	eigene
[853]	2023	Component segmentation of engineering drawings using Graph Convolutional Networks	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[854]	2023	Integration of Deep Learning for Automatic Recognition of 2D Engineering Drawings	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[855]	2023	2D Cross-View Object Segmentation and Perceptual Grouping in Computer-Aided Design Drawings	Maschinenbau	Erkennung + Kontextualisierung	mit ML	eigene
[856]	2024	Tolerance Information Extraction for Mechanical Engineering Drawings–A Digital Image Processing and Deep Learning-based Model	Maschinenbau	Erkennung	mit ML	eigene
[857]	2012	An improved example-driven symbol recognition approach in engineering drawings	Maschinenbau + Architektur	Erkennung	ohne ML	eigene + Open-Source
[858]	2018	Hand-written and machine-printed text classification in architecture, engineering & construction documents	Maschinenbau + Architektur	Erkennung	mit ML	eigene
[859]	2005	An image-based, trainable symbol recognizer for hand-drawn sketches	Maschinenbau + Elektrotechnik	Erkennung	mit ML	eigene

[860]	2006	An Efficient Graph-Based Symbol Recognizer	Maschinenbau + Elektrotechnik	Erkennung	mit ML	eigene
[861]	2007	An efficient graph-based recognizer for hand-drawn symbols	Maschinenbau + Elektrotechnik	Erkennung	mit ML	eigene
[191]	2011	Neural network-based symbol recognition using a few labeled samples	Maschinenbau + Elektrotechnik	Erkennung	mit ML	synthetisches Dataset + Open-Source
[862]	1994	Graphic symbol recognition using a signature technique	P&ID	Erkennung	ohne ML	eigene
[194]	1998	Computer interpretation of process and instrumentation drawings	P&ID	Erkennung	ohne ML	eigene
[863]	2006	Using process topology in plant-wide control loop performance assessment	P&ID	Kontextualisierung	ohne ML	-
[864]	2009	Graphic Symbol Recognition Using Auto Associative Neural Network Model	P&ID	Erkennung	mit ML	eigene
[192]	2015	A 2D Engineering Drawing and 3D Model Matching Algorithm for Process Plant	P&ID	Erkennung + Kontextualisierung	ohne ML	eigene
[865]	2016	Automatische Analyse und Erkennung graphischer Inhalte von SVG-basierten Engineering-Dokumenten	P&ID	Erkennung	ohne ML	eigene
[866]	2017	Heuristics-Based Detection to Improve Text/-Graphics Segmentation in Complex Engineering Drawings	P&ID	Erkennung	ohne ML	Industrie
[867]	2018	Symbols Classification in Engineering Drawings	P&ID	Erkennung	mit ML	eigene
[868]	2019	A Digitization and Conversion Tool for Imaged Drawings to Intelligent Piping and Instrumentation Diagrams (P&ID)	P&ID	Erkennung + Kontextualisierung	ohne ML	eigene
[869]	2019	Automatic Information Extraction from Piping and Instrumentation Diagrams	P&ID	Erkennung	mit ML	eigene
[196]	2019	Applying graph matching techniques to enhance reuse of plant design information	P&ID	Kontextualisierung	ohne ML	eigene
[870]	2019	Features Recognition from Piping and Instrumentation Diagrams in Image Format Using a Deep Learning Network	P&ID	Erkennung	mit ML	eigene
[871]	2020	Deep learning for symbols detection and classification in engineering drawings	P&ID	Erkennung	mit ML	eigene / Industrie
[872]	2020	Symbols in Engineering Drawings (SiED): An Imbalanced Dataset Benchmarked by Convolutional Neural Networks	P&ID	Erkennung	mit ML	Open-Source
[873]	2020	Object Detection in Design Diagrams with Machine Learning	P&ID	Erkennung	mit ML	synthetisches Dataset
[874]	2020	Deep Neural Network for Automatic Image Recognition of Engineering Diagrams	P&ID	Erkennung	mit ML	eigene
[875]	2020	CNN-Based Symbol Recognition in Piping Drawings	P&ID	Erkennung	mit ML	synthetisches Dataset
[876]	2020	Graph-Based Manipulation Rules for Piping and Instrumentation Diagrams	P&ID	Kontextualisierung	ohne ML	eigene
[877]	2020	Deep Learning for Text Detection and Recognition in Complex Engineering Diagrams	P&ID	Erkennung	mit ML	eigene
[878]	2020	Automatic Digitization of Engineering Diagrams using Deep Learning and Graph Search	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[879]	2020	Reducing human effort in engineering drawing validation	P&ID	Erkennung + Kontextualisierung	mit ML	eigene

[880]	2020	Integrating 2D and 3D Digital Plant Information Towards Automatic Generation of Digital Twins	P&ID	Erkennung + Kontextualisierung	ohne ML	eigene
[881]	2020	Component detection in piping and instrumentation diagrams of nuclear power plants based on neural networks	P&ID	Erkennung + Kontextualisierung	mit ML	Open-Source
[882]	2021	OSSR-PID: One-Shot Symbol Recognition in P&ID Sheets using Path Sampling and GCN	P&ID	Erkennung	mit ML	synthetisches Dataset
[883]	2021	Deep Learning-Based Method to Recognize Line Objects and Flow Arrows from Image-Format Piping and Instrumentation Diagrams for Digitization	P&ID	Erkennung	mit ML	eigene aus Industrie
[884]	2021	Group of components detection in engineering drawings based on graph matching	P&ID	Kontextualisierung	ohne ML	eigene
[885]	2021	Automatic Digitization of Engineering Diagrams using Intelligent Algorithms	P&ID	Erkennung	ohne ML	eigene
[886]	2021	Engineering Drawing Validation Based on Graph Convolutional Networks	P&ID	Erkennung	mit ML	eigene
[887]	2021	Digitize-PID: Automatic Digitization of Piping and Instrumentation Diagrams	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[888]	2021	Automatic digital twin data model generation of building energy systems from piping and instrumentation diagrams	P&ID	Erkennung	mit ML	eigene
[889]	2021	Identification of Objects in Oilfield Infrastructure using Engineering Diagram and Machine Learning Methods	P&ID	Erkennung	mit ML	eigene
[890]	2021	Object detection for P&ID images using various deep learning techniques	P&ID	Erkennung	mit ML	eigene
[195]	2022	Pattern Recognition Method for Detecting Engineering Errors on Technical Drawings	P&ID	Erkennung	mit ML	eigene
[891]	2022	Enhanced Symbol Recognition based on Advanced Data Augmentation for Engineering Diagrams	P&ID	Erkennung	mit ML	eigene + sythetische Erweitertung
[892]	2022	Modern Deep Learning Approaches for Symbol Detection in Complex Engineering Drawings	P&ID	Erkennung	mit ML	eigene
[893]	2022	End-to-end digitization of image format piping and instrumentation diagrams at an industrially applicable level	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[894]	2022	Automated Valve Detection in Piping and Instrumentation (P&ID) Diagrams	P&ID	Erkennung	mit ML	eigene
[895]	2023	A Symbol Recognition System for Single-Line Diagrams Developed Using a Deep-Learning Approach	P&ID	Erkennung	mit ML	eigene + sythetische Erweiterung
[896]	2023	Advancing P&ID Digitization with YOLOv5	P&ID	Erkennung	mit ML	eigene
[897]	2023	Improved P&ID Symbol Detection Algorithm Based on YOLOv5 Network	P&ID	Erkennung	mit ML	eigene
[898]	2023	Automatic anomaly detection in engineering diagrams using machine learning	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[899]	2023	Digitization of chemical process flow diagrams using deep convolutional neural networks	P&ID	Erkennung	mit ML	eigene
[900]	2023	Extraction of line objects from piping and instrumentation diagrams using an improved continuous line detection algorithm	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[901]	2023	Classification of Functional Types of Lines in P&IDs Using a Graph Neural Network	P&ID	Erkennung	mit ML	eigene
[902]	2023	Demonstrating Automated Generation of Simulation Models from Engineering Diagrams	P&ID	Erkennung + Kontextualisierung	mit ML	synthetisches Dataset

[903]	2023	A Complete Piping Identification Solution for Piping and Instrumentation Diagrams	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[904]	2024	Auto-Routing Systems (ARSs) with 3D Piping for Sustainable Plant Projects Based on Artificial Intelligence (AI) and Digitalization of 2D Drawings and Specifications	P&ID	Erkennung + Kontextualisierung	mit ML	eigene
[905]	2024	Rule-based continuous line classification using shape and positional relationships between objects in piping and instrumentation diagram	P&ID	Erkennung	ohne ML	eigene
[906]	2024	Image format pipeline and instrument diagram recognition method based on deep learning	P&ID	Erkennung + Kontextualisierung	mit ML	Open-Source
[907]	2024	Semi-supervised symbol detection for piping and instrumentation drawings	P&ID	Erkennung + Kontextualisierung	mit ML	eigene

B Ergänzungen zu Kapitel 4

B.1 Datenerzeugung - Algorithmen

B.1.1 Pillow

Algorithm 1: Überblick über den Algorithmus zur Datengenerierung mit dem Pillow-Paket

Data: Anzahl an Bildern **n**, Anzahl an Symbolen **s**, doBackground
Result: Datensatz mit Bildern und Label

```
for i ∈ n do
    image = createImage ();
    if doBackground then
        image = drawrndBackground(image);
    for j ∈ randInt(3,s) do
        bbLoc = randomLocation(image);
        if checkIntersec(bbLoc) then
            break
        else
            [image,BB] = drawSymbol(randInt(0,s),image,bbLoc);
    saveBBxml(BB);
    saveImage(image);
```

B.1.2 SketchGraph

Algorithm 2: Überblick über den Algorithmus zur Datenerzeugung mit SketchGraph

Data: Anzahl an Bildern **n**, Anzahl an Symbolen **s**, doRealistic, hand drawn factor **hf**
Result: Datensatz mit Bildern und Label

```
for i ∈ n do
    seq = createSketch (node);
    while i < drawSymbolwoPlanetGear(3,s) do
        if doRealistic then
            sym[i] = chooseSym(symPair);
            sym[i + 1] = chooseSym(symPair);
            seq = drawSymSeq(seq, sym[i, i + 1]);
            i ← i + 2 ;
            for c ∈ s − 1 do
                seq = drawConnection(seq);
        sym[i] = chooseSym(symclass);
        if instPlanetGear(sym)> 1 && sym[i] == PlanetGear then
            break;
        else
            seq = drawSymSeq(seq, sym);
            i ← i + 1 ;
    solvedSketch = SG_SketchFromSeq(seq);
    for j ∈ drawSymbolwoPlanetGear(3,s) do
        BB[j] = getBBfromSketch(solvedSketch);
    image = renderSketch(solvedSketch,hf);
    saveBBxml(BB);
    saveImage(image);
```

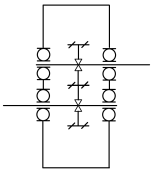
B.2 Datensätze - Übersicht

B.2.1 Getriebestufen

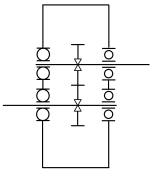
Tabelle 15: Übersicht der Symbol-Instanzen im Datensatz der unbekannten Getriebestufen ohne angewendete Skalierung

Kategorie	Anzahl	Kategorie	Anzahl
Kegelrad	36	Allgemeines Lager	216
Kegelrad Hohlwelle	0	Allgemeines Lager Hohlwelle	0
Kugelgelenk	0	Gelenk	0
Planeten-getriebe	0	Stirnrad gerade	82
Planetengetriebe Hohlwelle	0	Stirnrad gerade Hohlwelle	0
Festlager	0	Stirnrad schräg	54
Loslager	0	Stirnrad schräg Hohlwelle	0

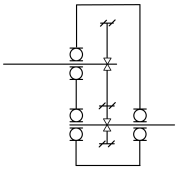
B.2.2 Übersicht aller Bilder



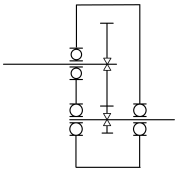
(a) Skizze 0 - Stufe 1



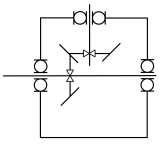
(b) Skizze 1 - Stufe 1



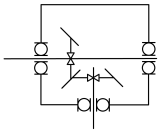
(c) Skizze 2 - Stufe 1



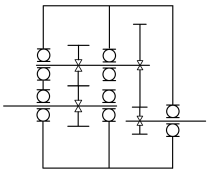
(d) Skizze 3 - Stufe 1



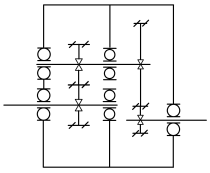
(e) Skizze 4 - Stufe 1



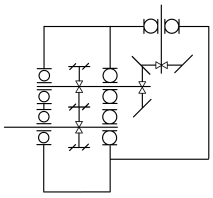
(f) Skizze 5 - Stufe 1



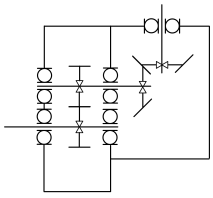
(a) Skizze 6 - Stufe 2



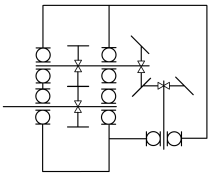
(b) Skizze 7 - Stufe 2



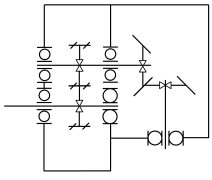
(c) Skizze 8 - Stufe 2



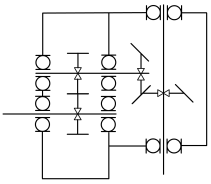
(d) Skizze 9 - Stufe 2



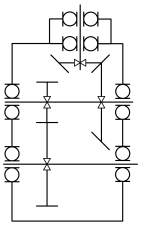
(e) Skizze 10 - Stufe 2



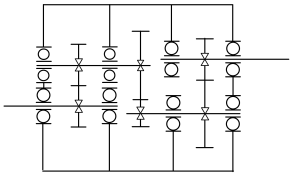
(f) Skizze 11 - Stufe 2



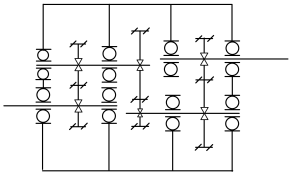
(a) Skizze 12 - Stufe 2



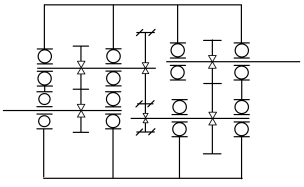
(b) Skizze 29 - Stufe 2



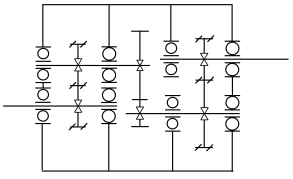
(c) Skizze 13 - Stufe 3



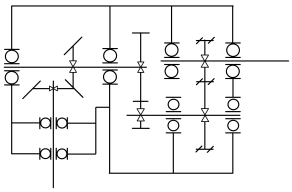
(d) Skizze 14 - Stufe 3



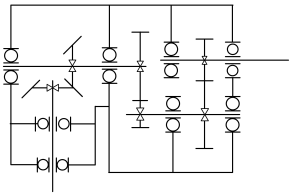
(e) Skizze 15 - Stufe 3



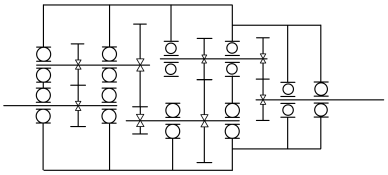
(f) Skizze 16 - Stufe 3



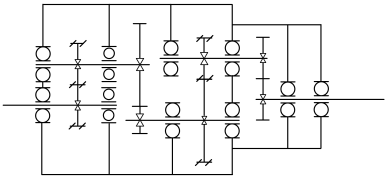
(a) Skizze 17 - Stufe 3



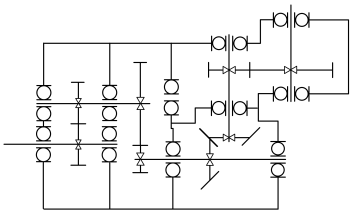
(b) Skizze 18 - Stufe 3



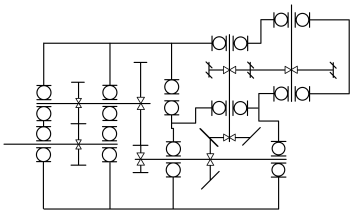
(c) Skizze 19 - Stufe 4



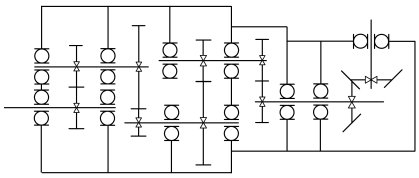
(d) Skizze 20 - Stufe 4



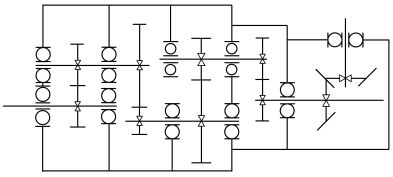
(e) Skizze 21 - Stufe 4



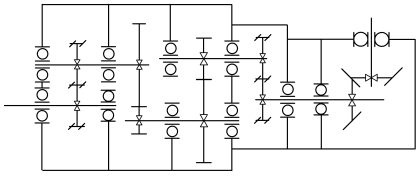
(f) Skizze 22 - Stufe 4



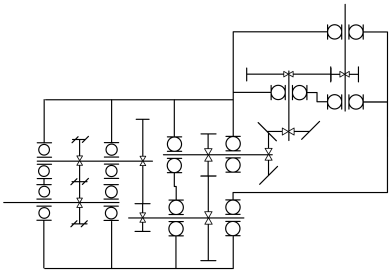
(a) Skizze 23 - Stufe 5



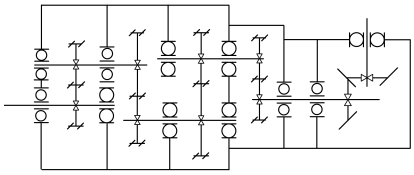
(b) Skizze 24 - Stufe 5



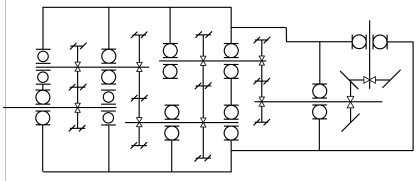
(c) Skizze 25 - Stufe 5



(d) Skizze 26 - Stufe 5



(e) Skizze 27 - Stufe 5



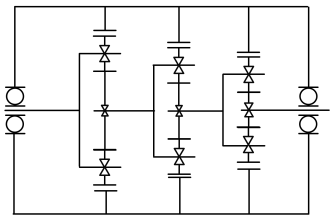
(f) Skizze 28 - Stufe 5

B.2.3 Unbekannte Baugruppen

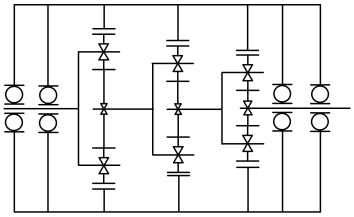
Tabelle 16: Übersicht über die Symbolinstanzen im unbekannten Baugruppendatensatz

Kategorie	**Anzahl**	**Kategorie**	**Anzahl**
Kegelrad	3	Allgemeines Lager	117
Kegelrad Hohlwelle	1	Allgemeines Lager Hohlwelle	17
Kugelgelenk	0	Gelenk	21
Planeten-getriebe	28	Stirnrad gerade	45
Planetengetriebe Hohlwelle	0	Stirnrad gerade Hohlwelle	0
Festlager	5	Stirnrad schräg	5
Loslager	4	Stirnrad schräg Hohlwelle	0

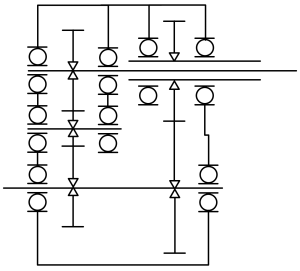
B.2.4 Übersicht aller Bilder



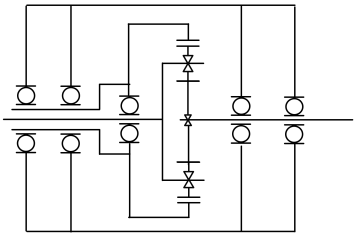
(a) Skizze Reduktion - 01



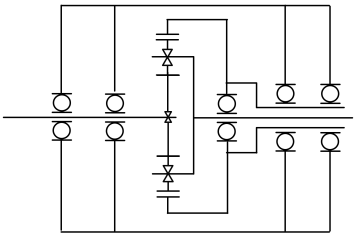
(b) Skizze Reduktion - 02



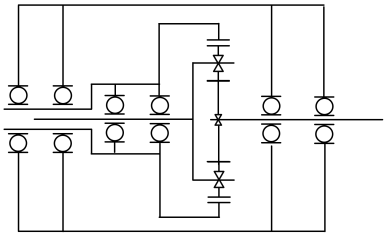
(c) Skizze Double-Rotor - 01



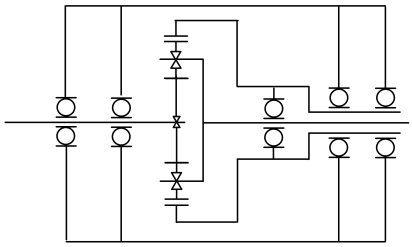
(d) Skizze Double-Rotor - 02



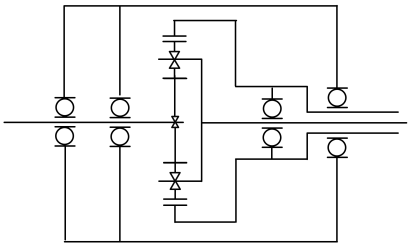
(e) Skizze Double-Rotor - 03



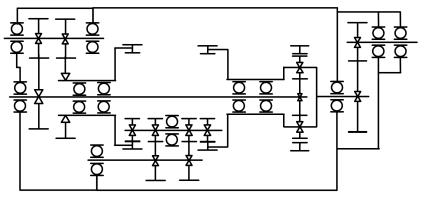
(f) Skizze Double-Rotor - 04



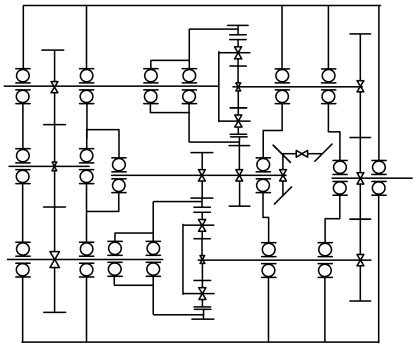
(a) Skizze Double-Rotor - 05



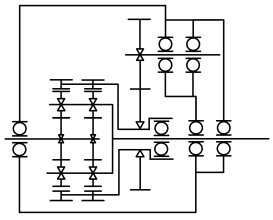
(b) Skizze Double-Rotor - 06



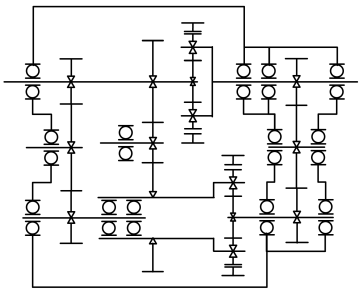
(c) Skizze Hydraulik-Rotor - 01



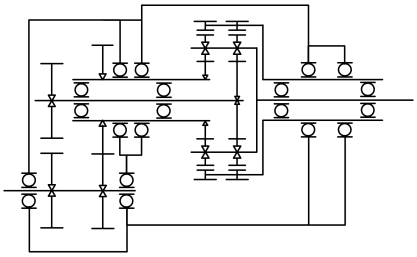
(d) Skizze Hydraulik-Rotor - 02



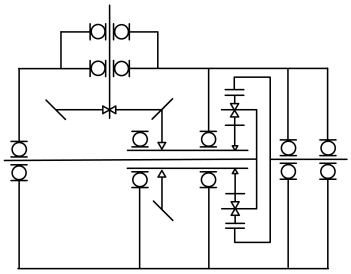
(e) Skizze Hydraulik-Rotor - 03



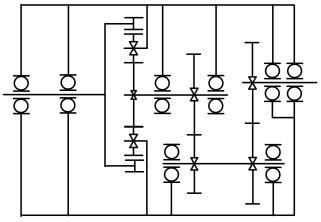
(f) Skizze Hydraulik-Rotor - 04



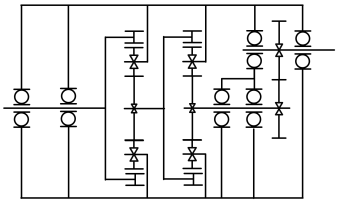
(a) Skizze Hydraulik-Rotor - 05



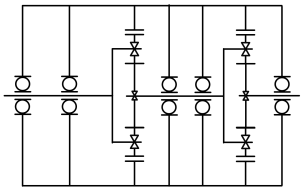
(b) Skizze Differential - 01



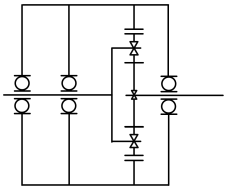
(c) Skizze Wind-Turbine - 01



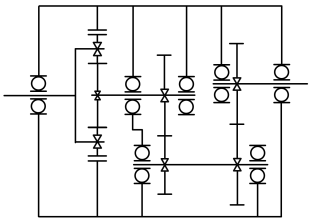
(d) Skizze Wind-Turbine - 02



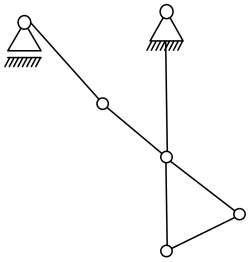
(e) Skizze Wind-Turbine - 03



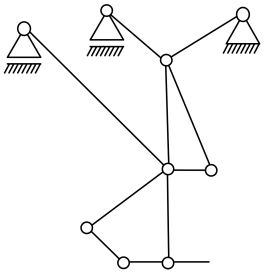
(f) Skizze Wind-Turbine - 04



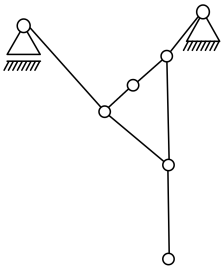
(a) Skizze Wind-Turbine - 05



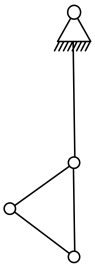
(b) Skizze Flugzeug-Fahrwerk - 01



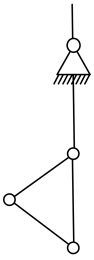
(c) Skizze Flugzeug-Fahrwerk - 02



(d) Skizze Flugzeug-Fahrwerk - 03



(e) Skizze Flugzeug-Fahrwerk - 04



(f) Skizze Flugzeug-Fahrwerk - 05

B.2.5 Übersicht der Bewertungsmetriken für beide Test-Datensätze

Tabelle 17: Bewertungsmetriken für alle Modelle und Trainingsdatensätze auf unbekannten Daten, wobei die besten Ergebnisse fett hervorgehoben und unterstrichen sind

Unbekannte Getriebestufen												
Datensatz	SketchGraph 1		SketchGraph 2		SketchGraph 3		SketchGraph 4		Pillow 1		Pillow 2	
Metriken	mAP50	mAP	mAP50	mAP	mAP50	mAP	mAP50	mAP	mAP50	mAP	mAP50	mAP
YOLOv5 S	0.563	0.303	0.180	0.073	0.993	**0.694**	0.992	0.684	0.388	0.138	0.716	0.245
YOLOv5 M	0.805	0.455	0.076	0.033	**0.994**	0.689	**0.994**	0.685	0.625	0.253	0.868	0.309
YOLOv5 L	0.742	0.444	0.084	0.050	0.993	0.686	0.994	0.686	0.817	0.397	0.832	0.315
Mask RCNN r50	0.149	0.056	0.080	0.031	0.741	0.389	0.834	0.440	0.439	0.268	0.235	0.154
Mask RCNN r101	0.105	0.048	0.135	0.057	0.949	0.522	0.785	0.407	0.127	0.079	0.351	0.224
Faster RCNN r50	0.286	0.117	0.241	0.110	0.736	0.526	0.757	0.535	0.042	0.013	0.197	0.105
Unbekannte Baugruppen												
YOLOv5 S	0.659	0.419	0.274	0.180	0.904	0.554	0.893	0.537	0.492	0.170	0.559	0.239
YOLOv5 M	0.717	0.432	0.193	0.149	0.904	**0.567**	0.868	0.537	0.485	0.188	0.561	0.248
YOLOv5 L	0.688	0.450	0.200	0.142	**0.906**	0.562	0.891	0.565	0.597	0.226	0.723	0.283
Mask RCNN r50	0.111	0.050	0.080	0.030	0.501	0.184	0.451	0.131	0.189	0.124	0.011	0.007
Mask RCNN r101	0.110	0.037	0.140	0.055	0.642	0.238	0.575	0.195	0.142	0.062	0.061	0.044
Faster RCNN r50	0.352	0.223	0.365	0.241	0.628	0.376	0.664	0.430	0.144	0.026	0.392	0.217

B.2.6 Häufigkeit der Symbole für die Iteration der SketchGraph Datenerzeugung

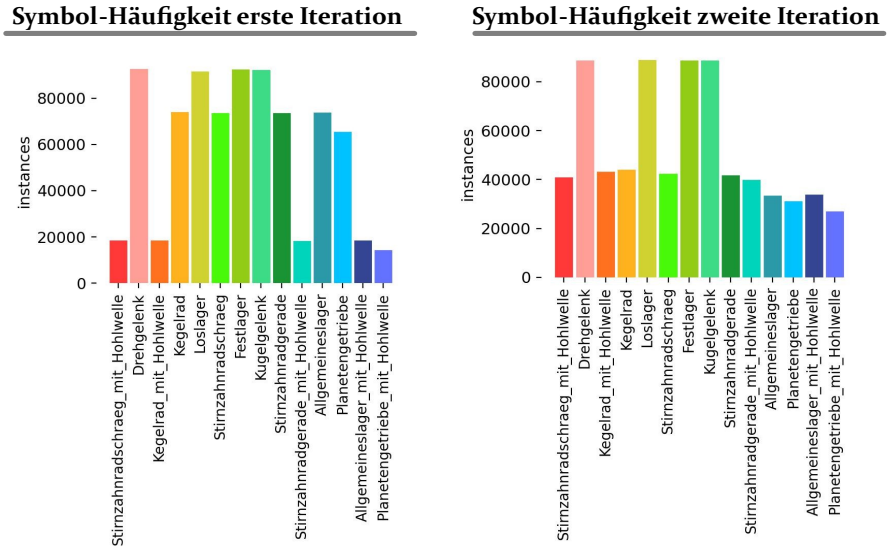


Bild 100: Häufigkeit der Symbole im alten und neuen Ansatz zur Datenerzeugung nach [S5]

B.2.7 Parameter und Ergebnisse für GED-Skizzen - Vergleich

Parameter:

- Dbscan: Epsilon 6, min-samples 3
- Thinning: size 1
- Point-Size: 20

Tabelle 18: Ergebnisse der Suche mit einer Handskizze als Input

Skizzennummer	GED-Distanz	Skizzennummer	GED-Distanz
0	12.0	16	19.0
1	10.0	17	23.0
2	13.0	18	26.0
3	11.0	19	24.0
4	13.0	20	43.0
5	13.0	21	46.0
6	**3.0**	22	40.0
7	7.0	23	40.0
8	7.0	24	60.0
9	5.0	25	49.0
10	5.0	26	60.0
11	7.0	27	49.0
12	10.0	28	66.0
13	10.0	29	53.0
14	19.0		
15	23.0		

C Ergänzungen zu Kapitel 5

C.1 Formel für Projektion

$$
\begin{aligned}
center &= \vec{C} + \vec{PC}; \\
normal &= \vec{PC} + \vec{t1} * \vec{n}; \\
\\
mit & \\
\vec{C} &= [0, 0, 0] \\
\vec{PC} &= PointCloudVertices \\
\vec{n} &= NormalsVertices \\
\vec{t1} &= \frac{-b+\sqrt{b^2-4*a*c}}{2*a} \\
a &= \vec{n} \cdot \vec{n} \\
b &= 2 * (PC_X * n_X - C_X * n_X + PC_Y * n_Y - C_Y * n_Y + \\
& \quad PC_Z * n_Z - C_Z * n_Z); \\
c &= PC_X^2 + PC_Y^2 + PC_Z^2 - r^2 + C_X^2 + C_Y^2 + C_Z^2 - \\
& \quad 2 * (PC_X * C_X + PC_Y * C_Y + PC_Z * C_Z) \\
r &= \sqrt{2 * max(|\vec{PC}|)^2} + 0.05
\end{aligned}
\tag{19}
$$

C.2 PointNet++

Ein anderer Ansatz zur Bauteilerkennung ist die Idee, das Prinzip von CNN-Modellen direkt auf Punktewolken zu übertragen, was zum PointNet von Qi et al. [85] führte. Das Ziel war, ein Vorgehen zu entwickeln, das unabhängig gegenüber Transformationen, Rotationen und Auflösung der Punktewolke ist. Die Umsetzung erfolgte über eine Kombination aus neuronalen Netz und Max-Pooling-Operation, welche eine Symmetriefunktion imitieren, die anschließend die Informationen jedes Knotens zusammenfasst. Die Anwendung des PointNet zeigte sehr gute Ergebnisse bei der Erkennung und Segmentierung von Komponenten oder 3D-Szenen und kann auch für Punktewolken aus FE-Simulationen eingesetzt werden [P3].

Allerdings hatte die Erkennung lokaler Merkmale in Geometrien noch Optimierungspotenzial, weshalb eine Erweiterung veröffentlicht wurde, das PointNet++ [225]. Die neue Version baut auf der ursprünglichen Struktur auf und führt einen hierarchischen Ablauf in die Architektur ein. Zunächst wird ein Punkt-Sampling durchgeführt, um die Eingabepunktewolke in überlappende Regionen zu unterteilen, wodurch die Identifizierung lokaler Merkmale durch das PointNet erfolgt. Eine Übersicht des grundlegenden PointNet++ Verfahrens ist in Bild 101 dargestellt.

Die unterschiedlichen Abstraktionslevel repräsentieren dabei den hierarchischen Aufbau, wobei ein Level aus einer Sampling-, Gruppierungs- und PointNet-Schicht besteht. Durch diese Anpassung konnte eine weitere Verbesserung in den gängigen Benchmark-Datensätzen (z. B. ModelNet40 [210]) erreicht werden. Zudem ist eine Erweiterung des Eingabe-Vektors der Punktewolke um andere numerische Werte ermöglicht worden.

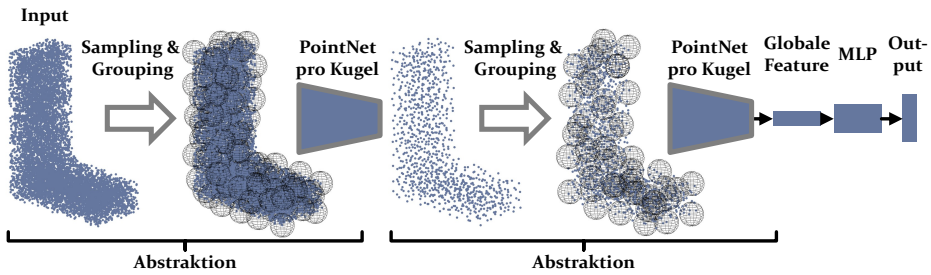
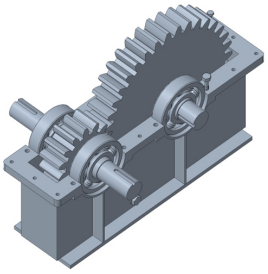
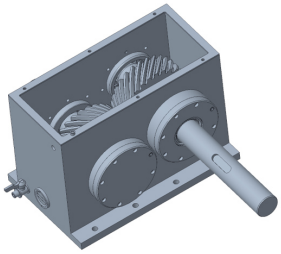
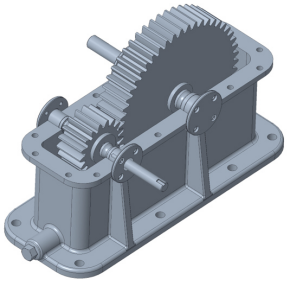
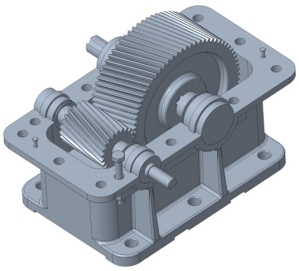
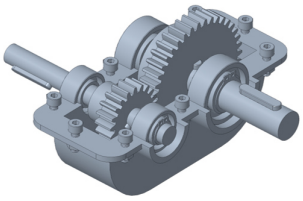


Bild 101: Überblick des Pointnet++ zur Bauteilerkennung nach [225]

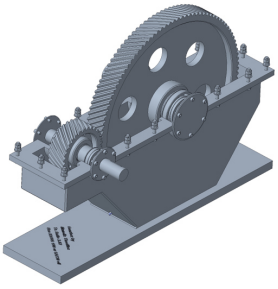
C.3 Demonstrator-Datensatz Skizze - Baugruppe

Tabelle 19: Übersicht aller Baugruppen des Demonstratordatensatzes mit ihrer Nummer bestehend aus *Bauteilnummer - Getriebestufen - Bauteilanzahl*

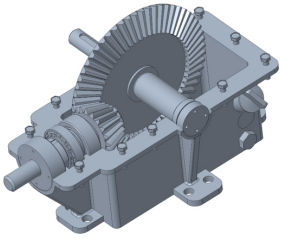
Bauteil-Nummer	Bild	Link
01-1-91		grabcad.com/library/gearbox-327
03-1-28		grabcad.com/library/parallel-shafts-helical-gearbox-1
09-1-30		grabcad.com/library/gear-box-_one-stage-2
15-1-30		grabcad.com/library/gearbox-reducer-6



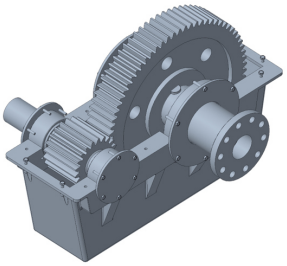
22-1-13 grabcad.com/library/gearbox-296



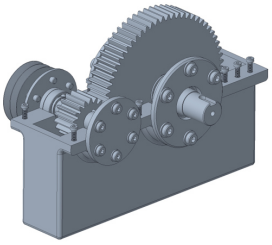
23-1-90 grabcad.com/library/gearbox–9



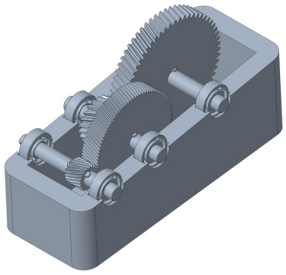
24-1-33 grabcad.com/library/gear-box-3-1



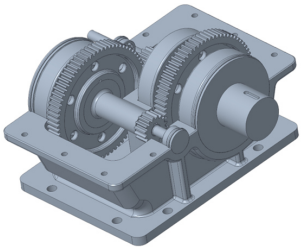
26-1-32 grabcad.com/library/helical-gearbox-5



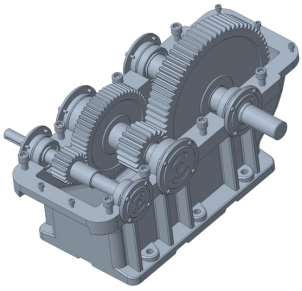
27-1-44 grabcad.com/library/gear-box-1-speed-1



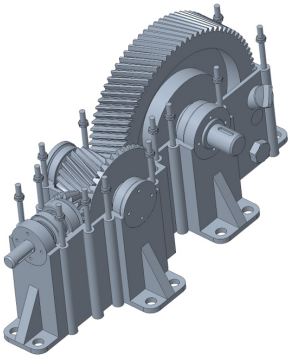
04-2-11 grabcad.com/library/simple-gear-box-4



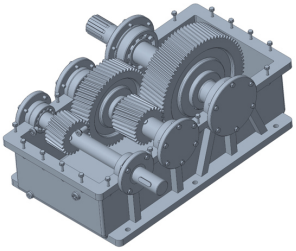
10-2-17 grabcad.com/library/gear-box-1-gear-reducer-1



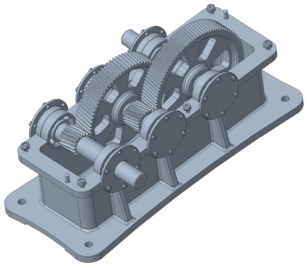
21-2-47 grabcad.com/library/gearbox-57



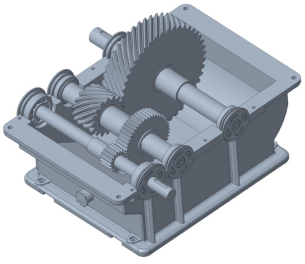
28-2-41 grabcad.com/library/gear-box-2-7



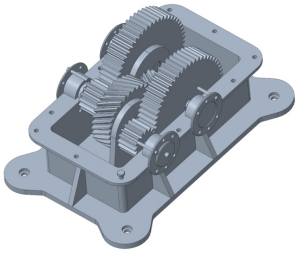
29-2-103 grabcad.com/library/gearbox-145



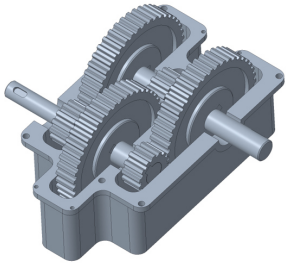
30-2-32 grabcad.com/library/2-stage-gearbox-2



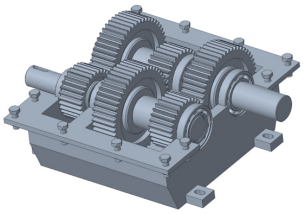
31-2-28 grabcad.com/library/mechanical-crane-s-gearbox-1



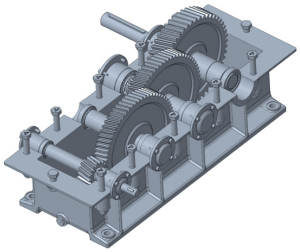
11-3-25 grabcad.com/library/gear-box-112



12-3-7 grabcad.com/library/gearbox-29



25-3-22 grabcad.com/library/gearbox-22



32-3-121 grabcad.com/library/gearbox-6

C.4 Pseudocode Graphaufbereitung

Listing C.1: Pseudocode der Methode zum Zusammenfassen von wiederkehrenden Mustern nach [S9]

```
Input: Graph G der CAD–Baugruppe
Output: Modifizierter Graph G der CAD–Baugruppe

L ← [Liste aller Musterbezeichnung und deren Graphen zum Zusammenfassen]

for jedes Muster M in L do
    if M als Subgraph in Graph G then
        Entferne alle Knoten aus M in G;
        Füge neuen Knoten mit Musterbezeichnung ein;
    end
end
```

Listing C.2: Pseudocode der Methode zum Ausschließen von Knoten bestimmter Bauteilklassen nach [S9]

```
Input: Graph G, BK Bauteilklassen die verbleiben sollen
Output: Modifizierter Graph G mit Knoten aus den Bauteilklassen BK

U = G ← [Ursprünglicher Graph]

for jeden Knoten K in G do
    if Bauteilklasse von K nicht in BK then
        Entferne K aus G;
    end
end

for jeden Knoten K1 in G do
    for jeden Knoten K2 in G do
        if K1 nicht gleich K2 and K1 und K2 nicht
        durch Kante verbunden then
            Finde kürzesten Pfad zwischen K1 und K2 in U;
            if alle Bauteilklassen der Knoten auf kürzesten Pfad
            nicht in BK then
                Verbinde K1 und K2 in G durch Kante;
            end
        end
    end
end
```

C.4.0.1Getriebestufen-Datensatz - Gesamte Ergebnisse

MCS - Confusion Matrix:

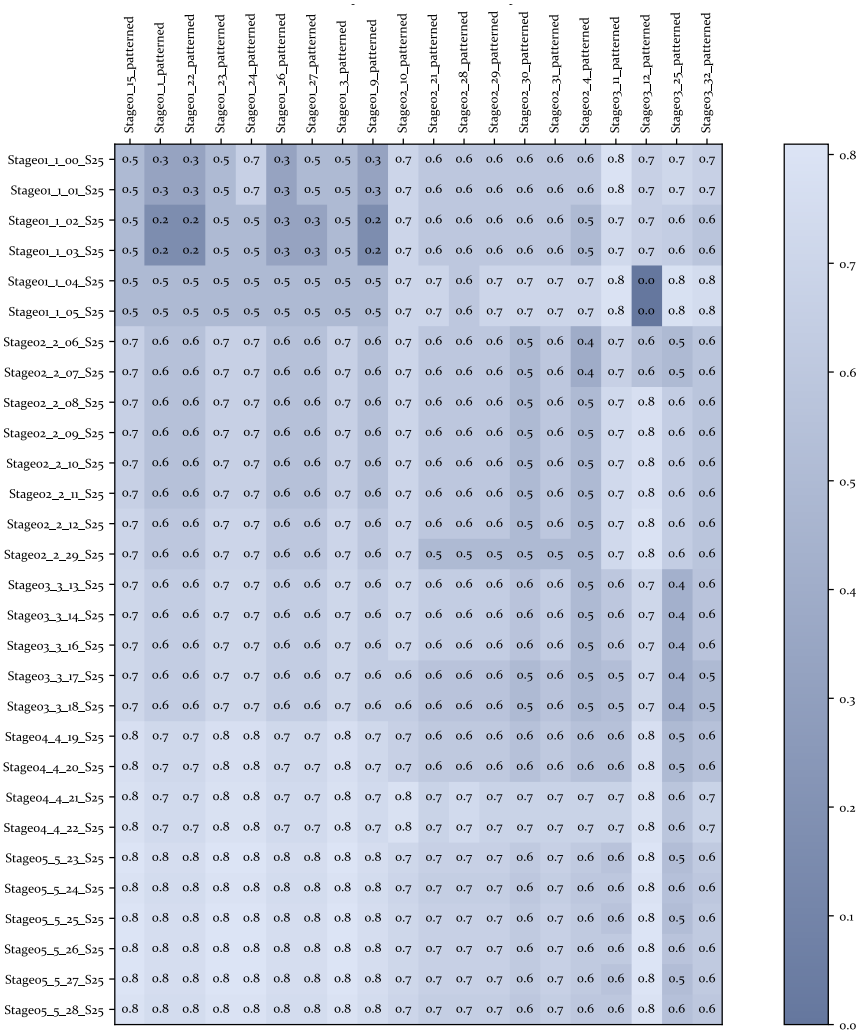


Bild 102: Wahrheitsmatrix für den MSC-Score

Jaccard Ähnlichkeit - Confusion Matrix:

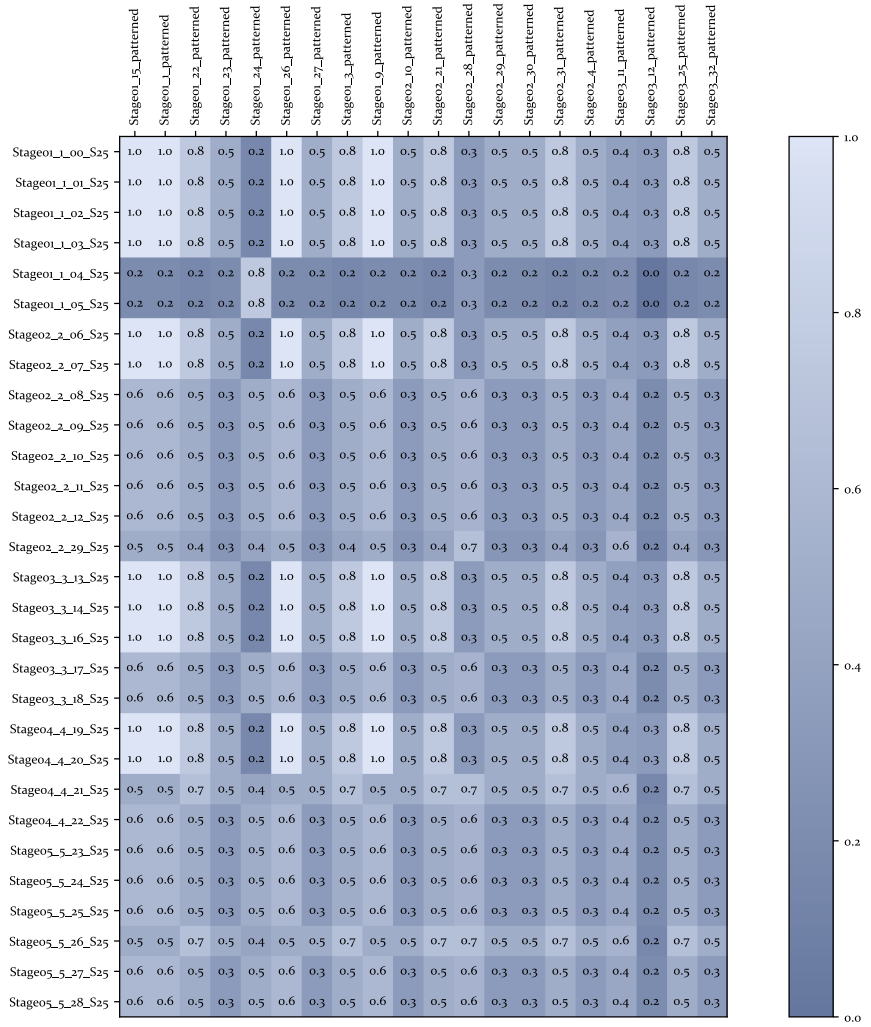


Bild 103: Wahrheitsmatrix für den Jaccard-Score

Eigenwerte - Confusion Matrix:

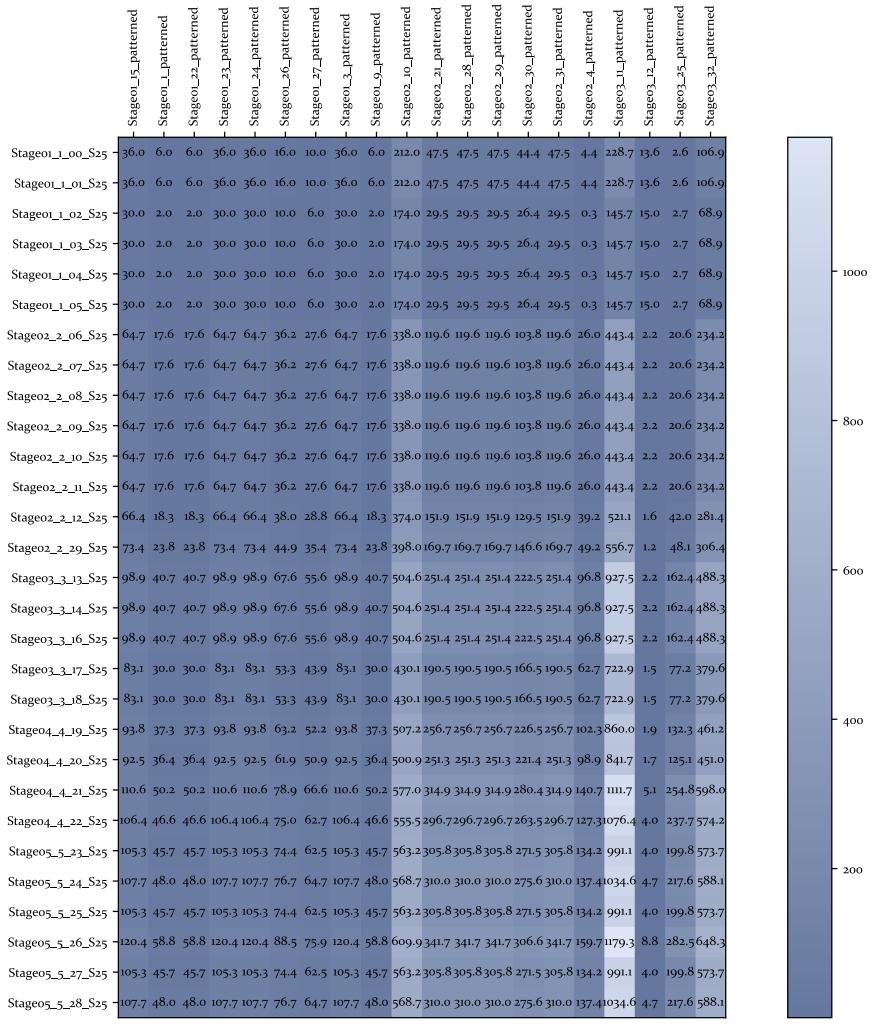


Bild 104: Wahrheitsmatrix für den Eigenwert-Score

GED - Confusion Matrix:

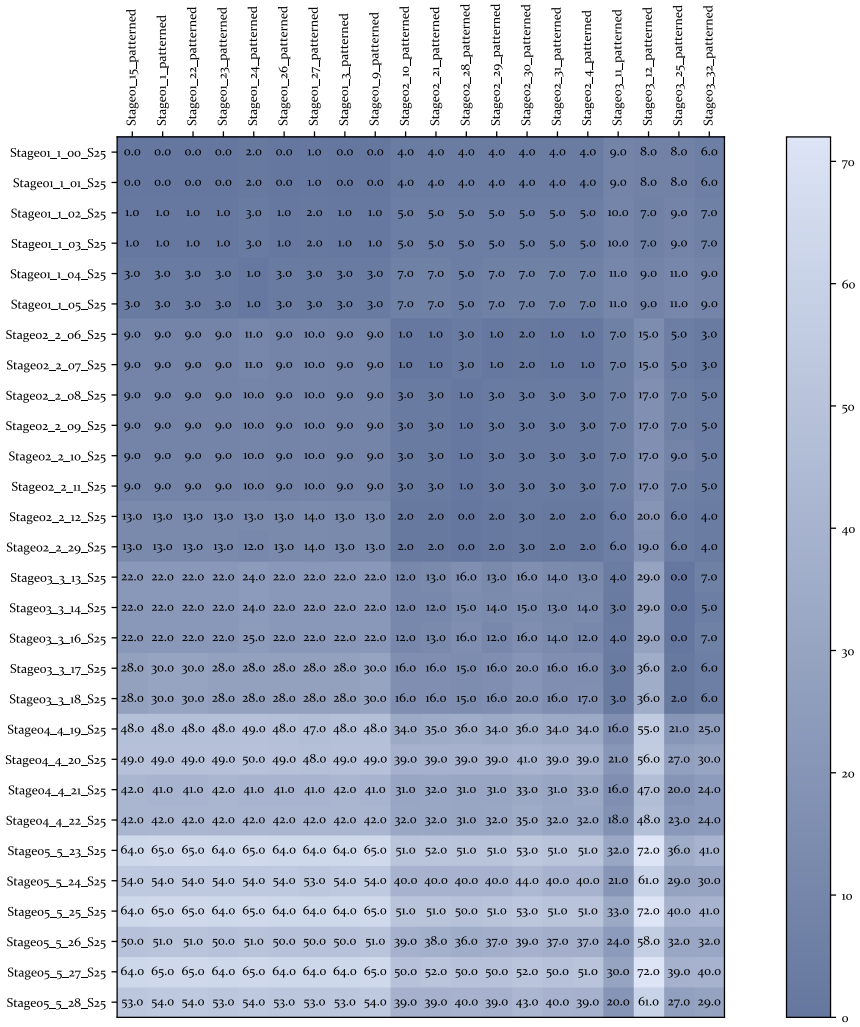
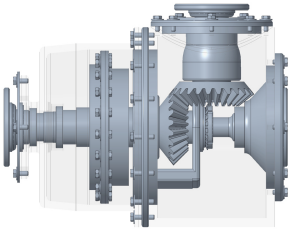
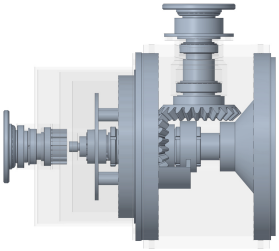
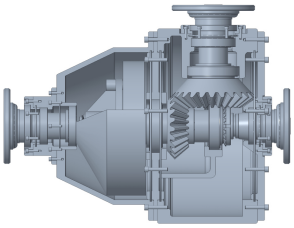
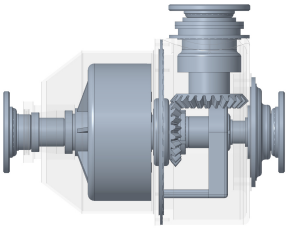
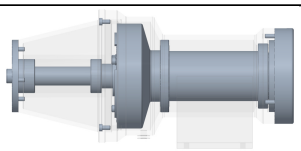
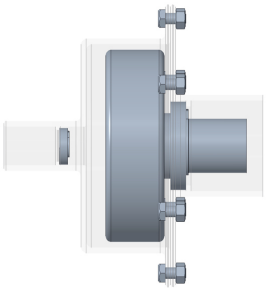
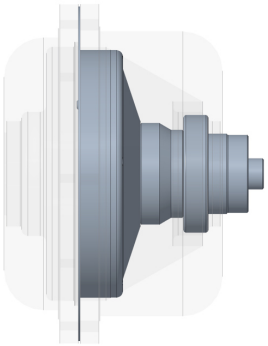
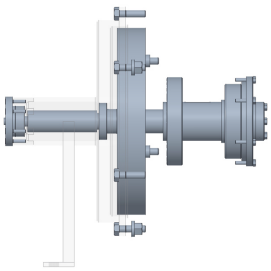


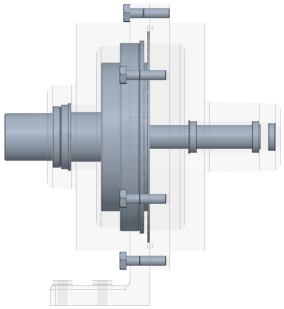
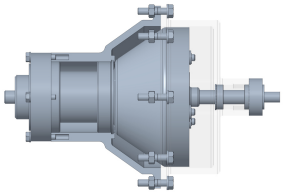
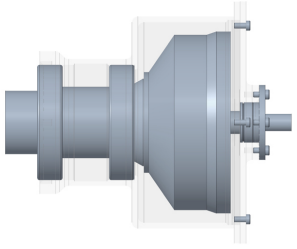
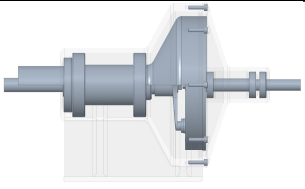
Bild 105: Wahrheitsmatrix für den GED-Score

