Fraunhofer Institut für Informations- und
Datenverarbeitung (IITB)
Karlsruhe

Geschäftsfeld Erkennungssysteme (ERS)

Ansichtsbasierte Erkennung und Lokalisierung von Objekten zur Initialisierung eines Verfolgungsprozesses

Diplomarbeit
von
Christian Plagemann

Eingereicht am 31.01.2004

Referenten: Prof. Dr. H. U. Steusloff, Prof. Dr. Ing. R. Dillmann
Betreuer: Dr. T. Müller, Dr. habil. J. Pauli

Kurzfassung

Ziel der vorliegenden Arbeit ist die Realisierung eines Systems zur räumlichen Lokalisierung von Objekten in komplexen Szenen. Das System soll im Rahmen des Projektes MQube am Fraunhofer Institut IITB in Karlsruhe zur Initialisierung eines Objektverfolgungsprozesses eingesetzt werden. In dem Projekt geht es um Planungen theatertechnischer Szenarien, bei denen verschiedene Objekte von mehreren Benutzern bewegt und diese Bewegungen von dem Verfolgungsprozess erfasst werden. Die sich aus dieser Anwendung ergebende Komplexität der Szene (z.B. schwierige Beleuchtungsverhältnisse, komplexer Hintergrund, Objektverdeckungen) stellt hohe Anforderungen an die Leistungsfähigkeit und Robustheit des Verfolgungs- bzw. Lokalisierungsverfahrens. Neben den in MQube gesetzten Zielen soll bei der Realisierung des Lokalisierungssystems auch eine darüber hinausgehende breite Einsetzbarkeit in anderen Anwendungen erreicht werden.

Im Hinblick auf die gestellten Anforderungen kann als Basis ein konkretes Verfahren in der Literatur ausfindig gemacht und als Basis verwendet werden. Das gewählte Verfahren ist ansichtsbasiert und verwendet eine probabilistische Modellierung des Objektaussehens. In der vorliegenden Arbeit wird dieses rein im Bildbereich operierende Erkennungsverfahren auf ein geeignetes 3D-Lokalisierungssystem ausgebaut. Ferner wird die Möglichkeit geschaffen, aus vorhandenen 3D-Formmodellen die für die Erkennung notwendige ansichtsbasierte Repräsentation zu erzeugen. Auf diese Weise wird das Ziel der breiten Einsetzbarkeit des realisierten Lokalisierungssystems weitestgehend erreicht und es können die im Projekt MQube vorhandenen Formmodelle unmittelbar genutzt werden.

Detailierte Untersuchungen der Erkennungs- und Lokalisierungsleistungen auf künstlichen und realen Daten belegen die weitgehende Praxistauglichkeit des Systems unter schwierigen Bedingungen.

Inhaltsverzeichnis

Kurzfassung **3**

Inhaltsverzeichnis **i**

Einleitung **1**

1 Verfahren zur Objekterkennung **3**

1.1 Struktur von Erkennungssystemen 4

1.2 Wissensrepräsentation 6

1.3 Modellanpassung 8

1.4 Verfahrensentscheidung 11

2 Ansichtsbasierte Objekterkennung **13**

2.1 Wissensrepräsentation 13

2.1.1 Repräsentation von Objektansichten und Bildern 14

2.1.2 Repräsentation von Modellansichten 16

2.1.3 Repräsentation von Szenenwissen 17

2.1.4 Notation 18

2.2 Lernverfahren 19

2.2.1 Generalisierung über einer Menge von Objektansichten 19

2.2.2 Ballung von Objektansichten 20

2.3 Erkennungsverfahren 22

2.3.1 Gütemaß für Merkmalspaarungen 23

2.3.2 Wahrscheinlichkeit einer Merkmalspaarung 25

2.3.3 Probabilistische Korrespondenzsuche 26

2.3.4 Verifikation 27

3 Realisierung des Lokalisierungssystems 29
3.1 Einbettung in das Objektverfolgungssystem . . . 30
3.2 Systemstruktur . . . 31
3.3 Praktische Umsetzung . . . 32

