

Partikel-Schwarm-Optimierung zur Objektlageerkennung in Tiefendaten

Von der Fakultät Konstruktions-, Produktions- und Fahrzeugtechnik
der Universität Stuttgart
zur Erlangung der Würde eines
Doktors der Ingenieurwissenschaften (Dr.-Ing.)
genehmigte Abhandlung

vorgelegt von
Dipl.-Math. techn. Thomas Ledermann
geboren in Leonberg

Hauptberichter: Prof. Dr.-Ing. Alexander Verl
Mitberichter: Prof. Dr.-Ing. Dieter Fritsch
Tag der mündlichen Prüfung: 14.10.2011

Institut für Steuerungstechnik der Werkzeugmaschinen
und Fertigungseinrichtungen
der Universität Stuttgart

2012

IPA-IAO Forschung und Praxis

Berichte aus dem
Fraunhofer-Institut für Produktionstechnik und
Automatisierung (IPA), Stuttgart,
Fraunhofer-Institut für Arbeitswirtschaft und
Organisation (IAO), Stuttgart,
Institut für Industrielle Fertigung und
Fabrikbetrieb (IFF), Universität Stuttgart
und Institut für Arbeitswissenschaft und
Technologiemanagement (IAT), Universität Stuttgart und
Institut für Steuerungstechnik der Werkzeugmaschinen
und Fertigungseinrichtungen (ISW), Universität Stuttgart

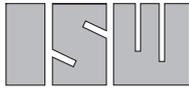
Herausgeber:

Univ.-Prof. Dr.-Ing. Prof. e.h. Dr.-Ing. e.h. Dr. h.c. mult. Engelbert Westkämper
und

Univ.-Prof. Dr.-Ing. habil. Prof. E.h. mult. Dr. h.c. mult. Hans-Jörg Bullinger
und

Univ.-Prof. Dr.-Ing. Dr.-Ing. E.h. Dieter Spath
und

Univ.-Prof. Dr.-Ing. Dr. h.c. Alexander Verl



Institut für Steuerungstechnik
der Werkzeugmaschinen und Fertigungseinrichtungen



Thomas Ledermann

Partikel-Schwarm-Optimierung zur Objektlageerkennung in Tiefendaten

Nr. 523

Dr.-Ing. Dipl.-Math. techn. Thomas Ledermann

Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA), Stuttgart

Univ.-Prof. Dr.-Ing. Prof. e.h. Dr.-Ing. e.h. Dr. h.c. mult. Engelbert Westkämper

ord. Professor an der Universität Stuttgart

Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA), Stuttgart

Univ.-Prof. Dr.-Ing. habil. Prof. E.h. mult. Dr. h.c. mult. Hans-Jörg Bullinger

ord. Professor an der Universität Stuttgart

Präsident der Fraunhofer-Gesellschaft, München

Univ.-Prof. Dr.-Ing. Dr.-Ing. E.h. Dieter Spath

ord. Professor an der Universität Stuttgart

Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO), Stuttgart

Univ.-Prof. Dr.-Ing. Dr. h.c. Alexander Verl

ord. Professor an der Universität Stuttgart

Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA), Stuttgart

D 93

ISBN 978-3-939890-97-3

Jost Jetter Verlag, Heimsheim

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils gültigen Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

© Jost-Jetter Verlag, Heimsheim 2012.

Printed in Germany.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Sollte in diesem Werk direkt oder indirekt auf Gesetze, Vorschriften oder Richtlinien (z. B. DIN, VDI, VDE) Bezug genommen oder aus ihnen zitiert worden sein, so kann der Verlag keine Gewähr für die Richtigkeit, Vollständigkeit oder Aktualität übernehmen. Es empfiehlt sich, gegebenenfalls für die eigenen Arbeiten die vollständigen Vorschriften oder Richtlinien in der jeweils gültigen Fassung hinzuzuziehen.

Druck: printsystem GmbH, Heimsheim

Geleitwort der Herausgeber

Über den Erfolg und das Bestehen von Unternehmen in einer marktwirtschaftlichen Ordnung entscheidet letztendlich der Absatzmarkt. Das bedeutet, möglichst frühzeitig absatzmarktorientierte Anforderungen sowie deren Veränderungen zu erkennen und darauf zu reagieren.

Neue Technologien und Werkstoffe ermöglichen neue Produkte und eröffnen neue Märkte. Die neuen Produktions- und Informationstechnologien verwandeln signifikant und nachhaltig unsere industrielle Arbeitswelt. Politische und gesellschaftliche Veränderungen signalisieren und begleiten dabei einen Wertewandel, der auch in unseren Industriebetrieben deutlichen Niederschlag findet.

Die Aufgaben des Produktionsmanagements sind vielfältiger und anspruchsvoller geworden. Die Integration des europäischen Marktes, die Globalisierung vieler Industrien, die zunehmende Innovationsgeschwindigkeit, die Entwicklung zur Freizeitgesellschaft und die übergreifenden ökologischen und sozialen Probleme, zu deren Lösung die Wirtschaft ihren Beitrag leisten muss, erfordern von den Führungskräften erweiterte Perspektiven und Antworten, die über den Fokus traditionellen Produktionsmanagements deutlich hinausgehen.

Neue Formen der Arbeitsorganisation im indirekten und direkten Bereich sind heute schon feste Bestandteile innovativer Unternehmen. Die Entkopplung der Arbeitszeit von der Betriebszeit, integrierte Planungsansätze sowie der Aufbau dezentraler Strukturen sind nur einige der Konzepte, welche die aktuellen Entwicklungsrichtungen kennzeichnen. Erfreulich ist der Trend, immer mehr den Menschen in den Mittelpunkt der Arbeitsgestaltung zu stellen - die traditionell eher technokratisch akzentuierten Ansätze weichen einer stärkeren Human- und Organisationsorientierung. Qualifizierungsprogramme, Training und andere Formen der Mitarbeiterentwicklung gewinnen als Differenzierungsmerkmal und als Zukunftsinvestition in *Human Resources* an strategischer Bedeutung.

Von wissenschaftlicher Seite muss dieses Bemühen durch die Entwicklung von Methoden und Vorgehensweisen zur systematischen Analyse und Verbesserung des Systems Produktionsbetrieb einschließlich der erforderlichen Dienstleistungsfunktionen unterstützt werden. Die Ingenieure sind hier gefordert, in enger Zusammenarbeit mit anderen Disziplinen, z. B. der Informatik, der Wirtschaftswissenschaften und der Arbeitswissenschaft, Lösungen zu erarbeiten, die den veränderten Randbedingungen Rechnung tragen.

Die von den Herausgebern langjährig geleiteten Institute, das

- Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA),
- Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO),
- Institut für Industrielle Fertigung und Fabrikbetrieb (IFF),
- Institut für Arbeitswissenschaft und Technologiemanagement (IAT),
- Institut für Steuerungstechnik der Werkzeugmaschinen und Fertigungseinrichtungen (ISW),

arbeiten in grundlegender und angewandter Forschung intensiv an den oben aufgezeigten Entwicklungen mit. Die Ausstattung der Labors und die Qualifikation der Mitarbeiter haben bereits in der Vergangenheit zu Forschungsergebnissen geführt, die für die Praxis von großem Wert waren. Zur Umsetzung gewonnener Erkenntnisse wird die Schriftenreihe „IPA-IAO - Forschung und Praxis“ herausgegeben. Der vorliegende Band setzt diese Reihe fort. Eine Übersicht über bisher erschienene Titel wird am Schluss dieses Buches gegeben.

Dem Verfasser sei für die geleistete Arbeit gedankt, dem Jost Jetter Verlag für die Aufnahme dieser Schriftenreihe in seine Angebotspalette und der Druckerei für saubere und zügige Ausführung. Möge das Buch von der Fachwelt gut aufgenommen werden.

Engelbert Westkämper Hans-Jörg Bullinger Dieter Spath Alexander Verl

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA in Stuttgart.

Mein großer Dank gilt Herrn Prof. Dr.-Ing. Alexander Verl für die Übernahme des Hauptberichts sowie für seine Unterstützung und Förderung, sowohl für diese Arbeit als auch für meine Tätigkeit am Fraunhofer IPA. Mein Dank gilt ebenfalls Herrn Prof. Dr.-Ing. Dieter Fritsch für die Durchsicht der Arbeit und für die Übernahme des Mitberichts. Für die Organisation des Promotionsprozesses möchte ich Frau Heide Kreuzburg danken.

Ein besonderer Dank gilt den Herren Dr.-Ing. Dennis Fritsch, Dr.-Ing. Kai Wegener und Dr.-Ing. Johannes Wößner für die Durchsicht meiner Manuskripte, für die vielen wertvollen Anregungen und für die motivierenden Worte sowie Frau Luzia Schuhmacher M. A. für ihre Lektor-Arbeiten.

Herrn Dipl.-Ing. Matthias Palzkill danke ich darüber hinaus für die außerordentlich konstruktive fachliche Zusammenarbeit.

Ebenfalls danken möchte ich Herrn Klaus Killmann für die stets engagierte Unterstützung bei der Realisierung der Versuchsaufbauten sowie meinen ehemaligen Studenten, die in Form von Praktika, Studien- und Diplomarbeiten zu dieser Arbeit beigetragen haben.

Meinen Eltern danke ich für ihre Förderung und Unterstützung, die sie mir während meines Studiums und meiner Promotion entgegengebracht haben.

Mein liebevoller Dank gilt meiner Frau Karin Valeria Ledermann und meinen Kindern Adrian, Jule und Felix für die Unterstützung und Rücksichtnahme während der Promotionszeit. Ihnen möchte ich diese Arbeit widmen.

Ottmarsheim im Juni 2012

Thomas Ledermann

Inhaltsverzeichnis

1	Einleitung	16
1.1	Motivation und Problemstellung	16
1.2	Zielsetzung	17
1.3	Vorgehensweise	18
2	Ausgangssituation	19
2.1	Begriffe und Definitionen	19
2.1.1	Tiefensensoren	19
2.1.2	Tiefendaten	19
2.1.3	Ordnungszustand	20
2.2	Stand der Technik und Forschung	21
2.2.1	Objektlageerkennungssysteme für die Handhabung von Objekten mittels Robotern	21
2.2.2	Optimierungsverfahren für die Objektlageerkennung	30
2.3	Folgerungen aus dem Stand der Technik und der Forschung	40
3	Analyse potentieller Einsatzszenarien für ein System zur Objektlageerkennung und Ableitung von Anforderungen	41
3.1	Analyse des Gießerei- und Schmiedeumfelds	41
3.2	Analyse der Entnahme von Getriebewellen für eine Ultraschall-Prüfung zur automatischen Qualitätssicherung	42
3.3	Analyse der Entnahme von Hohlrädern für die Zuführung in eine Bearbeitungsmaschine	43
3.4	Analyse der Entnahme von Gehäusedeckeln für eine manuelle Entgratung	44
3.5	Analyse der Integrationsmöglichkeit der Partikel-Schwarm-Optimierung in ein Objektlageerkennungsverfahren	45
3.6	Zusammenfassung und Strukturierung der Anforderungen an ein System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung für die Handhabung von Objekten mittels Robotern	46
3.6.1	Definition von Teilsystemen eines Systems zur Objektlageerkennung mit Partikel-Schwarm-Optimierung für die Handhabung von Objekten mittels Robotern	46
3.6.2	Strukturierung der Anforderungen und Zuordnung zu den definierten Teilsystemen	47

4	Lösungsansätze für ein System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung	48
4.1	Lösungsansätze zur Objektbeschreibung	48
4.2	Lösungsansätze für den Partikel-Schwarm	49
4.2.1	Modellierung eines Schwarmpartikels	49
4.2.2	Partikel-Schwarm-Steuerung	60
4.3	Lösungsansätze zur Datenvorverarbeitung der Sensordaten	61
4.3.1	Einlesen der Sensordaten	62
4.3.2	Lösungsansatz zur Auflösung von Mehrdeutigkeiten in den Abstandswerten	62
4.3.3	Verfahren zur Auflösung von Mehrdeutigkeiten in den Abstandswerten mittels Äquipotenzialen	65
4.3.4	Umgang mit Abschattungen durch Zentralprojektion	68
4.3.5	Filtern der Sensordaten	68
4.4	Lösungsansätze zur Segmentierung der Sensordaten	69
4.5	Zusammenführung der Teilkonzepte zu einem System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung	70
4.5.1	Gesamtsystem	70
4.5.2	Regelablauf	72
5	Entwicklung eines wissensbasierten Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels	75
5.1	Prinzipieller Ablauf der histogrammbasierten Bewertung eines Schwarmpartikels	75
5.2	Ermittlung, Klassifizierung und Eingrenzung der Einflussfaktoren auf die histogrammbasierte Bewertung eines Schwarmpartikels	76
5.3	Verfahren zur Erzeugung von Tiefendaten aus CAD-Daten und hiervon abgeleiteten Histogrammen für die Wissensbasis	77
5.3.1	Modellierung eines Tiefensensors zur Generierung der Tiefendaten aus CAD-Daten eines Objekts	77
5.3.2	Bestimmung des Sichtbereichs und der lateralen Auflösung des modellierten Tiefensensors	78
5.3.3	Bestimmung des Tiefenbereichs des modellierten Tiefensensors	80
5.3.4	Erstellung der Tiefenhistogramme für die Wissensbasis	80
5.3.5	Erstellung der Normalenvektorhistogramme für die Wissensbasis	81
5.3.6	Entwicklung eines Tools zur Erstellung der Wissensbasis	83

5.4	Aufstellung der Fitnessfunktion zur Bewertung eines Schwarmpartikels mithilfe des Histogrammvergleichs	84
5.4.1	Alternative Methoden des Histogrammvergleichs	84
5.4.2	Erweiterung der χ^2 -Divergenz für den Einsatz als Fitnessfunktion	87
5.4.3	Vergleichbarkeit der synthetischen Tiefendaten mit den realen Tiefendaten	88
5.5	Bewertung der durch die Wissensbasis erzeugten Lösungsräume	89
6	Realisierung und Verifizierung des Gesamtsystems	92
6.1	Zielsetzung und Planung der Versuchsdurchführung	92
6.2	Erstellung der Wissensbasis für die gewählten Testobjekte	93
6.3	Experimentelle Untersuchungen an synthetischen Daten einzelner Objekte	94
6.3.1	Generierung der Testdaten mithilfe des modellierten Tiefensensors	94
6.3.2	Auswahl der untersuchten Objektlagen	94
6.3.3	Getestete Schwarmparameter und erzielte Erkennungsraten	95
6.3.4	Interpretation der Ergebnisse	99
6.4	Experimentelle Untersuchungen an synthetischen Daten einer Szene mit mehreren Objekten	100
6.4.1	Generierung der Testdaten mithilfe des modellierten Tiefensensors	100
6.4.2	Auswahl der untersuchten Testszenen	101
6.4.3	Getestete Schwarmparameter und erzielte Ergebnisse	102
6.4.4	Zusammenfassung und Interpretation der Ergebnisse	103
6.5	Aufbau einer Robotertestzelle zur experimentellen Untersuchung an realen Sensordaten	104
6.5.1	Zellenlayout	104
6.5.2	Der eingesetzte Sensor	105
6.5.3	Gesamtablauf der Objekthandhabung	106
6.5.4	Erzielte Ergebnisse	107
6.6	Fazit aus den durchgeführten Tests	108
7	Zusammenfassung und Ausblick	110
8	Abstract	113
9	Literaturverzeichnis	116

Abkürzungen und Formelzeichen

Abkürzungen

CAD	Computer Aided Design
GA	Genetische Algorithmen
gbest	globel best Nachbarschaft eines Partikels
lbest	local best Nachbarschaft eines Partikels
OG	Orientierungsgrad
OZ	Ordnungszustand $OZ = OG / PG$
PG	Positionierungsgrad
PSO	Partikel-Schwarm-Optimierung
SIFT	Scale Invariant Feature Transform
STL	SurfaceTesselationLanguage – Beschreibung einer Oberfläche durch Dreiecke
VTK	Visualization Toolkit

Formelzeichen in lateinischen Buchstaben

A	[]	Abstandsdaten
a_{Gesamt}	[mm]	Sichtbereich des modellierten Sensors
a_i	[mm]	Abstände mit Index i
Anz_{maskiert}	[]	Anzahl maskierter Werte
a_{Schritt}	[mm]	laterale Auflösung
B	[]	Bezugspunkt/Ursprung eines 3-D-Sensors
c_i	[]	allgemeine Faktoren mit Index i
C_i	[mm ³]	Objektmerkmalsdaten zur Beschreibung des Schwarmpartikels p_i
$C_{i,Anz}$	[]	Anzahl der Messpunkte der Objektmerkmalsdaten C_i
D	[mm ³]	Punktwolke
d_i	[mm]	orthogonale Abstände
d_{Obj}	[mm]	Raumdiagonale der Bounding-Box um ein Objekt
E_B	[mm ³]	eine Ecke der Bounding-Box um ein Objekt
e_i	[]	Einheitsvektoren eines ortsfesten Koordinatensystems
e'_i	[]	Einheitsvektoren eines körperfesten Koordinatensystems eines starren Körpers. Sie ändern ihre Orientierung mit dem Objekt.
$E_{\text{systematisch}}$	[mm]	systematischer Messfehler des eingesetzten Tiefensensors
f	[]	Fitnessfunktion
F_B	[mm ²]	Bezugsebene

F_O	[mm ²]	Oberfläche
$g(x, y)$	[]	Approximationsfunktion für eine Fläche
g_u, g_v	[]	1. Ableitungen von g in Richtung der Bildkoordinaten u, v
$h, h^{(1)}, h^{(2)}$	[]	Gesamtzahl aller Messwerte in einem Histogramm
$H, H^{(1)}, H^{(2)}$	[]	Histogramme
H_m	[]	Teilbereich eines Histogramms mit Index m
H_N	[]	Normalenvektorhistogramm
$H_{N,Anz}$	[]	Anzahl der Teilbereiche von H_N in jeder Dimension
h_{Sensor}	[mm]	orthogonaler Abstand zwischen realem Tiefensensor und Boden
H_T	[]	Tiefenhistogramm
$H_{T,Anz}$	[]	Anzahl der Teilbereiche von H_T
$H_{T,Index}(i_x, i_y)$	[]	Index des Teilbereichs von H_T , in den das Element (i_x, i_y) der Tiefenmatrix T_M fällt
i_x, i_y	[]	Indizes der Tiefenmatrix T_M
I_x, I_y	[]	Intervalle natürlicher Zahlen
K	[mm ³]	Anfangspunkt eines Kantenzugs eines Objekts
L	[]	Lösungsraum, in dem sich die Schwarmpartikel für die Suche bewegen
L_R	[]	Lösungsraum, der nur die rotatorischen Freiheitsgrade enthält
M	[]	Anzahl der Teilbereiche eines Histogramms
M_B	[mm ³]	Mittelpunkt der Bounding-Box um ein Objekt
M_Q	[mm ³]	Mittelpunkt eines Querschnitts durch ein CAD-Modell
M_S	[]	Tiefenmatrix mit synthetisch generierten Tiefendaten
M_T	[]	Tiefenmatrix mit realen Tiefendaten
N	[mm ³]	Normalenvektor
$N_A(w_E)$	[]	Nachbarn von w_E , die einen kleineren oder gleichen euklidischen Abstand zu w_0^{\min} haben wie w_E
$N_{\tilde{A}qui}$	[]	Niveau eines Äquipotentials
n_{M_S}	[]	Dimension von M_S
n_x, n_y, n_z	[mm]	Komponenten des Normalenvektors in x-, y-, z-Richtung
$N(w_o)$	[]	Nachbarn von w_o
pb_i	[]	beste bisher gefundene Lösung des Partikels p_i
P_F	[mm ³]	ein frei definierter Punkt auf der Oberfläche eines Objekts
pg_i	[]	beste bisher gefundene Lösung eines Nachbarpartikels
p_i	[]	Schwarmpartikel mit dem Index i
P_i	[mm ³]	Objektposition zur Beschreibung des Schwarmpartikels p_i
$p_i(t)$	[]	Position des Schwarmpartikels p_i zur Zeit t
p_m	[]	Position innerhalb des Lösungsraums als gewichtetes Mittel von eigener bester Position und bester Position der Nachbarn

p^{Max}	[mm ³]	höchster Punkt eines Objekts
P_{OL}	[]	die Menge aller höchsten Punkte von T_S
q	[]	Quaternion
r	[mm]	Radius einer Kugel
R	[rad ³]	Rotation eines Objekts
R_i	[rad ³]	Objektorientierung zur Beschreibung des Schwarmpartikels p_i
r_S	[rad]	Rollwinkel des Tiefensensors
$R(\alpha, \beta, \gamma)$	[rad ³]	Rotation eines Objekts mit drei Eulerwinkeln (α, β, γ)
S	[mm ³]	Schwerpunkt des Objekts
$s_\alpha, s_\beta, s_\gamma$	[rad]	Schrittweite der Rotationen um die x-, y- und z-Achse in Eulerwinkeln
T	[mm ³]	Tiefendaten mit zugehöriger Abbildung $z = T(x, y)$
T_D	[mm ³]	äquidistant diskretisierte reale Tiefendaten mit zugehöriger Abbildung $z = T_D(i_x, i_y)$
$T_E(w_0^{\text{Min}})$	[]	Einzugstrichter des Minimums w_0^{Min}
t_S	[mm]	Translation von dem Ursprung eines ortsfesten Koordinatensystems zum Ursprung eines starren Körpers
T_S	[mm ³]	synthetische Tiefendaten mit zugehöriger Abbildung $z = T_S(x, y)$
T_{Sensor}	[mm]	Tiefenbereich, der von dem modellierten Sensor betrachtet wird
$T_{\text{Sensor,Rot}}$	[mm]	Tiefenbereich, der von einem Objekt bedeckt wird, in Abhängigkeit von der Rotation
v^{Max}	[]	Maximalgeschwindigkeit
$v(t)$	[]	Änderungsvektor eines Partikels p_i zum Zeitpunkt t
W	[]	Wissensbasis
w, w_0, w_E	[]	Elemente von W
x, y, z	[]	allgemeine Bezeichnung für Achsen eines Koordinatensystems oder der zugehörigen Komponenten
x', y', z'	[]	allgemeine Bezeichnung für Achsen eines Koordinatensystems oder der zugehörigen Komponenten, das einmal rotiert wurde
x'', y'', z''	[]	allgemeine Bezeichnung für Achsen eines Koordinatensystems oder der zugehörigen Komponenten, das zweimal rotiert wurde
x_i	[]	Koordinatenachsen durch den Schwerpunkt eines starren Körpers
x'_i	[]	körperfeste Koordinatenachsen eines starren Körpers durch dessen Schwerpunkt
X^{Max}	[]	Menge aller x-Koordinaten von P_{OL}
Y^{Max}	[]	Menge aller y-Koordinaten von P_{OL}
z_S	[mm]	Tiefenwert des Schwerpunktes der Bounding-Box um ein Objekt

Formelzeichen in griechischen Buchstaben

α_e	□	Explorationsfaktor
α_{Sensor}	□	Scanbereich eines Tiefensensors
$\alpha_{Schritt}$	[rad]	Winkelauflösung eines Tiefensensors
α_x	[rad]	Messwinkel in die x -Richtung
α_y	[rad]	Messwinkel in die y -Richtung
(α, β, γ)	[rad ³]	Rotationswinkel zur Beschreibung einer Objektorientierung mit Eulerwinkeln in der Yaw-Pitch-Roll Variante
χ	□	Constriction Factor, der die Konvergenz der Schwarmsuche sichert
$\chi^2(H^{(1)}, H^{(2)})$	□	χ^2 -Divergenz von $H^{(1)}$ und $H^{(2)}$
$\vartheta_1, \vartheta_2, \vartheta_3$	[rad]	Winkel zwischen x_1, x_2, x_3 und x'_3
ψ_1, ψ_2, ψ_3	[rad]	Winkel zwischen x_1, x_2, x_3 und x'_2
ϕ_1, ϕ_2, ϕ_3	[rad]	Winkel zwischen x_1, x_2, x_3 und x'_1
φ, θ	[rad]	Winkel zur Darstellung mittels Kugelkoordinaten
κ	□	Faktor innerhalb des Constriction Factors χ

Rechenoperatoren und Indizes

- Skalarmultiplikation zweier Vektoren
- * Faltung
- [] Gaußklammer; weist jeder reellen Zahl die größte ganze Zahl zu, die kleiner oder gleich der reellen Zahl ist
- Max* Maximalwert einer Variablen
- Min* Minimalwert einer Variablen
- T* transponierte Matrix oder Vektor

1 Einleitung

1.1 Motivation und Problemstellung

Roboter sind in vielen Produktionen nicht mehr wegzudenken. Laut Statistik sind derzeit über 1 Mio. Roboter weltweit im Einsatz [IFR 2009]. Die Automobilindustrie ist mit einem Marktanteil von 35 % immer noch die tragende Säule. Allerdings versuchen Roboterhersteller, Systemintegratoren und Forschungseinrichtungen seit einigen Jahren, Robotern den Weg in kleine und mittlere Unternehmen (KMU) zu ebnet [SMERobot 2008]. Neben der Erhöhung der Produktivität durch den Einsatz von Automatisierungslösungen spielt für viele Unternehmen die Erleichterung der körperlichen Arbeit für die Werker eine wichtige Rolle. Gerade in den Betrieben der Schmiede- und Gießereibranche ist dieser Trend aufgrund der häufig schweren oder unhandlichen Teile deutlich zu erkennen [Herbertz 2005].

Eine der Hauptherausforderungen für den Einsatz von Robotern in KMUs ist die Erhöhung ihrer Flexibilität, um den Anforderungen aufgrund häufiger Produktwechsel und sinkender Produktlebenszyklen gerecht zu werden. Diese Flexibilität kann insbesondere durch die Integration von geeigneten Sensoren erfolgen, die es dem Roboter ermöglichen, auf geänderte Situationen – wie beispielsweise einer variablen Position eines Werkstücks – zu reagieren. Als eine der Königsdisziplinen gilt dabei der „Griff-in-die-Kiste“, d.h. die Entnahme von chaotisch in einer Kiste liegenden Objekten mit einem Roboter.

Seit den 1980er Jahren werden Robotersysteme entwickelt und intensiv erforscht, die in der Lage sind, den Griff-in-die-Kiste zu lösen, indem die Lage von Objekten aus einem Haufen oder zumindest bei beliebiger Orientierung auf einer Ebene erkannt wird. [Müller 2001] zeigte auf, dass es entgegen der Erwartung kein realisiertes System gibt, das gleichzeitig die wichtigen Anforderungen an Flexibilität und Zuverlässigkeit erfüllt, um universell einsetzbar zu sein. Ein breiter Einsatz von Griff-in-die-Kiste-Systemen in industrielle Anwendungen ist auch weiterhin nicht erkennbar, vielmehr sind lediglich vereinzelt Speziallösungen zu finden [Verl 2009].

Bis vor einigen Jahren konzentrierten sich die meisten Forschungsaktivitäten im Bereich der Objektlageerkennung für Handhabungsaufgaben auf die Extraktion und Analyse von Informationen aus Intensitäts- oder Videobildern. Durch die Verfügbarkeit einer Vielzahl zuverlässiger 3-D-Sensoren mit unterschiedlichen Prinzipien befassen sich mehr und mehr Forscher mit der Analyse der Tiefendaten solcher 3-D-Sensoren zur Lösung des 3-D-Bin-Picking, wie dieser Forschungsbereich bezeichnet wird. Dieser Forschungsbereich, der den Einsatz neuer 3-D-Sensoren vorantreibt, gilt als die Zukunft der Robot-Vision [Fili 2008].

1.2 Zielsetzung

Ziel der vorliegenden Arbeit ist die Entwicklung eines flexiblen und fehlertoleranten Systems zur Objektlageerkennung mit Tiefendaten für die Anwendung im Schmiede- und Gießereibereich. Das Erkennungssystem soll die automatisierte Entnahme von typischen Objekten aus dem Schmiede- und Gießereibereich, die sortenrein in einem Haufen angeliefert werden, mit einem Roboter ermöglichen. Hierzu soll der Einsatz des noch jungen Forschungsgebiets der Partikel-Schwarm-Optimierung (PSO) untersucht und für den Einsatz zur Objektlageerkennung nutzbar gemacht werden.

Im Folgenden wird näher beschrieben, was unter „flexibel“ und „fehlertolerant“ im Zusammenhang mit dem System zur Objektlageerkennung verstanden wird.

Flexibilität bedeutet, dass das Erkennungssystem an bisher unbekannte Objekte und an unterschiedliche 3-D-Sensoren angepasst werden kann:

- Das Vorwissen, das benötigt wird, um die Lage der Objekte zu erkennen, soll automatisiert erweiterbar sein.
- Es sollen keine speziellen Zusammenhänge zwischen Objektmerkmalen, die zu einem komplexen Entscheidungsbaum führen, genutzt werden, um auch ohne Expertenwissen eine Erweiterung durchführen zu können.
- Es sollen nur die Tiefendaten genutzt werden. Weitere spezielle Sensoreigenschaften – wie Reflektivitätswerte – sollen nicht genutzt werden, um eine einfache Übertragbarkeit an verschiedene 3-D-Sensoren zu ermöglichen.

Diese Forderungen schränken nicht ein, dass in zukünftigen Entwicklungen Erweiterungen zum Einsatz kommen, die auch spezielle Objekt- oder Sensormerkmale und -eigenschaften in vorgelagerten oder nachfolgenden Schritten einsetzen, um zusätzliche Funktionalitäten zu ermöglichen.

Dem Begriff „fehlertolerant“ werden hier insbesondere folgende Aspekte zugeordnet:

- Das Erkennungssystem soll mit realen Sensordaten (inklusive Rauschen) vor realem Hintergrund arbeiten.
- Kleine Abweichungen in der Objektgeometrie – etwa durch Toleranzen, Grate oder Angussreste – wie sie gerade im Schmiede- und Gießereibereich anzutreffen sind, sollen die Zuverlässigkeit des Erkennungssystems nicht entscheidend verringern.

1.3 Vorgehensweise

Um dieses Ziel der Entwicklung eines solchen Erkennungssystems zu erreichen, wird wie folgt vorgegangen:

In Kapitel 2 erfolgt nach einer Einführung in die Ausgangssituation mit den wichtigsten Begriffen und Definitionen die Vorstellung des Stands der Technik und der Forschung. Diese umfasst einen Überblick über die Entwicklungen von Systemen zur Objektlageerkennung sowie eine Einführung in die Partikel-Schwarm-Optimierung.

Aufgrund der Analyse von drei Szenarien aus dem Schmiede- und Gießereibereich zur Entnahme von Objekten aus einer Kiste erfolgt in Kapitel 3 die Ableitung von Anforderungen an ein Verfahren zur Objektlageerkennung mit Partikel-Schwarm-Optimierung. Als Szenarien dienen hier die Entnahme von Getriebewellen, um diese einer Ultraschallprüfung zuzuführen, die Entnahme von Gehäusedeckeln, die anschließend manuell entgratet werden sollen, sowie die Entnahme von Hohlringen, die einer Bearbeitungsmaschine zugeführt werden.

Die Entwicklung von Lösungsansätzen für ein System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung erfolgt in Kapitel 4. Für dieses Erkennungssystem wird in Kapitel 5 eine Fitnessfunktion aufgestellt, die essentieller Bestandteil zur PSO-basierten Objektlageerkennung ist. Zusätzlich wird ein Software-Tool entwickelt, um das für jedes Objekt benötigte Vorwissen automatisch erstellen zu können.

Abschließend wird das realisierte Gesamtsystem in Kapitel 6 auf Grundlage von synthetisch erstellten und real mit einem Tiefensensor generierten Testszenarien validiert und in Bezug auf die zuvor zusammengetragenen Anforderungen bewertet. Insbesondere die synthetischen Szenarien mit mehreren Objekten, die auf Basis eines CAD-Modells des gesuchten Objekts erstellt und als Tiefendaten abgespeichert werden, sind eine wichtige Grundlage zur Bewertung des Objektlageerkennungsalgorithmus. Mithilfe der synthetischen Szenarien ist es einfach möglich, die gefundenen Objektlagen mit den tatsächlichen, synthetisch erstellten Objektlagen zu vergleichen. Abgerundet werden die Tests durch den Aufbau einer Roboterzelle, mit deren Hilfe komplette Entnahmezyklen von der Sensoraufnahme bis zum Ablegen eines gegriffenen Objekts getestet werden. Insgesamt liefern die durchgeführten Versuche wichtige Hinweise zur Umsetzung PSO-basierter Objektlageerkennungssysteme und dienen zur Identifikation von Optimierungspotenzialen für künftige Entwicklungen zur Kombination der beiden Forschungsgebiete PSO und wissensbasierter Objektlageerkennung.

2 Ausgangssituation

2.1 Begriffe und Definitionen

2.1.1 Tiefensensoren

Tiefensensoren dienen dazu, die Oberfläche einer Szene geometrisch zu erfassen. Sie haben gegenüber Farb- und Grauwertaufnahmen den Vorteil, dass aus ihnen die Information über die 3-D-Form des Objekts viel einfacher und genauer extrahiert werden kann [Verl 2007]. Die durch den Sensor gewonnenen Daten geben direkt Informationen über die 3-D-Form der Objekte in der Szene wieder. Jeder Pixel des Sensors gibt hierbei die Entfernung zu einem Punkt der Objektoberfläche wieder. Ein wesentlicher Vorteil der in der Praxis eingesetzten 3-D-Verfahren, die auf Reflektanzmessungen oder Triangulation beruhen, wie Lichtschnittverfahren, Lasertriangulation, Streifenprojektion oder Laserlaufzeitmessung ist, dass sie weitgehend unabhängig von der Objektfarbe und der Umgebungsbeleuchtung arbeiten. Die größten Schwierigkeiten solcher 3-D-Verfahren treten auf, wenn stark absorbierende, spiegelnde oder durchsichtige Materialien gemessen werden sollen. Zusätzlich erzeugen unterschiedliche Positionen des gemessenen Objekts gegenüber der Sensorposition unterschiedliche Tiefendaten. Dieser Einfluss wird im folgenden Abschnitt genauer betrachtet.

2.1.2 Tiefendaten

Typischerweise liefern Tiefensensoren **Abstandsdaten** $A = \{a_i; i \in \mathbb{N}\}$, die die Abstände a_i ($i \in \mathbb{N}$) von Punkten einer Oberfläche F_O zu einem festen Bezugspunkt B angeben (siehe Abbildung 2-1 links). B ist hierbei der Ursprung des 3-D-Sensors, der zur Messung der Abstände a_i genutzt wird. Somit liegt A als Zentralprojektion bezüglich B vor. Die Umrechnung dieser Abstandsdaten in das lokale Koordinatensystem (x, y, z) mit B als Ursprung liefert eine **Punktwolke** $D = \{(x_i, y_i, z_i) \in \mathbb{R}^3, i \in \mathbb{N}\}$ von F_O .

Die Zentralprojektion führt dazu, dass Objekte in gleicher rotatorischer Lage in unterschiedlichen Positionen unterschiedliche Abstandsdaten und Abschattungen liefern. Für die Objektlageerkennung wäre ideal, wenn gleiche rotatorische Lagen in unterschiedlichen Positionen zu gleichen Abstandsdaten ohne Abschattungen führen würden.

Solche Daten liefert die Parallelprojektion, ein Grenzfall der Zentralprojektion mit einem unendlich weit vom Objekt entfernten Betrachtungspunkt. Bei der Parallelprojektion wird nicht der Abstand von B zu F_O gemessen, sondern die orthogonale Distanz d_i ($i \in \mathbb{N}$) einer Bezugsebene F_B mit einem ortsfesten Koordinatensystem (x, y, z) mit Ursprung in B zu F_O (siehe Abbildung 2-1 rechts).

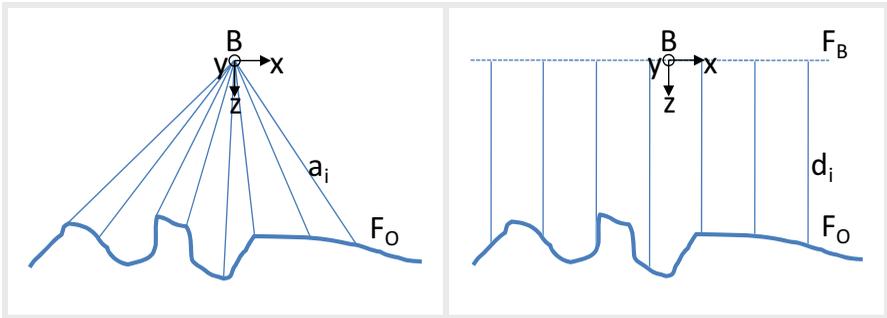


Abbildung 2-1 links: Abstandsdaten A einer Zentralprojektion, rechts: Tiefendaten T einer Parallelprojektion

Mithilfe des lokalen Koordinatensystems liefert die Parallelprojektion eine eindeutige Zuweisung eines z-Werts für jeden Punkt der Oberfläche F_O . Ergebnis ist somit eine Fläche, die folgender Gleichung genügt.

$$T: \mathbb{R}^2 \rightarrow \mathbb{R}: (x, y) \rightarrow T(x, y) =: z \quad \text{mit } (x, y) \in F_O \quad \text{Gleichung 2-1}$$

Diese Daten werden im Folgenden als **Tiefendaten** $T = \{(x, y, z) \in \mathbb{R}^3; T(x, y) = z\}$ bezeichnet.

2.1.3 Ordnungszustand

Der Ordnungszustand gibt an, in wie vielen Freiheitsgraden die räumliche Anordnung eines Objekts bestimmt ist. Dabei kann in rotatorische Freiheitsgrade (Orientierungsgrad OG) und in translatorische Freiheitsgrade (Positionierungsgrad PG) unterschieden werden. Der Orientierungsgrad OG gibt die Anzahl rotatorischer Freiheitsgrade an, in denen die Orientierung bestimmt ist. Entsprechend gibt der Positionierungsgrad PG die Anzahl translatorischer Freiheitsgrade an, in denen die Orientierung bestimmt ist [Schraft 2000].

Abbildung 2-2 gibt eine Übersicht über die möglichen Freiheitsgrade. Der hierbei gebrauchte Begriff des Ordnungszustands (OZ) ist eine Kombination aus Orientierungsgrad OG und Positionierungsgrad PG: $OZ = OG/PG$.

OZ = (OG/PG)	Werkstück-Rotation OG				
	→				
Werkstück-Translation PG	(0/0)	(1/0)	(2/0)	(3/0)	3 Positionsgrade unbestimmt (Haufwerk)
	(0/1)	(1/1)	(2/1)	(3/1)	1 Positionsgrad bestimmt (flächige Verteilung)
	(0/2)	(1/2)	(2/2)	(3/2)	2 Positionsgrade bestimmt (linienförmige Verteilung)
	(0/3)	(1/3)	(2/3)	(3/3)	3 Positionsgrade bestimmt (vereinzelte)
	3 Drehfreiheitsgrade unbestimmt	1 Drehfreiheitsgrad bestimmt	2 Drehfreiheitsgrade bestimmt	3 Drehfreiheitsgrade bestimmt	

Abbildung 2-2: Ordnungszustand OZ zur Beschreibung der Objektfreiheitsgrade [Schratt 2000]

Diese Freiheitsgrade spiegeln sich entsprechend in der Bereitstellungsart der Objekte, die der Roboter greifen soll, wider. Ein OZ = 0/0 sagt aus, dass sich die Werkstücke im ungeordneten Zustand im „Haufwerk“ befinden, d. h. weder Position noch Orientierung der Werkstücke sind festgelegt. Ein OZ = 3/3 sagt aus, dass die Werkstücke in allen Freiheitsgraden festgelegt sind, also geordnet und „vereinzelte“ vorliegen.

2.2 Stand der Technik und Forschung

2.2.1 Objektlageerkennungssysteme für die Handhabung von Objekten mittels Robotern

Die Kombination aus industrieller Bildverarbeitung mit Robotik ist eine der Erfolgsbranchen in Deutschland, die ihren Umsatz in den letzten zehn Jahren verdreifacht hat [Brodthmann 2008]. Für eine Vielzahl von Anwendungen reicht es in der Praxis nicht aus, dass der Roboter immer die gleiche Bewegung ausführt. Vielmehr ist es nötig, dass der Roboter durch den Einsatz von Bildverarbeitung auf Änderungen wie beispielsweise eine andere Objektlage reagieren kann. Deshalb werden zukünftig immer mehr Robot-Vision-Systeme mit 3-D-Sensoren zum Einsatz kommen [Fili 2008].