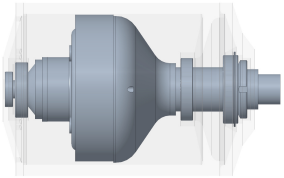
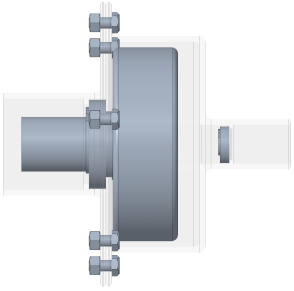
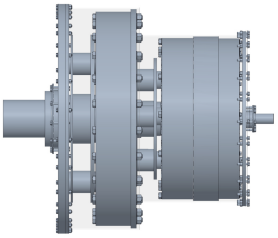
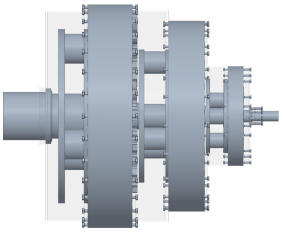
C.5 Demonstrator-Datensatz - Baugruppe

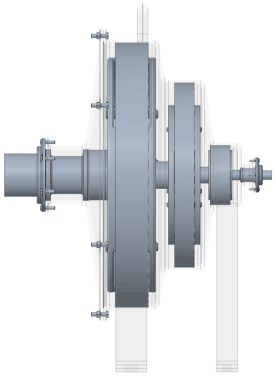
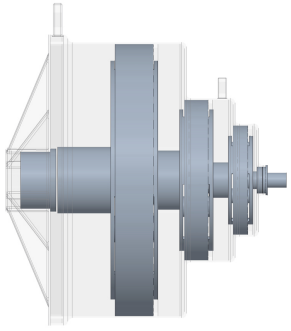
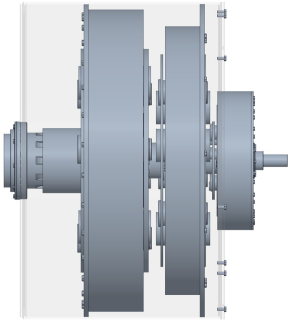
Tabelle 20: Übersicht aller Baugruppen des Assembly-Datensatzes

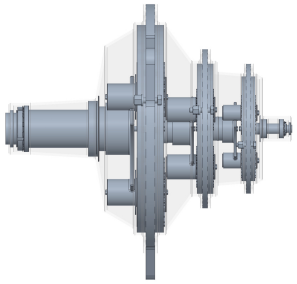
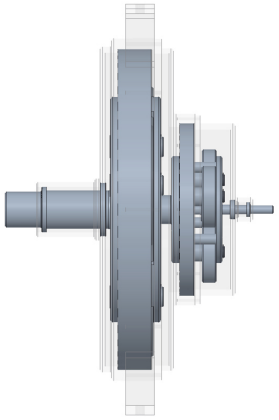
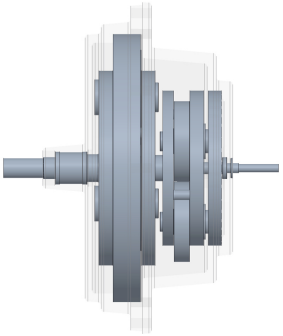
Bauteilklasse	Bezeichnung	Bild
Differential-01	2017-1-DIFFERENTIEL-ACHOU-BENASSON-BENRIDA-343-PARTS	
Differential-03	2017-1-DIFFERENTIEL-COSTE-ELDACHRI-CHAVIGNOT-209-PARTS	
Differential-04	2017-1-DIFFERENTIEL-LATETE-MALARD-MARTIN-246-PARTS	
Differential-05	2017-1-DIFFERENTIEL-LATHUILLE-GIODA-FRANCHETEAU-170-PARTS	

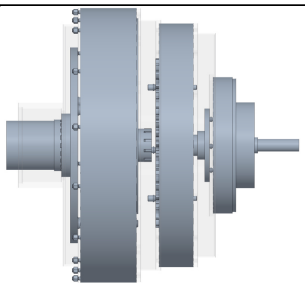
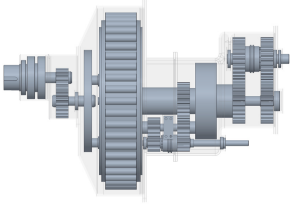
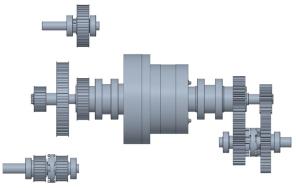
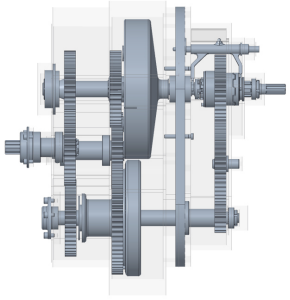
Double Rotor-05	2009-EOLIENNE-DOUBLE-ROTORS-CARVI-CLARION-FLORY-60-PARTS	
Double Rotor-06	2009-EOLIENNE-DOUBLE-ROTORS-CAZAUX-CORRECHER-44-PARTS	
Double Rotor-07	2009-EOLIENNE-DOUBLE-ROTORS-COMITI-DJEDIAT-20-PARTS	
Double Rotor-10	2009-EOLIENNE-DOUBLE-ROTORS-DE-KUYPERT-PIASCO-63-PARTS	

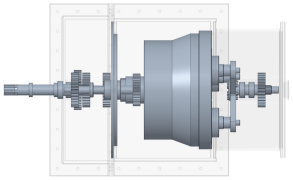
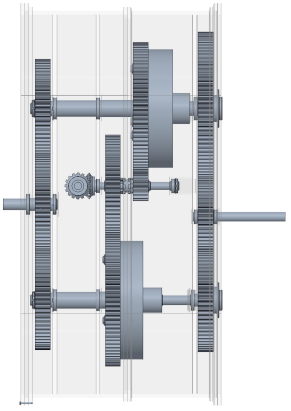
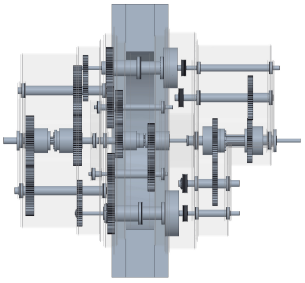
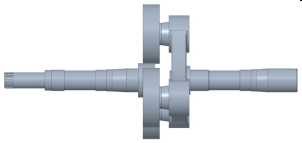
Double Rotor-11	2009-EOLIENNE-DOUBLE-ROTORS-DELUCA-POULY-103-PARTS	
Double Rotor-12	2009-EOLIENNE-DOUBLE-ROTORS-DEPARIS-CATANIA-71-PARTS	
Double Rotor-13	2009-EOLIENNE-DOUBLE-ROTORS-DUPONT-LEVY-42-PARTS	
Double Rotor-15	2009-EOLIENNE-DOUBLE-ROTORS-LESELLIER-GIACONIA-54-PARTS	

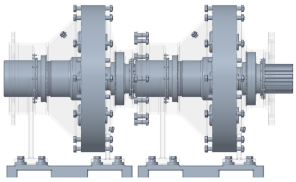
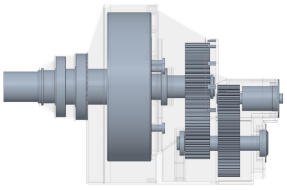
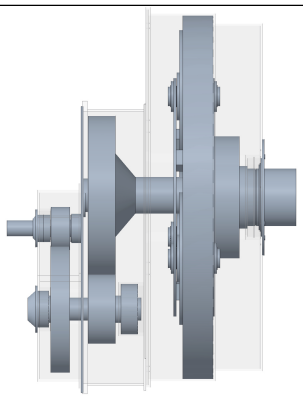
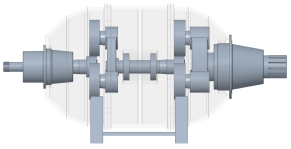
Double Rotor-16	2009-EOLIENNE-DOUBLE-ROTORS-PELENC-PRALONG-43-PARTS	
Double Rotor-17	2009-EOLIENNE-DOUBLE-ROTORS-VAYABOURY-DELESTRADE-44-PARTS	
Hydraulic Reduction-01	2010-REDUCTEUR-HYDROLIENNE-AGHER-FECOURT-HURTES-404-PARTS	
Hydraulic Reduction-02	2010-REDUCTEUR-HYDROLIENNE-ASSAIR-GODET-ROCHE-198-PARTS	

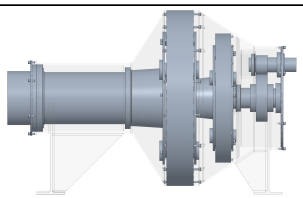
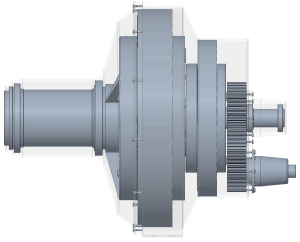
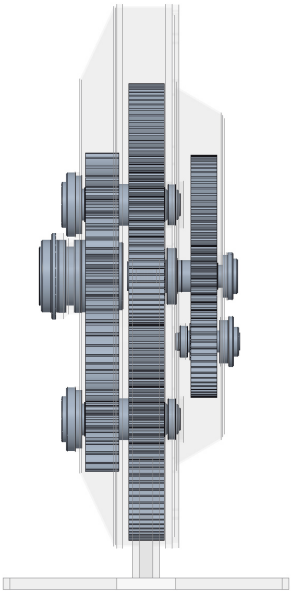
Hydraulic Reduction-03	2010-REDUCTEUR-HYDROLIENNE-BAGARRY-DE-SANTA-NICAISE-88-PARTS	
Hydraulic Reduction-04	2010-REDUCTEUR-HYDROLIENNE-BISACCIA-CELOUDOUX-CHOQUER-35-PARTS	
Hydraulic Reduction-05	2010-REDUCTEUR-HYDROLIENNE-BODA-SELLAM-STOIANOFF-PICLET-186-PARTS	

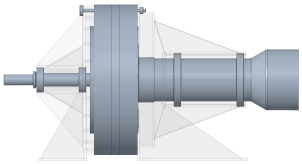
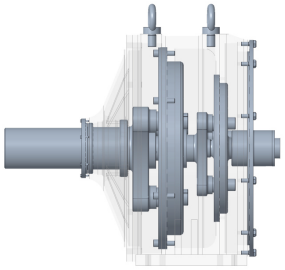
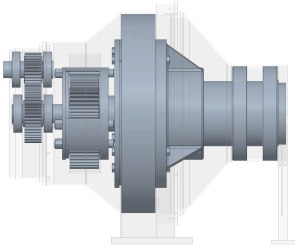
Hydraulic Reduction-06	2010-REDUCTEUR-HYDROLIENNE-BRELIVET-GHOLAMI-PRALONG-107-PARTS	
Hydraulic Reduction-07	2010-REDUCTEUR-HYDROLIENNE-GIRAUD-BASTONE-PIANCA-58-PARTS	
Hydraulic Reduction-08	2010-REDUCTEUR-HYDROLIENNE-HENRY-PERALDI-VARNET-TEYBER-40-PARTS	

Hydraulic Reduction-09	2010-REDUCTEUR-HYDROLIENNE-MONTIGNY-MUSELET-ESPIEUX-54-PARTS	
Hydraulic Rotor-01	2016-BOITE-TRANSFERT-BENHAGOUG-CHALEARD-CHANFRAY-RAOUL-281-PARTS	
Hydraulic Rotor-02	2016-BOITE-TRANSFERT-BLACHE-CHARILLON-DEVALETTE-FLANDIN-92-PARTS	
Hydraulic Rotor-04	2016-BOITE-TRANSFERT-CAPET-GROTARD-LUCK-MENUET-MIGEON-PENCENAT-125-PARTS	

Hydraulic Rotor-06	2016-BOITE-TRANSFERT-FANGAREZZI-MARCHINI-RAVAIL-THEVALIL-71-PARTS	
Hydraulic Rotor-07	2016-BOITE-TRANSFERT-GUMILAR-MARTINELLI-POUPAUD-TAVERNE-138-PARTS	
Hydraulic Rotor-08	2016-BOITE-TRANSFERT-MONTIEL-NIVAGGIOLI-PODEVIN-ROCHE-ROUSSEAU-175-PARTS	
Wind Turbine-01	2008-BOYER-GANIARIS-14Part	

Wind Turbine-05	2008 -MULTIPLICATEUR-EOLIENNE-AUBOURG-VAUTRIN-212-PARTS	
Wind Turbine-07	2008-MULTIPLICATEUR-EOLIENNE-BACUZZI-MARTINEZ-66-PARTS	
Wind Turbine-08	2008-MULTIPLICATEUR-EOLIENNE-BIAMONTI-OUAKNINE-88-PARTS	
Wind Turbine-09	2008-MULTIPLICATEUR-EOLIENNE-BOYER-GANIARIS-38-PARTS	

Wind Turbine-13	2008-MULTIPLICATEUR-EOLIENNE-DUROCHER-PETRUZELLA-128-PARTS	
Wind Turbine-14	2008-MULTIPLICATEUR-EOLIENNE-GAL-TOMASSINI-119-PARTS	
Wind Turbine-18	2008-MULTIPLICATEUR-EOLIENNE-MARKARIAN-LEPLUS-63-PARTS	

Wind Turbine-19	2008 -MULTIPLICATEUR- EOLIENNE- MARTINOD- SERRA- 29-PARTS	
Wind Turbine-20	2008- MULTIPLICATEUR- EOLIENNE-MICHEL- YEUNG- 78-PARTS	
Wind Turbine-22	2008- MULTIPLICATEUR- EOLIENNE-OLIVE- ARBONA- 90-PARTS	

C.6 Vergleich Bauteilerkennung

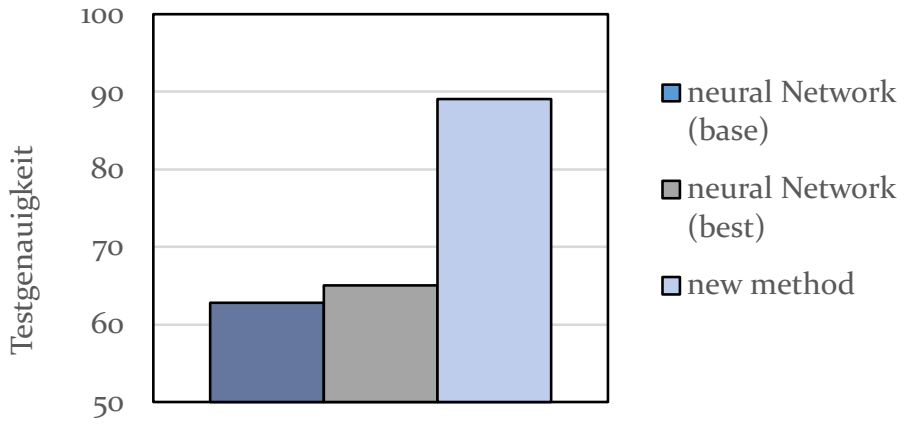


Bild 106: Ergebnis für den Vergleich zwischen neuer Methode und bestehendem Verfahren [231] nach [P7]

C.7 Vergleich Projektion - Stand der Technik

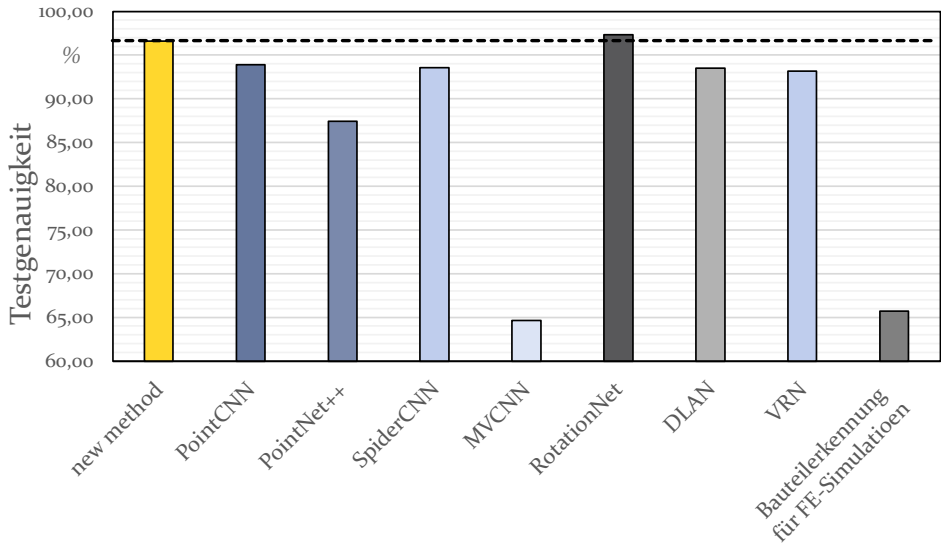


Bild 107: Vergleich der Methode mit anderen aus dem MCB-Datensatz nach [P7]

C.8 Ergebnis automatisierte Bauteilerkennung

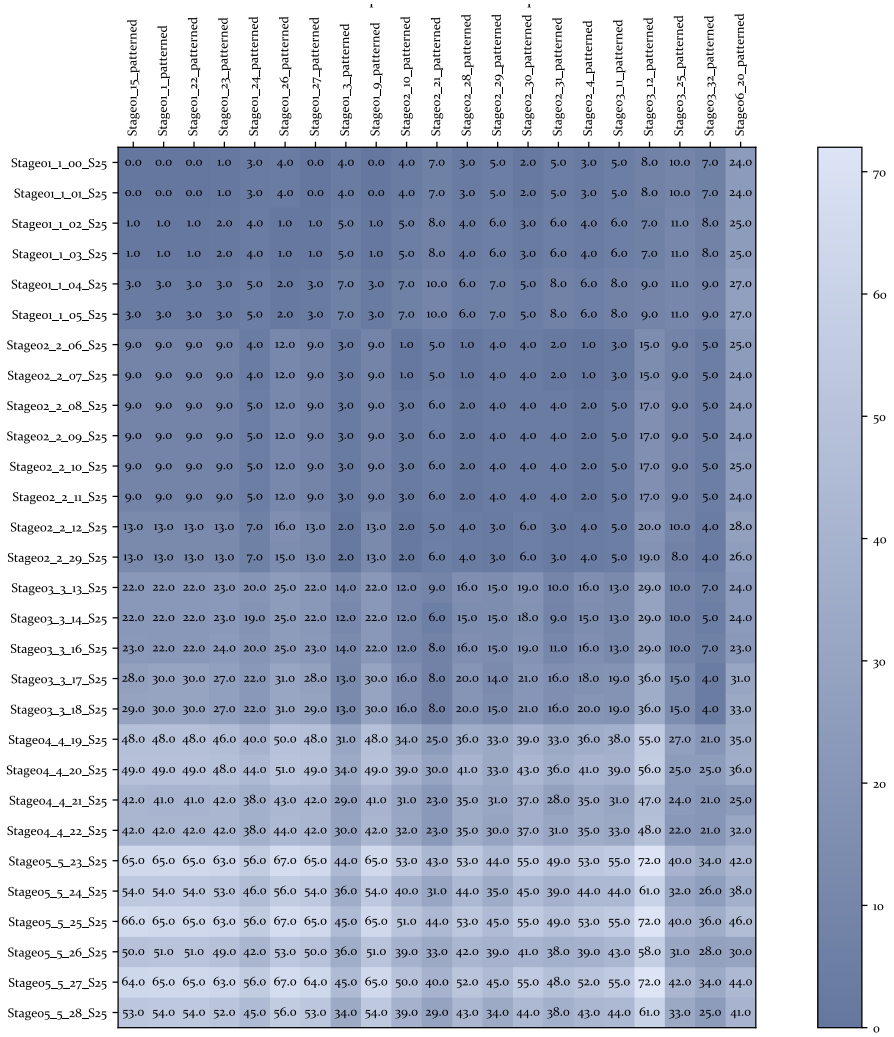


Bild 108: Wahrheitsmatrix für den GED-Score

C.9 Regel für Baugruppen

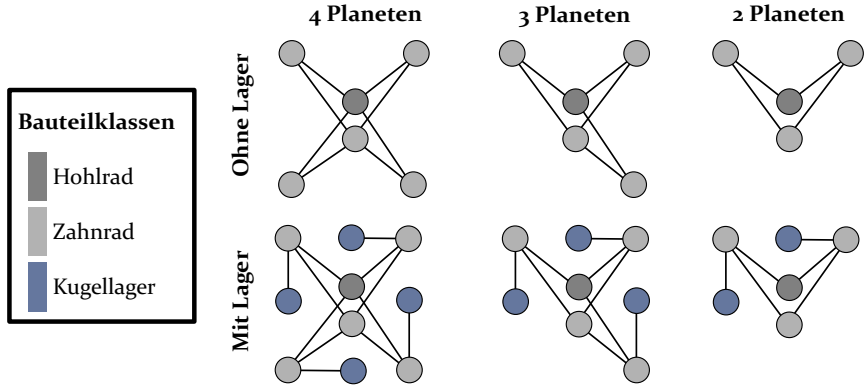


Bild 109: Darstellung der Regeln für die Reduktion der Planetenräder im Baugruppen-Graph nach [S9]

Tabelle 21: Übersicht der angewendeten Regel für den Baugruppen-Datensatz

Baugruppe	**Änderung**	**Bauteil 1**	**Bauteil 2**
DoubleRotor-Turbine_15	Kante einfügen	P-00010727-COURONNE_3	P-00010723-SATELLITE_2
DoubleRotor-Turbine_15	Kante einfügen	ROUESAT	P-00010723-SATELLITE_2
DoubleRotor-Turbine_16	Kante einfügen	P-00009862-COURONNE	P-00011415-SATELLITE
DoubleRotor-Turbine_16	Kante einfügen	P-00012423-PIGNON	P-00011415-SATELLITE
DoubleRotor-Turbine_17	Kante einfügen	P-00009064-ENGRENAGE_SORTIE_1	P-00009076-ARBRE_PIGNON_1#2
DoubleRotor-Turbine_17	Kante einfügen	COURONNE	P-00009076-ARBRE_PIGNON_1
HydraulicReduction_3	Kante einfügen	PRT_PIGNON_1_TRAIN_1	SATELLITE#1
HydraulicReduction_4	Kante einfügen	PLANETAIRE_TRAIN_1	SATELLITE_1
HydraulicReduction_4	Kante einfügen	PLANETAIRE_TRAIN_2	SATELLITE_2

HydraulicReduction_4	Kante einfügen	ROUE	SATELLITE_3
HydraulicReduction_5	Kante einfügen	COURONNE_2	ROUE
HydraulicReduction_5	Kante einfügen	PIGNON_3	ROUE#1
HydraulicReduction_6	Kante einfügen	PART_COURONNE_II	PART_SATELLITE_II
HydraulicReduction_7	Kante einfügen	ROUE_1	SATELLITE_2_1_1
HydraulicReduction_7	Kante einfügen	ROUE	SATELLITE_1_1_1
HydraulicReduction_8	Kante einfügen	ROUE	SATELLITE_2_1
HydraulicReduction_8	Kante einfügen	PIGNION_PLANETAIRE_1_1	SATELLITE_1_1
HydraulicReduction_9	Kante einfügen	VRAI_COURONNE_2_1_1_1	ROUE
HydraulicRotor_6	Kante einfügen	ROUE_3	FANGAREZZI_CPD_SATELLITE_PETIT_
HydraulicRotor_8	Kante einfügen	PIECE_ARBRE6_ROUE_D_—	PIECE_ARBRE8_ROUEDROITE_—
HydraulicRotor_8	Kante einfügen	ASS_SYM_ARBRE6PIECE_ARBRE6_ROUE	PIECE_ARBRE8_ROUEGAUCHE_—
RotorWindTurbin_8	Kante einfügen	ROUE	ROUE4
RotorWindTurbin_8	Kante einfügen	ROUE2	ROUE
RotorWindTurbin_8	Kante einfügen	COURONNE	ROUE4#2
RotorWindTurbin_9	Kante einfügen	ROUE3#1	ROUE
RotorWindTurbin_9	Kante einfügen	ROUE3#1	ROUE#2
RotorWindTurbin_9	Kante einfügen	ROUE3#1	ROUE#1

RotorWindTurbin_9	Kante einfügen	ROUE3	ROUE2
RotorWindTurbin_13	Kante einfügen	ROUE5	ROUE
RotorWindTurbin_13	Kante einfügen	ROUE5	ROUE#2
RotorWindTurbin_13	Kante einfügen	ROUE5	ROUE#1
RotorWindTurbin_13	Kante einfügen	ROUE2	ROUE7
RotorWindTurbin_13	Kante einfügen	ROUE6	ROUE3
RotorWindTurbin_14	Kante einfügen	ROUE5	ROUE6
RotorWindTurbin_19	Kante einfügen	ROUE2	ROUE
RotorWindTurbin_19	Kante einfügen	ROUE2	ROUE#2
RotorWindTurbin_19	Kante einfügen	ROUE2	ROUE#1
RotorWindTurbin_19	Kante einfügen	COURONNE#1	ROUE#5
RotorWindTurbin_19	Kante einfügen	ROUE6	ROUE#3
RotorWindTurbin_19	Kante einfügen	ROUE6	ROUE#4
RotorWindTurbin_19	Kante einfügen	ROUE6	ROUE#5
RotorWindTurbin_20	Kante einfügen	ROUE9	ROUE8
RotorWindTurbin_20	Kante einfügen	ROUE11	ROUE10
RotorWindTurbin_20	Kante einfügen	COURONNE8	ROUE10

C.10 Scores für Baugruppen-Datensatz

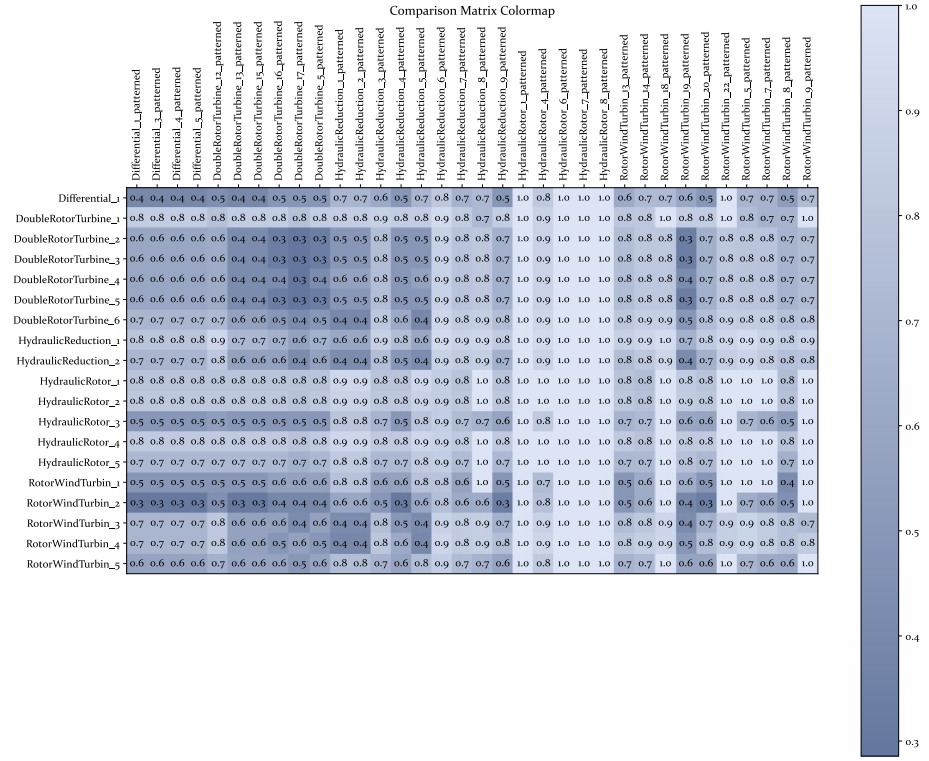


Bild 110: Wahrheitsmatrix für den MCS-Score

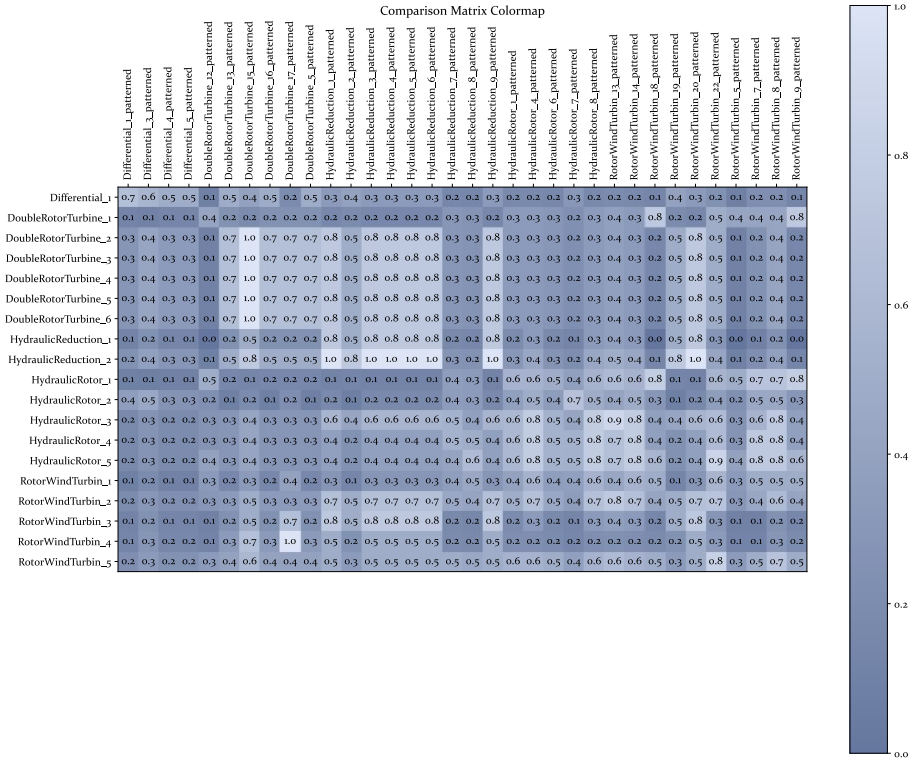


Bild 111: Wahrheitsmatrix für die Jaccard-Ähnlichkeit

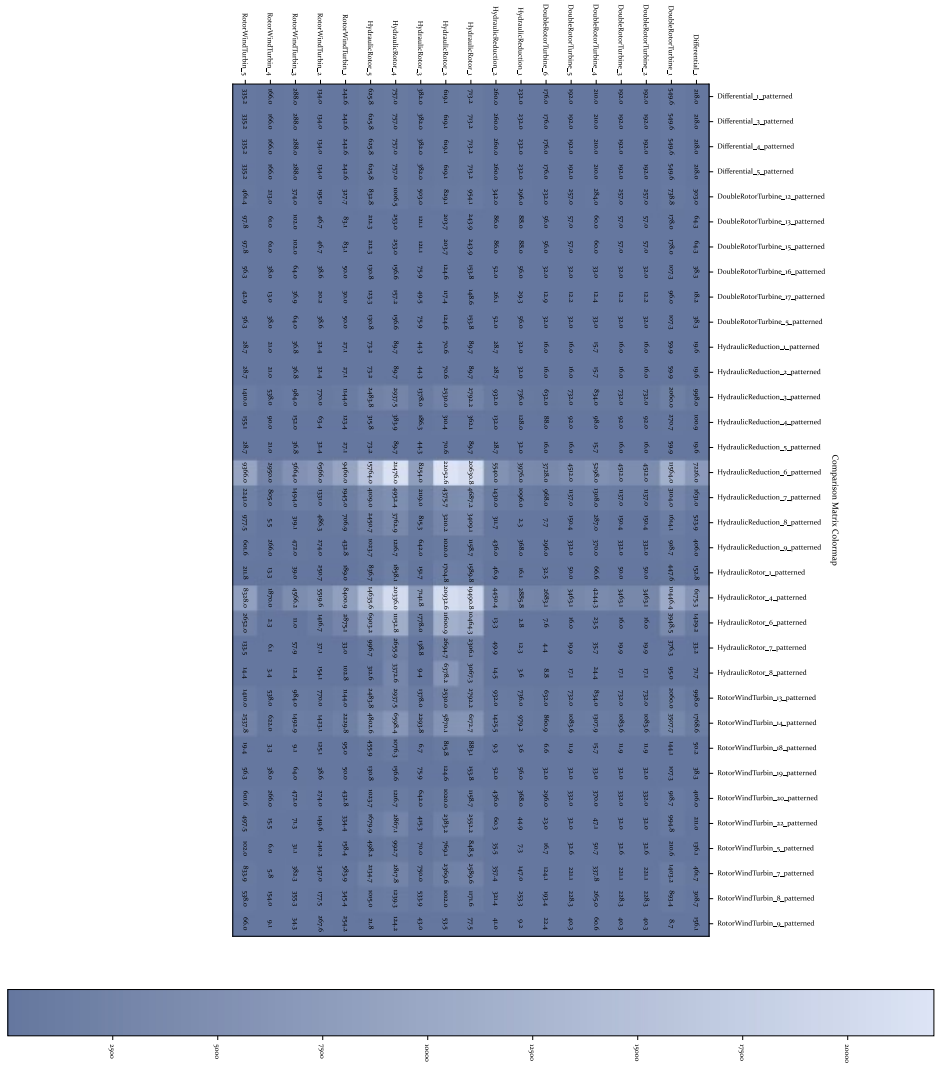


Bild 112: Wahrheitsmatrix für den Eigenvektor-Score

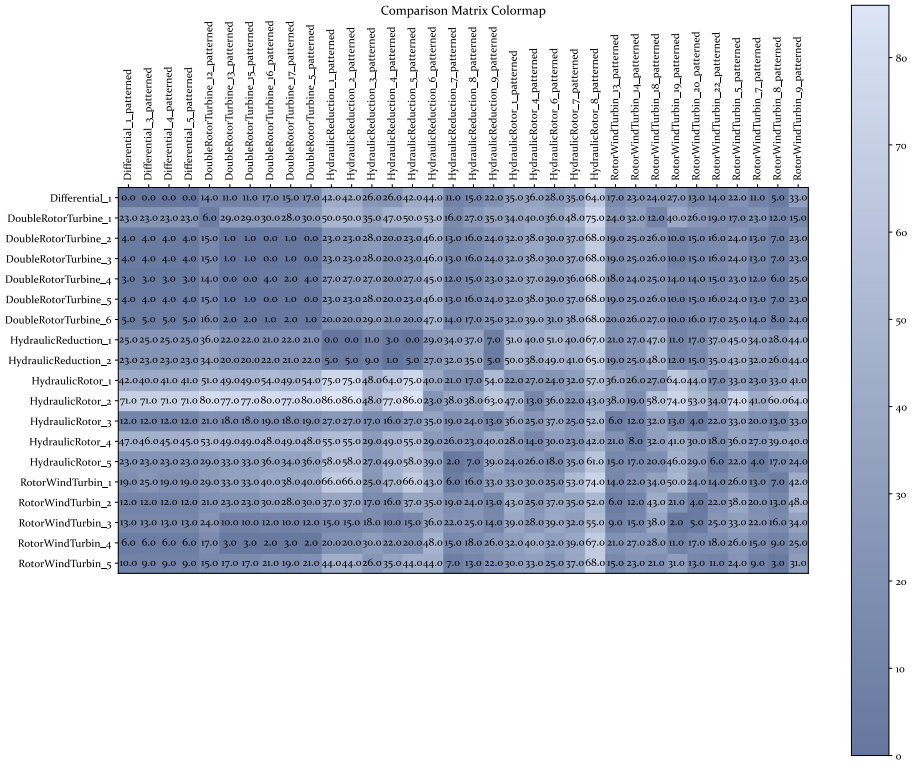


Bild 113: Wahrheitsmatrix für den GED-Score

D Ergänzungen zu Kapitel 6

D.1 Ähnlichkeitssuche Bauteile

Algorithm 3: Allgemeiner Ablauf des Algorithmus nach [P10]

Data: Faces and Vertices of imported Part
Result: Get largest combined Faces in Mesh
[$faces, vertices$] = `importMesh(`$filename$`)`;
[$VNew$] = `RotatePartToLargestFace(`$faces, vertices$`)`;
[$BBVert$,$BBFaces$] = `BoundingBox(`$faces$,$Vnew$`)`;
$IdxNormalBB$ = `normals(`$BBVert$,$BBFaces$`)` == $NormalTo$;
$AreaBB$ = sum(`AreaFaces(`$BBVert$, $BBFaces(IdxNormalBB)$`)`);
if $AreaBB/faceMaxArea >$ 0.12 **then**
    $VAligned$ = `alignPCA(`$vertices$,$faces$`)`;
else
    $DirTo$ = `getLargestEdge(`$Input$`)` ;
    $VAligned$ = `Rotated3D(`$VNew$,$DirTo$,[1,0,0]`)`;
[Ixx, Iyy] = `polySecAreaMoments(`$VAligned$`)`;
[$DistPolar$] = `polySignature(`$VAligned$,10`)`;
if $Ixx/Iyy ==$ 1 & `max(`$DistPolar$`)`−`min(`$DistPolar$`)` < 0.3 **then**
    $VAligned$ = `alignPCA(`$faces$,[$VAligned(1:2)$,0]`)`;
`checkXAlignment(`$VAligned$`)`;

Algorithm 4: Unterstützungsalgorithmen für die Teileausrichtung nach [P10]

```
Function RotatePartToLargestFace(faces, vertices):
    normals = GetFaceNormals(faces);
    (normalsAll,~,IC) = unique(normals);
    for n ∈ normalsAll do
        facesNormal = faces(n == IC);
        adjMatrix = AdjacencyMatrix(facesNormal);
        connGraph = ConnComponents(Graph(adjMatrix));
        forall connGraph with size(connGraph)> 2 do
            fSameN = GetFaces(connGraph);
            areaConn = sum(AreaFaces(vertices,fSameN));
        maxAreaNormals = max(areaConn);
    [faceMaxArea,locMaxArea] = max(maxAreaNormals);
    facesLargArea = fSameN(locMaxArea);
    VNew = Rotated3D(vertices,[0, 0, −1],normalsAll(locMaxArea)); return
     VNew
 Function getLargestEdge(facesLargArea):
    adjEdgeMatrix = AdjaEdgeCostMat(vertices, outline(facesLargArea));
    connGraphEdge = ConnComponents(Graph(adjEdgeMatrix));
    forall connGraphEdge with size(connGraphEdge)> 2 do
        cycles = cyclebasis(connGraphEdge);
        polyin = polyshapeAdd(VerticesCycle);
    Vpoly = sortboundaries(polyin);
    for i ∈ polyVertices do
        distance = dist(Vpoly,Vpoly(i + 1));
        direction = normalize(Vpoly(i + 1) − Vpoly(i));
    [DistMax,locDistMax] = max(distance);
    DirEdgeMax = [direction(locDistMax,1:2), 0];
    return DirEdgeMax

Function CheckXAlignment(vertices,faces):
    XFar = vertices(vertices(:, 1) ==max(vertices(:, 1)),3);
    XNear = vertices(vertices(:, 1) ==min(vertices(:, 1)),3);
    if dist(max(XFar) - min(XFar)) > dist(max(XNear) - min(XNear)) then
        Valgined=vertices∗rotZ(180);
    IdxComp = vertices(:, 1) == min(vertices(:, 1));
    if max(vertices(:, 3))>max(vertices(IdxComp)) then
        IdxXZMax = vertices(:, 3), 2) ==max(vertices(:, 3));
        VXMaxZ = vertices(IdxXZmax, 1);
        DistMin = dist(min(vertices(:,1)),VXMaxZ);
        DistMax = dist(max(vertices(:,1)),VXMaxZ);
        if DistMin > DistMax then
            Valgined=vertices∗rotZ(180);
    if max(XFar) == max(XNear) then
        AreasMax = AreaFaces(faces(IdxMax));
        AreasMin = AreaFaces(faces(IdxMin));
        if sum(AreasMax) > sum(AreasMin) then
            Valgined=vertices∗rotZ(180);
    return Valgined
```

D.1.1 Einfluss der Anzahl an Punkten pro Sampling

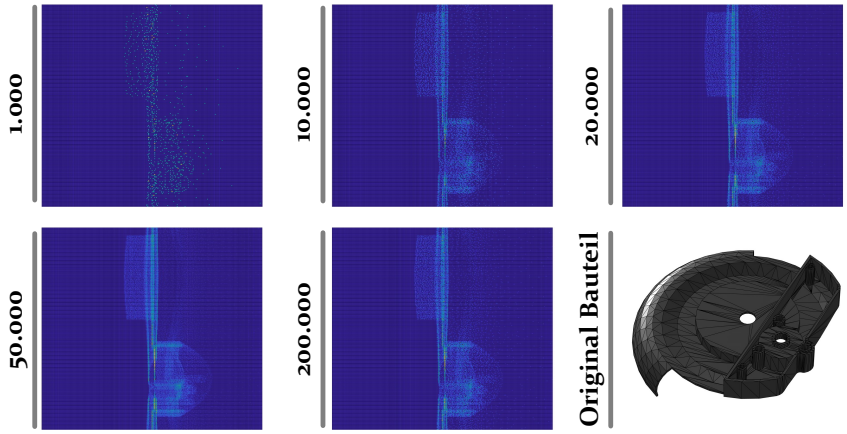


Bild 114: Beispiele für unterschiedlichen Sampling-Punkte und deren Auswirkung auf die erzeugten Matrizen nach [P10]

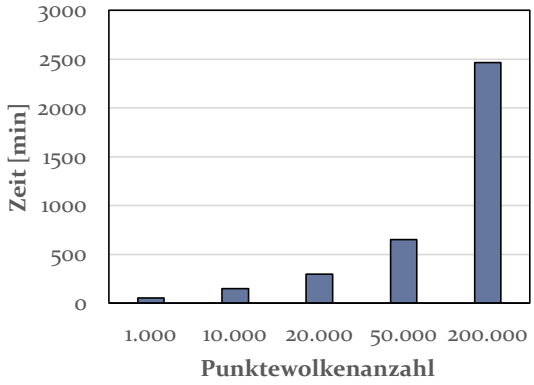


Bild 115: Zeitbedarf für die unterschiedlichen Sampling-Punkte für Punktewolken nach [P10]

D.1.2 Übersicht Studie Dimensionreduktionsgröße

Hyperparameter für Autoencoder:
Transfer Function: log-sig
Sparsity Proportion: 0.05
L2 Weight Regularization: 0.0001
Max Epochs: 1000

Tabelle 22: Ergebnisse der Dimensionsreduktionsstudie für unterschiedliche Verfahren

Reduktions-Nummer	Verfahren	View-Auswahl	NN	FT	ST	DCG
15	'AE'	24	0,865	0,526	0,69	0,824
15	'AE'	8	0,867	0,528	0,691	0,826
15	'AE'	4	0,853	0,516	0,681	0,818
15	'AE'	1	0,822	0,445	0,617	0,777
20	'AE'	24	0,864	0,524	0,689	0,823
20	'AE'	8	0,859	0,526	0,69	0,825
20	'AE'	4	0,864	0,524	0,684	0,824
20	'AE'	1	0,812	0,447	0,611	0,777
20	'AE'	24	0,855	0,526	0,686	0,823
22	'AE'	24	0,859	0,526	0,685	0,823
25	'AE'	24	0,864	0,523	0,686	0,822
25	'AE'	8	0,865	0,53	0,689	0,825
25	'AE'	4	0,871	0,526	0,688	0,825
25	'AE'	1	0,83	0,483	0,652	0,799
30	'AE'	24	0,872	0,521	0,684	0,821
30	'AE'	8	0,868	0,529	0,69	0,826
30	'AE'	4	0,858	0,519	0,682	0,821
30	'AE'	1	0,848	0,514	0,681	0,816
35	'AE'	24	0,872	0,519	0,68	0,82
35	'AE'	8	0,868	0,529	0,689	0,825
35	'AE'	4	0,871	0,53	0,689	0,827
35	'AE'	1	0,86	0,519	0,677	0,816
40	'AE'	24	0,866	0,516	0,674	0,817
40	'AE'	8	0,866	0,528	0,688	0,825
40	'AE'	4	0,864	0,528	0,687	0,826
40	'AE'	1	0,867	0,521	0,686	0,821
45	'AE'	24	0,865	0,515	0,674	0,817
45	'AE'	8	0,863	0,526	0,685	0,823

45	'AE'	4	0,863	0,529	0,687	0,826
45	'AE'	1	0,853	0,519	0,676	0,818
50	'AE'	24	0,87	0,513	0,67	0,816
50	'AE'	8	0,864	0,523	0,681	0,822
50	'AE'	4	0,861	0,527	0,687	0,826
50	'AE'	1	0,839	0,501	0,669	0,811
60	'AE'	24	0,872	0,511	0,666	0,814
60	'AE'	8	0,872	0,52	0,677	0,82
60	'AE'	4	0,871	0,528	0,683	0,825
60	'AE'	1	0,864	0,517	0,677	0,82
70	'AE'	24	0,868	0,51	0,66	0,811
70	'AE'	8	0,867	0,518	0,669	0,817
70	'AE'	4	0,874	0,528	0,684	0,826
70	'AE'	1	0,874	0,524	0,686	0,823
80	'AE'	24	0,867	0,508	0,658	0,81
80	'AE'	8	0,868	0,515	0,663	0,814
80	'AE'	4	0,87	0,523	0,68	0,823
80	'AE'	1	0,871	0,526	0,688	0,825
90	'AE'	24	0,87	0,507	0,656	0,809
90	'AE'	8	0,868	0,511	0,657	0,812
90	'AE'	4	0,868	0,521	0,678	0,821
90	'AE'	1	0,871	0,523	0,683	0,824
100	'AE'	24	0,868	0,505	0,651	0,807
100	'AE'	8	0,866	0,506	0,654	0,81
100	'AE'	4	0,873	0,519	0,671	0,82
100	'AE'	1	0,876	0,519	0,685	0,824
200	'AE'	24	0,864	0,495	0,634	0,799
200	'AE'	8	0,864	0,487	0,617	0,793
200	'AE'	4	0,871	0,501	0,635	0,804
200	'AE'	1	0,873	0,527	0,684	0,826
300	'AE'	24	0,863	0,492	0,624	0,795
300	'AE'	8	0,864	0,481	0,603	0,787
300	'AE'	4	0,867	0,486	0,614	0,793
300	'AE'	1	0,871	0,522	0,676	0,823
400	'AE'	24	0,864	0,491	0,621	0,794
400	'AE'	8	0,863	0,478	0,6	0,785
400	'AE'	4	0,867	0,483	0,608	0,791
400	'AE'	1	0,876	0,519	0,672	0,822

15	'PCA'	8	0,86	0,512	0,678	0,818
15	'PCA'	4	0,86	0,511	0,676	0,817
15	'PCA'	1	0,852	0,51	0,674	0,815
20	'PCA'	8	0,865	0,515	0,678	0,82
20	'PCA'	4	0,861	0,516	0,678	0,819
20	'PCA'	1	0,857	0,513	0,676	0,817
25	'PCA'	8	0,87	0,515	0,681	0,821
25	'PCA'	4	0,868	0,517	0,681	0,82
25	'PCA'	1	0,863	0,514	0,676	0,819
30	'PCA'	8	0,868	0,515	0,681	0,821
30	'PCA'	4	0,87	0,515	0,682	0,821
30	'PCA'	1	0,868	0,515	0,678	0,82
35	'PCA'	8	0,866	0,516	0,68	0,821
35	'PCA'	4	0,867	0,514	0,68	0,821
35	'PCA'	1	0,865	0,515	0,678	0,82
40	'PCA'	8	0,864	0,515	0,68	0,821
40	'PCA'	4	0,867	0,514	0,68	0,821
40	'PCA'	1	0,865	0,512	0,678	0,82
45	'PCA'	8	0,864	0,515	0,679	0,82
45	'PCA'	4	0,864	0,512	0,681	0,82
45	'PCA'	1	0,861	0,513	0,678	0,819
50	'PCA'	8	0,864	0,514	0,679	0,82
50	'PCA'	4	0,863	0,511	0,679	0,82
50	'PCA'	1	0,859	0,511	0,676	0,819
60	'PCA'	8	0,863	0,512	0,679	0,819
60	'PCA'	4	0,86	0,511	0,678	0,819
60	'PCA'	1	0,863	0,509	0,675	0,819
70	'PCA'	8	0,863	0,511	0,676	0,819
70	'PCA'	4	0,859	0,511	0,677	0,818
70	'PCA'	1	0,86	0,508	0,675	0,818
80	'PCA'	8	0,864	0,51	0,675	0,818
80	'PCA'	4	0,859	0,51	0,675	0,818
80	'PCA'	1	0,861	0,508	0,674	0,817
90	'PCA'	8	0,863	0,511	0,673	0,818
90	'PCA'	4	0,859	0,509	0,672	0,817
90	'PCA'	1	0,86	0,506	0,673	0,817
100	'PCA'	8	0,861	0,511	0,672	0,817
100	'PCA'	4	0,861	0,509	0,67	0,817

100	'PCA'	1	0,863	0,505	0,67	0,816
200	'PCA'	8	0,864	0,508	0,665	0,814
200	'PCA'	4	0,863	0,507	0,665	0,814
200	'PCA'	1	0,866	0,505	0,664	0,813
300	'PCA'	8	0,867	0,506	0,662	0,813
300	'PCA'	4	0,866	0,507	0,662	0,813
300	'PCA'	1	0,864	0,505	0,66	0,812
400	'PCA'	8	0,87	0,506	0,66	0,812
400	'PCA'	4	0,867	0,505	0,659	0,812
400	'PCA'	1	0,866	0,504	0,658	0,812
15	't-SNE'	24	0,849	0,507	0,666	0,81
15	't-SNE'	8	0,86	0,516	0,67	0,813
15	't-SNE'	4	0,858	0,516	0,668	0,812
15	't-SNE'	1	0,849	0,515	0,669	0,809
20	't-SNE'	24	0,849	0,507	0,666	0,81
20	't-SNE'	8	0,86	0,516	0,67	0,814
20	't-SNE'	4	0,857	0,516	0,668	0,812
20	't-SNE'	1	0,849	0,516	0,669	0,81
25	't-SNE'	24	0,85	0,507	0,666	0,81
25	't-SNE'	8	0,86	0,516	0,67	0,814
25	't-SNE'	4	0,857	0,516	0,668	0,812
25	't-SNE'	1	0,85	0,515	0,669	0,81
30	't-SNE'	24	0,849	0,507	0,665	0,81
30	't-SNE'	8	0,86	0,516	0,67	0,814
30	't-SNE'	4	0,857	0,516	0,668	0,812
30	't-SNE'	1	0,851	0,516	0,67	0,81
35	't-SNE'	24	0,85	0,506	0,666	0,81
35	't-SNE'	8	0,86	0,517	0,67	0,814
35	't-SNE'	4	0,857	0,516	0,668	0,812
35	't-SNE'	1	0,851	0,515	0,669	0,81
40	't-SNE'	24	0,849	0,507	0,665	0,81
40	't-SNE'	8	0,86	0,517	0,67	0,814
40	't-SNE'	4	0,857	0,516	0,669	0,812
40	't-SNE'	1	0,851	0,515	0,669	0,81
45	't-SNE'	24	0,849	0,507	0,665	0,81
45	't-SNE'	8	0,859	0,517	0,67	0,813
45	't-SNE'	4	0,857	0,516	0,668	0,812
45	't-SNE'	1	0,85	0,516	0,67	0,81

50	't-SNE'	24	0,849	0,507	0,665	0,81
50	't-SNE'	8	0,86	0,517	0,67	0,814
50	't-SNE'	4	0,857	0,516	0,668	0,812
50	't-SNE'	1	0,85	0,515	0,669	0,81
60	't-SNE'	24	0,849	0,507	0,665	0,81
60	't-SNE'	8	0,86	0,517	0,67	0,814
60	't-SNE'	4	0,856	0,516	0,668	0,812
60	't-SNE'	1	0,85	0,515	0,669	0,81
70	't-SNE'	24	0,849	0,507	0,666	0,81
70	't-SNE'	8	0,86	0,517	0,67	0,814
70	't-SNE'	4	0,858	0,516	0,668	0,812
70	't-SNE'	1	0,85	0,516	0,67	0,81
80	't-SNE'	24	0,848	0,507	0,665	0,81
80	't-SNE'	8	0,86	0,517	0,67	0,814
80	't-SNE'	4	0,857	0,516	0,668	0,812
80	't-SNE'	1	0,851	0,516	0,67	0,81
90	't-SNE'	24	0,849	0,506	0,666	0,81
90	't-SNE'	8	0,86	0,517	0,67	0,814
90	't-SNE'	4	0,859	0,516	0,669	0,812
90	't-SNE'	1	0,85	0,515	0,67	0,81
100	't-SNE'	24	0,849	0,507	0,666	0,81
100	't-SNE'	8	0,86	0,517	0,67	0,814
100	't-SNE'	4	0,858	0,516	0,668	0,812
100	't-SNE'	1	0,85	0,516	0,669	0,81
200	't-SNE'	24	0,848	0,507	0,665	0,81
200	't-SNE'	8	0,86	0,517	0,671	0,814
200	't-SNE'	4	0,857	0,516	0,668	0,812
200	't-SNE'	1	0,851	0,515	0,67	0,81
300	't-SNE'	24	0,849	0,507	0,665	0,81
300	't-SNE'	8	0,86	0,517	0,67	0,814
300	't-SNE'	4	0,857	0,516	0,668	0,812
300	't-SNE'	1	0,851	0,515	0,67	0,81
400	't-SNE'	24	0,85	0,507	0,666	0,81
400	't-SNE'	8	0,86	0,517	0,67	0,814
400	't-SNE'	4	0,857	0,515	0,668	0,812
400	't-SNE'	1	0,851	0,515	0,67	0,81

D.1.3 Siamese Network

Eine konkrete Umsetzung des Metric-Learnings ist das Siamese Network [908, 909] oder auch Tripet Network [910]. Im Falle des Siamese Networks werden zwei Bilder jeweils einem Neuronalen Netz übergeben, wobei sich beide Netze die Gewichte teilen. Danach sind beide Neuronalen Netze über ein weiteres miteinander verknüpft, wobei der Zielwert die Distanz zwischen den beiden Input-Bildern ist. Diese wird über die jeweilige Loss-Funktion abgebildet, wobei auf viele mögliche Berechnungsvarianten [468] zurückgegriffen werden kann.

Da für die gegebene Problemstellung keine Label zur Verfügung stehen, sollen die Ergebnis-Label des entwickelten Ansatzes als Eingabe genutzt werden. Dazu wird generell definiert, dass die ersten zehn Ergebnisse als „ähnlich“ eingestuft werden und die restlichen Geometrien das Label „unähnlich“ erhalten. Das Vorgehen ist in Bild 116 dargestellt, wobei die Ergebnis-Label als „synthetische Label“ bezeichnet werden, da sie nicht aus dem Original-Datensatz entnommen sind.

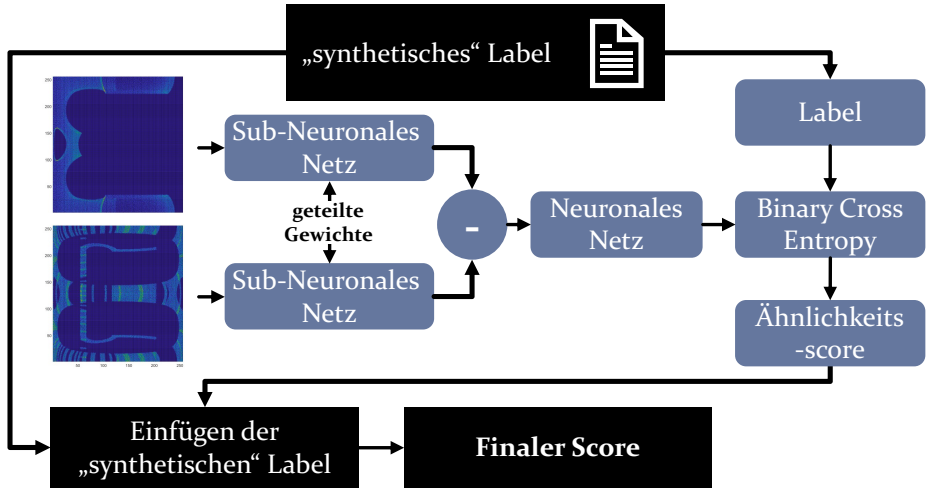


Bild 116: Vorgehen zur Nutzung der ermittelten Label für das Training eines Siamese Networks

Nach dem erfolgreichen Training werden dann die ursprünglichen Ergebnis-Label wiederum eingefügt, allerdings nur die Anzahl, die auch vorher als „ähnlich“ definiert wurde. Durch die zusätzliche Anwendung dieses Vorgehens konnte die Erkennungsgenauigkeit weiter gesteigert werden, was letztendlich folgende Ergebniswerte lieferte: NN: 0,872, FT: 0,561, ST: 0,734, DCG: 0,836

Der Aufbau des Siamese-Netzes war folgender: 2880-128-128-100 (beide Sub-Netze) // 100 für das Neuronale Netz

Der direkte Vergleich zu dem TLC+I-Pair aus der Tabelle 13 zeigt, dass der FT-Wert nicht erreicht werden konnte, dafür der ST deutlich übertroffen wurde (-0,007 vs. +0.0135). Dadurch konnte durch ein zusätzliches Modul die Erkennungsgenauigkeit gesteigert werden, allerdings auf Kosten der einfachen und universelleren Anwendbarkeit, da Zusatzparameter für Datensätze mit in das Vorgehen integriert werden müssen.