4 Gewinnung von Objektansichten 35
4.1 Methoden zur Gewinnung von Objektansichten . . . 35
4.1.1 Abbildung des realen Objektes . . . 36
4.1.2 Erzeugung künstlicher Objektansichten . . . 37
4.1.3 Formrekonstruktion und Erzeugung künstlicher Ansichten . . . 38
4.2 Gleichmässige Verteilung von Blickwinkeln . . . 38
4.3 Unterteilung sphärischer Dreiecksnetze . . . 41
4.3.1 Unterteilung über die Seitenmitten . . . 42
4.3.2 Unterteilung über die Schwerpunkte . . . 43
4.3.3 Vergleich und Kombination der Unterteilungsverfahren . . . 43
4.4 Praktische Umsetzung . . . 45

5 Schätzung der räumlichen Objektlage 49
5.1 Kameramodell . . . 50
5.2 Lage des Objektabbildes auf der Bildebene . . . 51
5.3 Transformationskette . . . 53
5.3.1 Verschiebung des Objektes auf den Ursprung . . . 53
5.3.2 Übergang zum Koordinatensystem der Objektansicht . . . 56
5.3.3 Von der Objektansicht in das Formmodell . . . 57
5.4 Lokalisierung . . . 58

6 Evaluierung des Lokalisierungssystems 61
6.1 Theoretische und praktische Voruntersuchungen . . . 61
6.1.1 Einfluss der projektiven Verzerrung . . . 61
6.1.2 Voraussetzungen des Verfahrens zur Graphenanpassung . . . 63
6.1.3 Leistungsfähigkeit des Merkmalsrepertoirs und der Graphendarstellung . . . 64
6.1.4 Einfluss der Umgebungsverteilung . . . 68
6.2 Untersuchungen anhand künstlicher Daten . . . 68

6.2.1 Testautomatisierung 68
6.2.2 Verlässlichkeit des Verifikationsmaßes 71
6.2.3 Lokalisierungsleistung und Genauigkeit 73
6.2.4 Zeitkomplexität 75
6.2.5 Einfluss der Trainingsmenge auf das Objektmodell 76
6.3 Ergebnisse bei realen Daten 78
6.3.1 Lokalisierungsleistung, Genauigkeit und Ergebnisse beim Objekt *Hantel* 78
6.3.2 Ergebnisse bei Freiformobjekten 81
6.3.3 Ergebnisse bei weiteren Objekten 81
6.4 Diskussion der Ergebnisse 88

7 Abschlussbetrachtungen **91**

A Merkmalsrepertoire **95**

B Ergänzende Berechnungen **97**
B.1 Kalibrierung einer virtuellen Kamera 97
B.2 Kombination von Lagetransformationen 99
B.3 Transformationen 100
B.4 Sphärische Winkel 102
B.5 Projektive Verzerrung von Objektansichten 102

C Modellansichten und Testbilder **107**
C.1 Modellansichten 107
C.1.1 Modellansichten aus 130 Trainingsbildern 107
C.1.2 Modellansichten aus 66 Trainingsbildern 109
C.1.3 Modellansichten aus 34 Trainingsbildern 110
C.2 Testbildmenge *M3-Hantel-Real* 110

Literaturverzeichnis **112**

Einleitung

Im Rahmen des Projekts MQube[1] wird am Fraunhofer Institut für Informations- und Datenverarbeitung IITB in Karlsruhe ein System zur Verfolgung von Objekten in Bildfolgen entwickelt. Zum Einsatz kommt die Objektverfolgung als Mittel zur Benutzerinteraktion innerhalb einer Augmented-Reality-Umgebung. Das Projekt MQube hat zum Ziel, Teams bei der Planung von Szenen, Räumen und Abläufen zu unterstützen. Dies geschieht durch die Anreicherung einer realen Szene auf einer Modellbühne mit virtuellen Objekten, welche über Projektionsbrillen für die Benutzer sichtbar sind. Um die Interaktion mit dem System, d.h. insbesondere mit den virtuellen Objekten, möglichst intuitiv zu gestalten, stehen dem Benutzer eine Reihe von realen Objekten als Hilfsmittel zur Verfügung. Die Bewegungen dieser Objekte werden von dem Verfolgungsprozess registriert, an das Hauptsystem weitergeleitet und dort in entsprechende Bewegungen der virtuellen Objekte umgesetzt.