Um die Leistungsfähigkeit solcher Robotersysteme, die Systeme zur Objektlageerkennung einsetzen, miteinander vergleichen zu können, gibt es in der Fachliteratur mehrere Ansätze [Wolter 1986, Müller 2001]. Da der Schwerpunkt dieser Arbeit die Objektlageerkennung ist, hat sich eine modifizierte Klassifikation nach folgenden Kriterien K1 bis K5 als zweckmäßig erwiesen:

- K1: Art der ausgewerteten Sensordaten
- K2: Genutzte Informationen über das Objekt
- K3: Bereitstellung der Objekte
- K4: Nachfolgender Prozess
- K5: Geometrische Besonderheiten der Objekte

2.2.1.1 K1: Art der ausgewerteten Sensordaten

Aufgrund der ausgewerteten Sensordaten lassen sich Systeme zur Objektlageerkennung in drei Hauptgruppen einteilen:

1. Gruppe: Kamerabasierte Objektlageerkennung (Auswertung von 2-D-Daten)
2. Gruppe: Kamerabasierte Objektlageerkennung in Kombination mit einem Tiefensensor (Kombination aus 2-D- und 3-D-Daten)
3. Gruppe: Tiefensensorbasierte Objektlageerkennung (Auswertung von 3-D-Daten)

2.2.1.2 K2: Genutzte Informationen über das Objekt

In der Fachliteratur gibt es hinsichtlich der genutzten Objektinformationen zwei grundlegend unterschiedliche Ansätze für die Objektlageerkennung. Im ersten Ansatz wird in den Sensordaten nach Merkmalen gesucht, die für einen Greifvorgang geeignet sind, ohne das Objekt selbst oder die Lage des Objekts zu identifizieren. Solche Ansätze mit einer Suche nach Greifmerkmalen sind insbesondere für die Entnahme unbekannter Objekte geeignet. Der zweite Ansatz ist auf die Entnahme bekannter Objekte gerichtet. Ziel ist es, aufgrund des Vorwissens über das Objekt mindestens eine Objektlage in den Sensordaten zu identifizieren. Aufgrund der Art, wie das Vorwissen über das Objekt zur Identifikation einer Lage modelliert wird, können die Verfahren dieses Ansatzes in drei Gruppen unterteilt werden:

- die Objektlageerkennung mit Wissensbasen, die Objektbeschreibungen enthalten,
- Verfahren, die generische Objektmodelle zugrunde legen und
- heuristische Verfahren zur Objektseparation und -rekonstruktion.

Die Zuordnung der im Folgenden betrachteten Ansätze zu den drei Gruppen erfolgt aufgrund des zugrunde liegenden Objektmodells. Das jeweils eingesetzte Bewertungsverfahren wird zusätzlich bei solchen Verfahren aufgeführt, bei denen die Bewertung direkt mit dem Objektmodell

zusammenhängt. Als klassische Bewertungsverfahren sind hier unter anderem stochastische Verfahren wie beispielsweise Hidden-Markov-Modelle [Hornegger 1994], lernende Verfahren wie Neuronale Netze [Heidemann 1996, Walter 1998] oder auch die Kombination aus Hidden-Markov-Modellen mit Neuronalen Netzen [Park 1999] sowie verschiedene Distanz-Kriterien zu nennen. Eine Übersicht solcher Bewertungsverfahren kann beispielsweise [Bennamoun 2002] entnommen werden.

Suche nach Greifmerkmalen

Beispiele von Erkennungssystemen, die nicht das gesamte Objekt identifizieren, sondern nach Greifmerkmalen suchen, sind der „Shrinking Algorithm“ für Vakuumgreifer [Birk 1981] oder „Edge Detection“ und „Region Growing“ für Backen-Greifer [Kelley 1985]. Untersuchte Beispiel-Objekte für die Suche nach Greifmerkmalen sind unter anderem Bremstrommeln [Belforte 1986], Rohrabdeckungen [Birk 1981] und Recycling-Kunststoffteile [Rubarth 1996].

Objektlageerkennung mit Wissensbasen

Bei Verfahren, die Wissensbasen mit entsprechenden Objektbeschreibungen nutzen, können Verfahren unterschieden werden, die lokale Merkmale verwenden, und solche Verfahren, die globale Merkmale verwenden.

Beispiele lokaler Merkmale sind zumeist geometrische Informationen wie Kreise [Kawasaki 1999, Lehmann 2000], Kantenzüge [Bahnu 1984], Ebenen [Onda 1995, Kristensen 2001] oder regelgeometrische Körper wie Zylinder oder Quader [Stotz 2008]. Der Einsatz signifikanter Eckpunkte von 2-D-Ansichten eines Objekts, die zur Laufzeit miteinander kombiniert werden, wurde von [Bebis 2002] untersucht, um so die Erkennung einer 3-D-Objektlage aus 2-D-Grauwertbildern zu ermöglichen.

Abstraktere Merkmale finden sich bei [Lowe 2004], der für sein SIFT-Verfahren (Scale Invariant Feature Transform) lokale rotations- und skalierungsinvariante Merkmale verwendet, die durch verschiedene Gaußfilterungen entstehen, um in Kameradaten Objekte zu lokalisieren. [Kirkegaard 2006] benutzt „harmonic shapes contexts“ zur Oberflächendarstellung von Objekten. Mehrere Arbeiten haben außerdem gezeigt, dass durch die Kombination mehrerer lokaler Merkmale die Zuverlässigkeit der Objektlageerkennung erhöht werden kann [Röhrdanz 1998, Ohba 1996].

Eine Schwierigkeit beim Einsatz lokaler Merkmale ist die exponentielle Komplexität bei der Zuordnung der Modellmerkmale zu den extrahierten Merkmalen in den Sensordaten [Müller 2001]. Um diese Komplexität zu reduzieren, wurden mehrere Methoden wie beispielsweise Merkmalsgraphen mit Baumsuche [Faugeras 1993], die Hough-Transformation [Stockmann

1988] oder das geometrische Hashing [Wolfson 1990, Grewe 1995, Baraquet 1997] entwickelt. Einen anderen Weg, um die Komplexität der Merkmalszuordnungen zu reduzieren, beschreiben [Hetzel 2001], [Wahl 2003] und [Siggelkow 2002], indem keine Zuordnung einzelner geometrischer Merkmale, die je nach Objektlage in ihrem Aussehen und ihrer Sichtbarkeit variieren können, durchgeführt wird. Stattdessen werden für die Identifikation des Objekts jeweils die gleichen immer vorhandenen Merkmale mit unterschiedlichen Ausprägungen eingesetzt. Als geeignete Merkmale dienen hierzu beispielsweise Tiefenhistogramm und Normalenvektorhistogramm in Kombination mit einem Krümmungsindex der Objektoberflächen. Erfolgreiche Anwendungen von Histogrammen sind beispielsweise die Objektidentifikation separierter Objekte [Hetzel 2001] und die automatische Bildanalyse und Indizierung großer Datenmengen [Siggelkow 2002].

Für globale Merkmale ist die Zuordnung von Modellmerkmalen zu den extrahierten Merkmalen in den Sensordaten deutlich einfacher. Nachteil der globalen Merkmale wiederum ist, dass Teilverdeckungen eines Objekts zum Scheitern der Objektlageerkennung führen können. Typische Vertreter globaler Merkmale sind der geometrische Schwerpunkt, die Orientierung der Hauptachse oder der Umfang [Müller 2001]. Weitere Methodenbeispiele sind der Einsatz von Constructive Solid Geometry [Biedermann 1987], die Zerlegung eines Objekts in Primitivobjekte, die durch boolesche Operatoren verknüpft und in einer Baumstruktur gespeichert werden, und Aspektgraphen [Koenderink 1979] mit der Objektbeschreibung durch eine Ansammlung zweidimensionaler Ansichten des Objekts aus unterschiedlichen Perspektiven. Zusätzlich kommen Musterbäume [Ettel 2002], volumenbasierte Octree-Repräsentation [Chen 1988] oder Extended Gaussian Image (EGI) Repräsentation [Little 1985, Horn 1984], die Winkelhistogramme von Oberflächenpunkten für konvexe Objekte nutzen, zum Einsatz. Ein weiteres typisches Beispiel eines globalen Merkmals, das für die 2-D-Objektlageerkennung eingesetzt wird, ist das Abspeichern eines Grauwertbilds des Objekts (als Pattern bezeichnet) mit anschließender Grauwertkorrelation [Matrox 2005].

Die „Eigen Window“ Methode nutzt eine Vielzahl an Teilansichten eines Objekts, die im Eigenraum abgebildet werden, für die Objektlageerkennung [Ohba 1997]. Ein weiterer Ansatz ist die Kombination von „Eigenimages“ mit der Auswertung von Tiefendaten, die zunächst mithilfe der Kantensuche segmentiert werden, um Objekte zu lokalisieren [Ghita 2003].

Objektlageerkennung mit generischen Objektmodellen

Als generische Objektmodelle werden beispielsweise Generalisierte Zylinder [Binford 1971] oder Superquadranten [Boughorbel 2003, Zhang 2003] eingesetzt. Hauptnachteil der Verfahren, die generische Objektmodelle einsetzen, ist, dass sie jeweils nur für bestimmte Objekte mit regelmäßiger Form anwendbar sind [Müller 2001]. Ein allgemeinerer Ansatz, der für beliebige polygonale Objekte eingesetzt werden kann, wurde in [Müller 1992] realisiert.

Objektlageerkennung mit heuristischen Verfahren

Ebenso wie beim Einsatz generischer Objektmodelle wird bei heuristischen Verfahren versucht, möglichst wenig spezielles Vorwissen über das gesuchte Objekt zu verwenden, um schnell ohne Erstellung einer Wissensbasis an unterschiedliche Objekte anpassbar zu sein. Ziel ist eine geeignete Segmentierung in Teilobjekte, um daraus Rückschlüsse auf das aktuell vorliegende Objekt zu ziehen. Hauptvorteil von heuristischen Verfahren ist die einfache Realisierbarkeit, da die hauptsächlich verwendete Eigenschaft die Konvexität von Objekten ist [Müller 2001]. Beispiele für eingesetzte Heuristiken, um zu entscheiden, welche konvexen Teilobjekte zum gleichen Objekt gehören, sind konvexe Übergänge [Trobina 1995] oder Höhengänge zwischen Teilobjekten [Wolf 1999]. Weitere Kriterien zur Segmentierung von Objektbereichen sind Konturanalysen [Beyer 1996] und die mittlere und Gaußsche Krümmung [Böhm 2005]. Hauptnachteil der heuristischen Verfahren ist, dass sie hauptsächlich auf konvexe Objekte beschränkt sind. Komplexe Objekte, die konkave Übergänge enthalten, werden bei der Erkennung nicht als zusammenhängendes Objekt erkannt, sondern in mehrere Teilobjekte zerlegt.

2.2.1.3 K3: Bereitstellung der Objekte

Die Bereitstellung der Objekte, die ein Roboter handhaben soll, kann in vielen Varianten erfolgen. In praktischen Anwendungen werden in der Hauptsache die geordnete oder ungeordnete, die statische oder dynamische sowie die sortenreine oder durchmischte Bereitstellung unterschieden. Um den Grad der Unordnung bei geordneter bzw. ungeordneter Bereitstellung zu beschreiben, wurde bereits der Begriff des Ordnungszustands (OZ) eingeführt. Ein typischer Vertreter der dynamischen Bereitstellung ist das Greifen eines Objekts vom Förderband.

2.2.1.4 K4: Nachfolgender Prozess

Neben der Bereitstellung sind auch Informationen über den nachfolgenden Prozess wichtig für das Robotersystem, das den Entnahmevergang durchführen soll. So können beispielsweise Genauigkeits- oder Orientierungsanforderungen bei der Ablage bereits Einfluss auf den eigentlichen Entnahmevergang haben.

2.2.1.5 K5: Geometrische Besonderheiten der Objekte

In diesem Kriterium wird die Vielzahl an Besonderheiten der zu handhabenden Objekte zusammengefasst. Insbesondere verformbare oder biegeschlaffe Objekte erschweren die Objektmodellierung. Ebenso können verhakende Objekte den anschließenden Entnahmevorgang erschweren.

2.2.1.6 Beispielanwendungen

Im Bereich der sensorgestützten Objektentnahme mit Robotern gibt es eine Vielzahl an umgesetzten Applikationen und Forschungsprojekten. Gerade Anwendungen, die noch ein Mindestmaß an Ordnung aufweisen (OG = 3 oder 2 und PG = 3 oder 2), können standardisiert mit 2-D-Bildverarbeitung gelöst werden, für die es inzwischen robuste und vielfach erprobte Verfahren gibt [Foitzik 2004]. Anbieter solcher Standardlösungen können beispielsweise [xpertgate 2010, xpertgate 2010a] oder [Elektronikpraxis 2006] entnommen werden. Sobald die Unordnung zunimmt oder wenn eine Adaptierbarkeit und Wiederverwendbarkeit für unterschiedliche Objekte in weiteren Anwendungen erreicht werden soll, scheint derzeit der Einsatz von Tiefensensoren, die geometrische Informationen der betrachteten Szene liefern, mit entsprechenden 3-D-Bildverarbeitungsalgorithmen der vielversprechendste Weg zu sein [Fritsch 2007]. Stellvertretend für Erkennungssysteme, die sich mit der Objektlageerkennung für die Handhabung von Objekten mittels Robotern in unterschiedlichen Ordnungszuständen beschäftigen, wird in Abbildung 2-3 eine repräsentative Auswahl an Beispielanwendungen B1 bis B8 vorgestellt und anschließend entsprechend der Kriterien K1 bis K5 eingeteilt.

B1: Entnahme teilgeordneter Gehäusedeckel

Anwendung / Beschreibung:

Lageerkennung von Gehäusedeckeln aus Gitterboxen mit Zwischenlagen für die Zuführung in eine Bearbeitungsmaschine auf Basis von 3-D-Daten eines Lichtschnittsensors, in denen die lokalen Merkmale Löcher und Kontur gesucht werden.

Firma / Literatur:

Fraunhofer IPA
[Gunnarson 2007, Ledermann 2005]



B2: Multiflex: Zuführen von Objekten in eine Bearbeitungsmaschine

Anwendung / Beschreibung:

Automatische Zuführung unsortierter Werkstücke in eine Bearbeitungsmaschine mit einem Palettenkipper, der die Werkstücke in die Maschine kippt. Die Werkstücke werden mit einem Förderband unter einem Kamera-system hindurch zum Roboter transportiert und dort gegriffen.

Firma / Literatur:

Svensk Industriautomation AB
[Svia 2005]



B3: Sortierung unterschiedlicher Objekte

Anwendung / Beschreibung:

Sortierung unterschiedlicher Objekte mit einem flexiblen Greifer und einer intelligenten Kamera. Die bekannten Objekte können beliebig ohne Überlappungen auf einer Ebene vor dem Roboter positioniert werden. Aufgrund einer Blobanalyse und Umfangsauswertung wird ein Objekt und dessen Lage ermittelt, gegriffen und in eine Ablage gelegt.

Firma / Literatur:

Fraunhofer IPA
[Leder mann 2007]



B4: Entnahme von Objekten und Ablage auf Formpalette für die Montage

Anwendung / Beschreibung:

Lageerkennung von Montageteilen auf Basis von abgespeicherten (Teil- und Gesamt-) Ansichten des Objekts mittels Korrelationsverfahren in Kameradaten. Zusätzlich wird kurz vor dem Griff mit einem Lichtschnittsensor der Normalenvektor der Objekt oberfläche bestimmt und der Greifer orthogonal zum Objekt ausgerichtet. Die Ablage erfolgt auf eine Formpalette.

Firma / Literatur:

FANUC Robotics
[Fanuc 2008, Fanuc 2005, Benes 2003]



B5: Vereinzelung von industriellen Objekten für die manuelle Montage

Anwendung / Beschreibung:

Vereinzelung von Montageteilen durch Auswertung von Stereokameradaten in Kombination mit einem projizierten Lichtgitter. Es werden ebene Flächen für den Einsatz eines Sauggreifers bestimmt. Nach der Entnahme wird das Objekt auf eine Ebene gelegt, um von dort für einen manuellen Montageprozess genommen zu werden.

Firma / Literatur:

Universität Graz
[Berger 2001, Berger 2000]



B6: Entnahme von Wasser-Pumpen-Teilen

Anwendung / Beschreibung:

Für die Entnahme von Wasser-Pumpen-Teilen werden auf Basis von zwei Kamerabildern lokale Merkmale wie ebene Flächen oder Löcher detektiert und auf dieser Basis die Objektlage bestimmt. Die Auswahl und das Einlernen der lokalen Merkmale erfolgt anhand von CAD-Daten der Objekte.

Firma / Literatur:

Scape Technologies
[Hardin 2008, Scape 2008]



B7: Entnahme von Kurbelwellen

Anwendung / Beschreibung:

Zur Entnahme von Kurbelwellen wird auf Basis eines Höhenprofils, das ein geschwenkter 2-D-Scanner liefert, der Roboter über der Kiste vorpositioniert. Dort erfolgt mit einer am Greifer montierten Kamera auf Basis von Teilansichten des Objekts eine Greifpunktbestimmung. Nach der Entnahme wird die tatsächliche Lage der Kurbelwelle im Greifer an einer separaten Kamerastation berechnet.

Firma / Literatur:

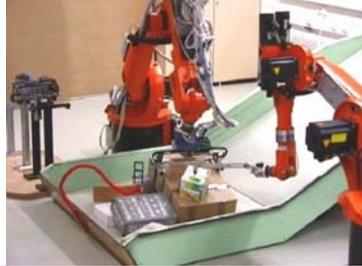
ThyssenKrupp Krause GmbH
[Bunke 2007, EADS 2004, SickInside 2002]



B8: Sortieren von Rücksendungen im Versandhandel

Anwendung / Beschreibung:

Im Versandhandel sollen Rücksendungen, die auf einer Rutsche an den Roboter angeliefert werden, entnommen werden. Die Geometrie der Objekte ist nicht bekannt. Objekte können sich auch verformen. Für die Entnahme eines Objekts werden Daten eines Lichtschnittsensors genutzt, in denen mittels Regionenwachstum ebene Flächen als Greifmerkmale gesucht werden.



Firma / Literatur:

Universität Karlsruhe
[Müller 2001]

Abbildung 2-3: repräsentative Auswahl an Beispielanwendungen zur Handhabung von Objekten mittels Robotern, die Objektlageerkennungssysteme einsetzen

Diese Erkennungssysteme können auf Basis der eingeführten Kriterien wie in Abbildung 2-4 dargestellt klassifiziert werden.

	K1: Sensordaten	K2: Objektinformation	K3: Bereitstellung			K4: Folgeprozess	K5: Geom. Besonderheit des Objekts
			OZ	Dynamik	Sortierung		
B1: Gehäusedeckel	3-D (LS)	lok. M.	(1/1)	stat.	sort.	Bearb.	keine
B2: Multiflex	2-D (Farbe)	k.A.	(1/1)	dyn.	sort.	Bearb.	keine
B3: Sortierung	2-D (Grau)	glob. M. WB	(0/1)	stat.	gem. bek.	Puffer	keine
B4: Formpalette	2-D/3-D (Grau/LS)	lok. M. WB	(0/0)	stat.	sort.	Formpalette	keine
B5: manuelle Montage	3-D (Stereo)	Greifmerkmale	(0/0)	stat.	sort.	manuelle Montage	ebene Fläche
B6: Wasser-Pumpen	2-D (Grau)	lok. M. WB	(0/0)	stat.	sort.	Transportband.	keine
B7: Kurbelwellen	2-D/3-D (Grau/GL)	lok. M. WB	(0/0)	stat.	sort.	Bearbeitungsmaschine	keine
B8: Versandhandel	3-D (LS)	Greifmerkmale	(0/0)	stat.	gem. unbek.	Puffer	verformbar

Legende:
 K1: Sensordaten: LS = Lichtschnittsensor; Farbe = Farbkamera; Grau = Grauwertkamera; Stereo = Stereokamerasystem mit Lichtgitter; GL = geschwenkter 2-D-Laserscanner
 K2: Objektinformation: lok. M. = lokale Merkmale; glob. M. = globale Merkmale; WB = Wissensbasis; k.A. = keine Angabe des Entwicklers
 K3: Bereitstellung: OZ = Ordnungszustand; stat. = statische Anlieferung; dyn. = dynamische Anlieferung auf Förderband; sort. = sortenreine Anlieferung bekannter Objekte; gem. bek. = gemischte Anlieferung bekannter Objekte; gem. unbek. = gemischte Anlieferung unbekannter Objekte

Abbildung 2-4: Klassifizierung des Standes der Technik und der Forschung Systemen zur Objektlageerkennung (Zusammenstellung auf Basis der o.g. Anwendungen und Literatur)

2.2.2 Optimierungsverfahren für die Objektlageerkennung

2.2.2.1 Traditionelle Optimierungsverfahren

Traditionelle Optimierungsverfahren benutzen exakte Methoden, um eine Lösung zu finden. Ist ein Problem lösbar, dann sollte der Algorithmus auch die global beste Lösung finden. Ein Beispiel hierfür ist die Brute-Force-Methode („Methode der rohen Gewalt“). Dieser natürliche und einfache Ansatz, um zu einer algorithmischen Lösung eines Problems zu kommen, besteht darin, alle potenziellen Lösungen durchzuprobieren, bis die richtige gefunden ist. Es ist klar, dass mit wachsendem Lösungsraum auch der Aufwand zur Lösung des Problems steigt. Weitere Verfahren wie beispielsweise die Divide-and-conquer-Methode („Teile und herrsche“), die ein Problem in kleinere Teilprobleme zerlegt, die lineare Optimierung, die typischerweise im Operations Research eingesetzt wird, oder die dynamische Programmierung, die dann erfolgreich eingesetzt werden kann, wenn das Problem aus vielen gleichartigen Teilproblemen besteht, können der Fachliteratur entnommen werden [Schrijver 1998, Bellman 2003, Fogel 2004].

2.2.2.2 Stochastische Algorithmen

Stochastische Algorithmen werden üblicherweise verwendet, um Lösungen nahe am Optimum zu finden. Sie beruhen auf der Annahme, dass geeignete Lösungen im Lösungsraum nahe beieinander liegen. Es konnte gezeigt werden, dass dies für die meisten Probleme der realen Welt zutrifft [Lovberg 2002, Spall 2003, Omran 2005]. Da das Ziel stochastischer Algorithmen darin liegt, annähernd-optimale Lösungen zu finden, besteht allerdings die Gefahr, das globale Optimum zu verfehlen.

Ein Beispiel ist der Hill-Climbing Algorithmus, in dem ein Individuum von einer zufällig gewählten Anfangsposition aus startet, um sein Umfeld im Lösungsraum nach besseren Lösungen zu durchsuchen. Findet es eine besser bewertete Position wechselt es an diese. Dieser Vorgang wird solange wiederholt, bis keine Verbesserung mehr eintritt, also ein „Gipfel“ im Umfeld des Individuums erklommen ist. Dieser „Gipfel“ kann auch ein lokales Optimum im Lösungsraum darstellen. Die Wahrscheinlichkeit das globale Optimum zu finden, kann sich durch mehrmalige Wiederholung des Vorgangs mit unterschiedlichen Startpositionen erhöhen. Ähnlich arbeitende Algorithmen sind das Simulated Annealing [Wegener 2005] oder die Tabu-Suche [Glover 1986].

2.2.2.3 Genetische Algorithmen

Die genetischen Algorithmen (GAs) sind eine Form der evolutionären Algorithmen, diese simulieren natürliche Selektion und Evolution nach biologischem Vorbild. Die erstaunlichste Eigenschaft der Evolution ist die relative Einfachheit ihrer Vorgehensweise und das Zusammenwirken der verschiedenen Steuerungsmechanismen. Im Gegensatz zu anderen

Optimierungsverfahren verwenden sie eine ganze Population von Lösungsvorschlägen (Individuen) anstatt einer einzelnen Lösung, die es zu verbessern gilt [Schöneburg 1993].

Für die GAs wird eine Startpopulation von Individuen initialisiert. Jedes Individuum repräsentiert eine potenzielle Lösung, dessen Qualität durch eine Bewertungsfunktion, der Fitnessfunktion, bestimmt wird. Durch Selektion werden bei jeder Iteration „Eltern“-Individuen nach vorgegebenen Kriterien ausgesucht, welche den Grundstock der nachfolgenden Generation bilden. Diese werden durch Rekombination (crossover) oder Mutation verändert. Das Verfahren wird bis zur Konvergenz wiederholt und die beste gefundene Lösung wird als Lösung nahe dem Optimum angesehen.

Ein bekannter Rekombinations-Operator ist das „Single-Point-Crossover“. Hierbei werden zwei Eltern an einer zufällig gewählten Bruchstelle getrennt und die linke oder rechte Seite beider Eltern wird getauscht, dadurch entstehen zwei neue Abkömmlinge. Bei der Mutation werden zufällig einzelne Werte der Individuen verändert. Ziel dieser Operation ist es, neue Zonen des Lösungsraums zu erkunden und Diversität zur Population hinzuzufügen, um der Gefahr entgegenzuwirken, in einem lokalen Optimum zu konvergieren.

2.2.2.4 Partikel-Schwarm-Optimierung (PSO)

Die Partikel-Schwarm-Optimierung (PSO) ist ebenso wie die genetischen Algorithmen populationsbasiert. Eine Population besteht aus einem Schwarm von Individuen, nachfolgend Partikel genannt, die zu Beginn im Lösungsraum verteilt werden. Vorbild der PSO war das soziale Verhalten von Vogelschwärmen [Kennedy 2001]. Im Gegensatz zu den GAs wird hierbei auch die Historie des Individuums berücksichtigt und orientiert sich hierbei nicht am „survival of the fittest“-Prinzip, sondern an der Imitation des sozialen Verhaltens [Omran 2005].

Das soziale Verhalten innerhalb von Gruppen lässt sich in zwei Ebenen unterteilen. Die höhere Ebene beinhaltet die Gruppendynamik – die Entstehung von Verhaltensmustern und die Fähigkeit, gemeinsam Probleme zu lösen. Die niedrige Ebene beschränkt sich auf das universelle Verhalten aller Lebewesen. Diese Ebene bildet die Grundlage des Verhaltens des Schwarms. Die niedrige Ebene beinhaltet dabei drei Grundschrte: Bewertung von äußeren Einflüssen, Vergleich mit anderen und Imitation anderer [Kennedy 2001]. Für jeden dieser drei Schritte können sowohl Aspekte in der Biologie und Soziologie als auch der menschlichen Gesellschaft gefunden werden. Eine kurze Übersicht dieser Aspekte sowie die jeweiligen Parallelen in der PSO sind in Abbildung 2-5 dargestellt.

	Biologie/Psychologie	Soziologie	Partikel-Schwarm
Bewertung (evaluate)	Selbst kleinste Lebewesen (Bakterien) sind in der Lage, äußere Einflüsse zu bewerten	Die Fähigkeit zur Bewertung wird als Grundlage jeglicher Lernfähigkeit angesehen. [Kennedy 2001]	Bewertung äußerer Einflüsse und des inneren Zustands durch eine Fitnessfunktion
Vergleich (compare)	Messen mit anderen führt zur Motivation zu lernen und sich zu verändern	Gruppenprozesse und Vorgänge entspringen dem Bewertungsbedürfnis, dazu vergleicht sich das Individuum mit anderen. [Festinger 1954]	Partikel können die eigene Qualität mit der anderer Partikel vergleichen
Nachahmung (imitate)	Nachahmen ist das Grundprinzip des Lernens. „Monkey see, monkey do“	Menschen und Tiere besitzen eine angeborene Neigung, das Verhalten anderer zu imitieren. [Morgan 1896]	Gut bewertete Partikel haben einen Einfluss auf die Bewegung der anderen Partikel

Abbildung 2-5: Parallelen zwischen Biologie/Psychologie, Soziologie und einem Partikel-Schwarm

Die Partikel im Schwarm

Bei der Suche nach dem Optimum eines Problems stellt jeder Partikel des Schwarms eine potentielle Lösung des Problems dar, deren Qualität durch eine problemspezifische Fitnessfunktion f bewertet wird. Die Partikel werden zu Beginn im Lösungsraum verteilt, Startparameter wie Position und Änderungsvektor werden mit Zufallswerten initialisiert. Die Bewegung, d.h. die Änderung der Position eines Partikels im Lösungsraum zwischen zwei Iterationsschritten, wird durch seine eigene, bisher beste Position und die Position des besten Partikels in seiner betrachteten Nachbarschaft beeinflusst.

In jeder Iteration werden deshalb folgende Schritte durchgeführt, wobei bei der Formulierung der Formeln von einem Minimierungsproblem ausgegangen wird, d.h. je kleiner der Fitnesswert, desto besser die Bewertung des Partikels.

1. Bewertung (evaluate)

Die aktuelle Position $p_i(t)$ des Partikels p_i im Lösungsraum zur Zeit t wird durch die Fitnessfunktion f bewertet. Ist diese Bewertung besser als alle vorherigen, so wird $p_i(t)$ als neue, beste bisher gefundene Lösung $pb_i(t)$ des Partikels gespeichert.

$$f(p_i(t)) \leq f(pb_i(t)) \Rightarrow pb_i(t) = p_i(t) \quad \text{Gleichung 2-2}$$

2. Vergleich (compare)

Jeder Partikel betrachtet die Bewertungen der Partikel in seiner Nachbarschaft, die beste in der Nachbarschaft vorhandene Position $p_n(t)$ wird als beste globale Lösung pg_i gespeichert.

$$f(p_n(t)) \leq f(p_i(t)) \forall p_i(t) \Rightarrow pg_i(t) = p_n(t) \quad \text{Gleichung 2-3}$$

3. Nachahmung (imitate)

Die Positionsänderung $v_i(t + 1)$ und damit die neue Position $p_i(t + 1)$ eines Partikels erfolgt in Abhängigkeit des Änderungsvektors der letzten Iteration $v_i(t)$, der aktuellen Position $p_i(t)$ und den bisher besten Positionen $pb_i(t)$ und $pg_i(t)$ sowie den zusätzlichen Faktoren φ_1 , φ_2 und α_e :

$$v_i(t + 1) = \alpha_e \cdot v_i(t) + \varphi_1(pb_i(t) - p_i(t)) + \varphi_2(pg_i(t) - p_i(t)) \quad \text{Gleichung 2-4}$$

$$p_i(t + 1) = p_i(t) + v_i(t + 1) \quad \text{Gleichung 2-5}$$

α_e wird als Explorationsfaktor bezeichnet, der den Einfluss der Geschwindigkeit des vorhergehenden Iterationsschritts kontrolliert. Ein großer Wert begünstigt die Exploration (das Erkunden des Lösungsraums), während ein kleiner Wert die Bindung zur aktuellen Position stärkt. φ_1 bzw. φ_2 geben den Einfluss der eigenen Erfahrung (kognitive Komponente) bzw. die Erfahrung der Nachbarn (soziale Komponente) wieder. Für φ_1 , φ_2 gilt, dass die Summe von φ_1 und φ_2 einen konstanten Wert $\varphi_G = \varphi_1 + \varphi_2$ ergeben muss. Diese Bedingung hat laut [Kennedy 2001] zur Folge, dass der Partikel p_i eine Kreisbahn um eine Position p_m beschreibt, die sich aus einem gewichteten Mittel der zwei besten Positionen $pb_i(t)$ und $pg_i(t)$ zusammensetzt:

$$p_m = \frac{\varphi_1 \cdot pb_i(t) + \varphi_2 \cdot pg_i(t)}{\varphi_G} \quad \text{Gleichung 2-6}$$

Um die Flugbahn des Partikels zu beschränken, und somit zu verhindern, dass der Partikel ins Unendliche „abdriftet“, wird eine Maximalgeschwindigkeit v^{Max} als Begrenzung des Änderungsvektors eingeführt:

$$v_i(t + 1) \geq v^{Max} \Rightarrow v_i(t + 1) = v^{Max} \quad \text{Gleichung 2-7}$$

$$v_i(t + 1) \leq -v^{Max} \Rightarrow v_i(t + 1) = -v^{Max} \quad \text{Gleichung 2-8}$$

Abbildung 2-6 stellt den Ablauf einer Suche als Pseudocode dar. Nach der Initialisierung erfolgt die iterative Suche durch Evaluierung mit der Fitnessfunktion, Vergleich mit den Nachbarpartikeln und Berechnung der neuen Position jedes Partikels.

```
procedure PSO;  
für jedes Partikel  $p_i$  do begin //initialisiere die Position jedes Partikels (INIT)  
    wähle zufällig  $p_i(0)$ ;  $v_i(0) = 0$ ; //zufällig im Suchraum  
end;  
wiederhole //bewege die Partikel des Schwarms  
    für jedes Partikel  $p_i$  do begin  
         $y := f(p_i(t))$ ; //berechne die Fitnessfunktion (EVALUATE)  
        if  $y \geq f(p_b(t))$  then  $p_b(t) := p_i(t)$ ; //speichere die eigene beste Position  
        if  $y \geq f(p_g(t))$  then  $p_g(t) := p_i(t)$ ; //speichere die beste Position der Nachbarn (COMPARE)  
    end;  
    für jedes Partikel  $p_i$  do begin  
         $v_i(t+1) := \alpha \cdot v_i(t) + \phi_1(p_b(t) - p_i(t)) + \phi_2(p_g(t) - p_i(t))$ ; //aktualisiere die Geschwindigkeit  
         $p_i(t+1) := p_i(t) + v_i(t+1)$ ; //berechne die neue Position des Partikels (IMITATE)  
    end;  
bis ein Abbruch-Kriterium (z.B. maximale Anzahl an Iterationen) erfüllt ist
```

Abbildung 2-6: Pseudo-Code der Partikel-Schwarm-Optimierung

Nachbarschaftsbeziehungen innerhalb des Schwarms

Wird für die Berechnung einer Partikel-Bewegung der gesamte Schwarm als Nachbarschaft betrachtet, so wird dies als „gbest“ (global-best) Modell bezeichnet. Kleinere Nachbarschaften, die als „lbest“ (local-best) Modell bezeichnet werden, beziehen eine geringere Anzahl k direkter Nachbarn in die Bestimmung der besten Position der Nachbarn mit ein. Hybride Modelle, die im Laufe des Suchvorgangs von einem lbest Modell zu einem gbest Modell wachsen, sind möglich und wurden beispielsweise in [Omran 2005] zur Segmentierung von Tiefendaten betrachtet.

Die Nachbarschaft bleibt ansonsten, unabhängig von der Bewegung der Partikel, während der kompletten Suche erhalten. Eine große Nachbarschaft, die einen Partikel mit sehr vielen Nachbarn direkt oder über eine geringe Anzahl von dazwischenliegenden Partikeln verbindet, erhöht die Konvergenzgeschwindigkeit des Schwarms, da sich die Information über die Lage eines Optimums sehr schnell verbreitet. Einen ähnlichen Einfluss auf die Verbreitung der Information hat eine gbest Nachbarschaft, bei der alle Partikel eines Schwarms benachbart sind. Diese schnelle Konvergenz kann dazu führen, dass der Schwarm in einem Suboptimum konvergiert.

Neben der Größe der Nachbarschaft hat auch die Topologie der Nachbarschaftsbeziehungen Einfluss auf die Geschwindigkeit, mit der sich Informationen über ein Optimum innerhalb des Schwarms ausbreiten. Typische Topologien, die hierbei zum Einsatz kommen können, sind die Ring-Struktur, die Gitter-Struktur oder die Kreuz-Struktur (siehe Abbildung 2-7). Die am häufigsten untersuchte und erfolgreich eingesetzte Struktur ist die Ringstruktur. Dies liegt

insbesondere daran, dass hier die Größe der lbest Nachbarschaften die meisten Variationsmöglichkeiten bietet.

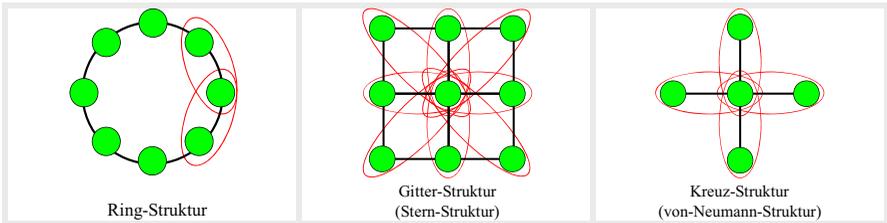


Abbildung 2-7: Nachbarschaftsbeziehungen der Partikel

Der Constriction Faktor χ

Um eine Konvergenz des Schwarms herbeizuführen, dient der Constriction Faktor χ . Dieser Faktor bewirkt nach [Kennedy 2001] eine langsame Konvergenz des Schwarms und gleichzeitig die Möglichkeit, den Lösungsraum weitläufig zu erkunden.

Der Constriction Faktor χ ist nach [Kennedy 2001] wie folgt definiert:

$$\chi = \frac{2\kappa}{\left|2 - \varphi_G - \sqrt{\varphi_G^2 - 4\varphi_G}\right|}, \varphi_G > 4 \quad \text{Gleichung 2-9}$$

Bei Anwendung des Constriction Faktor χ ändert sich Gleichung 2-4 zu:

$$v_i(t+1) = \chi \cdot \left(v_i(t) + \varphi_1(pb_i(t) - p_i(t)) + \varphi_2(pg_i(t) - p_i(t))\right) \quad \text{Gleichung 2-10}$$

[Kennedy 2001] empfiehlt neben dem Einsatz des Constriction Faktors χ zusätzlich, dass $\kappa \in [0,1]$ gelten sollte. Die Gleichung 2-9 ist nur für Werte von $\varphi_G > 4$ definiert. Als geeignete Werte werden $\varphi_G = 4,1$ und $\kappa = 1$ empfohlen, um eine Konvergenz zu sichern.

Nachteile der PSO

Wie alle stochastischen Verfahren hat die PSO zwei große Nachteile [Omran 2005, Lovberg 2002]:

- die Gefahr der zu frühen Konvergenz des Schwarms in lokale Optima
- die Performance der Optimierung ist stark vom betrachteten Problem abhängig

Der erste Nachteil, die zu frühe Konvergenz, ist nach [Angline 1998] bei den PSO weniger stark ausgeprägt als bei anderen evolutionären Algorithmen. Gerade bei stark multi-modalen Problemen – d.h. bei Problemen mit einer Vielzahl lokaler Optima – kommt dieser Vorteil zum

Tragen. Ein ähnliches Verhalten der frühen Konvergenz zeigen allerdings auch PSO, die eine gbest-Nachbarschaft nutzen. Hauptgrund für diese Problematik ist, dass Partikel mit gbest-Nachbarschaft gegen einen Punkt konvergieren, der zwischen der global besten Position und der eigenen besten Position liegt. Beweise, dass dieser Punkt noch nicht einmal ein lokales Optimum sein muss, finden sich in [van den Bergh 2002]. Auch der Verlust an Diversität innerhalb des Schwarms durch den schnellen Informationsfluss bei gbest Nachbarschaften verstärkt die Problematik. Neben einer Anzahl möglicher Modifikationen der Partikel-Schwarm-Optimierung [Omran 2005] können hier die bereits erwähnten lbest- oder hybriden Nachbarschaftsstrukturen Abhilfe leisten.

Um dem zweiten Nachteil, der Abhängigkeit der Performance vom betrachteten Problem, zu begegnen, gibt es in der Literatur mehrere Ansätze, die beispielsweise den Einsatz selbstadaptiver Parameter erfolgreich untersucht haben [Lovberg 2002, Shi 2001].

Verbesserungsmöglichkeiten für PSO

Von den beiden zuvor genannten Nachteilen ist insbesondere die Vermeidung der zu frühen Konvergenz vielfach untersucht worden. [Omran 2005], der die PSO erfolgreich in der Sensordatenverarbeitung zur Segmentierung von Tiefendaten eingesetzt hat, gibt einen Überblick über einige der bisher untersuchten Möglichkeiten. Zu nennen sind hier beispielsweise der Einsatz von Mutation oder die Erweiterung auf die parallele Verwendung mehrerer Schwärme (Multi-Schwarm). Eine automatische Neupositionierung von Partikeln, wenn sie zu nahe zueinander kommen, stellt [Mohais 2005] vor. So genannte dissipative PSO führen eine negative Entropie in den Schwarm ein, die zu Interaktionen zwischen den Partikeln führt, um die Konvergenz zu verbessern [Xie 2002, Shen 2007]. Eine weitere Möglichkeit bietet die Kombination mit anderen Optimierungsverfahren wie Simulated Annealing [Dong 2006, Cong 2008]. Ein Vergleich einiger Varianten durch [Montes de Oca 2006] zeigt deren Stärken, die dennoch alle stark von der gewählten Nachbarschaft abhängen. Zusammenfassend lässt sich festhalten, dass es eine Vielzahl erfolgreicher Erweiterungen der Standard PSO gibt, die herangezogen werden können, wenn für die Standard PSO Probleme durch zu frühe Konvergenz für ein Optimierungsproblem entstehen.

Vergleich PSO mit GA

Die Partikel-Schwarm-Optimierung (PSO) und die genetischen Algorithmen (GA) gleichen sich in einigen Grundlagen. Beide Verfahren basieren auf einer Population einfacher Individuen, die jeweils einen Lösungskandidaten im Lösungsraum darstellen. Im Gegensatz zur PSO bewegen sich die Individuen bei den GA nicht kontinuierlich durch den Lösungsraum. Vielmehr werden durch Selektion, Kreuzung und Mutation neue Individuen erzeugt, die einen neuen Platz im Lösungsraum einnehmen – diesen quasi anspringen.

Einer der Hauptvorteile der PSO gegenüber GA lässt sich bei der hier vorliegenden Aufgabe der Objektlageerkennung darstellen: Im Lösungsraum können sich mehr als eine richtige Lösung befinden. Bei der Verwendung von GA birgt dies die Gefahr von Fehlvererbungen, d.h. bei zwei Individuen, die sehr nahe an unterschiedlichen Minima liegen und als Elternpaar ausgewählt werden, könnte das daraus entstehende Individuum weiter entfernt von beiden Minima liegen als die Eltern. Ein Rücksprung auf die eigentlich besseren Lösungen der Eltern kann nicht mehr erfolgen. Hierdurch würde eine Verschlechterung der Lösung entstehen. Im Gegensatz hierzu arbeitet die PSO so, dass beide qualitativ hochwertigen Partikel ihre Nachbarn anziehen, die guten Lösungen selbst bleiben jedoch für die folgenden Iterationsschritte weiterhin erhalten.