E Ergänzungen zu Kapitel 7

E.1 Datensätze

Prinzipskizzen Suche
Sucheingabe Handskizze: Neu erstellt und wie beschrieben umgewandelt.
Datenbestand Prinzipskizzen: Skizzen aus unbekannten Datensatz für Erkennungsnachweis, siehe Bilder in Kapitel B.2.3

Vorgehen: GED

Skizze-Baugruppen Suche
Sucheingabe Handskizze: Skizze aus unbekanntem Datensatz aus Kapitel B.2.3.
Datenbestand Baugruppen: 3D Assembly Dataset, beschrieben in Tabelle 20

Vorgehen: GED

Ähnlichkeitssuche Bauteile
Sucheingabe Bauteil: Bauteil P-00009318-Porte
Datenbestand Baugruppen: ESB-Datensatz 20 + 10 zufällig ausgewählte Planententräger (porte_2_1_2, porte_2_1, porte_2, porte_1_2_3_4, porte_1_2_3, porte_1_2, porte_1_1_2, porte_1_1, porte_1, porte, part3_1_1_1, p-00012064-portesatellite_1_148, p-00011397-porte_sat_fini_parti, p-00011396-porte_sat_fini_parti, p-00010809-porte_1, p-00009318-porte, flasque_1_1, cage, cag_1_1_1, arbre_d_entree_1) aus dem 3D Assembly Dataset

Vorgehen: efficientnet + functionSum + eucledian Distanz

E.2 Suchergebnisse GED-Distanz - Prinzipskizzen

Tabelle 23: Übersicht der besten erzielten Suchergebnisse über alle getesteten Reduktionsverfahren

Skizzen-nummer	GED-Distanz	Skizzen-nummer	GED-Distanz
0	33.0	10	81.0
1	21.0	11	23.0
2	8.0	12	48.0
3	4.0	13	43.0
4	13.0	14	55.0
5	7.0	15	36.0
6	3.0	16	11.0
7	4.0	17	3.0
8	12.0	18	33.0
9	61.0	-	-

Literaturverzeichnis

[1] Muscolino, Holly; Machado, Amy; Rydning, John und Vesset, Dan. „UNTAPPED VALUE: What Every Executive Needs to Know About Unstructured Data". *IDC white paper* (2024).

[2] Rathinasamy, Shankar. „UNSTRUCTURED DATA—GROWTH AND CHALLENGES" (2015).

[3] Smith, Tim; Stiller, Ben; Guszcza, Jim und Davenpor, Tom. „Analytics and AI-driven enterprises thrive in the Age of With" (2019).

[4] Haan, Evert de; Padigar, Manjunath; El Kihal, Siham; Kübler, Raoul und Wieringa, Jaap E. „Unstructured data research in business: Toward a structured approach". *Journal of Business Research* 177 (2024), S. 114655.

[5] Kassner, Laura; Gröger, Christoph; Mitschang, Bernhard und Westkämper, Engelbert. „Product Life Cycle Analytics – Next Generation Data Analytics on Structured and Unstructured Data". *Procedia CIRP* 33 (2015), S. 35–40. https://www.sciencedirect.com/science/article/pii/S2212827115006514.

[6] Boucher, Michelle. „Reducing Non-Value Added Work in Engineering - Improving Efficiency with Real-Time Access to Design Information" (2014).

[7] Brown, Jim. „The Facts About Managing Product Data - The Real Story on PDM Value and Accessibility" (2015).

[8] Autodesk, Inc. „DER ULTIMATIVE LEITFADEN FÜR DIE WIEDERVERWENDUNG BEREITS VORHANDENER KONSTRUKTIONEN" (2016).

[9] Purdy, Kevin. *Formula 1 chief appalled to find team using Excel to manage 20,000 car parts*. 2024. https://arstechnica.com/cars/2024/03/formula-1-chief-appalled-to-find-team-using-excel-to-manage-20000-car-parts/ (besucht am 22. 04. 2024).

[10] Vlah, Daria; Kastrin, Andrej; Povh, Janez und Vukašinović, Nikola. „Data-driven engineering design: A systematic review using scientometric approach". *Advanced Engineering Informatics* 54 (2022), S. 101774.

[11] Begoli, Edmon und Horey, James. „Design principles for effective knowledge discovery from big data". *2012 Joint Working IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture*. IEEE. 2012, S. 215–218.

[12] Whitney, Daniel E. *Mechanical assemblies: their design, manufacture, and role in product development*. Bd. 1. Oxford university press New York, 2004.

[13] Pahl, Gerhard; Beitz, Wolfgang; Feldhusen, Jörg und Grote, Karl-Heinrich. *Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung*. 7. Aufl. Springer-Lehrbuch. Berlin, Heidelberg: Springer, 2007.

[14] Isaksson, Ola; Kossmann, Mario; Bertoni, Marco; Eres, Hakki; Monceaux, Anne; Bertoni, Alessandro; Wiseall, Steve und Zhang, Xinwei. „Value-driven design–a methodology to link expectations to technical requirements in the extended enterprise". *INCOSE International Symposium*. Bd. 23. 1. Wiley Online Library. 2013, S. 803–819.

[15] Tangelder, Johan WH und Veltkamp, Remco C. „A survey of content based 3D shape retrieval methods". *Proceedings Shape Modeling Applications, 2004.* (2004), S. 145–156.

[16] Feng, Yixiong; Zhao, Yuliang; Zheng, Hao; Li, Zhiwu und Tan, Jianrong. „Data-driven product design toward intelligent manufacturing: A review". *International Journal of Advanced Robotic Systems* 17.2 (2020), S. 1729881420911257.

[17] AberdeenGroup. *The Design Reuse Benchmark Report: Seizing the Opportunity to Shorten Product Development*. Hrsg. von Aberdeen Group. 2007.

[18] Pakkanen, Jarkko; Huhtala, Petri; Juuti, Tero und Lehtonen, Timo. „Achieving Benefits with Design Reuse in Manufacturing Industry". *Procedia CIRP* 50 (2016), S. 8–13.

[19] Ong, Soh Khim; Xu, Q. L. und Nee, Andrew Yeh Ching. *Design reuse in product development modeling, analysis and optimization*. Bd. 4. Series on manufacturing systems and technology. Singapore: WORLD SCIENTIFIC, 2008.

[20] Lachmayer, R und Gottwald, P. „An Approach to Integrate Data Mining into the Development Process". *Modelling and Management of Engineering Processes: Proceedings of the 3rd International Conference 2013*. Springer. 2014, S. 175–185.

[21] Johnson, Julie; Hurst, Ada und Safayeni, Frank. „Managing data-driven design: A survey of the literature and future directions". *Proceedings of the Design Society* 3 (2023), S. 2525–2534.

[22] Geissbauer, Reinhard; Schrauf, Stefan; Morr, Jochen-Thomas; Wunderlin, JT; Krause, JH und Odenkirchen, A. *Digital Product Development 2025: Agile, Collaborative, AI Driven and Customer Centric*. Englisch. 2019.

[23] Haffey, Mark und Duffy, Alex. „Knowledge discovery and data mining within a design environment". *From Knowledge Intensive CAD to Knowledge Intensive Engineering: IFIP TC5 WG5. 2. Fourth Workshop on Knowledge Intensive CAD May 22–24, 2000, Parma, Italy*. Springer. 2002, S. 59–74.

[24] Chandrasegaran, Senthil K; Ramani, Karthik; Sriram, Ram D; Horváth, Imré; Bernard, Alain; Harik, Ramy F und Gao, Wei. „The evolution, challenges, and future of knowledge representation in product design systems". *Computer-aided design* 45.2 (2013), S. 204–228.

[25] Chowdhery, Syed Azad; Bertoni, Marco; Wall, Johan und Larsson, Tobias. „A data-driven design framework for early stage PSS design exploration" (2020).

[26] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2221 Blatt 2 Entwicklung technischer Produkte und Systeme - Gestaltung individueller Produktentwicklungsprozesse*. Düsseldorf, 2019.

[27] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2223 - Methodisches Entwerfen technischer Produkte*. 2004.

[28] Seiffert, Ulrich und Rainer, Gotthard. *Virtuelle Produktentstehung für Fahrzeug und Antrieb im Kfz: Prozesse, Komponenten, Beispiele aus der Praxis*. Springer, 2008.

[29] Feldhusen, Jörg und Grote, Karl-Heinrich. *Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung*. 8., vollständig überarbeitete Auflage. Berlin und Heidelberg: Springer Vieweg, 2013.

[30] Ehrlenspiel, Klaus. *Kostengünstig Entwickeln und Konstruieren: Kostenmanagement bei der integrierten Produktentwicklung*. 7th ed. VDI-Buch Ser. Dordrecht: Springer, 2013.

[31] Kramer, Friedhelm und Kramer, Markus. *Bausteine der Unternehmensführung: Kundenzufriedenheit und Unternehmenserfolg*. Springer-Verlag, 2013.

[32] Kleinaltenkamp, Michael; Fließ, Sabine und Jacob, Frank. *Customer Integration: Von der Kundenorientierung zur Kundenintegration*. Springer-Verlag, 2013.

[33] (VDI), Verein Deutscher Ingenieure. *Entwicklungsmethodik für mechatronische Systeme (VDI 2206): Design methodology for mechatronic systems*. VDI, 2004. https://books.google.de/books?id=V5BjtAEACAAJ.

[34] Suh, Nam P. „Axiomatic design theory for systems". *Research in engineering design* 10 (1998), S. 189–209.

[35] Roth, Karlheinz. *Konstruieren mit Konstruktionskatalogen: Band 3: Verbindungen und Verschlüsse, Lösungsfindung*. 2. Auflage, wesentlich erweitert und neu gestaltet. Springer eBook Collection Computer Science and Engineering. Berlin, Heidelberg und s.l.: Springer Berlin Heidelberg, 1996.

[36] Roth, Karlheinz. *Konstruieren mit Konstruktionskatalogen: Band 1: Konstruktionslehre*. 3. Auflage, erweitert und neu gestaltet. Berlin, Heidelberg und s.l.: Springer Berlin Heidelberg, 2000.

[37] Roth, Karlheinz. *Konstruieren mit Konstruktionskatalogen: Band 2: Kataloge*. Springer Berlin Heidelberg, 2013. https://books.google.de/books?id=c8CoBgAAQBAJ.

[38] Rodenacker, Wolf Georg. *Methodisches Konstruieren: Grundlagen, Methodik, praktische Beispiele*. Dritte, überarbeitete Auflage. Bd. 27. Konstruktionsbücher. Berlin und Heidelberg: Springer, 1984.

[39] Koller, Rudolf. *Konstruktionslehre für den Maschinenbau: Grundlagen des methodischen Konstruierens*. Zweite, völlig neubearbeitete und erweiterte Auflage. Berlin und Heidelberg: Springer, 1985.

[40] Blessing, Luciënne. „Comparison of design models proposed in prescriptive literature". *The role of design in the spaping of technology, COST A3/COST A4 International research workshop, 1995*. 1995.

[41] Evbuomwan, Nosayaba; Sivaloganathan, Sangarappillai und Jebb, A. „A survey of design philosophies, models, methods and systems". *Proceedings of the institution of mechanical engineers, part b: journal of engineering manufacture* 210.4 (1996), S. 301–320.

[42] Gericke, Kilian und Blessing, Luciënne. „Comparisons of design methodologies and process models across domains: a literature review". *DS 68-1: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 1: Design Processes, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011*. 2011.

[43] Hubka, Vladimir. *Principles of engineering design*. Elsevier, 2015.

[44] Hall, Arthur David. *A methodology for systems engineering*. Bd. 24. van Nostrand, 1962.

[45] Pugh, Stuart. *Total design: integrated methods for successful product engineering*. Addison-Wesley, 1991.

[46] Cross, Nigel und Roozenburg, Norbert. „Modelling the design process in engineering and in architecture". *Journal of Engineering design* 3.4 (1992), S. 325–337.

[47] Lindemann, Udo. *Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden*. Springer, 2006.

[48] Lindemann, Udo. *Handbuch Produktentwicklung*. Carl Hanser Verlag GmbH Co KG, 2016.

[49] Weber, Christian; Werner, Horst; Deubel, Till u. a. „A different view on PDM and its future potentials". *DS 30: Proceedings of DESIGN 2002, the 7th International Design Conference, Dubrovnik*. 2002, S. 101–112.

[50] Hatchuel, Armand und Weil, Benoit. „A new approach of innovative Design: an introduction to CK theory." *DS 31: Proceedings of ICED 03, the 14th International Conference on Engineering Design, Stockholm*. 2003.

[51] Albers, Albert; Matthiesen, Sven und Lechner, Gisbert. „Konstruktionsmethodisches Grundmodell zum Zusammenhang von Gestalt und Funktion technischer Systeme". *Konstruktion* 7-8 (2002), S. 55–60.

[52] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2222 - Konstruktionsmethodik - Methodisches Entwickeln von Lösungsprinzipien*. 1997.

[53] Rachuri, Sudarsan; Subrahmanian, Eswaran; Bouras, Abdelaziz; Fenves, Steven J; Foufou, Sebti und Sriram, Ram D. „Information sharing and exchange in the context of product lifecycle management: Role of standards". *Computer-Aided Design* 40.7 (2008), S. 789–800.

[54] Shi, Feng; Chen, Liuqing; Han, Ji und Childs, Peter. „A data-driven text mining and semantic network analysis for design information retrieval". *Journal of Mechanical Design* 139.11 (2017).

[55] Weilkiens, Tim. *Systems Engineering mit SysML/UML: Anforderungen, Analyse, Architektur. Mit einem Geleitwort von Richard Mark Soley*. dpunkt. verlag, 2014.

[56] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2803 - Funktionenanalyse*. 2019.

[57] Defense, Department of. *System engineering fundamentals*. 2001.

[58] Bytheway, Charles W. *FAST creativity and innovation: Rapidly improving processes, product development and solving complex problems*. J. Ross Publishing, 2007.

[59] Bytheway, Charles W. „The creative aspects of FAST diagramming". *Proceedings of the SAVE Conference*. 1971, S. 301–312.

[60] Zwicky, Fritz. *Entdecken, Erfinden, Forschen im morphologischen Weltbild: Mit Diagrammen*. 2. Aufl. Bd. Bd. 5. Schriftenreihe der Fritz-Zwicky-Stiftung. Glarus: Baeschlin, 1989.

[61] Koller, Rudolf und Kastrup, Norbert. *Prinziplösungen zur Konstruktion technischer Produkte*. Springer-Verlag, 2013.

[62] Naefe, Paul und Naefe, Paul. „Arbeitsschritte des Konstruktionsprozesses und Methodenauswahl". *Methodisches Konstruieren: Auf den Punkt gebracht* (2018), S. 39–51.

[63] Buzan, Tony und Buzan, Barry. *Das Mind-Map Buch: Die beste Methode zur Steigerung Ihres geistigen Potenzials*. 4., aktualisierte und erweiterte Auflage. BBC. München: mvg Verlag, 2022.

[64] Schawel, Christian; Billing, Fabian; Schawel, Christian und Billing, Fabian. „6-3-5-Methode: (Kreativitätstechniken)". *Top 100 Management Tools: Das wichtigste Buch eines Managers Von ABC-Analyse bis Zielvereinbarung* (2014), S. 301–303.

[65] Cuhls, Kerstin. „Die Delphi-Methode – eine Einführung". Apr. 2019, S. 3–31.

[66] Rachuri, Sudarsan; Subrahmanian, Eswaran; Bouras, Abdelaziz; Fenves, Steven J.; Foufou, Sebti und Sriram, Ram D. „Information sharing and exchange in the context of product lifecycle management: Role of standards". *Computer-Aided Design* 40.7 (2008), S. 789–800.

[67] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2225 - Technisch-wirtschaftliches Konstruieren*. 1997.

[68] Dickie, H Ford. „ABC inventory analysis shoots for dollars not pennies". *Factory Management and Maintenance* 109.7 (1951), S. 92–94.

[69] Gürel, Emet. „SWOT ANALYSIS: A THEORETICAL REVIEW". *Journal of International Social Research* 10 (Aug. 2017), S. 994–1006.

[70] Zangemeister, Christof. *Nutzwertanalyse in der Systemtechnik: eine Methodik zur multidimensionalen Bewertung und Auswahl von Projektalternativen*. BoD–Books on Demand, 2014.

[71] Finkler, Michael. „Mammutprojekt Manufacturing-X". *CITplus* 27.4 (2024), S. 16–17.

[72] Kohn, Andreas. „Entwicklung einer Wissensbasis für die Arbeit mit Produktmodellen". Diss. Technische Universität München, 2014.

[73] Owen, Roderick und Horváth, Imre. „Towards product-related knowledge asset warehousing in enterprises". *Proceedings of the 4th international symposium on tools and methods of competitive engineering, TMCE*. Bd. 2002. Citeseer. 2002, S. 155–70.

[74] DIN, 199-1. *Technische Produktdokumentation-CAD-Modelle, Zeichnungen und Stücklisten-Teil 1: Begriffe*. 2002.

[75] Trent, H. M. „On the Construction of Schematic Diagrams for Mechanical Systems". *The Journal of the Acoustical Society of America* 30.8 (1958), S. 795–800.

[76] Bertsche, Bernd; Göhner, Peter; Jensen, Uwe; Schinköthe, Wolfgang und Wunderlich, Hans-Joachim. *Zuverlässigkeit mechatronischer Systeme: Grundlagen und Bewertung in frühen Entwicklungsphasen*. VDI-Buch. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. http://nbn-resolving.org/urn:nbn:de:bsz:31-epflicht-1519756.

[77] Labisch, Susanna und Weber, Christian. *Technisches Zeichnen: Intensiv und effektiv lernen und üben ; mit 55 Tabellen ; [mit aktueller Normung und 321 Übungen*. 2., überarb. Aufl. Studium. Wiesbaden: Vieweg, 2005.

[78] DIN ISO, 3952. *Vereinfachte Darstellungen in der Kinematik*. Standard. International Organization for Standardization, 1995.

[79] List, Ronald. *CATIA V5 - Grundkurs für Maschinenbauer: Bauteil- und Baugruppenkonstruktion, Zeichnungsableitung*. 4., aktualisierte und erw. Aufl. Studium. Wiesbaden: Vieweg + Teubner, 2009.

[80] Madsen, David A.; Madsen, David P.; Standiford, Kevin und Krulikowski, Alex. *Engineering drawing & design*. Sixth edition. Australia: Cengage Learning, 2017.

[81] Vajna, Sándor; Weber, Christian; Zeman, Klaus; Hehenberger, Peter; Gerhard, Detlef und Wartzack, Sandro. *CAx für Ingenieure: Eine praxisbezogene Einführung*. Springer-Verlag, 2018.

[82] Encaração, Jose; Strasser, Wolfgang und Klein, Reinhard. *Modellierung komplexer Objekte und photorealistische Bilderzeugung. 4., aktualisierte und wesentlich erw. Aufl. ed. Graphische Datenverarbeitung 2*. 1997.

[83] Guo, Yulan; Wang, Hanyun; Hu, Qingyong; Liu, Hao; Liu, Li und Bennamoun, Mohammed. „Deep Learning for 3D Point Clouds: A Survey". *IEEE transactions on pattern analysis and machine intelligence* 43.12 (2021), S. 4338–4364.

[84] Liu, Weiping; Sun, Jia; Li, Wanyi; Hu, Ting und Wang, Peng. „Deep Learning on Point Clouds and Its Application: A Survey". *Sensors* 19.19 (2019), S. 4188. https://www.mdpi.com/542410.

[85] Qi, Charles R; Su, Hao; Mo, Kaichun und Guibas, Leonidas J. „Pointnet: Deep learning on point sets for 3d classification and segmentation". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, S. 652–660.

[86] Chang, Kuang-Hua. *e-Design: Computer-aided engineering design*. London, UK: Academic Press, 2015. http://www.sciencedirect.com/science/book/9780123820389.

[87] Bézier, Pierre. *Numerical control; mathematics and applications*. John Wiley & Sons, 1972.

[88] De Boor, Carl. *B (asic)-Spline Basics*. Techn. Ber. WISCONSIN UNIV-MADISON MATHEMATICS RESEARCH CENTER, 1986.

[89] Eigner, Martin. *Modellbasierte Virtuelle Produktentwicklung*. Berlin, Heidelberg: Springer Berlin / Heidelberg, 2014. https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=1966086.

[90] Baumgart, Bruce G. „Geometric modeling for computer vision." 1974.

[91] Anderl, Reiner und Trippner, Dietmar. *STEP: Standard for the exchange of product model data; eine Einführung in die Entwicklung, Implementierung und industrielle Nutzung der Normenreihe ISO 10303 (STEP)*. Springer, 2000.

[92] Nagel, Roger N.; Braithwaite, Walt W. und Kennicott, Philip R. *Initial graphics exchange specification iges version 1.0*. 1980.

[93] ISO, 14306. *Industrielle Automatisierungssysteme und Integration - JT-Dateiformat für 3D-Visualisierung*. Standard. International Organization for Standardization, 2012.

[94] Kitsios, Vasileios. *Ihre Produktentwicklung Digitalisieren: In Einfachen Schritten Den Grundstein Für Die Digitale Transformation Legen*. Springer eBook Collection. Wiesbaden: Springer Fachmedien Wiesbaden GmbH, 2021.

[95] Hallmann, Martin; Goetz, Stefan und Schleich, Benjamin. „Mapping of GD&T information and PMI between 3D product models in the STEP and STL format". *Computer-Aided Design* 115 (2019), S. 293–306.

[96] Bénière, Roseline; Subsol, Gérard; Gesquière, Gilles; Le Breton, François und Puech, William. „A comprehensive process of reverse engineering from 3D meshes to CAD models". *Computer-Aided Design* 45.11 (2013), S. 1382–1393.

[97] Park, Jung Min; Lee, Byung Chai; Chae, Soo Won und Kwon, Ki Youn. „Surface reconstruction from FE mesh model". *Journal of Computational Design and Engineering* 6.2 (2019), S. 197–208.

[98] Prozessgestaltung, VDI-Gesellschaft Produkt- und; Product, VDI Society; Design, Process; e.V., VDI Verein Deutscher Ingenieure und German Engineers, VDI - The Association of. *VDI 2219 - Informationsverarbeitung in der Produktentwicklung - Einführung und Betrieb von PDM-Systemen*. 2016.

[99] Wittel, Herbert; Spura, Christian und Jannasch, Dieter. *Roloff/Matek Maschinenelemente*. 25. Auflage. Wiesbaden: Springer Vieweg, 2021.

[100] Isaksson, Ola und Eckert, Claudia. *Product Development 2040*. 2020.

[101] Liu, Changqing und Chen, Xiaoqian. „Data-driven design paradigm in engineering problems". *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 231.8 (2017), S. 1522–1534.

[102] Yang, Maolin; Jiang, Pingyu; Zang, Tianshuo und Liu, Yuhao. „Data-driven Intelligent Computational Design for Products: Method, Techniques, and Applications". *Journal of Computational Design and Engineering* (Juli 2023), qwad070. eprint: https://academic.oup.com/jcde/advance-article-pdf/doi/10.1093/jcde/qwad070/50833859/qwad070.pdf. https://doi.org/10.1093/jcde/qwad070.

[103] Tuarob, Suppawong; Lim, Sunghoon und Tucker, Conrad S. „Automated discovery of product feature inferences within large-scale implicit social media data". *Journal of Computing and Information Science in Engineering* 18.2 (2018).

[104] Stürmlinger, Tobias; Gladysz, Bartosz; Strauch, Markus und Albers, Albert. „Design with Industry 4.0 – Priorization of Sensor data for a smart data driven Product Development Process". Englisch. *Tools and methods of competitive engineering : proceedings of theTwelfth International Symposium on Tools and Methods of Competitive Engineering - TMCE 2018, May 07-11, Las Palmas de Gran Canaria, Spain. Proceedings of TMCE 2018, Las Palmas de Gran Canaria, Gran Canaria, Spain,7-11 May 2018. Ed.: Imre Horváth*. Delft University of Technology (TU Delft), 2018, S. 439–450.

[105] Li, Zhi; Tian, ZG; Wang, JW und Wang, Wai Ming. „Extraction of affective responses from customer reviews: an opinion mining and machine learning approach". *International Journal of Computer Integrated Manufacturing* 33.7 (2020), S. 670–685.

[106] Chaklader, Rushtin und Parkinson, Matthew B. „Data-Driven Sizing Specification Utilizing Consumer Text Reviews". *Journal of Mechanical Design* 139.11 (Okt. 2017). https://doi.org/10.1115/1.4037476.

[107] Jiang, Huimin; Kwong, Chun Kit und Yung, Kai Leung. „Predicting future importance of product features based on online customer reviews". *Journal of Mechanical Design* 139.11 (2017).

[108] Cantamessa, Marco; Montagna, Francesca; Altavilla, Stefania und Casagrande-Seretti, Alessandro. „Data-driven design: the new challenges of digitalization on product design and development". *Design Science* 6 (2020).

[109] Quan, Huafeng; Li, Shaobo; Zeng, Changchang; Wei, Hongjing und Hu, Jianjun. *Big Data driven Product Design: A Survey*. arXiv, 2021.

[110] Plappert, Stefan; Gembarski, Paul Christoph und Lachmayer, Roland. „The Use of Knowledge-Based Engineering Systems and Artificial Intelligence in Product Development: A Snapshot". *Information Systems Architecture and Technology: Proceedings of 40th Anniversary International Conference on Information Systems Architecture and Technology – ISAT 2019*. Hrsg. von Świątek, Jerzy; Borzemski, Leszek und Wilimowska, Zofia. Bd. 1051. Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2020, S. 62–73.

[111] Bertoni, Alessandro. „Data-driven design in concept development: systematic review and missed opportunities". 1 (2020), S. 101–110.

[112] Bruce, Peter C.; Bruce, Andrew und Gedeck, Peter. *Praktische Statistik für Data Scientists: 50+ essenzielle Konzepte mit R und Python*. 1. Auflage. Heidelberg: O'Reilly, 2021.

[113] Papp, Stefan; Weidinger, Wolfgang; Munro, Katherine; Ortner, Bernhard; Cadonna, Annalisa; Langs, Georg; Licandro, Roxane; Meir-Huber, Mario; Nikolić, Danko; Toth, Zoltan; Vesela, Barora; Wazir, Rania und Zauner, Günther. *Handbuch Data Science und KI: Mit Machine Learning und Datenanalyse Wert aus Daten generieren*. 2. Auflage. Hanser eLibrary. München: Hanser, 2022. https://www.hanser-elibrary.com/doi/book/10.3139/9783446472457.

[114] Tukey, John W. „The Future of Data Analysis". *The Annals of Mathematical Statistics* 33.1 (1962), S. 1–67.

[115] Kelleher, John D und Tierney, Brendan. *Data science*. MIT Press, 2018.

[116] Grus, Joel. *Einführung in Data Science: Grundprinzipien der Datenanalyse mit Python*. 2. Auflage. Heidelberg: O'Reilly, 2019. http://nbn-resolving.org/urn:nbn:de:bsz:31-epflicht-1388546.

[117] Braschler, Martin, Hrsg. *Applied data science: Lessons learned for the data-driven business*. Cham, Switzerland: Springer, 2019.

[118] Davenport, Thomas H; Patil, DJ und Scientist, D. „The sexiest job of the 21st century". *Harvard Business Review* 8 (2012).

[119] Samuel, Arthur L. „Some studies in machine learning using the game of checkers". *IBM Journal of research and development* 3.3 (1959), S. 210–229.

[120] Carbonell, Jaime G; Michalski, Ryszard S und Mitchell, Tom M. „An overview of machine learning". *Machine learning* (1983), S. 3–23.

[121] Mitchell, Tom M. *Machine learning*. International ed., [Reprint.] McGraw-Hill series in computer science. New York, NY: McGraw-Hill, 1997. https://ds.amu.edu.et/xmlui/bitstream/handle/123456789/14637/machine_learning%20-%20421%20pages.pdf?sequence=1&isallowed=y.

[122] Jiang, Hui. „Notation". *Machine Learning Fundamentals: A Concise Introduction*. Cambridge University Press, 2021, S. xvii–xviii.

[123] Michalski, Ryszard S und Kodratoff, Yves. „Research in machine learning: Recent progress, classification of methods, and future directions". *Machine learning* (1990), S. 3–30.

[124] Witten, Ian H.; Frank, Eibe und Hall, Mark A. *Data mining: Practical machine learning tools and techniques*. Third edition. The Morgan Kaufmann series in data management systems. Amsterdam: Morgan Kaufmann, 2011.

[125] Zhou, Zhi-Hua. *Machine learning*. Singapore: Springer, 2021.

[126] Jung, Alexander. *Machine Learning: The Basics*. Springer eBook Collection. Singapore: Springer, 2022.

[127] Alpaydın, Ethem. *Maschinelles Lernen*. 2., erweiterte Auflage. De Gruyter Studium. Berlin: De Gruyter, 2019. https://www.degruyter.com/isbn/9783110617894.

[128] Kane, Frank. *Hands-On Data Science and Python Machine Learning*. 1. Aufl. Birmingham: Packt Publishing Limited, 2017. https://www.wiso-net.de/document/PKEB__9781787280229420.

[129] Runkler, Thomas A. *Data Mining*. Wiesbaden: Vieweg+Teubner, 2010.

[130] Cleve, Jürgen und Lämmel, Uwe. *Data mining*. Walter de Gruyter GmbH & Co KG, 2020.

[131] Petersohn, Helge. *Data Mining: Verfahren, Prozesse, Anwendungsarchitektur*. Oldenbourg Wissenschaftsverlag GmbH, 2005.

[132] Bruce, Peter C; Bruce, Andrew und Gedeck, Peter. *Praktische Statistik für Data Scientists: 50+ essenzielle Konzepte mit R und Python*. O'Reilly, 2021.

[133] Alpaydin, Ethem. *Maschinelles Lernen*. Walter de Gruyter GmbH & Co KG, 2022.

[134] Hutter, Frank; Kotthoff, Lars und Vanschoren, Joaquin. *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.

[135] Shao, Kun; Tang, Zhentao; Zhu, Yuanheng; Li, Nannan und Zhao, Dongbin. „A survey of deep reinforcement learning in video games". *arXiv preprint arXiv:1912.10944* (2019).

[136] Goodfellow, Ian; Bengio, Yoshua und Courville, Aaron. *Deep learning*. MIT press, 2016.

[137] Deng, Li; Yu, Dong u. a. „Deep learning: methods and applications". *Foundations and trends® in signal processing* 7.3–4 (2014), S. 197–387.

[138] Janiesch, Christian; Zschech, Patrick und Heinrich, Kai. „Machine learning and deep learning". *Electronic Markets* 31.3 (2021), S. 685–695.

[139] LeCun, Yann; Bengio, Yoshua und Hinton, Geoffrey. „Deep learning". *nature* 521.7553 (2015), S. 436–444.

[140] LeCun, Yann; Boser, Bernhard; Denker, John S; Henderson, Donnie; Howard, Richard E; Hubbard, Wayne und Jackel, Lawrence D. „Backpropagation applied to handwritten zip code recognition". *Neural computation* 1.4 (1989), S. 541–551.

[141] LeCun, Yann; Bottou, Leon; Bengio, Yoshua und Haffner, Patrick. „Gradient-based learning applied to document recognition". *Proceedings of the IEEE* 86.11 (1998), S. 2278–2324.

[142] Krizhevsky, Alex; Hinton, Geoffrey u. a. „Learning multiple layers of features from tiny images" (2009).

[143] Everingham, Mark; Van Gool, Luc; Williams, Christopher KI; Winn, John und Zisserman, Andrew. „The pascal visual object classes (voc) challenge". *International journal of computer vision* 88.2 (2010), S. 303–338.

[144] Deng, Jia; Dong, Wei; Socher, Richard; Li, Li-Jia; Li, Kai und Fei-Fei, Li. „Imagenet: A large-scale hierarchical image database". *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, S. 248–255.

[145] Krizhevsky, Alex; Sutskever, Ilya und Hinton, Geoffrey E. „ImageNet Classification with Deep Convolutional Neural Networks". *Advances in Neural Information Processing Systems*. Hrsg. von Pereira, F.; Burges, C.J.; Bottou, L. und Weinberger, K.Q. Bd. 25. Curran Associates, Inc., 2012. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[146] Simonyan, Karen und Zisserman, Andrew. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv, 2015.

[147] Srivastava, Rupesh Kumar; Greff, Klaus und Schmidhuber, Jürgen. „Highway networks". *arXiv preprint arXiv:1505.00387* (2015).

[148] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing und Sun, Jian. „Deep Residual Learning for Image Recognition". *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[149] Jain, Ramesh C.; Kasturi, Rangachar und Schunck, Brian G. *Introduction to machine vision*. McGraw-Hill series in computer science Artificial intelligence. London: McGraw-Hill, 1995.

[150] Fisher, R. B. *Dictionary of Computer Vision and Image Processing*. 2nd edition. Chichester, West Sussex: John Wiley & Sons Inc, 2014. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119286462.

[151] Szeliski, Richard. *Computer Vision*. Cham: Springer International Publishing, 2022.

[152] Wiley, Victor und Lucas, Thomas. „Computer vision and image processing: a paper review". *International Journal of Artificial Intelligence Research* 2.1 (2018), S. 29–36.

[153] Everingham, Mark; Eslami, SM; Van Gool, Luc; Williams, Christopher KI; Winn, John und Zisserman, Andrew. „The pascal visual object classes challenge: A retrospective". *International journal of computer vision* 111.1 (2015), S. 98–136.

[154] Russakovsky, Olga; Deng, Jia; Su, Hao; Krause, Jonathan; Satheesh, Sanjeev; Ma, Sean; Huang, Zhiheng; Karpathy, Andrej; Khosla, Aditya; Bernstein, Michael u. a. „Imagenet large scale visual recognition challenge". *International journal of computer vision* 115.3 (2015), S. 211–252.

[155] Sinha, Rajat Kumar; Pandey, Ruchi und Pattnaik, Rohan. „Deep learning for computer vision tasks: a review". *arXiv preprint arXiv:1804.03928* (2018).

[156] Zhang, Xin; Yang, Yee-Hong; Han, Zhiguang; Wang, Hui und Gao, Chao. „Object class detection: A survey". *ACM Computing Surveys (CSUR)* 46.1 (2013), S. 1–53.

[157] Liu, Li; Ouyang, Wanli; Wang, Xiaogang; Fieguth, Paul; Chen, Jie; Liu, Xinwang und Pietikäinen, Matti. „Deep Learning for Generic Object Detection: A Survey". *International Journal of Computer Vision* 128.2 (2020), S. 261–318. https://link.springer.com/article/10.1007/s11263-019-01247-4.

[158] Sahu, Chandan K; Young, Crystal und Rai, Rahul. „Artificial intelligence (AI) in augmented reality (AR)-assisted manufacturing applications: a review". *International Journal of Production Research* 59.16 (2021), S. 4903–4959.

[159] Vahab, Abdul; Naik, Maruti S; Raikar, Prasanna G und Prasad, SR. „Applications of object detection system". *International Research Journal of Engineering and Technology (IRJET)* 6.4 (2019), S. 4186–4192.

[160] Zou, Zhengxia; Shi, Zhenwei; Guo, Yuhong und Ye, Jieping. „Object detection in 20 years: A survey". *arXiv preprint arXiv:1905.05055* (2019).

[161] Fischler, Martin A und Elschlager, Robert A. „The representation and matching of pictorial structures". *IEEE Transactions on computers* 100.1 (1973), S. 67–92.

[162] Osuna, Edgar; Freund, Robert und Girosit, Federico. „Training support vector machines: an application to face detection". *Proceedings of IEEE computer society conference on computer vision and pattern recognition*. IEEE. 1997, S. 130–136.

[163] Csurka, Gabriella; Dance, Christopher; Fan, Lixin; Willamowski, Jutta und Bray, Cédric. „Visual categorization with bags of keypoints". *Workshop on statistical learning in computer vision, ECCV*. Bd. 1. 1-22. Prague. 2004, S. 1–2.

[164] Lowe, David. „Sift-the scale invariant feature transform". *Int. J* 2.91-110 (2004), S. 2.

[165] Brown, Matthew; Szeliski, Richard und Winder, Simon. „Multi-image matching using multi-scale oriented patches". *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Bd. 1. IEEE. 2005, S. 510–517.

[166] Mikolajczyk, Krystian und Schmid, Cordelia. „A performance evaluation of local descriptors". *IEEE transactions on pattern analysis and machine intelligence* 27.10 (2005), S. 1615–1630.

[167] Dalal, Navneet und Triggs, Bill. „Histograms of oriented gradients for human detection". *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*. Bd. 1. Ieee. 2005, S. 886–893.

[168] Felzenszwalb, Pedro F; Girshick, Ross B; McAllester, David und Ramanan, Deva. „Object detection with discriminatively trained part-based models". *IEEE transactions on pattern analysis and machine intelligence* 32.9 (2009), S. 1627–1645.

[169] Felzenszwalb, Pedro F; Girshick, Ross B und McAllester, David. „Cascade object detection with deformable part models". *2010 IEEE Computer society conference on computer vision and pattern recognition*. Ieee. 2010, S. 2241–2248.

[170] Lin, Tsung-Yi; Maire, Michael; Belongie, Serge; Hays, James; Perona, Pietro; Ramanan, Deva; Dollar, Piotr und Zitnick, C Lawrence. „Microsoft coco: Common objects in context". *European conference on computer vision*. Springer. 2014, S. 740–755.

[171] Padilla, Rafael; Netto, Sergio L und Da Silva, Eduardo AB. „A survey on performance metrics for object-detection algorithms". *2020 international conference on systems, signals and image processing (IWSSIP)*. IEEE. 2020, S. 237–242.

[172] Jiao, Licheng; Zhang, Fan; Liu, Fang; Yang, Shuyuan; Li, Lingling; Feng, Zhixi und Qu, Rong. „A survey of deep learning-based object detection". *IEEE access* 7 (2019), S. 128837–128868.

[173] Zaidi, Syed Sahil Abbas; Ansari, Mohammad Samar; Aslam, Asra; Kanwal, Nadia; Asghar, Mamoona und Lee, Brian. „A survey of modern deep learning based object detection models". *Digital Signal Processing* (2022), S. 103514.

[174] Redmon, Joseph; Divvala, Santosh; Girshick, Ross und Farhadi, Ali. „You only look once: Unified, real-time object detection". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, S. 779–788.

[175] Jocher, Glenn; Chaurasia, Ayush und Qiu, Jing. *YOLO by Ultralytic*. 2023. https://github.com/ultralytics/ultralytics (besucht am 14. 09. 2023).

[176] Liu, Wei; Anguelov, Dragomir; Erhan, Dumitru; Szegedy, Christian; Reed, Scott; Fu, Cheng-Yang und Berg, Alexander C. „Ssd: Single shot multibox detector". *European conference on computer vision*. Springer. 2016, S. 21–37.

[177] Lin, Tsung-Yi; Goyal, Priya; Girshick, Ross; He, Kaiming und Dollár, Piotr. „Focal loss for dense object detection". *Proceedings of the IEEE international conference on computer vision*. 2017, S. 2980–2988.

[178] Zhao, Qijie; Sheng, Tao; Wang, Yongtao; Tang, Zhi; Chen, Ying; Cai, Ling und Ling, Haibin. *M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network*. 2018. https://arxiv.org/abs/1811.04533.

[179] Girshick, Ross; Donahue, Jeff; Darrell, Trevor und Malik, Jitendra. „Rich feature hierarchies for accurate object detection and semantic segmentation". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, S. 580–587.

[180] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing und Sun, Jian. „Spatial pyramid pooling in deep convolutional networks for visual recognition". *IEEE transactions on pattern analysis and machine intelligence* 37.9 (2015), S. 1904–1916.

[181] Girshick, Ross. „Fast r-cnn". *Proceedings of the IEEE international conference on computer vision*. 2015, S. 1440–1448.

[182] Ren, Shaoqing; He, Kaiming; Girshick, Ross und Sun, Jian. „Faster r-cnn: Towards real-time object detection with region proposal networks". *Advances in neural information processing systems* 28 (2015).

[183] He, Kaiming; Gkioxari, Georgia; Dollár, Piotr und Girshick, Ross. „Mask r-cnn". *Proceedings of the IEEE international conference on computer vision*. 2017, S. 2961–2969.

[184] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing und Sun, Jian. „Deep residual learning for image recognition". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, S. 770–778.

[185] Lin, Tsung-Yi; Dollár, Piotr; Girshick, Ross; He, Kaiming; Hariharan, Bharath und Belongie, Serge. „Feature pyramid networks for object detection". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, S. 2117–2125.

[186] Zhou, Longfei; Zhang, Lin und Konz, Nicholas. „Computer Vision Techniques in Manufacturing". *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2022).

[187] Moreno-García, Carlos Francisco; Elyan, Eyad und Jayne, Chrisina. „New trends on digitisation of complex engineering drawings". *Neural computing and applications* 31.6 (2019), S. 1695–1712.

[188] Valveny, Ernest und Dosch, Philippe. „Symbol recognition contest: A synthesis". *International Workshop on Graphics Recognition*. Springer. 2003, S. 368–385.

[189] Goyal, Shreya; Mistry, Vishesh; Chattopadhyay, Chiranjoy und Bhatnagar, Gaurav. „BRIDGE: Building Plan Repository for Image Description Generation, and Evaluation". *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019.

[190] Delalandre, Mathieu; Valveny, Ernest; Pridmore, Tony und Karatzas, Dimosthenis. „Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems". *International Journal of Document Analysis and Recognition (IJDAR)* 13.3 (2010), S. 187–207. https://link.springer.com/article/10.1007/s10032-010-0120-x.

[191] Fu, Luoting und Kara, Levent Burak. „Neural network-based symbol recognition using a few labeled samples". *Computers & Graphics* 35.5 (2011), S. 955–966.

[192] Wen, Rui; Tang, Weiqing und Su, Zhiyong. „A 2D Engineering Drawing and 3D Model Matching Algorithm for Process Plant". *2015 International Conference on Virtual Reality and Visualization (ICVRV)*. IEEE, 102015, S. 154–159.

[193] Yu, Yuhong; Samal, Ashok und Seth, Sharad C. „A system for recognizing a large class of engineering drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.8 (1997), S. 868–890.

[194] Howie, Cameron; Kunz, John; Binford, Thomas; Chen, T. und Law, Kincho H. „Computer interpretation of process and instrumentation drawings". *Advances in Engineering Software* 29.7-9 (1998), S. 563–570.

[195] Dzhusupova, Rimma; Banotra, Richa; Bosch, Jan und Olsson, Helena Holmstrom. „Pattern Recognition Method for Detecting Engineering Errors on Technical Drawings". *2022 IEEE World AI IoT Congress (AIIoT)*. IEEE, 2022.

[196] Rantala, Miia; Niemistö, Hannu; Karhela, Tommi; Sierla, Seppo und Vyatkin, Valeriy. „Applying graph matching techniques to enhance reuse of plant design information". *Computers in Industry* 107 (2019), S. 81–98.

[197] Vaxiviere, Pascal und Tombre, Karl. „Celesstin: CAD conversion of mechanical drawings". *Computer* 25.7 (1992), S. 46–54.

[198] Alwan, Salwan. „Unsupervised and hybrid vectorization techniques for 3D reconstruction of engineering drawings". Diss. Ecole nationale supérieure Mines-Télécom Atlantique, 2021.

[199] Banerjee, Purnendu; Choudhary, Sumit; Das, Supriya; Majumdar, Himadri; Roy, Rahul und Chaudhuri, Bidyut B. „Automatic Hyperlinking of Engineering Drawing Documents". *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*. IEEE, 2016.

[200] Yamakawa, Takuya; Dobashi, Yoshinori; Okabe, Makoto; Iwasaki, Kei und Yamamoto, Tsuyoshi. „Computer simulation of furniture layout when moving from one house to another". *Proceedings of the 33rd Spring Conference on Computer Graphics*. 2017, S. 1–8.

[201] Müller, Patrik; Gembarski, Paul Christoph und Lachmayer, Roland. „Detektion von Konstruktionsfehlern durch eine automatisierte Objekterkennung mittels Deep Learning". *KT-Kolloquium; 18*. Essen: Universität Duisburg-Essen. 2020.

[202] Dering, Matthew L und Tucker, Conrad S. „A convolutional neural network model for predicting a product's function, given its form". *Journal of Mechanical Design* 139.11 (2017), S. 111408.

[203] Xiao, Yun-Peng; Lai, Yu-Kun; Zhang, Fang-Lue; Li, Chunpeng und Gao, Lin. „A survey on deep geometry learning: From a representation perspective". *Computational Visual Media* 6.2 (2020), S. 113–133. https://link.springer.com/article/10.1007/s41095-020-0174-8.

[204] Cao, Wenming; Zheng, Canta; Yan, Zhiyue und Xie, Weixin. „Geometric deep learning: progress, applications and challenges". *Science China Information Sciences* 65.2 (2022).

[205] Gezawa, Abubakar Sulaiman; Zhang, Yan; Wang, Qicong und Yunqi, Lei. „A Review on Deep Learning Approaches for 3D Data Representations in Retrieval and Classifications". *IEEE Access* 8 (2020), S. 57566–57593.

[206] Lévy, Bruno und Zhang, Hao. „Elements of geometry processing". *SIGGRAPH Asia 2011 Courses*. New York, NY, USA: ACM, 2011.

[207] Xu, Kai; Kim, Vladimir G; Huang, Qixing; Mitra, Niloy und Kalogerakis, Evangelos. „Data-driven shape analysis and processing" (2016), S. 1–38.

[208] Rostami, Reihaneh; Bashiri, Fereshteh S.; Rostami, Behrouz und Yu, Zeyun. „A Survey on Data–Driven 3D Shape Descriptors". *Computer Graphics Forum* 38.1 (2019), S. 356–393.

[209] Shilane, Philip; Min, Patrick; Kazhdan, Michael und Funkhouser, Thomas. „The princeton shape benchmark". *Proceedings Shape Modeling Applications, 2004.* IEEE. 2004, S. 167–178.

[210] Wu, Zhirong; Song, Shuran; Khosla, Aditya; Yu, Fisher; Zhang, Linguang; Tang, Xiaoou und Xiao, Jianxiong. „3d shapenets: A deep representation for volumetric shapes". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, S. 1912–1920.

[211] Mo, Kaichun; Zhu, Shilin; Chang, Angel X; Yi, Li; Tripathi, Subarna; Guibas, Leonidas J und Su, Hao. „Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding". *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, S. 909–918.

[212] Chang, Angel X; Funkhouser, Thomas; Guibas, Leonidas; Hanrahan, Pat; Huang, Qixing; Li, Zimo; Savarese, Silvio; Savva, Manolis; Song, Shuran; Su, Hao u. a. „Shapenet: An information-rich 3d model repository". *arXiv preprint arXiv:1512.03012* (2015).

[213] Chang, Angel X; Mago, Rishi; Krishna, Pranav; Savva, Manolis und Fellbaum, Christiane. „Linking WordNet to 3D shapes". *Proceedings of the 9th Global Wordnet Conference*. 2018, S. 358–363.

[214] Lian, Zhouhui; Godil, Afzal; Fabry, Thomas; Furuya, Takahiko; Hermans, Jeroen; Ohbuchi, Ryutarou; Shu, Chang; Smeets, Dirk; Suetens, Paul; Vandermeulen, Dirk u. a. „SHREC'10 Track: Non-rigid 3D Shape Retrieval." *3DOR@ Eurographics*. 2010, S. 101–108.

[215] Li, Bo; Lu, Yijuan; Li, Chunyuan; Godil, Afzal; Schreck, Tobias; Aono, Masaki; Burtscher, Martin; Fu, Hongbo; Furuya, Takahiko; Johan, Henry u. a. „SHREC'14 track: Extended large scale sketch-based 3D shape retrieval". *Eurographics workshop on 3D object retrieval*. Bd. 2014. 2014, S. 121–130.

[216] Cosmo, Luca; Rodola, Emanuele; Bronstein, Michael M; Torsello, Andrea; Cremers, Daniel und Sahillioglu, Y. „SHREC'16: Partial matching of deformable shapes". *Proc. 3DOR* 2.9 (2016), S. 12.

[217] Bespalov, Dmitriy; Ip, Cheuk Yiu; Regli, William C und Shaffer, Joshua. „Benchmarking CAD search techniques". *Proceedings of the 2005 ACM symposium on Solid and physical modeling*. 2005, S. 275–286.

[218] Kim, Sangpil; Chi, Hyung-Gun; Hu, Xiao; Huang, Qixing und Ramani, Karthik. „A large-scale annotated mechanical components benchmark for classification and retrieval tasks with deep neural networks". *Computer Vision – ECCV 2020*. Lecture notes in computer science. Cham: Springer International Publishing, 2020, S. 175–191.

[219] Koch, Sebastian; Matveev, Albert; Jiang, Zhongshi; Williams, Francis; Artemov, Alexey; Burnaev, Evgeny; Alexa, Marc; Zorin, Denis und Panozzo, Daniele. „Abc: A big cad model dataset for geometric deep learning". *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, S. 9601–9611.

[220] Jayanti, Subramaniam; Kalyanaraman, Yagnanarayanan; Iyer, Natraj und Ramani, Karthik. „Developing an engineering shape benchmark for CAD models". *Computer-Aided Design* 38.9 (2006), S. 939–953.

[221] Jagadeesan, P; Wenzel, Jan; Corney, Jonathan R.; Yan, Xiu-Tian; Sherlock, Andrew; Torres-Sanchez, Carmen und Regli, William. „Validation of purdue engineering shape benchmark clusters by crowdsourcing". *Proceedings of the International Conference on Product Lifecycle Management*. Bath, UK, 2009.

[222] Brock, Andrew; Lim, Theodore; Ritchie, J. M. und Weston, Nick. *Generative and Discriminative Voxel Modeling with Convolutional Neural Networks*. arXiv, 2016.

[223] Furuya, Takahiko und Ohbuchi, Ryutarou. „Deep aggregation of local 3d geometric features for 3d model retrieval." *BMVC*. Bd. 7. 2016, S. 8.

[224] Ren, Mengwei; Niu, Liang und Fang, Yi. *3D-A-Nets: 3D Deep Dense Descriptor for Volumetric Shapes with Adversarial Networks*. 2017. https://arxiv.org/abs/1711.10108.

[225] Qi, Charles Ruizhongtai; Yi, Li; Su, Hao und Guibas, Leonidas J. „Pointnet++: Deep hierarchical feature learning on point sets in a metric space". *Advances in neural information processing systems* 30 (2017).

[226] Xu, Yifan; Fan, Tianqi; Xu, Mingye; Zeng, Long und Qiao, Yu. „SpiderCNN: Deep Learning on Point Sets with Parameterized Convolutional Filters". *Computer vision - ECCV 2018*. Hrsg. von Ferrari, Vittorio; Hebert, Martial; Sminchisescu, Cristian und Weiss, Yair. Lecture Notes in Computer Science. Cham: Springer, 2018, S. 90–105.

[227] Li, Jiaxin; Chen, Ben M. und Lee, Gim Hee. *SO-Net: Self-Organizing Network for Point Cloud Analysis*. 2018. https://arxiv.org/abs/1803.04249.

[228] Su, Hang; Maji, Subhransu; Kalogerakis, Evangelos und Learned-Miller, Erik. „Multi-View Convolutional Neural Networks for 3D Shape Recognition". *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.

[229] Kanezaki, Asako; Matsushita, Yasuyuki und Nishida, Yoshifumi. „Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 5010–5019.

[230] Yavartanoo, Mohsen; Kim, Eu Young und Lee, Kyoung Mu. „SPNet: Deep 3D Object Classification and Retrieval Using Stereographic Projection". Springer, Cham, 2019, S. 691–706. https://link.springer.com/chapter/10.1007/978-3-030-20873-8_44.

[231] Spruegel, Tobias C und Wartzack, Sandro. „Concept and application of automatic part-recognition with artificial neural networks for FE simulations". *DS 80-6 Proceedings of the 20th International Conference on Engineering Design (ICED 15) Vol 6: Design Methods and Tools-Part 2 Milan, Italy, 27-30.07. 15*. 2015, S. 183–194.

[232] You, Haoxuan; Feng, Yifan; Ji, Rongrong und Gao, Yue. „PVNet". *Proceedings of the 26th ACM international conference on Multimedia*. New York, NY, USA: ACM, 2018.

[233] Fawcett, Tom. *An introduction to ROC analysis*. Bd. 27. 2006.

[234] Han, Jiawei; Kamber, Micheline und Pei, Jian. *Data mining: Concepts and techniques*. 3. edition. The Morgan Kaufmann series in data management systems. Waltham, Ma: Morgan Kaufmann, 2012.

[235] Powers, David M.W. „Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation". *arXiv preprint arXiv:2010.16061* (2020).

[236] Peddireddy, Dheeraj; Fu, Xingyu; Shankar, Anirudh; Wang, Haobo; Joung, Byung Gun; Aggarwal, Vaneet; Sutherland, John W. und Jun, Martin Byung-Guk. „Identifying manufacturability and machining processes using deep 3D convolutional networks". *Journal of Manufacturing Processes* 64 (2021), S. 1336–1348.

[237] Pillai, Akhil; Reuter, Uwe und Thiele, Marko. „Estimation of spot weld design parameters using deep learning". *12th European LS-DYNA Conference*. Bd. 2019. 2019, S. 1–16.

[238] Ip, Cheuk Yiu und Regli, William C. „A 3D object classifier for discriminating manufacturing processes". *Computers & Graphics* 30.6 (2006), S. 903–916.

[239] Neb, Alexander; Schoenhof, Raoul und Briki, Iyed. „Automation Potential Analysis of Assembly Processes based on 3D Product Assembly Models in CAD Systems". *Procedia CIRP* 91 (2020), S. 237–242.

[240] Hyvärinen, Joonas. „Automating CAD assembly constraints using supervised deep learning". Diss. https://trepo.tuni.fi/handle/10024/133363.

[241] Dworschak, Fabian; Kügler, Patricia; Schleich, Benjamin und Wartzack, Sandro. „Integrating the Mechanical Domain into Seed Approach". *Proceedings of the Design Society: International Conference on Engineering Design* 1.1 (2019), S. 2587–2596.

[242] Qie, Yifan und Anwer, Nabil. „InvNet: Transfer Learning on Point Cloud for Invariant Surface Classification". *Advances on Mechanics, Design Engineering and Manufacturing IV: Proceedings of the International Joint Conference on Mechanics, Design Engineering & Advanced Manufacturing, JCM 2022, June 1-3, 2022, Ischia, Italy*. Springer. 2022, S. 1279–1289.

[243] Jin, Tian. „From Photos to 3D Design: A Product Shape Family Design and Modeling Framework with Image-Based Reconstruction and 3D Convolutional Neural Networks". Diss. Rutgers The State University of New Jersey, School of Graduate Studies, 2019. https://search.proquest.com/openview/8a2f509406698376 9b00f40e766ebb26/1?pq-origsite=gscholar&cbl=18750&diss=y.

[244] Lupinetti, Katia; Cabiddu, Daniela; Giannini, Franca und Monti, Marina. „A Web-Based Solution Supporting CAD Assembly Model Exploration and Analysis". *SN Computer Science* 3.1 (2022), S. 1–13. https://link.springer.com/article/10.1007/s42979-021-00853-9.

[245] Lupinetti, Katia; Pernot, Jean-Philippe; Monti, Marina und Giannini, Franca. „Content-based CAD assembly model retrieval: Survey and future challenges". *Computer-Aided Design* 113 (2019), S. 62–81.

[246] Chaudhari, Ashish M; Bilionis, Ilias und Panchal, Jitesh H. „Similarity in engineering design: A knowledge-based approach". *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Bd. 59278. American Society of Mechanical Engineers. 2019, V007T06A045.

[247] Dudenredaktion, (Hrsg.) *ähnlich. Duden online*. https://www.duden.de/rechtschreibung/aehnlich (besucht am 01. 03. 2022).

[248] Berchtold, Stefan. *Geometrie-basierte Suche ähnlicher Bauteile*. Ludwig Maximilians University Munich, 1997.

[249] Liu, Zhen-Bao; Bu, Shu-Hui; Zhou, Kun; Gao, Shu-Ming; Han, Jun-Wei und Wu, Jun. „A survey on partial retrieval of 3D shapes". *Journal of Computer Science and Technology* 28.5 (2013), S. 836–851.

[250] Loncaric, Sven. „A survey of shape analysis techniques". *Pattern recognition* 31.8 (1998), S. 983–1001.

[251] Li, Bo; Lu, Yijuan; Li, Chunyuan; Godil, Afzal; Schreck, Tobias; Aono, Masaki; Burtscher, Martin; Chen, Qiang; Chowdhury, Nihad Karim; Fang, Bin u. a. „A comparison of 3D shape retrieval methods based on a large-scale benchmark supporting multimodal queries". *Computer Vision and Image Understanding* 131 (2015), S. 1–27.

[252] Bustos, Benjamin; Keim, Daniel A; Saupe, Dietmar; Schreck, Tobias und Vranić, Dejan V. „Feature-based similarity search in 3D object databases". *ACM Computing Surveys (CSUR)* 37.4 (2005), S. 345–387.

[253] Madelmis, Athanasios; Daras, Petros; Tzovaras, Dimitrios und Strintzis, Michael G. „Three Dimensional Content-Based Search and Retrieval of CAD Objects". *Encyclopedia of Multimedia*. Boston, MA: Springer US, 2008, S. 853–859.

[254] Qin, Zheng; Jia, Ji und Qin, Jun. „Content based 3D model retrieval: A survey". *2008 International Workshop on Content-Based Multimedia Indexing*. IEEE. 2008, S. 249–256.