Die Aufgabe der vorliegenden Arbeit ist es, ein Verfahren zur Schätzung der räumlichen Lage von Objekten aus einer einzelnen Aufnahme zu realisieren, um mit dem Ergebnis den Verfolgungsprozess zu initialisieren. Auf der Modellbühne des Projektes MQube kommen verschiedene licht- und theatertechnische Installationen wie Spotscheinwerfer und Bühnennebel zum Einsatz, welche einen starken Einfluss auf die Kamerabilder haben. Durch die Hand des Benutzers, der ein Objekt in der Szene bewegt, kommt es zwangsläufig zu partiellen Verdeckungen des Objektes. Die möglicherweise stark variierenden Beleuchtungsverhältnisse, die auftretenden Verdeckungen der Objekte und der oft komplexe Bühnenhintergrund stellen harte Anforderungen an Erkennungs- und Verfolgungsverfahren. Aus diesem Grund stehen die Flexibilität des Ansatzes und seine Praxistauglichkeit in komplexen Realweltszenen im Vordergrund. Das stellt hohe Anforderungen an die Erkennungsleistung und die Robustheit der Lokalisierung. Der Genauigkeit der Lokalisierung kommt dagegen eine relativ geringe Bedeutung zu, da Ungenauigkeiten von der nachgeschalteten Objektverfolgung grundsätzlich ausgeglichen werden. Es genügt daher eine grobe Lokalisierungsgenauigkeit, die aber hinreichend sein muß, um das Aufsetzen der Objektverfolgung zu ermöglichen. Neben den für MQube gesetzten Zielen soll bei der Realisierung des Systems vor allem auch auf eine darüber hinausgehende Einsetztbarkeit in anderen Gebieten und Anwendungen,

[1] siehe www.mqube.de .

d.h. auf die dazu notwendige Flexibilität des Systems, Wert gelegt werden. Im Hinblick auf die gesetzten Ziele ist zunächst das hierfür am geeignetsten erscheinende Verfahren aus der Literatur ausfindig zu machen, es umzusetzen und schließlich entsprechend der Anforderungen auszubauen.

Die vorliegende Arbeit beginnt mit der Darstellung des Objekterkennungsproblems sowie einer Übersicht über die wichtigsten Ansätze aus der Fachliteratur. Kapitel 2 stellt den anhand der gesteckten Ziele ausgewählten ansichtsbasierten Erkennungsansatz vor. Kapitel 3 beschreibt die Erweiterung des Verfahrens auf ein Lokalisierungssystem. Kapitel 4 und Kapitel 5 gehen anschließend auf die Herleitung und Umsetzung der entwickelten Komponenten zur Ansichtengewinnung aus 3D-Formmodellen und zur räumlichen Lokalisierung ein. In Kapitel 6 werden Ergebnisse des Systems und Untersuchungen zur Leistungsfähigkeit präsentiert. Abschließend wird in Kapitel 7 ein Resümee gezogen zur Praxistauglichkeit des Ansatzes, die Stärken und Schwächen genannt sowie mögliche Anwendungsgebiete angegeben.

Kapitel 1

Verfahren zur Objekterkennung und Lokalisierung

In diesem Kapitel wird ein Überblick über den Problemkreis der Objekterkennung und Lokalisierung sowie über die entsprechenden Verfahren aus der Fachliteratur gegeben. Dabei steht das Problem der Erkennung eines einzelnen Objektes in einem zweidimensionalen Grauwertbild im Vordergrund. Verfahren, welche sich speziell mit dem Einsatz mehrerer Kameras, Bildfolgen oder mehrdimensionalen Aufnahmen auseinandersetzen, seien dabei ausgeklammert.

Die Aufgabe von Systemen zur Objekterkennung und Lokalisierung (im Folgenden kurz *Erkennungssysteme* genannt) ist es, reale Objekte in digitalen Bildern automatisch zu erkennen und deren räumliche Lage in der Szene zu schätzen. Es handelt sich dabei um ein Problem der Modellanpassung: Das jeweilige Objektmodell, gegeben durch eine systeminterne Darstellung von Form und Aussehen des Objektes, wird an das Objektabbild innerhalb des Kamerabildes angepasst.

Eingaben eines solchen Systems sind

- Beschreibungen der zu erkennenden Objekte in einer Objektdatenbank,
- ein mittels Sensoren aufgenommenes Abbild der Szene und
- Parameter zur Steuerung des Erkennungsprozesses.

Ausgaben des Systems sind

- Erkennungsergebnisse, bestehend aus der Identifikation eines Objektes, der Schätz-

ung von dessen Lage und oft einer Angabe über die Verlässlichkeit und Genauigkeit der Erkennung.

1.1 Struktur von Erkennungssystemen

Die meisten Erkennungssysteme arbeiten nach dem in Abbildung 1.1 dargestellten Schema. Systeme, die nicht exakt dieser Darstellung entsprechen, können oft anschaulich durch Angabe der Abweichungen von dem Schema beschrieben werden.