Insgesamt gibt es eine Vielzahl an Untersuchungen, die sich mit dem Vergleich von PSO und GA beschäftigen. Für viele Anwendungen zeigt sich, dass die PSO besser arbeitet als die GA. Sie konvergieren zumeist schneller und robuster als die GA [Kennedy 1998, Veeramachaneni 2003, Omran 2005]. Ebenso wurde gezeigt, dass PSO auch zum Trainieren von Neuronalen Netzen besser geeignet ist, als GA oder andere Trainings-Algorithmen [Eberhart 1998].

Nach [Shi 1998] reagiert die Leistung der PSO außerdem unempfindlich auf Änderungen der Schwarmgröße (natürlich darf die Größe nicht zu klein werden). Die Folge ist, dass eine PSO mit kleiner Schwarmgröße ähnlich leistungsfähig ist wie eine GA mit größerer Population [Lovberg 2002].

Parametrierung von PSO

Für die nachfolgende Entwicklung und Evaluierung eines PSO-basierten Verfahrens ist es entscheidend, einschätzen zu können, welche Parameter der PSO welchen Einfluss auf das Verhalten der Leistungsfähigkeit des Schwarms haben. Einen guten Überblick gibt [Carlisle 2001]. Dort wird ein „off-the-shelf-PSO“ aufgrund von Ergebnissen aus der Fachliteratur und eigener Versuche vorgeschlagen, der als Grundlage für die Planung der späteren Untersuchungen innerhalb dieser Arbeit genutzt wird.

Als Basis der Untersuchungen von [Carlisle 2001], die die Möglichkeiten der PSO zeigen, dienen mehrere der typischen Funktionen, an denen üblicherweise Optimierungsverfahren getestet werden: die Sphere-Funktion (De Jong's F1), die Rosenbrock-Funktion, die generalisierte Rastrigin-Funktion, die generalisierte Griewank-Funktion und Schaffer's F6-Funktion. Die Formulierungen der Funktionen können Abbildung 2-8 entnommen werden.

Funktion	Formel	
Sphere (De Jong F1) n-dimensional: $\mathbf{x} = (x_1, \dots, x_n)^T$	$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$	Gleichung 2-11
Rosenbrock n-dimensional: $\mathbf{x} = (x_1, \dots, x_n)^T$	$f(\mathbf{x}) = \sum_{i=1}^n (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	Gleichung 2-12
Rastrigrin n-dimensional: $\mathbf{x} = (x_1, \dots, x_n)^T$	$f(\mathbf{x}) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	Gleichung 2-13
Griewank n-dimensional: $\mathbf{x} = (x_1, \dots, x_n)^T$	$f(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	Gleichung 2-14
Schaffer F6 2-dimensional: $\mathbf{x} = (x_1, x_2)^T$	$f(\mathbf{x}) = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{(1.0 + 0.001(x_1^2 + x_2^2))^2}$	Gleichung 2-15

Abbildung 2-8: Funktionen zur Bestimmung eines „off-the-shelf-PSO“ nach [Carlisle 2001]

Die Rosenbrock-Funktion besitzt ihr Minimum innerhalb eines langen, engen, parabolisch verlaufenden Grabens. Das Besondere an dieser Funktion ist, dass der Graben einfach gefunden werden kann – zum globalen Minimum innerhalb des Grabens zu konvergieren, ist allerdings schwierig [Törn 1989]. Die Rastrigrin- und die Griewank-Funktion werden zur Überprüfung von Optimierungsverfahren herangezogen, da sie einen großen Lösungsraum mit einer Vielzahl an Minima besitzen. Das Besondere an der Schaffer F6-Funktion ist, dass sie unendlich viele lokale Minima in der Nähe des globalen Optimums besitzt. Abbildung 2-9 zeigt Darstellungen der drei Testfunktionen Rosenbrock-, Rastrigrin- und Griewank-Funktion mit der Dimension $n = 2$.

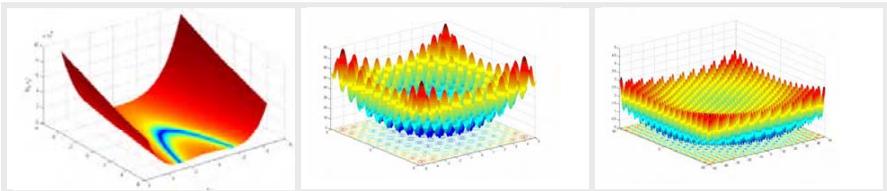


Abbildung 2-9: Darstellungen der Rosenbrock-, Rastrigrin- und Griewank-Funktion (Dimension $n = 2$)

Die wichtigsten Ergebnisse der Untersuchungen zum Einfluss der einzelnen Parameter auf die Leistungsfähigkeit der PSO von [Carlisle 2001], die als Grundlage für die Planung der Tests des innerhalb dieser Arbeit entwickelten Erkennungssystems dienen, sind:

1. Schwarmgröße:

Die Aussage von [Shi 1999], dass PSO unempfindlich auf Änderungen der Schwarmgröße reagiert, konnte grundsätzlich bestätigt werden. Die benötigte Mindestzahl an Partikeln eines Schwarms ist für alle untersuchten Funktionen unterschiedlich, um

zuverlässig die korrekte Lösung zu finden. Allerdings konnte gezeigt werden, dass eine Schwarmgröße von 30 Partikeln für die untersuchten Funktionen eine geeignete Wahl ist, um eine ausreichend schnelle Konvergenz mit vertretbarem Rechenaufwand, d.h. Anzahl der benötigten Aufrufe der Fitnessfunktion, zu erreichen.

2. Nachbarschaftsgröße:

Entgegen den Aussagen von [Eberhart 1995], die eine kleine lokale Nachbarschaft höher bewerten, zieht [Carlisle 2001], den Schluss, dass eine globale Nachbarschaft die beste Wahl ist, wenn das Augenmerk auf das Verhältnis von Aufwand zu Leistung gelegt wird.

3. Verhältnis von kognitivem zu sozialem Verhalten:

Für das Verhältnis aus den Gewichtungsfaktoren φ_1 und φ_2 , das beeinflusst, wie stark ein Partikel durch die beste Position der Nachbarn bzw. durch die eigene beste Position beeinflusst wird, gibt [Kennedy 1998] eine Gleichgewichtung von $\varphi_1 = \varphi_2 = 2,05$ als beste Möglichkeit an. [Carlisle 2001] erzielte eine etwas bessere Leistung bei einer höheren Gewichtung des kognitiven Verhaltens mit $\varphi_1 = 2,8$ und $\varphi_2 = 1,3$. Interessant hierbei ist, dass bestätigt werden konnte, dass die besten Ergebnisse erzielt werden, wenn $\varphi_1 + \varphi_2 = 4,1$ gilt.

2.2.2.5 Ameisenkolonie-Optimierung

Ameisenkolonie-Optimierung (ACO, englisch Ant Colony Optimization) sind eine weitere Form der evolutionsbasierten stochastischen Optimierungsverfahren. Ebenso wie PSO ist das Verfahren durch Beobachtungen der Natur entstanden – in diesem Fall durch das Verhalten realer Ameisenkolonien bei der Futtersuche. Die Informationen zwischen den Ameisen werden hierbei indirekt durch das Absondern chemischer Substanzen, den Pheromonen, übertragen [Colorni 1992].

Eine Ameise kann Pheromon in unterschiedlicher Stärke ablegen, um die eigenen Wege zu markieren. Während der Suche bewegen sich isolierte Ameisen nahezu zufällig. Sobald eine Ameise jedoch auf eine Pheromonspur trifft, folgt sie ihr mit einer umso höheren Wahrscheinlichkeit, je stärker diese Pheromonspur ist. Die Ameise selbst kann die Pheromonspur wieder verstärken, um damit eine noch größere Attraktivität der Spur zu erreichen. So ergibt sich eine positive Rückkopplungsschleife: je mehr Ameisen einem Pfad folgen und dort ihre Pheromone ablegen, desto attraktiver wird dieser für die gesamte Kolonie und zieht immer mehr Ameisen an. Dieser indirekte Weg der Kommunikation durch Pheromone wird als Stigmergie bezeichnet [Dorigo 2004]. Eines der bekanntesten Probleme, auf das die ACO erfolgreich angewendet wurde, ist das Problem des Traveling-Salesman. Die Ameisenkolonie-Optimierung und deren Anwendungen liegen außerhalb des Fokus dieser Arbeit und werden deshalb nicht intensiver betrachtet.

2.3 Folgerungen aus dem Stand der Technik und der Forschung

Die Aufbereitung des Stands der Technik der Robotersysteme mit Objektlageerkennung für die Entnahme von Objekten hat gezeigt, dass der Großteil der Verfahren eine Wissensbasis zur Objektbeschreibung nutzt. Gerade neue Ansätze legen ihr Augenmerk darauf, flexibel an unterschiedliche Objekte anpassbar zu sein. Hier bietet der Einsatz lokaler Merkmale das größte Potenzial [Heintz 2007]. Zu diesen Ansätzen zählen beispielsweise die Histogramme, die von [Hetzel 2001] und [Siggelkow 2002] bereits erfolgreich zur Objektidentifikation separater Objekte bzw. zur automatischen Bildanalyse und zum Indizieren großer Datenmengen eingesetzt wurden.

Die Entwicklung von Tiefensensoren hat zusätzlich dazu geführt, dass neuere und zukünftige Verfahren verstärkt auf Tiefendaten setzen. Dies zeigt sich auch darin, dass derzeit mehrere Erkennungssysteme, die die Kombinationen von 2-D- und 3-D-Daten für die Bestimmung der Objektlage nutzen, verfügbar sind. Nach Einschätzungen von Experten ist die Verwendung von 3-D-Daten zur Bestimmung der Lage der Werkstücke der nächste logische Entwicklungsschritt [Fili 2008].

Die Anwendung der Partikel-Schwarm-Optimierung ist ein relativ junges Forschungsgebiet. Trotz aller Nachteile, die PSO in seiner Grundformulierung hat, besitzt sie doch bemerkenswerte Vorteile beispielsweise gegenüber den genetischen Algorithmen. So ist die größte Stärke der PSO, dass sie auch mit vielen lokalen Minima bzw. Maxima innerhalb des Lösungsraums zurecht kommt – was je nach Wahl der Objektbeschreibung gerade bei der Unterscheidung von Objektlagen häufig vorkommen kann. Insbesondere die Parameterwahl hat einen entscheidenden Einfluss auf das Konvergenzverhalten des Schwarms [Carlisle 2001]. Allerdings existieren aus dem Stand der Technik und der Forschung weder Erkenntnisse bzgl. des Einsatzes der PSO für die Objektlageerkennung noch Untersuchungen, wie ein Partikel zu modellieren ist und wie die Parameter zu wählen sind, um ein robustes System zur Objektlageerkennung zu erhalten.

Aus dem Stand der Technik und der Forschung wird klar, dass es noch kein einsetzbares PSO-basiertes System zur Objektlageerkennung in Tiefendaten für die Entnahme mit einem Roboter gibt. Aufgrund der positiven Eigenschaften der Partikel-Schwarm-Optimierung und der aktuellen Entwicklungen im Bereich der Tiefensensoren liegt daher die Entwicklung eines solchen Erkennungssystems in Kombination mit einer geeigneten Objektmodellierung, die die Stärken der PSO unterstützt, nahe.

3 Analyse potentieller Einsatzszenarien für ein System zur Objektlageerkennung und Ableitung von Anforderungen

3.1 Analyse des Gießerei- und Schmiedeumfelds

Entsprechend der Zielsetzung dieser Arbeit soll das System zur Objektlageerkennung für Objekte aus dem Gießerei- und Schmiedebereich eingesetzt werden. Hauptabnehmer für Gießerei- und Schmiedeprodukte sind vorrangig die Fahrzeugindustrie sowie der Maschinen- und Anlagenbau [Benneker 2004, DGV 2007]. Bezeichnend für diese Branchen sind insbesondere die hohe Anzahl unterschiedlicher Produkte und häufige Produktwechsel [Herbertz 2005]. Allein in einem einzelnen Schmiede-Unternehmen ist beispielsweise in einem Gewichtsbereich von 5 kg bis 7 kg ein Produktspektrum von 100 unterschiedlichen Produkten mit einer Losgröße von mehreren tausend Stück pro Jahr zu finden [Ledermann 2005a, SMErobot 2008]. Um die Erfordernisse des Gießerei- und Schmiedebereichs zu erfüllen und die Partikel-Schwarm-Optimierung nutzen zu können, ergeben sich folgende Anforderungen an das System zur Objektlageerkennung, die in den nächsten Abschnitten hergeleitet werden:

Durch die hohe Produktvielfalt und die häufigen Produktwechsel ergibt sich die Anforderung:

A1: Das System zur Objektlageerkennung muss an unterschiedliche Objekte anpassbar sein.

Um die immer komplexer werdenden Objektgeometrien korrekt auszulegen und um Analysen für stabile und reproduzierbare Fertigungsprozesse zu schaffen, werden heute standardmäßig Simulationen der Umformprozesse beim Schmieden anhand von 3-D-CAD-Daten im STL-Format (SurfaceTesselationLanguage – Beschreibung einer Oberfläche durch Dreiecke) durchgeführt [IMU 2006]. Aus diesem Grund liegen im Gießerei- und Schmiedebereich für nahezu alle Objekte, selbst wenn sie kistenweise angeliefert werden, 3-D-CAD-Daten vor:

A2: Das System zur Objektlageerkennung soll 3-D-CAD-Daten der Objekte als Eingangsdaten verwenden.

Da die Objekte nicht einzeln, sondern als Haufwerk in der Kiste angeliefert werden, soll das Erkennungssystem für die Entnahme durch den Roboter die Lage eines oben liegenden Objekts bestimmen:

A3: Das System zur Objektlageerkennung soll die Lage von einem oben liegenden Objekt eines Objekthaufens bestimmen.

Wichtig für die Verwendbarkeit des Systems zur Objektlageerkennung in realen Anwendungen ist, dass es nicht nur in Labortests funktioniert, sondern auch mit realen Sensordaten mit den auftretenden Messfehlern zurecht kommt. Um dabei auch an zukünftigen Entwicklungen von

Tiefensensoren partizipieren zu können, soll das Erkennungssystem dabei grundsätzlich mit unterschiedlichen Tiefensensoren arbeiten können:

A4: Das System zur Objektlageerkennung muss robust gegenüber Messfehlern in realen Sensordaten mit Rauschen sein.

A5: Das System zur Objektlageerkennung soll mit Tiefendaten von unterschiedlichen Sensoren arbeiten können.

Die Unabhängigkeit von speziellen Tiefensensoren soll dabei auch dadurch erreicht werden, dass zusätzlich gilt:

A6: Das Erkennungssystem soll keine speziellen Zusatzinformationen eines Tiefensensors wie beispielsweise Reflektivitätsinformationen benötigen.

Um eine möglichst einfache Anwendbarkeit in der Praxis ohne spezielles Expertenwissen zu ermöglichen, soll weiterhin folgende Anforderung erfüllt sein:

A7: Die Objektbeschreibung, die für die Lageerkennung benötigt wird, soll für neue Objekte automatisiert erweiterbar sein.

Zusätzlich zu diesen grundlegenden Anforderungen sollen im Folgenden drei potentielle Szenarien betrachtet werden, um weitere Anforderungen an das System zur Objektlageerkennung abzuleiten.

3.2 Analyse der Entnahme von Getriebewellen für eine Ultraschall-Prüfung zur automatischen Qualitätssicherung

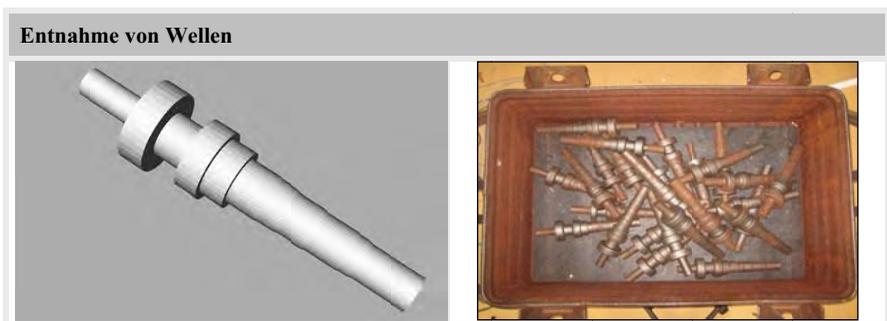


Abbildung 3-1: links: CAD-Modell einer Getriebewelle
rechts: Beispiel einer Kiste bei der Bereitstellung von Getriebewellen an einer Entnahmestation zur Qualitätssicherung

Ziel dieses Szenarios ist die Entnahme von geschmiedeten Getriebewellen aus einer Kiste und die lagerichtige Zuführung der Wellen in eine Ultraschallprüfstation, um dort innere und äußere Defekte zu detektieren. Typische Getriebewellen haben eine Länge von 250 mm bis 680 mm und ein Gewicht von 5 kg bis 7 kg [SMERobot 2008]. Die Anlieferung erfolgt sortenrein mit $OZ = (3/3)$. Das Szenario ist dadurch gekennzeichnet, dass die Wellen nach der Entnahme einzeln auf Prismen in einem Wasserbehälter abgelegt werden. Durch das Prisma ergeben sich für die Ablage Genauigkeitsforderungen von ± 5 mm translatorisch und weniger als 10° rotatorisch. Diese sollen bereits bei der Entnahme erfüllt werden. Zusätzlich soll ein Auswertezyklus maximal 10 Sekunden dauern, um noch wirtschaftlich einsetzbar zu sein:

A8: Das System zur Objektlageerkennung muss Erkennungsgenauigkeiten von $< \pm 5$ mm translatorisch und $< 10^\circ$ rotatorisch ermöglichen.

A9: Die Gesamtdauer für Sensordatenaufnahme und Objektlageerkennung darf 10 Sekunden nicht überschreiten.

3.3 Analyse der Entnahme von Hohlrädern für die Zuführung in eine Bearbeitungsmaschine

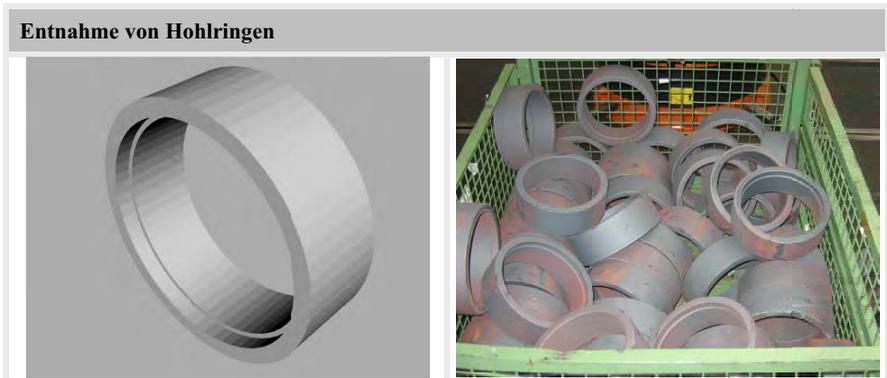


Abbildung 3-2: links: CAD-Modell eines Hohlrads, rechts: Beispiel einer Kiste bei der Bereitstellung von Hohlrädern an der Entnahmestation

Ziel dieses Szenarios ist die Entnahme von gegossenen Hohlringen aus einer Kiste und die Ablage auf ein Förderband mit Spannfütern für die Zuführung in eine Bearbeitungsmaschine. Die hier betrachteten Hohlräder haben einen Durchmesser von 265 mm und ein Gewicht von 8 kg. Die Anlieferung erfolgt sortenrein mit $OZ = (3/3)$. Durch das Förderband mit Spannfüter ergeben sich für die Ablage Genauigkeitsforderungen von mehreren Zentimetern translatorisch und weniger als $\pm 10^\circ$ rotatorisch. Ein Entnahmezyklus darf aufgrund der nachfolgenden

spanenden Bearbeitung mit einer Räumnadel maximal 80 Sekunden dauern. Das Szenario ist dadurch gekennzeichnet, dass der Hohlring Symmetrien aufweist, die bei der Beschreibung des Objekts für die Lageerkennung berücksichtigt werden sollen:

A10: Das System zur Objektlageerkennung soll Symmetrien des zu entnehmenden Objekts für die Objektbeschreibungen berücksichtigen können.

Die Anforderungen aus Kapitel 3.2 decken ansonsten das Szenario „Entnahme von Hohlringen für die Zuführung in eine Bearbeitungsmaschine“ hinreichend genau ab.

3.4 Analyse der Entnahme von Gehäusedeckeln für eine manuelle Entgratung

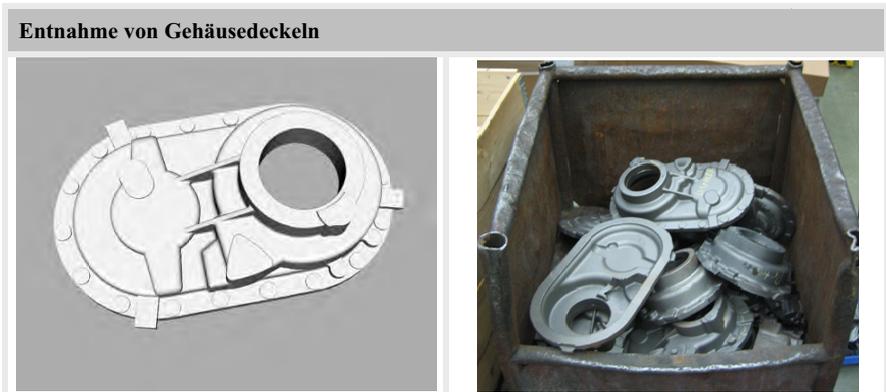


Abbildung 3-3: links: CAD-Modell eines Gehäusedeckels, rechts: Beispiel einer Kiste bei der Bereitstellung von Gehäusedeckeln an einer Entnahmestation

Ziel dieses Szenarios ist die Entnahme von gegossenen Gehäusedeckeln aus einer Kiste und die Ablage auf einen Arbeitstisch zum manuellen Entgraten. Die hier betrachteten Gehäusedeckel haben eine Länge 530 mm, eine Breite von 330 mm und eine Höhe 121 mm bei einem Gewicht von 15 kg. Die Anlieferung erfolgt sortenrein mit $OZ = (3/3)$. Durch die Ablage auf eine manuelle Entgratstation ergeben sich für die Ablage keine besonderen Genauigkeitsanforderungen. Ein Entnahmezyklus darf aufgrund der manuellen Entgratung maximal 60 Sekunden dauern.

Das Szenario ist dadurch gekennzeichnet, dass die gegossenen Gehäusedeckel nach dem Gussvorgang noch Grate und Angüsse besitzen:

A11: Das System zur Objektlageerkennung muss robust gegenüber kleinen Abweichungen in der Objektgeometrie durch Toleranzen oder Grate sein.

Die Anforderungen aus Kapitel 3.2 und Kapitel 3.3 decken ansonsten das Szenario „Entnahme von Gehäusedeckeln für eine manuelle Entgratung“ hinreichend genau ab.

3.5 Analyse der Integrationsmöglichkeit der Partikel-Schwarm-Optimierung in ein Objektlageerkennungsverfahren

Für den Einsatz eines Optimierungsverfahrens wie der Partikel-Schwarm-Optimierung ist die geeignete Form des Lösungsraums L , in dem sich die Partikel bewegen, entscheidend. Dieser Lösungsraum wird durch die Wahl zur Darstellung der Objektorientierungen beeinflusst. Aus diesem Grund sollen für die Auswahl einer geeigneten Darstellung der Objektorientierung zur Modellierung eines Schwarmpartikels zwei Kriterien beachtet werden:

- Es soll eine möglichst geringe Anzahl an unabhängigen Parametern zur Beschreibung einer Objektorientierung genutzt werden, um die Dimension des Lösungsraums gering zu halten, und
- die Definitionsbereiche der Parameter sollen aus je einem Intervall bestehen.

Bilden die Definitionsbereiche der Parameter keine Intervalle, sondern zerfallen in mehrere Bereiche, dann ist der Lösungsraum nicht als ein zusammenhängendes mehrdimensionales Volumen beschreibbar. Das würde eine komplizierte Anpassung der Schwarmparameter bei der Suche erfordern, da sichergestellt werden muss, dass alle Punkte aller Teilsuchräume mit gleicher Wahrscheinlichkeit untersucht werden, aber nie ein Partikel außerhalb der Teilsuchräume ist. Das bedeutet, dass Partikel zwischen den Teilsuchräumen „springen“ müssten und keine kontinuierliche Annäherung an das Optimum möglich ist (siehe Abbildung 3-4).

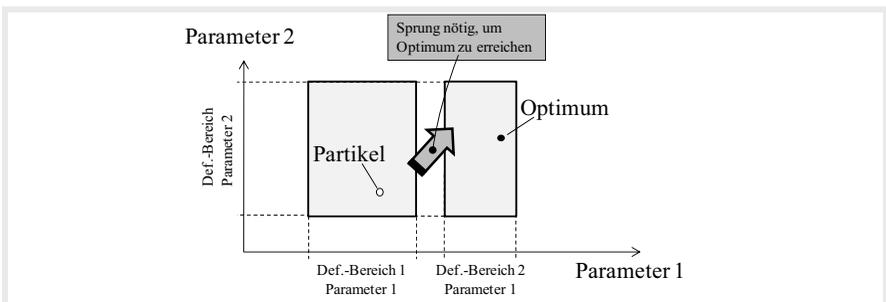


Abbildung 3-4: Beispiel eines nicht zusammenhängenden Lösungsraums mit zwei Parametern, da Parameter 1 einen Definitionsbereich aus zwei Intervallen besitzt

A12: Die Modellierung eines Partikels muss so erfolgen, dass die Parameter unabhängig voneinander sind und der Lösungsraum ein zusammenhängendes mehrdimensionales Volumen bildet.

Bevor die eigentliche Suche durch den Schwarm beginnen kann, müssen die Sensordaten entsprechend aufbereitet werden. Für jeden Iterationsschritt werden gleichzeitig die Positionen aller Partikel im Schwarm bewertet. Um die Performance des Gesamtsystems hoch zu halten, soll folgende Anforderung erfüllt werden:

A13: Die Segmentierung muss so funktionieren, dass sie für alle Partikel äquivalent abläuft.

Dies enthält gleichzeitig die Forderung, dass das System zur Objektlageerkennung entscheiden muss, wie die Segmentierung erfolgen muss und welche Sensorwerte jeweils für die Erkennung der Lage eines Objekts, d.h. die Bewertung der jeweiligen Partikel-Position genutzt werden:

A14: Ein Schwarmpartikel muss Informationen enthalten, welche Tiefenwerte für die gerade bewertete Objektlage genutzt werden.

3.6 Zusammenfassung und Strukturierung der Anforderungen an ein System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung für die Handhabung von Objekten mittels Robotern

3.6.1 Definition von Teilsystemen eines Systems zur Objektlageerkennung mit Partikel-Schwarm-Optimierung für die Handhabung von Objekten mittels Robotern

Ein Objektlageerkennungssystem besteht aus den drei Stufen Low-Level-Vision (Segmentierung), Intermediate-Level-Vision (Extraktion der Merkmale zur Objektbeschreibung) und High-Level-Vision (Auswertung der extrahierten Merkmale zur Objektlageerkennung) [Suk 1992]. Diesen drei Stufen wird zusätzlich eine Vorverarbeitung vorgeschaltet, um die Sensordaten aufzuwerten.

Die Wahl der Objektbeschreibung hat entscheidenden Einfluss auf die Funktionsweise jeder der drei Stufen. Gleichzeitig entscheidet sie darüber, wie ein einzelner Partikel des Schwarms zu modellieren ist, um die zur Bewertung der aktuellen Position im Lösungsraum benötigten Informationen bereit zu stellen. Somit ist zu erwarten, dass durch die Wahl der Objektbeschreibung zusätzliche Anforderungen an die verschiedenen Teilsysteme der Objektlageerkennung gestellt werden.

Das Gesamtsystem zur Objektlageerkennung mit Partikel-Schwarm-Optimierung benötigt somit neben dem Partikel-Schwarm selbst eine geeignete Objektbeschreibung sowie die Teilsysteme Datenvorverarbeitung und Segmentierung.

3.6.2 Strukturierung der Anforderungen und Zuordnung zu den definierten Teilsystemen

Für das Objekterkennungssystem wurden nach der Analyse der Aufgabenstellung folgende Anforderungen gefunden, die in Abbildung 3-5 strukturiert und ihr jeweiliger Einfluss den definierten Teilsystemen sowie der Objektbeschreibung zugeordnet sind. Die Zuordnung zu den Teilsystemen und der Objektbeschreibung gibt wichtige Hinweise, an welchen Stellen nachfolgend die Schwerpunkte für die Entwicklung von Lösungsansätzen zu legen sind.

Anforderungen an die Objektbeschreibung		
	Das System zur Objektlageerkennung muss an unterschiedliche Objekte anpassbar sein.	(A1)
	Das System zur Objektlageerkennung soll 3-D-CAD-Daten der Objekte als Eingangsdaten verwenden.	(A2)
	Das Erkennungssystem soll keine speziellen Zusatzinformationen eines Tiefensensors wie beispielsweise Reflektivitätsinformationen benötigen.	(A6)
	Die Objektbeschreibung, die für die Lageerkennung benötigt wird, soll für neue Objekte automatisiert erweiterbar sein.	(A7)
	Das System zur Objektlageerkennung soll Symmetrien des zu entnehmenden Objekts für die Objektbeschreibungen berücksichtigen können.	(A10)
	Das System zur Objektlageerkennung muss robust gegenüber kleinen Abweichungen in der Objektgeometrie durch Toleranzen oder Grate sein.	(A11)
Anforderungen an den Partikel-Schwarm		
	Die Modellierung eines Partikels muss so erfolgen, dass die Parameter unabhängig voneinander sind und der Lösungsraum ein zusammenhängendes mehrdimensionales Volumen bildet.	(A12)
	Ein Schwarmpartikel muss Informationen enthalten, welche Tiefenwerte für die gerade bewertete Objektlage genutzt werden.	(A14)
Anforderungen an die Datenvorverarbeitung		
	Das System zur Objektlageerkennung muss robust gegenüber Messfehlern in realen Sensordaten mit Rauschen sein.	(A4)
	Das System zur Objektlageerkennung soll mit Tiefendaten von unterschiedlichen Sensoren arbeiten können.	(A5)
Anforderungen an die Segmentierung		
	Das System zur Objektlageerkennung soll die Lage von einem oben liegenden Objekt eines Objekthaufens bestimmen.	(A3)
	Die Segmentierung muss so funktionieren, dass sie für alle Partikel äquivalent abläuft.	(A13)
Anforderungen an das Gesamtsystem		
	Das System zur Objektlageerkennung muss Genauigkeiten von $< \pm 5$ mm translatorisch und $< 10^\circ$ rotatorisch ermöglichen.	(A8)
	Die Gesamtdauer für Sensordatenaufnahme und Objektlageerkennung darf 10 Sekunden nicht überschreiten.	(A9)

Abbildung 3-5: Strukturierung der Anforderungen an die Teilsysteme und an das Gesamtsystem zur Objektlageerkennung

4 Lösungsansätze für ein System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung

4.1 Lösungsansätze zur Objektbeschreibung

Die Aufbereitung des Stands der Technik und der Forschung hat gezeigt, dass eine Kombination aus einer Wissensbasis zur Objektbeschreibung mit lokalen Merkmalen das größte Potenzial bietet, um flexibel an unterschiedliche Objekte anpassbar zu sein. Gegen den Einsatz globaler Merkmale spricht zusätzlich, dass nach Anforderung (A11) Abweichungen in der Objektgeometrie auftreten können, die zu starken Verfälschungen der globalen Merkmale führen können. Aus diesem Grund wird bei der Objektbeschreibung die Kombination aus einer Wissensbasis W mit lokalen Merkmalen weiter verfolgt.

Lokale Merkmale, die bereits erfolgreich zum Einsatz gebracht wurden, sind im Stand der Technik aufbereitet. In Abbildung 4-1 werden diese den erarbeiteten Anforderungen für die Objektbeschreibung gegenüber gestellt und bewertet:

Anforderung	Kreise/ Ellipsen	Kanten- züge	Ebene	Volumen- körper	SIFT	Histo- gramme ¹⁾
an unterschiedliche Objekte anpassbar (A1)	◐ ²⁾	●	◐ ²⁾	◐ ²⁾	●	●
3-D-CAD-Daten als Eingangsdaten (A2)	●	●	●	●	○ ⁵⁾	●
keine speziellen Zusatzinformationen (A6)	●	●	●	●	○ ⁵⁾	●
automatisiert erweiterbar (A7)	◐ ³⁾	◐ ³⁾	●	●	●	●
Symmetrien können berücksichtigt werden (A10)	●	●	●	●	●	●
robust gegenüber kleinen Abweichungen in der Objektgeometrie (A11)	◐	○	◐ ⁴⁾	●	●	●
Legende: ● Voll erfüllt ◐ Teilweise erfüllt ○ Unzureichend erfüllt ¹⁾ Kombination aus Tiefen- und Normalenvektorhistogrammen [Hetzl 2001] ²⁾ abhängig davon, ob das Merkmal in der Objektgeometrie vorhanden ist ³⁾ eine automatische Extraktion des Merkmals aus CAD-Daten ist bei komplexen Geometrien schwierig ⁴⁾ Abweichungen können zu einer Aufteilung einer Ebene führen ⁵⁾ die SIFT-Merkmale wurden für die Auswertung von Farbbildern entwickelt						

Abbildung 4-1: lokale Merkmale zur Objektbeschreibung und deren Bewertung zur Erfüllung der Anforderungen an die Objektbeschreibung

Die Merkmale Kreise/Ellipsen, Ebene und Volumenkörper schränken das Spektrum der Objekte ein, die beschrieben werden können, da nicht sichergestellt ist, dass in jedem Objekt das jeweilige Merkmal vorhanden ist. Dies kann durch Kombination der Merkmale umgangen werden. Dies

führt allerdings entsprechend dem Stand der Technik zu einer exponentiell steigenden Komplexität bei der Zuordnung der Modellmerkmale zu den extrahierten Merkmalen in den Sensordaten. Ebenso führt die Anforderung (A11), die kleine Abweichungen in der Objektgeometrie erlaubt, zu einer schlechten Bewertung der Merkmale Kreise/Ellipsen und Ebenen. Zusätzlich reagieren Kantenzüge empfindlich auf solche Abweichungen, so dass auch diese schlecht bewertet werden. SIFT-Merkmale wurden für Farbbilddaten entwickelt, so dass diese aufgrund der Anforderungen (A2) und (A6), 3-D-CAD-Daten ohne spezielle Zusatzinformationen zu nutzen, eine schlechte Bewertung erhalten.

Aufgrund der geforderten Robustheit gegenüber kleinen Abweichungen in der Objektgeometrie sowie der automatisierbaren Erweiterbarkeit der Wissensbasis mit unterschiedlichen Objekten auf Basis von CAD-Daten wird der Einsatz einer Kombination aus Tiefen- und Normalenvektorhistogrammen für die Objektbeschreibung favorisiert und es erfolgt in Kapitel 1 auf dieser Basis die Entwicklung eines wissensbasierten Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels.

4.2 Lösungsansätze für den Partikel-Schwarm

4.2.1 Modellierung eines Schwarmpartikels

Jeder Partikel p_i ($i \in \mathbb{N}$) des Schwarms repräsentiert eine potentielle Lösung des Optimierungsproblems, das mithilfe einer Fitnessfunktion bewertet wird. Zur Beschreibung und Bewertung von p_i für die gegebenen Tiefendaten $\mathbf{T} = \{(x, y, z) \in \mathbb{R}^3; T(x, y) = z\}$ werden genutzt:

1. Objektorientierung R_i
2. Objektposition P_i
3. Objektmerkmalsdaten $\mathbf{C}_i \subseteq \mathbf{T}$

Insgesamt soll p_i so modelliert werden, dass zur Bewertung eines Partikels mithilfe der Fitnessfunktion folgende Informationen genutzt werden können:

- R_i gibt an, mit welchem Eintrag aus der Wissensbasis die Partikel-Position verglichen werden soll,
- P_i gibt an, an welcher Stelle in den Tiefendaten der Vergleich stattfinden soll, und
- \mathbf{C}_i schließlich gibt an, welche Tiefendaten ausgehend von P_i einbezogen werden, um die vergleichende Bewertung durchzuführen.

Die Modellierung dieser drei Informationen R_i , P_i und C_i werden im Folgenden genauer betrachtet.

4.2.1.1 Objektorientierung R_i

Jedes Objekt besitzt insgesamt sechs Freiheitsgrade. Drei dieser Freiheitsgrade sind für die Orientierung des Objekts zuständig. Zur Beschreibung der Orientierung werden zunächst folgende Koordinatensysteme für das Objekt entsprechend der Mechanik starrer Körper eingeführt (siehe Abbildung 4-2):

- Ortsfestes Koordinatensystem mit den Einheitsvektoren e_i .
- Koordinatenachsen x_i durch den Schwerpunkt des Objekts, parallel zu den Achsen des ortsfesten Koordinatensystems. Dieses System hat dieselben Einheitsvektoren e_i wie das ortsfeste Koordinatensystem, aber einen anderen Ursprung (Schwerpunkt) nach der Translation t_S .
- Koordinatenachsen x'_i , welche durch den Schwerpunkt des Objekts verlaufen und mit diesem starr verbunden sind, also ein so genanntes körperfestes Koordinatensystem. Die Einheitsvektoren e'_i verändern ihre Richtung mit der Orientierung des Objekts.

Für die weitere Betrachtung der Orientierung eines Objekts werden lediglich die Koordinatenachsen x_i und x'_i benötigt, die Translation t_S wird deshalb vernachlässigt.

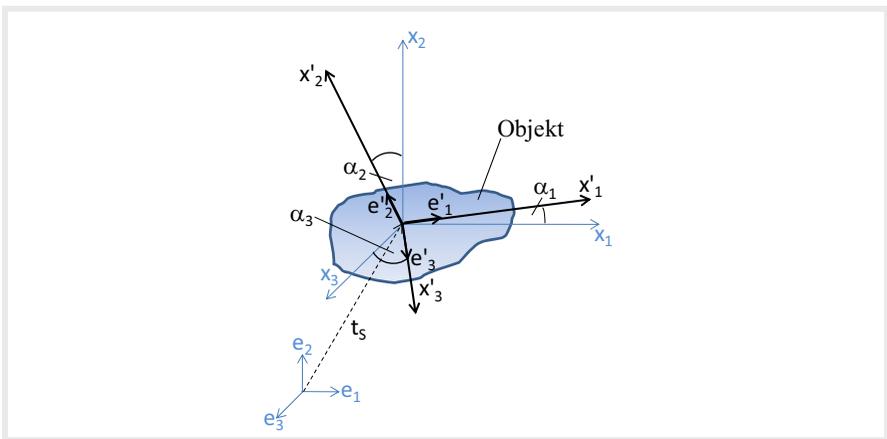


Abbildung 4-2: Koordinatensysteme zur Darstellung der Orientierung eines Objekts

Für die Darstellung einer Rotation R , die die Koordinatenachsen x_i in die Koordinatenachsen x'_i überführt, gibt es mehrere Möglichkeiten. Da sich die Wahl der Darstellung auf die Struktur des Lösungsraums auswirkt, werden im Folgenden die wichtigsten Varianten kurz betrachtet und anschließend aufgrund der in Kapitel 1 erarbeiteten Anforderungen bewertet:

- Darstellung mit Hilfe der Richtungskosinus,
- Darstellung mittels regulärer Parkettierungen einer Sichtkugel und des Rollwinkels,
- Darstellung mittels Kugelkoordinaten und des Rollwinkels,
- Darstellung mit Quaternionen,
- Darstellung mit Eulerwinkeln.

Richtungskosinus

Die Koordinatenachsen des Ausgangskoordinatensystems x_i können jeweils durch die Multiplikation mit dem Skalarprodukt der zugehörigen Einheitsvektoren e_i in die Koordinatenachsen des gedrehten Koordinatensystems x'_i übergeführt werden. Dies ist gleichbedeutend mit der Multiplikation mit dem Kosinus des jeweils zwischen den Achsen eingeschlossenen Winkels.

	$\begin{pmatrix} e'_1 \\ e'_2 \\ e'_3 \end{pmatrix} = \begin{pmatrix} e'_1 \bullet e_1 & e'_1 \bullet e_2 & e'_1 \bullet e_3 \\ e'_2 \bullet e_1 & e'_2 \bullet e_2 & e'_2 \bullet e_3 \\ e'_3 \bullet e_1 & e'_3 \bullet e_2 & e'_3 \bullet e_3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix}$ $= \begin{pmatrix} \cos \phi_1 & \cos \phi_2 & \cos \phi_3 \\ \cos \psi_1 & \cos \psi_2 & \cos \psi_3 \\ \cos \vartheta_1 & \cos \vartheta_2 & \cos \vartheta_3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix}$	<p>Gleichung 4-1</p>
<p>ϕ_1, ϕ_2, ϕ_3 bezeichnen die Winkel zwischen x_1, x_2, x_3 und x'_1 (siehe Skizze), ψ_1, ψ_2, ψ_3 bzw. $\vartheta_1, \vartheta_2, \vartheta_3$ entsprechend die Winkel zwischen x_1, x_2, x_3 und x'_2 bzw. x'_3.</p>		

Abbildung 4-3: Richtungskosinus zur Darstellung der Rotation von x_i nach x'_i

Diese Darstellung führt zu neun Richtungskosinus. Diese neun Größen unterliegen insgesamt sechs Bedingungen: Die Summe der Quadrate der Richtungskosinus eines jeden Vektors muss 1 sein und zudem müssen die Einheitsvektoren paarweise aufeinander senkrecht stehen. Von den neun Größen sind also nur drei unabhängig. Dies entspricht den drei Freiheitsgraden der Rotation. Insgesamt ist dies somit eine leicht nachvollziehbare Darstellung, die mit neun Größen allerdings mehr Informationen enthält, als eigentlich benötigt werden.