[255] Iyer, Natraj; Jayanti, Subramaniam; Lou, Kuiyang; Kalyanaraman, Yagnanarayanan und Ramani, Karthik. „Three-dimensional shape searching: state-of-the-art review and future trends". *Computer-Aided Design* 37.5 (2005), S. 509–530.

[256] Zehtaban, Leila und Roller, Dieter. „Beyond similarity comparison: Intelligent data retrieval for CAD/CAM designs". en. *Comput. Aided Des. Appl.* 10.5 (Jan. 2013), S. 789–802.

[257] Ankerst, Mihael; Kastenmüller, Gabi; Kriegel, Hans-Peter und Seidl, Thomas. „3D shape histograms for similarity search and classification in spatial databases". *Advances in Spatial Databases*. Lecture notes in computer science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, S. 207–226.

[258] Horn, Berthold Klaus Paul. „Extended gaussian images". *Proceedings of the IEEE* 72.12 (1984), S. 1671–1686.

[259] El-Mehalawi, Mohamed und Miller, R Allen. „A database system of mechanical components based on geometric and topological similarity. Part II: indexing, retrieval, matching, and similarity assessment". *Computer-Aided Design* 35.1 (2003), S. 95–105.

[260] Sundar, Hari; Silver, Deborah; Gagvani, Nikhil und Dickinson, Sven. „Skeleton based shape matching and retrieval". *2003 Shape Modeling International.* IEEE. 2003, S. 130–139.

[261] Hilaga, Masaki; Shinagawa, Yoshihisa; Kohmura, Taku und Kunii, Tosiyasu L. „Topology matching for fully automatic similarity estimation of 3D shapes". *Proceedings of the 28th annual conference on Computer graphics and interactive techniques - SIGGRAPH '01*. Not Known: ACM Press, 2001.

[262] Chen, Ding-Yun; Tian, Xiao-Pei; Shen, Yu-Te und Ouhyoung, Ming. „On visual similarity based 3D model retrieval". 22.3 (2003), S. 223–232.

[263] Lian, Zhouhui; Godil, Afzal und Sun, Xianfang. „Visual similarity based 3D shape retrieval using bag-of-features". *2010 Shape Modeling International Conference*. Aix-en-Provence, France: IEEE, Juni 2010.

[264] Alcantarilla, Pablo Fernández; Bartoli, Adrien und Davison, Andrew J. „KAZE features". *European conference on computer vision*. Springer. 2012, S. 214–227.

[265] Li, Liang; Zhang, Shusheng; Bai, Xiaoliang und Shao, Li. „Retrieving 3D model using compound-eye visual representation". *2013 International Conference on Computer-Aided Design and Computer Graphics*. IEEE. 2013, S. 172–179.

[266] Liu, Qiong. „A survey of recent view-based 3D model retrieval methods“ (Aug. 2012). arXiv: 1208.3670 [cs.CV].

[267] Papadakis, Panagiotis; Pratikakis, Ioannis; Theoharis, Theoharis und Perantonis, Stavros. „PANORAMA: A 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval“. en. *Int. J. Comput. Vis.* 89.2-3 (Sep. 2010), S. 177–192.

[268] Vranic, Dejan V. und Saupe, Dietmar. „3D model retrieval.“ Diss. Citeseer, 2004.

[269] Godil, Afzal; Lian, Zhouhui; Dutagaci, Helin; Fang, Rui; Vanamali, TP und Cheung, Chun Pan. „Benchmarks, performance evaluation and contests for 3D shape retrieval“. *Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop*. 2010, S. 42–47.

[270] Leifman, G; Katz, S; Tal, A und Meir, R. „Signatures of 3D models for retrieval“. *Proceedings of the 4th Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphics*. 2003, S. 159–163.

[271] Chen, Xiang; Gao, Shuming; Guo, Song und Bai, Jing. „A flexible assembly retrieval approach for model reuse“. *Computer-Aided Design* 44.6 (2012), S. 554–574.

[272] Wang, Pan; Li, Yuan; Zhang, Jie und Yu, Jianfeng. „An assembly retrieval approach based on shape distributions and Earth Mover's Distance“. *The International Journal of Advanced Manufacturing Technology* 86.9-12 (2016), S. 2635–2651. https://link.springer.com/article/10.1007/s00170-016-8368-z.

[273] Lupinetti, Katia; Giannini, Franca; Monti, Marina und Pernot, Jean-Philippe. „CAD assembly descriptors for knowledge capitalization and model retrieval“ (2016). https://sam.ensam.eu/handle/10985/11409.

[274] Lupinetti, Katia; Chiang, Lisa; Giannini, Franca; Monti, Marina und Pernot, Jean-Philippe. „Regular patterns of repeated elements in CAD assembly model retrieval“. *Computer-Aided Design and Applications* 14.4 (2016), S. 516–525.

[275] Lupinetti, Katia; Giannini, Franca; Monti, Marina und Pernot, Jean-Philippe. „Multi-criteria retrieval of CAD assembly models“. *Journal of Computational Design and Engineering* 5.1 (2018), S. 41–53.

[276] Lupinetti, Katia; Giannini, Franca; Monti, Marina und Pernot, Jean-Philippe. „Content-based multi-criteria similarity assessment of CAD assembly models“. *Computers in Industry* 112 (2019), S. 103111.

[277] Li, Zhi; Zhou, Xionghui; Liu, Wei; Niu, Qiang und Kong, Chuipin. „A similarity-based reuse system for injection mold design in automotive interior industry". *The International Journal of Advanced Manufacturing Technology* 87.5-8 (2016), S. 1783–1795.

[278] Kruger, Lorenz; Saske, Bernhard; Schwoch, Sebastian; Paetzold-Byhain, Kristin u. a. „Structure-based Similarity Searches for Improving Component Repetition in Plant Engineering". *DS 125: Proceedings of the 34th Symposium Design for X (DFX2023)*. 2023, S. 195–204.

[279] Charlton, C. T. und Wallace, Ken M. „A web broker for component retrieval in mechanical engineering". *Design Studies* 21.2 (2000), S. 167–186.

[280] Rea, Heather J.; Corney, Jonathan R.; Clark, Douglas E. R.; Pritchard, John; Breaks, Micheal L. und Macleod, Roddy A. „Part-sourcing in a Global Market". *Concurrent Engineering* 10.4 (2002), S. 325–333.

[281] Lou, Kuiyang; Jayanti, Subramaniam; Iyer, Natraj; Kalyanaraman, Yagnanarayanan; Prabhakar, Sunil und Ramani, Karthik. „A Reconfigurable 3D Engineering Shape Search System: Part II — Database Indexing, Retrieval, and Clustering". *ASME 2003 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (2003), S. 169–178.

[282] Iyer, Natraj; Lou, Kuiyang; Jayanti, Subramaniam; Kalyanaraman, Yagnanarayanan und Ramani, Karthik. „Early Results from 3DESS: A 3D Engineering Shape Search System" (2003).

[283] Lou, Kuiyang; Ramani, Karthik und Prabhakar, Sunil. „Content-based three-dimensional engineering shape search". *Proceedings. 20th International Conference on Data Engineering*. IEEE Comput. Soc, 2004.

[284] Iyer, Natraj; Kalyanaraman, Yagnanarayanan; Lou, Kuiyang; Jayanti, Subramaniam und Ramani, Karthik. „A Reconfigurable 3D Engineering Shape Search System: Part I — Shape Representation". *ASME 2003 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (2004), S. 89–98.

[285] Bai, Jing; Gao, Shuming; Tang, Weihua; Liu, Yusheng und Guo, Song. „Design reuse oriented partial retrieval of CAD models". *Computer-Aided Design* 42.12 (2010), S. 1069–1084.

[286] Pu, Jiantao; Kalyanaraman, Yagnanarayanan; Jayanti, Subramaniam; Ramani, Karthik und Pizlo, Zygmunt. „Navigation and discovery in 3D CAD repositories". *IEEE Computer Graphics and Applications* 27.4 (2007), S. 38–47.

[287] Regli, William C. und Cicirello, Vincent A. „Managing digital libraries for computer-aided design". *Computer-Aided Design* 32.2 (2000), S. 119–132.

[288] Iyer, Natraj; Jayanti, Subramaniam; Lou, Kuiyang; Kalyanaraman, Yagnanarayanan und Ramani, Karthik. „Shape-based searching for product lifecycle applications". *Computer-Aided Design* 37.13 (2005), S. 1435–1446.

[289] Zhang, Chao und Zhou, Guanghui. „A view-based 3D CAD model reuse framework enabling product lifecycle reuse". *Advances in Engineering Software* 127 (2019), S. 82–89.

[290] Ramanujan, Devarajan; Benjamin, William; Bernstein, William Z.; Elmqvist, Niklas und Ramani, Karthik. „ShapeSift: Suggesting Sustainable Options in Design Reuse From Part Repositories". *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (2014).

[291] Angrish, Atin; Craver, Benjamin und Starly, Binil. „"FabSearch": A 3D CAD Model-Based Search Engine for Sourcing Manufacturing Services". *Journal of Computing and Information Science in Engineering* 19.4 (2019).

[292] Giannini, Franca; Lupinetti, Katia und Monti, Marina. „Identification of Similar and Complementary Subparts in B-Rep Mechanical Models". *Journal of Computing and Information Science in Engineering* 17.4 (2017).

[293] Li, Zhi; Zhou, Xionghui und Liu, Wei. „A geometric reasoning approach to hierarchical representation for B-rep model retrieval". *Computer-Aided Design* 62 (2015), S. 190–202.

[294] You, Chun-Fong und Tsai, Yi-Lung. „3D solid model retrieval for engineering reuse based on local feature correspondence". *The International Journal of Advanced Manufacturing Technology* 46.5-8 (2010), S. 649–661. https://link.springer.com/article/10.1007/s00170-009-2113-9.

[295] El-Mehalawi, Mohamed und Allen Miller, R. „A database system of mechanical components based on geometric and topological similarity. Part I: representation". *Computer-Aided Design* 35.1 (2003), S. 83–94.

[296] Ascher, M.; Marefat, M. M. und Fasse, E. „A methodology for automatic retrieval of similarly shaped machinable components". *IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236)*. IEEE, 2001.

[297] McWherter, David; Peabody, Mitchell; Regli, William C. und Shokoufandeh, Ali. „Transformation Invariant Shape Similarity Comparison of Solid Models". *Volume 3: 6th Design for Manufacturing Conference*. American Society of Mechanical Engineers, 2001, S. 303–312.

[298] Flynn, Patrick J. und Jain, Anil K. „CAD-based computer vision: from CAD models to relational graphs". *Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 1989, S. 162–167.

[299] Joshi, S. und Chang, T. C. „Graph-based heuristics for recognition of machined features from a 3D solid model". *Computer-Aided Design* 20.2 (1988), S. 58–66.

[300] Ip, Cheuk Yiu; Lapadat, Daniel; Sieger, Leonard und Regli, William C. „Using shape distributions to compare solid models". *Proceedings of the seventh ACM symposium on Solid modeling and applications*. New York, NY, USA: ACM, 2002.

[301] Ramesh, Madhumati; Yip-Hoi, Derek und Dutta, Debasish. „Feature Based Shape Similarity Measurement for Retrieval of Mechanical Parts". *Journal of Computing and Information Science in Engineering* 1.3 (2001), S. 245–256.

[302] Mukai, Shinji; Furukawa, Susumu und Kuroda, Mitsuru. „An algorithm for deciding similarities of 3-D objects". *Proceedings of the seventh ACM symposium on Solid modeling and applications*. New York, NY, USA: ACM, 2002.

[303] Chu, Chih-Hsing; Cheng, Han-Chung; Wang, Eric und Kim, Yong-Se. „ANN-based 3D part search with different levels of detail (LOD) in negative feature decomposition". *Expert Systems with Applications* 36.8 (2009), S. 10905–10913. https://www.sciencedirect.com/science/article/pii/S0957417409001559.

[304] Cheng, Han-Chung; Chu, Chih-Hsing; Wang, Eric und Kim, Yong-Se. „3D Part Similarity Comparison based on Levels of Detail in Negative Feature Decomposition Using Artificial Neural Network". *Computer-Aided Design and Applications* 4.5 (2007), S. 619–628.

[305] Chu, Chih-Hsing und Hsu, Yung-Chang. „Similarity assessment of 3D mechanical components for design reuse". *Robotics and Computer-Integrated Manufacturing* 22.4 (2006), S. 332–341. https://www.sciencedirect.com/science/article/pii/S0736584505000591.

[306] Hong, Taesik; Lee, Kunwoo und Kim, Sungchan. „Similarity comparison of mechanical parts to reuse existing designs". *Computer-Aided Design* 38.9 (2006), S. 973–984. https://www.sciencedirect.com/science/article/pii/S0010448506000911.

[307] Bai, Jing; Liu, Yusheng und Gao, Shuming. „Multi-mode Solid Model Retrieval Based on Partial Matching". *2007 10th IEEE International Conference on Computer-Aided Design and Computer Graphics*. Hrsg. von Wang, Guoping. Piscataway, NJ: IEEE Service Center, 2007, S. 126–131.

[308] Mun, Duhwan; Cho, Junmyon und Ramani, Karthik. „A Method for Measuring Part Similarity Using Ontology and a Multi-Criteria Decision Making Method". *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (2010), S. 419–430.

[309] Zhang, Ruzhen und Zhou, Xionghui. „Similarity Assessment of Mechanical Parts Based on Integrated Product Information Model". *Journal of Computing and Information Science in Engineering* 11.1 (2011).

[310] Cicirello, Vincent A. und Regli, William C. „An approach to a feature-based comparison of solid models of machined parts". *AI EDAM* 16.5 (2002), S. 385–399.

[311] Tsai, Chieh-Yuan und Chang, C. Alec. „Fuzzy neural networks for intelligent design retrieval using associative manufacturing features". *Journal of Intelligent Manufacturing* 14.2 (2003), S. 183–195.

[312] Cicirello, V. und Regli, W. C. „Machining feature-based comparisons of mechanical parts". *Proceedings International Conference on Shape Modeling and Applications*. IEEE Computer Society, 2001.

[313] Elinson, Alexei; Nau, Dana S. und Regli, William C. „Feature-based similarity assessment of solid models". *Proceedings of the fourth ACM symposium on Solid modeling and applications - SMA '97*. New York, New York, USA: ACM Press, 1997.

[314] Huang, Bo; Zhang, Shusheng; Huang, Rui; Li, Xiuling und Zhang, Yajun. „An effective retrieval approach of 3D CAD models for macro process reuse". *The International Journal of Advanced Manufacturing Technology* 102.5-8 (2019), S. 1067–1089. https://link.springer.com/article/10.1007/s00170-018-2968-8.

[315] Xu, Changhong; Zhang, Shusheng; Huang, Bo; Li, Xiuling und Huang, Rui. „NC process reuse oriented effective subpart retrieval approach of 3D CAD models". *Computers in Industry* 90 (2017), S. 1–16.

[316] Huang, Rui; Zhang, Shusheng; Bai, Xiaoliang; Xu, Changhong und Huang, Bo. „An effective subpart retrieval approach of 3D CAD models for manufacturing process reuse". *Computers in Industry* 67 (2015), S. 38–53.

[317] Huang, Rui; Zhang, Shusheng; Bai, Xiaoliang; Xu, Changhong und Huang, Bo. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 229.7 (2015), S. 1229–1242.

[318] Krishna, Vamsi; Babu, B. Surendra; Rao, D. Nageswara und Shankar, N. V. S. „An approach to feature-based three-dimensional modelling and similarity assessment of turned components". *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 221.11 (2007), S. 1651–1656.

[319] Cardone, Antonio; Gupta, Satyandra K.; Deshmukh, Abhijit und Karnik, Mukul. „Machining feature-based similarity assessment algorithms for prismatic machined parts". *Computer-Aided Design* 38.9 (2006), S. 954–972. https://www.sciencedirect.com/science/article/pii/S0010448506001369.

[320] Dawei, Wang; Guangrong, Yan; Yi, Lei und Zhang, Jiaying. „A Retrieval Algorithm of Sheet Metal Parts Based on Relationships of Features". *Chinese Journal of Aeronautics* 25.3 (2012), S. 453–472. https://www.sciencedirect.com/science/article/pii/S100093611160410X.

[321] You, Chun-Fong; Tsai, Yi-Lung und Liu, Kun-Yu. „Representation and similarity assessment in case-based process planning and die design for manufacturing automotive panels". *The International Journal of Advanced Manufacturing Technology* 51.1-4 (2010), S. 297–310. https://link.springer.com/article/10.1007/s00170-010-2609-3.

[322] Li, Min; Fuh, J. Y. H.; Zhang, Y. F. und Qiu, Z. M. „General and Partial Shape Matching Approaches on Feature-Based CAD Models to Support Efficient Part Retrieval". *28th Computers and Information in Engineering Conference : presented at [the] 2008 ASME International Design Engineering Technical Conferences and Computers and*

Information in Engineering Conference : August 3-6, 2008, New York City, New York USA. Hrsg. von Author, Computers und Engineering Conference Corporate, Information in. ASME, 2008, S. 121–130.

[323] Li, Min; Zhang, Y. F. und Fuh, J. Y. H. „Retrieving Reusable 3D CAD Models Using Knowledge-Driven Dependency Graph Partitioning". *Computer-Aided Design and Applications* 7.3 (2010), S. 417–430.

[324] Martino, T. de; Falcidieno, B.; Giannini, F.; Hassinger, S. und Ovtcharova, J. „Feature-based modelling by integrating design and recognition approaches". *Computer-Aided Design* 26.8 (1994), S. 646–653.

[325] Chang, C. Alec und Tsai, Chieh Yuan. „Using ART1 neural networks with destructive solid geometry for design retrieving systems". *Computers in Industry* 34.1 (1997), S. 27–41.

[326] Qin, Feiwei; Gao, Shuming; Yang, Xiaoling; Li, Ming und Bai, Jing. „An ontology-based semantic retrieval approach for heterogeneous 3D CAD models". *Advanced Engineering Informatics* 30.4 (2016), S. 751–768.

[327] Gopalakrishnan, Sankara Hari und Suresh, Krishnan. „Exploiting Shape Similarity in Engineering Analysis" (2008), S. 63–71.

[328] Cardone, Antonio; Gupta, Satyandra K und Karnik, Mukul. „A survey of shape similarity assessment algorithms for product design and manufacturing applications". *J. Comput. Inf. Sci. Eng.* 3.2 (2003), S. 109–118.

[329] McWherter, David; Peabody, Mitchell; Regli, William C und Shokoufandeh, Ali. „Transformation invariant shape similarity comparison of solid models". *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Bd. 80241. American Society of Mechanical Engineers. 2001, S. 303–312.

[330] Cicirello, Vincent A und Regli, William C. „Resolving non-uniqueness in design feature histories". *Proceedings of the fifth ACM symposium on Solid modeling and applications*. 1999, S. 76–84.

[331] Zhang, Xianlin; Li, Xueming; Liu, Yang und Feng, Fangxiang. „A survey on freehand sketch recognition and retrieval". *Image and Vision Computing* 89 (2019), S. 67–87.

[332] Xu, Peng; Hospedales, Timothy M; Yin, Qiyue; Song, Yi-Zhe; Xiang, Tao und Wang, Liang. „Deep learning for free-hand sketch: A survey". *IEEE transactions on pattern analysis and machine intelligence* 45.1 (2022), S. 285–312.

[333] Cole, Forrester; Golovinskiy, Aleksey; Limpaecher, Alex; Barros, Heather Stoddart; Finkelstein, Adam; Funkhouser, Thomas und Rusinkiewicz, Szymon. „Where do people draw lines?“ *ACM SIGGRAPH 2008 papers*. 2008, S. 1–11.

[334] Saavedra, Jose M und Bustos, Benjamin. „An improved histogram of edge local orientations for sketch-based image retrieval“. *Pattern Recognition: 32nd DAGM Symposium, Darmstadt, Germany, September 22-24, 2010. Proceedings 32*. Springer. 2010, S. 432–441.

[335] Li, Bo; Lu, Yijuan; Godil, Afzal; Schreck, Tobias; Bustos, Benjamin; Ferreira, Alfredo; Furuya, Takahiko; Fonseca, Manuel J; Johan, Henry; Matsuda, Takahiro u. a. „A comparison of methods for sketch-based 3D shape retrieval“. *Computer Vision and Image Understanding* 119 (2014), S. 57–80.

[336] Eitz, Mathias; Hays, James und Alexa, Marc. „How do humans sketch objects?“ *ACM Transactions on graphics (TOG)* 31.4 (2012), S. 1–10.

[337] Eitz, Mathias; Richter, Ronald; Boubekeur, Tamy; Hildebrand, Kristian und Alexa, Marc. „Sketch-based shape retrieval“. *ACM Transactions on graphics (TOG)* 31.4 (2012), S. 1–10.

[338] Daras, Petros und Axenopoulos, Apostolos. „A 3D shape retrieval framework supporting multimodal queries“. *International Journal of Computer Vision* 89 (2010), S. 229–247.

[339] Yoon, Sang Min; Scherer, Maximilian; Schreck, Tobias und Kuijper, Arjan. „Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours“. *Proceedings of the 18th ACM international conference on Multimedia*. 2010, S. 193–200.

[340] Lu, Dawei; Ma, Huadong und Fu, Huiyuan. „Efficient Sketch-based 3D shape retrieval via view selection“. *Advances in Multimedia Information Processing–PCM 2013: 14th Pacific-Rim Conference on Multimedia, Nanjing, China, December 13-16, 2013. Proceedings 14*. Springer. 2013, S. 396–407.

[341] Kurita, Yukinori und Ohbuchi, Ryutarou. „View-Clustering and Manifold Learning for Sketch-Based 3D Model Retrieval“. *2013 International Conference on Cyberworlds*. IEEE. 2013, S. 282–285.

[342] Li, Bo und Johan, Henry. „Sketch-based 3D model retrieval by incorporating 2D-3D alignment“. *Multimedia tools and applications* 65 (2013), S. 363–385.

[343] Li, Bo; Lu, Yijuan und Shen, Jian. „A semantic tree-based approach for sketch-based 3D model retrieval“. *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE. 2016, S. 3880–3885.

[344] Sousa, Pedro und Fonseca, Manuel J. „Sketch-based retrieval of drawings using spatial proximity". *Journal of Visual Languages & Computing* 21.2 (2010), S. 69–80.

[345] Fonseca, Manuel J; Ferreira, Alfredo und Jorge, Joaquim A. „Sketch-based retrieval of vector drawings". *Sketch-based Interfaces and Modeling* (2011), S. 181–201.

[346] Li, Hanhui; Wu, Hefeng; He, Xiangjian; Lin, Shujin; Wang, Ruomei und Luo, Xiaonan. „Multi-view pairwise relationship learning for sketch based 3D shape retrieval". *2017 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE. 2017, S. 1434–1439.

[347] Zhu, Fan; Xie, Jin und Fang, Yi. „Learning cross-domain neural networks for sketch-based 3D shape retrieval". *Proceedings of the AAAI conference on artificial intelligence*. Bd. 30. 1. 2016.

[348] Ye, Yuxiang; Li, Bo und Lu, Yijuan. „3D sketch-based 3D model retrieval with convolutional neural network". *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE. 2016, S. 2936–2941.

[349] Chen, Jiaxin und Fang, Yi. „Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3d shape retrieval". *Proceedings of the European conference on computer vision (ECCV)*. 2018, S. 605–620.

[350] Chen, Jiaxin; Qin, Jie; Liu, Li; Zhu, Fan; Shen, Fumin; Xie, Jin und Shao, Ling. „Deep sketch-shape hashing with segmented 3d stochastic viewing". *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, S. 791–800.

[351] Dai, Guoxian; Xie, Jin; Zhu, Fan und Fang, Yi. „Deep correlated metric learning for sketch-based 3d shape retrieval". *Proceedings of the AAAI Conference on Artificial Intelligence*. Bd. 31. 1. 2017.

[352] Qi, Anran; Song, Yi-Zhe und Xiang, Tao. „Semantic Embedding for Sketch-Based 3D Shape Retrieval." *BMVC*. Bd. 3. 2018, S. 11–12.

[353] Kuwabara, Shutaro; Ohbuchi, Ryutarou und Furuya, Takahiko. „Query by partially-drawn sketches for 3D shape retrieval". *2019 International Conference on Cyberworlds (CW)*. IEEE. 2019, S. 69–76.

[354] Pu, Jiantao; Lou, Kuiyang und Ramani, Karthik. „A 2D sketch-based user interface for 3D CAD model retrieval". *Computer-aided design and applications* 2.6 (2005), S. 717–725.

[355] Pu, Jiantao und Ramani, Karthik. „A 3D model retrieval method using 2D freehand sketches". *Computational Science–ICCS 2005: 5th International Conference, Atlanta, GA, USA, May 22-25, 2005. Proceedings, Part II 5*. Springer. 2005, S. 343–346.

[356] Hou, Suyu und Ramani, Karthik. „Sketch-based 3D Engineering Part Class Browsing and Retrieval." *SBM*. 2006, S. 131–138.

[357] Hou, Suyu und Ramani, Karthik. „Classifier combination for sketch-based 3D part retrieval". *Computers & Graphics* 31.4 (2007), S. 598–609.

[358] Wang, Xinxiang; Wang, Jiale und Pan, Weifeng. „A sketch-based query interface for 3D CAD model retrieval". *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*. IEEE. 2014, S. 881–885.

[359] Kanai, Satoshi. „Content-based 3D mesh model retrieval from hand-written sketch". *International Journal on Interactive Design and Manufacturing (IJIDeM)* 2 (2008), S. 87–98.

[360] Liu, Yong-Jin; Luo, Xi; Joneja, Ajay; Ma, Cui-Xia; Fu, Xiao-Lan und Song, Dawei. „User-adaptive sketch-based 3-D CAD model retrieval". *IEEE Transactions on Automation Science and Engineering* 10.3 (2013), S. 783–795.

[361] Manda, Bharadwaj; Dhayarkar, Shubham; Mitheran, Sai; Viekash, VK und Muthuganapathy, Ramanathan. „'CADSketchNet'-An annotated sketch dataset for 3D CAD model retrieval with deep neural networks". *Computers & Graphics* 99 (2021), S. 100–113.

[362] Li, Bo; Lu, Yijuan; Ghumman, Azeem; Strylowski, Bradley; Gutierrez, Mario; Sadiq, Safiyah; Forster, Scott; Feola, Natacha und Bugerin, Travis. „KinectSBR: A kinect-assisted 3D sketch-based 3D model retrieval system". *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. 2015, S. 655–656.

[363] Wang, Jiale; He, Yuanjun; Tian, Haishan und Cai, Hongming. „Retrieving 3D CAD model by freehand sketches for design reuse". *Advanced Engineering Informatics* 22.3 (2008), S. 385–392.

[364] Qin, Fei-wei; Gao, Shu-ming; Yang, Xiao-ling; Bai, Jing und Zhao, Qu-hong. „A sketch-based semantic retrieval approach for 3D CAD models". *Applied Mathematics-A Journal of Chinese Universities* 32.1 (2017), S. 27–52.

[365] Qin, Feiwei; Qiu, Shi; Gao, Shuming und Bai, Jing. „3D CAD model retrieval based on sketch and unsupervised variational autoencoder". *Advanced Engineering Informatics* 51 (2022), S. 101427.

[366] Wen, Rui; Tang, Weiqing und Su, Zhiyong. „Topology based 2D engineering drawing and 3D model matching for process plant". *Graphical Models* 92 (2017), S. 1–15.

[367] Qin, Sheng-Feng; Wright, David K und Jordanov, Ivan N. „A conceptual design tool: a sketch and fuzzy logic based system". *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 215.1 (2001), S. 111–116.

[368] Matondang, Muhammad Zaini; Mardzuki, Samihah und Haron, Habibollah. „Transformation of engineering sketch to valid solid object". *Proc. of Intl. Conf. of The 9th Asia Pacific Industrial Engineering & Management Systems (APIEMS 08) Conference and The 11th Asia Pacific Regional Meeting of International Foundation for Production Research, Bali–Indonesia*. 2008, S. 2707–2715.

[369] Keshavarzi, Mohammad; Hotson, Clayton; Cheng, Chin-Yi; Nourbakhsh, Mehdi; Bergin, Michael und Rahmani Asl, Mohammad. „Sketchopt: Sketch-based parametric model retrieval for generative design". *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021, S. 1–6.

[370] Cheddi, Fatima; Serrhini, Mohammed und Ahmed, Tahiri. „From Sketch to Simulation in Virtual Laboratory Using Deep Learning". *Digital Technologies and Applications: Proceedings of ICDTA'22, Fez, Morocco, Volume 1*. Springer, 2022, S. 219–228.

[371] Tian, Chao; Masry, Mark und Lipson, Hod. „Physical sketching: Reconstruction and analysis of 3D objects from freehand sketches". *Computer-Aided Design* 41.3 (2009), S. 147–158.

[372] Andrews, Peter T. J.; Shahin, Tamer M.M. und Sivaloganathan, Sangarappillai. „Design reuse in a CAD environment — Four case studies". *Computers & Industrial Engineering* 37.1-2 (1999), S. 105–109.

[373] Schneidermeier, Tim. „Design Reuse – Usability als Schlüsselfaktor für nachhaltiges Produktdesign". *Information - Wissenschaft & Praxis* 63.5 (2012).

[374] Duffy, Alex H. B. und Ferns, A. F. „AN ANALYSIS OF DESIGN REUSE BENEFITS". *INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN*. Bd. 26. 1999. https://strathprints.strath.ac.uk/18794.

[375] van Heerden, Albert S.J.; Guenov, Marin D. und Molina-Cristóbal, Arturo. „Evolvability and design reuse in civil jet transport aircraft". *Progress in Aerospace Sciences* 108 (2019), S. 121–155.

[376] Siddique, Zahed und Repphun, Bill. „Estimating cost savings when implementing a product platform approach". *Concurrent Engineering* 9.4 (2001), S. 285–294.

[377] Busby, Jeremy S. „The problem with design reuse: an investigation into outcomes and antecedents". *Journal of Engineering Design* 10.3 (1999), S. 277–296.

[378] El Hani, M. A.; Rivest, Louis und Maranzana, Roland. „Product Data Reuse in Product Development: A Practitioner's Perspective". *Product lifecycle management*. Hrsg. von Rivest, Louis; Bouras, Abdelaziz und Louhichi, Borhen. IFIP International Federation for Information Processing. Heidelberg: Springer, 2012, S. 243–256. https://link.springer.com/chapter/10.1007/978-3-642-35758-9_21.

[379] Ezoji, Amer; Boujut, Jean-François und Pinquié, Romain. „Requirements for design reuse in open-source hardware: a state of the art". *Procedia CIRP* 100 (2021), S. 792–797.

[380] Long, Daniel und Ferguson, Scott. „A case study of evolvability and excess on the B-52 Stratofortress and F/A-18 Hornet". *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Bd. 58165. American Society of Mechanical Engineers. 2017, V004T05A026.

[381] Lim, Dongwook. *A systematic approach to design for lifelong aircraft evolution*. Georgia Institute of Technology, 2009.

[382] Sivaloganathan, Sangarappillai und Shahin, Tamer M. M. „Design reuse: An overview". *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 213.7 (1999), S. 641–654.

[383] Camba, Jorge D.; Contero, Manuel und Company, Pedro. „Parametric CAD modeling: An analysis of strategies for design reusability". *Computer-Aided Design* 74 (2016), S. 18–31. https://www.sciencedirect.com/science/article/pii/S0010448516000051.

[384] Huang, Rui; Zhang, Shusheng; Bai, Xiaoliang; Xu, Changhong und Huang, Bo. „An effective subpart retrieval approach of 3D CAD models for manufacturing process reuse". *Computers in industry* 67 (2015), S. 38–53.

[385] Clark, Alex. *Pillow (PIL Fork) Documentation*. 2015. https://buildmedia.readthedocs.org/media/pdf/pillow/latest/pillow.pdf.

[386] Seff, Ari; Ovadia, Yaniv; Zhou, Wenda und Adams, Ryan P. „Sketchgraphs: A large-scale dataset for modeling relational geometry in computer-aided design". *arXiv preprint arXiv:2007.08506* (2020).

[387] Abdulla, Waleed. *Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow*. https://github.com/matterport/Mask_RCNN. 2017.

[388] Jocher, Glenn; Ayush Chaurasia and Jirka Borovec and NanoCode012 and TaoXie and Yonghye Kwon and Kalen Michael and Liu Changyu and Jiacong Fang and Abhiram V and Laughing and tkianai and yxNONG and Piotr Skalski and Adam Hogan and Jebastin Nadar and imyhxy and Lorenzo Mammana, Alex Stoken and; AlexWang1900; Fati, Cristi; Montes, Diego; Hajek, Jan; Diaconu, Laurentiu; Minh, Mai Thanh; Marc; albinxavi; fatih; oleg und wanghaoyang0106. *ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support*. Version v6.0. Okt. 2021. https://doi.org/10.5281/zenodo.5563715.

[389] Wu, Yuxin; Kirillov, Alexander; Massa, Francisco; Lo, Wan-Yen und Girshick, Ross. *Detectron2*. https://github.com/facebookresearch/detectron2. 2019.

[390] Carion, Nicolas; Massa, Francisco; Synnaeve, Gabriel; Usunier, Nicolas; Kirillov, Alexander und Zagoruyko, Sergey. „End-to-end object detection with transformers". *European conference on computer vision*. Springer. 2020, S. 213–229.

[391] Han, Kai; Wang, Yunhe; Chen, Hanting; Chen, Xinghao; Guo, Jianyuan; Liu, Zhenhua; Tang, Yehui; Xiao, An; Xu, Chunjing; Xu, Yixing u. a. „A survey on visual transformer". *arXiv preprint arXiv:2012.12556* (2020).

[392] Wang, Yongxin; Kitani, Kris und Weng, Xinshuo. „Joint object detection and multi-object tracking with graph neural networks". *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, S. 13708–13715.

[393] Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, Xiaowei u. a. „A density-based algorithm for discovering clusters in large spatial databases with noise". *kdd*. Bd. 96. 34. 1996, S. 226–231.

[394] Fisher, Robert; Perkins, Simon; Walker, Ashley und Wolfart, Erik. „Hypermedia image processing reference". *England: John Wiley & Sons Ltd* (1996), S. 118–130.

[395] Hagberg, Aric; Swart, Pieter und S Chult, Daniel. *Exploring network structure, dynamics, and function using NetworkX*. Techn. Ber. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[396] McKinney, Wes. „Data Structures for Statistical Computing in Python". *Proceedings of the 9th Python in Science Conference*. Hrsg. von Walt, Stéfan van der und Millman, Jarrod. 2010, S. 56–61.

[397] The pandas, development team. *pandas-dev/pandas: Pandas*. Feb. 2020. https://doi.org/10.5281/zenodo.3509134.

[398] Abu-Aisheh, Zeina; Raveaux, Romain; Ramel, Jean-Yves und Martineau, Patrick. „An exact graph edit distance algorithm for solving pattern recognition problems". *4th International Conference on Pattern Recognition Applications and Methods 2015*. 2015.

[399] Levenshtein, Vladimir I. „Binary codes capable of correcting deletions, insertions, and reversals". *Soviet physics doklady*. Bd. 10. 8. Soviet Union. 1966, S. 707–710.

[400] Cordella, Luigi Pietro; Foggia, Pasquale; Sansone, Carlo; Vento, Mario u. a. „An improved algorithm for matching large graphs". *3rd IAPR-TC15 workshop on graph-based representations in pattern recognition*. 2001, S. 149–159.

[401] Cordella, Luigi Pietro; Foggia, Pasquale; Sansone, Carlo und Vento, Mario. „A (sub) graph isomorphism algorithm for matching large graphs". *IEEE transactions on pattern analysis and machine intelligence* 26.10 (2004), S. 1367–1372.

[402] Nyamsi, Eric A. *IT-Lösungen auf Basis von SysML und UML: Anwendungsentwicklung mit Eclipse UML Designer und Eclipse Papyrus*. Springer-Verlag, 2020.

[403] Wolny, Sabine; Mazak, Alexandra; Carpella, Christine; Geist, Verena und Wimmer, Manuel. „Thirteen years of SysML: a systematic mapping study". *Software and Systems Modeling* 19.1 (Mai 2019), S. 111–169. http://dx.doi.org/10.1007/s10270-019-00735-y.

[404] Friedenthal, Sanford; Moore, Alan und Steiner, Rick. „OMG Systems Modeling Language (OMG SysML™) Tutorial". *INCOSE International Symposium* 18 (Juni 2008), S. 1731–1862.

[405] Perry, Simon. *SysML for Systems Engineering: A Model-Based Approach*. Computing. Institution of Engineering und Technology, 2018. https://digital-library.theiet.org/content/books/pc/pbpc020e.

[406] Goetz, Stefan; Schleich, Benjamin und Wartzack, Sandro. „A new approach to first tolerance evaluations in the conceptual design stage based on tolerance graphs". *Procedia CIRP* 75 (2018), S. 167–172.

[407] Kestel, Philipp; Kügler, Patricia; Zirngibl, Christoph; Schleich, Benjamin und Wartzack, Sandro. „Ontology-based approach for the provision of simulation knowledge acquired by Data and Text Mining processes". *Advanced Engineering Informatics* 39 (2019), S. 292–305.

[408] ISO, 10303-21. *Industrielle Automatisierungssysteme und Integration - Produktdatendarstellung und -austausch - Teil 21: Implementierungsmethoden: Klartext-Kodierung der Austauschstruktur (ISO 10303-21:1994)*. Standard. DIN Deutsches Institut für Normung e. V., März 1996.

[409] Paviot, Thomas. *pythonocc*. Version 7.7.0. Dez. 2022. https://doi.org/10.5281/zenodo.7471333.

[410] Blanding, Doug. *kodaCAD*. 2024. https://github.com/dblanding/kodacad (besucht am 22. 01. 2024).

[411] OPEN CASCADE, SAS. *OpenCascade Technology Overview*. 2024. https://dev.opencascade.org/ (besucht am 22. 01. 2024).

[412] Dawson-Haggerty et al. *trimesh*. Version 3.2.0. https://trimesh.org/ (besucht am 08. 12. 2019).

[413] Pearson, Karl. „LIII. On lines and planes of closest fit to systems of points in space". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), S. 559–572. eprint: https://doi.org/10.1080/14786440109462720. https://doi.org/10.1080/14786440109462720.

[414] Hotelling, Harold. „Analysis of a complex of statistical variables into principal components." *Journal of educational psychology* 24.6 (1933), S. 417.

[415] Szegedy, Christian; Ioffe, Sergey; Vanhoucke, Vincent und Alemi, Alexander. „Inception-v4, inception-resnet and the impact of residual connections on learning". *Proceedings of the AAAI conference on artificial intelligence*. Bd. 31. 1. 2017.

[416] Huang, Gao; Liu, Zhuang; van der Maaten, Laurens und Weinberger, Kilian Q. „Densely Connected Convolutional Networks". *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[417] Szegedy, Christian; Vanhoucke, Vincent; Ioffe, Sergey; Shlens, Jon und Wojna, Zbigniew. „Rethinking the inception architecture for computer vision". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, S. 2818–2826.

[418] Tan, Mingxing und Le, Quoc. „Efficientnet: Rethinking model scaling for convolutional neural networks". *International conference on machine learning*. PMLR. 2019, S. 6105–6114.

[419] Li, Yangyan; Bu, Rui; Sun, Mingchao; Wu, Wei; Di, Xinhan und Chen, Baoquan. „Pointcnn: Convolution on x-transformed points". *Advances in neural information processing systems* 31 (2018).

[420] Xu, Yifan; Fan, Tianqi; Xu, Mingye; Zeng, Long und Qiao, Yu. „Spidercnn: Deep learning on point sets with parameterized convolutional filters". *Proceedings of the European conference on computer vision (ECCV)*. 2018, S. 87–102.

[421] Brock, Andrew; Lim, Theodore; Ritchie, James M und Weston, Nick. „Generative and discriminative voxel modeling with convolutional neural networks". *arXiv preprint arXiv:1608.04236* (2016).

[422] Conte, Donatello; Foggia, Pasquale; Sansone, Carlo und Vento, Mario. „Thirty years of graph matching in pattern recognition". *International journal of pattern recognition and artificial intelligence* 18.03 (2004), S. 265–298.

[423] Ullmann, Julian R. „An algorithm for subgraph isomorphism". *Journal of the ACM (JACM)* 23.1 (1976), S. 31–42.

[424] Bunke, Horst. „On a relation between graph edit distance and maximum common subgraph". *Pattern recognition letters* 18.8 (1997), S. 689–694.

[425] Houbraken, Maarten; Demeyer, Sofie; Michoel, Tom; Audenaert, Pieter; Colle, Didier und Pickavet, Mario. „The Index-based Subgraph Matching Algorithm with General Symmetries (ISMAGS): exploiting symmetry for faster subgraph enumeration". *PloS one* 9.5 (2014), e97896.

[426] Bunke, Horst und Shearer, Kim. „A graph distance metric based on the maximal common subgraph". *Pattern recognition letters* 19.3-4 (1998), S. 255–259.

[427] Neuhaus, Michel und Bunke, Horst. *Bridging the gap between graph edit distance and kernel machines*. Bd. 68. World Scientific, 2007.

[428] Riesen, Kaspar. „Structural pattern recognition with graph edit distance". *Advances in computer vision and pattern recognition* (2015).

[429] Solé-Ribalta, Albert; Serratosa, Francesc und Sanfeliu, Alberto. „On the graph edit distance cost: properties and applications". *International Journal of Pattern Recognition and Artificial Intelligence* 26.05 (2012), S. 1260004.

[430] Koutra, Danai; Parikh, Ankur; Ramdas, Aaditya und Xiang, Jing. „Algorithms for graph similarity and subgraph matching". *Proc. Ecol. inference conf.* Bd. 17. Citeseer. 2011.

[431] Jaccard, Paul. „Lois de distribution florale dans la zone alpine". *Bull Soc Vaudoise Sci Nat* 38 (1902), S. 69–130.

[432] Yang, Jingkang; Zhou, Kaiyang und Liu, Ziwei. „Full-spectrum out-of-distribution detection". *International Journal of Computer Vision* (2023), S. 1–16.

[433] Yang, Jingkang; Zhou, Kaiyang; Li, Yixuan und Liu, Ziwei. *Generalized Out-of-Distribution Detection: A Survey*. 2024. arXiv: 2110.11334 [cs.CV].

[434] Malik, Momin M. „A hierarchy of limitations in machine learning". *arXiv preprint arXiv:2002.05193* (2020).

[435] Geng, Chuanxing; Huang, Sheng-jun und Chen, Songcan. „Recent Advances in Open Set Recognition: A survey". *IEEE transactions on pattern analysis and machine intelligence* 43.10 (2020), S. 3614–3631.

[436] Fumera, Giorgio und Roli, Fabio. „Support vector machines with embedded reject option". *Pattern Recognition with Support Vector Machines: First International Workshop, SVM 2002 Niagara Falls, Canada, August 10, 2002 Proceedings*. Springer. 2002, S. 68–82.

[437] Tax, David M.J. und Duin, Robert P.W. „Growing a multi-class classifier with a reject option". *Pattern Recognition Letters* 29.10 (2008), S. 1565–1570. https://www.sciencedirect.com/science/article/pii/S016786550800113X.

[438] Perera, Pramuditha; Oza, Poojan und Patel, Vishal M. *One-Class Classification: A Survey*. 2021. arXiv: 2101.03064 [cs.CV].

[439] Seliya, Naeem; Abdollah Zadeh, Azadeh und Khoshgoftaar, Taghi M. „A literature review on one-class classification and its potential applications in big data". *Journal of Big Data* 8.1 (2021), S. 1–31.

[440] Khan, Shehroz S und Madden, Michael G. „A survey of recent trends in one class classification". *Artificial Intelligence and Cognitive Science: 20th Irish Conference, AICS 2009, Dublin, Ireland, August 19-21, 2009, Revised Selected Papers 20*. Springer. 2010, S. 188–197.

[441] Cover, Thomas und Hart, Peter. „Nearest neighbor pattern classification". *IEEE transactions on information theory* 13.1 (1967), S. 21–27.

[442] Wang, Zhou; Bovik, Alan C; Sheikh, Hamid R und Simoncelli, Eero P. „Image quality assessment: from error visibility to structural similarity". *IEEE transactions on image processing* 13.4 (2004), S. 600–612.

[443] The MathWorks, Inc. *2-D correlation coefficient*. Natick, Massachusetts, United States, 2022. https://de.mathworks.com/help/images/ref/corr2.html.

[444] The MathWorks, Inc. *Structural similarity (SSIM) index for measuring image quality*. Natick, Massachusetts, United States, 2022. https://de.mathworks.com/help/images/ref/ssim.html.

[445] The MathWorks, Inc. *Peak signal-to-noise ratio (PSNR)*. Natick, Massachusetts, United States, 2022. https://de.mathworks.com/help/images/ref/psnr.html.

[446] Mathews, Anu; Sejal, N und Venugopal, KR. „Analysis of Content Based Image Retrieval using Deep Feature Extraction and Similarity Matching". *International Journal of Advanced Computer Science and Applications* 13.12 (2022).

[447] Peng, Xushan; Zhang, Xiaoming; Li, Yongping und Liu, Bangquan. „Research on image feature extraction and retrieval algorithms based on convolutional neural network". *Journal of Visual Communication and Image Representation* 69 (2020), S. 102705.

[448] Liu, Ruoyu; Wei, Shikui; Zhao, Yao und Yang, Yi. „Indexing of the CNN features for the large scale image search". *Multimedia Tools and Applications* 77 (2018), S. 32107–32131.

[449] Yuan, Zheng-Wu und Zhang, Jun. „Feature extraction and image retrieval based on AlexNet". *Eighth International Conference on Digital Image Processing (ICDIP 2016)*. Bd. 10033. SPIE. 2016, S. 65–69.

[450] Ayesha, Shaeela; Hanif, Muhammad Kashif und Talib, Ramzan. „Overview and comparative study of dimensionality reduction techniques for high dimensional data". *Information Fusion* 59 (2020), S. 44–58.

[451] Lee, John A und Verleysen, Michel. „Unsupervised dimensionality reduction: Overview and recent advances". *The 2010 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2010, S. 1–8.

[452] Sorzano, Carlos Oscar Sánchez; Vargas, Javier und Montano, A Pascual. „A survey of dimensionality reduction techniques". *arXiv preprint arXiv:1403.2877* (2014).

[453] Fodor, Imola K. *A survey of dimension reduction techniques*. Techn. Ber. Lawrence Livermore National Lab., CA (US), 2002.

[454] Singh, Archana; Yadav, Avantika und Rana, Ajay. „K-means with Three different Distance Metrics". *International Journal of Computer Applications* 67.10 (2013).

[455] Cha, Sung-Hyuk. „Comprehensive survey on distance/similarity measures between probability density functions". *City* 1.2 (2007), S. 1.

[456] Székely, Gábor J; Rizzo, Maria L und Bakirov, Nail K. „Measuring and testing dependence by correlation of distances" (2007).

[457] Szegedy, Christian; Liu, Wei; Jia, Yangqing; Sermanet, Pierre; Reed, Scott; Anguelov, Dragomir; Erhan, Dumitru; Vanhoucke, Vincent und Rabinovich, Andrew. „Going deeper with convolutions". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, S. 1–9.

[458] Iandola, Forrest N; Han, Song; Moskewicz, Matthew W; Ashraf, Khalid; Dally, William J und Keutzer, Kurt. „SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size". *arXiv preprint arXiv:1602.07360* (2016).

[459] Zhang, Xiangyu; Zhou, Xinyu; Lin, Mengxiao und Sun, Jian. „Shufflenet: An extremely efficient convolutional neural network for mobile devices". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 6848–6856.

[460] Redmon, Joseph und Farhadi, Ali. „Yolov3: An incremental improvement". *arXiv preprint arXiv:1804.02767* (2018).

[461] Zoph, Barret; Vasudevan, Vijay; Shlens, Jonathon und Le, Quoc V. „Learning transferable architectures for scalable image recognition". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 8697–8710.

[462] Prasath, VB; Alfeilat, Haneen Arafat Abu; Hassanat, Ahmad; Lasassmeh, Omar; Tarawneh, Ahmad S; Alhasanat, Mahmoud Bashir und Salman, Hamzeh S Eyal. „Distance and similarity measures effect on the performance of K-Nearest Neighbor classifier–A review". *arXiv preprint arXiv:1708.04321* (2017).

[463] Wang, Yasi; Yao, Hongxun und Zhao, Sicheng. „Auto-encoder based dimensionality reduction". *Neurocomputing* 184 (2016), S. 232–242.

[464] Hinton, Geoffrey E und Roweis, Sam. „Stochastic neighbor embedding". *Advances in neural information processing systems* 15 (2002).

[465] Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz und Polosukhin, Illia. „Attention is all you need". *Advances in neural information processing systems* 30 (2017).

[466] Raghu, Maithra; Unterthiner, Thomas; Kornblith, Simon; Zhang, Chiyuan und Dosovitskiy, Alexey. „Do vision transformers see like convolutional neural networks?" *Advances in neural information processing systems* 34 (2021), S. 12116–12128.

[467] Paul, Sayak und Chen, Pin-Yu. „Vision transformers are robust learners". *Proceedings of the AAAI conference on Artificial Intelligence*. Bd. 36. 2. 2022, S. 2071–2081.

[468] Kaya, Mahmut und Bilge, Hasan Şakir. „Deep metric learning: A survey". *Symmetry* 11.9 (2019), S. 1066.

[469] Kalogerakis, Evangelos; Hertzmann, Aaron und Singh, Karan. „Learning 3D mesh segmentation and labeling". *ACM SIGGRAPH 2010 papers*. 2010, S. 1–12.

[470] Hetyei, Csaba und Szlivka, Ferenc. „Counter-rotating dual rotor wind turbine layout optimisation" (2021).

[471] Jenkins, Peter E; Younis, Abdalfadel und Chen, Yuxuan. „Design and Analysis of a Dual Rotor Turbine with a Shroud Using Flow Simulations". *Journal of Power and Energy Engineering* 5.4 (2017), S. 25–40.

[472] Nguyenc, Tuan C. *Is this odd-looking wind turbine the most efficient you can buy?* 2014.

[473] Sasaki, Noriyuki; Murakami, Mitsunori; Nozawa, Kazuo; Soejima, Shunji; Shiraki, Akira; Aono, Takeshi; Fujimoto, Tomeo; Funeno, Isao; Ishii, Norio und Onogi, Hiroshi. „Design system for optimum contra-rotating propellers". *Journal of marine science and technology* 3 (1998), S. 3–21.

[474] Strack, W.C.; Knip, G.; Weisbrich, A.L.; Godston, J. und Bradley, E. *Technology and benefits of aircraft counter rotation propellers*. Techn. Ber. 1981.

[475] Chang, Yupeng; Wang, Xu; Wang, Jindong; Wu, Yuan; Yang, Linyi; Zhu, Kaijie; Chen, Hao; Yi, Xiaoyuan; Wang, Cunxiang; Wang, Yidong u. a. „A survey on evaluation of large language models". *ACM Transactions on Intelligent Systems and Technology* 15.3 (2024), S. 1–45.

[476] Kaddour, Jean; Harris, Joshua; Mozes, Maximilian; Bradley, Herbie; Raileanu, Roberta und McHardy, Robert. „Challenges and applications of large language models". *arXiv preprint arXiv:2307.10169* (2023).

[477] Tian, Yu; Liu, Ang; Dai, Yun; Nagato, Keisuke und Nakao, Masayuki. „Systematic synthesis of design prompts for large language models in conceptual design". *CIRP Annals* (2024).

[478] Di, Wei; Bhardwaj, Anurag und Wei, Jianing. *Deep Learning Essentials: Your Hands-on Guide to the Fundamentals of Deep Learning and Neural Network Modeling*. Packt Publishing, 2018.

[479] Bi, Ying; Xue, Bing und Zhang, Mengjie. „Computer Vision and Machine Learning". *Genetic Programming for Image Classification: An Automated Approach to Feature Learning*. Cham: Springer International Publishing, 2021, S. 11–48. https://doi.org/10.1007/978-3-030-65927-1_2.

[480] Scherer, Dominik; Müller, Andreas und Behnke, Sven. „Evaluation of pooling operations in convolutional architectures for object recognition". *International conference on artificial neural networks*. Springer. 2010, S. 92–101.

[481] Tan, Pang-Ning; Steinbach, Michael und Kumar, Vipin. *Introduction to data mining*. Pearson internat. ed., [Nachdr.] Pearson international edition. Boston und Munich: Pearson Addison-Wesley, 2010.

[482] Bramer, Max. *Principles of Data Mining*. 3rd ed. 2016. Undergraduate Topics in Computer Science. London: Springer London und Imprint: Springer, 2016.

[483] Bradley, Andrew P. „The use of the area under the ROC curve in the evaluation of machine learning algorithms". *Pattern Recognition* 30.7 (1997), S. 1145–1159.

[484] Wong, Andrew K. C.; Lu, Si W und Rioux, Marc. „Recognition and shape synthesis of 3-D objects based on attributed hypergraphs". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11.3 (1989), S. 279–290.

[485] Frosini, Patrizio. „Measuring shapes by size functions". *Intelligent Robots and Computer Vision X: Algorithms and Techniques*. Bd. 1607. SPIE. 1992, S. 122–133.

[486] Shum, Heung-Yeung; Hebert, Martial und Ikeuchi, Katsushi. „On 3D shape similarity". *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE. 1996, S. 526–531.

[487] Suzuki, Motofumi T; Kato, Toshikazu und Tsukune, Hideo. „3D object retrieval based on subjective measures". *Proceedings Ninth International Workshop on Database and Expert Systems Applications (Cat. No. 98EX130)*. IEEE. 1998, S. 850–855.

[488] Johnson, Andrew E und Hebert, Martial. „Using spin images for efficient object recognition in cluttered 3D scenes". *IEEE Transactions on pattern analysis and machine intelligence* 21.5 (1999), S. 433–449.

[489] Edelsbrunner, Herbert; Letscher, David und Zomorodian, Afra. „Topological persistence and simplification". *Proceedings 41st annual symposium on foundations of computer science*. IEEE. 2000, S. 454–463.

[490] Paquet, Eric; Rioux, Marc; Murching, Anil; Naveen, Thumpudi und Tabatabai, Ali. „Description of shape information for 2-D and 3-D objects". *Signal processing: Image communication* 16.1-2 (2000), S. 103–122.

[491] Paquet, Eric und Rioux, Marc. „Nefertiti: A tool for 3-D shape databases management". *SAE transactions* (1999), S. 387–393.

[492] Vranic, Dejan V.; Saupe, Dietmar und Richter, Jörg. „Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics". *2001 IEEE Fourth Workshop on Multimedia Signal Processing (Cat. No. 01TH8564)*. IEEE. 2001, S. 293–298.

[493] Zaharia, Titus und Preteux, Francoise J. „3D-shape-based retrieval within the MPEG-7 framework". *Nonlinear image processing and pattern analysis xii*. Bd. 4304. SPIE. 2001, S. 133–145.

[494] Novotni, Marcin und Klein, Reinhard. „A geometric approach to 3D object comparison". *Proceedings International Conference on Shape Modeling and Applications*. IEEE. 2001, S. 167–175.

[495] Osada, Robert; Funkhouser, Thomas; Chazelle, Bernard und Dobkin, David. „Shape distributions". en. *ACM Trans. Graph.* 21.4 (Okt. 2002), S. 807–832.

[496] Zaharia, Titus und Prêteux, Françoise. „Shape-based retrieval of 3D mesh models". *Proceedings. IEEE International Conference on Multimedia and Expo*. Bd. 1. IEEE. 2002, S. 437–440.

[497] Elad, Michael; Tal, Ayellet und Ar, Sigal. „Content based retrieval of VRML objects—an iterative and interactive approach". *Multimedia 2001: Proceedings of the Eurographics Workshop in Manchester, United Kingdom, September 8–9, 2001*. Springer. 2002, S. 107–118.

[498] Mahmoudi, Saïd und Daoudi, Mohamed. „3D models retrieval by using characteristic views". *2002 International Conference on Pattern Recognition*. Bd. 2. IEEE. 2002, S. 457–460.

[499] Kazhdan, Michael; Funkhouser, Thomas und Rusinkiewicz, Szymon. „Rotation invariant spherical harmonic representation of 3 d shape descriptors". *Symposium on geometry processing*. Bd. 6. 2003, S. 156–164.

[500] Körtgen, Marcel; Park, Gil-Joo; Novotni, Marcin und Klein, Reinhard. „3D shape matching with 3D shape contexts". *The 7th central European seminar on computer graphics*. Bd. 3. Budmerice Slovakia. 2003, S. 5–17.

[501] Leibe, Bastian und Schiele, Bernt. „Analyzing appearance and contour based methods for object categorization". *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.* Bd. 2. IEEE. 2003, S. II–409.

[502] Vranic, Dejan V. „An improvement of rotation invariant 3D-shape based on functions on concentric spheres“. *Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429)*. Bd. 3. IEEE. 2003, S. III–757.

[503] Tangelder, Johan WH und Veltkamp, Remco C. „Polyhedral model retrieval using weighted point sets“. *International journal of image and graphics* 3.01 (2003), S. 209–229.

[504] Funkhouser, Thomas; Min, Patrick; Kazhdan, Michael; Chen, Joyce; Halderman, Alex; Dobkin, David und Jacobs, David. „A search engine for 3D models“. *ACM Trans. Graph.* 22.1 (Jan. 2003), S. 83–105.

[505] Ohbuchi, Ryutarou; Nakazawa, Masatoshi und Takei, Tsuyoshi. „Retrieving 3D shapes based on their appearance“. *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*. 2003, S. 39–45.

[506] Bespalov, Dmitriy; Regli, William C und Shokoufandeh, Ali. „Reeb graph based shape retrieval for CAD“. *International design engineering technical conferences and computers and information in engineering conference*. Bd. 36991. 2003, S. 229–238.

[507] Frome, Andrea; Huber, Daniel; Kolluri, Ravi; Bülow, Thomas und Malik, Jitendra. „Recognizing objects in range data using regional point descriptors“. *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part III 8*. Springer. 2004, S. 224–237.

[508] Bustos, Benjamin; Keim, Daniel A; Saupe, Dietmar; Schreck, Tobias und Vranic, Dejan V. „Using entropy impurity for improved 3D object similarity search“. *2004 IEEE International Conference on Multimedia and Expo (ICME)(IEEE Cat. No. 04TH8763)*. Bd. 2. IEEE. 2004, S. 1303–1306.

[509] Iyer, Natraj; Jayanti, Subramaniam; Lou, Kuiyang; Kalyanaraman, Yagnanarayanan und Ramani, Karthik. „A multi-scale hierarchical 3D shape representation for similar shape retrieval“. *Proceedings of the TMCE*. Bd. 2004. 2004.

[510] Ohbuchi, Ryutarou; Minamitani, Takahiro und Takei, Tsuyoshi. „Shape-similarity search of 3D models by using enhanced shape functions“. *International Journal of Computer Applications in Technology* 23.2-4 (2005), S. 70–85.

[511] Vranic, Dejan V. „Desire: A Composite 3D-Shape Descriptor“. *2005 IEEE International Conference on Multimedia and Expo*. IEEE, 2005, S. 962–965.

[512] Huang, Kohsia S. und Trivedi, Mohan M. „3D shape context based gesture analysis integrated with tracking using omni video array". *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*. IEEE. 2005, S. 80–80.

[513] Leifman, George; Meir, Ron und Tal, Ayellet. „Semantic-oriented 3d shape retrieval using relevance feedback". *The Visual Computer* 21 (2005), S. 865–875.

[514] Hou, Suyu; Lou, Kuiyang und Ramani, Karthik. „SVM-based semantic clustering and retrieval of a 3D model database". *Computer-Aided Design and Applications* 2.1-4 (2005), S. 155–164.

[515] Bespalov, Dmitriy; Regli, William C und Shokoufandeh, Ali. „Local feature extraction and matching partial objects". *Computer-Aided Design* 38.9 (2006), S. 1020–1037.

[516] Chaouch, Mohamed und Verroust-Blondet, Anne. „Enhanced 2D/3D approaches based on relevance index for 3D-shape retrieval". *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*. IEEE. 2006, S. 36–36.

[517] Laga, Hamid; Takahashi, Hiroki und Nakajima, Masayuki. „Spherical wavelet descriptors for content-based 3D model retrieval". *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*. IEEE. 2006, S. 15–15.

[518] Passalis, Geogios; Theoharis, Theoharis und Kakadiaris, Ioannis A. „Ptk: A novel depth buffer-based shape descriptor for three-dimensional object retrieval". *The Visual Computer* 23 (2007), S. 5–14.

[519] Shih, Jau-Ling; Lee, Chang-Hsing und Wang, Jian Tang. „A new 3D model retrieval approach based on the elevation descriptor". *Pattern Recognition* 40.1 (2007), S. 283–295.

[520] Kuo, Chen-Tsung und Cheng, Shyi-Chyi. „3D model retrieval using principal plane analysis and dynamic programming". *Pattern Recognition* 40.2 (2007), S. 742–755.

[521] Hou, Xin; Zhang, XuTang und Liu, WenJian. „Using enhanced shape distributions to compare CAD models". *Advances in Multimedia Information Processing–PCM 2007: 8th Pacific Rim Conference on Multimedia, Hong Kong, China, December 11-14, 2007. Proceedings 8*. Springer. 2007, S. 385–388.

[522] Rustamov, Raif M u. a. „Laplace-Beltrami eigenfunctions for deformation invariant shape representation". *Symposium on geometry processing*. Bd. 257. 2007, S. 225–233.

[523] Jain, Varun und Zhang, Hao. „A spectral approach to shape-based retrieval of articulated 3D models“. *Computer-Aided Design* 39.5 (2007), S. 398–407.

[524] Chaouch, Mohamed und Verroust-Blondet, Anne. „A new descriptor for 2D depth image indexing and 3D model retrieval“. *2007 IEEE international conference on image processing*. Bd. 6. IEEE. 2007, S. VI–373.

[525] Xu, Dong und Li, Hua. „3D shape retrieval integrated with classification information“. *Fourth International Conference on Image and Graphics (ICIG 2007)*. IEEE. 2007, S. 774–779.

[526] Ohbuchi, Ryutarou; Yamamoto, Akihiro und Kobayashi, Jun. „Learning semantic categories for 3D model retrieval“. *Proceedings of the international workshop on Workshop on multimedia information retrieval*. 2007, S. 31–40.

[527] Laga, Hamid und Nakajima, Masayuki. „A boosting approach to content-based 3D model retrieval“. *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*. 2007, S. 227–234.

[528] Biasotti, Silvia; Giorgi, Daniela; Marini, Simone; Spagnuolo, Michela und Falcidieno, Bianca. „3D classification via structural prototypes“. *Semantic Multimedia: Second International Conference on Semantic and Digital Media Technologies, SAMT 2007, Genoa, Italy, December 5-7, 2007. Proceedings 2*. Springer. 2007, S. 140–143.

[529] Ansary, Tarik Filali; Daoudi, Mohamed und Vandeborre, Jean-Philippe. „A bayesian 3-d search engine using adaptive views clustering“. *IEEE Transactions on Multimedia* 9.1 (2006), S. 78–88.

[530] Papadakis, Panagiotis; Pratikakis, Ioannis; Perantonis, Stavros und Theoharis, Theoharis. „Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation“. *Pattern Recognition* 40.9 (2007), S. 2437–2452.

[531] Tal, Ayellet und Zuckerberger, Emanuel. „Mesh retrieval by components“. *Advances in Computer Graphics and Computer Vision: International Conferences VISAPP and GRAPP 2006, Setúbal, Portugal, February 25-28, 2006, Revised Selected Papers*. Springer. 2007, S. 44–57.

[532] Siddiqi, Kaleem; Zhang, Juan; Macrini, Diego; Shokoufandeh, Ali; Bouix, Sylvain und Dickinson, Sven. „Retrieving articulated 3-D models using medial surfaces“. *Machine vision and applications* 19 (2008), S. 261–275.

[533] Papadakis, Panagiotis; Pratikakis, Ioannis; Theoharis, Theoharis; Passalis, Georgios und Perantonis, Stavros. „3D Object Retrieval using an Efficient and Compact Hybrid Shape Descriptor". *Eurographics Workshop on 3D object retrieval*. 2008.

[534] Ben-Chen, Mirela und Gotsman, Craig. „Characterizing shape using conformal factors." *3DOR@ Eurographics*. Citeseer. 2008, S. 1–8.

[535] Ohbuchi, Ryutarou; Osada, Kunio; Furuya, Takahiko und Banno, Tomohisa. „Salient local visual features for shape-based 3D model retrieval". *2008 IEEE International Conference on Shape Modeling and Applications*. IEEE. 2008, S. 93–102.

[536] Laga, Hamid und Nakajima, Masayuki. „Supervised learning of salient 2D views of 3D models". *The Journal of the Society for Art and Science* 7.4 (2008), S. 124–131.

[537] Laga, Hamid und Nakajima, Masayuki. „Supervised learning of similarity measures for content-based 3D model retrieval". *Large-Scale Knowledge Resources. Construction and Application: Third International Conference on Large-Scale Knowledge Resources, LKR 2008, Tokyo, Japan, March 3-5, 2008. Proceedings*. Springer. 2008, S. 210–225.

[538] Yamamoto, Akihiro; Tezuka, Masaki; Shimizu, Toshiya und Ohbuchi, Ryutarou. „SHREC'08 entry: Semi-supervised learning for semantic 3d model retrieval". *2008 IEEE International Conference on Shape Modeling and Applications*. IEEE. 2008, S. 241–243.

[539] Leng, Biao und Qin, Zheng. „A powerful relevance feedback mechanism for content-based 3D model retrieval". *Multimedia Tools and Applications* 40 (2008), S. 135–150.

[540] Ohbuchi, Ryutarou und Furuya, Takahiko. „Accelerating bag-of-features sift algorithm for 3d model retrieval". *Proc. SAMT 2008 Workshop on Semantic 3D Media (S-3D)*. 2008, S. 23–30.

[541] Ohbuchi, Ryutarou und Shimizu, Toshiya. „Ranking on semantic manifold for shape-based 3d model retrieval". *Proceedings of the 1st ACM international conference on Multimedia information retrieval*. 2008, S. 411–418.

[542] Osada, Kunio; Furuya, Takahiko und Ohbuchi, Ryutarou. „SHREC—08 entry: Local 2D visual features for CAD Model retrieval". *2008 IEEE International Conference on Shape Modeling and Applications*. IEEE. 2008, S. 237–238.

[543] Papadakis, Panagiotis; Pratikakis, Ioannis; Trafalis, Theodore; Theoharis, Theoharis und Perantonis, Stavros. „Relevance feedback in content-based 3D object retrieval a comparative study". *Computer-Aided Design and Applications* 5.5 (2008), S. 753–763.

[544] Akgül, Ceyhun Burak; Sankur, Bülent; Yemez, Yücel und Schmitt, Francis JM. „Similarity Score Fusion by Ranking Risk Minimization for 3D Object Retrieval." *3DOR@ Eurographics*. 2008, S. 41–48.

[545] Li, Xiaolan; Godil, Afzal und Wagan, Asim. „Spatially enhanced bags of words for 3D shape retrieval". *Advances in Visual Computing: 4th International Symposium, ISVC 2008, Las Vegas, NV, USA, December 1-3, 2008. Proceedings, Part I 4*. Springer. 2008, S. 349–358.

[546] Toldo, Roberto; Castellani, Umberto und Fusiello, Andrea. „Visual vocabulary signature for 3D object retrieval and partial matching". *Proceedings of the 2nd Eurographics conference on 3D Object Retrieval*. 2009, S. 21–28.

[547] Tatsuma, Atsushi und Aono, Masaki. „Multi-Fourier spectra descriptor and augmentation with spectral clustering for 3D shape retrieval". *The Visual Computer* 25.8 (2009), S. 785–804.

[548] Jing, Hui; Huang, Mei Fa und Li, Cong. „3D Mechanical Models Retrieval Based on Combined Histograms for Rapid Product Design". *Applied Mechanics and Materials*. Bd. 16. Trans Tech Publ. 2009, S. 65–69.

[549] Furuya, Takahiko und Ohbuchi, Ryutarou. „Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features". *Proceedings of the ACM international conference on image and video retrieval*. 2009, S. 1–8.

[550] Akgül, Ceyhun Burak; Sankur, Bülent; Yemez, Yücel und Schmitt, Francis. „3D model retrieval using probability density-based shape descriptors". *IEEE transactions on pattern analysis and machine intelligence* 31.6 (2009), S. 1117–1133.

[551] Sun, Jian; Ovsjanikov, Maks und Guibas, Leonidas. „A concise and provably informative multi-scale signature based on heat diffusion". *Computer graphics forum*. Bd. 28. 5. Wiley Online Library. 2009, S. 1383–1392.

[552] Carlsson, Gunnar. „Topology and data". *Bulletin of the American Mathematical Society* 46.2 (2009), S. 255–308.

[553] Mademlis, Athanasios; Daras, Petros; Tzovaras, Dimitrios und Strintzis, Michael G. „3D object retrieval using the 3D shape impact descriptor". *Pattern Recognition* 42.11 (2009), S. 2447–2459.

[554] Li, Fei; Dai, Qionghai; Xu, Wenli und Er, Guihua. „Weighted subspace distance and its applications to object recognition and retrieval with image sets". *IEEE Signal Processing Letters* 16.3 (2009), S. 227–230.

[555] Ohbuchi, Ryutarou und Furuya, Takahiko. „Scale-weighted dense bag of visual features for 3D model retrieval from a partial view 3D model". *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE. 2009, S. 63–70.

[556] Zhu, KP; Wong, YS; Lu, Wen Feng und Fuh, Jerry YH. „A diffusion wavelet approach for 3-D model matching". *Computer-Aided Design* 41.1 (2009), S. 28–36.

[557] Lian, Zhouhui; Godil, Afzal und Sun, Xianfang. „Visual similarity based 3D shape retrieval using bag-of-features". *2010 Shape modeling international conference*. IEEE. 2010, S. 25–36.

[558] Jégou, Hervé; Douze, Matthijs; Schmid, Cordelia und Pérez, Patrick. „Aggregating local descriptors into a compact image representation". *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE. 2010, S. 3304–3311.

[559] Laga, Hamid. „Semantics-driven approach for automatic selection of best views of 3D shapes". *3DOR@ Eurographics*. 2010, S. 15–22.

[560] Pratikakis, Ioannis; Spagnuolo, Michela; Theoharis, Theoharis und Veltkamp, Remco. „Learning the compositional structure of man-made objects for 3d shape retrieval". *Eurographics Workshop on 3D Object Retrieval*. 2010.

[561] Ohbuchi, Ryutarou; Tezuka, Masaki; Furuya, Takahiko und Oyobe, Takashi. „Squeezing bag-of-features for scalable and semantic 3D model retrieval". *2010 International Workshop on Content Based Multimedia Indexing (CBMI)*. IEEE. 2010, S. 1–6.

[562] Makadia, Ameesh und Daniilidis, Kostas. „Spherical correlation of visual representations for 3D model retrieval". *International Journal of Computer Vision* 89 (2010), S. 193–210.

[563] Gao, Yue; Yang, You; Dai, Qionghai und Zhang, Naiyao. „Representative views re-ranking for 3d model retrieval with multi-bipartite graph reinforcement model". *Proceedings of the 18th ACM international conference on Multimedia*. 2010, S. 947–950.

[564] Gao, Yue; Dai, Qionghai und Zhang, Nai-Yao. „3D model comparison using spatial structure circular descriptor". *Pattern Recognition* 43.3 (2010), S. 1142–1151.

[565] Ohbuchi, Ryutarou und Furuya, Takahiko. „Distance metric learning and feature combination for shape-based 3D model retrieval". *Proceedings of the ACM workshop on 3D object retrieval*. 2010, S. 63–68.

[566] Ruggeri, Mauro R; Patanè, Giuseppe; Spagnuolo, Michela und Saupe, Dietmar. „Spectral-driven isometry-invariant matching of 3D shapes". *International Journal of Computer Vision* 89 (2010), S. 248–265.

[567] Sfikas, Konstantinos; Theoharis, Theoharis und Pratikakis, Ioannis. „ROSy+: 3D object pose normalization based on PCA and reflective object symmetry with application in 3D object retrieval". *International Journal of Computer Vision* 91.3 (2011), S. 262–279.

[568] Aubry, Mathieu; Schlickewei, Ulrich und Cremers, Daniel. „The wave kernel signature: A quantum mechanical approach to shape analysis". *2011 IEEE international conference on computer vision workshops (ICCV workshops)*. IEEE. 2011, S. 1626–1633.

[569] Pan, Xiang; You, Qian; Liu, Zhi und Chen, Qi Hua. „3D shape retrieval by Poisson histogram". *Pattern Recognition Letters* 32.6 (2011), S. 787–794.

[570] Li, Bo und Johan, Henry. „View context based 2D sketch-3D model alignment". *2011 IEEE workshop on applications of computer vision (WACV)*. IEEE. 2011, S. 45–50.

[571] Axenopoulos, Apostolos; Litos, Georgios und Daras, Petros. „3D model retrieval using accurate pose estimation and view-based similarity". *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*. 2011, S. 1–8.

[572] Chen, Langshi; Leng, Biao; Xiong, Zhang und Chen, Chen. „A new framework for composing vectorial semantic labels in 3D model retrieval". *2011 International Conference on Photonics, 3D-Imaging, and Visualization*. Bd. 8205. SPIE. 2011, S. 51–56.

[573] Gao, Yue; Wang, Meng; Zha, Zheng-Jun; Tian, Qi; Dai, Qionghai und Zhang, Naiyao. „Less is more: efficient 3-D object retrieval with query view selection". *IEEE Transactions on Multimedia* 13.5 (2011), S. 1007–1018.

[574] Bronstein, Alexander M; Bronstein, Michael M; Guibas, Leonidas J und Ovsjanikov, Maks. „Shape google: Geometric words and expressions for invariant shape retrieval". *ACM Transactions on Graphics (TOG)* 30.1 (2011), S. 1–20.

[575] Gao, Yue; Wang, Meng; Ji, Rongrong; Zha, Zhengjun und Shen, Jialie. „k-Partite graph reinforcement and its application in multimedia information retrieval". *Information Sciences* 194 (2012), S. 224–239.

[576] Wen, Yue; Gao, Yue; Hong, Richang; Luan, Huanbo; Liu, Qiong; Shen, Jialie und Ji, Rongrong. „View-based 3D object retrieval by bipartite graph matching". *Proceedings of the 20th ACM international conference on Multimedia*. 2012, S. 897–900.

[577] Gao, Yue; Tang, Jinhui; Hong, Richang; Yan, Shuicheng; Dai, Qionghai; Zhang, Naiyao und Chua, Tat-Seng. „Camera constraint-free view-based 3-D object retrieval". *IEEE Transactions on Image Processing* 21.4 (2011), S. 2269–2281.

[578] Gao, Yue; Wang, Meng; Tao, Dacheng; Ji, Rongrong und Dai, Qionghai. „3-D object retrieval and recognition with hypergraph analysis". *IEEE transactions on image processing* 21.9 (2012), S. 4290–4303.

[579] Endoh, Megumi; Yanagimachi, Tomohiro und Ohbuchi, Ryutarou. „Efficient manifold learning for 3D model retrieval by using clustering-based training sample reduction". *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2012, S. 2345–2348.

[580] Kawamura, Shun; Usui, Kazuya; Furuya, Takahiko und Ohbuchi, Ryutarou. „Local Goemetrical Feature with Spatial Context for Shape-based 3D Model Retrieval." *3DOR@ Eurographics*. 2012, S. 55–58.

[581] Bustos, Benjamin; Schreck, Tobias; Walter, Michael; Barrios, Juan Manuel; Schaefer, Matthias und Keim, Daniel. „Improving 3D similarity search by enhancing and combining 3D descriptors". *Multimedia Tools and Applications* 58 (2012), S. 81–108.

[582] Kim, Vladimir G; Li, Wilmot; Mitra, Niloy J; DiVerdi, Stephen und Funkhouser, Thomas. „Exploring collections of 3d models using fuzzy correspondences". *ACM Transactions on Graphics (TOG)* 31.4 (2012), S. 1–11.

[583] Darom, Tal und Keller, Yosi. „Scale-invariant features for 3-D mesh models". *IEEE Transactions on Image Processing* 21.5 (2012), S. 2758–2769.

[584] Tam, Gary KL und Lau, Rynson WH. „Embedding retrieval of articulated geometry models". *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), S. 2134–2146.

[585] Zhu, Kunpeng; San Wong, Yoke; Loh, Han Tong und Lu, Wen Feng. „3D CAD model retrieval with perturbed Laplacian spectra". *Computers in industry* 63.1 (2012), S. 1–11.

[586] Daras, Petros; Axenopoulos, Apostolos und Litos, Georgios. „Investigating the effects of multiple factors towards more accurate 3-D object retrieval". *IEEE Transactions on multimedia* 14.2 (2011), S. 374–388.

[587] Li, Bo und Johan, Henry. „3D model retrieval using hybrid features and class information". *Multimedia tools and applications* 62.3 (2013), S. 821–846.

[588] Chowdhury, Nihad Karim; Tatsuma, Atsushi und Aono, Masaki. „A New Shape Descriptor based on Local and Global Feature for 3D Shape Retrieval". *ITE Technical Report 37.56*. The Institute of Image Information und Television Engineers. 2013, S. 19–22.

[589] Bae, Min Soo und Park, In Kyu. „Content-based 3D model retrieval using a single depth image from a low-cost 3D camera". *The Visual Computer* 29 (2013), S. 555–564.

[590] Aono, Masaki; Koyanagi, Hitoshi und Tatsuma, Atsushi. „3D shape retrieval focused on holes and surface roughness". *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*. Kaohsiung, Taiwan: IEEE, Okt. 2013.

[591] Sipiran, Ivan; Bustos, Benjamin und Schreck, Tobias. „Data-aware 3D partitioning for generic shape retrieval". *Computers & graphics* 37.5 (2013), S. 460–472.

[592] Li, Chunyuan und Hamza, A Ben. „Symmetry discovery and retrieval of nonrigid 3D shapes using geodesic skeleton paths". *Multimedia tools and applications* 72 (2014), S. 1027–1047.

[593] Barra, Vincent und Biasotti, Silvia. „3D shape retrieval using kernels on extended Reeb graphs". *Pattern Recognition* 46.11 (2013), S. 2985–2999.

[594] Biasotti, Silvia; Cerri, Andrea; Giorgi, Daniela und Spagnuolo, Michela. „Phog: Photometric and geometric functions for textured shape retrieval". *Computer Graphics Forum*. Bd. 32. 5. Wiley Online Library. 2013, S. 13–22.

[595] Bonaventura, Xavier; Guo, Jianwei; Meng, Weiliang; Feixas, Miquel; Zhang, Xiaopeng und Sbert, Mateu. „3D shape retrieval using viewpoint information-theoretic measures". *Computer Animation and Virtual Worlds* 26.2 (2015), S. 147–156.

[596] Sfikas, Konstantinos; Theoharis, Theoharis und Pratikakis, Ioannis. „3D object retrieval via range image queries in a bag-of-visual-words context". *The Visual Computer* 29 (2013), S. 1351–1361.

[597] Li, Pengjie; Ma, Huadong und Ming, Anlong. „Combining topological and view-based features for 3D model retrieval". *Multimedia tools and applications* 65 (2013), S. 335–361.

[598] Lian, Zhouhui; Godil, Afzal; Sun, Xianfang und Xiao, Jianguo. „CM-BOF: visual similarity-based 3d shape retrieval using clock matching and bag-of-features". *Machine Vision and Applications* 24 (2013), S. 1685–1704.

[599] Zou, Kuan-Sheng; Ip, Wai-Hung; Wu, Chun-Ho; Chen, Zeng-Qiang; Yung, Kai-Leung und Chan, Ching-Yuen. „A novel 3D model retrieval approach using combined shape distribution". en. *Multimed. Tools Appl.* 69.3 (Apr. 2014), S. 799–818.

[600] Zhu, Zhuotun; Wang, Xinggang; Bai, Song; Yao, Cong und Bai, Xiang. „Deep learning representation using autoencoder for 3D shape retrieval". *Neurocomputing* 204 (2016), S. 41–50.

[601] Sfikas, Konstantinos; Theoharis, Theoharis und Pratikakis, Ioannis. „Pose normalization of 3D models via reflective symmetry on panoramic views". *The Visual Computer* 30 (2014), S. 1261–1274.

[602] Furuya, Takahiko und Ohbuchi, Ryutarou. „Fusing Multiple Features for Shape-based 3D Model Retrieval." *BMVC*. Citeseer. 2014.

[603] Bai, Xiang; Rao, Cong und Wang, Xinggang. „Shape vocabulary: a robust and efficient shape representation for shape matching". en. *IEEE Trans. Image Process.* 23.9 (Sep. 2014), S. 3935–3949.

[604] Litman, Roee; Bronstein, Alex; Bronstein, Michael und Castellani, Umberto. „Supervised learning of bag-of-features shape descriptors using sparse coding". *Computer Graphics Forum*. Bd. 33. 5. Wiley Online Library. 2014, S. 127–136.

[605] Li, Chunyuan; Ovsjanikov, Maks und Chazal, Frederic. „Persistence-based structural recognition". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, S. 1995–2002.

[606] Ding, Ke und Liu, Yun-Hui. „Sphere image for 3-D model retrieval". *IEEE transactions on multimedia* 16.5 (2014), S. 1369–1376.

[607] Liu, Zhenbao; Bu, Shuhui; Han, Junwei und Wu, Jun. „Sparse patch coding for 3D model retrieval". *MultiMedia Modeling: 20th Anniversary International Conference, MMM 2014, Dublin, Ireland, January 6-10, 2014, Proceedings, Part II 20*. Springer. 2014, S. 116–127.

[608] Tabia, Hedi; Laga, Hamid; Picard, David und Gosselin, Philippe-Henri. „Covariance descriptors for 3D shape matching and retrieval". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, S. 4185–4192.

[609] Mehrdad, Vahid und Ebrahimnezhad, Hossein. „3D object retrieval based on histogram of local orientation using one-shot score support vector machine". en. *Front. Comput. Sci.* 9.6 (Dez. 2015), S. 990–1005.

[610] Chen, Qiang; Fang, Bin; Yu, Yong-Mei und Tang, Yan. „3D CAD model retrieval based on the combination of features". *Multimedia Tools and Applications* 74 (2015), S. 4907–4925.

[611] Bai, Xiang; Bai, Song; Zhu, Zhuotun und Latecki, Longin Jan. „3D shape matching via two layer coding". *IEEE transactions on pattern analysis and machine intelligence* 37.12 (2015), S. 2361–2373.

[612] Maturana, Daniel und Scherer, Sebastian. „Voxnet: A 3d convolutional neural network for real-time object recognition". *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE. 2015, S. 922–928.

[613] Fang, Yi; Xie, Jin; Dai, Guoxian; Wang, Meng; Zhu, Fan; Xu, Tiantian und Wong, Edward. „3d deep shape descriptor". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, S. 2319–2328.

[614] Masci, Jonathan; Boscaini, Davide; Bronstein, Michael und Vandergheynst, Pierre. „Geodesic convolutional neural networks on riemannian manifolds". *Proceedings of the IEEE international conference on computer vision workshops*. 2015, S. 37–45.

[615] Boscaini, Davide; Masci, Jonathan; Melzi, Simone; Bronstein, Michael M; Castellani, Umberto und Vandergheynst, Pierre. „Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks". *Computer graphics forum*. Bd. 34. 5. Wiley Online Library. 2015, S. 13–23.

[616] Shi, Baoguang; Bai, Song; Zhou, Zhichao und Bai, Xiang. „Deeppano: Deep panoramic representation for 3-d shape recognition". *IEEE Signal Processing Letters* 22.12 (2015), S. 2339–2343.

[617] Furuya, Takahiko und Ohbuchi, Ryutarou. „Diffusion-on-manifold aggregation of local features for shape-based 3d model retrieval". *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. 2015, S. 171–178.

[618] Leng, Biao; Liu, Yu; Yu, Kai; Zhang, Xiangyang und Xiong, Zhang. „3D object understanding with 3D convolutional neural networks". *Information sciences* 366 (2016), S. 188–201.

[619] Furuya, Takahiko und Ohbuchi, Ryutarou. „Accurate aggregation of local features by using K-sparse autoencoder for 3D model retrieval“. *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*. 2016, S. 293–297.

[620] Han, Zhizhong; Liu, Zhenbao; Han, Junwei; Vong, Chi-Man; Bu, Shuhui und Li, Xuelong. „Unsupervised 3D local feature learning by circle convolutional restricted Boltzmann machine“. *IEEE Transactions on Image Processing* 25.11 (2016), S. 5331–5344.

[621] Qi, Charles R; Su, Hao; Nießner, Matthias; Dai, Angela; Yan, Mengyuan und Guibas, Leonidas J. „Volumetric and multi-view cnns for object classification on 3d data“. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, S. 5648–5656.

[622] Garcia-Garcia, Alberto; Gomez-Donoso, Francisco; Garcia-Rodriguez, Jose; Orts-Escolano, Sergio; Cazorla, Miguel und Azorin-Lopez, J. „Pointnet: A 3d convolutional neural network for real-time object class recognition“. *2016 International joint conference on neural networks (IJCNN)*. IEEE. 2016, S. 1578–1584.

[623] Bai, Song; Bai, Xiang; Zhou, Zhichao; Zhang, Zhaoxiang und Jan Latecki, Longin. „Gift: A real-time and scalable 3d shape search engine“. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, S. 5023–5032.

[624] Boscaini, Davide; Masci, Jonathan; Rodolà, Emanuele; Bronstein, Michael M und Cremers, Daniel. „Anisotropic diffusion descriptors“. *Computer Graphics Forum*. Bd. 35. 2. Wiley Online Library. 2016, S. 431–441.

[625] Guo, Haiyun; Wang, Jinqiao; Gao, Yue; Li, Jianqiang und Lu, Hanqing. „Multi-view 3D object retrieval with deep embedding network“. *IEEE Transactions on Image Processing* 25.12 (2016), S. 5526–5537.

[626] Lin, Xin; Zhu, Kunpeng und Wang, Qingguo. „Anisotropic diffusion map based spectral embedding for 3D CAD model retrieval“. *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*. IEEE. 2016, S. 1078–1081.

[627] Zhuang, Ting; Zhang, Xutang; Hou, Zhenxiu; Zuo, Wangmeng und Liu, Yan. „A novel 3D CAD model retrieval method based on vertices classification and weights combination optimization“. *Mathematical Problems in Engineering* 2017 (2017).

[628] Shih, Jau-Ling; Lee, Chang-Hsing; Hou, Yao-Wen und Yeh, Po-Ting. „Three-dimensional model retrieval using dynamic multi-descriptor fusion". *Journal of Electronic Science and Technology* 15.2 (2017), S. 169–177.

[629] Getto, Roman; Kuijper, Arjan und Fellner, Dieter W. „Unsupervised 3D object retrieval with parameter-free hierarchical clustering". *Proceedings of the Computer Graphics International Conference*. 2017, S. 1–6.

[630] Sedaghat, Nima; Zolfaghari, Mohammadreza; Amiri, Ehsan und Brox, Thomas. „Orientation-boosted voxel nets for 3D object recognition". *arXiv preprint arXiv:1604.03351* (2016).

[631] Han, Zhizhong; Liu, Zhenbao; Han, Junwei; Vong, Chi-Man; Bu, Shuhui und Chen, Chun Lung Philip. „Mesh convolutional restricted Boltzmann machines for unsupervised learning of features with structure preservation on 3-D meshes". *IEEE transactions on neural networks and learning systems* 28.10 (2016), S. 2268–2281.

[632] Klokov, Roman und Lempitsky, Victor. „Escape from cells: Deep kd-networks for the recognition of 3d point cloud models". *Proceedings of the IEEE international conference on computer vision*. 2017, S. 863–872.

[633] Xie, Jin; Zhu, Fan; Dai, Guoxian; Shao, Ling und Fang, Yi. „Progressive shape-distribution-encoder for learning 3D shape representation". *IEEE Transactions on Image Processing* 26.3 (2017), S. 1231–1242.

[634] Xie, Jin; Dai, Guoxian; Zhu, Fan; Wong, Edward K und Fang, Yi. „Deepshape: Deep-learned shape descriptor for 3d shape retrieval". *IEEE transactions on pattern analysis and machine intelligence* 39.7 (2016), S. 1335–1345.

[635] Han, Zhizhong; Liu, Zhenbao; Han, Junwei; Vong, Chi-Man; Bu, Shuhui und Chen, CL Philip. „Unsupervised learning of 3-D local features from raw voxels based on a novel permutation voxelization strategy". *IEEE transactions on cybernetics* 49.2 (2017), S. 481–494.

[636] You, Haoxuan; Feng, Yifan; Ji, Rongrong und Gao, Yue. „Pvnet: A joint convolutional network of point cloud and multi-view for 3d shape recognition". *Proceedings of the 26th ACM international conference on Multimedia*. 2018, S. 1310–1318.

[637] Su, Yuting; Li, Wenhui; Liu, Anan und Nie, Weizhi. „Hierarchical Graph Structure Learning for Multi-View 3D Model Retrieval." *IJCAI*. 2018, S. 913–919.

[638] Ma, Yao; Ren, Zhaochun; Jiang, Ziheng; Tang, Jiliang und Yin, Dawei. „Multi-dimensional network embedding with hierarchical structure". *Proceedings of the eleventh ACM international conference on web search and data mining*. 2018, S. 387–395.

[639] Dominguez, Miguel; Dhamdhere, Rohan; Petkar, Atir; Jain, Saloni; Sah, Shagan und Ptucha, Raymond. „General-purpose deep point cloud feature extractor". *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2018, S. 1972–1981.

[640] Li, Jiaxin; Chen, Ben M und Lee, Gim Hee. „So-net: Self-organizing network for point cloud analysis". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 9397–9406.

[641] Yang, Yaoqing; Feng, Chen; Shen, Yiru und Tian, Dong. „Foldingnet: Point cloud auto-encoder via deep grid deformation". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 206–215.

[642] He, Xinwei; Zhou, Yang; Zhou, Zhichao; Bai, Song und Bai, Xiang. „Triplet-center loss for multi-view 3d object retrieval". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, S. 1945–1954.

[643] Lee, Tang; Lin, Yen-Liang; Chiang, HungYueh; Chiu, Ming-Wei; Hsu, Winston und Huang, Polly. „Cross-domain image-based 3d shape retrieval by view sequence learning". *2018 international conference on 3D vision (3DV)*. IEEE. 2018, S. 258–266.

[644] Xu, Cheng; Leng, Biao; Zhang, Cheng und Zhou, Xiaochen. „Emphasizing 3d properties in recurrent multi-view aggregation for 3d shape retrieval". *Proceedings of the AAAI Conference on Artificial Intelligence*. Bd. 32. 1. 2018.

[645] Lin, Kyaw Zaw; Xu, Weipeng; Sun, Qianru; Theobalt, Christian und Chua, Tat-Seng. „Learning a disentangled embedding for monocular 3d shape retrieval and pose estimation". *arXiv preprint arXiv:1812.09899* (2018).

[646] Li, Haisheng; Sun, Li; Wu, Xiaoqun und Cai, Qiang. „Scale-invariant wave kernel signature for non-rigid 3D shape retrieval". *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE. 2018, S. 448–454.

[647] Luciano, Lorenzo und Hamza, A Ben. „Geodesic-based 3D shape retrieval using sparse autoencoders". *Proceedings of the 11th Eurographics Workshop on 3D Object Retrieval*. 2018, S. 21–28.

[648] Benjelloun, Mohammed; Dadi, El Wardani und Daoudi, El Mostafa. „3D shape retrieval basing on representatives of classes". *arXiv preprint arXiv:1810.09008* (2018).

[649] Benjelloun, Mohammed; Dadi, El Wardani; Daoudi, El Mostafa und Larhmam, Mohammed Amine. „Content-based 3D shape retrieval using deep learning approach". *Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications*. 2018, S. 1–4.

[650] Shao, Ruiting; Lei, Yang; Fan, Jian und Liu, Jerry. „3D Shape Retrieval using Volumetric and Image CNNs: A Meta-Algorithmic Approach". *Electronic Imaging* 30 (2018), S. 1–6.

[651] Rucco, Matteo; Giannini, Franca; Lupinetti, Katia und Monti, Marina. „A methodology for part classification with supervised machine learning". *Ai Edam* 33.1 (2019), S. 100–113.

[652] Han, Zhizhong; Shang, Mingyang; Liu, Zhenbao; Vong, Chi-Man; Liu, Yu-Shen; Zwicker, Matthias; Han, Junwei und Chen, CL Philip. „SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention". *IEEE Transactions on Image Processing* 28.2 (2018), S. 658–672.

[653] Lin, Xin; Zhu, Kunpeng und Wang, Qing-Guo. „Three-dimensional CAD model matching with anisotropic diffusion maps". *IEEE Transactions on Industrial Informatics* 14.1 (2017), S. 265–274.

[654] Machalica, Dawid und Matyjewski, Marek. „CAD models clustering with machine learning". *Archive of mechanical engineering* 66.2 (2019).

[655] Feng, Yutong; Feng, Yifan; You, Haoxuan; Zhao, Xibin und Gao, Yue. „Meshnet: Mesh neural network for 3d shape representation". *Proceedings of the AAAI conference on artificial intelligence*. Bd. 33. 01. 2019, S. 8279–8286.

[656] Liu, Yongcheng; Fan, Bin; Xiang, Shiming und Pan, Chunhong. „Relation-shape convolutional neural network for point cloud analysis". *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, S. 8895–8904.

[657] Gao, Zan; Wang, DY; Wan, SH; Zhang, Hua und Wang, YL. „Cognitive-inspired class-statistic matching with triple-constrain for camera free 3D object retrieval". *Future Generation Computer Systems* 94 (2019), S. 641–653.

[658] Jiang, Jianwen; Bao, Di; Chen, Ziqiang; Zhao, Xibin und Gao, Yue. „MLVCNN: Multi-loop-view convolutional neural network for 3D shape retrieval". *Proceedings of the AAAI Conference on Artificial Intelligence*. Bd. 33. 01. 2019, S. 8513–8520.

[659] Ouyang, Yi; Guo, Bin; Tang, Xing; He, Xiuqiang; Xiong, Jian und Yu, Zhiwen. „Learning cross-domain representation with multi-graph neural network". *arXiv preprint arXiv:1905.10095* (2019).

[660] Hanocka, Rana; Hertz, Amir; Fish, Noa; Giryes, Raja; Fleishman, Shachar und Cohen-Or, Daniel. „Meshcnn: a network with an edge". *ACM Transactions on Graphics (ToG)* 38.4 (2019), S. 1–12.

[661] Liu, Zhijian; Tang, Haotian; Lin, Yujun und Han, Song. „Point-voxel cnn for efficient 3d deep learning". *Advances in neural information processing systems* 32 (2019).

[662] Cheraghian, Ali und Petersson, Lars. „3dcapsule: Extending the capsule architecture to classify 3d point clouds". *2019 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2019, S. 1194–1202.

[663] Bold, Naranchimeg; Zhang, Chao und Akashi, Takuya. „3D point cloud retrieval with bidirectional feature match". *IEEE Access* 7 (2019), S. 164194–164202.

[664] You, Haoxuan; Feng, Yifan; Zhao, Xibin; Zou, Changqing; Ji, Rongrong und Gao, Yue. „PVRNet: Point-view relation neural network for 3D shape recognition". *Proceedings of the AAAI conference on artificial intelligence*. Bd. 33. 01. 2019, S. 9119–9126.

[665] Han, Zhizhong; Lu, Honglei; Liu, Zhenbao; Vong, Chi-Man; Liu, Yu-Shen; Zwicker, Matthias; Han, Junwei und Chen, CL Philip. „3D2SeqViews: Aggregating sequential views for 3D global feature learning by CNN with hierarchical attention aggregation". *IEEE Transactions on Image Processing* 28.8 (2019), S. 3986–3999.

[666] He, Xinwei; Huang, Tengteng; Bai, Song und Bai, Xiang. „View n-gram network for 3d object retrieval". *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, S. 7515–7524.

[667] Xu, Cheng; Li, Zhaoqun; Qiu, Qiang; Leng, Biao und Jiang, Jingfei. „Enhancing 2D representation via adjacent views for 3D shape retrieval". *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, S. 3732–3740.

[668] Zhou, Wen; Jia, Jinyuan; Huang, Chengxi und Cheng, Yongqing. „Web3D learning framework for 3D shape retrieval based on hybrid convolutional neural networks". *Tsinghua Science and Technology* 25.1 (2019), S. 93–102.

[669] Zhu, Jing; Rizzo, John-Ross und Fang, Yi. „Learning domain-invariant feature for robust depth-image-based 3D shape retrieval". *Pattern recognition letters* 119 (2019), S. 24–33.

[670] Li, Zhaoqun; Xu, Cheng und Leng, Biao. „Rethinking loss design for large-scale 3D shape retrieval". *arXiv preprint arXiv:1906.00546* (2019).

[671] Wang, Zihao; Lin, Hongwei; Yu, Xiaofeng und Hamza, Yusuf Fatihu. „A dimensional reduction guiding deep learning architecture for 3D shape retrieval". *Computers & Graphics* 81 (2019), S. 82–91.

[672] Wang, Huibing; Li, Haohao; Peng, Jinjia und Fu, Xianping. „Multi-feature distance metric learning for non-rigid 3D shape retrieval". *Multimedia Tools and Applications* 78.21 (2019), S. 30943–30958.

[673] Zhu, Zongxiao; Rao, Cong; Bai, Song und Latecki, Longin Jan. „Training convolutional neural network from multi-domain contour images for 3D shape retrieval". *Pattern Recognition Letters* 119 (2019), S. 41–48.

[674] Xu, Cheng; Leng, Biao; Chen, Bo; Zhang, Cheng und Zhou, Xiaochen. „Learning discriminative and generative shape embeddings for three-dimensional shape retrieval". *IEEE Transactions on Multimedia* 22.9 (2019), S. 2234–2245.

[675] Li, Haohao; Wang, Shengfa; Li, Nannan; Su, Zhixun und Liu, Ximin. „Non-rigid 3D shape retrieval based on multi-view metric learning". *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE. 2019, S. 441–446.

[676] Lu, Guoyu und Han, Yahong. „3D Shape Retrieval through Multilayer RBF Neural Network". *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2019, S. 2394–2398.

[677] Zhu, Yiran; Kang, Jiaqi; Lv, Chenlei; Xu, Shu; Xue, Yanping; Wang, Xingce und Wu, Zhongke. „Skeleton Tree based Non-rigid 3D Shape Retrieval". *Proceedings of the 12th International Symposium on Visual Information Communication and Interaction*. 2019, S. 1–2.

[678] Zhang, Qingqing und Duan, Chunmei. „Anisotropic Laplace-Beltrami Operators for Non-Rigid 3D Shape Retrieval". *2019 10th International Conference on Information Technology in Medicine and Education (ITME)*. IEEE. 2019, S. 607–611.

[679] Zhang, Chao; Zhou, Guanghui; Yang, Haidong; Xiao, Zhongdong und Yang, Xiongjun. „View-based 3-D CAD model retrieval with deep residual networks". *IEEE Transactions on Industrial Informatics* 16.4 (2019), S. 2335–2345.

[680] Gao, Zan; Xue, Haixin und Wan, Shaohua. „Multiple discrimination and pairwise CNN for view-based 3D object retrieval". *Neural Networks* 125 (2020), S. 290–302.

[681] Nie, Weizhi; Zhao, Yue; Liu, An-An; Gao, Zan und Su, Yuting. „Multi-graph convolutional network for unsupervised 3D shape retrieval". *Proceedings of the 28th ACM international conference on multimedia*. 2020, S. 3395–3403.

[682] Li, Zirui; Xu, Junyu; Zhao, Yue; Li, Wenhui und Nie, Weizhi. „Mpan: Multi-part attention network for point cloud based 3d shape retrieval". *IEEE Access* 8 (2020), S. 157322–157332.

[683] Fu, Rao; Yang, Jie; Sun, Jiawei; Zhang, Fang-Lue; Lai, Yu-Kun und Gao, Lin. „RISA-Net: Rotation-invariant structure-aware network for fine-grained 3D shape retrieval". *arXiv preprint arXiv:2010.00973* (2020).

[684] Afolabi, Oladapo; Yang, Allen Y und Sastry, S Shankar. „Extending deepsdf for automatic 3d shape retrieval and similarity transform estimation". *arXiv preprint arXiv:2004.09048* (2020).

[685] Li, Zhaoqun. „Gram Regularization for Multi-view 3D Shape Retrieval". *arXiv preprint arXiv:2011.07733* (2020).

[686] Bai, Junjie; Gong, Biao; Zhao, Yining; Lei, Fuqiang; Yan, Chenggang und Gao, Yue. „Multi-scale representation learning on hypergraph for 3D shape retrieval and recognition". *IEEE Transactions on Image Processing* 30 (2021), S. 5327–5338.

[687] Uy, Mikaela Angelina; Kim, Vladimir G; Sung, Minhyuk; Aigerman, Noam; Chaudhuri, Siddhartha und Guibas, Leonidas J. „Joint learning of 3d shape retrieval and deformation". *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, S. 11713–11722.

[688] Lin, Ming-Xian; Yang, Jie; Wang, He; Lai, Yu-Kun; Jia, Rongfei; Zhao, Binqiang und Gao, Lin. „Single image 3d shape retrieval via cross-modal instance and category contrastive learning". *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, S. 11405–11415.

[689] Luo, Luqing; Tang, Lulu; Liu, Rui; Zhang, Xiaoli und Yang, Zhi-Xin. „Multi-Modality Learning for Non-Rigid 3D Shape Retrieval via Structured Sparsity Regularizations". *IEEE Sensors Journal* 21.20 (2021), S. 22985–22994.

[690] Nie, Weizhi; Zhao, Yue; Nie, Jie; Liu, An-An und Zhao, Sicheng. „CLN: Cross-domain learning network for 2D image-based 3D shape retrieval". *IEEE Transactions on Circuits and Systems for Video Technology* 32.3 (2021), S. 992–1005.

[691] Jin, Jiongchao; Xu, Huanqiang; Tang, Zehao; Ji, Pengliang und Xiong, Zhang. „PREMA: Part-based REcurrent Multi-view Aggregation Network for 3D Shape Retrieval". *2021 2nd International Conference on Computer Science and Management Technology (ICCSMT)*. IEEE. 2021, S. 311–318.

[692] Kim, Hyungki; Yeo, Changmo; Cha, Moohyun und Mun, Duhwan. „A method of generating depth images for view-based shape retrieval of 3D CAD models from partial point clouds". *Multimedia Tools and Applications* 80 (2021), S. 10859–10880.

[693] Chen, Jiangce; Ilies, Horea T und Ding, Caiwen. „Graph-based shape analysis for heterogeneous geometric datasets: Similarity, retrieval and substructure matching". *Computer-Aided Design* 143 (2022), S. 103125.

[694] Agapaki, Eva und Brilakis, Ioannis. „Geometric digital twinning of industrial facilities: Retrieval of industrial shapes". *arXiv preprint arXiv:2202.04834* (2022).

[695] Krahe, Carmen; Marinov, Milan; Schmutz, Theresa; Hermann, Yannik; Bonny, Mike; May, Marvin und Lanza, Gisela. „AI based geometric similarity search supporting component reuse in engineering design". *Procedia CIRP* 109 (2022), S. 275–280.

[696] Wang, Dong; Wang, Bin; Yao, Hongxun und Liu, Hong. „Center-push loss for joint view-based 3D model classification and retrieval feature learning". *Signal, Image and Video Processing* 17.4 (2023), S. 873–880.

[697] Otsu, Nobuyuki. „A Threshold Selection Method from Gray-Level Histograms". *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), S. 62–66.

[698] Kasturi, Rangachar; Bow, Sing T.; El-Masri, Wassim; Shah, Jayesh; Gattiker, James R. und Mokate, Umesh B. „A system for interpretation of line drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.10 (1990), S. 978–992.

[699] Tanigawa, Satohide; Hori, Osamu und Shimotsuji, Shigeyoshi. „Precise line detection from an engineering drawing using a figure fitting method based on contours and skeletons". *Proceedings of the 12th IAPR International Conference on Pattern Recognition (Cat. No.94CH3440-5)*. IEEE Comput. Soc. Press, 1994.

[700] Han, C.-C und Fan, K.-C. „Skeleton generation of engineering drawings via contour matching". *Pattern Recognition* 27.2 (1994), S. 261–275.

[701] Den Hartog, J.E. und Kate, Ton K. ten. „Finding arrows in utility maps using a neural network". *Proceedings of the 12th IAPR International Conference on Pattern Recognition (Cat. No.94CH3440-5)*. IEEE Comput. Soc. Press, 1994.

[702] Messmer, Bruno T. und Bunke, Horst. „Automatic learning and recognition of graphical symbols in engineering drawings". Springer, Berlin, Heidelberg, 1996, S. 123–134. https://link.springer.com/chapter/10.1007/3-540-61226-2_11.

[703] Chiang, John Y.; Tue, S.C. und Leu, Y.C. „A NEW ALGORITHM FOR LINE IMAGE VECTORIZATION". *Pattern Recognition* 31.10 (1998), S. 1541–1549.

[704] Sauvola, J. und Pietikäinen, M. „Adaptive document image binarization". *Pattern Recognition* 33.2 (2000), S. 225–236.

[705] Hilaire, Xavier und Tombre, Karl. „Robust and accurate vectorization of line drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.6 (2006), S. 890–904.

[706] Zhang, Jia Yi; Zhao, Long Fei und Hao, Yong Ping. „Multi-Level Block Information Extraction in Engineering Drawings Based on Depth-First Algorithm". *Automation Equipment and Systems* 468-471 (2012), S. 2100–2103. https://www.scientific.net/amr.468-471.2100.

[707] Rezaei, S. B.; Shanbehzadeh, J. und Sarrafzadeh, Hossein. „Adaptive document image skew estimation". *97898814* (2017). https://www.researchbank.ac.nz/handle/10652/3869.

[708] Liu, Tianbo; Hua, Quanbin; Yuan, Shuai; Yin, Lixue und Cheng, Gong. „Anchor Point based Hough Transformation for Automatic Line Detection of Engineering Drawings". *2019 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*. IEEE, 2019.

[709] Vedaldi, Andrea; Bischof, Horst; Brox, Thomas und Frahm, Jan-Michael. *Deep Vectorization of Technical Drawings*. Bd. 12358. Cham: Springer International Publishing, 2020.

[710] Yang, D. S.; Webster, J. L.; Renmdell, L. A.; Garrett, J. H. und Shaw, D. S. „Management of graphical symbols in a CAD environment: A neural network approach". *Proceedings of 1993 IEEE Conference on Tools with Al (TAI-93)*. IEEE Comput. Soc. Press, 1993, S. 272–279.

[711] Yang, Dershung; Rendell, Larry A; Webster, Julie L; Shaw, Doris S und Garrett Jr, James H. „SYMBOL RECOGNITION IN A CAD ENVIRONMENT USING A NEURAL NETWORK". *International Journal on Artificial Intelligence Tools* 03.02 (1994), S. 157–185.

[712] Lladós, Josep; López-Krahe, Jaime und Martí, Enric. „A system to understand hand-drawn floor plans using subgraph isomorphism and Hough transform". *Machine Vision and Applications* 10.3 (1997), S. 150–158. https://link.springer.com/article/10.1007/s001380050068.

[713] Ah-Soon, Christian. „A constraint network for symbol detection in architectural drawings". Springer, Berlin, Heidelberg, 1998, S. 80–90. https://link.springer.com/chapter/10.1007/3-540-64381-8_41.

[714] Dosch, Philippe; Tombre, Karl; Ah-Soon, Christian und Masini, Gérald. „A complete system for the analysis of architectural drawings". *International Journal on Document Analysis and Recognition* 3.2 (2000), S. 102–116. https://link.springer.com/article/10.1007/pl00010901.

[715] Ah-Soon, Christian und Tombre, Karl. „Architectural symbol recognition using a network of constraints". *Pattern Recognition Letters* 22.2 (2001), S. 231–248.

[716] Lladós, Josep; Martí, Enric und Villanueva, Juan J. „Symbol recognition by error-tolerant subgraph matching between region adjacency graphs". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.10 (2001), S. 1137–1143.

[717] Song, Jiqiang; Su, Feng; Tai, Chiew-Lan und Cai, Shijie. „An object-oriented progressive-simplification-based vectorization system for engineering drawings: model, algorithm, and performance". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.8 (2002), S. 1048–1060.

[718] Valveny, Ernest und Marti, Enric. „A model for image generation and symbol recognition through the deformation of lineal shapes". *Pattern Recognition Letters* 24.15 (2003), S. 2857–2867.

[719] Cao, Yang; Li, Heng und Liang, Yong. „Using engineering drawing interpretation for automatic detection of version information in CADD engineering drawing". *Automation in Construction* 14.3 (2005), S. 361–367.

[720] Lu, Tong; Yang, Huafei; Yang, Ruoyu und Cai, Shijie. „Automatic analysis and integration of architectural drawings". *International Journal of Document Analysis and Recognition (IJDAR)* 9.1 (2007), S. 31–47. https://link.springer.com/article/10.1007/s10032-006-0029-6.

[721] Lu, Tong; Yang, Yubin; Yang, Ruoyu und Cai, Shijie. „Knowledge Extraction from Structured Engineering Drawings". *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*. IEEE, 2008.

[722] Le Bodic, Pierre; Locteau, Hervé; Adam, Sébastien; Héroux, Pierre; Lecourtier, Yves und Knippel, Arnaud. „Symbol Detection Using Region Adjacency Graphs and Integer Linear Programming". *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009.

[723] Rusiñol, Marçal; Lladós, Josep und Sánchez, Gemma. „Symbol spotting in vectorized technical drawings through a lookup table of region strings". *Pattern Analysis and Applications* 13.3 (2010), S. 321–331. https://link.springer.com/article/10.1007/s10044-009-0161-2.

[724] Nayef, Nibal und Breuel, Thomas M. „Statistical Grouping for Segmenting Symbols Parts from Line Drawings, with Application to Symbol Spotting". *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011.

[725] Barducci, Alessio und Marinai, Simone. „Object recognition in floor plans by graphs of white connected components". *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. 2012, S. 298–301.

[726] Nayef, Nibal. „Geometric-based symbol spotting and retrieval in technical line drawings". Diss. Technische Universität Kaiserslautern, 2013. https://kluedo.ub.uni-kl.de/frontdoor/index/index/docid/3386.

[727] Nayef, Nibal und Breuel, Thomas M. „Building a Symbol Library from Technical Drawings by Identifying Repeating Patterns". Springer, Berlin, Heidelberg, 2013, S. 69–78. https://link.springer.com/chapter/10.1007/978-3-642-36824-0_7.

[728] Nayef, Nibal und Breuel, Thomas M. „Combining geometric matching with SVM to improve symbol spotting". SPIE, 2013, S. 141–149. https://www.spiedigitallibrary.org/conference-proceedings-of-spie/8658/86580g/combining-geometric-matching-with-svm-to-improve-symbol-spotting/10.1117/12.2002795.short.

[729] Nayef, Nibal und Breuel, Thomas M. „Efficient symbol retrieval by building a symbol index from a collection of line drawings". SPIE, 2013, S. 320–331. https://www.spiedigitallibrary.org/conference-proceedings-of-spie/8658/86580y/efficient-symbol-retrieval-by-building-a-symbol-index-from-a/10.1117/12.2008532.short.

[730] Dutta, Anjan; Lladós, Josep und Pal, Umapada. „A symbol spotting approach in graphical documents by hashing serialized graphs". *Pattern Recognition* 46.3 (2013), S. 752–768.

[731] Zhang, Han und Li, Xueqing. „Data Extraction from DXF File and Visual Display". Springer, Cham, 2014, S. 286–291. https://link.springer.com/chapter/10.1007/978-3-319-07857-1_51.

[732] Riba, Pau; Dutta, Anjan; Llados, Josep und Fornes, Alicia. „Graph-Based Deep Learning for Graphics Classification". *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2017.

[733] Drapeau, Jordan; Géraud, Thierry; Coustaty, Mickaël; Chazalon, Joseph; Burie, Jean-Christophe; Eglin, Véronique und Bres, Stéphane. „Extraction of Ancient Map Contents Using Trees of Connected Components". Springer, Cham, 2018, S. 115–130. https://link.springer.com/chapter/10.1007/978-3-030-02284-6_9.

[734] Ziran, Zahra und Marinai, Simone. „Object Detection in Floor Plan Images". Springer, Cham, 2018, S. 383–394. https://link.springer.com/chapter/10.1007/978-3-319-99978-4_30.

[735] Renton, Guillaume; Heroux, Pierre; Gauzere, Benoit und Adam, Sebastien. „Graph Neural Network for Symbol Detection on Document Images". *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE, 2019.

[736] Rezvanifar, Alireza; Cote, Melissa und Branzan Albu, Alexandra. „Symbol spotting for architectural drawings: state-of-the-art and new industry-driven developments". *IPSJ Transactions on Computer Vision and Applications* 11.1 (2019), S. 1–22. https://ipsjcva.springeropen.com/articles/10.1186/s41074-019-0055-1.

[737] Rezvanifar, Alireza; Cote, Melissa und Albu, Alexandra Branzan. „Symbol Spotting on Digital Architectural Floor Plans Using a Deep Learning-based Framework". *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2020.

[738] Surikov, Ilya Y.; Nakhatovich, Mikhail A.; Belyaev, Sergey Y. und Savchuk, Daniil A. „Floor Plan Recognition and Vectorization Using Combination UNet, Faster-RCNN, Statistical Component Analysis and Ramer-Douglas-Peucker". Springer, Singapore, 2020, S. 16–28. https://link.springer.com/chapter/10.1007/978-981-15-6648-6_2.

[739] Goyal, Shreya; Chattopadhyay, Chiranjoy und Bhatnagar, Gaurav. „Knowledge-driven description synthesis for floor plan interpretation". *International Journal of Document Analysis and Recognition (IJDAR)* 24.1-2 (2021), S. 19–32. https://link.springer.com/article/10.1007/s10032-021-00367-3.

[740] Goyal, Shreya. „Fine Grained Feature Representation Using Computer Vision Techniques for Understanding Indoor Space". Diss. Indian Institute of Technology Jodhpur, Jodhpur und Computer Science and Engineering, 2021. http://theses.iitj.ac.in:8080/jspui/handle/123456789/243.

[741] Fan, Zhiwen; Zhu, Lingjie; Li, Honghua; Chen, Xiaohao; Zhu, Siyu und Tan, Ping. „FloorPlanCAD: A Large-Scale CAD Drawing Dataset for Panoptic Symbol Spotting". *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2021.

[742] Mishra, Shashank; Hashmi, Khurram Azeem; Pagani, Alain; Liwicki, Marcus; Stricker, Didier und Afzal, Muhammad Zeshan. „Towards Robust Object Detection in Floor Plan Images: A Data Augmentation Approach". *Applied Sciences* 11.23 (2021), S. 11174. https://www.mdpi.com/2076-3417/11/23/11174.