Reguläre Parkettierungen einer Sichtkugel und des Rollwinkels

Die reguläre Parkettierung einer Sichtkugel, auch als geodätische Kuppel bezeichnet, bietet die Möglichkeit einer möglichst homogenen Verteilung der Ansichten [Leibe 2001]. Ausgehend von

einem platonischen Körper, häufig einem Oktaeder, werden die Dreiecksflächen kontinuierlich immer weiter unterteilt. Bei jeder Unterteilung wird ein Dreieck an den Fußpunkten der Seitenhalbierenden in vier Teildreiecke zerlegt. Die neu entstandenen Eckpunkte werden anschließend auf die Kugeloberfläche der Umkugel des Oktaeders projiziert [Holst 2004]. Dies kann beliebig oft auf die neu entstehenden Dreiecke wiederholt werden. Mit jedem Iterationsschritt erhöht sich so die Anzahl der Dreiecke auf das Vierfache. Die folgende Abbildung 4-4 zeigt eine zwei- und dreifache Unterteilung eines Oktaeders, bestehend aus 128 und 512 Flächen. Zur Darstellung der Objektorientierungen können die Schwerpunkte der Dreiecke als Sichtpunkte genutzt werden. Dies führt bereits bei grober Unterteilung zu einer großen Anzahl an Orientierungen – bei einer sechsfachen Unterteilung entstehen bereits 32 768 Flächen.



Abbildung 4-4: zwei- und dreifache Unterteilung eines Oktaeders [Holst 2004]

Anschaulich repräsentiert jeder Vektor vom Schwerpunkt einer Dreiecksfläche (x, y, z) zum Mittelpunkt der Umkugel des Objekts je eine Blickrichtung des Sensors auf das Objekt. Neben der aufwendigen Bestimmung dieser Vektoren muss zusätzlich beachtet werden, dass keine Ansichten beinhaltet sind, in der das Objekt um diesen Vektor rotiert ist. Um somit alle möglichen Objektorientierungen mit drei Freiheitsgraden zu erstellen muss ein vierter Parameter r_s eingeführt werden, der den Rollwinkel des Sensors darstellt. Eine einzelne Objektorientierung kann somit durch das Quadrupel (x, y, z, r_s) dargestellt werden.

Ein weiterer Nachteil neben dem Bedarf an vier Parametern ist die gegenseitige Abhängigkeit der einzelnen Parameter x , y und z , da diese der Bedingung

$$\sqrt{x^2 + y^2 + z^2} = r \quad \text{Gleichung 4-2}$$

genügen müssen, in der r dem Sensorabstand entspricht. Die Definitionsbereiche für die Parameter x , y und z bilden somit keine zusammenhängenden Intervalle, was die Gestaltung des Lösungsraums erschwert, da dieser kein einzelnes Volumen beschreibt.

Kugelkoordinaten und Rollwinkel

Für die Darstellung der Objektrotationen mithilfe von Kugelkoordinaten wird ebenso wie bei der regulären Parkettierung anschaulich gesprochen nicht das Objekt rotiert, sondern der Sensor um das Objekt bewegt. Der mithilfe der zwei Winkel φ und θ und dem Radius r dargestellte Punkt auf der Kugeloberfläche bestimmt die Position des Tiefensensors mit Blick auf den Mittelpunkt der Kugel, welche auch dem Schwerpunkt des Objektes entspricht. Der zusätzliche Winkel r_s gibt den Rollwinkel des Tiefensensors an (siehe Abbildung 4-5).

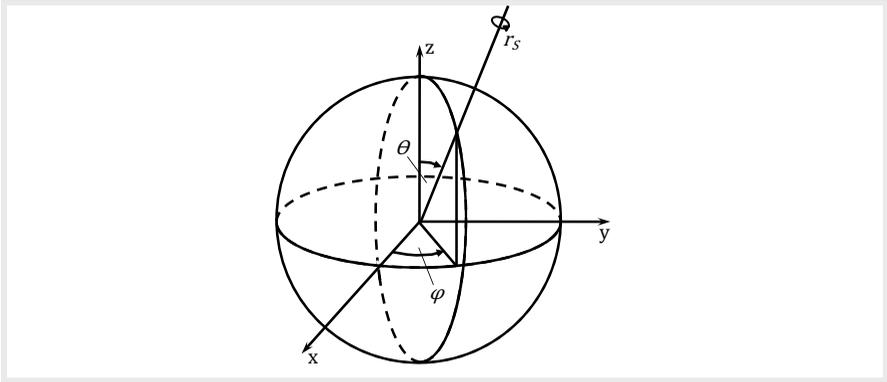


Abbildung 4-5: Darstellung einer Objektorientierung mittels Kugelkoordinaten θ und ϕ sowie Rollwinkel r

Nachteil dieser Darstellung ist, dass es keine eindeutige Zuweisung zwischen einer Orientierung und den dazugehörigen drei Winkeln gibt, d.h. mehrere Kombinationen $(\varphi, \theta, r, r_s)$ erzeugen dieselbe Lage. Zusätzlich sind hier die Abstände zwischen verschiedenen Ansichten inhomogen, d.h. im Bereich von $\theta \approx 90^\circ$ (Äquator) sind die Abstände sehr groß, während sie im Bereich von $\theta \approx 0^\circ$ (Pole) sehr klein sind. Vorteil dieser Definition ist, dass durch Setzen von $r = 1$ ein Parameter wegfallen kann und der entstehende Lösungsraum zusammenhängend ist, da φ, θ und r_s unabhängig voneinander sind und geschlossene Intervalle bilden.

Quaternionen

Die Quaternionen (oder auch Hamilton-Zahlen) sind eine Verallgemeinerung der komplexen Zahlen. Im Gegensatz zu den komplexen Zahlen, die aus einem Realteil sowie einem Imaginärteil bestehen, besitzen Quaternionen einen Realteil mit drei Imaginärteilen.

Ein Quaternion q wurde ursprünglich 1843 von Hamilton wie folgt beschrieben:

$$q = w + xi + yj + zk \quad \text{Gleichung 4-3}$$

w, x, y und z sind hierbei reelle Zahlen, i, j und k sind Imaginärzahlen.

Mit Hilfe der Quaternionen können Drehungen im Raum beschrieben werden. Genutzt wird dies heutzutage im Bereich der interaktiven Computergrafik, insbesondere bei Computerspielen, sowie bei der Steuerung und Regelung von Satelliten. Des Weiteren werden Quaternionen auch zur Programmierung von Industrierobotern (z.B. ABB) eingesetzt. Da Quaternionen Quadrupel sind, beschreiben sie Punkte in einem vierdimensionalen Raum. Um nun Rotationen im \mathbb{R}^3 beschreiben zu können, wird die Anzahl der Freiheitsgrade auf drei eingeschränkt, indem von dem Quaternion gefordert wird, dass es die Länge 1 besitzt, also ein Einheitsquaternion ist. Quaternionen können insbesondere dann ihre Stärke ausspielen, wenn viele Drehungen miteinander kombiniert (multipliziert) werden oder Rotationen in umgekehrter Reihenfolge ausgeführt werden müssen. Detaillierte Betrachtungen von Quaternionen und deren Eigenschaften können beispielsweise [Tönnies 1997] entnommen werden.

Eulerwinkel

Eulerwinkel werden eingesetzt, um Rotationen durch die Angabe von drei Winkeln darzustellen, die jeweils die Rotation um eine bestimmte Achse beschreiben und nacheinander ausgeführt werden. Prinzipiell sind hierbei die Drehachsen sowie die Drehreihenfolge beliebig. Da sie allerdings entscheidenden Einfluss auf das Ergebnis haben, müssen sie vorher festgelegt werden. Von insgesamt 12 gleichwertigen möglichen Beschreibungen haben sich folgende drei Varianten etabliert – die Striche an den Achsbezeichnungen geben an, dass jeweils um die neue Koordinatenachse gedreht wird:

- x -Konvention (z, x', z'')
- y -Konvention (z, y', z'')
- Yaw-Pitch-Roll (z, y', x'') in der Luftfahrt

Nachteil der Darstellung einer Objektorientierung mit Eulerwinkeln ist, dass sie keine eindeutige Zuweisung zwischen einer Orientierung und den dazugehörigen drei Rotationswinkeln bietet, da mehrere Kombinationen dieselbe Orientierung beschreiben können. Besonders bekannt ist hier das Phänomen des „Gimbal Lock“, bei dem die Achse der ersten Drehung und der dritten zusammenfallen. Bei der Definition der Orientierung nach der Yaw-Pitch-Roll-Variante liegen die kritischen Punkte bei einer Rotation um die y' -Achse mit $\pm \pi/2$.

Für die Yaw-Pitch-Roll-Variante ist das erhaltene Koordinatensystem mit den Achsen x'', y'' und z'' das körperfeste System. Die Winkel γ (Rotation um x'') und β (Rotation um y') geben dabei

die Lage der z'' -Achse gegenüber dem körperfesten Koordinatensystem an, d.h. die Drehung und Kippung. Der Winkel α (Rotation um z) beschreibt die Eigendrehung des Körpers um dieses Koordinatensystem. Hauptwertebereiche sind: $\alpha \in [-\pi, \pi]$, $\beta \in [-\pi/2, \pi/2]$, $\gamma \in [-\pi, \pi]$.

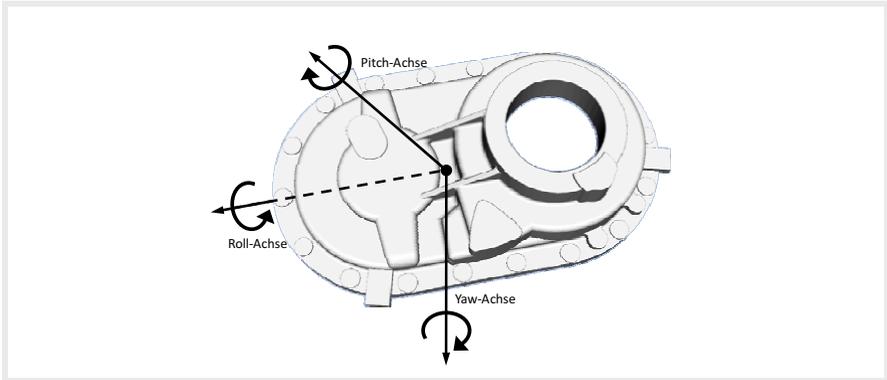


Abbildung 4-6: Yaw-Pitch-Roll-Variante der Eulerwinkel

Wesentliche Vorteile dieser Darstellungsart sind die sehr gute Anschaulichkeit und intuitive Anwendbarkeit sowie die kompakte und übersichtliche Schreibweise.

Bewertung der Lösungsvarianten zur Darstellung der Objektorientierung R_i eines Schwarmpartikels

Abbildung 4-7 fasst die beschriebenen Darstellungsmöglichkeiten für die Objektorientierung R_i mit der jeweils benötigten Anzahl der Parameter zusammen und gibt an, ob es sich bei dem entstehenden Lösungsraum, um ein zusammenhängendes mehrdimensionales Volumen handelt.

Darstellungsvariante	Merkmale	Skizze
Richtungskosinus	9 Parameter	
	Lösungsraum nicht zusammenhängend	
reguläre Parkettierung	4 Parameter	
	Lösungsraum nicht zusammenhängend	
Kugelkoordinaten und Rollwinkel	3 Parameter	
	Lösungsraum zusammenhängend	
Quaternionen	4 Parameter	(nicht darstellbar)
	Lösungsraum nicht zusammenhängend	
Yaw-Pitch-Roll-Variante der Eulerwinkel	3 Parameter	
	Lösungsraum zusammenhängend	

Abbildung 4-7: Lösungsansätze für die Darstellung der Objektorientierung R_i

In Bezug auf die Anzahl der Parameter und deren Definitionsbereiche bietet der Einsatz der Richtungskosinus mit neun Parametern deutliche Nachteile. Die reguläre Parkettierung sowie die Quaternionen benötigen jeweils vier Parameter. Durch die gegenseitige Abhängigkeit der einzelnen Parameter bei der regulären Parkettierung entstehen nicht-zusammenhängende Definitionsbereiche für die Parameter. Eulerwinkel und Kugelkoordinaten mit Rollwinkel haben je drei Parameter, die jeweils einen zusammenhängenden Definitionsbereich besitzen und somit einen zusammenhängenden Lösungsraum bilden.

Ein zusätzliches Merkmal der Darstellungen zur Objektorientierung ist die homogene Verteilung der Rotationen im Lösungsraum. Hier bietet die Darstellung mit der regulären Parkettierung die größten Vorteile. Alle anderen Darstellungen haben die Eigenschaft, dass die Orientierungen an den Polen der Einheitskugel deutlich dichter beieinander liegen als an ihrem Äquator. Diese Nicht-Homogenität hat jedoch lediglich den Nachteil, dass mehr Objektorientierungen in der Wissensbasis vorgehalten werden müssen, als im Idealfall benötigt werden, um einen vorgegebenen Abstand zwischen zwei Orientierungen zu erreichen. Zusammenhängende Definitionsbereiche der Parameter, die ein „Springen“ der Schwarmpartikel vermeiden und eine kontinuierliche Annäherung zum Optimum ermöglichen, müssen deutlich höher gewichtet werden und wurden auch in Anforderung (A12) entsprechend gefordert.

Dies führt dazu, dass sowohl Kugelkoordinaten mit Rollwinkel als auch Eulerwinkel für die Darstellung der Objektorientierung geeignet sind. Aufgrund der zusätzlich sehr guten Anschaulichkeit der Yaw-Pitch-Roll-Variante der Eulerwinkel wird diese Variante für die weitere Vorgehensweise gewählt. Die Schreibweise $R(\alpha, \beta, \gamma)$ bedeutet somit eine Rotation des Objekts in Eulerwinkeln mit den drei Winkeln (α, β, γ) .

4.2.1.2 Objektposition P_i

Neben den drei Freiheitsgraden der Objektrotation R_i müssen während der Optimierung auch die verbleibenden drei translatorischen Freiheitsgrade, die Objektposition P_i , durch den Partikelschwarm bestimmt werden. P_i legt hierbei insbesondere jeweils den Bezugspunkt für die Objektmerkmalsdaten C_i fest, die angeben, welche Tiefendaten zur Bewertung einer Objektlage einbezogen werden müssen. In Abbildung 4-8 werden alternative Ansätze zur Wahl von P_i konzipiert. Die Ansätze sind danach unterteilt, ob P_i entweder frei im Raum definiert werden kann oder ob P_i auf der Oberfläche des Objekts zu liegen kommt. Die Abbildung umreißt die große Vielfalt an Möglichkeiten bei der Wahl von P_i . Gerade durch das Vorliegen von 3-D-CAD-Daten des Objekts kann eine Vielzahl an geometrischen Möglichkeiten herangezogen werden. Entscheidend bei der Wahl von P_i ist, dass P_i bei der Erstellung der Wissensbasis automatisch extrahiert werden kann. Dies ist wichtig, um sowohl bei der Erstellung der Wissensbasis die zugehörigen Objektmerkmalsdaten C_i bestimmen zu können, als auch bei der Objektlagerkennung die relevanten Tiefendaten automatisch extrahieren zu können. Deshalb zeigt Abbildung 4-8 neben einer vergleichenden Bewertung der Ansätze aufgrund der Anpassbarkeit an unterschiedliche Objekte, auch Bewertungen hinsichtlich der automatischen Extrahierbarkeit aus den CAD-Daten und der automatischen Extrahierbarkeit aus den Tiefendaten.

		Beispiel-Skizze	an unterschiedliche Objekte anpassbar	aus den CAD-Daten automatisch extrahierbar	aus den Tiefdaten automatisch extrahierbar
frei im Raum definiert	Mittelpunkt M_B der Bounding-Box		●	●	○ ¹⁾
	Schwerpunkt S des Objekts		●	◐	○ ²⁾
	Mittelpunkt M_Q eines Querschnitts		●	●	○ ¹⁾
	eine Ecke E_B der Bounding-Box		●	●	○ ¹⁾
auf der Oberfläche des Objekts definiert	Anfangspunkt eines Kantenzugs K des Objekts		●	◐	◐
	höchster Punkt p^{Max} des Objekts		◐ ³⁾	●	●
	ein frei definierter Punkt P_F auf der Oberfläche des Objekts		●	◐ ⁴⁾	◐ ⁵⁾

Legende: ● Voll erfüllt ◐ Teilweise erfüllt ○ Unzureichend erfüllt

¹⁾ liegt i.a. nicht auf der Oberfläche der Tiefdaten, kann somit nicht aus den Daten extrahiert werden
²⁾ für eine automatische Extraktion wird zusätzlich die Dichte des Objekts benötigt
³⁾ es können mehrere höchste Punkte auftreten; welcher zu wählen ist, muss zusätzlich definiert werden
⁴⁾ es müssen ausreichend viele Punkte auf den Seiten des Objekts definiert werden, so dass für jede Orientierung mindestens ein Punkt zu sehen ist
⁵⁾ jeder Punkt der Tiefdaten kann ein frei definierter Punkt sein, da er sich durch kein Merkmal auszeichnet – eine echte Extraktion ist somit nicht möglich

Abbildung 4-8: Lösungsansätze und Bewertung für die Wahl der Objektposition P_i

Alle Varianten, die P_i nicht auf der Objektoberfläche, sondern frei im Raum definieren, schneiden bei der automatischen Extrahierbarkeit aus den Tiefendaten sehr schlecht ab, da kein eindeutig bestimmbarer Zusammenhang zwischen den Tiefenwerten und P_i besteht. Deutlich besser schneiden hier die Punkte auf der Oberfläche des Objekts ab.

Aufgrund dieser Betrachtung und da dieser Punkt unabhängig von speziellen Objekteigenschaften ist, wird im Rahmen dieser Arbeit der höchste Punkt P^{Max} zur Bestimmung der Objektposition P_i gewählt.

In der Wissensbasis W ist somit jeder Objektrotation $R(\alpha, \beta, \gamma)$ ein höchster Punkt P^{Max} zugeordnet, um die Position des Objekts im Raum zu beschreiben. Durch die Verwendung von CAD-Daten zur Generierung der synthetischen Tiefendaten T_S (analoge Definition wie für T) für die Wissensbasis kann dieser entsprechend exakt bestimmt werden. Für den Zusammenhang zwischen P^{Max} und der zugehörigen Rotation $R(\alpha, \beta, \gamma)$ wird im Rahmen dieser Arbeit folgende Schreibweise eingeführt:

$$\begin{aligned} P^{Max}|_{(\alpha, \beta, \gamma)} &:= (x^{Max}, y^{Max}, z^{Max}) \in T_S \\ \text{mit } z^{Max} &:= T_S(x^{Max}, y^{Max}) \geq z \quad \forall (x, y, z) \in T_S \end{aligned} \quad \text{Gleichung 4-4}$$

Für die Objektlageerkennung in den vom Sensor gelieferten Tiefendaten T bedeutet die Verwendung des höchsten Punkts anschaulich, dass während der Suche nach der besten Lösung die Objektposition P_i auf der Oberfläche der Sensordaten entlang läuft und jeweils als P^{Max} angenommen wird, bis eine optimale Kombination aus Position P_i und Rotation R_i gefunden wurde.

Hiermit ergibt sich, dass $P_i \in T$ gelten muss. Dies hat den Vorteil, dass während der Suche nach der optimalen Position zwei Dimensionen (x, y) optimiert werden müssen. Die dritte Dimension z ergibt sich aus dem jeweiligen Tiefenwert $z = T(x, y)$.

Ein weiterer Vorteil dieser Wahl des Punkts zur Darstellung der Objektposition ist, dass der damit bestimmte Punkt der Objektlage definitiv auf der Oberfläche der Sensordaten liegt und somit direkt für die nachfolgende automatische Entnahme des Objekts mit einem Roboter herangezogen werden kann.

4.2.1.3 Objektmerkmalsdaten C_i

Um Anforderung (A14) zu erfüllen, sind die Objektmerkmalsdaten C_i ein wichtiges Element bei der Modellierung eines Partikels. C_i enthalten die Information, welche Tiefenwerte die aktuell betrachtete potentielle Objektlage stützen und somit bei der Bewertung des Partikels einbezogen werden sollen. Da zur Berechnung der Fitnessfunktion Histogramme genutzt werden, bedeutet dies, dass für die Untersuchung der betrachteten Objektorientierung eine modellbasierte

Segmentierung [Jähne 2002] durchgeführt wird, die sämtliche Tiefenwerte, die zur potentiellen Objektlage gehören, von den restlichen Tiefenwerten, dem Hintergrund, trennt.

Die Definition der Objektmerkmalsdaten C_i in den einzelnen Orientierungen als Grundlage für diese Segmentierung muss bereits während der Generierung der Wissensbasis erfolgen und dort gespeichert werden. Unter der Annahme, dass bei der Erstellung der Wissensbasis alle synthetischen Tiefendaten T_S , die nicht vom Objekt belegt sind, mit einem Wert kleiner 0 initialisiert werden und alle übrigen Tiefenwerte positiv sind, ergibt sich die Anweisung zur Erstellung der Objektmerkmalsdaten zu:

$$C_i = \{(x, y, z) \in T_S, T_S(x, y) = z \geq 0\} \quad \text{Gleichung 4-5}$$

Zusätzlich bietet der Einsatz der C_i einen weiteren Vorteil. Da bei der Erstellung von Histogrammen aus Tiefenwerten lediglich die Tiefeninformation erhalten bleibt, die Ortsinformation jedoch verloren geht, sind diese eindimensionalen diskreten Verteilungen invariant gegenüber Drehungen um die z-Achse. In Kombination mit C_i entstehen für Objekte mit unterschiedlichen Drehungen um die z-Achse jedoch unterschiedliche Histogramme und können während der Suche nach der optimalen Objektlage unterschieden werden.

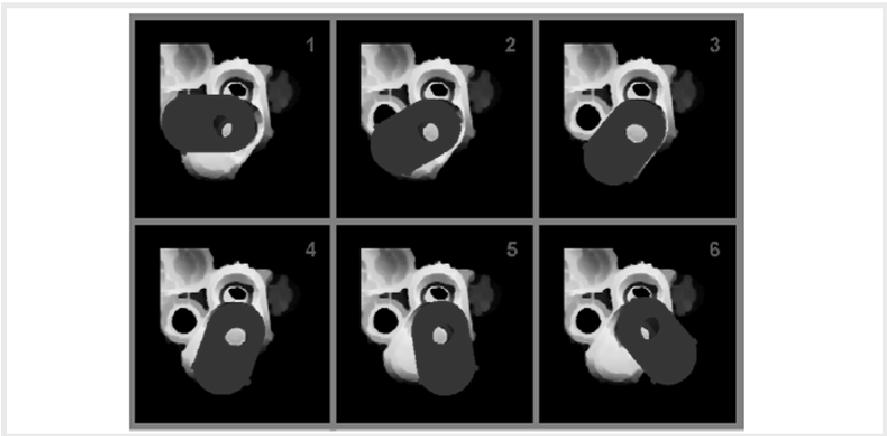


Abbildung 4-9: Beispiel von Objektmerkmalsdaten C_i (dunkelgrau) eines Gehäusedeckels in unterschiedlichen Orientierungen (Drehung um die z-Achse), die über ein Sensorbild gelegt werden. Die Position des höchsten Punkts p^{Max} ist jeweils konstant.

4.2.2 Partikel-Schwarm-Steuerung

Hauptaufgabe der Partikel-Schwarm-Steuerung ist, die Bewegungen der Schwarmpartikel im Lösungsraum durchzuführen und hierbei jeweils die Informationen über die beste eigene Position

eines Partikels sowie über die beste Nachbarposition zu verwalten. Zusätzlich muss die Steuerung dafür sorgen, dass Partikel, die aufgrund ihrer berechneten Bewegung den Rand des Lösungsraums überschreiten würden, erkannt werden und wieder definiert in den Lösungsraum zurückgesetzt werden.

Zur Lösung dieser Randproblematik gibt es in der Hauptsache die drei Möglichkeiten: den Partikel zufällig im Lösungsraum zu positionieren, den Partikel am Rand zu reflektieren oder eine Art „Tunnelung“ der Partikel durchzuführen und diese auf der gegenüberliegenden Seite des Lösungsraums wieder eintreten zu lassen.

Für die Objektrotationen gilt, dass ein Winkel von 365° , der außerhalb des Lösungsraums liegen würde, die gleiche Objektrotation liefert wie ein durch Tunnelung entstehender Winkel von 5° . Ein durch Reflexion entstehender Winkel von 355° liefert dagegen eine andere Objektrotation. Somit ist hier eine zyklische Fortführung des Lösungsraums durch den Einsatz der Tunnelung, die den Lösungsraum zu einem 3-Torus [Geiges 1997] erweitert, für die Objektorientierung R_i sinnvoll und soll durch die Steuerung umgesetzt werden.

Im Gegensatz hierzu gilt für die Objektposition P_i , dass diese zyklische Fortführung des Lösungsraums als 3-Torus bei einer Überschreitung des Randes die Suche des Partikels von einem Rand der Sensordaten zum gegenüberliegenden Rand der Sensordaten versetzen würde, was einer komplett anderen Region der betrachteten Kiste entspricht. Deshalb soll für die Objektpositionen P_i die Steuerung ein Reflektieren des Partikels am Rand benutzen.

Neben dieser Randproblematik muss die Partikel-Schwarm-Steuerung zusätzlich eine Bedienerschnittstelle zur Verfügung stellen, um neben dem zu suchenden Objekt auch die für die Suche wichtigsten Parameter wie Schwarmgröße, Nachbarschaftstopologie, Nachbarschaftsgröße sowie die Gewichtungen des kognitiven und des sozialen Verhaltens eingeben zu können.

4.3 Lösungsansätze zur Datenvorverarbeitung der Sensordaten

Das Teilsystem zur Datenvorverarbeitung der Sensordaten liefert die Schnittstelle zum Einlesen und Filtern der Sensordaten zwischen dem Tiefensensor und der ersten Stufe der Objektlageerkennung, der Segmentierung. Aus der Betrachtung der Tiefensensoren für die Objektlageerkennung geht hervor, dass Tiefendaten, die durch Parallelprojektion gewonnen werden, Vorteile gegenüber der Zentralprojektion aufweisen. Neben der Mehrdeutigkeiten in den Abstandswerten zur gleichen (x, y) -Position zählen hierzu auch Abschattungen. Lösungsansätze für die Datenvorverarbeitung der Sensordaten setzen sich somit aus folgenden Funktionen zusammen:

- Einlesen der Sensordaten,

- Auflösung von Mehrdeutigkeiten in den Abstandswerten,
- Umgang mit Abschattungen durch Zentralprojektion und
- Rauschfilterung.

4.3.1 Einlesen der Sensordaten

Gemäß der Anforderung (A5) aus Kapitel 1 muss das System zur Objektlageerkennung in der Lage sein, die Sensordaten unterschiedlicher Tiefensensoren einzulesen und mit diesen zu arbeiten. Zur Erstellung der Histogramme, die zur Bewertung einer Objektlage genutzt werden, sind Punktinformationen am geeignetsten. Deshalb werden Oberflächendarstellungen, die von manchen Sensoren optional, beispielsweise im STL-Format, geliefert werden, nicht betrachtet. Je nach eingesetztem Messprinzip und Sensorhersteller variiert die Bereitstellung der Punktinformationen von seriellen Schnittstellen mit Polarkoordinaten [Sick 2005] bis zu Fire-Wire-Schnittstellen mit äquidistant gerasterten kartesischen Koordinaten [Breuckmann 2008].

Für die weitere Betrachtung wird von der allgemeinen Darstellung einer Punktwolke D in kartesischen Koordinaten ausgegangen. Diese wird durch das Modul zum Einlesen der Sensordaten entweder aus einer Datei von der Festplatte oder aus einer virtuellen Datei im Shared-Memory-Bereich des eingesetzten Rechners eingelesen.

4.3.2 Lösungsansatz zur Auflösung von Mehrdeutigkeiten in den Abstandswerten

Eine Punktwolke D kann zu jedem Punkt (x, y) einer Oberfläche F_o mehrere z -Werte enthalten (siehe Abbildung 4-10). Außerdem liefert ein Objekt, das von einer Seite der Kiste bei gleichbleibender Orientierung auf die andere Seite der Kiste geschoben wird, aufgrund der Zentralprojektion unterschiedliche Sensordaten. Für die Objektlageerkennung wäre ideal, wenn ein Objekt in unterschiedlichen Positionen zu gleichen Sensordaten führen würde. Die Parallelprojektion liefert solche Daten (siehe Kapitel 2.1.2). Ziel ist es deshalb, aus der Punktwolke D , eine Parallelprojektion zu emulieren, so dass $T(x, y) = z$ gilt mit einem z -Wert, der möglichst der tatsächlichen Oberfläche der betrachteten Szene entspricht.

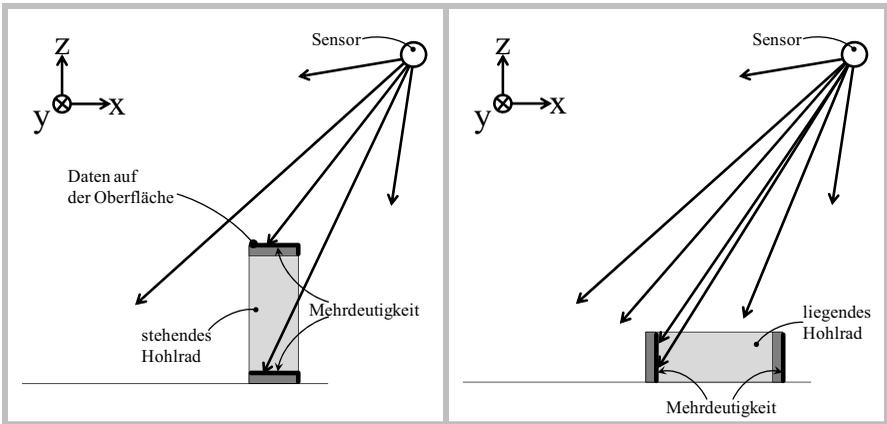


Abbildung 4-10: Mehrdeutigkeit der (x, y) -Messung am Beispiel eines liegenden und eines stehenden Hohlrades in Seitenansicht

Für die Bestimmung eines solchen z -Werts auf der Oberfläche spielen zwei Kriterien eine wichtige Rolle. Die Empfindlichkeit des Bestimmungsverfahrens gegenüber Rauschen und die Sicherheit, dass der ermittelte z -Wert auf der Oberfläche liegt. Auf Basis dieser Kriterien werden in Abbildung 4-11 fünf Ansätze für die Ermittlung des z -Werts bewertet.

Kriterium	Median-Filter	Mittelwertbildung	Kleinsten z-Wert	Häufigster z-Wert	Größe des Messwinkels ¹⁾
Unempfindlichkeit gegen Rauschen	●	◐	○	●	◐
Sicherheit, dass Messwert auf der Oberfläche liegt ²⁾	◐	○	●	◐	●

Legende: ● gut erfüllt ◐ bedingt erfüllt ○ Unzureichend erfüllt

¹⁾ Messwinkel, der bei der Datenaufnahme eines Abstandswerts genutzt wurde (Abbildung 4-12)

²⁾ mit optimalen Messwerten ohne Rauschen

Abbildung 4-11: Ansätze für die Bestimmung eines realitätsnahen z -Werts und deren Bewertung

Der Median-Filter ist sehr gut geeignet, um Ausreißer in den Sensordaten, die beispielsweise durch Rauschen entstanden sind, zu eliminieren [Jähne 2002]. Tritt jedoch eine Situation wie in Abbildung 4-10 auf, so ist es nicht eindeutig vorhersagbar, wo der mittlere Wert liegt. Er kann sowohl auf dem Mantel, also der Oberfläche, des Hohlrades als auch im Innenradius des Hohlrades zu liegen kommen. Deshalb erfüllt der Median-Filter das Kriterium, das fordert, dass der Messwert auf der Oberfläche des Objekts liegt, nur teilweise. Noch schlechter schneidet die Mittelwertbildung ab, da hier durch die Mittelung aller z -Werte der resultierende z -Wert zwischen dem Mantel des Hohlrades und dem Innenradius zu liegen kommt, mit Sicherheit jedoch nicht auf dem Mantel des Hohlrades. Wie gut der Mittelwert mit Rauschen zurecht

kommt, hängt hauptsächlich von der Stärke und der Häufigkeit der Ausreißer ab. Deshalb ist er nur bedingt anwendbar.

Die Wahl des kleinsten z -Werts bietet den Vorteil, dass sich bei rauschfreien Sensordaten der ausgewählte z -Wert immer auf der Oberfläche befindet. Durch Rauschen bedingte Ausreißer können den Wert allerdings stark verfälschen. Diesen Nachteil behebt die Suche des z -Werts, der am häufigsten auftritt. Allerdings kann dieser Wert ähnlich wie beim Median-Filter unterhalb der Oberfläche auf dem Innenradius des Hohlrades liegen.

Eine wichtige Zusatzinformation, die bei sorgsamer Implementierung einer Sensorschnittstelle vorhanden ist, bieten die Winkel, die bei der Messung eines Abstandswerts verwendet wurden. Beispielsweise tasten Lichtschnittsensoren und geschwenkte 2-D-Scanner eine Oberfläche zeilenweise ab. Die Nummer jeder Zeile gibt Informationen zu dem Messwinkel α_x in x -Richtung, während die Position eines Abstandswerts innerhalb einer Zeile Informationen zu dem Messwinkel α_y in y -Richtung gibt. Bei gleichem (x, y) -Wert liegt ein Abstandswert mit großem Messwinkel immer auf der Oberfläche, während ein Abstandswert mit einem kleinen Messwinkel unterhalb der Oberfläche in Zwischenräumen misst, wie beispielsweise dem Innenradius (siehe Abbildung 4-12). Rauschen kann bei diesem Ansatz ebenfalls dazu führen, dass der gewählte Abstandswert verfälscht ist.

Aufgrund dieser Betrachtung fällt die Entscheidung, wie ein z -Wert für die Emulation der Parallelprojektion bestimmt wird, auf eine Kombination aus den drei Ansätzen Größe des Messwinkels, häufigster z -Wert und Mittelwertbildung, um deren jeweilige Vorteile zu nutzen.

Zur Bestimmung des z -Werts wird somit folgende Vorgehensweise gewählt: Hauptkriterium ist die Größe des Messwinkels, so dass für eine (x, y) -Position der Messwert gewählt wird, der den größten Messwinkel aufweist. Zusätzlich muss die Bedingung, dass innerhalb eines um diesen Wert gebildeten Intervalls eine Mindestzahl an weiteren z -Werten auftritt, erfüllt sein. Ansonsten wird ein z -Wert mit kleinerem Messwinkel gewählt, der die Mindestzahl an ähnlichen z -Werten in dem definierten Intervall erfüllt. Abschließend werden alle in diesem Intervall liegenden z -Werte gemittelt.

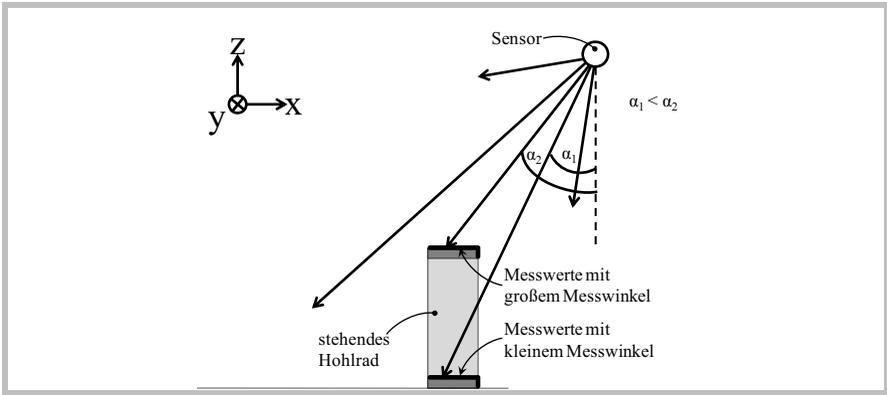


Abbildung 4-12: Zusammenhang zwischen Größe des Messwinkels und Lage des Abstandswerts auf der Oberfläche am Beispiel eines stehenden Hohlrads in Seitenansicht

4.3.3 Verfahren zur Auflösung von Mehrdeutigkeiten in den Abstandswerten mittels Äquipotenzialen

Für die Auflösung von Mehrdeutigkeiten in den Sensordaten soll im weiteren Verlauf dieser Arbeit davon ausgegangen werden, dass die im vorhergehenden Abschnitt erarbeitete Emulation der Parallelprojektion mit äquidistanten Abständen in x - und y -Richtung erstellt wird. Dies bietet zum einen den Vorteil, dass das Ergebnis der Emulation einfach als Tiefenmatrix M_T mit Tiefenwerten als Elementeinträgen dargestellt werden kann, zum anderen fallen bei einer geeigneten Wahl der Schrittweite mehrere z -Werte in ein Matrixelement, die zur Erhöhung der Robustheit gegen Rauschen für eine Mittelung genutzt werden können.

Gleichung 2-1 kann hierdurch diskret dargestellt werden zu:

$$T_D: \mathbb{N}^2 \rightarrow \mathbb{R}: (i_x, i_y) \rightarrow T_D(i_x, i_y) = z \quad \text{mit } (i_x, i_y) \in I_x \times I_y \subseteq \mathbb{N}^2 \quad \text{Gleichung 4-6}$$

Ohne Beschränkung der Allgemeinheit kann $I_x \times I_y$ als rechteckiges Gitter betrachtet werden mit $I_x = [0..n] \subseteq \mathbb{N}$ und $I_y = [0..m] \subseteq \mathbb{N}$, ($m, n \in \mathbb{N}$).

M_T ergibt sich damit zu:

$$M_T = \begin{pmatrix} T_D(0,0) & \cdots & T_D(0,n) \\ \vdots & \ddots & \vdots \\ T_D(m,0) & \cdots & T_D(m,n) \end{pmatrix} \quad \text{Gleichung 4-7}$$

Eine geeignete Wahl der Schrittweiten in x - bzw. y -Richtung in Abhängigkeit von der lateralen Auflösung des Tiefensensors ist wichtig bei der Berechnung der Fitnessfunktion und wird deshalb im Kapitel 1 betrachtet.

Für den Einsatz während der Objektlageerkennung, muss die Bestimmung des z -Werts schnell erfolgen. Für jedes Matrixelement (i_x, i_y) alle z -Werte, die in dieses Element fallen, in einer Liste unbekannter Länge zu verwalten und anschließend auszuwerten, ist speicher- und rechenintensiv. Das Verfahren wird deshalb so konzipiert, dass ein z -Wert möglichst nur ein einziges Mal betrachtet werden muss. Er wird entweder für die Bestimmung des Eintrags in das Matrixelement genutzt oder verworfen.

Um dies zu erreichen, dienen die zu einem Abstandswert a_i ($i \in \mathbb{N}$) zugehörigen Messwinkel (α_x, α_y) , wobei der Winkel 0° in der Mitte des Messbereichs liegt: Die Abstandswerte a_i und damit die zugehörigen z -Werte $z(a_i)$ werden anhand dieser Messwinkel in Klassen mit unterschiedlichen Niveaus – im Folgenden Äquipotenziale genannt – eingeteilt. Abstandswerte, die den äußeren Äquipotenzialen zugeordnet werden, erhalten ein höheres Niveau, als jene in der Mitte (Messwinkel 0°). Das Äquipotenzialniveau, das jedem Abstandswert a_i zugewiesen wird, ergibt sich direkt aus den Messwinkeln, die bei der Erstellung des Abstandswerts verwendet wurden, nach folgender Formel:

$$N_{\text{Äqui}}(a_i) = \sqrt{(\alpha_x)^2 + (\alpha_y)^2} \quad \text{Gleichung 4-8}$$

Der Ablauf zur Bestimmung der z -Werte für jedes Matrixelement kann mit dieser Festlegung Abbildung 4-13 entnommen werden.