[743] Evangelou, Iordanis; Savelonas, Michalis und Papaioannou, Georgios. „PU learning-based recognition of structural elements in architectural floor plans". *Multimedia Tools and Applications* 80.9 (2021), S. 13235–13252. https://link.springer.com/article/10.1007/s11042-020-10295-9.

[744] Park, Sungsoo und Kim, Hyeoncheol. „3DPlanNet: generating 3D models from 2D floor plan images using ensemble methods". *Electronics* 10.22 (2021), S. 2729.

[745] Sarkar, Sourish; Pandey, Pranav und Kar, Sibsambhu. *Automatic Detection and Classification of Symbols in Engineering Drawings*. arXiv, 2022.

[746] Fan, Zhiwen; Chen, Tianlong; Wang, Peihao und Wang, Zhangyang. „CADTransformer: Panoptic Symbol Spotting Transformer for CAD Drawings". *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022.

[747] Zheng, Zhaohua; Li, Jianfang; Zhu, Lingjie; Li, Honghua; Petzold, Frank und Tan, Ping. „GAT-CADNet: Graph Attention Network for Panoptic Symbol Spotting in CAD Drawings". *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022.

[748] Shehzadi, Tahira; Hashmi, Khurram Azeem; Pagani, Alain; Liwicki, Marcus; Stricker, Didier und Afzal, Muhammad Zeshan. „Mask-Aware Semi-Supervised Object Detection in Floor Plans". *Applied Sciences* 12.19 (2022), S. 9398. https://www.mdpi.com/2076-3417/12/19/9398.

[749] Jakubik, Johannes; Hemmer, Patrick; Vössing, Michael; Blumenstiel, Benedikt; Bartos, Andrea und Mohr, Kamilla. „Designing a Human-in-the-Loop System for Object Detection in Floor Plans". *Proceedings of the AAAI Conference on Artificial Intelligence*. Bd. 36. 11. 2022, S. 12524–12530.

[750] Mafipour, M Saeed; Ahmed, Daniyal; Vilgertshofer, Simon und Borrmann, Andre. „Digitalization of 2D Bridge Drawings Using Deep Learning Models". *Prof. of the 30th Int. Conference on Intelligent Computing in Engineering (EG-ICE)*. 2023.

[751] Faltin, Benedikt; Schönfelder, Phillip und König, Markus. „Improving Symbol Detection on Engineering Drawings Using a Keypoint-Based Deep Learning Approach" ().

[752] Smith, William AP und Pillatt, Toby. „You only look for a symbol once: An object detector for symbols and regions in documents". *International Conference on Document Analysis and Recognition*. Springer. 2023, S. 227–243.

[753] Stefenon, Stefano Frizzo; Cristoforetti, Marco und Cimatti, Alessandro. „Towards Automatic Digitalization of Railway Engineering Schematics". *International Conference of the Italian Association for Artificial Intelligence*. Springer. 2023, S. 453–466.

[754] Huang, Wenjun; Sun, Qun; Yu, Anzhu; Guo, Wenyue; Xu, Qing; Wen, Bowei und Xu, Li. „Leveraging Deep Convolutional Neural Network for Point Symbol Recognition in Scanned Topographic Maps". *ISPRS International Journal of Geo-Information* 12.3 (2023), S. 128.

[755] Schönfelder, Phillip; Stebel, Fynn; Andreou, Nikos und König, Markus. „Deep learning-based text detection and recognition on architectural floor plans". *Automation in Construction* 157 (2024), S. 105156.

[756] Janssen, Rik D.T. und Vossepoel, Albert M. „Adaptive Vectorization of Line Drawing Images". *Computer Vision and Image Understanding* 65.1 (1997), S. 38–56.

[757] Yang, Su. „Symbol recognition via statistical integration of pixel-level constraint histograms: a new descriptor". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.2 (2005), S. 278–281.

[758] Zhang, Wan; Wenyin, Liu und Zhang, Kun. „Symbol recognition with kernel density matching". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.12 (2006), S. 2020–2024.

[759] Rusiñol, Marçal und Lladós, Josep. „Symbol Spotting in Technical Drawings Using Vectorial Signatures". Springer, Berlin, Heidelberg, 2006, S. 35–46. https://link.springer.com/chapter/10.1007/11767978_4.

[760] Barrat, Sabine; Tabbone, Salvatore und Nourrissier, Patrick. „A Bayesian classifier for symbol recognition". 2007, 9 pages. https://hal.inria.fr/inria-00179764/ (besucht am 2007).

[761] Yu, Yajie; Zhang, Wan und Liu, Wenyin. „A New Syntactic Approach to Graphic Symbol Recognition". *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. IEEE, 2007.

[762] Terrades, O Ramos; Valveny, Ernest und Tabbone, Salvatore. „On the Combination of Ridgelets Descriptors for Symbol Recognition". Springer, Berlin, Heidelberg, 2008, S. 40–50. https://link.springer.com/chapter/10.1007/978-3-540-88188-9_5.

[763] Luqman, Muhammad Muzzamil; Brouard, Thierry und Ramel, Jean-Yves. „Graphic Symbol Recognition Using Graph Based Signature and Bayesian Network Classifier". *2009 10th International Conference on Document Analysis and Recognition*. IEEE, 2009.

[764] Barrat, Sabine und Tabbone, Salvatore. „A Bayesian network for combining descriptors: application to symbol recognition". *International Journal of Document Analysis and Recognition (IJDAR)* 13.1 (2010), S. 65–75. https://link.springer.com/article/10.1007/s10032-009-0103-y.

[765] Coustaty, Mickael; Bertet, Karell; Visani, Muriel und Ogier, Jean-Marc. „A new adaptive structural signature for symbol recognition by using a galois lattice as a classifier". *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41.4 (2011), S. 1136–1148.

[766] Ghosh, Swarnendu; Shaw, Prasenjit; Das, Nibaran und Santosh, K. C. „GSD-Net: Compact Network for Pixel-Level Graphical Symbol Detection". *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. IEEE, 2019.

[767] Elliman, Dave. „TIF2VEC, An Algorithm for Arc Segmentation in Engineering Drawings". Springer, Berlin, Heidelberg, 2002, S. 350–358. https://link.springer.com/chapter/10.1007/3-540-45868-9_31.

[768] Santosh, K. C.; Wendling, Laurent und Lamiroy, Bart. „BoR: BAG-OF-RELATIONS FOR SYMBOL RETRIEVAL". *International Journal of Pattern Recognition and Artificial Intelligence* 28.06 (2014), S. 1450017.

[769] Karasneh, Bilal und Chaudron, Michel R.V. „Img2UML: A System for Extracting UML Models from Images". *2013 39th Euromicro Conference on Software Engineering and Advanced Applications*. IEEE, 2013, S. 134–137.

[770] Ho-Quang, Truong; Chaudron, Michel R.V.; Samuelsson, Ingimar; Hjaltason, Joel; Karasneh, Bilal und Osman, Hafeez. „Automatic Classification of UML Class Diagrams from Images". *2014 21st Asia-Pacific Software Engineering Conference*. IEEE, 2014, S. 399–406.

[771] Shcherban, Sergei; Liang, Peng; Li, Zengyang und Yang, Chen. „Multiclass classification of UML diagrams from images using deep learning". *International Journal of Software Engineering and Knowledge Engineering* 31.11n12 (2021), S. 1683–1698.

[772] Bunke, Horst. „Automatic Interpretation of Lines and Text in Circuit Diagrams". *Pattern Recognition Theory and Applications*. Springer, Dordrecht, 1982, S. 297–310. https://link.springer.com/chapter/10.1007/978-94-009-7772-3_18.

[773] Groen, Frans C.A; Sanderson, Arthur C. und Schlag, John F. „Symbol recognition in electrical diagrams using probabilistic graph matching". *Pattern Recognition Letters* 3.5 (1985), S. 343–350.

[774] Okazaki, Akio; Kondo, Takashi; Mori, Kazuhiro; Tsunekawa, Shou und Kawamoto, Eiji. „An automatic circuit diagram reader with loop-structure-based symbol recognition". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10.3 (1988), S. 331–341.

[775] Fahn, Chin-Shyurng; Wang, Jhing-Fa und Lee, Jau-Yien. „A topology-based component extractor for understanding electronic circuit diagrams". *Computer Vision, Graphics, and Image Processing* 43.2 (1988), S. 279.

[776] Lee, Seong-Whan; Kim, J. H.I.N. und Groen, Frans C. A. „TRANSLATION-, ROTATION- AND SCALE- INVARIANT RECOGNITION OF HAND-DRAWN SYMBOLS IN SCHEMATIC DIAGRAMS". *International Journal of Pattern Recognition and Artificial Intelligence* 04.01 (1990), S. 1–25.

[777] Kim, S. H.; Suh, J. W. und Kim, J. H. „Recognition of logic diagrams by identifying loops and rectilinear polylines". *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR '93)*. IEEE Comput. Soc. Press, 1993.

[778] Cheng, T.; Khan, J.; Liu, H. und Yun, D.Y.Y. „A symbol recognition system". *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR '93)*. IEEE Comput. Soc. Press, 1993.

[779] Hamada, A. H. „A new system for the analysis of schematic diagrams". *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR '93)*. IEEE Comput. Soc. Press, 1993.

[780] Yu, Bin. „Automatic understanding of symbol-connected diagrams". *Proceedings of 3rd International Conference on Document Analysis and Recognition*. IEEE Comput. Soc. Press, 1995.

[781] Bailey, Donald; Norman, Andrew und Moretti, Giovanni. „Electronic Schematic Recognition" (1995).

[782] Yan, Luo und Wenyin, Liu. „Engineering drawings recognition using a case-based approach". *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings*. IEEE Comput. Soc, 2003.

[783] Zesheng, Shi; Jing, Yang; Chunhong, Jiang und Yonggui, Wu. „Symbol recognition in electronic diagrams using decision tree". *Proceedings of 1994 IEEE International Conference on Industrial Technology - ICIT '94*. IEEE, 2003.

[784] Baum, Larry; Boose, John; Boose, Molly; Chaplin, Carey und Provine, Ron. „Extracting System-Level Understanding from Wiring Diagram Manuals". Springer, Berlin, Heidelberg, 2004, S. 100–108. https://link.springer.com/chapter/10.1007/978-3-540-25977-0_9.

[785] Ouyang, Tom Y. und Davis, Randall. „A visual approach to sketched symbol recognition" (2009). https://dspace.mit.edu/handle/1721.1/71572.

[786] Feng, Guihuan; Viard-Gaudin, Christian und Sun, Zhengxing. „On-line hand-drawn electric circuit diagram recognition using 2D dynamic programming". *Pattern Recognition* 42.12 (2009), S. 3215–3223.

[787] De, Paramita; Mandal, Sekhar und Bhowmick, Partha. „Recognition of electrical symbols in document images using morphology and geometric analysis". *2011 International Conference on Image Information Processing*. IEEE, 2011, S. 1–6.

[788] Santosh, K. C.; Lamiroy, Bart und Wendling, Laurent. „Symbol recognition using spatial relations". *Pattern Recognition Letters* 33.3 (2012), S. 331–341.

[789] Datta, Ramanath; Mandal, Paramita De Sekhar und Chanda, Bhabatosh. „Detection and identification of logic gates from document images using mathematical morphology". *2015 Fifth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*. IEEE, 2015.

[790] Rabbani, Mahdi; Khoshkangini, Reza; Nagendraswamy, H. S. und Conti, Mauro. „Hand Drawn Optical Circuit Recognition". *Procedia Computer Science* 84 (2016), S. 41–48.

[791] Agarwal, Shubham; Agrawal, Mohit und Chaudhury, Santanu. „Recognizing Electronic Circuits to Enrich Web Documents for Electronic Simulation". Springer, Cham, 2017, S. 60–74. https://link.springer.com/chapter/10.1007/978-3-319-52159-6_5.

[792] Stoitchkov, Deian und Vilgertshofer, Simon. „Analysis of methods for automated symbol recognition in technical drawings". *Technical University of Munich* (2018). https://publications.cms.bgu.tum.de/theses/2018_stoitchkov_vilgertshofer.pdf.

[793] Datta, Ramanath; Mandal, Sekhar und Biswas, Samit. „Automatic Abstraction of Combinational Logic Circuit from Scanned Document Page Images". *Pattern Recognition and Image Analysis* 29.2 (2019), S. 212–223.

[794] Peng, Zhang; Yan, Geng; Zhongshan, Qu; Huiyong, Liu; Mouying, Li und Shengnan, Liu. „CIM/G graphics automatic generation in substation primary wiring diagram based on image recognition". *Journal of Physics: Conference Series*. Bd. 1617. 1. IOP Publishing. 2020, S. 012007.

[795] Mizanur Rahman, Syed; Bayer, Johannes und Dengel, Andreas. „Graph-Based Object Detection Enhancement for Symbolic Engineering Drawings". *Document Analysis and Recognition – ICDAR 2021 Workshops*. Hrsg. von Barney Smith, Elisa H. und Pal, Umapada. Bd. 12916. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, S. 74–90.

[796] Thoma, Felix; Bayer, Johannes; Li, Yakun und Dengel, Andreas. „A public ground-truth dataset for handwritten circuit diagram images". *International Conference on Document Analysis and Recognition*. Springer. 2021, S. 20–27.

[797] Ramadhan, Dilovan S. und Al-Khaffaf, Hasan S. M. „Symbol Spotting in Electronic Images Using Morphological Filters and Hough Transform". *Science Journal of University of Zakho* 10.3 (2022), S. 119–129. http://sjuoz.uoz.edu.krd/index.php/sjuoz/article/view/874.

[798] Shen, Chao; Lv, Peisen; Mao, Mengge; Li, Wenbin; Zhao, Kun und Yan, Zhen. „Substation One-Line Diagram Automatic Generation Based On Image Recongnition". *2022 Global Conference on Robotics, Artificial Intelligence and Information Technology (GCRAIT)*. 2022, S. 247–251.

[799] Bayer, Johannes; Roy, Amit Kumar und Dengel, Andreas. „Instance segmentation based graph extraction for handwritten circuit diagram images". *arXiv preprint arXiv:2301.03155* (2023).

[800] Uzair, Waqas; Chai, Douglas und Rassau, Alexander. „Electronet: An enhanced model for small-scale object detection in electrical schematic diagrams" (2023).

[801] Bhanbhro, Hina; Hooi, Yew Kwang; Zakaria, Mohammad Nordin Bin; Hassan, Zaira und Pitafi, Shahneela. „Single Line Electrical Drawings (SLED): A Multiclass Dataset Benchmarked by Deep Neural Networks". *2023 IEEE 13th International Conference on System Engineering and Technology (ICSET)*. IEEE. 2023, S. 66–71.

[802] Yang, Chen; Wang, Jingyu; Yang, Lvyang; Shi, Dongyuan und Duan, Xianzhong. „Intelligent Digitization of Substation One-Line Diagrams Based on Computer Vision". *IEEE Transactions on Power Delivery* 38.6 (2023), S. 3912–3923.

[803] Song, Aibo; Kun, Huang; Peng, Bowen; Chen, Rui; Zhao, Kun; Qiu, Jingyi und Wang, Kaixuan. „EDRS: an Automatic System to Recognize Electrical Drawings". *2021 China Automation Congress (CAC)*. 2021, S. 5438–5443.

[804] Wenyin, Liu; Zhang, Wan und Yan, Luo. „An interactive example-driven approach to graphics recognition in engineering drawings". *International Journal of Document Analysis and Recognition (IJDAR)* 9.1 (2007), S. 13–29. https://link.springer.com/article/10.1007/s10032-006-0025-x.

[805] Qureshi, Rashid Jalal; Ramel, Jean-Yves; Barret, Didier und Cardot, Hubert. „Spotting Symbols in Line Drawing Images Using Graph Representations". Springer, Berlin, Heidelberg, 2008, S. 91–103. https://link.springer.com/chapter/10.1007/978-3-540-88188-9_10.

[806] Yu, Yuhong; Samal, Ashok und Seth, Sharad. „Isolating symbols from connection lines in a class of engineering drawings". *Pattern Recognition* 27.3 (1994), S. 391–404.

[807] Pham, The-Anh; Delalandre, Mathieu; Barrat, Sabine und Ramel, Jean-Yves. „Accurate junction detection and characterization in line-drawing images". *Pattern Recognition* 47.1 (2014), S. 282–295.

[808] Dori, Dov und Wenyin, Liu. „Vector-based segmentation of text connected to graphics in engineering drawings". Springer, Berlin, Heidelberg, 1996, S. 322–331. https://link.springer.com/chapter/10.1007/3-540-61577-6_33.

[809] Joseph, Steve H. „Processing of engineering line drawings for automatic input to CAD". *Pattern Recognition* 22.1 (1989), S. 1–11.

[810] Krause, Frank Lothar; Jansen, Helmut; Großmann, G. und Spur, Guenter H.C. „Automatic Scanning and Interpretation of Engineering Drawings for CAD-Processes". *CIRP Annals* 38.1 (1989), S. 437–441.

[811] Nagasamy, Vijay und Langrana, Noshir A. „Engineering drawing processing and vectorization system". *Computer Vision, Graphics, and Image Processing* 49.1 (1990), S. 125–126.

[812] Lysak, Daniel B. und Kasturi, Rangachar. „Interpretation of line drawings with multiple views". *Proceedings. 10th International Conference on Pattern Recognition*. IEEE Comput. Soc. Press, 1990.

[813] Kultanen, Pekka. „Randomized Hough Transform (RHT) in Engineering Drawing Vectorization System". *Proc. MVA* (1990). https://www.cvl.iis.u-tokyo.ac.jp/mva/proceedings/commemorativedvd/1990/papers/1990173.pdf.

[814] Lai, Chan P. und Kasturi, Rangachar. „Detection of dashed lines in engineering drawings and maps". *Proc. First Int. Conf. on Document Analysis and Recognition*. 1991, S. 507–515.

[815] Dori, Dov. „Dimensioning analysis". *Communications of the ACM* 35.10 (1992), S. 92–103.

[816] Joseph, S. H. und Pridmore, Tony. „Knowledge-directed interpretation of mechanical engineering drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.9 (1992), S. 928–940.

[817] Min, Weidong; Tang, Zesheng und Tang, Long. „Recognition of dimensions in engineering drawings based on arrowhead". *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR '93)*. IEEE Comput. Soc. Press, 1993.

[818] Lai, Chan Pyng und Kasturi, R. „Detection of dimension sets in engineering drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16.8 (1994), S. 848–855.

[819] Vaxiviere, Pascal und Tombre, Karl. *Knowledge organization and interpretation process in engineering drawing interpretation*. 1994. https://members.loria.fr/ktombre/das94_vaxiviere_tombre.pdf.

[820] Collin, Suzanne und Colnet, Dominique. „SYNTACTIC ANALYSIS OF TECHNICAL DRAWING DIMENSIONS". *International Journal of Pattern Recognition and Artificial Intelligence* 08.05 (1994), S. 1131–1148.

[821] Das, Atish K. und Langrana, Noshir A. „Recognition of dimension sets and integration with vectorized engineering drawings". *Proceedings of 3rd International Conference on Document Analysis and Recognition*. IEEE. IEEE Comput. Soc. Press, 1995.

[822] Dori, Dov. „Vector-based arc segmentation in the machine drawing understanding system environment". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17.11 (1995), S. 1057–1068.

[823] Capellades, María A. und Camps, Octavia I. „Functional parts detection in engineering drawings: Looking for the screws". Springer, Berlin, Heidelberg, 1996, S. 246–259. https://link.springer.com/chapter/10.1007/3-540-61226-2_20.

[824] He, Shoujie und Abe, Norihiro. „A clustering-based approach to the separation of text strings from mixed text/graphics documents". *Proceedings of 13th International Conference on Pattern Recognition*. IEEE, 1996.

[825] Chen, Yuan; Langrana, Noshir A. und Das, Atish K. „Perfecting Vectorized Mechanical Drawings". *Computer Vision and Image Understanding* 63.2 (1996), S. 273–286.

[826] Priestnall, Gary; Marston, R. E. und Elliman, David G. „Arrowhead recognition during automated data capture". *Pattern Recognition Letters* 17.3 (1996), S. 277–286.

[827] Dori, Dov. „Orthogonal Zig-Zag: An algorithm for vectorizing engineering drawings compared with Hough Transform". *Advances in Engineering Software* 28.1 (1997), S. 11–24.

[828] Lu, Zhaoyang. „Detection of text regions from digital engineering drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.4 (1998), S. 431–439.

[829] Stahovich, Thomas F.; Davis, Randall und Shrobe, Howard. „Generating multiple new designs from a sketch". *Artificial Intelligence* 104.1-2 (1998), S. 211–264.

[830] Dori, Dov und Velkovitch, Yelena. „Segmentation and Recognition of Dimensioning Text from Engineering Drawings". *Computer Vision and Image Understanding* 69.2 (1998), S. 196–201.

[831] Ablameyko, Sergey; Bereishik, Vladimir; Frantskevich, Oleg; Homenko, Maria und Paramonova, Nadeshda. „A system for automatic recognition of engineering drawing entities". *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)*. IEEE Comput. Soc, 1998.

[832] Dori, Dov und Wenyin, Liu. „Automated CAD conversion with the Machine Drawing Understanding System: concepts, algorithms, and performance". *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 29.4 (1999), S. 411–416.

[833] Prabhu, B. S. „Automatic extraction of manufacturable features from CADD models using syntactic pattern recognition techniques". *International Journal of Production Research* 37.6 (1999), S. 1259–1281.

[834] Habed, Adlane und Boufama, Boubakeur. „Dimension sets detection in technical drawings". *Proc. IAPR*. Bd. 99. 1999, S. 217–223. http://w3.gel.ulaval.ca/~parizeau/vi99/pdf-files/s8/paper53.pdf.

[835] Devaux, Pierre M.; Lysak, Daniel B. und Kasturi, Rangachar. „A complete system for the intelligent interpretation of engineering drawings". *International Journal of Document Analysis and Recognition (IJDAR)* 2.2-3 (1999), S. 120–131. https://link.springer.com/article/10.1007/s100320050042.

[836] Adam, Sébastien; Ogier, Jean-Marc; Cariou, Claude; Mullot, Rémy; Labiche, Jacques und Gardes, Joel. „Symbol and character recognition: application to engineering drawings". *International Journal on Document Analysis and Recognition* 3.2 (2000), S. 89–101. https://link.springer.com/article/10.1007/s100320000033.

[837] Müller, Stefan und Rigoll, Gerhard. „Engineering Drawing Database Retrieval Using Statistical Pattern Spotting Techniques". Springer, Berlin, Heidelberg, 2000, S. 246–255. https://link.springer.com/chapter/10.1007/3-540-40953-x_21.

[838] Prabhu, B.S; Biswas, S. und Pande, S.S. „Intelligent system for extraction of product data from CADD models". *Computers in Industry* 44.1 (2001), S. 79–95.

[839] Ramel, Jean-Yves und Vincent, Nicole. „Strategy for Line Drawing Understanding". Springer, Berlin, Heidelberg, 2004, S. 1–12. https://link.springer.com/chapter/10.1007/978-3-540-25977-0_1.

[840] Wendling, Laurent und Tabbone, Salvatore. „A new way to detect arrows in line drawings". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.7 (2004), S. 935–941.

[841] Ondrejcek, Michal; Kastner, Jason; Kooper, Rob und Bajcsy, Peter. „Information extraction from scanned engineering drawings". *National Center for Supercomputing Applications, University of Illinoisat Urbana-Champaign, Image Spatial DataAnalysis Group* (2009).

[842] Jiang, Zhiqiang; Feng, Xilan; Xianzhang Feng und Yuanpeng Liu. „An information extraction of title panel in engineering drawings and automatic generation system of three statistical tables". *2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)*. IEEE, 2010.

[843] Intwala, Aditya Mukesh; Kharade, Kiran; Chaugule, Ravindra und Magikar, Atul. „Dimensional Arrow Detection from CAD Drawings". *Indian Journal of Science and Technology* 9.21 (2016).

[844] Brock, Andrew; Lim, Theodore; Ritchie, J. M. und Weston, Nick. „ConvNet-Based Optical Recognition for Engineering Drawings". *ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (2017), DETC2017–68186. https://researchportal.hw.ac.uk/en/publications/convnet-based-optical-rec%5C-og%5C-ni%5C-tion-for-en%5C-gi%5C-neer%5C-ing-draw%5C-ings.

[845] Alwan, Salwan; Caillec, Jean-Marc und Meur, Gerard. „Detection of Primitives in Engineering Drawing using Genetic Algorithm". *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*. SCITEPRESS - Science and Technology Publications, 2019.

[846] Scheibel, Beate; Mangler, Juergen und Rinderle-Ma, Stefanie. „Extraction of dimension requirements from engineering drawings for supporting quality control in production processes". *Computers in Industry* 129 (2021), S. 103442.

[847] van Daele, Dries; Decleyre, Nicholas; Dubois, Herman und Meert, Wannes. „An Automated Engineering Assistant: Learning Parsers for Technical Drawings". *Proceedings of the AAAI Conference on Artificial Intelligence* 35.17 (2021), S. 15195–15203. https://ojs.aaai.org/index.php/aaai/article/view/17783.

[848] Zhang, Wentai; Chen, Quan; Koz, Can; Xie, Liuyue; Regmi, Amit; Yamakawa, Soji; Furuhata, Tomotake; Shimada, Kenji und Kara, Levent Burak. „Data Augmentation of Engineering Drawings for Data-Driven Component Segmentation". American Society of Mechanical Engineers Digital Collection, 2022.

[849] Haar, Christoph; Kim, Hangbeom und Koberg, Lukas. „AI-Based Engineering and Production Drawing Information Extraction". *International Conference on Flexible Automation and Intelligent Manufacturing*. Springer. 2022, S. 374–382.

[850] Xie, Liuyue; Lu, Yao; Furuhata, Tomotake; Yamakawa, Soji; Zhang, Wentai; Regmi, Amit; Kara, Levent und Shimada, Kenji. „Graph neural network-enabled manufacturing method classification from engineering drawings". *Computers in Industry* 142 (2022), S. 103697.

[851] Kashevnik, Alexey; Shilov, Nikolay; Teslya, Nikolay; Hasan, Fudail; Kitenko, Andrey; Dukareva, Veronika; Abdurakhimov, Marat; Zingarevich, Alexander und Blokhin, Dmitry. „An Approach to Engineering Drawing Organization: Title Block Detection and Processing". *IEEE Access* (2023).

[852] Kashevnik, Alexey; Ali, Ammar und Mayatin, Alexander. „AI-Based Method for Frame Detection in Engineering Drawings". *2023 International Russian Smart Industry Conference (SmartIndustryCon)*. IEEE. 2023, S. 225–229.

[853] Zhang, Wentai; Joseph, Joe; Yin, Yue; Xie, Liuyue; Furuhata, Tomotake; Yamakawa, Soji; Shimada, Kenji und Kara, Levent Burak. „Component segmentation of engineering drawings using Graph Convolutional Networks". *Computers in Industry* 147 (2023), S. 103885. https://www.sciencedirect.com/science/article/pii/S0166361523000350.

[854] Lin, Yi-Hsin; Ting, Yu-Hung; Huang, Yi-Cyun; Cheng, Kai-Lun und Jong, Wen-Ren. „Integration of Deep Learning for Automatic Recognition of 2D Engineering Drawings". *Machines* 11.8 (2023), S. 802.

[855] Besbes, Mohamed Dhia Elhak; Ferdousi, Zahra Vahidi; Tabia, Hedi und Fradi, Mouna. „2D Cross-View Object Segmentation and Perceptual Grouping in Computer-Aided Design Drawings". *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, S. 1745–1754.

[856] Xu, Yuanping; Zhang, Chaolong; Xu, Zhijie; Kong, Chao; Tang, Dan; Deng, Xin; Li, Tukun und Jin, Jin. „Tolerance Information Extraction for Mechanical Engineering Drawings–A Digital Image Processing and Deep Learning-based Model". *CIRP Journal of Manufacturing Science and Technology* 50 (2024), S. 55–64.

[857] Guo, Tiantian; Zhang, Hui und Wen, Yamei. „An improved example-driven symbol recognition approach in engineering drawings". *Computers & Graphics* 36.7 (2012), S. 835–845.

[858] Das, Supriya; Banerjee, Purnendu; Seraogi, Bhagesh; Majumder, Himadri; Mukkamala, Srinivas; Roy, Rahul und Chaudhuri, Bidyut Baran. „Hand-written and machine-printed text classification in architecture, engineering & construction documents". *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE. 2018, S. 546–551.

[859] Kara, Levent Burak und Stahovich, Thomas F. „An image-based, trainable symbol recognizer for hand-drawn sketches". *Computers & Graphics* 29.4 (2005), S. 501–517.

[860] Lee, WeeSan; Kara, Levent Burak und Stahovich, Thomas F. *An Efficient Graph-Based Symbol Recognizer*. 2006. https://www.researchgate.net/profile/levent-kara/publication/240755155_an_efficient_graph-based_symbol_recognizer/links/0046353bdfa212c48d000000/an-efficient-graph-based-symbol-recognizer.pdf.

[861] Lee, WeeSan; Kara, Levent Burak und Stahovich, Thomas F. „An efficient graph-based recognizer for hand-drawn symbols". *Computers & Graphics* 31.4 (2007), S. 554–567.

[862] Della Ventura, Anna und Schettini, Raimondo. „Graphic symbol recognition using a signature technique". *Proceedings of the 12th IAPR International Conference on Pattern Recognition (Cat. No.94CH3440-5)*. IEEE Comput. Soc. Press, 1994.

[863] Yim, Sok Yee; Ananthakumar, Hari Govind; Benabbas, Benabbas; Horch, Alexander; Drath, Rainer und Thornhill, Nina F. „Using process topology in plant-wide control loop performance assessment". *Computers & Chemical Engineering* 31.2 (2006), S. 86–99.

[864] Gellaboina, Mahesh Kumar und Venkoparao, Vijendran G. „Graphic Symbol Recognition Using Auto Associative Neural Network Model". *2009 Seventh International Conference on Advances in Pattern Recognition*. IEEE, 2009.

[865] Hoang, Xuan Luu; Arroyo, Esteban und Fay, Alexander. „Automatische Analyse und Erkennung graphischer Inhalte von SVG-basierten Engineering-Dokumenten". *at - Automatisierungstechnik* 2 (2016), S. 133–146. https://www.degruyter.com/document/doi/10.1515/auto-2015-0089/html.

[866] Moreno-García, Carlos Francisco; Elyan, Eyad und Jayne, Chrisina. „Heuristics-Based Detection to Improve Text/Graphics Segmentation in Complex Engineering Drawings". Springer, Cham, 2017, S. 87–98. https://link.springer.com/chapter/10.1007/978-3-319-65172-9_8.

[867] Elyan, Eyad; Moreno-García, Carlos und Jayne, Chrisina. „Symbols Classification in Engineering Drawings“. *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018.

[868] Kang, Sung-O; Lee, Eul-Bum und Baek, Hum-Kyung. „A Digitization and Conversion Tool for Imaged Drawings to Intelligent Piping and Instrumentation Diagrams (P&ID)“. *Energies* 12.13 (2019), S. 2593. https://www.mdpi.com/491776.

[869] Rahul, Rohit; Paliwal, Shubham; Sharma, Monika und Vig, Lovekesh. *Automatic Information Extraction from Piping and Instrumentation Diagrams*. arXiv, 2019.

[870] Yu, Eun-Seop; Cha, Jae-Min; Lee, Taekyong; Kim, Jinil und Mun, Duhwan. „Features Recognition from Piping and Instrumentation Diagrams in Image Format Using a Deep Learning Network“. *Energies* 12.23 (2019), S. 4425. https://www.mdpi.com/578736.

[871] Elyan, Eyad; Jamieson, Laura und Ali-Gombe, Adamu. *Deep learning for symbols detection and classification in engineering drawings*. 2020. https://www.dbpia.co.kr/journal/articledetail?nodeid=node10527450.

[872] Elyan, Eyad; Moreno-García, Carlos Francisco und Johnston, Pamela. „Symbols in Engineering Drawings (SiED): An Imbalanced Dataset Benchmarked by Convolutional Neural Networks“. Springer, Cham, 2020, S. 215–224. https://link.springer.com/chapter/10.1007/978-3-030-48791-1_16.

[873] Nurminen, Jukka K.; Rainio, Kari; Numminen, Jukka-Pekka; Syrjänen, Timo; Paganus, Niklas und Honkoila, Karri. „Object Detection in Design Diagrams with Machine Learning“. Springer, Cham, 2020, S. 27–36. https://link.springer.com/chapter/10.1007/978-3-030-19738-4_4.

[874] Yun, Dong-Yeol; Seo, Seung-Kwon; Zahid, Umer und Lee, Chul-Jin. „Deep Neural Network for Automatic Image Recognition of Engineering Diagrams“. *Applied Sciences* 10.11 (2020), S. 4005.

[875] Zhang, Yuxi; Cai, Jiannan und Cai, Hubo. *CNN-Based Symbol Recognition in Piping Drawings*. American Society of Civil Engineers, 2020.

[876] Bayer, Johannes und Sinha, Arka. *Graph-Based Manipulation Rules for Piping and Instrumentation Diagrams*. Center for Open Science, 2020.

[877] Jamieson, Laura; Moreno-Garcia, Carlos Francisco und Elyan, Eyad. „Deep Learning for Text Detection and Recognition in Complex Engineering Diagrams". *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.

[878] Mani, Shouvik; Haddad, Michael A.; Constantini, Dan; Douhard, Willy; Li, Qiwei und Poirier, Louis. „Automatic Digitization of Engineering Diagrams using Deep Learning and Graph Search". *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2020.

[879] Rica, Elena; Moreno-García, Carlos Francisco; Álvarez, Susana und Serratosa, Francesc. „Reducing human effort in engineering drawing validation". *Computers in Industry* 117 (2020), S. 103198.

[880] Sierla, Seppo; Azangoo, Mohammad; Fay, Alexander; Vyatkin, Valeriy und Papakonstantinou, Nikolaos. „Integrating 2D and 3D Digital Plant Information Towards Automatic Generation of Digital Twins". *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*. IEEE, 2020.

[881] Gao, Wei; Zhao, Yunfei und Smidts, Carol. „Component detection in piping and instrumentation diagrams of nuclear power plants based on neural networks". *Progress in Nuclear Energy* 128 (2020), S. 103491.

[882] Paliwal, Shubham; Sharma, Monika und Vig, Lovekesh. „OSSR-PID: One-Shot symbol recognition in P&ID sheets using path sampling and GCN". *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2021, S. 1–8.

[883] Moon, Yoochan; Lee, Jinwon; Mun, Duhwan und Lim, Seungeun. „Deep Learning-Based Method to Recognize Line Objects and Flow Arrows from Image-Format Piping and Instrumentation Diagrams for Digitization". *Applied Sciences* 11.21 (2021), S. 10054.

[884] Rica, Elena; Álvarez, Susana und Serratosa, Francesc. „Group of components detection in engineering drawings based on graph matching". *Engineering Applications of Artificial Intelligence* 104 (2021), S. 104404.

[885] Ghadekar, Premanand; Joshi, Shaunak; Swain, Debabrata; Acharya, Biswaranjan; Pradhan, Manas Ranjan und Patro, Pramoda. „Automatic Digitization of Engineering Diagrams using Intelligent Algorithms". *Journal of Computer Science* 17.9 (2021), S. 833–838. http://research.skylineuniversity.ac.ae/id/eprint/146/.

[886] Shakhshir, Nadir und Serratosa, Francesc. *Engineering Drawing Validation Based on Graph Convolutional Networks*. 2021. http://deim.urv.cat/~francesc.serratosa/2021_02_06_nadir_shakhshir_mistic_masterthesis.pdf.

[887] Paliwal, Shubham; Jain, Arushi; Sharma, Monika und Vig, Lovekesh. „Digitize-PID: Automatic Digitization of Piping and Instrumentation Diagrams". Springer, Cham, 2021, S. 168–180. https://link.springer.com/chapter/10.1007/978-3-030-75015-2_17.

[888] Stinner, Florian; Wiecek, Martin; Baranski, Marc; Kümpel, Alexander und Müller, Dirk. „Automatic digital twin data model generation of building energy systems from piping and instrumentation diagrams". *arXiv preprint arXiv:2108.13912* (2021).

[889] Ismail, Muhammad Hami Asma'i Bin. „Identification of Objects in Oilfield Infrastructure using Engineering Diagram and Machine Learning Methods". *2021 IEEE Symposium on Computers & Informatics (ISCI)*. 2021, S. 19–24.

[890] Gada, Mihir. „Object detection for P&ID images using various deep learning techniques". *2021 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE. 2021, S. 1–5.

[891] Bin, Ong Kai; Hooi, Yew Kwang; Kadir, Said Jadid Abdul; Fujita, Haruhiro und Rosli, Luqman Hakim. „Enhanced Symbol Recognition based on Advanced Data Augmentation for Engineering Diagrams". *International Journal of Advanced Computer Science and Applications* 13.5 (2022).

[892] Bhanbhro, Hina; Hooi, Yew Kwang; Hassan, Zaira und Sohu, Najamuddin. „Modern Deep Learning Approaches for Symbol Detection in Complex Engineering Drawings". *2022 International Conference on Digital Transformation and Intelligence (ICDI)*. IEEE. 2022, S. 121–126.

[893] Kim, Byung Chul; Kim, Hyungki; Moon, Yoochan; Lee, Gwang und Mun, Duhwan. „End-to-end digitization of image format piping and instrumentation diagrams at an industrially applicable level". *Journal of Computational Design and Engineering* 9.4 (2022), S. 1298–1326.

[894] Gupta, Mohit; Wei, Chialing und Czerniawski, Thomas. „Automated Valve Detection in Piping and Instrumentation (P&ID) Diagrams". *Proceedings of the 39th International Symposium on Automation and Robotics in Construction, Bogota, Colombia*. 2022, S. 12–15.

[895] Bhanbhro, Hina; Kwang Hooi, Yew; Kusakunniran, Worapan und Amur, Zaira Hassan. „A Symbol Recognition System for Single-Line Diagrams Developed Using a Deep-Learning Approach". *Applied Sciences* 13.15 (2023), S. 8816.

[896] Gajbhiye, Shreya M; Bhamre, SR; Tadepalli, LN Teja; Pillai, MR und Uplaonkar, Deepak. „Advancing P&ID Digitization with YOLOv5". *2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS)*. IEEE. 2023, S. 1–6.

[897] Xiao, Xin; Li, Zhongtao; Zhao, Shuai; Yang, Lei; Zhao, Fu und Ge, Chuan. „Improved P&ID Symbol Detection Algorithm Based on YOLOv5 Network". *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. 2023, S. 120–126.

[898] Shin, Ho-Jin; Lee, Ga-Young und Lee, Chul-Jin. „Automatic anomaly detection in engineering diagrams using machine learning". *Korean Journal of Chemical Engineering* 40.11 (2023), S. 2612–2623.

[899] Theisen, Maximilian F; Flores, Kenji Nishizaki; Balhorn, Lukas Schulze und Schweidtmann, Artur M. „Digitization of chemical process flow diagrams using deep convolutional neural networks". *Digital Chemical Engineering* 6 (2023), S. 100072.

[900] Moon, Yoochan; Han, Seung-Tae; Lee, Jinwon und Mun, Duhwan. „Extraction of line objects from piping and instrumentation diagrams using an improved continuous line detection algorithm". *Journal of Mechanical Science and Technology* 37.4 (2023), S. 1959–1972.

[901] Kim, Gwangsik und Kim, Byung Chul. „Classification of functional types of lines in P&IDs using a graph neural network". *IEEE Access* (2023).

[902] Stürmer, Jan Marius; Graumann, Marius und Koch, Tobias. „Demonstrating Automated Generation of Simulation Models from Engineering Diagrams". *2023 International Conference on Machine Learning and Applications (ICMLA)*. IEEE. 2023, S. 1156–1162.

[903] Liu, Shengyong; Li, Zhongtao; Zhao, Shuai; Yang, Lei; Zhao, Fu und Ge, Chuan. „A Complete Piping Identification Solution for Piping and Instrumentation Diagrams". *2023 IEEE International Conference on High Performance Computing & Communications, Data Science & Systems, Smart City & Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*. IEEE. 2023, S. 9–15.

[904] Kang, Dong-Han; Choi, So-Won; Lee, Eul-Bum und Kang, Sung-O. „Auto-Routing Systems (ARSs) with 3D Piping for Sustainable Plant Projects Based on Artificial Intelligence (AI) and Digitalization of 2D Drawings and Specifications". *Sustainability* 16.7 (2024), S. 2770.

[905] Han, Seung-Tae; Moon, Yoochan; Lee, Hyunsik und Mun, Duhwan. „Rule-based continuous line classification using shape and positional relationships between objects in piping and instrumentation diagram". *Expert Systems with Applications* (2024), S. 123366.

[906] Su, Guanqun; Zhao, Shuai; Li, Tao; Liu, Shengyong; Li, Yaqi; Zhao, Guanglong und Li, Zhongtao. „Image format pipeline and instrument diagram recognition method based on deep learning". *Biomimetic Intelligence and Robotics* 4.1 (2024), S. 100142.

[907] Gupta, Mohit; Wei, Chialing und Czerniawski, Thomas. „Semi-supervised symbol detection for piping and instrumentation drawings". *Automation in Construction* 159 (2024), S. 105260.

[908] Hadsell, Raia; Chopra, Sumit und LeCun, Yann. „Dimensionality reduction by learning an invariant mapping". *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*. Bd. 2. IEEE. 2006, S. 1735–1742.

[909] Bell, Sean und Bala, Kavita. „Learning visual similarity for product design with convolutional neural networks". *ACM transactions on graphics (TOG)* 34.4 (2015), S. 1–10.

[910] Hoffer, Elad und Ailon, Nir. „Deep metric learning using triplet network". *Similarity-Based Pattern Recognition: Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015. Proceedings 3*. Springer. 2015, S. 84–92.

Verzeichnis promotionsbezogener, eigener Publikationen

[P1] Pickel, Jessica; Bickel, Sebastian; Goetz, Stefan und Wartzack, Sandro. „Integration of product development data for further ontological utilization". *Proceedings of the Design Society* 4 (2024), S. 463–472.

[P2] Spruegel, Tobias C; Bickel, Sebastian; Schleich, Benjamin und Wartzack, Sandro. „Approach and application to transfer heterogeneous simulation data from finite element analysis to neural networks". *Journal of Computational Design and Engineering* 8.1 (2021), S. 298–315.

[P3] Bickel, Sebastian; Goetz, Stefan und Wartzack, Sandro. „Detection of Singularities in Finite Element Simulations: A 3D Deep Learning Approach". *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Bd. 87295. American Society of Mechanical Engineers. 2023, V002T02A039.

[P4] Bickel, Sebastian; Schleich, Benjamin und Wartzack, Sandro. „Detection and classification of symbols in principle sketches using deep learning". *Proceedings of the Design Society* 1 (2021), S. 1183–1192.

[P5] Bickel, Sebastian; Goetz, Stefan und Wartzack, Sandro. „Detection of plausibility and error reasons in finite element simulations with deep learning networks". *Algorithms* 16.4 (2023), S. 209.

[P6] Bickel, Sebastian; Goetz, Stefan und Wartzack, Sandro. „Symbol Detection in Mechanical Engineering Sketches: Experimental Study on Principle Sketches with Synthetic Data Generation and Deep Learning." *Applied Sciences (2076-3417)* 14.14 (2024).

[P7] Bickel, Sebastian; Schleich, Benjamin und Wartzack, Sandro. „A New Projection Based Method for the Classification of Mechanical Components Using Convolutional Neural Networks". *Proceedings of the Design Society* 2 (2022), S. 1501–1510.

[P8] Qie, Yifan; Bickel, Sebastian; Wartzack, Sandro; Schleich, Benjamin und Anwer, Nabil. „A function-oriented surface reconstruction framework for reverse engineering". *CIRP Annals* 70.1 (2021), S. 135–138.

[P9] Bickel, Sebastian; Sauer, Christopher; Schleich, Benjamin und Wartzack, Sandro. „Comparing CAD part models for geometrical similarity: A concept using machine learning algorithms". *Procedia CIRP* 96 (2021), S. 133–138.

[P10] Bickel, Sebastian; Schleich, Benjamin und Wartzack, Sandro. „A novel shape retrieval method for 3D mechanical components based on object projection, pre-trained deep learning models and autoencoder". *Computer-Aided Design* 154 (2023), S. 103417.

Verzeichnis promotionsbezogener, studentischer Arbeiten

[S1] Baron, Adrian. „Application of data mining methods for the recognition and classification of components in a technical drawing". Projektarbeit. 2019.

[S2] Cao, Zhaojie. „Analyse von Methoden zur Objekt-Erkennung und deren Eignung für die Produktentwicklung". Projektarbeit. 2020.

[S3] Rebhan, Marcel. „Anwendung von Objekt-Erkennungs-Verfahren zur Detektion und Klassifizierung unterschiedlicher Bestandteile von prinzipiellen Lösungen". Masterarbeit. 2021.

[S4] Bittner, Julian. „Entwicklung eines Vorgehensmodells zur Detektion und Kontextualisierung unterschiedlicher Bestandteile in handgezeichneten technischen Skizzen". Bachelorarbeit. 2023.

[S5] Lieg, Leonard. „Verbesserung und Erweiterung einer Methode zur automatisierten Erkennung und Nutzbarmachung von Prinzipskizzen". Masterarbeit. 2024.

[S6] Seitz, Markus Johannes. „Untersuchung der Anwendbarkeit von Visual Transformer zur Erkennung von Bauteilen der Produktentwicklung". Masterarbeit. 2023.

[S7] Kassler, Ulli. „Klassifizierung von CAD-Bauteilen durch die Anwendung der Projektionsmethode und neuronaler Netze". Masterarbeit. 2021.

[S8] Frastaoui, Mostafa. „Entwicklung eines Vorgehensmodells zur Übertragung von CAD-Baugruppen in Graphen". Masterarbeit. 2022.

[S9] Lang, Louis. „Entwicklung eines Vorgehensmodells zur Suche von ähnlichen Baugruppen auf Basis von Prinzip-Skizzen mittels Machine Learning". Masterarbeit. 2023.

[S10] Näpflein, Josua. „Untersuchung von Einflussfaktoren auf die Ähnlichkeitssuche mittels Projektionsmethode". Masterarbeit. 2021.

[S11] Seidel, Leon. „Analyse von Methoden zur Shape Segmentation und deren Eignung für die Produktentwicklung". Bachelorarbeit. 2021.

[S12] Schäfer, Florian. „Verbesserung der Ähnlichkeitssuche von CAD-Geometrien mittels Projektionsmethode durch Unsupervised Learning Verfahren". Projektarbeit. 2022.

[S13] Mühlberger, Leon. „Entwicklung eines Vorgehensmodells zur Anwendung von Metric Learning zur Verbesserung der Ähnlichkeitssuche von CAD-Bauteilen". Bachelorarbeit. 2023.

Reihenübersicht

Koordination der Reihe (Stand 2025):
Geschäftsstelle Maschinenbau, Dr.-Ing. Oliver Kreis, www.mb.fau.de/diss/

Im Rahmen der Reihe sind bisher die nachfolgenden Bände erschienen.

Band 1 – 52
Fertigungstechnik – Erlangen
ISSN 1431-6226
Carl Hanser Verlag, München

Band 53 – 307
Fertigungstechnik – Erlangen
ISSN 1431-6226
Meisenbach Verlag, Bamberg

ab Band 308
FAU Studien aus dem Maschinenbau
ISSN 2625-9974
FAU University Press, Erlangen

Die Zugehörigkeit zu den jeweiligen Lehrstühlen ist wie folgt gekennzeichnet:

Lehrstühle:

FAPS	Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik
FMT	Lehrstuhl für Fertigungsmesstechnik
KTmfk	Lehrstuhl für Konstruktionstechnik
LFT	Lehrstuhl für Fertigungstechnologie
LGT	Lehrstuhl für Gießereitechnik
LPT	Lehrstuhl für Photonische Technologien
REP	Lehrstuhl für Ressourcen- und Energieeffiziente Produktionsmaschinen

Band 1: Andreas Hemberger
Innovationspotentiale in der rechnerintegrierten Produktion durch wissensbasierte Systeme
FAPS, 208 Seiten, 107 Bilder. 1988. ISBN 3-446-15234-2.

Band 2: Detlef Classe
Beitrag zur Steigerung der Flexibilität automatisierter Montagesysteme durch Sensorintegration und erweiterte Steuerungskonzepte
FAPS, 194 Seiten, 70 Bilder. 1988. ISBN 3-446-15529-5.

Band 3: Friedrich-Wilhelm Nolting
Projektierung von Montagesystemen
FAPS, 201 Seiten, 107 Bilder, 1 Tab. 1989.ISBN 3-446-15541-4.

Band 4: Karsten Schlüter
Nutzungsgradsteigerung von Montagesystemen durch den Einsatz der Simulationstechnik
FAPS, 177 Seiten, 97 Bilder. 1989. ISBN 3-446-15542-2.

Band 5: Shir-Kuan Lin
Aufbau von Modellen zur Lageregelung von Industrierobotern
FAPS, 168 Seiten, 46 Bilder. 1989. ISBN 3-446-15546-5.

Band 6: Rudolf Nuss
Untersuchungen zur Bearbeitungsqualität im Fertigungssystem Laserstrahlschneiden
LFT, 206 Seiten, 115 Bilder, 6 Tab. 1989. ISBN 3-446-15783-2.

Band 7: Wolfgang Scholz
Modell zur datenbankgestützten Planung automatisierter Montageanlagen
FAPS, 194 Seiten, 89 Bilder. 1989. ISBN 3-446-15825-1.

Band 8: Hans-Jürgen Wißmeier
Beitrag zur Beurteilung des Bruchverhaltens von Hartmetall-Fließpreßmatrizen
LFT, 179 Seiten, 99 Bilder, 9 Tab. 1989. ISBN 3-446-15921-5.

Band 9: Rainer Eisele
Konzeption und Wirtschaftlichkeit von Planungssystemen in der Produktion
FAPS, 183 Seiten, 86 Bilder. 1990. ISBN 3-446-16107-4.

Band 10: Rolf Pfeiffer
Technologisch orientierte Montageplanung am Beispiel der Schraubtechnik
FAPS, 216 Seiten, 102 Bilder, 16 Tab. 1990. ISBN 3-446-16161-9.

Band 11: Herbert Fischer
Verteilte Planungssysteme zur Flexibilitätssteigerung der rechnerintegrierten Teilefertigung
FAPS, 201 Seiten, 82 Bilder. 1990. ISBN 3-446-16105-8.

Band 12: Gerhard Kleineidam
CAD/CAP: Rechnergestützte Montagefeinplanung
FAPS, 203 Seiten, 107 Bilder. 1990. ISBN 3-446-16112-0.

Band 13: Frank Vollertsen
Pulvermetallurgische Verarbeitung eines übereutektoiden verschleißfesten Stahls
LFT, XIII u. 217 Seiten, 67 Bilder, 34 Tab. 1990. ISBN 3-446-16133-3.

Band 14: Stephan Biermann
Untersuchungen zur Anlagen- und Prozeßdiagnostik für das Schneiden mit CO_2-Hochleistungslasern
LFT, VIII u. 170 Seiten, 93 Bilder, 4 Tab. 1991. ISBN 3-446-16269-0.

Band 15: Uwe Geißler
Material- und Datenfluß in einer flexiblen Blechbearbeitungszelle
LFT, 124 Seiten, 41 Bilder, 7 Tab. 1991. ISBN 3-446-16358-1.

Band 16: Frank Oswald Hake
Entwicklung eines rechnergestützten Diagnosesystems für automatisierte Montagezellen
FAPS, XIV u. 166 Seiten, 77 Bilder. 1991. ISBN 3-446-16428-6.

Band 17: Herbert Reichel
Optimierung der Werkzeugbereitstellung durch rechnergestützte Arbeitsfolgenbestimmung
FAPS, 198 Seiten, 73 Bilder, 2 Tab. 1991. ISBN 3-446-16453-7.

Band 18: Josef Scheller
Modellierung und Einsatz von Softwaresystemen für rechnergeführte Montagezellen
FAPS, 198 Seiten, 65 Bilder. 1991. ISBN 3-446-16454-5.

Band 19: Arnold vom Ende
Untersuchungen zum Biegeumformen mit elastischer Matrize
LFT, 166 Seiten, 55 Bilder, 13 Tab. 1991. ISBN 3-446-16493-6.

Band 20: Joachim Schmid
Beitrag zum automatisierten Bearbeiten von Keramikguß mit Industrierobotern
FAPS, XIV u. 176 Seiten, 111 Bilder, 6 Tab. 1991. ISBN 3-446-16560-6.

Band 21: Egon Sommer
Multiprozessorsteuerung für kooperierende Industrieroboter in Montagezellen
FAPS, 188 Seiten, 102 Bilder. 1991. ISBN 3-446-17062-6.

Band 22: Georg Geyer
Entwicklung problemspezifischer Verfahrensketten in der Montage
FAPS, 192 Seiten, 112 Bilder. 1991. ISBN 3-446-16552-5.

Band 23: Rainer Flohr
Beitrag zur optimalen Verbindungstechnik in der Oberflächenmontage (SMT)
FAPS, 186 Seiten, 79 Bilder. 1991. ISBN 3-446-16568-1.

Band 24: Alfons Rief
Untersuchungen zur Verfahrensfolge Laserstrahlschneiden und -schweißen in der Rohkarosseriefertigung
LFT, VI u. 145 Seiten, 58 Bilder, 5 Tab. 1991. ISBN 3-446-16593-2.

Band 25: Christoph Thim
Rechnerunterstützte Optimierung von Materialflußstrukturen in der Elektronikmontage durch Simulation
FAPS, 188 Seiten, 74 Bilder. 1992. ISBN 3-446-17118-5.

Band 26: Roland Müller
CO_2 -Laserstrahlschneiden von kurzglasverstärkten Verbundwerkstoffen
LFT, 141 Seiten, 107 Bilder, 4 Tab. 1992. ISBN 3-446-17104-5.

Band 27: Günther Schäfer
Integrierte Informationsverarbeitung bei der Montageplanung
FAPS, 195 Seiten, 76 Bilder. 1992. ISBN 3-446-17117-7.

Band 28: Martin Hoffmann
Entwicklung einerCAD/CAM-Prozeßkette für die Herstellung von Blechbiegeteilen
LFT, 149 Seiten, 89 Bilder. 1992. ISBN 3-446-17154-1.

Band 29: Peter Hoffmann
Verfahrensfolge Laserstrahlschneiden und –schweißen: Prozeßführung und Systemtechnik in der 3D-Laserstrahlbearbeitung von Blechformteilen
LFT, 186 Seiten, 92 Bilder, 10 Tab. 1992. ISBN 3-446-17153-3.

Band 30: Olaf Schrödel
Flexible Werkstattsteuerung mit objektorientierten Softwarestrukturen
FAPS, 180 Seiten, 84 Bilder. 1992. ISBN 3-446-17242-4.

Band 31: Hubert Reinisch
Planungs- und Steuerungswerkzeuge zur impliziten Geräteprogrammierung in Roboterzellen
FAPS, XI u. 212 Seiten, 112 Bilder. 1992. ISBN 3-446-17380-3.

Band 32: Brigitte Bärnreuther
Ein Beitrag zur Bewertung des Kommunikationsverhaltens von Automatisierungsgeräten in flexiblen Produktionszellen
FAPS, XI u. 179 Seiten, 71 Bilder. 1992. ISBN 3-446-17451-6.

Band 33: Joachim Hutfless
Laserstrahlregelung und Optikdiagnostik in der Strahlführung einer CO_2-Hochleistungslaseranlage
LFT, 175 Seiten, 70 Bilder, 17 Tab. 1993. ISBN 3-446-17532-6.

Band 34: Uwe Günzel
Entwicklung und Einsatz eines Simulationsverfahrens für operative und strategische Probleme der Produktionsplanung und -steuerung
FAPS, XIV u. 170 Seiten, 66 Bilder, 5 Tab. 1993. ISBN 3-446-17604-7.

Band 35: Bertram Ehmann
Operatives Fertigungscontrolling durch Optimierung auftragsbezogener Bearbeitungsabläufe in der Elektronikfertigung
FAPS, XV u. 167 Seiten, 114 Bilder. 1993. ISBN 3-446-17658-6.

Band 36: Harald Kolléra
Entwicklung eines benutzerorientierten Werkstattprogrammiersystems für das Laserstrahlschneiden
LFT, 129 Seiten, 66 Bilder, 1 Tab. 1993. ISBN 3-446-17719-1.

Band 37: Stephanie Abels
Modellierung und Optimierung von Montageanlagen in einem integrierten Simulationssystem
FAPS, 188 Seiten, 88 Bilder. 1993. ISBN 3-446-17731-0.

Band 38: Robert Schmidt-Hebbel
Laserstrahlbohren durchflußbestimmender Durchgangslöcher
LFT, 145 Seiten, 63 Bilder, 11 Tab. 1993. ISBN 3-446-17778-7.

Band 39: Norbert Lutz
Oberflächenfeinbearbeitung keramischer Werkstoffe mit XeCl-Excimerlaserstrahlung
LFT, 187 Seiten, 98 Bilder, 29 Tab. 1994. ISBN 3-446-17970-4.

Band 40: Konrad Grampp
Rechnerunterstützung bei Test und Schulung an Steuerungssoftware von SMD-Bestücklinien
FAPS, 178 Seiten, 88 Bilder. 1995. ISBN 3-446-18173-3.

Band 41: Martin Koch
Wissensbasierte Unterstützung derAngebotsbearbeitung in der Investitionsgüterindustrie
FAPS, 169 Seiten, 68 Bilder. 1995. ISBN 3-446-18174-1.