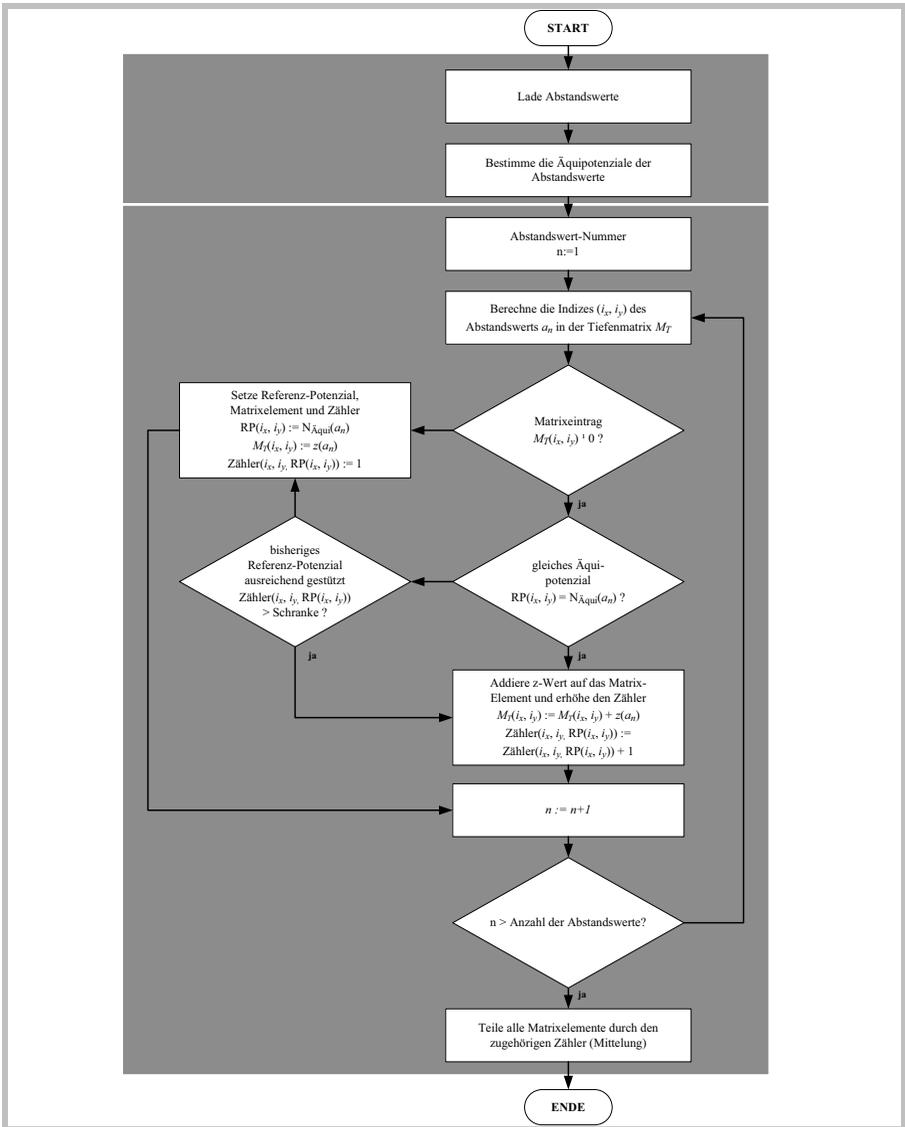


Abbildung 4-13: Ablaufdiagramm für die Bestimmung der z-Werte der Tiefenmatrix mittels Äquipotenzialen

4.3.4 Umgang mit Abschattungen durch Zentralprojektion

Neben Mehrdeutigkeiten in den Messwerten sind auch Bereiche, in denen keine Abstandswerte aufgrund von Abschattungen vorhanden sind, problematisch für die Objektlageerkennung. Denn auch nach der Emulation der Parallelprojektion aus der Punktwolke D enthalten die Matrixelemente von T_M , die in Abschattungen liegen, keine definierten z -Werte der Oberfläche der Szene. Insbesondere, wenn sich ein Objekt in der Abschattung befindet, muss beim Vergleich des verdeckten Objekts mit der Wissensbasis dafür gesorgt werden, dass die Abschattung möglichst geringe Störungen bei der Berechnung der Fitnessfunktion erzeugt. Um dies zu gewährleisten, werden die abgeschatteten Bereiche speziell als leere Bereiche markiert und maskiert und bei der Berechnung separat betrachtet. Eine Interpolation aus den Nachbarwerten, um künstlich Einträge in den Matrixelementen zu generieren, erfolgt nicht.

4.3.5 Filtern der Sensordaten

Anforderung (A4) aus Kapitel 1 fordert, dass das System zur Objektlageerkennung robust gegenüber Messfehlern in realen Sensordaten mit Rauschen sein muss. Aus diesem Grund müssen die Daten in ihrer Qualität noch verbessert werden, da sie trotz der Mittelung bei der Emulation der Parallelprojektion noch folgende Merkmale aufweisen können:

- Selbst glatte Oberflächen erscheinen noch recht uneben – häufig fallen zu wenige Messwerte in ein Matrixelement, um Messfehler komplett zu glätten
- Leere Matrixelemente – dies können sowohl in Abschattungen liegende Bereiche sein, als auch Elemente, die keine Messwerte enthalten, da die Abtastfrequenz des Tiefensensors zu gering ist.

Es liegt somit eine lückenhafte und mit Rauschen behaftete Matrix vor, die allerdings durch die bisherigen Vorverarbeitungsschritte einer Parallelprojektion sehr ähnlich ist.

Um die Güte der Daten weiter zu verbessern, werden weitere Standard-Filteroperationen auf die Matrix angewendet, die im Rahmen der Arbeit nicht detaillierter betrachtet werden:

- Entfernen nicht verlässlicher Einträge: Matrixelemente, die eine Mindestanzahl an Messwerten unterschreiten, werden als leer markiert.
- Glätten der Oberflächen mit einem angepassten Medianfilter, der leere Matrixelemente ignoriert.
- Schließen der Lücken durch Anwendung einer Kombination aus Dilatation und Erosion (Closing).

Weitergehende Informationen zu Medianfilter und Dilatation, Erosion sowie Closing können [Jähne 2002] entnommen werden.

Ein Beispiel nach Abschluss aller Vorverarbeitungsschritte kann Abbildung 4-14 entnommen werden:

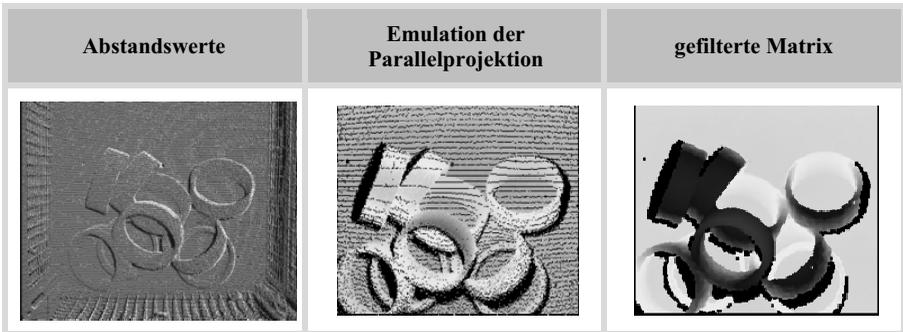


Abbildung 4-14: Beispiele von Abstandswerten, einer Emulation der Parallelprojektion und einer gefilterten Matrix nach Anwendung aller Vorverarbeitungsschritte anhand des Testobjekts Hohlrad

4.4 Lösungsansätze zur Segmentierung der Sensordaten

Aufgabe der Segmentierung ist die Trennung der Tiefendaten eines Objekts in der Szene vom Hintergrund. Für den Einsatz der Partikel-Schwarm-Optimierung bedeutet dies, dass für eine Objektlage, die durch einen Partikel repräsentiert wird, diejenigen Tiefenwerte ausgewählt werden, die zur Bewertung des Partikels genutzt werden und somit die Hypothese stützen. Aufgrund der durchgeführten Modellierung eines Partikels im Schwarm beinhaltet diese bereits eine lagespezifische Segmentierung, die auf Basis der Objektmerkmalsdaten C_i als angepasstes modellbasiertes Verfahren durchgeführt wird.

Ziel einer vorgelagerten allgemeinen Segmentierung ist, potentielle Objektpositionen P_i als Bezugspunkte für die anschließende lagespezifische Segmentierung zu bestimmen. Gängige Verfahren sind neben modellbasierten Verfahren die pixelbasierte Segmentierung, die kantenbasierte Segmentierung sowie die regionenbasierte Segmentierung in unterschiedlichen Varianten [Jähne 2002, Steinbrecher 1993]. P_i wurde als höchster Punkt P^{Max} des synthetisch rotierten Objekts definiert. Um nun von einem oben liegenden Objekt eines Objekthaufens den entsprechenden Punkt in den Tiefendaten zu ermitteln, ist es ausreichend, den höchsten Punkt des Haufens zu bestimmen. Obwohl dieser Punkt den kleinsten Abstand zum Tiefensensor hat, und somit nicht ein Maximum, sondern ein Minimum der Tiefendaten bildet (da der Abstand

zwischen Sensor und Kiste gemessen wird), soll zur Konsistenz der Bezeichnungen und aus Gründen der Anschaulichkeit dieser Punkt trotzdem als P^{Max} bezeichnet werden.

Einzelne Punkte von T_D können trotz der durchgeführten Vorverarbeitungsschritte mit Rauschen behaftet sein. Um entsprechend der ermittelten Anforderungen robust gegen Rauschen in realen Messwerten zu sein, werden mehrere Punkte in einem Bereich der Tiefendaten als mögliche Kandidaten für P^{Max} markiert. Die Auswahl dieser Punkte erfolgt mithilfe der sehr einfachen und schnellen pixelbasierte Segmentierung (Blob-Suche), die ausgehend von einem z-Wert durch die Vorgabe eines maximalen z-Abstands (Schwellwert) Bereiche markiert.

Die Anzahl der Punkte, die bei dieser einfachen Segmentierung markiert werden, lässt sich durch die Wahl des Schwellwerts leicht beeinflussen. Ein Beispiel der Segmentierung mit unterschiedlichen Schwellwerten, die zu unterschiedlichen Bereichen führen, ist in Abbildung 4-15 veranschaulicht.

Insgesamt erfolgt somit die Segmentierung zur Objektlageerkennung in zwei Schritten – einer allgemeinen, sehr schnellen pixelbasierten Segmentierung, die für alle Partikel gleich abläuft, und einer lagespezifischen, modellbasierten Segmentierung, die auf Basis der Objektmerkmalsdaten C_i arbeitet.

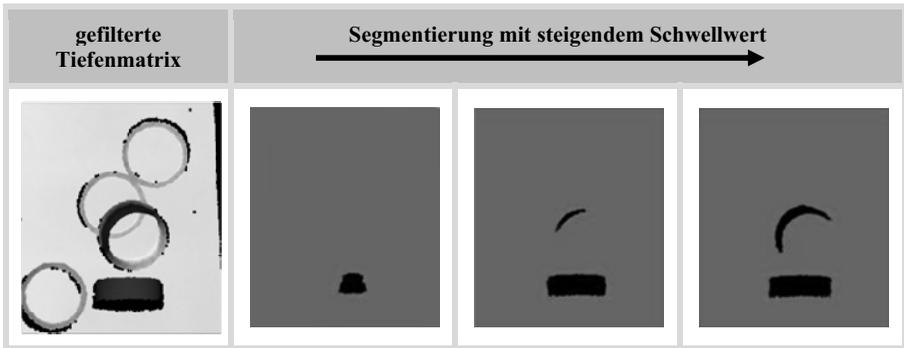


Abbildung 4-15: Beispiel segmentierter Sensordaten mit steigenden Schwellwerten

4.5 Zusammenführung der Teilkonzepte zu einem System zur Objektlageerkennung mit Partikel-Schwarm-Optimierung

4.5.1 Gesamtsystem

Die verschiedenen Aspekte aus den Abschnitten 4.1 bis 4.4 werden im Folgenden zusammengeführt. Kern der Architektur ist die Schwarm-Steuerung mit Bedieninterface. Um

diese werden die anderen Teilsysteme sowie die benötigte Wissensbasis angeordnet. Das heißt, es gibt eine zentrale Steuerungseinheit, die sich der Dienste anderer Teilsysteme bedient. Jedes Teilsystem ist über eine einfache Schnittstelle angebunden. Abbildung 4-16 zeigt die Gesamtarchitektur.

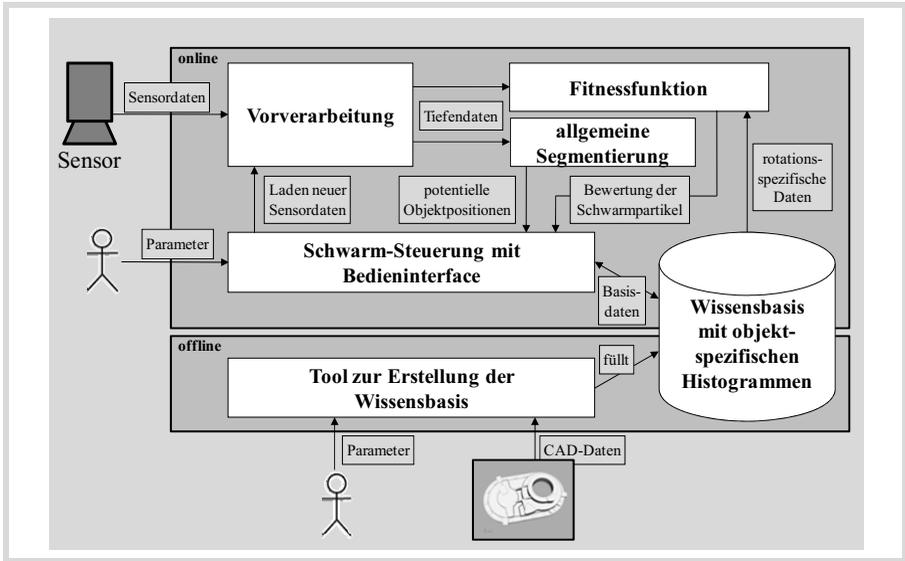


Abbildung 4-16: Gesamtlösungskonzept für den Einsatz der Partikel-Schwarm-Optimierung zur histogrammbasierten Objektlageerkennung

Um die Objektlageerkennung mithilfe der Kombination aus Partikel-Schwarm-Optimierung und Histogrammen durchzuführen, wurde das Erkennungssystem in zwei Teile verlegt - einen offline-Teil und einen online-Teil. Als Verbindung dieser beiden Teile dient die Wissensbasis W , die offline gefüllt und online zur Bewertung eines Schwarmpartikels genutzt wird.

Das offline-Tool erhält als Eingangsgrößen die CAD-Daten des Objekts, das gefunden werden soll, sowie die wichtigsten Parameter zur Erstellung der Histogramme und zur Eingrenzung der benötigten rotatorischen Freiheitsgrade. Aufgrund dieser einfachen Vorgaben kann die Erstellung der Wissensbasis erfolgen.

Den Kern des online-Teils des Erkennungssystems bildet die Schwarm-Steuerung. Die Steuerung kommuniziert sowohl mit den Teilsystemen zur Vorverarbeitung und allgemeinen Segmentierung als auch mit dem Teilsystem zur Auswertung der Fitnessfunktion. Außerdem tauscht es mit der

Wissensbasis grundlegende Basisdaten aus – beispielsweise, um welches Objekt es sich handelt und wie die Größe des Lösungsraums ist.

Die Sensordaten, die als Abstandswerte geliefert werden, werden durch die Vorverarbeitung aufbereitet und an die Segmentierung und die Fitnessfunktion übergeben.

Vorteil dieses Konzepts, in dem die Sensordatenverarbeitung sowie die Wissensbasis getrennt von der eigentlichen Steuerung des Schwarms sind, ist, dass der Sensor mit zugehöriger Vorverarbeitung schnell getauscht werden kann. Außerdem ist es auch möglich, zukünftig andere Verfahren statt der Histogramme zur Bewertung der Schwarmposition einfach einzusetzen, solange sie zum entwickelten Modell eines einzelnen Partikels passen. Die Verfahren müssen also eine Position in den Tiefendaten sowie eine Rotation und die Punkte, die eine Lage charakterisieren, liefern. Dies ist allerdings für viele Verfahren möglich, so dass dies keine nennenswerte Einschränkung bedeutet.

4.5.2 Regelablauf

In der Wissensbasis W liegen Informationen über mehrere Objekte vor. Nachdem der Benutzer (oder in einer späteren Praxisanwendung eine übergeordnete Steuerung) das zu lokalisierende Objekt ausgewählt hat, werden die benötigten Informationen für die Lageerkennung aus der Wissensbasis geladen und die eigentliche histogrammbasierte Objektlageerkennung mit PSO begonnen. Zunächst werden die Sensordaten, die der eingesetzte Sensor liefert, eingelesen und vorverarbeitet (vgl. Abschnitt 4.3) sowie die allgemeine Segmentierung zur Vorauswahl von Objektpositionen durchgeführt (vgl. Abschnitt 4.4). Für die Initialisierung des Schwarms, in der unter anderem die Schwarmgröße, die Nachbarschaftsumgebung sowie die maximalen Iterationen vorgegeben werden, hat der Benutzer die Möglichkeit, die Parameter manuell einzugeben.

Die Initialisierung der Suche erfolgt getrennt sowohl für die Position P_i als auch für die Rotation R_i jedes Partikels. Da die Suche nach einer optimalen Position des Objekts innerhalb des im vorhergehenden Schritt segmentierten Bereichs stattfindet, werden alle Partikel in diesem Bereich zufällig verteilt. Zusätzlich wird für jeden Partikel zufällig die Rotation für die anschließende Suche initialisiert. Entsprechend dem allgemeinen Vorgehen beim Einsatz der PSO werden außerdem die Änderungsvektoren für P_i und R_i für den ersten Iterationsschritt zufällig initialisiert.

Die Erkennung mit Hilfe der Partikel-Schwarm-Optimierung ist in zwei Schleifen unterteilt. Beide Schleifen durchlaufen jeweils alle Partikel des Schwarms. Schleife 1 dient dazu, zunächst die eigene Position im Lösungsraum mit Hilfe der Fitnessfunktion zu bewerten und anschließend mit den Nachbarn zu vergleichen. Aufgrund dieser Informationen erfolgt in Schleife 2 die Berechnung der neuen Position jedes Partikels im Lösungsraum, wobei die eigene bisher beste

Position sowie die bisher beste Position der Nachbarn einbezogen werden. Dies wird so lange wiederholt, bis alle Iterationsschritte erfüllt wurden, oder die Bewertung eines Partikels besser als eine vorgegebene Schranke ist.

Nach Abschluss eines Erkennungszyklus, nach dessen Ende mit Hilfe der PSO die optimierte Objektlage vorliegt, kann mit einem Roboter dieses Objekt entnommen werden und mit dem Einlesen neuer Sensordaten ein neuer Erkennungszyklus begonnen werden.

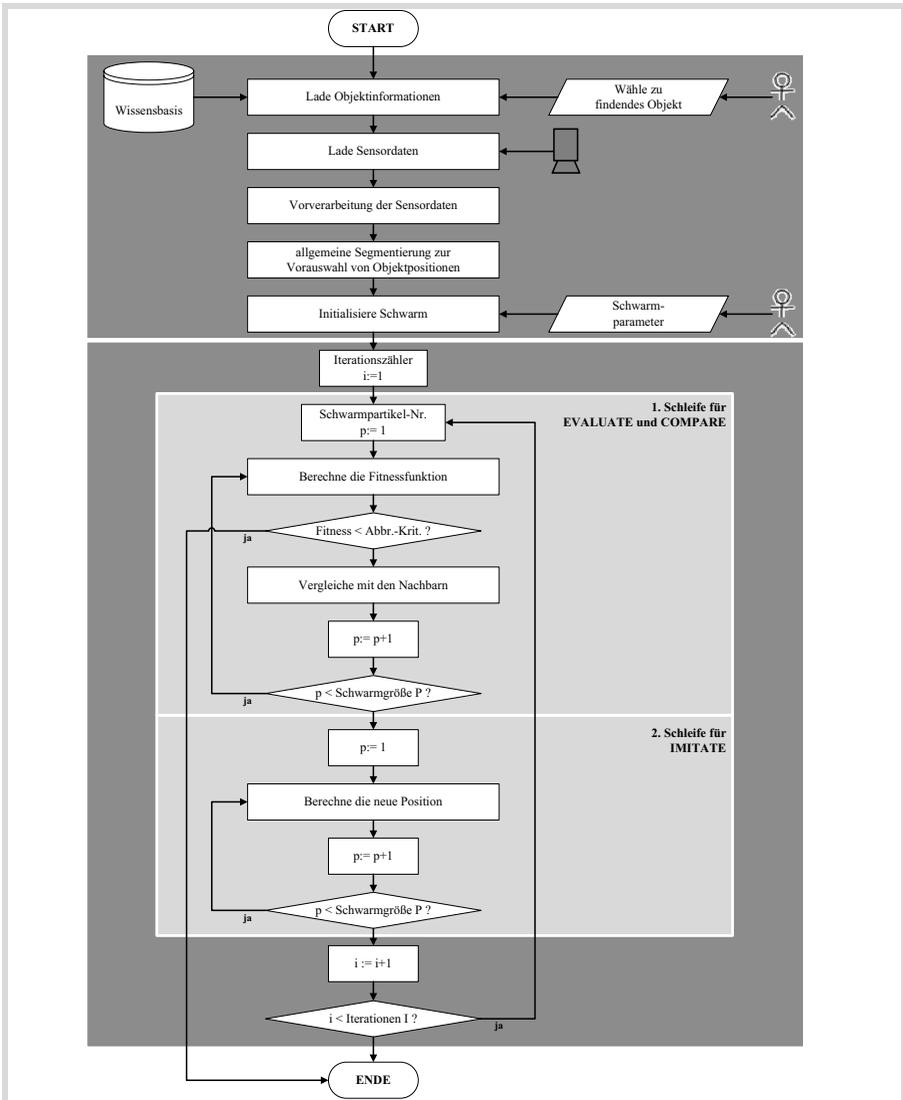


Abbildung 4-17: Regelablauf der Objektlageerkennung mit Partikel-Schwarm-Optimierung

5 Entwicklung eines wissensbasierten Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels

5.1 Prinzipieller Ablauf der histogrammbasierten Bewertung eines Schwarmpartikels

Die Bewertung eines Partikels p_i lässt sich in drei Phasen einteilen: die Abfrage der benötigten Daten aus der Wissensbasis, die Erstellung der Tiefen- und Normalenvektorhistogramme aus den Tiefendaten T und der Vergleich der Histogramme aus der Wissensbasis mit den Histogrammen, die aus den Tiefendaten erstellt wurden, mithilfe der Fitnessfunktion.

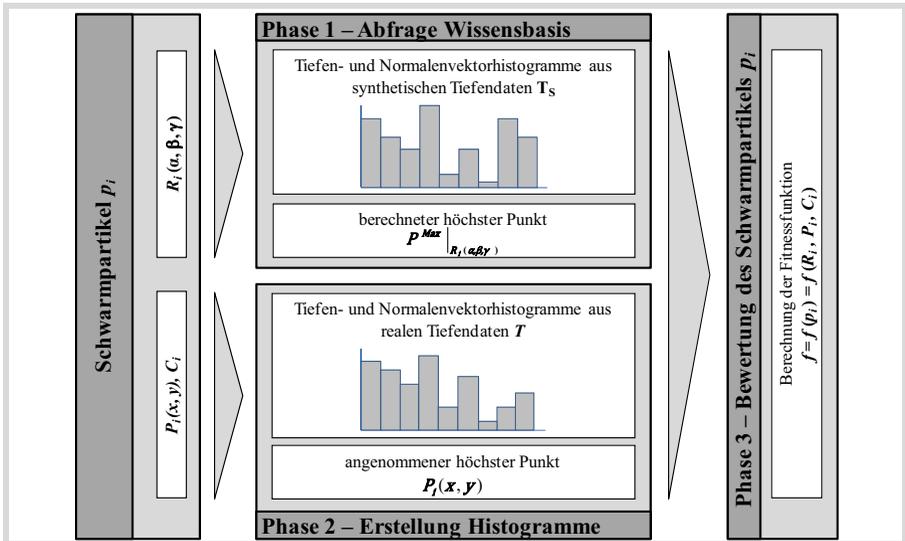


Abbildung 5-1: Prinzipieller Ablauf der histogrammbasierten Bewertung eines Schwarmpartikels

Hierbei beschreiben die Histogramme der Wissensbasis ein Objekt im Idealzustand einer Parallelprojektion mit synthetischen Tiefendaten T_S ohne Messrauschen. Je nach Auflösung und auftretenden Messfehlern des Sensors kann die Wissensbasis entsprechend angepasst werden. Zur Bewertung eines Schwarmpartikels stellt die Fitnessfunktion einen Zusammenhang zwischen den Histogrammen aus der Wissensbasis und den Sensordaten her. Die Fitnessfunktion, die entscheidend für den Vergleich zwischen Wissensbasis und realen Sensordaten ist, wird im Folgenden genauer betrachtet.

5.2 Ermittlung, Klassifizierung und Eingrenzung der Einflussfaktoren auf die histogrammbasierte Bewertung eines Schwarmpartikels

Die Bewertung eines Schwarmpartikels mittels Tiefen- und Normalenvektorhistogrammen werden von den folgenden Faktoren beeinflusst:

- Tiefenmessbereich des Sensors,
- Winkelauflösung des Sensors
- Öffnungswinkel des Sensors,
- Messfehler des Sensors,
- Abstand des Sensors zur Kiste,
- rotatorische Auflösung der Wissensbasis,
- Auflösung der Tiefenhistogramme,
- Wertebereich der Tiefenhistogramme,
- Auflösung der Normalenvektorhistogramme,
- Abmessungen des zu erkennenden Objekts,
- Abmessungen der Kiste,
- Wahl der Histogramm-Vergleichsfunktion,
- Umgang mit maskierten Bereichen in den Tiefendaten.

In Abbildung 5-2 werden die Einflussfaktoren klassifiziert und zusammengefasst.

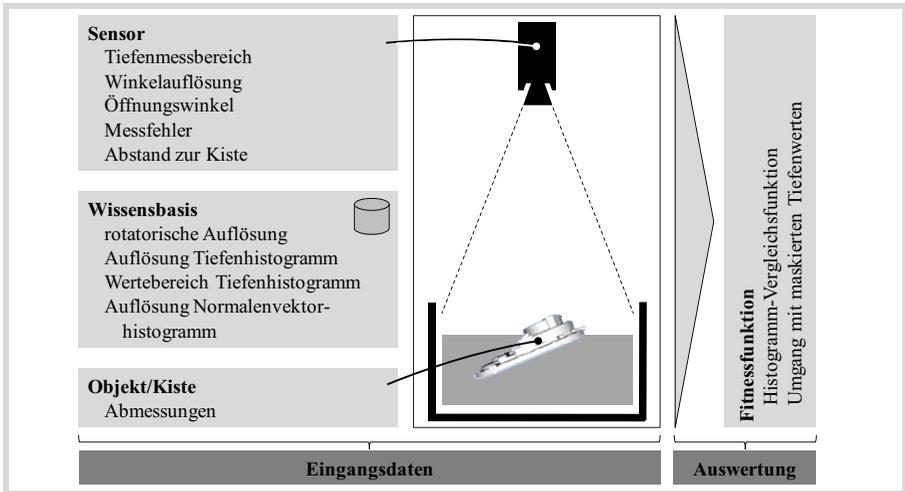


Abbildung 5-2: Einflussfaktoren auf die Bewertung eines Schwarmpartikels mit Tiefen- und Normalenvektorhistogrammen

5.3 Verfahren zur Erzeugung von Tiefendaten aus CAD-Daten und hiervon abgeleiteten Histogrammen für die Wissensbasis

5.3.1 Modellierung eines Tiefensensors zur Generierung der Tiefendaten aus CAD-Daten eines Objekts

Die Modellierung des Tiefensensors erfolgt auf Basis einer Parallelprojektion, um deren Vorteile für die Erstellung der Wissensbasis W auszunutzen. Trotz dieser Vereinfachung muss durch geeignete Parametereinstellungen erreicht werden, dass die generierten Tiefendaten mit den realen Tiefendaten korrespondieren.

Abbildung 5-3 zeigt das Ersatzmodell des realen Tiefensensors in Seitenansicht bei einem Schnitt durch die xz -Ebene sowie das Sensormodell zur Generierung der Tiefendaten aus CAD-Daten eines Objekts. Links in der Abbildung sind die charakteristischen geometrischen Parameter des Tiefensensors dargestellt, die bei der Modellierung des Sensors zur Erstellung der synthetischen Tiefendaten berücksichtigt werden. Die rechte Seite der Abbildung zeigt die geometrischen Parameter des modellierten Sensors sowie die für die Modellierung des Sensors wichtigen geometrischen Größen der CAD-Daten eines Objekts. Für den Schnitt durch die zweite Messebene des Tiefensensors, die yz -Ebene, gilt die Modellierung mit entsprechenden Parametern analog. Die Parameter des Modells wurden so gewählt, dass unterschiedliche Tiefensensoren mit dem Modell beschrieben und entsprechende synthetische Tiefendaten als Parallelprojektion erstellt werden können.

Als Ausgangsdaten liefert der modellierte Tiefensensor Tiefenwerte in Form einer quadratischen Matrix M_S mit Dimension n_{M_S} . Für die Dimension n_{M_S} , die laterale Auflösung des Sensors $a_{Schritt}$ und den Sichtbereich des Sensors a_{Gesamt} gilt der Zusammenhang:

$$n_{M_S} = \left\lceil \frac{a_{Gesamt}}{a_{Schritt}} + 0,5 \right\rceil \quad \text{Gleichung 5-1}$$

Hierbei wird der Quotient aus a_{Gesamt} und $a_{Schritt}$ auf die nächstgrößere ganze Zahl gerundet.

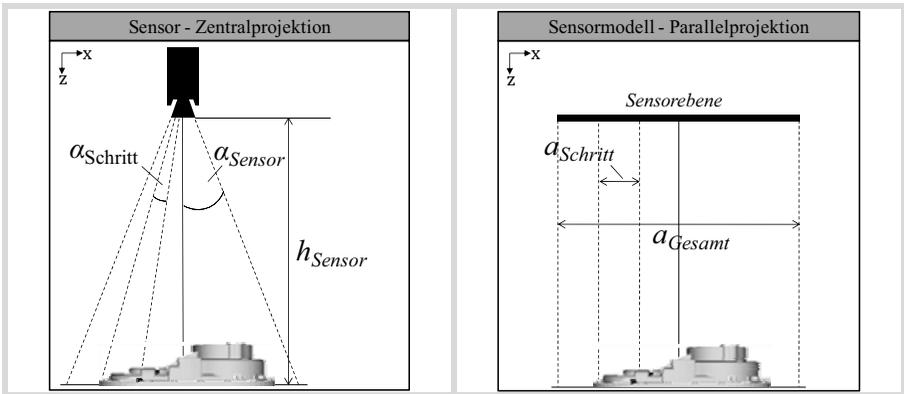


Abbildung 5-3: Darstellung der geometrischen Parameter eines Tiefensensors sowie die geometrischen Größen des Sensormodells und der CAD-Daten eines Objekts

5.3.2 Bestimmung des Sichtbereichs und der lateralen Auflösung des modellierten Tiefensensors

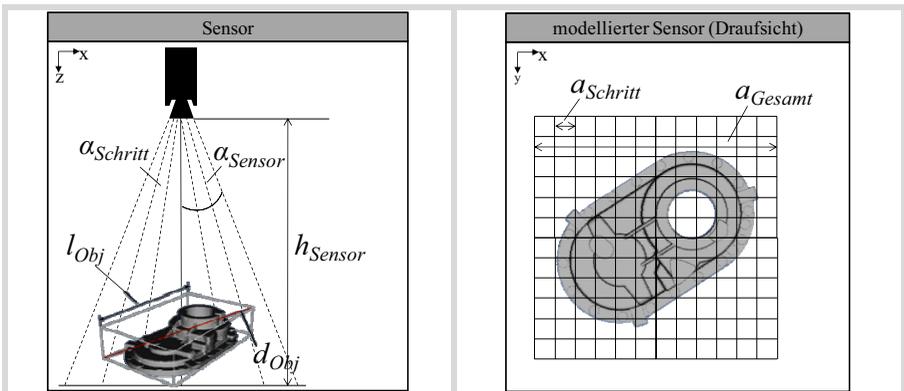


Abbildung 5-4: Schematische Darstellung der benötigten geometrischen Parameter zur Bestimmung des Sichtbereichs und der lateralen Auflösung des modellierten Tiefensensors

Zur Bestimmung der Dimension der Tiefenmatrix und des Sichtbereichs des modellierten Tiefensensors müssen folgende drei Parameter bekannt sein:

- d_{Obj} : Die Raumdiagonale der Bounding-Box um das Objekt
- α_{Sensor} : Der Scanbereich des realen Tiefensensors
- $\alpha_{Schritt}$: Die Winkelauflösung des realen Tiefensensors

Damit das Objekt in allen Orientierungen komplett im Sichtbereich a_{Gesamt} des modellierten Sensors zu liegen kommt und a_{Gesamt} möglichst gut ausgenutzt wird, muss a_{Gesamt} der längsten Ausdehnung des Objekts entsprechen. Dies ist die Raumdiagonale d_{Obj} der Bounding-Box. Somit gilt:

$$a_{Gesamt} = d_{Obj} \quad \text{Gleichung 5-2}$$

Für die Dimension der Matrix gilt Gleichung 5-1. Zusammen mit Gleichung 5-2 ergibt sich:

$$n_{M_S} = \left\lceil \frac{d_{Obj}}{a_{Schritt}} + 0,5 \right\rceil \quad \text{Gleichung 5-3}$$

Diese Berechnung ist unabhängig vom realen Tiefensensor. Der Parameter $a_{Schritt}$ kann beliebig vorgegeben werden.

Für die Erstellung der Wissensbasis zur Objektlageerkennung muss die laterale Auflösung der durch den modellierten Sensor generierten Tiefenmatrix mit den Daten des realen Sensors übereinstimmen. Aus diesem Grund gilt zusätzlich, dass $a_{Schritt}$ größer oder gleich dem größten Abstand zwischen zwei realen Messpunkten am Rand des Messbereichs mit maximalem Abstand zum Boden h_{Sensor} sein muss:

$$a_{Schritt} \geq h_{Sensor} \cdot [\tan(\alpha_{Sensor}) - \tan(\alpha_{Sensor} - \alpha_{Schritt})] \quad \text{Gleichung 5-4}$$

Insgesamt ergibt sich also, dass für die maximal zu wählende Dimension der quadratischen Tiefenmatrix gilt:

$$n_{M_S} \leq \left\lceil \frac{d_{Obj}}{h_{Sensor} \cdot [\tan(\alpha_{Sensor}) - \tan(\alpha_{Sensor} - \alpha_{Schritt})]} + 0,5 \right\rceil \quad \text{Gleichung 5-5}$$

Dieser Zusammenhang zwischen den realen Messwerten des Tiefensensors und der gewählten Dimension des modellierten Tiefensensors ist auch für die Emulation der Tiefenwerte aus den realen Messdaten wichtig (siehe Kapitel 4.3.3), da n_{M_S} auch für die Erstellung von M_T genutzt werden muss. Eine zu groß gewählte Dimension führt zu einem zu kleinen Abstand $a_{Schritt}$ zwischen zwei Matrixelementen, der leere Elemente in der Tiefenmatrix M_T bei der Emulation der Parallelprojektion aus realen Sensordaten erzeugt. Im Gegensatz dazu führt eine kleine Wahl der Dimension dazu, dass bei der Emulation der Parallelprojektion mehr Tiefenwerte in ein Matrixelement fallen und somit die Zuverlässigkeit des z-Werts in diesem Element erhöht wird.

Um auch für Tiefensensoren mit unterschiedlichen lateralen Auflösungen in x- und y-Richtung homogene Tiefendaten zu generieren, sollte in diesem Fall die laterale Auflösung mit dem größeren Abstand zwischen zwei Messpunkten für beide Richtungen gewählt werden.

5.3.3 Bestimmung des Tiefenbereichs des modellierten Tiefensensors

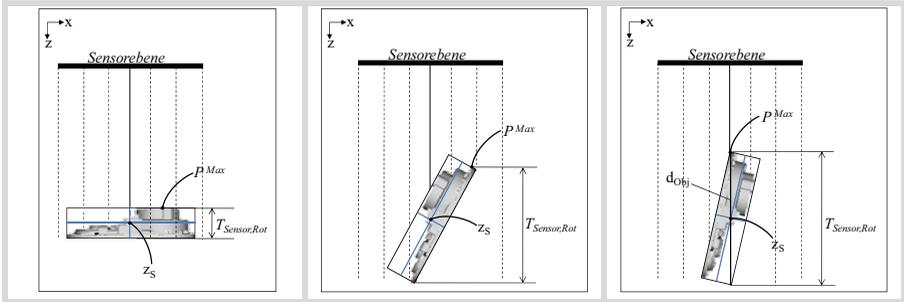


Abbildung 5-5: Darstellung verschiedener Orientierungen des Objekts mit dem Mittelpunkt der Bounding-Box z_S als Drehpunkt sowie den jeweils zugehörigen Tiefenbereichen T_{Sensor} und höchsten Punkten p^{Max}

Der Tiefenbereich $T_{Sensor,Rot}$, der von einem Objekt bedeckt wird, variiert je nach Orientierungen des Objekts (siehe Abbildung 5-5). Für die Erstellung der Tiefenhistogramme besteht die Notwendigkeit, einen allgemeinen Tiefenbereich T_{Sensor} für alle Objektorientierungen so vorzugeben, dass unabhängig von der jeweiligen Orientierung das Objekt komplett in den Tiefenbereich passt. Eine zu große Wahl von T_{Sensor} würde allerdings die hieraus erstellten Tiefenhistogramme ohne Not verschlechtern. Ein geeigneten Wert für T_{Sensor} bietet die Raumdiagonale d_{Obj} (siehe Abbildung 5-5 rechts). Der Tiefenbereich, der von dem modellierten Sensor vermessen werden muss, ergibt sich somit direkt aus dem Tiefenwert z_S des Schwerpunktes der Bounding-Box des Objekts sowie der Länge der Raumdiagonalen d_{Obj} der Bounding-Box:

$$T_{Sensor} = \left[z_S - \frac{d_{Obj}}{2}, z_S + \frac{d_{Obj}}{2} \right] \quad \text{Gleichung 5-6}$$

5.3.4 Erstellung der Tiefenhistogramme für die Wissensbasis

Tiefenhistogramme H_T beschreiben die Verteilung der Tiefenwerte in einer Szene, indem der Tiefenbereich T_{Sensor} in entsprechende Teilbereiche unterteilt wird. Die Feinheit von H_T wird durch die Anzahl $H_{T,Anz}$ der Teilbereiche bestimmt. Der Index $H_{T,Index}(i_x, i_y)$ des Teilbereichs, in den ein Tiefenwert $T_D(i_x, i_y)$ eingeordnet wird, ergibt sich aus:

$$H_{T,Index}(i_x, i_y) = \left| \frac{T_D(i_x, i_y) - \left(z_S - \frac{d_{Obj}}{2}\right)}{z_S + \frac{d_{Obj}}{2} - \left(z_S - \frac{d_{Obj}}{2}\right)} \cdot H_{T,Anz} \right| \quad \text{Gleichung 5-7}$$

Ein Objekt kann je nach Orientierung unterschiedliche Tiefenwerte im Tiefenbereich T_{Sensor} annehmen, außerdem variiert P^{Max} (siehe Abbildung 5-5). Der Schwerpunkt der Bounding-Box z_S bleibt jeweils konstant. Für eine Vergleichbarkeit der unterschiedlichen H_T muss P^{Max} , der zur Beschreibung einer Objektlage dient, immer in den gleichen Histogrammbereich sortiert werden. Dies kann durch eine virtuelle Verschiebung des Objekts in Richtung des Sensors erreicht werden, so dass P^{Max} nach der Verschiebung den Abstand 0 zur Sensorebene hat. Gleichung 5-7 kann dann vereinfacht werden zu:

$$H_{T,Index}(i_x, i_y) = \left| \frac{T_D(i_x, i_y)}{d_{Obj}} \cdot H_{T,Anz} \right| \quad \text{Gleichung 5-8}$$

In verschiedenen Tests hat sich gezeigt, dass bei der Wahl der Anzahl der Tiefenbereiche $H_{T,Anz}$ der systematische Messfehler $E_{systematisch}$ des eingesetzten Sensors beachtet werden sollte. Als geeignete Einstellung hat sich ergeben:

$$H_{T,Anz} = \left\lceil \frac{d_{Obj}}{2 \cdot E_{systematisch}} \right\rceil + 1 \quad \text{Gleichung 5-9}$$

5.3.5 Erstellung der Normalenvektorhistogramme für die Wissensbasis

Normalenvektorhistogramme H_N stellen die Verteilung der Steigung der Normalenvektoren $N = (n_x, n_y, n_z)^T$ eines Tiefenwertes zu seinen Nachbarn dar. Mit einem einfachen Ansatz ergibt sich hier für den \mathbb{R}^3 somit ein dreidimensionales Histogramm mit den Ableitungen in allen drei Dimensionen. Zwei Möglichkeiten, durch Dimensionsreduzierung eine zweidimensionale Darstellung zu erhalten, werden in [Leibe 2001] dargestellt.

Die erste Möglichkeit ist das einfache Weglassen der z-Komponente (es gilt $z \geq 0$), was einer Projektion der durch die normierten Normalenvektoren aufgespannten Halbkugel auf die xy-Ebene entspricht. Der Vorteil dieser Vorgehensweise ist, dass sie sehr einfach ist. Da sich die Fläche des Histogramms hierdurch auf einen Kreis innerhalb der zweidimensionalen Matrix beschränkt, gehen hierdurch allerdings wichtige Informationen verloren.

Die zweite Möglichkeit ist die Überführung der Normalenvektoren in eine Darstellung durch ein Paar von Winkeln $(\varphi, \theta) \in [-\pi/2, \pi/2]^2$ als Kugelkoordinaten. [Leibe 2000] zeigt, dass diese Möglichkeit geeigneter für die Erstellung der Normalenvektorhistogramme ist, da die Instabilität

der Darstellung durch eine einfache Drehung des Koordinatensystems an die Seite der Kugel verschoben wird. Ein zur effizienten Berechnung von Ableitungen eingesetztes Verfahren durch lokale Flächenapproximation wird von [Leibe 2001] vorgestellt und kann ausführlich auch [Jiang 1997] entnommen werden.

Für die lokale Flächenapproximation wird eine Approximationsfunktion $g(x, y)$ bestimmt, so dass in einer Umgebung eines Tiefenwertes der Fehler zwischen g und den Tiefenwerten T_D minimiert wird. Mit einer quadratischen Funktion

$$g(x, y) = c_1x^2 + c_2xy + c_3y^2 + c_4x + c_5y + c_6 \quad \text{Gleichung 5-10}$$

ergeben sich die in Abbildung 5-6 dargestellten Gleichungen. Da die Tiefendaten T_D äquidistant sind, können die 1. Ableitungen in Weltkoordinaten zur Bestimmung der Normalenvektoren durch die entsprechenden 1. Ableitungen in den Bildkoordinaten g_u und g_v dargestellt werden. Für g_u und g_v wurden die separierten eindimensionalen Faltmatrizen statt der zweidimensionalen Faltmatrizen angegeben, da dies den Rechenaufwand bei der Implementierung reduziert.

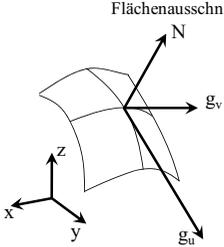
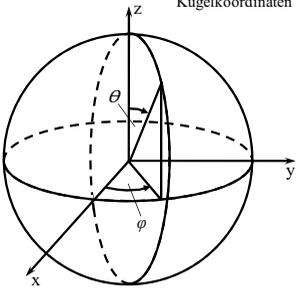
<p>Flächenausschnitt</p> 	$g_u = \frac{1}{50} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} (-2 \ -1 \ 0 \ 1 \ 2) * T_D(i_x, i_y)$	<p>Gleichung 5-11</p>
	$g_v = \frac{1}{50} \begin{pmatrix} 2 \\ 1 \\ 0 \\ -1 \\ -2 \end{pmatrix} (1 \ 1 \ 1 \ 1 \ 1) * T_D(i_x, i_y)$	<p>Gleichung 5-12</p>
<p>Kugelkoordinaten</p> 	$n_x = -\frac{g_u}{\sqrt{1 + g_u^2 - g_v^2}}$	<p>Gleichung 5-13</p>
	$n_y = -\frac{g_v}{\sqrt{1 + g_u^2 - g_v^2}}$	<p>Gleichung 5-14</p>
	$n_z = -\frac{1}{\sqrt{1 + g_u^2 - g_v^2}}$	<p>Gleichung 5-15</p>
	$\varphi = \arctan\left(\frac{n_z}{n_y}\right)$	<p>Gleichung 5-16</p>
	$\theta = \arctan\left(\frac{\sqrt{n_y^2 + n_x^2}}{n_x}\right)$	<p>Gleichung 5-17</p>

Abbildung 5-6: Gleichungen zur Berechnung der Normalenvektorhistogramme

5.3.6 Entwicklung eines Tools zur Erstellung der Wissensbasis

5.3.6.1 Aufbau des Tools zur Erstellung der Wissensbasis

Für das Tool zur Erstellung der Wissensbasis W wird die open-source Bibliothek Visualization Toolkit [VTK 2010] eingesetzt. VTK baut auf der plattform- und programmiersprachenunabhängigen Programmierschnittstelle OpenGL (Open Graphics Library) auf und stellt umfangreiche Funktionalitäten zur Bildbearbeitung und Visualisierung von 2-D- und 3-D-Objekten zur Verfügung. Mithilfe von VTK wird ein CAD-Modell eines Objekts im STL-Format eingelesen, in die benötigten Orientierungen rotiert und gerendert sowie die für die Wissensbasis benötigten Daten generiert.

5.3.6.2 Parametrierung des Tools

Um die Daten für die Wissensbasis an das aktuelle Objekt und den eingesetzten Sensor anpassen zu können, kann der Benutzer entsprechende Parameter vorgeben:

- CAD-Modell des Objekts im STL-Format
- Winkelbereiche $[\alpha^{Min}, \alpha^{Max}]$, $[\beta^{Min}, \beta^{Max}]$ und $[\gamma^{Min}, \gamma^{Max}]$ der Rotationen um die x-, y- und z-Achse in Eulerwinkeln
- Schrittweite s_α , s_β und s_γ der Rotationen um die x-, y- und z-Achse in Eulerwinkeln
- Tiefenbereich T_{Sensor} des Sensors
- Auflösung $H_{T,Anz}$ der Tiefenhistogramme (Anzahl der Teilbereiche)
- Auflösung $H_{N,Anz}$ der Normalenvektorhistogramme (Anzahl der Teilbereiche in jeder Dimension)
- Dimension n_{M_S} der Tiefenmatrix in Pixeln zur Einstellung der lateralen Auflösung

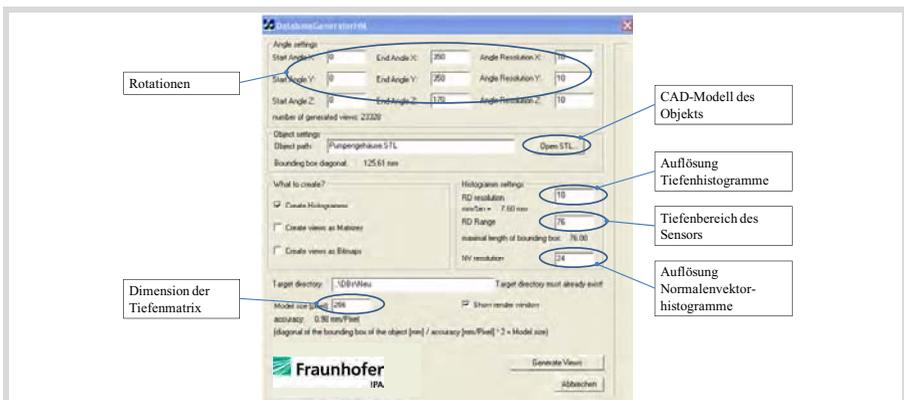


Abbildung 5-7: Benutzeroberfläche zur Parametrierung des Tools zur Erstellung der Wissensbasis

5.4 Aufstellung der Fitnessfunktion zur Bewertung eines Schwarmpartikels mithilfe des Histogrammvergleichs

5.4.1 Alternative Methoden des Histogrammvergleichs

Um die Histogramme aus der Wissensbasis W mit den Histogrammen zu vergleichen, die aus den Tiefendaten der realen Szene erstellt werden, gibt es eine Reihe von Möglichkeiten. Eine ausführliche Betrachtung von geeigneten Verfahren kann [Siggelkow 2002] entnommen werden. [Siggelkow 2002] legt seinen Schwerpunkt hierbei darauf, auch Histogramme vergleichen zu können, bei denen die Verteilung auf die Teilbereiche gegeneinander verschoben ist. Da im Rahmen dieser Arbeit durch die zuvor betrachteten Vorverarbeitungen die Histogrammgrößen und die zugehörigen Tiefenbereiche vereinheitlicht sind, ist diese Komplexität nicht notwendig. Vielmehr eignen sich einfache „bin-to-bin“-Vergleiche, die jeweils Histogramm-Bereiche mit gleichen Indizes miteinander vergleichen. Die wichtigsten dieser Verfahren werden im Folgenden kurz vorgestellt. $H^{(1)}$ und $H^{(2)}$ bezeichnen hierbei die beiden Histogramme mit der Dimension M . $h^{(1)}$ und $h^{(2)}$ bezeichnen die Gesamtzahl aller Messwerte in dem jeweiligen Histogramm. Ein Histogramm H wird durch einen Vektor $H = (H_0, H_1, \dots, H_{M-1})^T$ dargestellt.

5.4.1.1 Quadratischer euklidischer Abstand

Die einfachste Vergleichsmethode ist der quadratische euklidische Abstand

$$\varepsilon(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} (H_m^{(1)} - H_m^{(2)})^2. \quad \text{Gleichung 5-18}$$

Geometrisch lässt sich diese Vergleichsmethode als die Länge – genauer das Quadrat der Länge – einer $(M - 1)$ -dimensionalen Vektor-Differenz auffassen. Für eine perfekte Übereinstimmung zwischen den beiden Histogrammen liefert die Differenzberechnung den Wert null. Großer Nachteil des euklidischen Abstands ist die bekannte Empfindlichkeit gegenüber Störungen in den Daten, die auch bei den später untersuchten realen Sensordaten auftreten.

5.4.1.2 Histogramm-Durchschnitt

Eine weitere Vergleichsmethode ist der Durchschnitt von zwei Histogrammen, der sich aus der Summe der Mindesteinträge der einzelnen Teilbereiche ergibt:

$$\cap(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} \min(H_m^{(1)} - H_m^{(2)}). \quad \text{Gleichung 5-19}$$

Anschaulich berechnet diese Methode den übereinstimmenden Teil der einzelnen Teilbereiche. Bei einer perfekten Übereinstimmung von zwei Histogrammen liefert diese Berechnung die Summe aller im Histogramm vorhandenen Messwerte. Durch einen Normierungsschritt kann ein Maximalwert von 1 erreicht werden. Trotz erfolgreicher Anwendung zur Klassifikation von Farb-Histogrammen [Swain 1991] ist ein Nachteil der Methode der hohe Anteil an Fehl-Klassifikationen.

5.4.1.3 Kullback-Leibler-Divergenz

Die Kullback-Leibler-Divergenz [Kullback 1951] oder relative Entropie ist allgemein für zwei Wahrscheinlichkeitsdichtefunktionen p_1 und p_2 definiert als:

$$K(p_1, p_2) = \int p_1(\xi) \log \frac{p_1(\xi)}{p_2(\xi)} d\xi. \tag{Gleichung 5-20}$$

Für den diskreten Fall mit zwei Histogrammen ändert sich die Formel zu:

$$K(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} H_m^{(1)} \log \frac{H_m^{(1)}}{H_m^{(2)}}. \tag{Gleichung 5-21}$$

Die Kullback-Leibler-Divergenz hat ihren Ursprung in der Informationstheorie. Mit ihrer Hilfe lässt sich beschreiben, wie viele Bereiche eines Histogramms ungenutzt bleiben, wenn bei der Codierung $H^{(1)}$ statt $H^{(2)}$ verwendet würde. Folglich bedeutet ein großer Wert eine starke Abweichung der beiden Darstellungen, während ein kleiner Wert eine höhere Übereinstimmung zeigt. Nachteil ist der hohe Aufwand zur Berechnung der Kullback-Leibler-Divergenz durch die Verwendung des Logarithmus.

5.4.1.4 Minkowski-Abstand

Der Minkowski-Abstand ist eine allgemeine Definition, um den Abstand zwischen zwei Punkten im $(M - 1)$ -dimensionalen Raum zu berechnen. Sie leiten sich aus den p -Normen ab. Der Minkowski-Abstand wird über folgende Formel berechnet:

$$L_p(H^{(1)}, H^{(2)}) = \left(\sum_{m=0}^{M-1} |H_m^{(1)} - H_m^{(2)}|^p \right)^{1/p} \tag{Gleichung 5-22}$$

Vorteil dieser Vergleichsmethode ist die einfache Berechnung (in Abhängigkeit der Wahl von p). Für $p = 2$ wird die Minkowski-Distanz zur euklidischen Distanz. Großer Nachteil dieser Methode ist, dass mit steigendem p große Differenzen durch die Potenzbildung immer stärker gewichtet werden und damit einzelne Ausreißer die Gesamtbewertung stören können.

5.4.1.5 χ^2 -Divergenz

Der χ^2 -Test hat seinen Ursprung in der Statistik, um zwei Häufigkeitsverteilungen miteinander zu vergleichen. Nach der Berechnung der Abweichung der beiden Häufigkeitsverteilungen wird unter Vorgabe eines Signifikanz-Niveaus entschieden, ob die beiden Histogramme als gleich bewertet werden. Da die bei der Objektlageerkennung erzeugten Histogramme aus realen Messdaten zu sehr niedrigen Signifikanz-Niveaus führen, wird im Rahmen dieser Arbeit, wie von [Hetzel 2001] vorgeschlagen, die einfache χ^2 -Divergenz für den Vergleich von zwei Histogrammen betrachtet.

Die häufigste Form zur Berechnung der χ^2 -Divergenz ist für den Einstichprobenfall definiert. Hierbei wird das Histogramm, das aus einer Datenmenge mit unbekannter Wahrscheinlichkeitsverteilung erstellt wurde, mit einem Histogramm mit einer bekannten idealen Verteilung oder mit einem anderen Histogramm mit gleicher Anzahl an Messwerten verglichen. Für den Vergleich einer idealen Verteilung $P = (p_0, p_1, \dots, p_{M-1})^T$, mit p_0, \dots, p_{M-1} als ideale Wahrscheinlichkeitsverteilung für die Histogrammzellen, wird die χ^2 -Divergenz durch folgende Formel ermittelt:

$$\chi^2(H^{(1)}, P) = \sum_{m=0}^{M-1} \frac{(H_m^{(1)} - h^{(1)}p_m)^2}{h^{(1)}p_m} \quad \text{Gleichung 5-23}$$

Für den Vergleich zweier Histogramme mit gleicher Anzahl an Messwerten sieht die Formel folgendermaßen aus:

$$\chi^2(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} \frac{(H_m^{(1)} - H_m^{(2)})^2}{H_m^{(1)} + H_m^{(2)}} \quad \text{Gleichung 5-24}$$

In der Fachliteratur kommen hierbei sowohl eine unsymmetrische (Gleichung 5-23) als auch eine symmetrische Formulierung (Gleichung 5-24) zum Einsatz, die sich darin unterscheiden, ob lediglich durch den Inhalt eines Histogramms dividiert wird, oder ob durch die Summe aus beiden Histogrammeinträgen dividiert wird. Vorteil der symmetrischen Formulierung ist, dass sie deutlich gemäßiger auf Histogramm-Bereiche ohne Messwerte reagiert, da durch die Summation ein Ergebnis der Division, das gegen Unendlich strebt, verhindert wird.

Für die Anwendung in der Objektlageerkennung ist ein weiterer Fall, der Zweistichprobenfall, interessanter, bei dem untersucht wird, ob zwei Datenmengen Stichproben aus einer gleichen, aber unbekanntem Verteilung sein können. Gleichzeitig wird hierbei der Einfluss, den unterschiedliche Datenmengen in den Histogrammen haben können, kompensiert:

$$\chi^2(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} \left(\frac{\left(H_m^{(1)} - h^{(1)} \frac{H_m^{(1)} + H_m^{(2)}}{h^{(1)} + h^{(2)}} \right)^2}{h^{(1)} \frac{H_m^{(1)} + H_m^{(2)}}{h^{(1)} + h^{(2)}}} + \frac{\left(H_m^{(2)} - h^{(2)} \frac{H_m^{(1)} + H_m^{(2)}}{h^{(1)} + h^{(2)}} \right)^2}{h^{(2)} \frac{H_m^{(1)} + H_m^{(2)}}{h^{(1)} + h^{(2)}}} \right) \quad \text{Gleichung 5-25}$$

5.4.1.6 Folgerung

Die Betrachtung alternativer Methoden zum Histogrammvergleich hat die Grundlagen für das weitere Vorgehen zur Entwicklung eines Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels gelegt. Die zu erwartende fehlende Robustheit gegen Störungen in den Daten beziehungsweise der zu erwartende hohe Anteil an Fehl-Klassifikationen führen dazu, dass der quadratische euklidische Abstand, die Minkowski-Distanz sowie der Histogramm-Durchschnitt für die Entwicklung des Verfahrens nicht weiter betrachtet werden. Des Weiteren legt der hohe Berechnungsaufwand zur Auswertung der Kullback-Leibler-Divergenz durch die Verwendung des Logarithmus die Entscheidung nahe, auch diese Methode nicht weiter zu betrachten.

Somit kommt für die weitere Entwicklung des Verfahrens die χ^2 -Divergenz zum Einsatz. Wichtig ist auch, dass die Erweiterung der in der Betrachtung dargestellten 1-dimensionalen Variante auf den 2-dimensionalen Fall zum Vergleich der Normalenvektorhistogramme leicht möglich ist.

5.4.2 Erweiterung der χ^2 -Divergenz für den Einsatz als Fitnessfunktion

Zusätzlich zu den Tiefen- und Normalenvektorhistogrammen liegen noch zwei weitere Informationen vor, die neben dem reinen Histogrammvergleich in die Berechnung der Fitnessfunktion zur Bewertung der Position eines Schwarmpartikels einbezogen werden sollen: die Anzahl der maskierten Werte $Anz_{maskiert}$ innerhalb C_i , da Schwarmpartikel, die viele maskierte Werte beinhalten, schlechter bewertet werden sollen, und die Anzahl der Messpunkte $C_{i,Anz}$ der Objektmerkmalsdaten C_i , da Schwarmpartikel, die viele Tiefenwerte für die Bewertung einbeziehen, höher bewertet werden sollen.

Maskierte Werte entstehen entweder durch Abschattungen in den Sensordaten, oder weil für einen Matrixeintrag aufgrund der vorliegenden Sensordaten kein zuverlässiger Wert berechnet werden konnte. Als Erweiterung, um die Anzahl solcher maskierten Werte $Anz_{maskiert}$ einzubeziehen, wurde in dieser Arbeit folgende Vorgehensweise gewählt. Durch Interpretation von $Anz_{maskiert}$ als eigenen, virtuellen Histogramm-Bereich, der nur für reale Tiefendaten Einträge enthalten kann (da es bei synthetischen Daten keinen Grund für Maskierungen gibt), ergeben sich zwei Vorteile. Zum einen werden hierdurch die Gesamtzahlen aller Messwerte $h^{(1)}$ und $h^{(2)}$ in den beiden Histogrammen trotz maskierter Werte wieder gleich, so dass die χ^2 -

Divergenz nach Gleichung 5-24 berechnet werden kann. Zum anderen bewirkt der stets leere Histogramm-Bereich der synthetischen Tiefendaten, dass nach Abschluss der χ^2 -Divergenz-Berechnung für die tatsächlichen Tiefenbereiche einfach $Anz_{maskiert}$ auf den aktuellen Wert der χ^2 -Divergenz aufaddiert werden kann.

Um die Anzahl der Messpunkte $C_{i,Anz}$ der Objektmerkmalsdaten C_i in die Fitnessfunktion einfließen zu lassen, erfolgt nach Abschluss der Berechnung der χ^2 -Divergenz noch eine Normierung des Wertebereichs der χ^2 -Divergenz sowie eine zusätzliche Gewichtung von Objektmerkmalsdaten C_i mit einer großen Anzahl an Messpunkten $C_{i,Anz}$ mithilfe der maximalen Anzahl der Messpunkte $C_{i,Anz}^{Max}$ und der minimalen Anzahl der Messpunkte $C_{i,Anz}^{Min}$ aller in der Wissensbasis W vorhandenen Objektorientierungen.

Die Normierung erfolgt aufgrund folgender Abschätzung:

$$\chi^2(H^{(1)}, H^{(2)}) = \sum_{m=0}^{M-1} \frac{(H_m^{(1)} - H_m^{(2)})^2}{H_m^{(1)} + H_m^{(2)}} \leq \sum_{m=0}^{M-1} \frac{(H_m^{(1)} + H_m^{(2)})^2}{H_m^{(1)} + H_m^{(2)}} = 2 \cdot C_{i,Anz} \quad \text{Gleichung 5-26}$$

Die Gewichtung der Objektmerkmalsdaten C_i mit einer großen Anzahl an Messpunkten $C_{i,Anz}$ erfolgt durch die Einführung eines Gewichtungsfaktors mit Wertebereich zwischen 0 und 1:

$$\bar{\chi}^2(H^{(1)}, H^{(2)}) = \frac{1}{2} \left(2 - \frac{C_{i,Anz} - C_{i,Anz}^{Min}}{C_{i,Anz}^{Max} - C_{i,Anz}^{Min}} \right) \cdot \chi^2(H^{(1)}, H^{(2)}) \quad \text{Gleichung 5-27}$$

Insgesamt ergibt sich damit die Berechnung der Fitnessfunktion zu:

$$f(p_i) = \frac{\chi^2(H^{(1)}, H^{(2)}) + Anz_{maskiert}}{4 \cdot C_{i,Anz}} \cdot \left(2 - \frac{C_{i,Anz} - C_{i,Anz}^{Min}}{C_{i,Anz}^{Max} - C_{i,Anz}^{Min}} \right) \quad \text{Gleichung 5-28}$$

5.4.3 Vergleichbarkeit der synthetischen Tiefendaten mit den realen Tiefendaten

5.4.3.1 Eindeutige Bestimmung von P^{Max}

In den meisten Fällen der synthetischen Tiefendaten T_S gibt es keinen explizit bestimmbar Punkt P^{Max} . Ursache hierfür ist die Geometrie des Objekts, wenn beispielsweise eine ebene Fläche parallel zum Sensor liegt. In diesen Fällen muss dennoch eindeutig definiert sein, welcher Punkt als P^{Max} zu wählen ist.

Sei P_{OL} die Menge aller höchsten Punkte (Maxima) von T_S . Um die Eindeutigkeit bei der Bestimmung des Maximums zu gewährleisten, wird das Maximum mit der niedrigsten x - und y -Koordinate gewählt.

Sei \mathbf{X}^{Max} die Menge aller x -Koordinaten von P_{OL} und \mathbf{Y}^{Max} die Menge aller y -Koordinaten von P_{OL} , dann wird P^{Max} bestimmt durch

$$P^{\text{Max}} = \min_{x \in \mathbf{X}^{\text{Max}}} \min_{y \in \mathbf{Y}^{\text{Max}}} P_{OL}.$$

5.4.3.2 Auswahl des Tiefenbereichs aus den realen Sensordaten

Ziel bei der Erstellung der Tiefenhistogramme aus den synthetischen Tiefendaten \mathbf{T}_S ist es, den höchsten Punkt P^{Max} eines Objektes immer auf die Sensorebene des modellierten Sensors zu setzen. Eine einfache Möglichkeit, die die Anforderungen für den Praxiseinsatz erfüllt und deshalb im Rahmen dieser Arbeit zum Einsatz kommt, ist die Verschiebung aller Punkte um den Betrag der Differenz zwischen P^{Max} und der Sensorebene (siehe Kapitel 5.3.4).

Eine analoge Verschiebung muss für die realen Tiefendaten \mathbf{T}_D erfolgen, um die Vergleichbarkeit zu gewährleisten. In \mathbf{T}_D wird hierzu analog zu P^{Max} für \mathbf{T}_S das Minimum von h^{Min} bestimmt und eine Verschiebung auf die Sensorebene durchgeführt. Zusätzlich müssen die Tiefenwerte, die definitiv nicht zu dem obersten Objekt gehören können, da sie einen größeren Abstand von h^{Min} als die Raumdiagonale d_{Obj} der Bounding-Box lang ist, ausgeblendet werden.

Für ein ermitteltes Minimum von \mathbf{T}_D mit einem Tiefenwert h^{Min} ergeben sich die Indizes der Tiefenhistogramme entsprechend zu:

$$H_{T,Index}(i_x, i_y) = \begin{cases} \left\lfloor \frac{T_D(i_x, i_y)}{d_{Obj}} \cdot H_{T,Anz} \right\rfloor, & T_D(i_x, i_y) \geq h^{\text{min}} + d_{Obj} \\ -1, & \text{sonst} \end{cases} \quad \text{Gleichung 5-29}$$

wobei -1 als Standardwert gesetzt wird, der die Tiefenwerte markiert, die nicht im Tiefenbereich liegen und somit nicht in das Histogramm einsortiert werden.

Durch Erweiterung dieser Vorgehensweise auf lokale Minima in den Tiefendaten können auch weitere Objekte, die nicht an oberster Stelle in der Kiste liegen, für die Histogrammerstellung und damit für die Objektlageerkennung genutzt werden.

5.5 Bewertung der durch die Wissensbasis erzeugten Lösungsräume

Die Gestalt des Lösungsraums – beispielsweise die Anzahl oder die Ausprägtheit der Minima – hat direkten Einfluss auf die Partikel-Schwarm-Optimierung. In der vorliegenden Arbeit wurde eine histogrammbasierte Fitnessfunktion für die Bewertung einer Partikelposition im Lösungsraum gewählt. Dabei erstellt ein Software-Tool automatisch die Wissensbasis anhand der CAD-Daten des aktuell gewählten Objekts. Um den dabei entstehenden Lösungsraum zu bewerten und damit abschätzen zu können, wie schwierig es der PSO vermutlich fällt, das

Minimum zu bestimmen, liefert die Betrachtung des Lösungsraums mit einer einfachen analytischen Bewertung der auftretenden Minima wichtige Anhaltspunkte.

Diese Bewertung des Lösungsraums arbeitet auf Basis synthetisch erzeugter Tiefendaten mit einzelnen Objekten, die direkt aus dem Tool zur Erstellung der Wissensbasis gewonnen werden können. Auf diese Weise ist es möglich, wichtige Informationen darüber zu liefern, welche Objektorientierungen bereits bei optimalen Randbedingungen aufgrund der Gestalt des Lösungsraums eine zuverlässige und welche eine unsichere Lageerkennung erwarten lassen. Diese Informationen über Objektlagen, die eine unsichere Lageerkennung erwarten lassen, fließen in die Planung der Tests des folgenden Kapitels ein.

Im Folgenden wird zur Vereinfachung der Schreibweise die Bewertung unter der Annahme durchgeführt, dass in der Wissensbasis die Daten eines einzigen Objekts vorhanden sind. Für eine Wissensbasis mit mehreren Objekten gelten die Aussagen analog für den Teil der Daten aus der Wissensbasis, die dem aktuell ausgewählten Objekt zugeordnet sind.

Ein Schwarmpartikel p_i bewegt sich bei der Suche nach der optimalen Objektlage durch einen fünfdimensionalen Lösungsraum – drei rotatorische Parameter für R_i und zwei translatorische Parameter für P_i . Da für eine vorgegebene Objektlage die Objektposition $P^{Max}|_{(\alpha,\beta,\gamma)}$ eindeutig bestimmt und konstant gehalten werden kann, können die beiden translatorischen Parameter für die Analyse des Lösungsraums vernachlässigt werden, so dass der nun zu betrachtende Lösungsraum L_R dreidimensional ist.

Für die Bewertung eines Lösungsraums werden aufgrund ihres Einflusses auf das korrekte Auffinden des Minimums durch ein Optimierungsverfahren folgende Eigenschaften des Lösungsraums betrachtet:

- Anzahl der Minima
- relative Qualität der lokalen Minima:
 - euklidische Distanz zum globalen Minimum
 - Differenz des Funktionswerts zwischen lokalem und globalem Minimum
- Größe des Einzugstrichters des globalen Minimums

Die Hauptaufgabe besteht somit darin, die lokalen und das globale Minimum zu bestimmen und hieraus die oben genannten Eigenschaften auszuwerten. Der Lösungsraum ist eine diskrete, skalarwertige Funktion, die einem dreidimensionalen äquidistanten Gitter skalare Werte mithilfe der Fitnessfunktion f zuweist.

Eine Möglichkeit, Minima einer Funktion zu bestimmen, besteht darin, die Nullstellen der ersten Ableitung zu betrachten. Für diskrete Funktionen können Ableitungen mithilfe numerischer

Verfahren zur numerischen Differentiation wie beispielsweise der Finite-Differenzen-Approximation angenähert werden [Quarteroni 2002]. Ein weiterer Weg, der im Rahmen dieser Arbeit verfolgt wird, ist, direkt auf die Definition eines Minimums für diskrete Funktionen zurückzugreifen. L_R hat dann ein lokales Minimum in einem Punkt $w_0 \in W$, wenn

$$L_R(w) \geq L_R(w_0) \quad \forall w \in N(w_0) \tag{Gleichung 5-30}$$

wobei $N(w_0)$ die Nachbarn von w_0 bezeichnen, die in den drei Rotationsrichtungen α, β, γ eine Schrittweite $s_\alpha, s_\beta, s_\gamma$ entfernt sind.

Da das globale Minimum w_0^{Min} von L_R der vorgegebenen Objektrotation entspricht, können mit Hilfe dieser einfachen Vergleichsoperationen die Anzahl der Minima des Lösungsraums sowie die relative Qualität der einzelnen Minima bestimmt werden.

Die Bestimmung der Größe des Einzugstrichters $T_E(w_0^{Min})$ des globalen Minimums w_0^{Min} erfolgt durch Betrachtung aller Objektrotationen $w_E \in W$, die den Bedingungen genügen, dass

$$\begin{aligned} (i) \quad & L_R(w) \geq L_R(w_E) \quad \forall w \in N_A(w_E) \\ (ii) \quad & w \in T_E(w_0^{Min}) \quad \forall w \in N_A(w_E) \end{aligned} \tag{Gleichung 5-31}$$

wobei $N_A(w_E) \subseteq N(w_E)$ die Nachbarn von w_E bezeichnen, die einen kleineren oder gleichen euklidischen Abstand zu w_0^{Min} haben wie w_E . Da $w_0^{Min} \in T_E(w_0^{Min})$ gilt, wird für die Bestimmung von $T_E(w_0^{Min})$ im Rahmen dieser Arbeit ein Regionenwachstumsverfahren [Rosenfeld 1982] eingesetzt.

6 Realisierung und Verifizierung des Gesamtsystems

6.1 Zielsetzung und Planung der Versuchsdurchführung

In den vorhergehenden Kapiteln wurde ein histogrammbasiertes Verfahren zur Objektlageerkennung mit Partikel-Schwarm-Optimierung konzipiert und entwickelt. Dieses Verfahren soll nun getestet und abschließend in einer Roboterzelle für den Praxiseinsatz untersucht werden.

Ziel der Untersuchungen ist es, Aussagen über die Anpassbarkeit an verschiedene Objekte, die Genauigkeit der Objektlageerkennung, die Auswertzeit, die Zuverlässigkeit der Erkennung anhand der Erkennungsrate sowie die Robustheit gegenüber inhomogenem Hintergrund und Messfehlern in den Sensordaten zu erhalten.

Die Untersuchungen sind hierzu in drei Teile aufgeteilt. Teil 1 arbeitet mit einzelnen, synthetisch aus CAD-Daten erzeugten Objektlagen. Auf Basis dieser Objektlagen können allgemeine Aussagen zu den Parametereinstellungen der Partikel-Schwarm-Optimierung abgeleitet werden, um eine zuverlässige Konvergenz zu erhalten. Aufbauend auf diesen Ergebnissen werden mit den ermittelten Parametereinstellungen im zweiten Teil der Untersuchungen Szenen mit mehreren Objekten in synthetisch erstellten Daten untersucht und abschließend im dritten Teil eine Roboterzelle aufgebaut, die anhand realer Sensordaten Getriebewellen in einer Kiste lokalisiert und entnimmt.

Als Testobjekte kommen die drei Objekte – Getriebewellen, Hohlringe und Gehäusedeckel - zum Einsatz, die in Kapitel 1 innerhalb der potentiellen Einsatzszenarien betrachtet wurden (siehe Abbildung 6-1).



Abbildung 6-1: Die drei Testobjekte für die durchgeführten Untersuchungen

Insbesondere im ersten Teil der Untersuchungen sollen jeweils die wichtigsten Parameter der Partikel-Schwarm-Optimierung variiert und deren Einfluss auf die Erkennungsrate ermittelt werden. Im Einzelnen sind dies folgende Parameter:

- Schwarmgröße
- Anzahl der Iterationsschritte
- Nachbarschaftsgröße

Für die Evaluierung des entwickelten Verfahrens zur PSO-basierten Objektlageerkennung wird der Constriction Factor eingesetzt sowie die Gewichtungen des kognitiven Verhaltens auf $\varphi_1 = 2.8$ und des sozialen Verhaltens auf $\varphi_2 = 1.3$ gesetzt werden. Als Nachbarschaftstopologie wird in allen Tests die übliche Sternstruktur gewählt. Die Nachbarschaftsgröße variiert jeweils von einer lbest Nachbarschaft mit Größe eins bis zu einer gbest Nachbarschaft. Als Testlagen werden jeweils Lagen ausgewählt, die Aufgrund der Struktur und der Analyse der Lösungsräume als schwierig eingestuft werden. Als schwierig für die Optimierung werden vier Situationen betrachtet:

- Lagen, deren Lösungsräume viele Minima in der Umgebung des globalen Minimums besitzen,
- Lagen, deren Lösungsräume einen kleinen euklidischen Abstand zwischen lokalem und globalem Minimum in der Umgebung des globalen Minimums aufweisen,
- Lagen, deren Lösungsräume ein lokales Minimum mit einem großen Einzugstrichter besitzen sowie
- Lagen, deren Lösungsräume ein globales Minimum mit einem kleinen Einzugstrichter besitzen.

6.2 Erstellung der Wissensbasis für die gewählten Testobjekte

Die Wissensbasis wurde zur Durchführung der Tests für die drei ausgewählten Testobjekte erstellt. Als Parameter wurden folgende Werte gewählt:

- Laterale Auflösung des Sensors: 3,5 mm
- Wertebereich der Tiefenhistogramme: 256 mm
- Auflösung der Tiefenhistogramme: 32 Teilbereiche
- Auflösung der Normalenvektorhistogramme: 24 Teilbereiche pro Dimension
- rotatorische Auflösung der Wissensbasis: 5 Grad

Durch die Festlegung der Parameter ist sichergestellt, dass aufgrund der Wahl des realen Sensors die gleiche Wissensbasis sowohl für die Tests mit synthetischen Daten als auch für die Tests mit realen Sensordaten genutzt werden kann. Somit ist eine Übertragbarkeit der jeweils ermittelten Testergebnisse gegeben.

6.3 Experimentelle Untersuchungen an synthetischen Daten einzelner Objekte

6.3.1 Generierung der Testdaten mithilfe des modellierten Tiefensensors

Für eine erste Untersuchung des PSO-basierten Verfahrens zur Objektlageerkennung werden synthetische Szenen mit einzelnen Objekten betrachtet. Das Tool zur Erstellung der Wissensbasis bietet hierzu neben der Erstellung der Histogramme die Möglichkeit, synthetische Tiefendaten der aktuellen Objektrotation zu erstellen und abzuspeichern. Die laterale Auflösung kann dabei manuell eingestellt werden und wurde für die Untersuchungen so eingestellt, dass sie der Einstellung der Wissensbasis entspricht.

6.3.2 Auswahl der untersuchten Objektlagen

Die Validierung des entwickelten Systems zur Objektlageerkennung erfolgt anhand umfangreicher Tests mit unterschiedlichen Objekten in unterschiedlichen Lagen. Für die Tests wurden mehrere Parameter-Varianten für die Schwarmoptimierung vorgenommen und bewertet. Abbildung 6-2 zeigt eine Auswahl an untersuchten Objektlagen mit den zuvor definierten Eigenschaften des Lösungsraums der Beispielobjekte Hohlrad, Getriebewelle und Gehäusedeckel, mit deren Hilfe die erzielten Ergebnisse dargestellt werden.

	kleiner euklidischer Abstand zwischen lokalem und globalem Minimum	viele Minima im Lösungsraum	großer Einzugstrichter eines lokalen Minimums	kleiner Einzugstrichter des globalen Minimums
Hohlrad	$(\alpha/\beta/\gamma) = (120/0/0)$	$(\alpha/\beta/\gamma) = (135/0/270)$	$(\alpha/\beta/\gamma) = (180/0/255)$	$(\alpha/\beta/\gamma) = (210/0/315)$
Getriebewelle	$(\alpha/\beta/\gamma) = (0/120/225)$	$(\alpha/\beta/\gamma) = (0/150/195)$	$(\alpha/\beta/\gamma) = (0/105/165)$	$(\alpha/\beta/\gamma) = (0/210/60)$
Gehäusedeckel	$(\alpha/\beta/\gamma) = (60/15/0)$	$(\alpha/\beta/\gamma) = (210/125/10)$	$(\alpha/\beta/\gamma) = (240/120/300)$	$(\alpha/\beta/\gamma) = (45/180/240)$

Abbildung 6-2: Übersicht über die ausgewählten Objektlagen für die getesteten Objekte

6.3.3 Getestete Schwarmparameter und erzielte Erkennungsraten

Entscheidend für die Bewertung des entwickelten Systems ist die Zuverlässigkeit der Erkennung, das heißt die erzielbare Erkennungsrate der einzelnen Objektlagen. Der Einfluss der Schwarmgröße und Nachbarschaftsgröße sowie der Anzahl der Iterationen kann den folgenden Schaubildern (Abbildung 6-3 bis Abbildung 6-6) entnommen werden. Jeder Versuchsdurchlauf mit den vorgegebenen Parametern wurde hierbei 50-mal wiederholt.

Die Schwarmgröße variiert zwischen 10 und 100 Partikeln, die Nachbarschaftsgröße zwischen einem Nachbarn und einer gbest-Nachbarschaft. Für das Verhältnis von kognitivem zu sozialem Verhalten wurden – wie von [Carlisle 2001] vorgeschlagen – standardmäßig die Werte $\varphi_1 = 2.8$ und $\varphi_2 = 1.3$ gewählt.

Die Partikel bewegen sich während der Optimumsuche stetig durch den Lösungsraum, während die Wissensbasis diskret aufgebaut ist. Für die Bewertung eines Partikels anhand der Wissensbasis wird deshalb die Position eines Partikels auf die jeweils nächstliegenden Winkel der Wissensbasis gerundet. Aus dem gleichen Grund wird auch die bei der Schwarmuche gefundene Objektlage als korrekt eingestuft, wenn die Abweichung von der tatsächlich vorgegebenen Lage maximal der Schrittweite der Wissensbasis, also 5° , entspricht.