Band 42: Armin Gropp
Anlagen- und Prozeßdiagnostik beim Schneiden mit einem gepulsten Nd:YAG-Laser
LFT, 160 Seiten, 88 Bilder, 7 Tab. 1995. ISBN 3-446-18241-1.

Band 43: Werner Heckel
Optische 3D-Konturerfassung und on-line Biegewinkelmessung mit dem Lichtschnittverfahren
LFT, 149 Seiten, 43 Bilder, 11 Tab. 1995. ISBN 3-446-18243-8.

Band 44: Armin Rothhaupt
Modulares Planungssystem zur Optimierung der Elektronikfertigung
FAPS, 180 Seiten, 101 Bilder. 1995. ISBN 3-446-18307-8.

Band 45: Bernd Zöllner
Adaptive Diagnose in der Elektronikproduktion
FAPS, 195 Seiten, 74 Bilder, 3 Tab. 1995. ISBN 3-446-18308-6.

Band 46: Bodo Vormann
Beitrag zur automatisierten Handhabungsplanung komplexer Blechbiegeteile
LFT, 126 Seiten, 89 Bilder, 3 Tab. 1995. ISBN 3-446-18345-0.

Band 47: Peter Schnepf
Zielkostenorientierte Montageplanung
FAPS, 144 Seiten, 75 Bilder. 1995. ISBN 3-446-18397-3.

Band 48: Rainer Klotzbücher
Konzept zur rechnerintegrierten Materialversorgung in flexiblen Fertigungssystemen
FAPS, 156 Seiten, 62 Bilder. 1995. ISBN 3-446-18412-0.

Band 49: Wolfgang Greska
Wissensbasierte Analyse undKlassifizierung von Blechteilen
LFT, 144 Seiten, 96 Bilder. 1995. ISBN 3-446-18462-7.

Band 50: Jörg Franke
Integrierte Entwicklung neuer Produkt- und Produktionstechnologien für räumliche spritzgegossene Schaltungsträger (3-D MID)
FAPS, 196 Seiten, 86 Bilder, 4 Tab. 1995. ISBN 3-446-18448-1.

Band 51: Franz-Josef Zeller
Sensorplanung und schnelle Sensorregelung für Industrieroboter
FAPS, 190 Seiten, 102 Bilder, 9 Tab. 1995. ISBN 3-446-18601-8.

Band 52: Michael Solvie
Zeitbehandlung und Multimedia-Unterstützung in Feldkommunikationssystemen
FAPS, 200 Seiten, 87 Bilder, 35 Tab. 1996. ISBN 3-446-18607-7.

Band 53: Robert Hopperdietzel
Reengineering in der Elektro- und Elektronikindustrie
FAPS, 180 Seiten, 109 Bilder, 1 Tab. 1996. ISBN 3-87525-070-2.

Band 54: Thomas Rebhahn
Beitrag zur Mikromaterialbearbeitung mit Excimerlasern - Systemkomponenten und Verfahrensoptimierungen
LFT, 148 Seiten, 61 Bilder, 10 Tab. 1996. ISBN 3-87525-075-3.

Band 55: Henning Hanebuth
Laserstrahlhartlöten mit Zweistrahltechnik
LFT, 157 Seiten, 58 Bilder, 11 Tab. 1996. ISBN 3-87525-074-5.

Band 56: Uwe Schönherr
Steuerung und Sensordatenintegration für flexible Fertigungszellen mitkooperierenden Robotern
FAPS, 188 Seiten, 116 Bilder, 3 Tab. 1996. ISBN 3-87525-076-1.

Band 57: Stefan Holzer
Berührungslose Formgebung mit-Laserstrahlung
LFT, 162 Seiten, 69 Bilder, 11 Tab. 1996. ISBN 3-87525-079-6.

Band 58: Markus Schultz
Fertigungsqualität beim 3D-Laserstrahlschweißen vonBlechformteilen
LFT, 165 Seiten, 88 Bilder, 9 Tab. 1997. ISBN 3-87525-080-X.

Band 59: Thomas Krebs
Integration elektromechanischer CA-Anwendungen über einem STEP-Produktmodell
FAPS, 198 Seiten, 58 Bilder, 8 Tab. 1997. ISBN 3-87525-081-8.

Band 60: Jürgen Sturm
Prozeßintegrierte Qualitätssicherung in der Elektronikproduktion
FAPS, 167 Seiten, 112 Bilder, 5 Tab. 1997. ISBN 3-87525-082-6.

Band 61: Andreas Brand
Prozesse und Systeme zur Bestückung räumlicher elektronischer Baugruppen (3D-MID)
FAPS, 182 Seiten, 100 Bilder. 1997. ISBN 3-87525-087-7.

Band 62: Michael Kauf
Regelung der Laserstrahlleistung und der Fokusparameter einer CO_2-Hochleistungslaseranlage
LFT, 140 Seiten, 70 Bilder, 5 Tab. 1997. ISBN 3-87525-083-4.

Band 63: Peter Steinwasser
Modulares Informationsmanagement in der integrierten Produkt- und Prozeßplanung
FAPS, 190 Seiten, 87 Bilder. 1997. ISBN 3-87525-084-2.

Band 64: Georg Liedl
Integriertes Automatisierungskonzept für den flexiblen Materialfluß in der Elektronikproduktion
FAPS, 196 Seiten, 96 Bilder, 3 Tab. 1997. ISBN 3-87525-086-9.

Band 65: Andreas Otto
Transiente Prozesse beim Laserstrahlschweißen
LFT, 132 Seiten, 62 Bilder, 1 Tab. 1997. ISBN 3-87525-089-3.

Band 66: Wolfgang Blöchl
Erweiterte Informationsbereitstellung an offenen CNC-Steuerungen zur Prozeß- und Programmoptimierung
FAPS, 168 Seiten, 96 Bilder. 1997. ISBN 3-87525-091-5.

Band 67: Klaus-Uwe Wolf
Verbesserte Prozeßführung und Prozeßplanung zur Leistungs- und Qualitätssteigerung beim Spulenwickeln
FAPS, 186 Seiten, 125 Bilder. 1997. ISBN 3-87525-092-3.

Band 68: Frank Backes
Technologieorientierte Bahnplanung für die 3D-Laserstrahlbearbeitung
LFT, 138 Seiten, 71 Bilder, 2 Tab. 1997. ISBN 3-87525-093-1.

Band 69: Jürgen Kraus
Laserstrahlumformen von Profilen
LFT, 137 Seiten, 72 Bilder, 8 Tab. 1997. ISBN 3-87525-094-X.

Band 70: Norbert Neubauer
Adaptive Strahlführungen für CO_2-Laseranlagen
LFT, 120 Seiten, 50 Bilder, 3 Tab. 1997. ISBN 3-87525-095-8.

Band 71: Michael Steber
Prozeßoptimierter Betrieb flexibler Schraubstationen in der automatisierten Montage
FAPS, 168 Seiten, 78 Bilder, 3 Tab. 1997. ISBN 3-87525-096-6.

Band 72: Markus Pfestorf
Funktionale 3D-Oberflächenkenngrößen in der Umformtechnik
LFT, 162 Seiten, 84 Bilder, 15 Tab. 1997. ISBN 3-87525-097-4.

Band 73: Volker Franke
Integrierte Planung und Konstruktion von Werkzeugen für die Biegebearbeitung
LFT, 143 Seiten, 81 Bilder. 1998. ISBN 3-87525-098-2.

Band 74: Herbert Scheller
Automatisierte Demontagesysteme und recyclinggerechte Produktgestaltung elektronischer Baugruppen
FAPS, 184 Seiten, 104 Bilder, 17 Tab. 1998. ISBN 3-87525-099-0.

Band 75: Arthur Meßner
Kaltmassivumformung metallischer Kleinstteile – Werkstoffverhalten, Wirkflächenreibung, Prozeßauslegung
LFT, 164 Seiten, 92 Bilder, 14 Tab. 1998. ISBN 3-87525-100-8.

Band 76: Mathias Glasmacher
Prozeß- und Systemtechnik zum Laserstrahl-Mikroschweißen
LFT, 184 Seiten, 104 Bilder, 12 Tab. 1998. ISBN 3-87525-101-6.

Band 77: Michael Schwind
Zerstörungsfreie Ermittlung mechanischer Eigenschaften von Feinblechen mit dem Wirbelstromverfahren
LFT, 124 Seiten, 68 Bilder, 8 Tab. 1998. ISBN 3-87525-102-4.

Band 78: Manfred Gerhard
Qualitätssteigerung in der Elektronikproduktion durch Optimierung der Prozeßführung beim Löten komplexer Baugruppen
FAPS, 179 Seiten, 113 Bilder, 7 Tab. 1998. ISBN 3-87525-103-2.

Band 79: Elke Rauh
Methodische Einbindung der Simulation in die betrieblichen Planungs- und Entscheidungsabläufe
FAPS, 192 Seiten, 114 Bilder, 4 Tab. 1998. ISBN 3-87525-104-0.

Band 80: Sorin Niederkorn
Meßeinrichtung zur Untersuchung der Wirkflächenreibung bei umformtechnischen Prozessen
LFT, 99 Seiten, 46 Bilder, 6 Tab. 1998. ISBN 3-87525-105-9.

Band 81: Stefan Schuberth
Regelung der Fokuslage beim Schweißen mit CO_2-Hochleistungslasern unter Einsatz von adaptiven Optiken
LFT, 140 Seiten, 64 Bilder, 3 Tab. 1998. ISBN 3-87525-106-7.

Band 82: Armando Walter Colombo
Development and Implementation of Hierarchical Control Structures of Flexible Production Systems Using High Level Petri Nets
FAPS, 216 Seiten, 86 Bilder. 1998. ISBN 3-87525-109-1.

Band 83: Otto Meedt
Effizienzsteigerung bei Demontage und Recycling durch flexible Demontagetechnologien und optimierte Produktgestaltung
FAPS, 186 Seiten, 103 Bilder. 1998. ISBN 3-87525-108-3.

Band 84: Knuth Götz
Modelle und effiziente Modellbildung zur Qualitätssicherung in der Elektronikproduktion
FAPS, 212 Seiten, 129 Bilder, 24 Tab. 1998. ISBN 3-87525-112-1.

Band 85: Ralf Luchs
Einsatzmöglichkeiten leitender Klebstoffe zur zuverlässigen Kontaktierung elektronischer Bauelemente in der SMT
FAPS, 176 Seiten, 126 Bilder, 30 Tab. 1998. ISBN 3-87525-113-7.

Band 86: Frank Pöhlau
Entscheidungsgrundlagen zur Einführung räumlicher spritzgegossener Schaltungsträger (3-D MID)
FAPS, 144 Seiten, 99 Bilder. 1999. ISBN 3-87525-114-8.

Band 87: Roland T. A. Kals
Fundamentals on the miniaturization of sheet metal working processes
LFT, 128 Seiten, 58 Bilder, 11 Tab. 1999. ISBN 3-87525-115-6.

Band 88: Gerhard Luhn
Implizites Wissen und technisches Handeln am Beispiel der Elektronikproduktion
FAPS, 252 Seiten, 61 Bilder, 1 Tab. 1999. ISBN 3-87525-116-4.

Band 89: Axel Sprenger
Adaptives Streckbiegen von Aluminium-Strangpreßprofilen
LFT, 114 Seiten, 63 Bilder, 4 Tab. 1999. ISBN 3-87525-117-2.

Band 90: Hans-Jörg Pucher
Untersuchungen zur Prozeßfolge Umformen, Bestücken und Laserstrahllöten von Mikrokontakten
LFT, 158 Seiten, 69 Bilder, 9 Tab. 1999. ISBN 3-87525-119-9.

Band 91: Horst Arnet
Profilbiegen mit kinematischer Gestalterzeugung
LFT, 128 Seiten, 67 Bilder, 7 Tab. 1999. ISBN 3-87525-120-2.

Band 92: Doris Schubart
Prozeßmodellierung und Technologieentwicklung beim Abtragen mit CO_2-Laserstrahlung
LFT, 133 Seiten, 57 Bilder, 13 Tab. 1999. ISBN 3-87525-122-9.

Band 93: Adrianus L. P. Coremans
Laserstrahlsintern von Metallpulver - Prozeßmodellierung, Systemtechnik, Eigenschaften laserstrahlgesinterter Metallkörper
LFT, 184 Seiten, 108 Bilder, 12 Tab. 1999. ISBN 3-87525-124-5.

Band 94: Hans-Martin Biehler
Optimierungskonzepte für Qualitätsdatenverarbeitung und Informationsbereitstellung in der Elektronikfertigung
FAPS, 194 Seiten, 105 Bilder. 1999. ISBN 3-87525-126-1.

Band 95: Wolfgang Becker
Oberflächenausbildung und tribologische Eigenschaften excimerlaserstrahlbearbeiteter Hochleistungskeramiken
LFT, 175 Seiten, 71 Bilder, 3 Tab. 1999. ISBN 3-87525-127-X.

Band 96: Philipp Hein
Innenhochdruck-Umformen von Blechpaaren: Modellierung, Prozeßauslegung und Prozeßführung
LFT, 129 Seiten, 57 Bilder, 7 Tab. 1999. ISBN 3-87525-128-8.

Band 97: Gunter Beitinger
Herstellungs- und Prüfverfahren für thermoplastische Schaltungsträger
FAPS, 169 Seiten, 92 Bilder, 20 Tab. 1999. ISBN 3-87525-129-6.

Band 98: Jürgen Knoblach
Beitrag zur rechnerunterstützten verursachungsgerechten Angebotskalkulation von Blechteilen mit Hilfe wissensbasierter Methoden
LFT, 155 Seiten, 53 Bilder, 26 Tab. 1999. ISBN 3-87525-130-X.

Band 99: Frank Breitenbach
Bildverarbeitungssystem zur Erfassung der Anschlußgeometrie elektronischer SMT-Bauelemente
LFT, 147 Seiten, 92 Bilder, 12 Tab. 2000. ISBN 3-87525-131-8.

Band 100: Bernd Falk
Simulationsbasierte Lebensdauervorhersage für Werkzeuge der Kaltmassivumformung
LFT, 134 Seiten, 44 Bilder, 15 Tab. 2000. ISBN 3-87525-136-9.

Band 101: Wolfgang Schlögl
Integriertes Simulationsdaten-Management für Maschinenentwicklung und Anlagenplanung
FAPS, 169 Seiten, 101 Bilder, 20 Tab. 2000. ISBN 3-87525-137-7.

Band 102: Christian Hinsel
Ermüdungsbruchversagen hartstoffbeschichteter Werkzeugstähle in der Kaltmassivumformung
LFT, 130 Seiten, 80 Bilder, 14 Tab. 2000. ISBN 3-87525-138-5.

Band 103: Stefan Bobbert
Simulationsgestützte Prozessauslegung für das Innenhochdruck-Umformen von Blechpaaren
LFT, 123 Seiten, 77 Bilder. 2000. ISBN 3-87525-145-8.

Band 104: Harald Rottbauer
Modulares Planungswerkzeug zum Produktionsmanagement in der Elektronikproduktion
FAPS, 166 Seiten, 106 Bilder. 2001.
ISBN 3-87525-139-3.

Band 105: Thomas Hennige
Flexible Formgebung von Blechen durch Laserstrahlumformen
LFT, 119 Seiten, 50 Bilder. 2001.
ISBN 3-87525-140-7.

Band 106: Thomas Menzel
Wissensbasierte Methoden für die rechnergestützte Charakterisierung und Bewertung innovativer Fertigungsprozesse
LFT, 152 Seiten, 71 Bilder. 2001.
ISBN 3-87525-142-3.

Band 107: Thomas Stöckel
Kommunikationstechnische Integration der Prozeßebene in Produktionssysteme durch Middleware-Frameworks
FAPS, 147 Seiten, 65 Bilder, 5 Tab.
2001. ISBN 3-87525-143-1.

Band 108: Frank Pitter
Verfügbarkeitssteigerung von Werkzeugmaschinen durch Einsatz mechatronischer Sensorlösungen
FAPS, 158 Seiten, 131 Bilder, 8 Tab.
2001. ISBN 3-87525-144-X.

Band 109: Markus Korneli
Integration lokaler CAP-Systeme in einen globalen Fertigungsdatenverbund
FAPS, 121 Seiten, 53 Bilder, 11 Tab.
2001. ISBN 3-87525-146-6.

Band 110: Burkhard Müller
Laserstrahljustieren mit Excimer-Lasern - Prozeßparameter und Modelle zur Aktorkonstruktion
LFT, 128 Seiten, 36 Bilder, 9 Tab.
2001. ISBN 3-87525-159-8.

Band 111: Jürgen Göhringer
Integrierte Telediagnose via Internet zum effizienten Service von Produktionssystemen
FAPS, 178 Seiten, 98 Bilder, 5 Tab.
2001. ISBN 3-87525-147-4.

Band 112: Robert Feuerstein
Qualitäts- und kosteneffiziente Integration neuer Bauelementetechnologien in die Flachbaugruppenfertigung
FAPS, 161 Seiten, 99 Bilder, 10 Tab.
2001. ISBN 3-87525-151-2.

Band 113: Marcus Reichenberger
Eigenschaften und Einsatzmöglichkeiten alternativer Elektroniklote in der Oberflächenmontage (SMT)
FAPS, 165 Seiten, 97 Bilder, 18 Tab.
2001. ISBN 3-87525-152-0.

Band 114: Alexander Huber
Justieren vormontierter Systeme mit dem Nd:YAG-Laser unter Einsatz von Aktoren
LFT, 122 Seiten, 58 Bilder, 5 Tab.
2001. ISBN 3-87525-153-9.

Band 115: Sami Krimi
Analyse und Optimierung von Montagesystemen in der Elektronikproduktion
FAPS, 155 Seiten, 88 Bilder, 3 Tab.
2001. ISBN 3-87525-157-1.

Band 116: Marion Merklein
Laserstrahlumformen von Aluminiumwerkstoffen - Beeinflussung der Mikrostruktur und der mechanischen Eigenschaften
LFT, 122 Seiten, 65 Bilder, 15 Tab.
2001. ISBN 3-87525-156-3.

Band 117: Thomas Collisi
Ein informationslogistisches Architekturkonzept zur Akquisition simulationsrelevanter Daten
FAPS, 181 Seiten, 105 Bilder, 7 Tab.
2002. ISBN 3-87525-164-4.

Band 118: Markus Koch
Rationalisierung und ergonomische Optimierung im Innenausbau durch den Einsatz moderner Automatisierungstechnik
FAPS, 176 Seiten, 98 Bilder, 9 Tab.
2002. ISBN 3-87525-165-2.

Band 119: Michael Schmidt
Prozeßregelung für das Laserstrahl-Punktschweißen in der Elektronikproduktion
LFT, 152 Seiten, 71 Bilder, 3 Tab.
2002. ISBN 3-87525-166-0.

Band 120: Nicolas Tiesler
Grundlegende Untersuchungen zum Fließpressen metallischer Kleinstteile
LFT, 126 Seiten, 78 Bilder, 12 Tab.
2002. ISBN 3-87525-175-X.

Band 121: Lars Pursche
Methoden zur technologieorientierten Programmierung für die 3D-Lasermikrobearbeitung
LFT, 111 Seiten, 39 Bilder, 0 Tab.
2002. ISBN 3-87525-183-0.

Band 122: Jan-Oliver Brassel
Prozeßkontrolle beim Laserstrahl-Mikroschweißen
LFT, 148 Seiten, 72 Bilder, 12 Tab.
2002. ISBN 3-87525-181-4.

Band 123: Mark Geisel
Prozeßkontrolle und -steuerung beim Laserstrahlschweißen mit den Methoden der nichtlinearen Dynamik
LFT, 135 Seiten, 46 Bilder, 2 Tab.
2002. ISBN 3-87525-180-6.

Band 124: Gerd Eßer
Laserstrahlunterstützte Erzeugung metallischer Leiterstrukturen auf Thermoplastsubstraten für die MID-Technik
LFT, 148 Seiten, 60 Bilder, 6 Tab.
2002. ISBN 3-87525-171-7.

Band 125: Marc Fleckenstein
Qualität laserstrahl-gefügter Mikroverbindungen elektronischer Kontakte
LFT, 159 Seiten, 77 Bilder, 7 Tab.
2002. ISBN 3-87525-170-9.

Band 126: Stefan Kaufmann
Grundlegende Untersuchungen zum Nd:YAG- Laserstrahlfügen von Silizium für Komponenten der Optoelektronik
LFT, 159 Seiten, 100 Bilder, 6 Tab.
2002. ISBN 3-87525-172-5.

Band 127: Thomas Fröhlich
Simultanes Löten von Anschlußkontakten elektronischer Bauelemente mit Diodenlaserstrahlung
LFT, 143 Seiten, 75 Bilder, 6 Tab.
2002. ISBN 3-87525-186-5.

Band 128: Achim Hofmann
Erweiterung der Formgebungsgrenzen beim Umformen von Aluminiumwerkstoffen durch den Einsatz prozessangepasster Platinen
LFT, 113 Seiten, 58 Bilder, 4 Tab. 2002. ISBN 3-87525-182-2.

Band 129: Ingo Kriebitzsch
3 - D MID Technologie in der Automobilelektronik
FAPS, 129 Seiten, 102 Bilder, 10 Tab. 2002. ISBN 3-87525-169-5.

Band 130: Thomas Pohl
Fertigungsqualität und Umformbarkeit laserstrahlgeschweißter Formplatinen aus Aluminiumlegierungen
LFT, 133 Seiten, 93 Bilder, 12 Tab. 2002. ISBN 3-87525-173-3.

Band 131: Matthias Wenk
Entwicklung eines konfigurierbaren Steuerungssystems für die flexible Sensorführung von Industrierobotern
FAPS, 167 Seiten, 85 Bilder, 1 Tab. 2002. ISBN 3-87525-174-1.

Band 132: Matthias Negendanck
Neue Sensorik und Aktorik für Bearbeitungsköpfe zum Laserstrahlschweißen
LFT, 116 Seiten, 60 Bilder, 14 Tab. 2002. ISBN 3-87525-184-9.

Band 133: Oliver Kreis
Integrierte Fertigung - Verfahrensintegration durch Innenhochdruck-Umformen, Trennen und Laserstrahlschweißen in einem Werkzeug sowie ihre tele- und multimediale Präsentation
LFT, 167 Seiten, 90 Bilder, 43 Tab. 2002. ISBN 3-87525-176-8.

Band 134: Stefan Trautner
Technische Umsetzung produktbezogener Instrumente der Umweltpolitik bei Elektro- und Elektronikgeräten
FAPS, 179 Seiten, 92 Bilder, 11 Tab. 2002. ISBN 3-87525-177-6.

Band 135: Roland Meier
Strategien für einen produktorientierten Einsatz räumlicher spritzgegossener Schaltungsträger (3-D MID)
FAPS, 155 Seiten, 88 Bilder, 14 Tab. 2002. ISBN 3-87525-178-4.

Band 136: Jürgen Wunderlich
Kostensimulation - Simulationsbasierte Wirtschaftlichkeitsregelung komplexer Produktionssysteme
FAPS, 202 Seiten, 119 Bilder, 17 Tab. 2002. ISBN 3-87525-179-2.

Band 137: Stefan Novotny
Innenhochdruck-Umformen von Blechen aus Aluminium- und Magnesiumlegierungen bei erhöhter Temperatur
LFT, 132 Seiten, 82 Bilder, 6 Tab. 2002. ISBN 3-87525-185-7.

Band 138: Andreas Licha
Flexible Montageautomatisierung zur Komplettmontage flächenhafter Produktstrukturen durch kooperierende Industrieroboter
FAPS, 158 Seiten, 87 Bilder, 8 Tab. 2003. ISBN 3-87525-189-X.

Band 139: Michael Eisenbarth
Beitrag zur Optimierung der Aufbau- und Verbindungstechnik für mechatronische Baugruppen
FAPS, 207 Seiten, 141 Bilder, 9 Tab. 2003. ISBN 3-87525-190-3.

Band 140: Frank Christoph
Durchgängige simulationsgestützte Planung von Fertigungseinrichtungen der Elektronikproduktion
FAPS, 187 Seiten, 107 Bilder, 9 Tab. 2003. ISBN 3-87525-191-1.

Band 141: Hinnerk Hagenah
Simulationsbasierte Bestimmung der zu erwartenden Maßhaltigkeit für das Blechbiegen
LFT, 131 Seiten, 36 Bilder, 26 Tab. 2003. ISBN 3-87525-192-X.

Band 142: Ralf Eckstein
Scherschneiden und Biegen metallischer Kleinstteile - Materialeinfluss und Materialverhalten
LFT, 148 Seiten, 71 Bilder, 19 Tab. 2003. ISBN 3-87525-193-8.

Band 143: Frank H. Meyer-Pittroff
Excimerlaserstrahlbiegen dünner metallischer Folien mit homogener Lichtlinie
LFT, 138 Seiten, 60 Bilder, 16 Tab. 2003. ISBN 3-87525-196-2.

Band 144: Andreas Kach
Rechnergestützte Anpassung von Laserstrahlschneidbahnen an Bauteilabweichungen
LFT, 139 Seiten, 69 Bilder, 11 Tab. 2004. ISBN 3-87525-197-0.

Band 145: Stefan Hierl
System- und Prozeßtechnik für das simultane Löten mit Diodenlaserstrahlung von elektronischen Bauelementen
LFT, 124 Seiten, 66 Bilder, 4 Tab. 2004. ISBN 3-87525-198-9.

Band 146: Thomas Neudecker
Tribologische Eigenschaften keramischer Blechumformwerkzeuge-Einfluss einer Oberflächenendbearbeitung mittels Excimerlaserstrahlung
LFT, 166 Seiten, 75 Bilder, 26 Tab. 2004. ISBN 3-87525-200-4.

Band 147: Ulrich Wenger
Prozessoptimierung in der Wickeltechnik durch innovative maschinenbauliche und regelungstechnische Ansätze
FAPS, 132 Seiten, 88 Bilder, 0 Tab. 2004. ISBN 3-87525-203-9.

Band 148: Stefan Slama
Effizienzsteigerung in der Montage durch marktorientierte Montagestrukturen und erweiterte Mitarbeiterkompetenz
FAPS, 188 Seiten, 125 Bilder, 0 Tab. 2004. ISBN 3-87525-204-7.

Band 149: Thomas Wurm
Laserstrahljustieren mittels Aktoren-Entwicklung von Konzepten und Methoden für die rechnerunterstützte Modellierung und Optimierung von komplexen Aktorsystemen in der Mikrotechnik
LFT, 122 Seiten, 51 Bilder, 9 Tab. 2004. ISBN 3-87525-206-3.

Band 150: Martino Celeghini
Wirkmedienbasierte Blechumformung: Grundlagenuntersuchungen zum Einfluss von Werkstoff und Bauteilgeometrie
LFT, 146 Seiten, 77 Bilder, 6 Tab.
2004. ISBN 3-87525-207-1.

Band 151: Ralph Hohenstein
Entwurf hochdynamischer Sensor- und Regelsysteme für die adaptiveLaserbearbeitung
LFT, 282 Seiten, 63 Bilder, 16 Tab.
2004. ISBN 3-87525-210-1.

Band 152: Angelika Hutterer
Entwicklung prozessüberwachender Regelkreise für flexible Formgebungsprozesse
LFT, 149 Seiten, 57 Bilder, 2 Tab.
2005. ISBN 3-87525-212-8.

Band 153: Emil Egerer
Massivumformen metallischer Kleinstteile bei erhöhter Prozesstemperatur
LFT, 158 Seiten, 87 Bilder, 10 Tab.
2005. ISBN 3-87525-213-6.

Band 154: Rüdiger Holzmann
Strategien zur nachhaltigen Optimierung von Qualität und Zuverlässigkeit in der Fertigung hochintegrierter Flachbaugruppen
FAPS, 186 Seiten, 99 Bilder, 19 Tab.
2005. ISBN 3-87525-217-9.

Band 155: Marco Nock
Biegeumformen mit Elastomerwerkzeugen Modellierung, Prozessauslegung und Abgrenzung des Verfahrens am Beispiel des Rohrbiegens
LFT, 164 Seiten, 85 Bilder, 13 Tab.
2005. ISBN 3-87525-218-7.

Band 156: Frank Niebling
Qualifizierung einer Prozesskette zum Laserstrahlsintern metallischer Bauteile
LFT, 148 Seiten, 89 Bilder, 3 Tab.
2005. ISBN 3-87525-219-5.

Band 157: Markus Meiler
Großserientauglichkeit trockenschmierstoffbeschichteter Aluminiumbleche im Presswerk Grundlegende Untersuchungen zur Tribologie, zum Umformverhalten und Bauteilversuche
LFT, 104 Seiten, 57 Bilder, 21 Tab.
2005. ISBN 3-87525-221-7.

Band 158: Agus Sutanto
Solution Approaches for Planning of Assembly Systems in Three-Dimensional Virtual Environments
FAPS, 169 Seiten, 98 Bilder, 3 Tab.
2005. ISBN 3-87525-220-9.

Band 159: Matthias Boiger
Hochleistungssysteme für die Fertigung elektronischer Baugruppen auf der Basis flexibler Schaltungsträger
FAPS, 175 Seiten, 111 Bilder, 8 Tab.
2005. ISBN 3-87525-222-5.

Band 160: Matthias Pitz
Laserunterstütztes Biegen höchstfester Mehrphasenstähle
LFT, 120 Seiten, 73 Bilder, 11 Tab.
2005. ISBN 3-87525-223-3.

Band 161: Meik Vahl
Beitrag zur gezielten Beeinflussung des Werkstoffflusses beim Innenhochdruck-Umformen von Blechen
LFT, 165 Seiten, 94 Bilder, 15 Tab.
2005. ISBN 3-87525-224-1.

Band 162: Peter K. Kraus
Plattformstrategien - Realisierung einer varianz- und kostenoptimierten Wertschöpfung
FAPS, 181 Seiten, 95 Bilder, 0 Tab.
2005. ISBN 3-87525-226-8.

Band 163: Adrienn Cser
Laserstrahlschmelzabtrag - Prozessanalyse und -modellierung
LFT, 146 Seiten, 79 Bilder, 3 Tab.
2005. ISBN 3-87525-227-6.

Band 164: Markus C. Hahn
Grundlegende Untersuchungen zur Herstellung von Leichtbauverbundstrukturen mit Aluminiumschaumkern
LFT, 143 Seiten, 60 Bilder, 16 Tab.
2005. ISBN 3-87525-228-4.

Band 165: Gordana Michos
Mechatronische Ansätze zur Optimierung von Vorschubachsen
FAPS, 146 Seiten, 87 Bilder, 17 Tab.
2005. ISBN 3-87525-230-6.

Band 166: Markus Stark
Auslegung und Fertigung hochpräziser Faser-Kollimator-Arrays
LFT, 158 Seiten, 115 Bilder, 11 Tab.
2005. ISBN 3-87525-231-4.

Band 167: Yurong Zhou
Kollaboratives Engineering Management in der integrierten virtuellen Entwicklung der Anlagen für die Elektronikproduktion
FAPS, 156 Seiten, 84 Bilder, 6 Tab.
2005. ISBN 3-87525-232-2.

Band 168: Werner Enser
Neue Formen permanenter und lösbarer elektrischer Kontaktierungen für mechatronische Baugruppen
FAPS, 190 Seiten, 112 Bilder, 5 Tab.
2005. ISBN 3-87525-233-0.

Band 169: Katrin Melzer
Integrierte Produktpolitik bei elektrischen und elektronischen Geräten zur Optimierung des Product-Life-Cycle
FAPS, 155 Seiten, 91 Bilder, 17 Tab.
2005. ISBN 3-87525-234-9.

Band 170: Alexander Putz
Grundlegende Untersuchungen zur Erfassung der realen Vorspannung von armierten Kaltfließpresswerkzeugen mittels Ultraschall
LFT, 137 Seiten, 71 Bilder, 15 Tab.
2006. ISBN 3-87525-237-3.

Band 171: Martin Prechtl
Automatisiertes Schichtverfahren für metallische Folien - System- und Prozesstechnik
LFT, 154 Seiten, 45 Bilder, 7 Tab.
2006. ISBN 3-87525-238-1.

Band 172: Markus Meidert
Beitrag zur deterministischen Lebensdauerabschätzung von Werkzeugen der Kaltmassivumformung
LFT, 131 Seiten, 78 Bilder, 9 Tab.
2006. ISBN 3-87525-239-X.

Band 173: Bernd Müller
Robuste, automatisierte Montagesysteme durch adaptive Prozessführung und montageübergreifende Fehlerprävention am Beispiel flächiger Leichtbauteile
FAPS, 147 Seiten, 77 Bilder, 0 Tab.
2006. ISBN 3-87525-240-3.

Band 174: Alexander Hofmann
Hybrides Laserdurchstrahlschweißen von Kunststoffen
LFT, 136 Seiten, 72 Bilder, 4 Tab.
2006. ISBN 978-3-87525-243-9.

Band 175: Peter Wölflick
Innovative Substrate und Prozesse mit feinsten Strukturen für bleifreie Mechatronik-Anwendungen
FAPS, 177 Seiten, 148 Bilder, 24 Tab. 2006.
ISBN 978-3-87525-246-0.

Band 176: Attila Komlodi
Detection and Prevention of Hot Cracks during Laser Welding of Aluminium Alloys Using Advanced Simulation Methods
LFT, 155 Seiten, 89 Bilder, 14 Tab. 2006. ISBN 978-3-87525-248-4.

Band 177: Uwe Popp
Grundlegende Untersuchungen zum Laserstrahlstrukturieren von Kaltmassivumformwerkzeugen
LFT, 140 Seiten, 67 Bilder, 16 Tab. 2006. ISBN 978-3-87525-249-1.

Band 178: Veit Rückel
Rechnergestützte Ablaufplanung und Bahngenerierung Für kooperierende Industrieroboter
FAPS, 148 Seiten, 75 Bilder, 7 Tab. 2006. ISBN 978-3-87525-250-7.

Band 179: Manfred Dirscherl
Nicht-thermische Mikrojustiertechnik mittels ultrakurzer Laserpulse
LFT, 154 Seiten, 69 Bilder, 10 Tab. 2007. ISBN 978-3-87525-251-4.

Band 180: Yong Zhuo
Entwurf eines rechnergestützten integrierten Systems für Konstruktion und Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID)
FAPS, 181 Seiten, 95 Bilder, 5 Tab. 2007. ISBN 978-3-87525-253-8.

Band 181: Stefan Lang
Durchgängige Mitarbeiterinformation zur Steigerung von Effizienz und Prozesssicherheit in der Produktion
FAPS, 172 Seiten, 93 Bilder. 2007. ISBN 978-3-87525-257-6.

Band 182: Hans-Joachim Krauß
Laserstrahlinduzierte Pyrolyse präkeramischer Polymere
LFT, 171 Seiten, 100 Bilder. 2007. ISBN 978-3-87525-258-3.

Band 183: Stefan Junker
Technologien und Systemlösungen für die flexibel automatisierte Bestückung permanent erregter Läufer mit oberflächenmontierten Dauermagneten
FAPS, 173 Seiten, 75 Bilder. 2007. ISBN 978-3-87525-259-0.

Band 184: Rainer Kohlbauer
Wissensbasierte Methoden für die simulationsgestützte Auslegung wirkmedienbasierter Blechumformprozesse
LFT, 135 Seiten, 50 Bilder. 2007. ISBN 978-3-87525-260-6.

Band 185: Klaus Lamprecht
Wirkmedienbasierte Umformung tiefgezogener Vorformen unter besonderer Berücksichtigung maßgeschneiderter Halbzeuge
LFT, 137 Seiten, 81 Bilder. 2007. ISBN 978-3-87525-265-1.

Band 186: Bernd Zolleiß
Optimierte Prozesse und Systeme für die Bestückung mechatronischerBaugruppen
FAPS, 180 Seiten, 117 Bilder. 2007. ISBN 978-3-87525-266-8.

Band 187: Michael Kerausch
Simulationsgestützte Prozessauslegung für das Umformen lokal wärmebehandelter Aluminiumplatinen
LFT, 146 Seiten, 76 Bilder, 7 Tab. 2007. ISBN 978-3-87525-267-5.

Band 188: Matthias Weber
Unterstützung der Wandlungsfähigkeit von Produktionsanlagen durch innovative Softwaresysteme
FAPS, 183 Seiten, 122 Bilder, 3 Tab. 2007. ISBN 978-3-87525-269-9.

Band 189: Thomas Frick
Untersuchung der prozessbestimmenden Strahl-Stoff-Wechselwirkungen beim Laserstrahlschweißen von Kunststoffen
LFT, 104 Seiten, 62 Bilder, 8 Tab. 2007. ISBN 978-3-87525-268-2.

Band 190: Joachim Hecht
Werkstoffcharakterisierung und Prozessauslegung für die wirkmedienbasierte Doppelblech-Umformung von Magnesiumlegierungen
LFT, 107 Seiten, 91 Bilder, 2 Tab. 2007. ISBN 978-3-87525-270-5.

Band 191: Ralf Völkl
Stochastische Simulation zur Werkzeuglebensdaueroptimierung und Präzisionsfertigung in der Kaltmassivumformung
LFT, 178 Seiten, 75 Bilder, 12 Tab. 2008. ISBN 978-3-87525-272-9.

Band 192: Massimo Tolazzi
Innenhochdruck-Umformen verstärkter Blech-Rahmenstrukturen
LFT, 164 Seiten, 85 Bilder, 7 Tab. 2008. ISBN 978-3-87525-273-6.

Band 193: Cornelia Hoff
Untersuchung der Prozesseinflussgrößen beim Presshärten des höchstfesten Vergütungsstahls 22MnB5
LFT, 133 Seiten, 92 Bilder, 5 Tab. 2008. ISBN 978-3-87525-275-0.

Band 194: Christian Alvarez
Simulationsgestützte Methoden zur effizienten Gestaltung von Lötprozessen in der Elektronikproduktion
FAPS, 149 Seiten, 86 Bilder, 8 Tab. 2008. ISBN 978-3-87525-277-4.

Band 195: Andreas Kunze
Automatisierte Montage von makromechatronischen Modulen zur flexiblen Integration in hybride Pkw-Bordnetzsysteme
FAPS, 160 Seiten, 90 Bilder, 14 Tab. 2008. ISBN 978-3-87525-278-1.

Band 196: Wolfgang Hußnätter
Grundlegende Untersuchungen zur experimentellen Ermittlung und zur Modellierung von Fließortkurven bei erhöhten Temperaturen
LFT, 152 Seiten, 73 Bilder, 21 Tab. 2008. ISBN 978-3-87525-279-8.

Band 197: Thomas Bigl
Entwicklung, angepasste Herstellungsverfahren und erweiterte Qualitätssicherung von einsatzgerechten elektronischen Baugruppen
FAPS, 175 Seiten, 107 Bilder, 14 Tab. 2008. ISBN 978-3-87525-280-4.

Band 198: Stephan Roth
Grundlegende Untersuchungen zum Excimerlaserstrahl-Abtragen unter Flüssigkeitsfilmen
LFT, 113 Seiten, 47 Bilder, 14 Tab. 2008. ISBN 978-3-87525-281-1.

Band 199: Artur Giera
Prozesstechnische Untersuchungen zum Rührreibschweißen metallischer Werkstoffe
LFT, 179 Seiten, 104 Bilder, 36 Tab. 2008. ISBN 978-3-87525-282-8.

Band 200: Jürgen Lechler
Beschreibung und Modellierung des Werkstoffverhaltens von presshärtbaren Bor-Manganstählen
LFT, 154 Seiten, 75 Bilder, 12 Tab. 2009. ISBN 978-3-87525-286-6.

Band 201: Andreas Blankl
Untersuchungen zur Erhöhung der Prozessrobustheit bei der Innenhochdruck-Umformung von flächigen Halbzeugen mit vor- bzw. nachgeschalteten Laserstrahlfügeoperationen
LFT, 120 Seiten, 68 Bilder, 9 Tab. 2009. ISBN 978-3-87525-287-3.

Band 202: Andreas Schaller
Modellierung eines nachfrageorientierten Produktionskonzeptes für mobile Telekommunikationsgeräte
FAPS, 120 Seiten, 79 Bilder, 0 Tab. 2009. ISBN 978-3-87525-289-7.

Band 203: Claudius Schimpf
Optimierung von Zuverlässigkeitsuntersuchungen, Prüfabläufen und Nacharbeitsprozessen in der Elektronikproduktion
FAPS, 162 Seiten, 90 Bilder, 14 Tab. 2009. ISBN 978-3-87525-290-3.

Band 204: Simon Dietrich
Sensoriken zur Schwerpunktslagebestimmung der optischen Prozessemissionen beim Laserstrahltiefschweißen
LFT, 138 Seiten, 70 Bilder, 5 Tab. 2009. ISBN 978-3-87525-292-7.

Band 205: Wolfgang Wolf
Entwicklung eines agentenbasierten Steuerungssystems zur Materialflussorganisation im wandelbaren Produktionsumfeld
FAPS, 167 Seiten, 98 Bilder. 2009. ISBN 978-3-87525-293-4.

Band 206: Steffen Polster
Laserdurchstrahlschweißen transparenter Polymerbauteile
LFT, 160 Seiten, 92 Bilder, 13 Tab. 2009. ISBN 978-3-87525-294-1.

Band 207: Stephan Manuel Dörfler
Rührreibschweißen von walzplattiertem Halbzeug und Aluminiumblech zur Herstellung flächiger Aluminiumschaum-Sandwich-Verbundstrukturen
LFT, 190 Seiten, 98 Bilder, 5 Tab. 2009. ISBN 978-3-87525-295-8.

Band 208: Uwe Vogt
Seriennahe Auslegung von Aluminium Tailored Heat Treated Blanks
LFT, 151 Seiten, 68 Bilder, 26 Tab. 2009. ISBN 978-3-87525-296-5.

Band 209: Till Laumann
Qualitative und quantitative Bewertung der Crashtauglichkeit von höchstfesten Stählen
LFT, 117 Seiten, 69 Bilder, 7 Tab. 2009. ISBN 978-3-87525-299-6.

Band 210: Alexander Diehl
Größeneffekte bei Biegeprozessen-Entwicklung einer Methodik zur Identifikation und Quantifizierung
LFT, 180 Seiten, 92 Bilder, 12 Tab. 2010. ISBN 978-3-87525-302-3.

Band 211: Detlev Staud
Effiziente Prozesskettenauslegung für das Umformen lokal wärmebehandelter und geschweißter Aluminiumbleche
LFT, 164 Seiten, 72 Bilder, 12 Tab. 2010. ISBN 978-3-87525-303-0.

Band 212: Jens Ackermann
Prozesssicherung beim Laserdurchstrahlschweißen thermoplastischer Kunststoffe
LPT, 129 Seiten, 74 Bilder, 13 Tab. 2010. ISBN 978-3-87525-305-4.

Band 213: Stephan Weidel
Grundlegende Untersuchungen zum Kontaktzustand zwischen Werkstück und Werkzeug bei umformtechnischen Prozessen unter tribologischen Gesichtspunkten
LFT, 144 Seiten, 67 Bilder, 11 Tab. 2010. ISBN 978-3-87525-307-8.

Band 214: Stefan Geißdörfer
Entwicklung eines mesoskopischen Modells zur Abbildung von Größeneffekten in der Kaltmassivumformung mit Methoden der FE-Simulation
LFT, 133 Seiten, 83 Bilder, 11 Tab. 2010. ISBN 978-3-87525-308-5.

Band 215: Christian Matzner
Konzeption produktspezifischer Lösungen zur Robustheitssteigerung elektronischer Systeme gegen die Einwirkung von Betauung im Automobil
FAPS, 165 Seiten, 93 Bilder, 14 Tab. 2010. ISBN 978-3-87525-309-2.

Band 216: Florian Schüßler
Verbindungs- und Systemtechnik für thermisch hochbeanspruchte und miniaturisierte elektronische Baugruppen
FAPS, 184 Seiten, 93 Bilder, 18 Tab. 2010. ISBN 978-3-87525-310-8.

Band 217: Massimo Cojutti
Strategien zur Erweiterung der Prozessgrenzen bei der Innhochdruck-Umformung von Rohren und Blechpaaren
LFT, 125 Seiten, 56 Bilder, 9 Tab. 2010. ISBN 978-3-87525-312-2.

Band 218: Raoul Plettke
Mehrkriterielle Optimierung komplexer Aktorsysteme für das Laserstrahljustieren
LFT, 152 Seiten, 25 Bilder, 3 Tab. 2010. ISBN 978-3-87525-315-3.

Band 219: Andreas Dobroschke
Flexible Automatisierungslösungen für die Fertigung wickeltechnischer Produkte
FAPS, 184 Seiten, 109 Bilder, 18 Tab. 2011. ISBN 978-3-87525-317-7.

Band 220: Azhar Zam
Optical Tissue Differentiation for Sensor-Controlled Tissue-Specific Laser Surgery
LPT, 99 Seiten, 45 Bilder, 8 Tab. 2011. ISBN 978-3-87525-318-4.

Band 221: Michael Rösch
Potenziale und Strategien zur Optimierung des Schablonendruckprozesses in der Elektronikproduktion
FAPS, 192 Seiten, 127 Bilder, 19 Tab. 2011. ISBN 978-3-87525-319-1.

Band 222: Thomas Rechtenwald
Quasi-isothermes Laserstrahlsintern von Hochtemperatur-Thermoplasten - Eine Betrachtung werkstoff-prozessspezifischer Aspekte am Beispiel PEEK
LPT, 150 Seiten, 62 Bilder, 8 Tab. 2011. ISBN 978-3-87525-320-7.

Band 223: Daniel Craiovan
Prozesse und Systemlösungen für die SMT-Montage optischer Bauelemente auf Substrate mit integrierten Lichtwellenleitern
FAPS, 165 Seiten, 85 Bilder, 8 Tab. 2011. ISBN 978-3-87525-324-5.

Band 224: Kay Wagner
Beanspruchungsangepasste Kaltmassivumformwerkzeuge durch lokal optimierte Werkzeugoberflächen
LFT, 147 Seiten, 103 Bilder, 17 Tab. 2011. ISBN 978-3-87525-325-2.

Band 225: Martin Brandhuber
Verbesserung der Prognosegüte des Versagens von Punktschweißverbindungen bei höchstfesten Stahlgüten
LFT, 155 Seiten, 91 Bilder, 19 Tab. 2011. ISBN 978-3-87525-327-6.

Band 226: Peter Sebastian Feuser
Ein Ansatz zur Herstellung von pressgehärteten Karosseriekomponenten mit maßgeschneiderten mechanischen Eigenschaften: Temperierte Umformwerkzeuge. Prozessfenster, Prozesssimuation und funktionale Untersuchung
LFT, 195 Seiten, 97 Bilder, 60 Tab. 2012. ISBN 978-3-87525-328-3.

Band 227: Murat Arbak
Material Adapted Design of Cold Forging Tools Exemplified by Powder Metallurgical Tool Steels and Ceramics
LFT, 109 Seiten, 56 Bilder, 8 Tab. 2012. ISBN 978-3-87525-330-6.

Band 228: Indra Pitz
Beschleunigte Simulation des Laserstrahlumformens von Aluminiumblechen
LPT, 137 Seiten, 45 Bilder, 27 Tab. 2012. ISBN 978-3-87525-333-7.

Band 229: Alexander Grimm
Prozessanalyse und -überwachung des Laserstrahlhartlötens mittels optischer Sensorik
LPT, 125 Seiten, 61 Bilder, 5 Tab. 2012. ISBN 978-3-87525-334-4.

Band 230: Markus Kaupper
Biegen von höhenfesten Stahlblechwerkstoffen - Umformverhalten und Grenzen der Biegbarkeit
LFT, 160 Seiten, 57 Bilder, 10 Tab. 2012. ISBN 978-3-87525-339-9.

Band 231: Thomas Kroiß
Modellbasierte Prozessauslegung für die Kaltmassivumformung unter Brücksichtigung der Werkzeug- und Pressenauffederung
LFT, 169 Seiten, 50 Bilder, 19 Tab. 2012. ISBN 978-3-87525-341-2.

Band 232: Christian Goth
Analyse und Optimierung der Entwicklung und Zuverlässigkeit räumlicher Schaltungsträger (3D-MID)
FAPS, 176 Seiten, 102 Bilder, 22 Tab. 2012. ISBN 978-3-87525-340-5.

Band 233: Christian Ziegler
Ganzheitliche Automatisierung mechatronischer Systeme in der Medizin am Beispiel Strahlentherapie
FAPS, 170 Seiten, 71 Bilder, 19 Tab. 2012. ISBN 978-3-87525-342-9.

Band 234: Florian Albert
Automatisiertes Laserstrahllöten und -reparaturlöten elektronischer Baugruppen
LPT, 127 Seiten, 78 Bilder, 11 Tab. 2012. ISBN 978-3-87525-344-3.

Band 235: Thomas Stöhr
Analyse und Beschreibung des mechanischen Werkstoffverhaltens von presshärtbaren Bor-Manganstählen
LFT, 118 Seiten, 74 Bilder, 18 Tab. 2013. ISBN 978-3-87525-346-7.

Band 236: Christian Kägeler
Prozessdynamik beim Laserstrahlschweißen verzinkter Stahlbleche im Überlappstoß
LPT, 145 Seiten, 80 Bilder, 3 Tab. 2013. ISBN 978-3-87525-347-4.

Band 237: Andreas Sulzberger
Seriennahe Auslegung der Prozesskette zur wärmeunterstützten Umformung von Aluminiumblechwerkstoffen
LFT, 153 Seiten, 87 Bilder, 17 Tab. 2013. ISBN 978-3-87525-349-8.

Band 238: Simon Opel
Herstellung prozessangepasster Halbzeuge mit variabler Blechdicke durch die Anwendung von Verfahren der Blechmassivumformung
LFT, 165 Seiten, 108 Bilder, 27 Tab. 2013. ISBN 978-3-87525-350-4.

Band 239: Rajesh Kanawade
In-vivo Monitoring of Epithelium Vessel and Capillary Density for the Application of Detection of Clinical Shock and Early Signs of Cancer Development
LPT, 124 Seiten, 58 Bilder, 15 Tab. 2013. ISBN 978-3-87525-351-1.

Band 240: Stephan Busse
Entwicklung und Qualifizierung eines Schneidclinchverfahrens
LFT, 119 Seiten, 86 Bilder, 20 Tab. 2013. ISBN 978-3-87525-352-8.

Band 241: Karl-Heinz Leitz
Mikro- und Nanostrukturierung mit kurz und ultrakurz gepulster Laserstrahlung
LPT, 154 Seiten, 71 Bilder, 9 Tab. 2013. ISBN 978-3-87525-355-9.

Band 242: Markus Michl
Webbasierte Ansätze zur ganzheitlichen technischen Diagnose
FAPS, 182 Seiten, 62 Bilder, 20 Tab. 2013. ISBN 978-3-87525-356-6.

Band 243: Vera Sturm
Einfluss von Chargenschwankungen auf die Verarbeitungsgrenzen von Stahlwerkstoffen
LFT, 113 Seiten, 58 Bilder, 9 Tab. 2013. ISBN 978-3-87525-357-3.

Band 244: Christian Neudel
Mikrostrukturelle und mechanisch-technologische Eigenschaften widerstandspunktgeschweißter Aluminium-Stahl-Verbindungen für den Fahrzeugbau
LFT, 178 Seiten, 171 Bilder, 31 Tab. 2014. ISBN 978-3-87525-358-0.

Band 245: Anja Neumann
Konzept zur Beherrschung der Prozessschwankungen im Presswerk
LFT, 162 Seiten, 68 Bilder, 15 Tab. 2014. ISBN 978-3-87525-360-3.

Band 246: Ulf-Hermann Quentin
Laserbasierte Nanostrukturierung mit optisch positionierten Mikrolinsen
LPT, 137 Seiten, 89 Bilder, 6 Tab. 2014. ISBN 978-3-87525-361-0.

Band 247: Erik Lamprecht
Der Einfluss der Fertigungsverfahren auf die Wirbelstromverluste von Stator-Einzelzahnblechpaketen für den Einsatz in Hybrid- und Elektrofahrzeugen
FAPS, 148 Seiten, 138 Bilder, 4 Tab. 2014. ISBN 978-3-87525-362-7.

Band 248: Sebastian Rösel
Wirkmedienbasierte Umformung von Blechhalbzeugen unter Anwendung magnetorheologischer Flüssigkeiten als kombiniertes Wirk- und Dichtmedium
LFT, 148 Seiten, 61 Bilder, 12 Tab. 2014. ISBN 978-3-87525-363-4.

Band 249: Paul Hippchen
Simulative Prognose der Geometrie indirekt pressgehärteter Karosseriebauteile für die industrielle Anwendung
LFT, 163 Seiten, 89 Bilder, 12 Tab. 2014. ISBN 978-3-87525-364-1.

Band 250: Martin Zubeil
Versagensprognose bei der Prozesssimulation von Biegeumform- und Falzverfahren
LFT, 171 Seiten, 90 Bilder, 5 Tab. 2014. ISBN 978-3-87525-365-8.

Band 251: Alexander Kühl
Flexible Automatisierung der Statorenmontage mit Hilfe einer universellen ambidexteren Kinematik
FAPS, 142 Seiten, 60 Bilder, 26 Tab. 2014. ISBN 978-3-87525-367-2.

Band 252: Thomas Albrecht
Optimierte Fertigungstechnologien für Rotoren getriebeintegrierter PM-Synchronmotoren von Hybridfahrzeugen
FAPS, 198 Seiten, 130 Bilder, 38 Tab. 2014. ISBN 978-3-87525-368-9.

Band 253: Florian Risch
Planning and Production Concepts for Contactless Power Transfer Systems for Electric Vehicles
FAPS, 185 Seiten, 125 Bilder, 13 Tab. 2014. ISBN 978-3-87525-369-6.

Band 254: Markus Weigl
Laserstrahlschweißen von Mischverbindungen aus austenitischen und ferritischen korrosionsbeständigen Stahlwerkstoffen
LPT, 184 Seiten, 110 Bilder, 6 Tab. 2014. ISBN 978-3-87525-370-2.

Band 255: Johannes Noneder
Beanspruchungserfassung für die Validierung von FE-Modellen zur Auslegung von Massivumformwerkzeugen
LFT, 161 Seiten, 65 Bilder, 14 Tab. 2014. ISBN 978-3-87525-371-9.

Band 256: Andreas Reinhardt
Ressourceneffiziente Prozess- und Produktionstechnologie für flexible Schaltungsträger
FAPS, 123 Seiten, 69 Bilder, 19 Tab. 2014. ISBN 978-3-87525-373-3.

Band 257: Tobias Schmuck
Ein Beitrag zur effizienten Gestaltung globaler Produktions- und Logistiknetzwerke mittels Simulation
FAPS, 151 Seiten, 74 Bilder. 2014. ISBN 978-3-87525-374-0.

Band 258: Bernd Eichenhüller
Untersuchungen der Effekte und Wechselwirkungen charakteristischer Einflussgrößen auf das Umformverhalten bei Mikroumformprozessen
LFT, 127 Seiten, 29 Bilder, 9 Tab. 2014. ISBN 978-3-87525-375-7.

Band 259: Felix Lütteke
Vielseitiges autonomes Transportsystem basierend auf Weltmodellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren
FAPS, 152 Seiten, 54 Bilder, 20 Tab. 2014. ISBN 978-3-87525-376-4.

Band 260: Martin Grüner
Hochdruck-Blechumformung mit formlos festen Stoffen als Wirkmedium
LFT, 144 Seiten, 66 Bilder, 29 Tab. 2014. ISBN 978-3-87525-379-5.

Band 261: Christian Brock
Analyse und Regelung des Laserstrahltiefschweißprozesses durch Detektion der Metalldampffackelposition
LPT, 126 Seiten, 65 Bilder, 3 Tab. 2015. ISBN 978-3-87525-380-1.

Band 262: Peter Vatter
Sensitivitätsanalyse des 3-Rollen-Schubbiegens auf Basis der Finite Elemente Methode
LFT, 145 Seiten, 57 Bilder, 26 Tab. 2015. ISBN 978-3-87525-381-8.

Band 263: Florian Klämpfl
Planung von Laserbestrahlungen durch simulationsbasierte Optimierung
LPT, 169 Seiten, 78 Bilder, 32 Tab. 2015. ISBN 978-3-87525-384-9.

Band 264: Matthias Domke
Transiente physikalische Mechanismen bei der Laserablation von dünnen Metallschichten
LPT, 133 Seiten, 43 Bilder, 3 Tab. 2015. ISBN 978-3-87525-385-6.

Band 265: Johannes Götz
Community-basierte Optimierung des Anlagenengineerings
FAPS, 177 Seiten, 80 Bilder, 30 Tab.
2015. ISBN 978-3-87525-386-3.

Band 266: Hung Nguyen
Qualifizierung des Potentials von Verfestigungseffekten zur Erweiterung des Umformvermögens aushärtbarer Aluminiumlegierungen
LFT, 137 Seiten, 57 Bilder, 16 Tab.
2015. ISBN 978-3-87525-387-0.

Band 267: Andreas Kuppert
Erweiterung und Verbesserung von Versuchs- und Auswertetechniken für die Bestimmung von Grenzformänderungskurven
LFT, 138 Seiten, 82 Bilder, 2 Tab.
2015. ISBN 978-3-87525-388-7.

Band 268: Kathleen Klaus
Erstellung eines Werkstofforientierten Fertigungsprozessfensters zur Steigerung des Formgebungsvermögens von Alumi-niumlegierungen unter Anwendung einer zwischengeschalteten Wärmebehandlung
LFT, 154 Seiten, 70 Bilder, 8 Tab.
2015. ISBN 978-3-87525-391-7.

Band 269: Thomas Svec
Untersuchungen zur Herstellung von funktionsoptimierten Bauteilen im partiellen Presshärtprozess mittels lokal unterschiedlich temperierter Werkzeuge
LFT, 166 Seiten, 87 Bilder, 15 Tab.
2015. ISBN 978-3-87525-392-4.