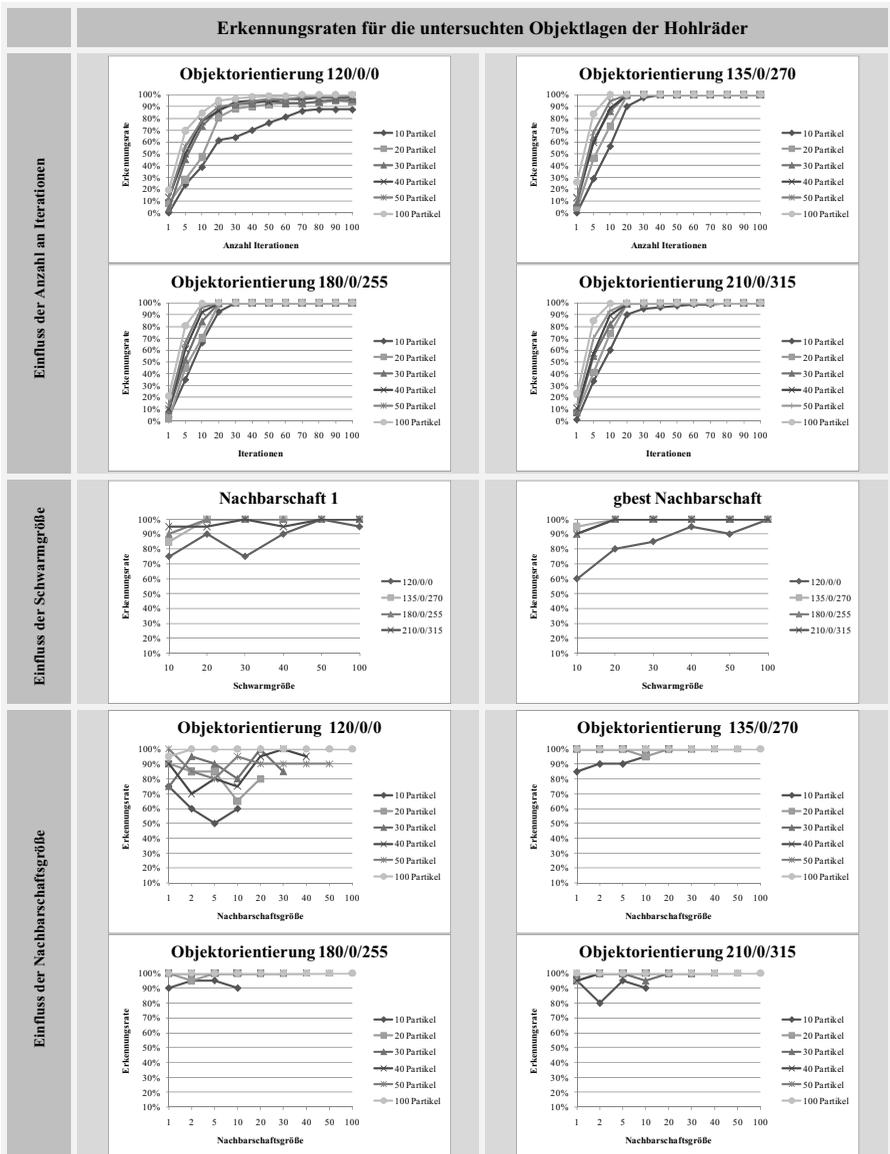


Abbildung 6-3: Einfluss der Anzahl an Iterationen, der Schwarmgröße und der Nachbarschaftsgröße auf die Erkennungsraten bei der Erkennung eines einzelnen Hohlrads

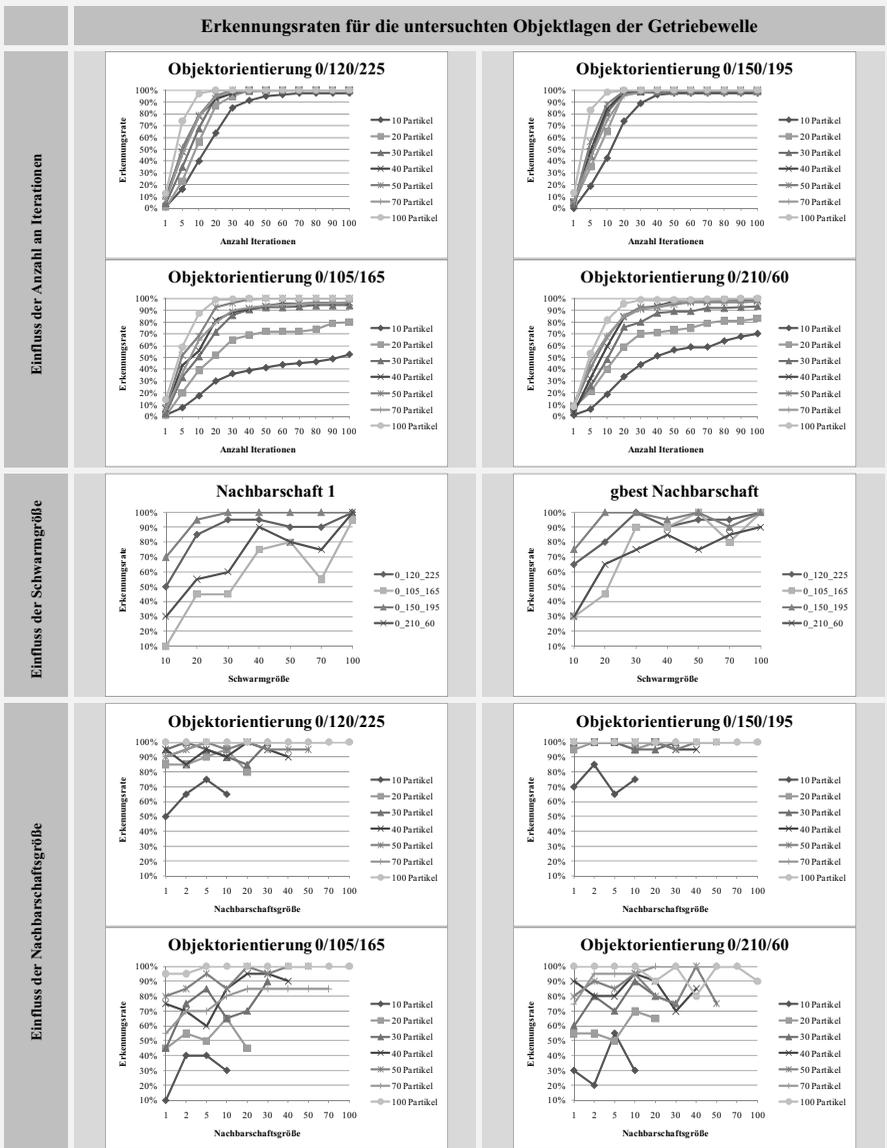


Abbildung 6-4: Einfluss der Anzahl an Iterationen, der Schwarmgröße und der Nachbarschaftsgröße auf die Erkennungsraten bei der Erkennung einer einzelnen Getriebewelle

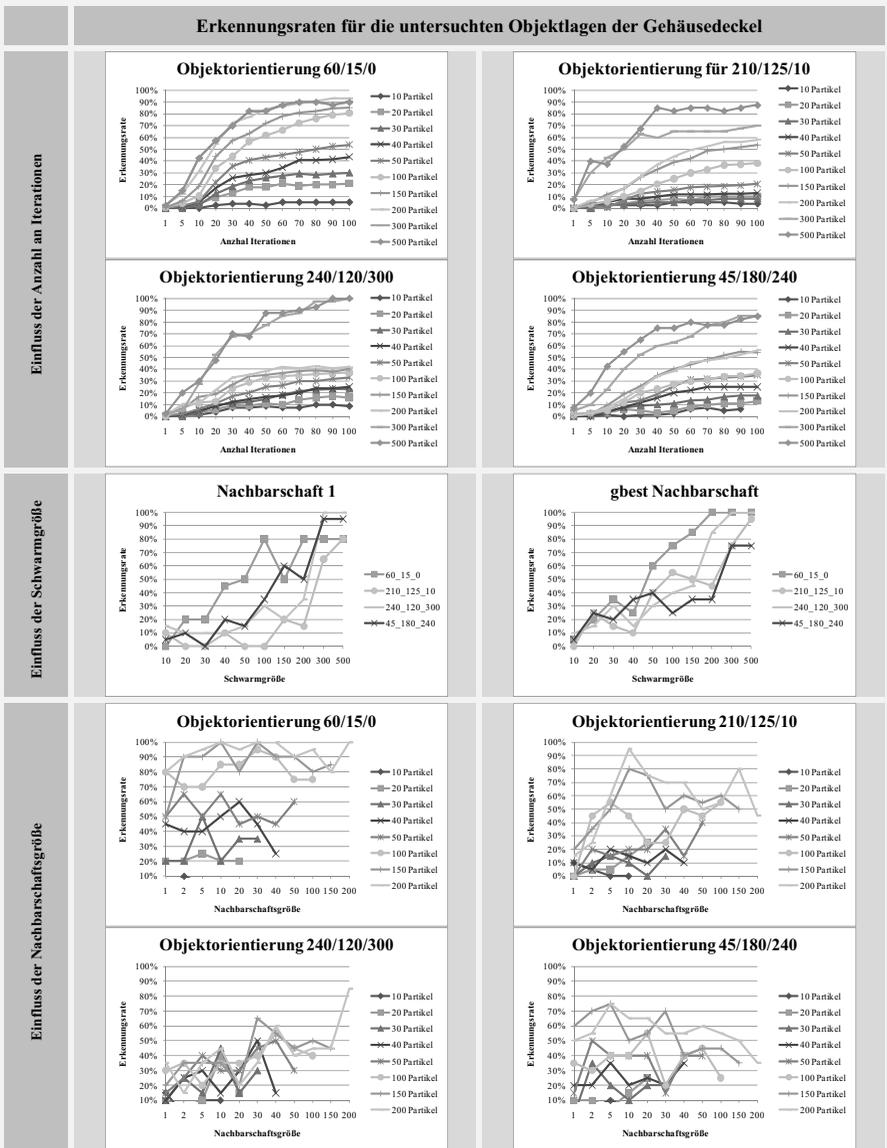


Abbildung 6-5: Einfluss der Anzahl an Iterationen, der Schwarmgröße und der Nachbarschaftsgröße auf die Erkennungsraten bei der Erkennung eines einzelnen Gehäusedeckels

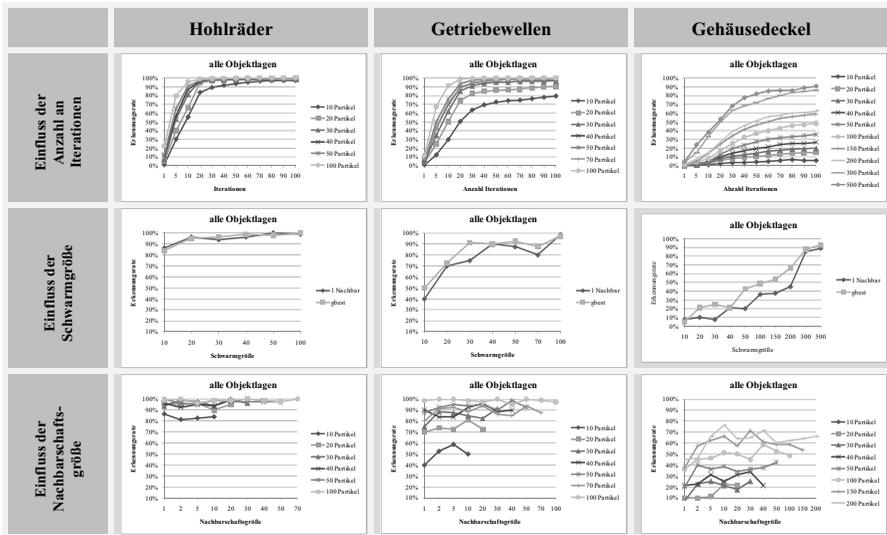


Abbildung 6-6: durchschnittliche Erkennungsraten für einzelne Hohlräder, Getriebewellen und Gehäusedeckel

6.3.4 Interpretation der Ergebnisse

Bei der Interpretation der Ergebnisse anhand von Abbildung 6-3 bis Abbildung 6-5 ist zu beachten, dass für die Darstellung der Erkennungsrate in Abhängigkeit von der Anzahl an Iterationen jeweils über alle Nachbarschaftsgrößen gemittelt wird. Für die Darstellung der Erkennungsrate in Abhängigkeit von der Schwarmgröße sowie für die Darstellung der Erkennungsrate in Abhängigkeit von der Nachbarschaftsgröße wurde jeweils die Erkennungsrate nach 20 Iterationen (Gehäusedeckel 100 Iterationen) angegeben.

Insgesamt zeigen die durchgeführten Tests mit dem Testobjekt Hohlrad, dass das entwickelte histogrammbasierte Verfahren zur Objektlageerkennung mit Partikel-Schwarm-Optimierung für alle Testlagen sehr gute Erkennungsraten erzielt. Die Erkennungsraten steigen mit zunehmender Anzahl an Partikel. Dies lässt den Schluss zu, dass die Fitnessfunktion für die zu lösende Aufgabe geeignet ist. Die stabile Konvergenz hin zur korrekten Objektlage bestätigt diese Interpretation. Für die meisten Objektlagen kann bei einer Schwarmgröße von 100 Partikeln in allen Durchläufen bereits nach 10 Iterationen die Objektlage bestimmt werden. Lediglich bei der Objektlage mit einem kleinen euklidischen Abstand zwischen dem globalen Optimum und den lokalen Optima, beispielsweise $R(\alpha, \beta, \gamma) = (120/0/0)$, sind weitere Iterationen nötig, um eine sichere Erkennung zu gewährleisten. Die sichere Erkennung kann dabei sogar erreicht werden, obwohl zur Darstellung der Ergebnisse über alle Nachbarschaftsgrößen gemittelt wurde. Für die

Wahl einer geeigneten Nachbarschaftsgröße lässt sich festhalten, dass für das Hohlrad die gbest-Nachbarschaft über alle Schwarmgrößen stabil hohe Erkennungsraten liefert. Allerdings sind für kleinere Schwärme von 10 Partikeln bereits ab einer Nachbarschaft von 5 Partikeln hohe Erkennungsraten vorhanden.

Insgesamt zeigen die Tests für das Hohlrad sehr gute Erkennungsraten bei einer geeigneten Wahl von Schwarmgröße und Nachbarschaftsgröße. Am geeignetsten unter Beachtung des Aufwands an benötigten Auswertungen der Fitnessfunktion zeigt sich die Kombination aus 100 Schwarmpartikeln mit 2 Nachbarn und 20 Iterationen.

Ähnlich sehr gute Erkennungsraten zeigen auch die Tests mit den Getriebewellen. Als aufwändigere Objektlagen fallen Lagen mit einem großen Einzugstrichter eines lokalen Minimums, beispielsweise $R(\alpha, \beta, \gamma) = (0/105/165)$, sowie Lagen mit einem kleinen Trichter des globalen Minimums, beispielsweise $R(\alpha, \beta, \gamma) = (0/210/60)$, auf. Für diese Lagen sind Schwärme ab einer Größe von 40 Partikeln geeignet, um Erkennungsraten größer 95% zu erzielen. Die Wahl einer größeren Nachbarschaft führt dabei nicht zu einer größeren Erkennungsrate. Ansonsten ist das Konvergenzverhalten ähnlich wie bei den Tests mit dem Hohlrad. Als geeignetste Kombination unter Beachtung des Aufwands an benötigten Auswertungen der Fitnessfunktion zeigt sich die Kombination aus 70 Schwarmpartikeln mit 5 Nachbarn und 20 Iterationen.

Die Tests anhand der definierten Objektlagen für die Gehäusedeckel bestätigen die Interpretation der Ergebnisse von Hohlrad und Getriebewelle. Die Schaubilder zeigen den gleichen charakteristischen Verlauf – allerdings auf einem niedrigeren Erkennungsniveau. Hohlrad und Getriebewelle weisen jeweils eine Symmetrieachse auf, die dazu führt, dass der Lösungsraum, der vom Schwarm bei der Suche betrachtet wird, eine Dimension weniger aufweist als der Lösungsraum des Gehäusedeckels. Dieser große Lösungsraum des Gehäusedeckels führt dazu, dass eine deutlich langsamere Konvergenz zur optimalen Lösung hin stattfindet. Eine hohe Erkennungsrate bedingt somit entweder einen größeren Schwarm oder eine höhere Anzahl an Iterationen. Beispielsweise kann mit einer Schwarmgröße von 500 Partikeln eine sehr gute Erkennungsrate von 95% erzielt werden.

6.4 Experimentelle Untersuchungen an synthetischen Daten einer Szene mit mehreren Objekten

6.4.1 Generierung der Testdaten mithilfe des modellierten Tiefensensors

Die Generierung der synthetischen Testdaten erfolgt in zwei Schritten. Die Objekte werden als CAD-Modelle mithilfe der CAD-Software SolidWorks [Solidwork 2008] innerhalb einer Kiste platziert. Hierbei werden die Orientierung und Position der CAD-Modelle frei gewählt. Die

gefüllte Kiste wird als STL-Daten abgespeichert und zusätzlich die Orientierung und Position der platzierten CAD-Modelle gespeichert. Mithilfe eines Tools, das ebenso wie das Tool zur Generierung der Wissensbasis auf Basis von VTK arbeitet, wird in einem zweiten Schritt aus diesen STL-Dateien mithilfe des modellierten Tiefensensors eine Punktwolke generiert. Pro Objekt werden vier Testkisten mit mehreren Objekten im Rahmen der Untersuchungen mit dem Verfahren zur PSO-basierten Objektlageerkennung getestet. Die Extraktion der Kiste liegt nicht im Fokus der Arbeit, deshalb wird für alle Szenen die gleiche Kiste genutzt, die automatisch vor dem Start der Partikel-Schwarm-Suche aus den Sensordaten ausgeblendet wird.

6.4.2 Auswahl der untersuchten Testszenen

Hauptziel der Tests mit synthetischen Sensordaten ist es, die Robustheit des entwickelten Partikel-Schwarm-Optimierungsverfahrens gegenüber inhomogenem Hintergrund zu untersuchen. Aus diesem Grund wurden für jedes Objekt mehrere Kisten so mit CAD-Modellen gefüllt, dass mehrere Objekte in der Kiste liegen, die den Hintergrund beeinflussen können. Abbildung 6-7 zeigt jeweils 4 untersuchte Beispielkisten der drei untersuchten Testobjekte.

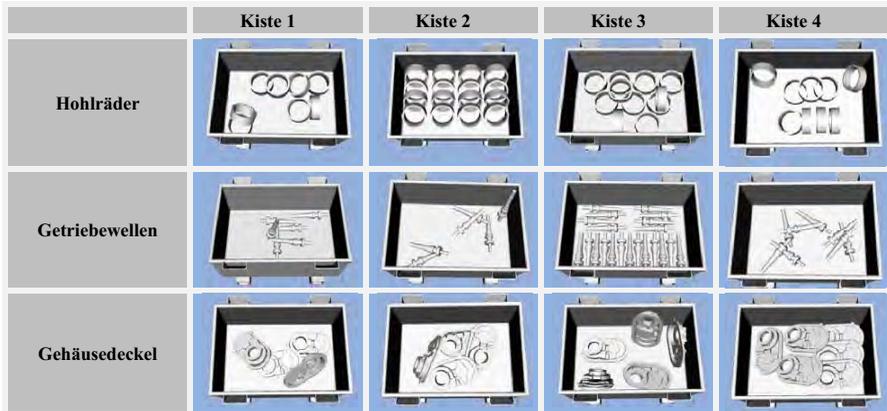


Abbildung 6-7 Beispiele synthetischer Sensordaten von Kisten mit den getesteten Objekten

6.4.3 Getestete Schwarmparameter und erzielte Ergebnisse

		Kiste 1	Kiste 2	Kiste 3	Kiste 4
Hohlräder	synth. Testscan				
	Ergebnis ¹⁾				
Getriebewellen	synth. Testscan				
	Ergebnis ¹⁾				
Gehäusedeckel	synth. Testscan				
	Ergebnis ¹⁾				

¹⁾ gefundene Objektlage wurde in Farbe mit weißer Umrahmung dargestellt

Abbildung 6-8: Darstellung der Testdaten als Parallelprojektion sowie das Ergebnis der jeweils gefundenen Objektlage

Zum Testen des Einflusses der Schwarmgröße, der Nachbarschaftsgröße und der Anzahl der Iterationen auf die Erkennungsrate wurden für alle drei Testobjekte mehrere Tests mit entsprechenden Parametern durchgeführt. Die Suche wurde dabei jeweils nach 20 Iterationen (Gehäusedeckel 100 Iterationen) beendet.

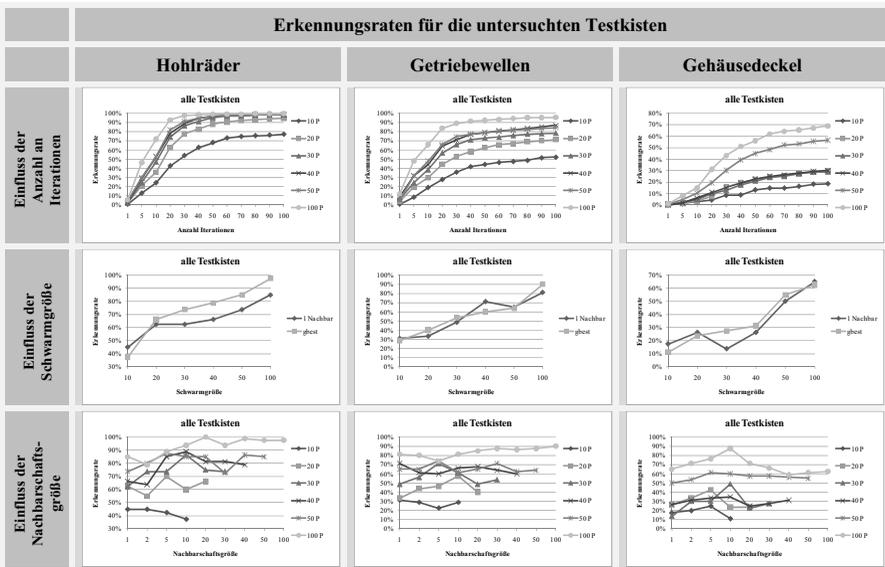


Abbildung 6-9: Einfluss der Iterationsanzahl, Schwarmgröße und Nachbarschaftsgröße auf die durchschnittliche Erkennungsraten für die Testkisten von Hohlrädern, Getriebewellen und Gehäusedeckeln

6.4.4 Zusammenfassung und Interpretation der Ergebnisse

Die Darstellung der Ergebnisse in Abbildung 6-9 erfolgt analog der Darstellung der Tests an den Einzelobjekten. Dies bedeutet, dass für die Darstellung der Erkennungsraten in Abhängigkeit von der Anzahl an Iterationen jeweils über alle Nachbarschaftsgrößen gemittelt ist und für die Darstellung der Erkennungsraten in Abhängigkeit von der Schwarmgröße sowie für die Darstellung der Erkennungsraten in Abhängigkeit von der Nachbarschaftsgröße jeweils die Erkennungsraten nach 20 Iterationen (Gehäusedeckel 100 Iterationen) dargestellt wird. Zusätzlich hierzu sind alle Erkennungsraten über alle Kisten für jedes Testobjekt nochmals gemittelt.

Die durchgeführten Tests mit synthetisch gefüllten Kisten bestätigen die sehr guten Erkennungsraten der zuvor durchgeführten Tests an den Einzelobjekten. Der im Gegensatz zu den Einzelobjekten nun variable Hintergrund durch dort liegende Objekte hat wie erwartet deutlichen Einfluss auf die Erkennungsraten. So ist beispielsweise für die Getriebewelle nun die Kombination aus einem Schwarm mit 100 Schwarmpartikeln, 30 Nachbarn und 100 Iterationen für sehr gute Erkennungsraten am geeignetsten.

Für die Testobjekte Hohlrad und Getriebewelle sind wieder sehr gute Erkennungsraten von knapp unter 100% bei 100 Iterationen für alle Testkisten erreichbar. Für 20 Iterationen, wie sie in den Darstellungen des Einflusses von Schwarmgröße bzw. Nachbarschaftsgröße zu sehen sind, sind für einen Schwarm mit 100 Schwarmpartikeln bereits sehr gute Ergebnisse von ca. 90% Erkennungsrate erzielbar.

Wie erwartet zeigen auch die Tests anhand der Kisten mit Gehäusedeckeln einen ähnlichen Verlauf wie bei den Einzelobjekten – ebenfalls auf einem deutlich niedrigeren Niveau der Erkennungsrate als für Hohlrad und Getriebewellen.

Insgesamt konnte durch die sehr guten Erkennungsraten der synthetisch gefüllten Kisten die Eignung des im Rahmen des letzten Kapitels entwickelten Verfahrens gezeigt werden. Trotz aller Aussagekraft, der Tests mit den synthetischen Daten – insbesondere durch die für die Auswertung hilfreiche Kenntnis der tatsächlich vorliegenden Objektlage – ist für den Praxiseinsatz des Verfahrens allerdings der erfolgreiche Test anhand von realen Sensordaten von besonderer Bedeutung, der anhand des Aufbaus einer entsprechenden Roboterzelle durchgeführt wird.

6.5 Aufbau einer Robotertestzelle zur experimentellen Untersuchung an realen Sensordaten

6.5.1 Zellenlayout

Abbildung 6-10 gibt eine Ansicht der Roboterzelle, die zu Testzwecken aufgebaut wurde, wieder.

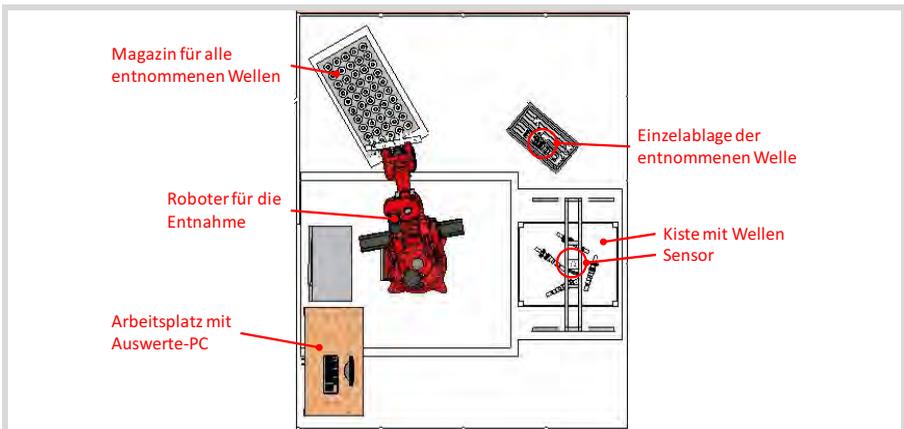


Abbildung 6-10: Layout der Roboterzelle zum Testen der Objektlageerkennung

6.5.2 Der eingesetzte Sensor

Der eingesetzte Sensor ist die Kombination aus einem 2-D-Scanmodul und einer Schwenkeinheit. Das Scanmodul ist das Laser-Messsystem LMS 400 der Sick AG. Als Schwenkeinheit wird der PowerCube PW 070 der Firma SCHUNK GmbH & Co. KG [Schunk 2005] verwendet. Der Scanner führt aufgrund der Schwenkeinheit eine Zentralprojektion durch und liefert nach einer Transformation in Kartesische Daten eine Punktwolke der betrachteten Kiste.

Der LMS 400 arbeitet mit einem Rotlicht-Laser, der aus einer Diode gespeist wird und mit Hilfe eines Spiegels entlang einer Linie wandert. Ein Vorteil dieses Systems ist, dass es sich um ein aktives System handelt, das keine zusätzliche Beleuchtung der Objekte benötigt.

Das Scanmodul liefert mit einer Scanfrequenz von bis zu 500 Hz den Abstand zu einer Vielzahl an Punkten der betrachteten Oberfläche. Diese Abstände errechnet der Sensor mit Hilfe der Phasenverschiebung zwischen ausgesandtem und empfangenem Strahl, die aus der Laufzeit des Lichtes und der verwendeten Wellenlänge resultieren. Eine detaillierte Beschreibung dieses Phasenshiftverfahrens kann beispielsweise [Jähne 2002] entnommen werden.

Der LMS 400 erreicht mithilfe dieses Messverfahrens folgende Werte [Sick 2005]:

- Arbeitsreichweite: 700 bis 3000 Millimeter
- Maximaler Öffnungswinkel: 70°
- Winkelauflösung: einstellbar von $0,1^\circ$ bis 1°
- Winkelfehler: $0,1^\circ$
- Systematischer Messfehler: 4 Millimeter
- Statistischer Messfehler: 3 bis 10 Millimeter

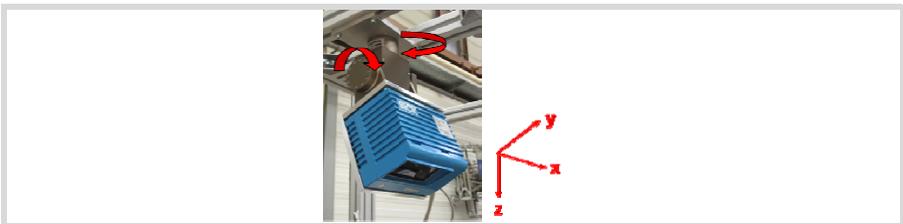


Abbildung 6-11: der eingesetzte Sensor als Kombination aus einem Sick LMS 400 und Schunk PowerCube PW 070

Zur Überprüfung der Praxistauglichkeit des entwickelten Verfahrens soll im Folgenden eine Robotertestzelle aufgebaut und experimentell untersucht werden. Hierfür wurde der Sensor in einer Höhe von 2600 mm statisch über der Kiste befestigt. Als Testobjekte wurden die

Getriebewellen gewählt. Die benötigten zusätzlichen Teilsysteme zur Greifpunktberechnung und zur Roboterbahnplanung, um eine Welle kollisionsfrei aus der Kiste zu entnehmen, wurden im Rahmen des Aufbaus der Roboterzelle mit den wichtigsten Funktionalitäten implementiert. Diese Teilsysteme liegen nicht im Fokus dieser Arbeit. Deshalb sollen sie im Weiteren nicht betrachtet werden.

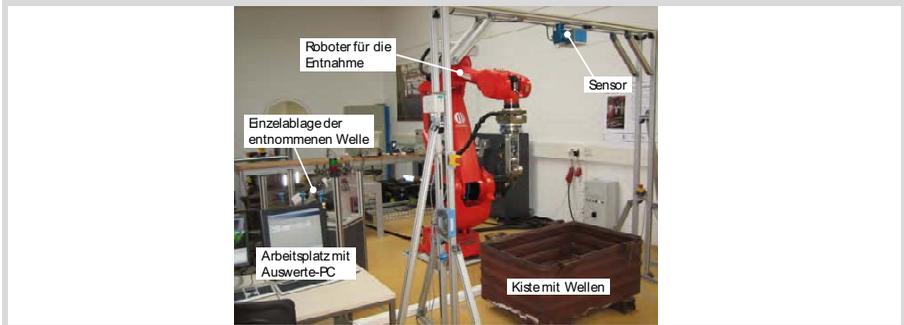


Abbildung 6-12: Foto der Roboterzelle

6.5.3 Gesamtablauf der Objekthandhabung

Als Szenario für den Gesamtablauf wurde ein typisches Szenario aus der Qualitätssicherung gewählt. In diesem Szenario wird nach der Entnahme einer Welle auf einer Einzelablage eine Qualitätssicherungsprüfung (z.B. Ultraschallprüfung) simuliert und anschließend das geprüfte Objekt für den nachfolgenden Prozess in ein Bereitstellungsmagazin abgelegt. Der Gesamtablauf ergibt sich wie in Abbildung 6-13 dargestellt. Hierbei sind keine Fehlerbehandlungen oder sonstige Sonderfunktionen wie manuelle Benutzereingriffe betrachtet, da dies keine Erkenntnisse für die Objektlageerkennung bietet. Außerdem wurde auch keine Betrachtung durchgeführt, welche Schritte parallel ausgeführt werden könnten, um die Zykluszeit zu verkürzen.

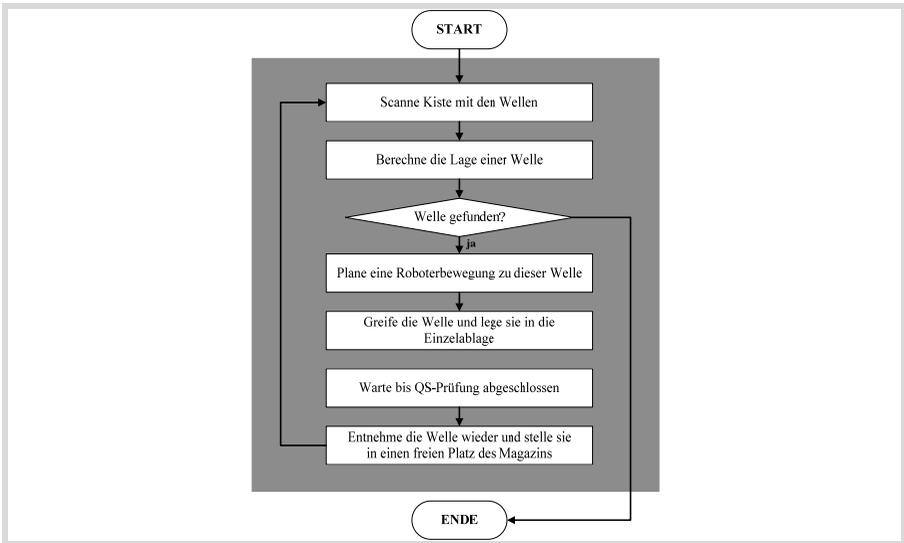


Abbildung 6-13: Gesamtablauf eines Entnahmezyklus

6.5.4 Erzielte Ergebnisse

Mehrere Entnahmezyklen mit der aufgebauten Roboterzelle haben gezeigt, dass das System zur PSO-basierten Objektlageerkennung auch unter praxisrelevanten Bedingungen funktioniert. Als Parameter wurden dabei aufgrund der Ergebnisse der Tests mit synthetischen Sensordaten 100 Partikel mit 30 Nachbarn und 100 Iterationen gewählt. Typischerweise dauerte eine Schwarmssuche ca. 4 Sekunden, bis das Ergebnis vorliegt. Die Tests mit der Roboterzelle zeigten, dass es mit diesen Einstellungen beispielsweise bei einer Kiste mit 50 Teilen zu vier falsch erkannten Objektlagen kam, was einer Erkennungsrate von 92% entspricht und somit in dem aufgrund der vorherigen Tests erwarteten Bereich liegt. Eine Lage wurde in diesem Fall als korrekt erkannt bezeichnet, wenn die Welle mit der vorhandenen Wissensbasis erkannt und mit dem eingesetzten Greifer erfolgreich mit dem Roboter aus der Kiste entnommen werden konnte. Teilweise erfolgte dabei die Roboterbewegung nicht vollautomatisch, da gerade im Kistenrandbereich die rudimentär umgesetzte kollisionsfreie Greifpunktberechnung keine ausreichenden Ergebnisse lieferte. Die folgende Abbildung 6-14 zeigt beispielhaft vier Aufnahmen erfolgreicher Greifvorgänge.

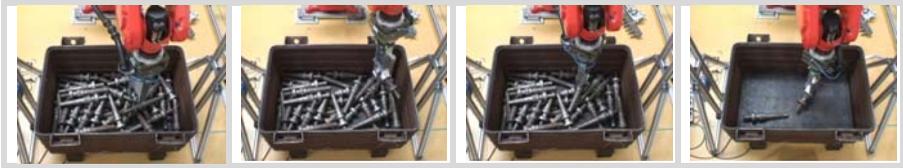


Abbildung 6-14: Beispiele von erkannten und gegriffenen Objekten

6.6 Fazit aus den durchgeführten Tests

Die Versuchsergebnisse der PSO-basierten Objektlageerkennung sowie der Robotertestzelle lassen folgende Schlüsse zu:

Anhand synthetischer Testdaten von Einzelobjekten und gefüllten Kisten konnten für die untersuchten Testobjekte Hohlrad, Getriebewelle und Gehäusedeckel geeignete Schwarm-Parameter ermittelt werden. Mit diesen Parametern konnten in den Tests Erkennungsraten von bis zu 100% erreicht werden. Das entwickelte System zur PSO-basierten Objektlageerkennung mit Histogrammen erzielt auch mit realen Sensordaten eine Erkennungsrate von 92%. Bei falsch erkannter Objektlage konnte in den meisten Fällen durch nochmaliges Scannen und Auswerten die korrekte Objektlage, die ein Greifen ermöglichte, erreicht werden. Für die durchgeführten Tests war das Ziel, immer ein oben liegendes Objekt zu erkennen. Während der Tests zeigte sich jedoch, dass selbst bei einem oben liegenden Objekt komplexe Greifsituationen entstehen können, die die Definition zusätzlicher Greifmöglichkeiten verlangte. Zukünftige Anlagen sollten deshalb die Möglichkeit haben, mehr als ein Objekt zu erkennen, um die Anzahl an benötigten Greifmöglichkeiten gering zu halten. Für die in der Roboterzelle untersuchten Getriebewellen ist die erzielte Auswertzeit ausreichend. Um auch in Anwendungen einsetzbar zu sein, die deutlich kürze Taktzeiten erfordern, sollten zukünftig weitere Maßnahmen ergriffen werden. Eine Möglichkeit ist beispielsweise, die Parallelisierung der Auswertung. Gerade die Partikel-Schwarm-Optimierung ist aufgrund ihres Konzept dazu geeignet, beispielsweise durch die Portierung auf die Prozessoren der Grafikkarte (englisch: Graphics Processing Unit - GPU) die hohe Anzahl an Prozessoren zur Auswertung zu nutzen.

Insgesamt zeigen die erzielten Ergebnisse mit der Roboterzelle, dass das entwickelte Erkennungssystem die gestellten Anforderungen erfüllt. Abbildung 6-15 gibt eine Übersicht der in Kapitel 1 erarbeiteten Anforderungen sowie eine Begründung für deren Erfüllung.

Anforderung	Erfüllung	Begründung
Anforderungen an die Objektbeschreibung		
(A1)	✓	Durch die Entwicklung eines Werkzeugs, mit dem automatisch die benötigten Daten für die Wissensbasis erstellt werden können, ist das Erkennungssystem an unterschiedliche Objekte anpassbar.
(A2)	✓	Das Werkzeug zur Erstellung der Daten für die Wissensbasis arbeitet mit 3-D-CAD-Daten.
(A6)	✓	Das Erkennungssystem arbeitet direkt auf den Tiefendaten des Sensors.
(A7)	✓	Die für die Erkennung benötigten Daten können automatisch aus den CAD-Daten gewonnen werden.
(A10)	✓	Symmetrien eines Objekts können durch die Anpassung des betrachteten Rotationsbereichs in der Wissensbasis abgebildet werden.
(A11)	✓	Durch die Flächenbetrachtung bei der Lageerkennung fallen kleine Abweichungen in der Objektgeometrie nicht ins Gewicht.
Anforderungen an den Partikel-Schwarm		
(A12)	✓	Die zur Modellierung einer Objektlage gewählten Parameter bilden einen zusammenhängenden Lösungsraum.
(A14)	✓	Das Modell eines Partikels im Schwarm berücksichtigt den Zusammenhang zwischen der aktuell untersuchten Objektlage und den zugehörigen Tiefenwerten.
Anforderungen an die Datenvorverarbeitung		
(A4)	✓	Tests mit realen Sensordaten haben gezeigt, dass das Erkennungssystem robust gegenüber Rauschen in den Sensordaten ist.
(A5)	✓	Durch die Auswertung der Tiefendaten und durch Skalierbarkeit der Wissensbasis ist das Erkennungssystem an unterschiedliche Tiefensensoren anpassbar.
Anforderungen an die Segmentierung		
(A3)	✓	Die durchgeführten Untersuchungen haben gezeigt, dass selbst im ungeordneten Haufwerk ein oben liegendes Objekt lokalisiert werden kann.
(A13)	✓	Die Segmentierung arbeitet Blob-basiert und ist damit unabhängig von der aktuell untersuchten Objektlage.
Anforderungen an das Gesamtsystem		
(A8)	✓	Während der Tests mit der Roboterzelle konnte die geforderte Genauigkeit erreicht werden.
(A9)	✓	Die Auswertung während der Tests mit der Roboterzelle dauerte etwa 4 Sekunden.

Abbildung 6-15: Überprüfung der Anforderungen an das Gesamtsystem zur Objektlageerkennung

7 Zusammenfassung und Ausblick

Roboter, die Werkstücke von einem Förderband greifen, gehören in vielen Produktionsanlagen inzwischen zum gewohnten Bild. Ermöglicht wurde dies durch die rasanten Entwicklungen in der Bildverarbeitung, um die Lage von Werkstücken auf dem Förderband zu bestimmen. Nach Einschätzungen von Experten ist die Verwendung von 3-D-Daten zur Bestimmung der Lage der Werkstücke der nächste logische Entwicklungsschritt. In den letzten Jahren wurde eine Vielzahl an zuverlässigen 3-D-Sensoren entwickelt, die solche 3-D-Daten liefern. Anders als bei Kameras, die Intensitäts- oder Videobilder zur Verfügung stellen, können mit Hilfe der 3-D-Sensoren direkte geometrische Informationen einer Szene ausgewertet werden. Damit sollen auch komplexe Situationen wie beispielsweise das 3-D-Bin-Picking, die Entnahme von Werkstücken mit einem Roboter aus einer Kiste, aufgelöst werden.

Um dies leisten zu können, müssen Roboter mit nicht vorhersagbaren Objektlagen und gegenseitigen Überdeckungen vor variablem Hintergrund zurechtkommen. Roboter müssen ihr Wissen über die Objekte nutzen, um eigenständig eine kollisionsfreie Entnahme zu ermöglichen. Die zuverlässige Objektlageerkennung innerhalb der Kiste gilt hierbei als Schlüsselfunktion für das 3-D-Bin-Picking.

Aktuelle Ansätze zur Objektlageerkennung, die für den Einsatz im Produktionsumfeld entwickelt werden, arbeiten derzeit verstärkt auf einer Kombination von 2-D-Daten mit lokal gemessenen 3-D-Daten, um die Lage eines Objekts zu erhalten. Um flexibel an unterschiedliche Objekte anpassbar zu sein, legen gerade neue Ansätze ihr Augenmerk auf den Einsatz einer Wissensbasis zur Objektbeschreibung. Die Bestimmung der Objektlage erfolgt dabei durch die Suche nach der besten Übereinstimmung zwischen einer Objektlage innerhalb der Wissensbasis mit den 3-D-Daten. Für eine schnelle Suche bieten Optimierungsverfahren geeignete Möglichkeiten.

Ziel der vorliegenden Arbeit war es daher, Lösungsansätze und Konzepte für den Einsatz des relativ jungen Forschungsgebiets der Partikel-Schwarm-Optimierung zur Objektlageerkennung zu schaffen, die es Robotern ermöglichen, Objekte in einer Kiste zu lokalisieren und zu entnehmen. Hierzu wurde die Partikel-Schwarm-Optimierung eingeführt, die wichtigsten Eigenschaften vorgestellt und für den Einsatz in einem System zur Objektlageerkennung nutzbar gemacht. Das System zur Objektlageerkennung soll dabei Anforderungen bezüglich eines Einsatzes im Gießerei- und Schmiedeumfelds genügen. Neben den theoretischen Grundlagen wie die Modellierung eines Schwarmpartikels und die Entwicklung eines geeigneten wissensbasierten Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels sollte die technische Machbarkeit anhand einer Roboterzelle nachgewiesen und das Verfahren verifiziert werden.

Ausgehend vom Stand der Technik wurden typische Einsatzszenarien für die Entnahme von Objekten aus einer Kiste im Gießerei- und Schmiedebereich hinsichtlich ihrer Randbedingungen an ein System zur Objektlageerkennung analysiert. Als Objekte dienten hierbei Getriebewellen, Hohlringe und Getriebedeckel. Besonderes Augenmerk wurde dabei auf die Genauigkeitsanforderungen bei der Lageerkennung und die Anforderungen an die Objektbeschreibung für den Einsatz mit der Partikel-Schwarm-Optimierung gelegt. Diese Analyse lieferte die Anforderungen an ein PSO-basiertes System zur Objektlageerkennung. Weiterhin wurden Teilsysteme definiert und der Einfluss der Anforderungen an die Teilsysteme diskutiert.