Band 270: Tobias Schrader
Grundlegende Untersuchungen zur Verschleißcharakterisierung beschichteter Kaltmassivumformwerkzeuge
LFT, 164 Seiten, 55 Bilder, 11 Tab.
2015. ISBN 978-3-87525-393-1.

Band 271: Matthäus Brela
Untersuchung von Magnetfeld-Messmethoden zur ganzheitlichen Wertschöpfungsoptimierung und Fehlerdetektion an magnetischen Aktoren
FAPS, 170 Seiten, 97 Bilder, 4 Tab.
2015. ISBN 978-3-87525-394-8.

Band 272: Michael Wieland
Entwicklung einer Methode zur Prognose adhäsiven Verschleißes an Werkzeugen für das direkte Presshärten
LFT, 156 Seiten, 84 Bilder, 9 Tab.
2015. ISBN 978-3-87525-395-5.

Band 273: René Schramm
Strukturierte additive Metallisierung durch kaltaktives Atmosphärendruckplasma
FAPS, 136 Seiten, 62 Bilder, 15 Tab.
2015. ISBN 978-3-87525-396-2.

Band 274: Michael Lechner
Herstellung beanspruchungsangepasster Aluminiumblechhalbzeuge durch eine maßgeschneiderte Variation der Abkühlgeschwindigkeit nach Lösungsglühen
LFT, 136 Seiten, 62 Bilder, 15 Tab.
2015. ISBN 978-3-87525-397-9.

Band 275: Kolja Andreas
Einfluss der Oberflächenbeschaffenheit auf das Werkzeugeinsatzverhalten beim Kaltfließpressen
LFT, 169 Seiten, 76 Bilder, 4 Tab.
2015. ISBN 978-3-87525-398-6.

Band 276: Marcus Baum
Laser Consolidation of ITO Nanoparticles for the Generation of Thin Conductive Layers on Transparent Substrates
LPT, 158 Seiten, 75 Bilder, 3 Tab.
2015. ISBN 978-3-87525-399-3.

Band 277: Thomas Schneider
Umformtechnische Herstellung dünnwandiger Funktionsbauteile aus Feinblech durch Verfahren der Blechmassivumformung
LFT, 188 Seiten, 95 Bilder, 7 Tab.
2015. ISBN 978-3-87525-401-3.

Band 278: Jochen Merhof
Sematische Modellierung automatisierter Produktionssysteme zur Verbesserung der IT-Integration zwischen Anlagen-Engineering und Steuerungsebene
FAPS, 157 Seiten, 88 Bilder, 8 Tab.
2015. ISBN 978-3-87525-402-0.

Band 279: Fabian Zöller
Erarbeitung von Grundlagen zur Abbildung des tribologischen Systems in der Umformsimulation
LFT, 126 Seiten, 51 Bilder, 3 Tab.
2016. ISBN 978-3-87525-403-7.

Band 280: Christian Hezler
Einsatz technologischer Versuche zur Erweiterung der Versagensvorhersage bei Karosseriebauteilen aus höchstfesten Stählen
LFT, 147 Seiten, 63 Bilder, 44 Tab.
2016. ISBN 978-3-87525-404-4.

Band 281: Jochen Bönig
Integration des Systemverhaltens von Automobil-Hochvoltleitungen in die virtuelle Absicherung durch strukturmechanische Simulation
FAPS, 177 Seiten, 107 Bilder, 17 Tab.
2016. ISBN 978-3-87525-405-1.

Band 282: Johannes Kohl
Automatisierte Datenerfassung für diskret ereignisorientierte Simulationen in der energieflexibelen Fabrik
FAPS, 160 Seiten, 80 Bilder, 27 Tab.
2016. ISBN 978-3-87525-406-8.

Band 283: Peter Bechtold
Mikroschockwellenumformung mittels ultrakurzer Laserpulse
LPT, 155 Seiten, 59 Bilder, 10 Tab.
2016. ISBN 978-3-87525-407-5.

Band 284: Stefan Berger
Laserstrahlschweißen thermoplastischer Kohlenstofffaserverbundwerkstoffe mit spezifischem Zusatzdraht
LPT, 118 Seiten, 68 Bilder, 9 Tab.
2016. ISBN 978-3-87525-408-2.

Band 285: Martin Bornschlegl
Methods-Energy Measurement - Eine Methode zur Energieplanung für Fügeverfahren im Karosseriebau
FAPS, 136 Seiten, 72 Bilder, 46 Tab.
2016. ISBN 978-3-87525-409-9.

Band 286: Tobias Rackow
Erweiterung des Unternehmenscontrollings um die Dimension Energie
FAPS, 164 Seiten, 82 Bilder, 29 Tab.
2016. ISBN 978-3-87525-410-5.

Band 287: Johannes Koch
Grundlegende Untersuchungen zur Herstellung zyklisch-symmetrischer Bauteile mit Nebenformelementen durch Blechmassivumformung
LFT, 125 Seiten, 49 Bilder, 17 Tab.
2016. ISBN 978-3-87525-411-2.

Band 288: Hans Ulrich Vierzigmann
Beitrag zur Untersuchung der tribologischen Bedingungen in der Blechmassivumformung - Bereitstellung von tribologischen Modellversuchen und Realisierung von Tailored Surfaces
LFT, 174 Seiten, 102 Bilder, 34 Tab. 2016. ISBN 978-3-87525-412-9.

Band 289: Thomas Senner
Methodik zur virtuellen Absicherung der formgebenden Operation des Nasspressprozesses von Gelege-Mehrschichtverbunden
LFT, 156 Seiten, 96 Bilder, 21 Tab. 2016. ISBN 978-3-87525-414-3.

Band 290: Sven Kreitlein
Der grundoperationsspezifische Mindestenergiebedarf als Referenzwert zur Bewertung der Energieeffizienz in der Produktion
FAPS, 185 Seiten, 64 Bilder, 30 Tab. 2016. ISBN 978-3-87525-415-0.

Band 291: Christian Roos
Remote-Laserstrahlschweißen verzinkter Stahlbleche in Kehlnahtgeometrie
LPT, 123 Seiten, 52 Bilder, 0 Tab. 2016. ISBN 978-3-87525-416-7.

Band 292: Alexander Kahrimanidis
Thermisch unterstützte Umformung von Aluminiumblechen
LFT, 165 Seiten, 103 Bilder, 18 Tab. 2016. ISBN 978-3-87525-417-4.

Band 293: Jan Tremel
Flexible Systems for Permanent Magnet Assembly and Magnetic Rotor Measurement / Flexible Systeme zur Montage von Permanentmagneten und zur Messung magnetischer Rotoren
FAPS, 152 Seiten, 91 Bilder, 12 Tab. 2016. ISBN 978-3-87525-419-8.

Band 294: Ioannis Tsoupis
Schädigungs- und Versagensverhalten hochfester Leichtbauwerkstoffe unter Biegebeanspruchung
LFT, 176 Seiten, 51 Bilder, 6 Tab. 2017. ISBN 978-3-87525-420-4.

Band 295: Sven Hildering
Grundlegende Untersuchungen zum Prozessverhalten von Silizium als Werkzeugwerkstoff für das Mikroscherschneiden metallischer Folien
LFT, 177 Seiten, 74 Bilder, 17 Tab. 2017. ISBN 978-3-87525-422-8.

Band 296: Sasia Mareike Hertweck
Zeitliche Pulsformung in der Lasermikromaterialbearbeitung – Grundlegende Untersuchungen und Anwendungen
LPT, 146 Seiten, 67 Bilder, 5 Tab. 2017. ISBN 978-3-87525-423-5.

Band 297: Paryanto
Mechatronic Simulation Approach for the Process Planning of Energy-Efficient Handling Systems
FAPS, 162 Seiten, 86 Bilder, 13 Tab. 2017. ISBN 978-3-87525-424-2.

Band 298: Peer Stenzel
Großserientaugliche Nadelwickeltechnik für verteilte Wicklungen im Anwendungsfall der E-Traktionsantriebe
FAPS, 239 Seiten, 147 Bilder, 20 Tab. 2017. ISBN 978-3-87525-425-9.

Band 299: Mario Lušić
Ein Vorgehensmodell zur Erstellung montageführender Werkerinformationssysteme simultan zum Produktentstehungsprozess
FAPS, 174 Seiten, 79 Bilder, 22 Tab. 2017. ISBN 978-3-87525-426-6.

Band 300: Arnd Buschhaus
Hochpräzise adaptive Steuerung und Regelung robotergeführter Prozesse
FAPS, 202 Seiten, 96 Bilder, 4 Tab. 2017. ISBN 978-3-87525-427-3.

Band 301: Tobias Laumer
Erzeugung von thermoplastischen Werkstoffverbunden mittels simultanem, intensitätsselektivem Laserstrahlschmelzen
LPT, 140 Seiten, 82 Bilder, 0 Tab. 2017. ISBN 978-3-87525-428-0.

Band 302: Nora Unger
Untersuchung einer thermisch unterstützten Fertigungskette zur Herstellung umgeformter Bauteile aus der höherfesten Aluminiumlegierung EN AW-7020
LFT, 142 Seiten, 53 Bilder, 8 Tab. 2017. ISBN 978-3-87525-429-7.

Band 303: Tommaso Stellin
Design of Manufacturing Processes for the Cold Bulk Forming of Small Metal Components from Metal Strip
LFT, 146 Seiten, 67 Bilder, 7 Tab. 2017. ISBN 978-3-87525-430-3.

Band 304: Bassim Bachy
Experimental Investigation, Modeling, Simulation and Optimization of Molded Interconnect Devices (MID) Based on Laser Direct Structuring (LDS) / Experimentelle Untersuchung, Modellierung, Simulation und Optimierung von Molded Interconnect Devices (MID) basierend auf Laser Direktstrukturierung (LDS)
FAPS, 168 Seiten, 120 Bilder, 26 Tab. 2017. ISBN 978-3-87525-431-0.

Band 305: Michael Spahr
Automatisierte Kontaktierungsverfahren für flachleiterbasierte Pkw-Bordnetzsysteme
FAPS, 197 Seiten, 98 Bilder, 17 Tab. 2017. ISBN 978-3-87525-432-7.

Band 306: Sebastian Suttner
Charakterisierung und Modellierung des spannungszustandsabhängigen Werkstoffverhaltens der Magnesiumlegierung AZ31B für die numerische Prozessauslegung
LFT, 150 Seiten, 84 Bilder, 19 Tab. 2017. ISBN 978-3-87525-433-4.

Band 307: Bhargav Potdar
A reliable methodology to deduce thermo-mechanical flow behaviour of hot stamping steels
LFT, 203 Seiten, 98 Bilder, 27 Tab. 2017. ISBN 978-3-87525-436-5.

Band 308: Maria Löffler
Steuerung von Blechmassivumformprozessen durch maßgeschneiderte tribologische Systeme
LFT, viii u. 166 Seiten, 90 Bilder, 5 Tab. 2018. ISBN 978-3-96147-133-1.

Band 309: Martin Müller
Untersuchung des kombinierten Trenn- und Umformprozesses beim Fügen artungleicher Werkstoffe mittels Schneidclinchverfahren
LFT, xi u. 149 Seiten, 89 Bilder, 6 Tab. 2018. ISBN: 978-3-96147-135-5.

Band 310: Christopher Kästle
Qualifizierung der Kupfer-Drahtbondtechnologie für integrierte Leistungsmodule in harschen Umgebungsbedingungen
FAPS, xii u. 167 Seiten, 70 Bilder, 18 Tab. 2018. ISBN 978-3-96147-145-4.

Band 311: Daniel Vipavc
Eine Simulationsmethode für das 3-Rollen-Schubbiegen
LFT, xiii u. 121 Seiten, 56 Bilder, 17 Tab. 2018. ISBN 978-3-96147-147-8.

Band 312: Christina Ramer
Arbeitsraumüberwachung und autonome Bahnplanung für ein sicheres und flexibles Roboter-Assistenzsystem in der Fertigung
FAPS, xiv u. 188 Seiten, 57 Bilder, 9 Tab. 2018. ISBN 978-3-96147-153-9.

Band 313: Miriam Rauer
Der Einfluss von Poren auf die Zuverlässigkeit der Lötverbindungen von Hochleistungs-Leuchtdioden
FAPS, xii u. 209 Seiten, 108 Bilder, 21 Tab. 2018.
ISBN 978-3-96147-157-7.

Band 314: Felix Tenner
Kamerabasierte Untersuchungen der Schmelze und Gasströmungen beim Laserstrahlschweißen verzinkter Stahlbleche
LPT, xxiii u. 184 Seiten, 94 Bilder, 7 Tab. 2018. ISBN 978-3-96147-160-7.

Band 315: Aarief Syed-Khaja
Diffusion Soldering for High-temperature Packaging of Power Electronics
FAPS, x u. 202 Seiten, 144 Bilder, 32 Tab. 2018. ISBN 978-3-87525-162-1.

Band 316: Adam Schaub
Grundlagenwissenschaftliche Untersuchung der kombinierten Prozesskette aus Umformen und Additive Fertigung
LFT, xi u. 192 Seiten, 72 Bilder, 27 Tab. 2019. ISBN 978-3-96147-166-9.

Band 317: Daniel Gröbel
Herstellung von Nebenformelementen unterschiedlicher Geometrie an Blechen mittels Fließpressverfahren der Blechmassivumformung
LFT, x u. 165 Seiten, 96 Bilder, 13 Tab. 2019. ISBN 978-3-96147-168-3.

Band 318: Philipp Hildenbrand
Entwicklung einer Methodik zur Herstellung von Tailored Blanks mit definierten Halbzeugeigenschaften durch einen Taumelprozess
LFT, ix u. 153 Seiten, 77 Bilder, 4 Tab. 2019. ISBN 978-3-96147-174-4.

Band 319: Tobias Konrad
Simulative Auslegung der Spann- und Fixierkonzepte im Karosseriebau: Bewertung der Baugruppenmaßhaltigkeit unter Berücksichtigung schwankender Einflussgrößen
LFT, x u. 203 Seiten, 134 Bilder, 32 Tab. 2019. ISBN 978-3-96147-176-8.

Band 320: David Meinel
Architektur applikationsspezifischer Multi-Physics-Simulationskonfiguratoren am Beispiel modularer Triebzüge
FAPS, xii u. 166 Seiten, 82 Bilder, 25 Tab. 2019.
ISBN 978-3-96147-184-3.

Band 321: Andrea Zimmermann
Grundlegende Untersuchungen zum Einfluss fertigungsbedingter Eigenschaften auf die Ermüdungsfestigkeit kaltmassivumgeformter Bauteile
LFT, ix u. 160 Seiten, 66 Bilder, 5 Tab. 2019. ISBN 978-3-96147-190-4.

Band 322: Christoph Amann
Simulative Prognose der Geometrie nassgepresster Karosseriebauteile aus Gelege-Mehrschichtverbunden
LFT, xvi u. 169 Seiten, 80 Bilder, 13 Tab. 2019. ISBN 978-3-96147-194-2.

Band 323: Jennifer Tenner
Realisierung schmierstofffreier Tiefziehprozesse durch maßgeschneiderte Werkzeugoberflächen
LFT, x u. 187 Seiten, 68 Bilder, 13 Tab. 2019. ISBN 978-3-96147-196-6.

Band 324: Susan Zöller
Mapping Individual Subjective Values to Product Design
KTmfk, xi u. 223 Seiten, 81 Bilder, 25 Tab. 2019.
ISBN 978-3-96147-202-4.

Band 325: Stefan Lutz
Erarbeitung einer Methodik zur semiempirischen Ermittlung der Umwandlungskinetik durchhärtender Wälzlagerstähle für die Wärmebehandlungssimulation
LFT, xiv u. 189 Seiten, 75 Bilder, 32 Tab. 2019. ISBN 978-3-96147-209-3.

Band 326: Tobias Gnibl
Modellbasierte Prozesskettenabbildung rührreibgeschweißter Aluminiumhalbzeuge zur umformtechnischen Herstellung höchstfester Leichtbau-strukturteile
LFT, xii u. 167 Seiten, 68 Bilder, 17 Tab. 2019. ISBN 978-3-96147-217-8.

Band 327: Johannes Bürner
Technisch-wirtschaftliche Optionen zur Lastflexibilisierung durch intelligente elektrische Wärmespeicher
FAPS, xiv u. 233 Seiten, 89 Bilder, 27 Tab. 2019.
ISBN 978-3-96147-219-2.

Band 328: Wolfgang Böhm
Verbesserung des Umformverhaltens von mehrlagigen Aluminiumblechwerkstoffen mit ultrafeinkörnigem Gefüge
LFT, ix u. 160 Seiten, 88 Bilder, 14 Tab. 2019. ISBN 978-3-96147-227-7.

Band 329: Stefan Landkammer
Grundsatzuntersuchungen, mathematische Modellierung und Ableitung einer Auslegungsmethodik für Gelenkantriebe nach dem Spinnenbeinprinzip
LFT, xii u. 200 Seiten, 83 Bilder, 13 Tab. 2019. ISBN 978-3-96147-229-1.

Band 330: Stephan Rapp
Pump-Probe-Ellipsometrie zur Messung transienter optischer Materialeigen-schaften bei der Ultrakurzpuls-Lasermaterialbearbeitung
LPT, xi u. 143 Seiten, 49 Bilder, 2 Tab. 2019. ISBN 978-3-96147-235-2.

Band 331: Michael Scholz
Intralogistics Execution System mit integrierten autonomen, servicebasierten Transportentitäten
FAPS, xi u. 195 Seiten, 55 Bilder, 11 Tab. 2019. ISBN 978-3-96147-237-6.

Band 332: Eva Bogner
Strategien der Produktindividualisierung in der produzierenden Industrie im Kontext der Digitalisierung
FAPS, ix u. 201 Seiten, 55 Bilder, 28 Tab. 2019. ISBN 978-3-96147-246-8.

Band 333: Daniel Benjamin Krüger
Ein Ansatz zur CAD-integrierten muskuloskelettalen Analyse der Mensch-Maschine-Interaktion
KTmfk, x u. 217 Seiten, 102 Bilder, 7 Tab. 2019. ISBN 978-3-96147-250-5.

Band 334: Thomas Kuhn
Qualität und Zuverlässigkeit laserdirektstrukturierter mechatronisch integrierter Baugruppen (LDS-MID)
FAPS, ix u. 152 Seiten, 69 Bilder, 12 Tab. 2019. ISBN 978-3-96147-252-9.

Band 335: Hans Fleischmann
Modellbasierte Zustands- und Prozessüberwachung auf Basis sozio-cyber-physischer Systeme
FAPS, xi u. 214 Seiten, 111 Bilder, 18 Tab. 2019. ISBN 978-3-96147-256-7.

Band 336: Markus Michalski
Grundlegende Untersuchungen zum Prozess- und Werkstoffverhalten bei schwingungsüberlagerter Umformung
LFT, xii u. 197 Seiten, 93 Bilder, 11 Tab. 2019. ISBN 978-3-96147-270-3.

Band 337: Markus Brandmeier
Ganzheitliches ontologiebasiertes Wissensmanagement im Umfeld der industriellen Produktion
FAPS, xi u. 255 Seiten, 77 Bilder, 33 Tab. 2020.
ISBN 978-3-96147-275-8.

Band 338: Stephan Purr
Datenerfassung für die Anwendung lernender Algorithmen bei der Herstellung von Blechformteilen
LFT, ix u. 165 Seiten, 48 Bilder, 4 Tab. 2020. ISBN 978-3-96147-281-9.

Band 339: Christoph Kiener
Kaltfließpressen von gerad- und schrägverzahnten Zahnrädern
LFT, viii u. 151 Seiten, 81 Bilder, 3 Tab. 2020. ISBN 978-3-96147-287-1.

Band 340: Simon Spreng
Numerische, analytische und empirische Modellierung des Heißcrimpprozesses
FAPS, xix u. 204 Seiten, 91 Bilder, 27 Tab. 2020.
ISBN 978-3-96147-293-2.

Band 341: Patrik Schwingenschlögl
Erarbeitung eines Prozessverständnisses zur Verbesserung der tribologischen Bedingungen beim Presshärten
LFT, x u. 177 Seiten, 81 Bilder, 8 Tab. 2020.
ISBN 978-3-96147-297-0.

Band 342: Emanuela Affronti
Evaluation of failure behaviour of sheet metals
LFT, ix u. 136 Seiten, 57 Bilder, 20 Tab. 2020.
ISBN 978-3-96147-303-8.

Band 343: Julia Degner
Grundlegende Untersuchungen zur Herstellung hochfester Aluminiumblechbauteile in einem kombinierten Umform- und Abschreckprozess
LFT, x u. 172 Seiten, 61 Bilder, 9 Tab. 2020.
ISBN 978-3-96147-307-6.

Band 344: Maximilian Wagner
Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen
FAPS, xxi u. 181 Seiten, 111 Bilder, 15 Tab. 2020.
ISBN 978-3-96147-309-0.

Band 345: Stefan Härter
Qualifizierung des Montageprozesses hochminiaturisierter elektronischer Bauelemente
FAPS, ix u. 194 Seiten, 97 Bilder, 28 Tab. 2020. ISBN 978-3-96147-314-4.

Band 346: Toni Donhauser
Ressourcenorientierte Auftragsregelung in einer hybriden Produktion mittels betriebsbegleitender Simulation
FAPS, xix u. 242 Seiten, 97 Bilder, 17 Tab. 2020.
ISBN 978-3-96147-316-8.

Band 347: Philipp Amend
Laserbasiertes Schmelzkleben von Thermoplasten mit Metallen
LPT, xv u. 154 Seiten, 67 Bilder. 2020. ISBN 978-3-96147-326-7.

Band 348: Matthias Ehlert
Simulationsunterstützte funktionale Grenzlagenabsicherung
KTmfk, xvi u. 300 Seiten, 101 Bilder, 73 Tab. 2020.
ISBN 978-3-96147-328-1.

Band 349: Thomas Sander
Ein Beitrag zur Charakterisierung und Auslegung des Verbundes von Kunststoffsubstraten mit harten Dünnschichten
KTmfk, xiv u. 178 Seiten, 88 Bilder, 21 Tab. 2020.
ISBN 978-3-96147-330-4.

Band 350: Florian Pilz
Fließpressen von Verzahnungselementen an Blechen
LFT, x u. 170 Seiten, 103Bilder, 4 Tab. 2020. ISBN 978-3-96147-332-8.

Band 351: Sebastian Josef Katona
Evaluation und Aufbereitung von Produktsimulationen mittels abweichungsbehafteter Geometriemodelle
KTmfk, ix u. 147 Seiten, 73 Bilder, 11 Tab. 2020.
ISBN 978-3-96147-336-6.

Band 352: Jürgen Herrmann
Kumulatives Walzplattieren. Bewertung der Umformeigenschaften mehrlagiger Blechwerkstoffe der ausscheidungshärtbaren Legierung AA6014
LFT, x u. 157 Seiten, 64 Bilder, 5 Tab. 2020. ISBN 978-3-96147-344-1.

Band 353: Christof Küstner
Assistenzsystem zur Unterstützung der datengetriebenen Produktentwicklung
KTmfk, xii u. 219 Seiten, 63 Bilder, 14 Tab. 2020.
ISBN 978-3-96147-348-9.

Band 354: Tobias Gläßel
Prozessketten zum Laserstrahlschweißen von flachleiterbasierten Formspulenwicklungen für automobile Traktionsantriebe
FAPS, xiv u. 206 Seiten, 89 Bilder, 11 Tab. 2020.
ISBN 978-3-96147-356-4.

Band 355: Andreas Meinel
Experimentelle Untersuchung der Auswirkungen von Axialschwingungen auf Reibung und Verschleiß in Zylinderrol-lenlagern
KTmfk, xii u. 162 Seiten, 56 Bilder, 7 Tab. 2020.
ISBN 978-3-96147-358-8.

Band 356: Hannah Riedle
Haptische, generische Modelle weicher anatomischer Strukturen für die chirurgische Simulation
FAPS, xxx u. 179 Seiten, 82 Bilder, 35 Tab. 2020.
ISBN 978-3-96147-367-0.

Band 357: Maximilian Landgraf
Leistungselektronik für den Einsatz dielektrischer Elastomere in aktorischen, sensorischen und integrierten sensomotorischen Systemen
FAPS, xxiii u. 166 Seiten, 71 Bilder, 10 Tab. 2020.
ISBN 978-3-96147-380-9.

Band 358: Alireza Esfandyari
Multi-Objective Process Optimization for Overpressure Reflow Soldering in Electronics Production
FAPS, xviii u. 175 Seiten, 57 Bilder, 23 Tab. 2020.
ISBN 978-3-96147-382-3.

Band 359: Christian Sand
Prozessübergreifende Analyse komplexer Montageprozessketten mittels Data Mining
FAPS, XV u. 168 Seiten, 61 Bilder, 12 Tab. 2021.
ISBN 978-3-96147-398-4.

Band 360: Ralf Merkl
Closed-Loop Control of a Storage-Supported Hybrid Compensation System for Improving the Power Quality in Medium Voltage Networks
FAPS, xxvii u. 200 Seiten, 102 Bilder, 2 Tab. 2021.
ISBN 978-3-96147-402-8.

Band 361: Thomas Reitberger
Additive Fertigung polymerer optischer Wellenleiter im Aerosol-Jet-Verfahren
FAPS, xix u. 141 Seiten, 65 Bilder, 11 Tab. 2021. ISBN 978-3-96147-400-4.

Band 362: Marius Christian Fechter
Modellierung von Vorentwürfen in der virtuellen Realität mit natürlicher Fingerinteraktion
KTmfk, x u. 188 Seiten, 67 Bilder, 19 Tab. 2021.
ISBN 978-3-96147-404-2.

Band 363: Franziska Neubauer
Oberflächenmodifizierung und Entwicklung einer Auswertemethodik zur Verschleißcharakterisierung im Presshärteprozess
LFT, ix u. 177 Seiten, 42 Bilder, 6 Tab. 2021. ISBN 978-3-96147-406-6.

Band 364: Eike Wolfram Schäffer
Web- und wissensbasierter Engineering-Konfigurator für roboterzentrierte Automatisierungslösungen
FAPS, xxiv u. 195 Seiten, 108 Bilder, 25 Tab. 2021.
ISBN 978-3-96147-410-3.

Band 365: Daniel Gross
Untersuchungen zur kohlenstoffdioxidbasierten kryogenen Minimalmengenschmierung
REP, xii u. 184 Seiten, 56 Bilder, 18 Tab. 2021. ISBN 978-3-96147-412-7.

Band 366: Daniel Junker
Qualifizierung laser-additiv gefertigter Komponenten für den Einsatz im Werkzeugbau der Massivumformung
LFT, vii u. 142 Seiten, 62 Bilder, 5 Tab. 2021. ISBN 978-3-96147-416-5.

Band 367: Tallal Javied
Totally Integrated Ecology Management for Resource Efficient and Eco-Friendly Production
FAPS, xv u. 160 Seiten, 60 Bilder, 13 Tab. 2021. ISBN 978-3-96147-418-9.

Band 368: David Marco Hochrein
Wälzlager im Beschleunigungsfeld – Eine Analysestrategie zur Bestimmung des Reibungs-, Axialschub- und Temperaturverhaltens von Nadelkränzen –
KTmfk, xiii u. 279 Seiten, 108 Bilder, 39 Tab. 2021.
ISBN 978-3-96147-420-2.

Band 369: Daniel Gräf
Funktionalisierung technischer Oberflächen mittels prozessüberwachter aerosolbasierter Drucktechnologie
FAPS, xxii u. 175 Seiten, 97 Bilder, 6 Tab. 2021.
ISBN 978-3-96147-433-2.

Band 370: Andreas Gröschl
Hochfrequent fokusabstandsmodulierte Konfokalsensoren für die Nanokoordinatenmesstechnik
FMT, x u. 144 Seiten, 98 Bilder, 6 Tab. 2021. ISBN 978-3-96147-435-6.

Band 371: Johann Tüchsen
Konzeption, Entwicklung und Einführung des Assistenzsystems D-DAS für die Produktentwicklung elektrischer Motoren
KTmfk, xii u. 178 Seiten, 92 Bilder, 12 Tab. 2021.
ISBN 978-3-96147-437-0.

Band 372: Max Marian
Numerische Auslegung von Oberflächenmikrotexturen für geschmierte tribologische Kontakte
KTmfk, xviii u. 276 Seiten, 85 Bilder, 45 Tab. 2021.
ISBN 978-3-96147-439-4.

Band 373: Johannes Strauß
Die akustooptische Strahlformung in der Lasermaterialbearbeitung
LPT, xvi u. 113 Seiten, 48 Bilder. 2021. ISBN 978-3-96147-441-7.

Band 374: Martin Hohmann
Machine learning and hyper spectral imaging: Multi Spectral Endoscopy in the Gastro Intestinal Tract towards Hyper Spectral Endoscopy
LPT, x u. 137 Seiten, 62 Bilder, 29 Tab. 2021. ISBN 978-3-96147-445-5.

Band 375: Timo Kordaß
Lasergestütztes Verfahren zur selektiven Metallisierung von epoxidharzbasierten Duromeren zur Steigerung der Integrationsdichte für dreidimensionale mechatronische Package-Baugruppen
FAPS, xviii u. 198 Seiten, 92 Bilder, 24 Tab. 2021.
ISBN 978-3-96147-443-1.

Band 376: Philipp Kestel
Assistenzsystem für den wissensbasierten Aufbau konstruktionsbegleitender Finite-Elemente-Analysen
KTmfk, xviii u. 209 Seiten, 57 Bilder, 17 Tab. 2021.
ISBN 978-3-96147-457-8.

Band 377: Martin Lerchen
Messverfahren für die pulverbettbasierte additive Fertigung zur Sicherstellung der Konformität mit geometrischen Produktspezifikationen
FMT, x u. 150 Seiten, 60 Bilder, 9 Tab. 2021. ISBN 978-3-96147-463-9.

Band 378: Michael Schneider
Inline-Prüfung der Permeabilität in weichmagnetischen Komponenten
FAPS, xxii u. 189 Seiten, 79 Bilder, 14 Tab. 2021.
ISBN 978-3-96147-465-3.

Band 379: Tobias Sprügel
Sphärische Detektorflächen als Unterstützung der Produktentwicklung zur Datenanalyse im Rahmen des Digital Engineering
KTmfk, xiii u. 213 Seiten, 84 Bilder, 33 Tab. 2021.
ISBN 978-3-96147-475-2.

Band 380: Tom Häfner
Multipulseffekte beim Mikro-Materialabtrag von Stahllegierungen mit Pikosekunden-Laserpulsen
LPT, xxviii u. 159 Seiten, 57 Bilder, 13 Tab. 2021.
ISBN 978-3-96147-479-0.

Band 381: Björn Heling
Einsatz und Validierung virtueller Absicherungsmethoden für abweichungs-behaftete Mechanismen im Kontext des Robust Design
KTmfk, xi u. 169 Seiten, 63 Bilder, 27 Tab. 2021.
ISBN 978-3-96147-487-5.

Band 382: Tobias Kolb
Laserstrahl-Schmelzen von Metallen mit einer Serienanlage – Prozesscharakterisierung und Erweiterung eines Überwachungssystems
LPT, xv u. 170 Seiten, 128 Bilder, 16 Tab. 2021. ISBN 978-3-96147-491-2.

Band 383: Mario Meinhardt
Widerstandselementschweißen mit gestauchten Hilfsfügeelementen - Umformtechnische Wirkzusammenhänge zur Beeinflussung der Verbindungsfestigkeit
LFT, xii u. 189 Seiten, 87 Bilder, 4 Tab. 2022. ISBN 978-3-96147-473-8.

Band 384: Felix Bauer
Ein Beitrag zur digitalen Auslegung von Fügeprozessen im Karosseriebau mit Fokus auf das Remote-Laserstrahlschweißen unter Einsatz flexibler Spanntechnik
LFT, xi u. 185 Seiten, 74 Bilder, 12 Tab. 2022. ISBN 978-3-96147-498-1.

Band 385: Jochen Zeitler
Konzeption eines rechnergestützten Konstruktionssystems für optomechatronische Baugruppen
FAPS, xix u. 172 Seiten, 88 Bilder, 11 Tab. 2022.
ISBN 978-3-96147-499-8.

Band 386: Vincent Mann
Einfluss von Strahloszillation auf das Laserstrahlschweißen hochfester Stähle
LPT, xiii u. 172 Seiten, 103 Bilder, 18 Tab. 2022. ISBN 978-3-96147-503-2.

Band 387: Chen Chen
Skin-equivalent opto-/elastofluidic in-vitro microphysiological vascular models for translational studies of optical biopsies
LPT, xx u. 126 Seiten, 60 Bilder, 10 Tab. 2022.
ISBN 978-3-96147-505-6.

Band 388: Stefan Stein
Laser drop on demand joining as bonding method for electronics assembly and packaging with high thermal requirements
LPT, x u. 112 Seiten, 54 Bilder, 10 Tab. 2022. ISBN 978-3-96147-507-0

Band 389: Nikolaus Urban
Untersuchung des Laserstrahlschmelzens von Neodym-Eisen-Bor zur additiven Herstellung von Permanentmagneten
FAPS, x u. 174 Seiten, 88 Bilder, 18 Tab. 2022. ISBN 978-3-96147-501-8.

Band 390: Yiting Wu
Großflächige Topographiemessungen mit einem Weißlichtinterferenzmikroskop und einem metrologischen Rasterkraftmikroskop
FMT, xii u. 142 Seiten, 68 Bilder, 11 Tab. 2022. ISBN: 978-3-96147-513-1.

Band 391: Thomas Papke
Untersuchungen zur Umformbarkeit hybrider Bauteile aus Blechgrundkörper und additiv gefertigter Struktur
LFT, xii u. 194 Seiten, 71 Bilder, 16 Tab. 2022. ISBN 978-3-96147-515-5.

Band 392: Bastian Zimmermann
Einfluss des Vormaterials auf die mehrstufige Kaltumformung vom Draht
LFT, xi u. 182 Seiten, 36 Bilder, 6 Tab. 2022. ISBN 978-3-96147-519-3.

Band 393: Harald Völkl
Ein simulationsbasierter Ansatz zur Auslegung additiv gefertigter FLM-Faserverbundstrukturen
KTmfk, xx u. 204 Seiten, 95 Bilder, 22 Tab. 2022.
ISBN 978-3-96147-523-0.

Band 394: Robert Schulte
Auslegung und Anwendung prozessangepasster Halbzeuge für Verfahren der Blechmassivumformung
LFT, x u. 163 Seiten, 93 Bilder, 5 Tab. 2022. ISBN 978-3-96147-525-4.

Band 395: Philipp Frey
Umformtechnische Strukturierung metallischer Einleger im Folgeverbund für mediendichte Kunststoff-Metall-Hybridbauteile
LFT, ix u. 180 Seiten, 83 Bilder, 7 Tab. 2022. ISBN 978-3-96147-534-6.

Band 396: Thomas Johann Luft
Komplexitätsmanagement in der Produktentwicklung - Holistische Modellierung, Analyse, Visualisierung und Bewertung komplexer Systeme
KTmfk, xiii u. 510 Seiten, 166 Bilder, 16 Tab. 2022.
ISBN 978-3-96147-540-7.

Band 397: Li Wang
Evaluierung der Einsetzbarkeit des lasergestützten Verfahrens zur selektiven Metallisierung für die Verbesserung passiver Intermodulation in Hochfrequenzanwendungen
FAPS, xxii u.151 Seiten, 72 Bilder, 22 Tab. 2022. ISBN 978-3-96147-542-1.

Band 398: Sebastian Reitelshöfer
Der Aerosol-Jet-Druck Dielektrischer Elastomere als additives Fertigungsverfahren für elastische mechatronische Komponenten
FAPS, xxv u. 206 Seiten, 87 Bilder, 13 Tab. 2022.
ISBN 978-3-96147-547-6.

Band 399: Alexander Meyer
Selektive Magnetmontage zur Verringerung des Rastmomentes permanenterregter Synchronmotoren
FAPS, xv u. 164 Seiten, 90 Bilder, 18 Tab. 2022. ISBN 978-3-96147-555-1.

Band 400: Rong Zhao
Design verschleißreduzierender amorpher Kohlenstoffschichtsysteme für trockene tribologische Gleitkontakte
KTmfk, x u. 148 Seiten, 69 Bilder, 14 Tab. 2022.
ISBN 978-3-96147-557-5.

Band 401: Christian P. J. Schwarzer
Kupfersintern als Fügetechnologie für Leistungselektronik
FAPS, xxvii u. 234 Seiten, 125 Bilder, 24 Tab. 2022.
ISBN 978-3-96147-566-7.

Band 402: Alexander Horn
Grundlegende Untersuchungen zur Gradierung der mechanischen Eigenschaften pressgehärteter Bauteile durch eine örtlich begrenzte Aufkohlung
LFT, xii u. 204 Seiten, 58 Bilder, 6 Tab. 2022. ISBN 978-3-96147-568-1.

Band 403: Artur Klos
Werkstoff- und umformtechnische Bewertung von hochfesten Aluminiumblechwerkstoffen für den Karosseriebau
LFT, x u. 192 Seiten, 73 Bilder, 12 Tab. 2022. ISBN 978-3-96147-572-8.

Band 404: Harald Schmid
Ganzheitliche Erarbeitung eines Prozessverständnisses von Tiefziehprozessen mit Ziehsicken auf Basis mechanischer und tribologischer Analysen
LFT, xiii u. 211 Seiten, 78 Bilder, 5 Tab. 2022. ISBN 978-3-96147-577-3.

Band 405: Johannes Henneberg
Blechmassivumformung von Funktionsbauteilen aus Bandmaterial
LFT, viii u. 176 Seiten, 101 Bilder, 2 Tab. 2022. ISBN 978-3-96147-579-7.

Band 406: Anton Schmailzl
Festigkeits- und zeitoptimierte Prozessführung beim quasi-simultanen Laser-Durchstrahlschweißen
LPT, xiii u. 157 Seiten, 84 Bilder, 7 Tab. 2022. ISBN 978-3-96147-583-4.

Band 407: Alexander Wolf
Modellierung und Vorhersage menschlichen Interaktionsverhaltens zur Analyse der Mensch-Produkt Interaktion
KTmfk, x u. 207 Seiten, 69 Bilder, 10 Tab. 2022.
ISBN 978-3-96147-585-8.

Band 408: Tim Weikert
Modifikationen amorpher Kohlenstoffschichten zur Anpassung der Reibungsbedingungen und zur Erhöhung des Verschleißschutzes
KTmfk, xvii u. 258 Seiten, 91 Bilder, 9 Tab. 2022.
ISBN 978-3-96147-589-6.

Band 409: Stefan Götz
Frühzeitiges konstruktionsbegleitendes Toleranzmanagement
KTmfk, ix u. 276 Seiten, 127 Bilder, 13 Tab. 2022.
ISBN 978-3-96147-593-3.

Band 410: Markus Hubert
Einsatzpotenziale der Rotationsschneidtechnologie in der Verarbeitung von metallischen Funktionsfolien für mechatronische Produkte
FAPS, xviii u. 139 Seiten, 86 Bilder, 7 Tab. 2022.
ISBN 978-3-96147-603-9.

Band 411: Manfred Vogel
Grundlagenuntersuchungen und Erarbeitung einer Methodik zur Herstellung maßgeschneiderter Halbzeuge auf Basis eines neuartigen flexiblen Walzprozesses
LFT, ix u. 176 Seiten, 61 Bilder, 11 Tab. 2022. ISBN 978-3-96147-605-3.

Band 412: Michael Weigelt
Multidimensionale Optionenanalyse alternativer Antriebskonzepte für die individuelle Langstreckenmobilität
FAPS, xv u. 222 Seiten, 89 Bilder, 38 Tab. 2022. ISBN 978-3-96147-607-7.

Band 413: Frank Bodendorf
Machine Learning im Cost Engineering des Supply Managements
FAPS, xiii u. 165 Seiten, 75 Bilder, 13 Tab. 2023.
ISBN 978-3-96147-609-1.

Band 414: Maximilian Metzner
Planung und Simulation taktiler, intelligenter und kollaborativer Roboterfähigkeiten in der Montage
FAPS, xix u. 174 Seiten, 72 Bilder, 3 Tab. 2023.
ISBN 978-3-96147-611-4.

Band 415: Tina Buker
Ein Ansatz zur Reduktion produktinduzierter Nutzerstigmatisierung durch Förderung einer gleichermaßen gebrauchstauglichen wie emotionalen Produktgestalt
KTmfk, x u. 236 Seiten, 54 Bilder, 44 Tab. 2022.
ISBN 978-3-96147-613-8.

Band 416: Marlene Kuhn
Model-based Traceability System Development for Complex Manufacturing Applying Blockchain and Graphs
FAPS, xv u. 167 Seiten, 63 Bilder, 10 Tab. 2022. ISBN 978-3-96147-615-2.

Band 417: Benjamin Lengenfelder
Remote photoacoustic sensing using speckle-analysis for biomedical imaging
LPT, xv u. 124 Seiten, 86 Bilder, 10 Tab. 2023. ISBN 978-3-96147-617-6.

Band 418: Benjamin Pohrer
Analyse des Zusammenhangs zwischen dem tribochemischen Aufbau von Grenzschichten und der Ausbildung von White Etching Crack-Schäden
KTmfk, xv u. 258 Seiten, 103 Bilder, 10 Tab. 2023. ISBN 978-3-96147-621-3.

Band 419: Matthias Friedlein
Zuverlässigkeitsmethoden zur Beschleunigung von Qualifizierungsuntersuchungen für Steckkontakte
FAPS, xxv u. 162 Seiten, 98 Bilder, 7 Tab. 2023.
ISBN 978-3-96147-625-1.

Band 420: Thomas Stoll
Laser Powder Bed Fusion von Kupfer auf Aluminiumoxid-Keramiken
FAPS, xxvii u. 236 Seiten, 103 Bilder, 11 Tab. 2023. ISBN 978-3-96147-631-2.

Band 421: Eric Eschner
Relation of Particle Motion and Process Zone Formation as a Basis for Sensing Approaches within PBF-LB/M
LPT, xiv u. 143 Seiten, 87 Bilder, 0 Tab. 2023.
ISBN 978-3-96147-633-6.

Band 422: Fanuel Mehari
Laser-induced Breakdown Spectroscopy (LIBS) as a diagnostics tool for biological tissue analysis.
LPT, xv u. 145 Seiten, 68 Bilder, 12 Tab. 2023. ISBN 978-3-96147-641-1.

Band 423: Uwe Leicht
Ultraschallüberlagertes Umformen und Verstemmen von Stahlwerkstoffen
LFT, xi u. 165 Seiten, 65 Bilder, 6 Tab. 2023. ISBN 978-3-96147-643-5.

Band 424: Thomas Braun
Potenzialanalyse der plasmabasierten, strukturierten Metallisierung thermoaktiver Oberflächen im industriellen Hausbau
FAPS, xvii u. 152 Seiten, 72 Bilder, 11 Tab. 2023. ISBN 978-3-96147-653-4.

Band 425: Reinhardt Seidel
Modellbasierte Optimierung des Selektivwellenlötprozesses
FAPS, xxii u. 167 Seiten, 73 Bilder, 23 Tab. 2023.
ISBN 978-3-96147-651-0.

Band 426: Matthias Lenzen
Maßgeschneiderte Werkstoffcharakterisierung für die numerische Auslegung von Blechumformprozessen
LFT, xi u. 187 Seiten, 77 Bilder, 13 Tab. 2023. ISBN 978-3-96147-663-3.

Band 427: Matthias Graser
Analyse lokaler Kurzzeitwärmebehandlungsmethoden zur Verbesserung des Umformverhaltens und der Bauteileigenschaften von Aluminiumstrangpresshohlprofilen
LFT, xi u. 169 Seiten, 81 Bilder, 1 Tab. 2023.
ISBN 978-3-96147-666-4.

Band 428: Markus Lieret
Sicheres autonomes Flugrobotersystem für den Einsatz im Produktions- und Logistikumfeld
FAPS, xix u. 198 Seiten, 54 Bilder, 7 Tab. 2023.
ISBN 978-3-96147-668-8.

Band 429: Petar Vukovic
Simulation komplexer Kommunikationssysteme in der Fertigungsautomatisierung
FAPS, xiv u. 163 Seiten, 57 Bilder, 21 Tab. 2023. ISBN 978-3-96147-673-2.

Band 430: Fabian Knieps
Finite Elemente Simulation dünnster Verpackungsstähle: Entwicklung einer geeigneten Charakterisierungs- und Validierungsstrategie
LFT, xix, 189 Seiten, 122 Bilder, 17 Tab. 2023. ISBN 978-3-96147-689-3

Band 431: Julian Seßner
Multimodale Bildsegmentierung gering strukturierter Umgebungen für die Navigation am Beispiel eines Assistenzsystems für sehbeeinträchtigte Personen
FAPS, xxv, 203 Seiten, 57 Bilder, 25 Tab. 2023. ISBN 978-3-96147-697-8

Band 432: Benjamin Samuel Lutz
Smart Manufacturing System for Process Optimization Regarding Deviations among Material Batches
FAPS, xix, 208 Seiten, 77 Bilder, 14 Tab. 2023. ISBN 978-3-96147-703-6

Band 433: Michael Jüttner
Bewertung von Kantenpressungen auf Basis von Simulationen mehrfach überrollter elasto-plastischer Kontakte
KTmfk, xii, 162 Seiten, 59 Bilder, 7 Tab. 2024.
ISBN 978-3-96147-713-5.

Band 434: Sebastian Wiesenmayer
Untersuchungen zur Stoffflusssteuerung beim Fügen durch Umformen von hochfesten Aluminiumlegierungen mittels lokaler Kurzzeitwärmebehandlung
LFT, xii u. 197 Seiten, 81 Bilder, 19 Tab. 2024. ISBN 978-3-96147-715-9.

Band 435: Clara-Maria Kuball
Grundlegende Untersuchungen zur umformtechnischen Herstellung von Halbhohlstanznieten aus hochverfestigenden Werkstoffen
LFT, viii u. 180 Seiten, 64 Bilder, 13 Tab. 2024.
ISBN 978-3-96147-717-3.

Band 436: Martin Roth
Sampling-based Tolerance-Cost Optimization: The Key to Optimal Tolerance Allocation
KTmfk, xxxvii u. 337 Seiten, 97 Bilder, 56 Tab. 2024.
ISBN 978-3-96147-719-7.

Band 437: Stephan Schirdewahn
Verbesserung des tribologischen Einsatzverhaltens im Presshärteprozess durch Verwendung maßgeschneiderter laserimplantierter Werkzeuge
LFT, viii u. 177 Seiten, 63 Bilder, 7 Tab. 2024. ISBN 978-3-96147-721-0.

Band 438: Andreas Rohrmoser
Erarbeitung eines grundlegenden Verständnisses zum Fließpressen betriebsangepasster Verzahnungen für den Einsatz in der Materialpaarung Metall-Kunststoff
LFT, x u. 166 Seiten, 94 Bilder, 6 Tab. 2024. ISBN 978-3-96147-723-4.

Band 439: Andreas Selmaier
DMAICS-Zyklus zur Digitalisierung in produzierenden Unternehmen
FAPS, xv u. 185 Seiten, 73 Bilder, 19 Tab. 2024. ISBN 978-3-96147-733-3.

Band 440: Thomas Kistner
Entwicklung von Modellen der Oberflächenform für die Messunsicherheitsbestimmung von taktilen Koordinatenmessungen durch Simulation
FMT, xiii u. 134 Seiten, 64 Bilder, 9 Tab. 2024.
ISBN 978-3-96147-735-7.

Band 441: Fabian Dworschak
Selbstverstärkendes Lernen als Beitrag zur Automatisierung der Anpassungskonstruktion
KTmfk, x, 205 Seiten, 87 Bilder, 16 Tab. 2024. ISBN 978-3-96147-739-5.

Band 442: Martin Killmann
Vorspannung und Ermüdung von Kaltmassivumformwerkzeugen für unrunde Bauteilgeometrien
LFT, ix u. 160 Seiten, 85 Bilder, 6 Tab. 2024. ISBN 978-3-96147-741-8.

Band 443: Ann-Kathrin Wiemann
Beitrag zur Messunsicherheitsermittlung für die Messung großer Zahnräder zur Erweiterung des virtuellen Koordinatenmessgerätes der Physikalisch-Technischen Bundesanstalt
FMT, xxiv u. 133 Seiten, 41 Bilder, 22 Tab. 2024.
ISBN 978-3-96147-747-0.

Band 444: Andreas Jobst
Eigenspannungen beim Voll-Vorwärts-Fließpressen – Entstehung, Einstellung und Stabilität
LFT, ix u. 162 Seiten, 89 Bilder, 5 Tab. 2024.
ISBN 978-3-96147-749-4.

Band 445: Martin Anton Kraus
Grundlagenwissenschaftliche Untersuchungen zum Kaltfließpressen kleinster metallischer Pin-Strukturen aus der Blechebene
LFT, x u. 191 Seiten, 64 Bilder, 10 Tab. 2024.
ISBN 978-3-96147-757-9.

Band 446: Moritz Späth
Assessment of microcirculation by shifted position-diffuse reflectance imaging (SP-DRI)
LPT, xi u. 109 Seiten, 37 Bilder, 8 Tab. 2024. ISBN 978-3-96147-759-3.

Band 447: Benedikt Göddeke
Methode zur Auswahl eines alternativen Blechwerkstoffes für flexibel gewalzte und direkt warmumgeformte Karosseriestrukturbauteile
LFT, xvi u. 222 Seiten, 66 Bilder, 27 Tab. 2024. ISBN 978-3-96147-762-3.

Band 448: Jan Hafenecker
Systematische Untersuchung zur Blechumformung von Hybridbauteilen
LFT, xxi u. 173 Seiten, 78 Bilder, 6 Tab. 2024.
ISBN 978-3-96147-764-7.

Band 449: Sven Wirsching
Numerische Optimierung von sekundären, geschmierten Punkt- und Ellipsenkontakten zur gezielten Auslegung von Rollenlagern
KTmfk, xxv u. 206 Seiten, 70 Bilder, 16 Tab. 2024.
ISBN 978-3-96147-766-1.

Band 450: Tobias Lechler
Der Digitale Zwilling im Ramp-down automatisierter Produktionssysteme
FAPS, xvii u. 152 Seiten, 83 Bilder, 4 Tab. 2024.
ISBN 978-3-96147-691-6.

Band 451: Eva Russwurm
Kombinierte Material- und Energieflusssimulation gleichstrombasierter, automatisierter Produktionsanlagen
FAPS, xvi u. 165 Seiten, 67 Bilder, 13 Tab. 2024.
ISBN 978-3-96147-770-8.

Band 452: Janik Schaude
Untersuchungen zur rasterkraft- und konfokalmikroskopischen Charakterisierung nanometrologischer Referenzkörper
FMT, x u. 167 Seiten, 74 Bilder, 4 Tab. 2024.
ISBN 978-3-96147-776-0.

Band 453: Philipp Bräuer
Nahinfrarot-Laser-Strukturierung und nasschemische Metallisierung von Aluminiumoxid mittels Laser-Absorptionsschicht
FAPS, xx u. 152 Seiten, 55 Bilder, 19 Tab. 2024.
ISBN 978-3-96147-780-7.

Band 454: Andreas Michael Müller
Definition and applications of the spatially discrete metrological descriptor framework for triangle mesh geometry data in 3D coordinate metrology
FMT, x u. 220 Seiten, 75 Bilder, 16 Tab. 2024.
ISBN 978-3-96147-782-1.

Band 455: Sina Martin
Einstellung anisotroper Eigenschaften dielektrischer Elastomersysteme mittels eines selektiven Druckverfahrens
FAPS, xxiii u. 166 Seiten, 76 Bilder, 6 Tab. 2024.
ISBN 978-3-96147-789-0.

Band 456: Manuel Reck
Grundlegendes Prozessverständnis zur Herstellung von Bauteilen mit texturierten Oberflächen durch Mikroprägen in einem kombinierten Umformprozess
LFT, ix u. 170 Seiten, 83 Bilder, 10 Tab. 2025.
ISBN 978-3-96147-798-2.

Band 457: Simone Neermann
Untersuchung photonischer Sinterverfahren und Entwicklung eines neuen elektrischen Messverfahrens zur Qualifizierung der Sinterung gedruckter Elektronik
FAPS, xix u. 145 Seiten, 75 Bilder, 6 Tab. 2025. ISBN 978-3-96147-800-2.

Band 458: Christian Witzgall
Berücksichtigung vorausgegangener Ermüdungsschädigung in der Crashsimulation kurzfaserverstärkter Thermoplaste
KTmfk, xxiv u. 201 Seiten, 109 Bilder, 19 Tab. 2025.
ISBN 978-3-96147-802-6.

Band 459: Lorenz Butzhammer
3-D-Scantrajektorien für die dimensionelle Röntgen-Computertomographie durch adaptive Messobjektneigung
FMT, xvi u. 212 Seiten, 81 Bilder, 2 Tab. 2025.
ISBN 978-3-96147-812-5.

Band 460: Benjamin Gerschütz
Identifikation von Anwendungsfällen und Integration datengetriebener Methoden im Produktentwicklungsprozess
KTmfk, xi u. 209 Seiten, 56 Bilder, 17 Tab. 2025.
ISBN 978-3-96147-814-9.

Band 461: Andreas Winkler
Numerische Verschleißmodellierung von trocken-, misch- und grenzreibungsbehafteten Wälzkontakten
KTmfk, xx u. 263 Seiten, 118 Bilder, 30 Tab. 2025.
ISBN 978-3-96147-816-3

Band 462: Barbara Hansen
Methode zur Berücksichtigung von Schwankungen der tribologischen Halbzeugeigenschaften für die umformtechnische Herstellung von Karosseriebauteilen im Serienprozess
LFT, x u. 163 Seiten, 64 Bilder, 13 Tab. 2025.
ISBN 978-3-96147-820-0

Band 463: Sebastian Hörlin
Integration von Kurzzeitspeichern zur Erhöhung der Energieeffizienz von elektrischen Antriebssträngen
FAPS, xxiii u. 155 Seiten, 78 Bilder, 38 Tab. 2025.
ISBN 978-3-96147-824-8

Band 464: Matthias Müller
Potenziale und Perspektiven eines Ringschmiersystems zur Schmierstoffver-sorgung von Wälz- und Gleitlagern
KTmfk, xiii u. 237 Seiten, 90 Bilder, 27 Tab. 2025.
ISBN 978-3-96147-826-2

Band 465: Andreas Hetzel
Erarbeitung und Qualifizierung einer Wärmebehandlungsstrategie zur Stoffflusssteuerung in der Blechmassivumformung
LFT, xiii u. 203 Seiten, 74 Bilder, 9 Tab. 2025.
ISBN 978-3-96147-830-9

Band 466: Sebastian Bickel
Entwicklung eines Vorgehensmodells zur automatisierten Ähnlichkeitssuche in den frühen Phasen der Produktentwicklung
KTmfk, xx u. 337 Seiten, 116 Bilder, 22 Tab. 2025.
ISBN 978-3-96147-832-3

Abstract

Use of existing data in the product development process promises increased product quality, fewer risks in development and less time and cost requirements. Overall, fewer resources are required for the development of new products, which leads to sustainable product development. However, the different data formats in the concept and design phases (e.g. principle sketches, components or assemblies) are a challenge for the use of similarity search.

As a result, a modular procedure has been developed as part of this thesis, which enables the search in the early phases of product development through the use of data-driven methods. Firstly, a solution for principle sketches was developed, which recognises the sketches and then automatically converts them into a graph. This forms the basis for the search for similar sketches, whereby different formats can be used as search input. In addition to the sketch search, a procedure is also presented in the work, which enables the linking of sketches and 3D assemblies, whereby 3D data is integrated into the development process at an early stage. Finally, a similarity search for components was created and then compared with the state of the art. Overall, the various modules have the same objective: to find and utilise existing data as effectively as possible.

Das Nutzen vorhandener Daten im Produktentwicklungsprozess verspricht eine erhöhte Produktqualität, weniger Risiken in der Entwicklung, sowie geringeren Zeit- und Kostenbedarf. Insgesamt sind dadurch weniger Ressourcen für die Entwicklung neuer Produkte notwendig, was zu einer nachhaltigen Produktentwicklung führt. Allerdings sind die verschiedenen Datenformate in den Konzept- und Entwurfsphasen (z.B. Prinzipskizzen, Bauteile oder Baugruppen) eine Herausforderung für den Einsatz der Ähnlichkeitssuche.

Demzufolge ist im Rahmen dieser Arbeit ein modulares Vorgehensmodell entwickelt worden, welches die Suche in den frühen Phasen der Produktentwicklung durch den Einsatz datengetriebener Methoden ermöglicht. Zunächst ist eine Lösung für Prinzipskizzen erarbeitet worden, welche die Skizzen erkennt und anschließend automatisiert in einen Graphen überführt. Dieser ist die Grundlage für die Suche nach ähnlichen Skizzen, wobei unterschiedliche Formate als Suchinput verwendet werden können. Neben der Skizzensuche ist auch ein Vorgehensmodell in der Arbeit präsentiert, welches die Verknüpfung von Skizzen und 3D-Baugruppen ermöglicht, wodurch frühzeitig 3D-Daten in den Entwicklungsprozess integriert werden. Zuletzt ist eine Ähnlichkeitssuche für Bauteile erarbeitet und anschließend mit dem Stand der Technik verglichen worden. Insgesamt umfassen die verschiedenen Module die gleiche Zielsetzung, die vorhandenen Daten so gut wie möglich aufzufinden und nutzbar zu machen.

ISBN 978-3-96147-832-3

9 783961 478323

FAU UNIVERSITY PRESS 2025