Auf Basis der erarbeiteten Anforderungen wurden grundlegende Lösungsansätze für die einzelnen Teilsysteme betrachtet und zu einer Gesamtkonzeptlösung zusammengeführt. Vorrangig erschien hier die grundlegende Untersuchung eines wissensbasierten Verfahrens zur histogrammbasierten Bewertung eines Schwarmpartikels. Hierfür wurde der 3-D-Sensor modelliert und die geeignete Erstellung von Tiefen- und Normalenvektorhistogrammen erarbeitet. Um das Gesamtsystem an unterschiedliche Objekte anpassen zu können, wurde ein Tool zur Erstellung der benötigten Histogramme für die Wissensbasis erstellt.

Für eine Abschätzung, ob das jeweilige Objekt durch die erstellte Wissensbasis gut beschrieben wird, d.h. ob eine Objektorientierung eindeutig identifiziert und gleichzeitig gegenüber anderen Objektorientierungen abgegrenzt werden kann, wurde eine Möglichkeit entwickelt, die Wissensbasis auf Basis des zugehörigen Lösungsraums zu bewerten und Objektlagen auszuwählen, die eine schwierige Erkennung erwarten lassen. Für die drei in der Analyse betrachteten Objekte wurden anschließend umfangreiche Testläufe mit unterschiedlichen Parametereinstellungen für das PSO-System durchgeführt. Als Testszenarien wurden hierbei Einzelobjekte, synthetisch erstellte Szenen mit mehreren Objekten und reale mit einem Tiefensensor erstellte Szenen untersucht. Die Ergebnisse zeigen, dass das Gesamtsystem robust funktioniert und für alle drei Objekte mit mehreren Parametereinstellungen für das PSO-System gute Lageerkennungsergebnisse erzielt werden können.

Um die Praxistauglichkeit der Entwicklungen zu bestätigen, wurde zum Abschluss dieser Tests eine Roboterzelle mit einem Tiefensensor aufgebaut und das Verfahren zur Objektlageerkennung hinsichtlich seiner Zuverlässigkeit untersucht. Als Testobjekte kamen die Getriebewellen zum Einsatz, die aus einer Kiste entnommen und auf einer Prismaablage positioniert wurden.

Das PSO-basierte System zur Objektlageerkennung in Tiefendaten auf Basis von Histogrammen ist in der Lage, flexibel an unterschiedliche Objekte angepasst zu werden und eignet sich somit für den Einsatz in unterschiedlichen Aufgaben der praktischen Anwendung. Das Konzept zur PSO-basierten Objektlageerkennung wurde so erstellt, dass die Fitnessfunktion einfach erweitert

oder ausgetauscht werden kann und somit alle Verfahren als Fitnessfunktion anwendbar sind, die ausgehend von einem Bezugspunkt – beispielsweise der höchste Punkt eines Objekts – definierte Tiefenwerte einbezieht, um die Fitness zu berechnen. Zukünftige Entwicklungen, die beispielsweise die Erkennungsleistung noch weiter erhöhen, können somit leicht in das im Rahmen dieser Arbeit untersuchte Erkennungssystem integriert werden.

Im Rahmen des Aufbaus der Roboterzelle wurden die zur Entnahme benötigten Greifpunktberechnungen und Bahnplanungen mit Kollisionsvermeidung vereinfacht implementiert, da sie und auch der nachgelagerte Prozessschritt nicht im Fokus der Arbeit lagen. Um das entwickelte System zur Objektlageerkennung in reale Anwendungen in der Produktion einzusetzen, sind hier zukünftig noch umfangreiche Untersuchungen und Entwicklungsarbeiten zu leisten, um die notwendige Robustheit des Gesamtsystems zu erreichen.

8 Abstract

Robots which grasp parts from a conveyor belt are a well known sight in many production lines. This was made possible by rapid advances in camera based image processing to determine the position of work pieces on the conveyor belt. Experts agree that the use of 3-D data to get the position of objects will be the next logical step. Especially in complex situations the use of direct geometric information of a scene instead of intensity or video images is the advantage of 3-D sensors over standard cameras. Such complex situations are typical for 3-D Bin-picking – grasping of objects with a robot out of a bin.

In other words robots must cope with unpredictable situations and mutual overlapping objects in front of a variable background. To generate a collision-free movement and grasping procedure independently robots must use their knowledge of the objects. The reliable object localization within the box is known as a key function here.

Current applications for object localization that are developed for production environment often work with a combination of 2-D data and locally measured 3-D data to localize an object. A deeper look in recent approaches shows that most of the object localization algorithms use knowledge bases for object descriptions. The knowledge base can be flexibly adapted to different objects if necessary. The determination of the object location is done by searching for the best match between many assumed object locations in the 3-D data and the knowledge base. For a quick search optimization methods provide suitable opportunities.

The aim of this thesis was therefore to create solutions and concepts by the use of Particle Swarm Optimization that will enable robots to locate and to grasp objects out of a bin. Particle Swarm Optimization is a relatively young research field for a population based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling. Starting with a short introduction to particle swarm optimization the most important properties were presented and possibilities for the use in a system for object localization were shown. Theoretical work such as modeling single swarm particles and developing a knowledge-based method suitable for histogram-based evaluation of a particle swarm was performed. In this context the goal of the thesis was to get a system for object localization that meets the requirements of applications in foundry and forging industry. For this reason the technical feasibility of the entire system should be demonstrated by a robot cell.

Starting with the state of the art, typical application scenarios in the foundry and forging industry for grasping objects out of a bin were analyzed and typical requirements and boundary conditions for an object localization system were derived. Gear shafts, inner gears and housing covers served as objects in these application scenarios. Particular attention was paid to the accuracy

requirements of the object localization module and the demands on the object description for use with the particle swarm optimization. This analysis provided the requirements of a PSO-based system for object localization. Furthermore, subsystems were defined and the impact of the requirements on the subsystems was discussed.

Based on these requirements different possible solutions for the subsystems were analyzed. By combining selected solutions for each subsystem the entire system for object localization was designed. In order to realize the designed system a fundamental study of a knowledge-based method for evaluating a swarm particle with histograms was performed. For this purpose an applicable model of the 3-D sensor and appropriate histogram creation methods for depth- and normal vector histograms were developed. Furthermore, a tool was created to generate and store required histograms in a knowledge base. This is necessary to adjust the system for object localization to various objects by expanding the knowledge base.

It is very important that an object is well described in the knowledge base. In this context well described means that an object orientation can be clearly identified and distinguished from other object orientations. For an evaluation of whether an object is well described in the knowledge base a way was proposed to assess the knowledge base on the basis of the corresponding solution space.

The performance of the system for object localization was tested with the three objects considered in the analysis. The tests were conducted with different parameter settings for the PSO system especially in concerns of number of iterations, swarm size and neighborhood structure. To simulate different complexity levels several test scenarios were defined: single objects, synthetically generated scenes with multiple objects and scenes that are scanned with a real depth sensor. The results show that the overall system is robust and reliable and can be obtained for all three objects. For all three objects the PSO-system showed good results with multiple parameter settings.

Furthermore, to confirm the practical feasibility of the developments and to complete the tests, a robot cell was built up. Besides the robot itself the cell consisted of a depth sensor, a gripper, a box with the objects and a storage position. The robot cell was used to show the reliability of the system for object localization. As test objects, gear shafts were used which were taken from a box and placed on the storage position.

The results of this work can be a basis for future work in different areas. At first, the results can be used more-or-less directly within a real application for the tested gear shafts, where the gear shafts should be placed on a prismatic storage with an ultrasonic detection system.

Furthermore, the PSO-based system for object localization in depth data on the basis of histograms can be flexibly adapted to different objects and is therefore suitable for use in various practical applications.

In addition, the concept of PSO-based object position detection has been created in a way that the fitness function can be easily extended or replaced. Therefore, all methods can be used as a fitness function, which uses a reference point to define the depth values and calculate the fitness of a single swarm particle. A reference point can be for instance the highest point of an object. Future developments to increase the recognition performance even further, can thus easily be integrated within the framework.

And finally, the gripping point calculation and robot path planning with collision avoidance were implemented in a simple way since they and the following tasks lay not in the focus of this thesis. To adapt the system for object localization to real-world applications in production applications, additional research and development works are to be made in order to achieve the necessary robustness of the overall bin-picking-system.

9 Literaturverzeichnis

- Angline 1998 Angline, P.: Evolutionary Optimization versus Particle Swarm Optimization: Philosophy and Performance Difference. In: Proceedings of the Seventh Annual Conference on Evolutionary Programming, pp. 601-610, 1998.
- Bahnu 1984 Bahnu, B.; Faugeras, O. D.: Shape Matching of Two-Dimensional Objects. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No. 2, pp. 137-155, 1984.
- Baraquet 1997 Baraquet, G.; Sharir, M.: Partial Surface and Volume Matching in Three Dimensions. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 9, pp. 1-21, September 1997.
- Bebis 2002 Bebis, G.; Louis, S.; Varol, Y.; Yfantis, A.: Genetic Object Recognition Using Combinations of Views. In: Proceedings of IEEE Transactions on Evolutionary Computation, Vol. 6, No. 2, pp. 132-146, April 2002.
- Belforte 1986 Belforte, G.; Alfio, D. N.; Romiti, A.: Brake Drums Bin-Picking. In: Proceedings of the 15th International Symposium on Automotive Technology and Automation ISATA 1986, Vol. 1, pp. 1-15, Oktober 1986.
- Bellman 2003 Bellman, R.: Dynamic Programming, Dover: Dover Publications Inc., 2003.
- Benes 2003 Benes, J.: American Machinist: Robots that see and feel (2003) [http://www.americanmachinist.com/304/Issue/Article/False/8329/Issue\(07.08.2008\)](http://www.americanmachinist.com/304/Issue/Article/False/8329/Issue(07.08.2008)).
- Bennamoun 2002 Bennamoun, M.; Mamic, G. J.: Object recognition: Fundamentals and case studies. London, Berlin, Heidelberg [u.a.]: Springer, 2002.
- Benneker 2004 Benneker, H.: Massivumformung in Deutschland: Daten, Fakten und Trends. In: Schmiede-Journal, März 2004, S. 13-16.
- Berger 2000 Berger, M.; Bachler, G.; Scherer, S.: Vision Guided Bin Picking and Mounting in a Flexible Assembly Cell. In: Proceedings of the 13th International Conference on Industrial and Engineering Applications of

Artificial Intelligence and Expert Systems (IEA/AIE 2000), New Orleans, Louisiana, USA, June 19 - 22, 2000, pp. 109-118.

- Berger 2001 Berger, M.: Integrating Appearance Based and Model Based Pose Determination. Dissertation, Universität Graz, Januar 2001.
- Beyer 1996 Beyer, U.; Smieja, F.: A model-based approach to recognition and measurement of partially hidden objects in complex scenes – JANUS Project, Sep 20, 1996, REFLEX Report No. 1996/3.
- Biedermann 1987 Biedermann, I.: Recognition-by-components: a theory of human image understanding. In: Psychological Review, Vol. 94, No. 2, pp. 115-147, 1987.
- Binford 1971 Binford, T. O.: Visual perception by computer. In: Proc. IEEE Conference on Systems and Controls (Miami, Florida, December), pp. 292-301, 1971.
- Birk 1981 Birk, J. R.; Kelley, R. B.; Martin, H. A. S.: An Orienting Robot for Feeding Workpieces Stored in Bins. In: IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-11, No. 2, Februar 1981.
- Böhm 2005 Böhm, J.: Modellbasierte Segmentierung und Objekterkennung aus Distanzbildern. In: Schriftenreihe Deutsche Geodätische Kommission bei der Bayerischen Akademie der Wissenschaften: Reihe C, Dissertationen, München, 2005.
- Boughorbel 2003 Boughorbel, F.; Zhang, Y.; Kang, S.; Chidambaram, U.; Abidi, B.; Koschan, A.; Abidi, M.: Laser Ranging and Video Imaging for bin picking. In: Assembly Automation, Vol. 23, pp. 53-59, March 2003.
- Breuckmann 2008 Breuckmann AG, Metronor AS: naviSCAN 3D Scannen und Prüfen ohne Grenzen. Prospekt Mai 2008.
- Brodtmann 2008 Brodtmann, T.: Branche erwartet für 2008 Rekordergebnis, VDMA Pressemitteilung (München, 4. Juni 2008), VDMA Robotik und Automation.
- Bunke 2007 Bunke, J.: Griff in die Kiste (28.03.2007). Online. E-Mail von joerg.bunke@thyssenkrupp.com (28.03.2007).

- Carlisle 2001 Carlisle, A.; Dozier, G.: An Off-The-Shelf PSO: In: Proceedings of the 2001 Workshop on Particle Swarm Optimization, pp. 1-6, Indianapolis, IN.
- Chen 1988 Chen, H. H.; Huand, T. S.: A survey of construction and manipulation of octrees. In: Computer Vision Graphics Image Processing, Vol. 43, pp. 409-431, 1988.
- Colomi 1992 Colomi, A.; Dorigo, M.; Maniezzo, V.: An Investigation of Some Properties of an Ant Algorithm. In: Proceedings of the Parallel Problem Solving from Nature Conference (PPSN 92). Brussels, Männer, R. & Manderick, B.(Eds.), Elsevier Publishing, Amsterdam, pp. 509-520.
- Cong 2008 Cong, L.; Hu, C.; Guo, Z.; Jiang, Y.; Sha, L.: Improvement of original particle swarm optimization algorithm based on simulated annealing algorithm. In: 27th Chinese Control Conference (CCC 2008). Kunming, China, 16-18 July 2008, pp. 671-676.
- Courtney 2003 Courtney, P.; Böttcher, P.: ECVision White paper on industrial applications of cognitive vision. European Research Network for Cognitive Computer Vision Systems. April 2003.
- DGV 2007 Deutscher Gießereiverband: DGV-Wirtschafts- und Branchendaten für Deutschland und Europa (2007) <http://www.dgv.de/wirtdat.htm> (14.08.2008).
- Dong 2006 Dong, C.; Qui, Z.: Particle Swarm Optimization Algorithm Based on the Idea of Simulated Annealing. In: International Journal of Computer Science and Network Security, Vol. 6, No. 10, October 2006, pp.152-157.
- Dorigo 2004 Dorigo, M.; Stützle, T.: Ant colony optimization. Cambridge, Mass., MIT Press, 2004.
- EADS 2004 EADS Astrium: Griff in die Kiste (2004). http://cs.astrium.eads.net/3da/mvs_griffindiekiste.html (25.08.2004).
- Eberhart 1995 Eberhart, R.; Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: Proc. Sixth International Symposium on Micro Machine

- and Human Science (Nagoya, Japan) 1995, IEEE Service Center, Piscataway, NJ, pp. 39-43, 1995.
- Eberhart 1998 Eberhart, R. C.; Shi, Y.: Evolving Artificial Neural Networks. In: Proceedings of the International Conference on Neural networks and Brain, Beijing, China, pp. 5-13, 1998.
- Elektronikpraxis 2006 Elektronikpraxis: Übersicht Bildverarbeitung (2006)
<http://www.elektrotechnik.vogel.de/filesserver/vogelonline/files/38.pdf>
(07.08.2008).
- Ettel 2002 Ettelt, E. M.: Ansichtsbasierte Objekterkennung mit Hilfe optimierter Musterbäume. Dissertation, München, Techn. Univ., 2002.
- Fanuc 2005 Schutzrecht EP 1 589 483 A2 (26.20.2005): Object picking system. FANUC Ltd. Pr.: JP 2004128171, 23.04.2004.
- Fanuc 2008 FANUC Ltd: FANUC Integrated Vision Function iRvision (2008)
<http://www.fanuc.co.jp/en/product/robot/intelligence/irvision.html>
(09.09.2008).
- Faugeras 1993 Faugeras, O.: Three-Dimensional Computer Vision, A Geometric Viewpoint. Cambridge, Mass., MIT Press, 1993.
- Festinger 1954 Festinger, L.: A theory of social comparison processes. In: Human Relations, 1954, 7(2), pp. 117-140.
- Fili 2008 Fili, W.: Die Zukunft der Robot-Vision ist mehrdimensional.
(18.02.2008)
<http://www.maschinenmarkt.vogel.de/index.cfm?pid=1496&pk=109516>
(05.08.2008).
- Fogel 2004 Fogel, D. B.; Michalewicz, Z.: How to Solve It, Modern Heuristics. Berlin: Springer, 3. Auflage, 2004.
- Foitzik 2004 Foitzik, B.: Robot-Vision: Zukunftsträchtige Techniksynergie zeigt zuweilen Schwächen. In: Industrieanzeiger 23, S. 36-38, 2004.
- Fritsch 2006 Fritsch, D.: Materialflusssimulationen als effizientes Werkzeug zur Planung und Realisierung von Automatisierungslösungen. In: Schraft, Rolf Dieter (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und

Automatisierung IPA: Effiziente Planung und Entwicklung von Automatisierungslösungen. Fraunhofer IPA Workshop F 129, 02. Mai 2006, Stuttgart. Stuttgart: FpF-Verein zur Förderung produktionstechnischer Forschung, S. 40-54, 2006.

- Fritsch 2006a Fritsch, D.; Wöltje, K.: Roboter in der Intralogistik. In: wt Werkstattstechnik online 96, Nr. 9, S. 623-630, 2006.
- Fritsch 2007 Fritsch, D. et al.: Abschlussbericht zum Projekt Branchenunabhängige Basismodule für ein Kommissionierrobotersystem "KomRob". Stuttgart: Fraunhofer IPA, 2007.
- Geiges 1997 Geiges, H.: Mannigfaltige Geometrien. Elemente der Mathematik 52 (1997), Basel: Birkhäuser Verlag, pp. 93-107, 1997.
- Ghita 2003 Ghita, O.; Whelan, P. F.: A bin picking system based on depth from defocus. In: Machine Vision and Applications archive, Vol. 13, Issue 4, pp. 234-244, February 2003.
- Glover 1986 Glover, F.: Future paths for integer programming and links to artificial intelligence. In: Comput. Oper. Res. 13 (1986), No. 5, pp. 533-549. URL [http://dx.doi.org/10.1016/0305-0548\(86\)90048-1](http://dx.doi.org/10.1016/0305-0548(86)90048-1).
- Grewe 1995 Grewe, L.; Kak, A. C.: Interactive Learning of a Multiple-Attribute Hash Table Classifier for Fast Object Recognition. In: Computer Vision and Image Understanding (CVIU), Vol. 61(3), pp. 387-416, May 1995.
- Gunnarson 2007 Gunnarson, K.: 3D Vision, Robots and Movement – Putting it all Together. In: Robotic Industries Association et al.: International Robots & Vision Show / CD-ROM : Technical Conference Proceedings. June 12-14, 2007, Rosemont (Chicago), Illinois, USA, 2007.
- Hardin 2008 Hardin, W.: Quick-Pick, Robotic Bin-picking workcell uses FireWire and model-based analysis to unload water-pump parts. In: Vision Systems Design, pp. 25-31, September 2008.
- Hardin 2009 Hardin, R. H.; Sloane, N. J. A.; Smith, W. D.: Tables of Spherical Codes with Icosahedral Symmetry. (2009) <http://www2.research.att.com/~njas/icosahedral.codes/index.html> (13.09.2009).

- Heidemann 1996 Heidemann, G.; Ritter, H.: A Neural 3-D Object Recognition Architecture Using Optimized Gabor Filters. In: Proceedings of 13th International Conference on Pattern Recognition, Vienna, Jgg. IV, pp. 70-74. IEEE Computer Society Press, 1996.
- Heintz 2007 Heintz, R.: Neues Verfahren zur invarianten Objekterkennung und -lokalisierung auf der Basis lokaler Merkmale. Dissertation, Universität Karlsruhe, 2007.
- Herbertz 2005 Herbertz, R.; Hermann, H.; Ledermann, T. u.a.: Entwicklung und Erprobung eines modularen Handhabungskonzepts für die umformende Fertigung von Schüttgütern. In: Schmiede-Journal, S. 28-29, September 2005.
- Hetzel 2001 Hetzel, G.; Leibe, B.; Levi, P.; Schiele, B.: 3D Object Recognition from Range Images using Local Feature Histograms. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01). Kauai Island, Hawaii, Vol. 2, pp. 394-399, December 2001.
- Holst 2004 Holst, M.: Geometriebeschreibung mit punktbasierten Modellen, Ersetzungsstrategien und Rendering. Diplomarbeit, Universität Rostock, Fakultät für Informatik und Elektrotechnik, April 2004.
- Horn 1984 Horn, B. K. P.: Extended Gaussian Images. In: Proceedings of the IEEE, Vol. 82, No. 12 (December), pp. 1671-1686, 1984.
- Hornegger 1994 Hornegger, J.; Niemann, H.; Paulus, D.; Schlottke, G.: Object Recognition using Hidden Markov Models. In: Gelsema, E. S. and Kanal, L. N.: Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems. 16. Machine Intelligence and Pattern Recognition. Amsterdam: Elsevier, pp. 37-44, 1994.
- IFR 2009 IFR Statistical Department: World Robotics 2009.
- IMU 2006 Industrieverband Massivumformung (IMU): Simulation in der Massivumformung. Info-Reihe Massivumformung, Extraausgabe. Februar 2006.
- Jähne 2002 Jähne, B.: Digitale Bildverarbeitung. 5. überarb. und erw. Aufl. Berlin, Heidelberg: Springer, 2002.

- Jiang 1997 Jiang, X.; Bunke, H.: Dreidimensionales Computersehen: Gewinnung und Analyse von Tiefenbildern. Berlin: Springer, 1997.
- Kawasaki 1999 Kawasaki Ltd.: Application Example of „PC-Based Robot Control“: Bin-Picking Robot, Bin-Picking System Using Robot. Informationsblatt, 1999.
- Kelley 1985 Kelley, R. B.; Gouin, P. R.: Using Neighbourhood Operators to Detect Graspable Inside and Outside Edges. In: SPIE Vol. 595 Computer Vision for Robots, 1985.
- Kennedy 1998 Kennedy, J.; Spears, W. M.: Matching Algorithms to Problems: An Experimental Test of the Particle Swarm and Some Genetic Algorithms on the Multimodal Problem Generator. In: Proceedings of the IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, USA, 1998.
- Kennedy 2001 Kennedy, J.; Eberhart, R. C.: Swarm Intelligence. San Francisco: Morgan Kaufmann, 2001.
- Kirkegaard 2006 Kirkegaard, J.; Moeslund, T. B.: Pose Estimation Using Structured Light and Harmonic Shape Contexts. In: Advances in Computer Graphics and Computer Vision: International Conferences VISAPP and GRAPP 2006, Setúbal, Portugal, February, pp. 25-28, 2006.
- Koenderink 1979 Koenderink, J. J.; Van Doorn, A. J.: Internal representation of solid shape with respect to vision. In: Biological Cybernetics, Vol. 32, No. 4, pp. 211-216, 1979.
- Kristensen 2001 Kristensen, S.; Estable, S.; Kossow, M.; Brösel, R.: Bin-picking with a solid state range camera. In: Robotics and Autonomous Systems, Volume 35, Issues 3-4, pp. 143-151, 30 June 2001.
- Kullback 1951 Kullback, S.; Leibler, R. A.: On Information and Sufficiency. In: Annals of Mathematical Statistics 22 (1), pp. 79–86, 1951.
- Ledermann 2002 Ledermann, T.; Schraft, R. D.: Intelligente Entnahme chaotisch gelagerter Objekte. In: Dillmann, Rüdiger (Tagungsleitung) u.a.; VDI/VDE-Gesellschaft Meß- und Automatisierungstechnik (GMA) u.a.: Robotik 2002: Leistungsstand, Anwendungen, Visionen, Trends.

Tagung Ludwigsburg, 19. und 20. Juni 2002. Düsseldorf: VDI Verlag, S. 257-262 (VDI-Berichte 1679), 2002.

- Ledermann 2003 Schraft, R. D.; Ledermann, T.: Intelligent picking of chaotically stored objects. In: Automated Imaging Association u.a.: ISR 2003 - Tutorial #5 - Cutting Edge Research in Robotics: 34th International Symposium on Robotics. June 2-5, 2003, Rosemont (Chicago), Illinois. Ann Arbor, MI, USA, 2003.
- Ledermann 2004 Ledermann, T.: Einsatz optischer Sensoren für die intelligente und flexible Handhabungstechnik. In: Schraft, Rolf Dieter (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA: Objekterkennung in der Robotik: Technologien, Methoden, Anwendungen, 21. Oktober 2004, Stuttgart. Stuttgart: FpF - Verein zur Förderung produktionstechnischer Forschung, S. 53-62, 2004.
- Ledermann 2005 Ledermann, T.: Effiziente Bildverarbeitung für die Logistik. In: Schraft, Rolf Dieter (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA: Roboter in der Intralogistik: Aktuelle Trends. Moderne Technologien. Neue Anwendungen. Fraunhofer IPA Technologieforum F 113, 16. Februar 2005, Stuttgart. Stuttgart: FpF - Verein zur Förderung produktionstechnischer Forschung, S. 36-48, 2005.
- Ledermann 2005a Ledermann, T.: MOPRUM: Modulare Prozessautomatisierung in der Metallumformung. In: Schraft, Rolf Dieter (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA: Robotik und Handhabungstechnik in der Produktion: Aktuelle Trends, moderne Technologien, neue Anwendungen. Fraunhofer IPA Technologieforum 8. und 9. Dezember 2005, Stuttgart. Stuttgart : FpF - Verein zur Förderung produktionstechnischer Forschung, S. 195-200, 2005.
- Ledermann 2006 Ledermann, T.: Ansichtenbasierte 3-D-Objektlageerkennung für die Handhabungstechnik. In: Schraft, Rolf Dieter (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA u.a.: Objekterkennung in der Robotik: Technologien - Methoden - Anwendungen, 2. Workshop F 135, 6. Oktober 2006, Stuttgart. Stuttgart: FpF - Verein zur Förderung produktionstechnischer Forschung, S. 128-140, 2006.

- Ledermann 2007 Ledermann, T.: Bildverarbeitungssysteme für Handhabungsaufgaben im innerbetrieblichen Materialfluss. In: Westkämper, Engelbert (Hrsg.) u.a.: Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA: Roboter in der Intralogistik: Aktuelle Trends - Neue Technologien - Moderne Anwendungen. Fraunhofer IPA Technologieforum F 144, 13. März 2007, Stuttgart. Stuttgart: FpF - Verein zur Förderung produktionstechnischer Forschung, S. 31-44, 2007.
- Ledermann 2007a Ledermann, T.: Swarm-based 3-D pose recognition in Depth Data for 3-D bin picking. In: Robotic Industries Association u.a.: International Robots & Vision Show / CD-ROM : Technical Conference Proceedings. June 12-14, 2007, Rosemont (Chicago), Illinois, USA. 11 S., 2007.
- Ledermann 2007b Ledermann, T.; Wegener, K.: Handhabung von Objekten durch 3D-Objektlageerkennung. In: Robotworld 2 (2007), Nr. 1, S. 40-41, 2007.
- Lehmann 2000 Lehmann, K.; Neugebauer, J.-G.; Schraft, R. D.: Depalletizing Of Tires With Robust Object Localization. In: Proceedings of the 31th International Symposium on Robotics (ISR), Ottawa, pp. 206-207, 2000.
- Leibe 2001 Leibe, B.: Segmentierungsfreie Objekterkennung unter Verwendung von Tiefenbildern. Diplomarbeit, Universität Stuttgart, Fakultät für Informatik, 2001.
- Little 1985 Little, J. J.: Extended Gaussian images, mixed volumes, shape reconstruction. In: Proceedings of the first annual symposium on Computational geometry, Baltimore, Maryland, United States, ACM, New York, USA, pp. 15-23, 1985.
- Lovberg 2002 Lovberg, M.: Improving Particle Swarm Optimization by Hybridization of Stochastic Search Heuristics and Self Organized Critically. Master's Thesis. Department of Computer Science, University of Aarhus, Denmark, 2002.
- Lowe 2004 Lowe, D. G.: Distinctive image features from scale-invariant keypoints. In: International Journal of Computer Vision, 60, 2 (2004), pp. 91-110, 2004.

- Matrox 2005 Matrox Imaging Library, Handbuch, 2005.
- Mohais 2005 Mohais, A. S.; Mendes, R.; Ward, C.; Posthoff, C.: Neighborhood Re-Structuring in Particle Swarm Optimization. In: Proceedings of Advances in Artificial Intelligence (AI 2005): 18th Australian Joint Conference on Artificial Intelligence, Sydney, Australia, December 2005, pp. 776-785, 2005.
- Morgan 1896 Morgan, L. C.: On Modification and Variation. First published in Science 1896, NS 4, No. 99, pp. 733-740, 1896.
- Müller 1992 Müller, C.: Verwendung von Bildauswertungsmethoden zur Erkennung und Lagebestimmung von generischen polyedrischen Objekten im Raum. Dissertation, Universität Karlsruhe, Fakultät für Informatik, Dissertationen zur Künstlichen Intelligenz, St. Augustin: infix-Verlag, 1992.
- Müller 1999 Müller, M.; Längle, T.; Wörn, H.: Grasping and reliable picking of overlapping objects in unstructured scenes. In: Proceedings of the International Conference on Advanced Robotics ICAR, Tokyo, p. 6, 1999.
- Müller 2000 Müller, M.; Wörn, H.: Planning of rapid grasp operations in unstructured scenes. In: Proceedings of the International Conference on Intelligent Robots and Systems, 2000. (IROS 2000): October 31 - November 5, 2000, Kagawa University, Takamatsu, Japan, IEEE Operations Center, Piscataway, NJ, p. 7, 2000.
- Müller 2001 Müller, M.: Entnahmestrategien zur Objektvereinzelung mittels Industrierobotern. Dissertation, Karlsruhe Universität, Fakultät für Informatik, 2001.
- Müller 2001a Müller, M.; Wörn, H.: A new generation robot system for object picking. In: Conference documentation of the International Conference on Multisensor Fusion and Integration for Intelligent Systems, 2001. (MFI 2001): August 20 - 22, 2001, Kongresshaus, Baden-Baden, Germany, VDI/VDE-GMA, Düsseldorf. pp. 55-60, 2001.
- Ohba 1996 Ohba, K.; Ikeuchi, K.: Recognition of the Multi-Specularity Objects Using the Eigen-Window. In: Proceedings of ICPR, pp. 692-696, 1996.

- Ohba 1997 Ohba, K.; Ikeuchi, K.: Detectability, Uniqueness and Reliability of Eigen-Windows for Stable Verification of Partially Occluded Objects. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, Issue 9, pp. 1043-1048, September 1997.
- Omran 2005 Omran, M. G. H.: Particle Swarm Optimization Methods for Pattern Recognition and Image Processing. Ph.D. Thesis, Department of Computer Science, University of Pretoria, South Africa, 2005.
- Onda 1995 Onda, T.; Igura, H.; Niwakawa, M.: A Handling System for Randomly Placed Casting Parts Using Plane Fitting Technique. In: Proceedings of IEEE / RSJ International Conference on Intelligent Robots and Systems 1995, Vol. 3, pp. 435-440, August 1995.
- Pannekamp 2005 Pannekamp, J.; Ledermann, T.: Was Tiefenbilder aussagen: Neue Anwendungen in Robotik, Navigation, Sicherheitstechnik und Interaktion. In: Machine Vision: Die Zukunft der Industriellen Bildverarbeitung (IBV) - Volume Deutsch/Englisch. Würzburg: Vogel, S. 18-19, 2005.
- Park 1999 Park, R.-H.; Ham, Y. K.: 3D object recognition in range images using hidden markov models and neural networks. In: Pattern Recognition, Band 32 (1999), pp. 729-742, 1999.
- Quarteroni 2002 Quarteroni, A.; Sacco, R.; Saleri, F.: Numerische Mathematik 2. Berlin: Springer, 2002.
- Röhrdanz 1998 Röhrdanz, F.: Modellbasierte automatische Greifplanung. Dissertation, Technische Universität Braunschweig, Shaker Verlag, Fortschritte in der Robotik, Band 3, Aachen, 1998.
- Rosenfeld 1982 Rosenfeld, A.; Kak, A. C.: Digital Picture Processing, Bd. I and II. San Diego: Academic Press, 2. Aufl., 1982.
- Rubarth 1996 Rubarth, W.; Gerke, W.; Seier, H.: Stör- und Wertstoffseparation in Abfallbehandlungsanlagen. Umwelt, Vol. 26, No. 10, S. 43-46, 1996.
- Scape 2008 Scape Technologies A/S: Home (2008)
<http://www.scapetechnologies.com/page/Show.action?page=6>
(09.09.2008)

- Schöneburg 1993 Schöneburg, E.; Heinzmann, F.; Feddersen, S.: Genetische Algorithmen und Evolutionsstrategien: Eine Einführung in Theorie und Praxis der simulierten Evolution. Bonn, Paris, Reading (Mass.): Addison Wesley Verlag, 1993.
- Schraft 2000 Schraft, R. D.: Vorlesungsskript Automatisierungstechnik, Institut für Industrielle Fertigung und Fabrikbetrieb Universität Stuttgart, 2000.
- Schraft 2003 Schraft, R. D.; Ledermann, T.: Intelligent picking of chaotically stored objects. In: Assembly Automation 23 (2003), 1, pp. 38-42, 2003.
- Schrijver 1998 Schrijver, A.: Theory of Linear and Integer Programming. Hoboken, New Jersey: John Wiley & Sons, 1998.
- Shen 2007 Shen, X.; Wei, K.; Wu, D.; Tong, Y.; Li, Y.: A Dynamic Adaptive Dissipative Particle Swarm Optimization with Mutation Operation. In: Proceedings of the IEEE International Conference on Control and Automation, Guangzhou, China, May 30 - June 1, pp. 586-589, 2007.
- Shi 1998 Shi, Y.; Eberhart, R. C.: Parameter Selection in Particle Swarm Optimization. In: Proceedings of the Conference on Evolutionary Programming VII (EP 98). Springer Berlin/Heidelberg: Springer, pp. 591-600, 1998.
- Shi 1999 Shi, Y.; Eberhart, R. C.: Empirical Study of Particle Swarm Optimization. In: Congress on Evolutionary Computing, Vol. III, pp. 1945-1950, 1999.
- Shi 2001 Shi, Y.; Eberhart, R. C.: Fuzzy Adaptive Particle Swarm Optimization. In: Proceedings of the Congress on Evolutionary Computation. Seoul, South Korea, pp. 101-106, 2001.
- Sick 2005 Sick AG: Laser-Messsystem LMS 400 Betriebsanleitung: Waldkirch, 2005.
- SickInside 2002 N.N.: 3D-Lageerkennung durch Scannen und Schwenken. SickInside, Kundenmagazin von SICK, S.18, 1/2002.
- Siggelkow 2002 Siggelkow, S.: Feature Histograms for Content-Based Image Retrieval. Dissertation, Albert-Ludwigs-Universität Freiburg, Fakultät für Angewandte Wissenschaften, 2002.

- SMErobot 2008 SMErobot™-Projekt: Demonstrator D2 - Plug and Produce Detailed Specification (internal report).
- Solidwork 2008 Solid Works: SolidWorks - 3D Mechanical Design and 3D CAD Software (2008) <http://www.solidworks.de/> (11.10.2008).
- Spall 2003 Spall, J.: Introduction to Stochastic Search and Optimization. Hoboken, New Jersey: John Wiley & Sons, 2003.
- Steinbrecher 1993 Steinbrecher, R.: Bildverarbeitung in der Praxis. München, Wien: Oldenbourg, 1993.
- Stockmann 1988 Stockmann, G. C.; Chen, S.-W.; Hu, G.; Shrikhande, N.: Sensing and Recognition of Rigid Objects Using Structured Light. In: IEEE Control Systems Magazine, Vol. 8, Issue 3, pp.14-22, 1988.
- Stotz 2008 Stotz, M.; Verl, A.: Ein neuer Ansatz zur Erkennung der Objektlage von ungeordneten Bauteilen für die automatisierte Handhabung. In: Tagungsband der Robotik 2008. Leistungsstand, Anwendungen, Visionen, Trends: Tagung München, 11. und 12. Juni 2008. Düsseldorf: VDI-Verlag, 2008 (VDI-Berichte 1212), S.131-134, 2008.
- Suk 1992 Suk, M.; Bhandarkar, M. S.: Three-Dimensional Object Recognition from Range Images. New-York: Springer-Verlag, 1992.
- Svia 2005 SVIA Industrial Automation: Multiflex (2005) <http://www.svia.se/de/products.asp?id=2> (04.09.2008).
- Swain 1991 Swain, M.; Ballard, D.: Color indexing. In: International Journal of Computer Vision, 7(1): pp. 11–32, 1991.
- Tönnies 1997 Tönnies, K. D.; Lemke, H. U.: 3D-Computergrafische Darstellungen. München, Wien: Oldenbourg, 1994.
- Törn 1989 Törn, A.; Zilinskas, A.: Global Optimization. Lecture Notes in Computer Science, N° 350. Berlin: Springer-Verlag, 1989.
- Trobina 1995 Trobina, M.: From Planar Patches to Grasps: a 3D Robot Vision System Handling Unmodeled Objects. Dissertation, Hochschulschriften-Nr.: Diss. ETH 11326, Eidgenössische Technische Hochschule, Zürich, 1995.

- Van den Bergh 2002 Van den Bergh, F.; Engelbrecht A. P.: A new Locally Convergent Particle Swarm Optimizer. In: Proceedings of the IEEE Conference on Systems, Man and Cybernetics, Hammamet, Tunisia, pp. 96-101, 2002.
- Veeramachaneni 2003 Veeramachaneni, K.; Peram, T.; Mohan, C.; Osadciw, L. A.: Optimization Using Particle Swarm With Near Neighbor Interactions. Lecture Notes Computer Science, Vol. 2723, Berlin: Springer, 2003.
- Verl 2007 Verl, A.; Fritsch, D.; Ledermann, T.; Mütterich, H.: Greifen und Sehen als Schlüsseltechnologien für Roboter: Flexible Automatisierung von Handhabungs- und Montageaufgaben durch innovative Greifer und intelligente 3-D-Bildverarbeitung. In: Wt Werkstattstechnik 97 (2007), Nr. 9, S. 694-699, 2007.
- Verl 2009 Verl, A.; Stotz, M.; Kühnle, J.: 3-D-4-Robots: Neues System zur automatisierten Handhabung ungeordneter Werkstücke: Lösung des Griffs in die Kiste. In: Schweizer Logistik-Katalog. 34 (2009), S. 122-124, 2009.
- VTK 2010 The Visualization ToolKit: VTK Home Page (2010)
<http://www.vtk.org/> (14.08.2010).
- Wahl 2003 Wahl, E.; Hillenbrand, U.; Hirzinger, G.: Surflet-Pair-Relation Histograms: A Statistical 3D-Shape Representation for Rapid Classification. In: Proceedings of the International Conference on 3-D Digital Imaging and Modeling (3DIM 2003), Banff (Canada), pp. 474-481, October 2003.
- Walter 1998 Walter, P.: Optimized feature exploitation for 3D Object Recognition using ART Neural Network. In: Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, (IECON '98). 31. Aug - 4. Sep 1998, Piscataway, NJ: IEEE Service Center, Vol. 4, pp. 2063-2068, 1998.
- Wegener 2003 Wegener, K.; Ledermann, T.; Schraft, R. D.: Ein Baukastensystem für die intelligente Handhabungstechnik. In: Automatisierungstechnische Praxis 45 (2003), Nr. 11, S. 36-39, 2003.

- Wegener 2005 Wegener, I.: Simulated Annealing Beats Metropolis in Combinatorial Optimization. Lecture Notes in Computer Science, 3580. Berlin/Heidelberg: Springer, 2005.
- Wolf 1999 Wolf, A.: Sortieren von Leichtverpackungen mit Handhabungssystemen. Dissertation, Heimsheim: Jost-Jetter, 1999 (IPA-IAO-Forschung und -Praxis 302).
- Wolfson 1990 Wolfson, H. J.: Model-Based Object Recognition by Geometric Hashing. In: Proceedings of 1st European Conference on Computer Vision (ECCV'90), Lecture Notes in Computer Science, Vol. 427, Berlin: Springer Verlag, pp. 526-536, 1990.
- Wolter 1986 Wolter, J. D.; Volz, R. A.; Woo, A. C.: Automatic Generation of Gripping Positions for Assembly. In: Robot Grippers. Pham, D. T.; Heginbotham, W. B.; Berlin: Springer-Verlag, pp. 55-74, 1986.
- Xie 2002 Xie, X.-F.; Zhang, W.-J.; Yang, Z.-L.: A Dissipative Particle Swarm Optimization. In: Proceedings of the IEEE congress on Evolutionary Computation, pp. 1456-1461, 2002.
- xpertgate 2010 xpertgate: Positionsbestimmung von Kartons zur Portalrobotersteuerung (2010)
<http://www.xpertgate.de/applikation/Karton-Postionsbestimmung/>
(05.08.2010)
- xpertgate 2010a xpertgate: Entpalettieren teilgeordneter Kurbelgehäuse (2010)
http://www.xpertgate.de/applikation/Entpalettieren_Kurbelgehaeuse/
(05.08.2010)
- Zhang 2003 Zhang, Y.; Koschan, A.; Abidi, M.: Superquadrics Based 3D Object Representation of Automotive Parts Utilizing Part Decomposition. In: Proceedings of SPIE 6th International Conference on Quality Control by Artificial Vision, Gatlinburg, TN, Vol. 5132, pp. 241-251, May 2003.