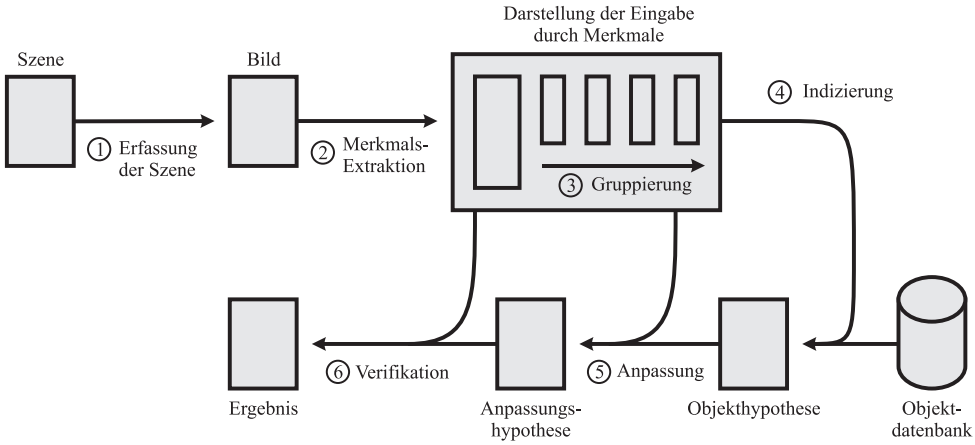

Abbildung 1.1: Verarbeitungsschema eines typischen Erkennungssystems. Datenrepräsentationen sind durch Rechtecke dargestellt, Verarbeitungsschritte durch Pfeile.

Die Verarbeitungsschritte eines typischen Erkennungssystems:

1. **Erfassung** der Szene:
   Je nach Anwendungsgebiet sind unterschiedliche Sensoren einsetzbar: lichtempfindlich (farb- oder nur helligkeitsempfindlich, Infrarot), Radar, Ultraschall / Sonar, Röntgen, NMR (Kernspin-Resonanz-Tomographie), MRT (Magnet-Resonanz-Tomographie) oder PET (Positronen-Emissions-Tomographie oder Szintigraphie). Das Ergebnis der Messung kann als zweidimensionales Bild oder als so genannte *Raumbild* oder *Oberflächenprofil* vorliegen. Im letzteren Fall ist es repräsentiert durch eine 3D-Punktwolke oder ein 2D-Tiefenbild. Techniken zur Erzeugung von Raumbildern sind: Triangulierung, aktive De-/Fokussierung, Radar, Sonar und Moire-Interferometrie.

2. **Merkmalsextraktion**:
   Aus dem Eingabebild werden *Bildmerkmale* extrahiert und in einer eigenen Datenrepräsentation abgelegt. Üblicherweise benutzt der Erkennungsprozess im Weiteren nur noch diese Darstellung der Eingabedaten und greift nicht mehr auf das

Originalbild zu. In dieser Arbeit wird der Begriff *Merkmal* als Oberbegriff für strukturbeschreibende Einheiten verwendet, die sich aus einem Bild extrahieren lassen. Wenn nicht mit dem Prädikat *global* versehen, sind damit implizit *lokale Merkmale* gemeint, also Bildelemente mit beschränkter räumlicher Ausdehnung, denen eine Position im Bild zugeordnet werden kann.

3. **Gruppierung** der Merkmale:
   Die extrahierten Merkmale werden zu Gruppen zusammengefasst und oft in einem Graph oder hierarchisch in einem Baum gespeichert. Kritierien für die Gruppierung können - nach dem oft angewandten Prinzip des *perzeptual grouping* aus der Psychologie - Parallelität von Kanten, Colinearität von Kanten oder Nähe von Endpunkten sein. Gruppen sollten Objektgrenzen nicht überschreiten, was im Allgemeinen und besonders bei komplexem Hintergrund nicht garantiert werden kann. Aus diesem Grund wird das Problem der Gruppierung von Bildmerkmalen, welches eng mit dem Problem der Segmentierung von Bildern verwandt ist, als schwer angesehen. Viele Publikationen weisen darauf hin, dass bessere Gruppierungs- und Segmentierungsverfahren eine deutliche Leistungssteigerung der Objekterkennungsverfahren zur Folge hätten.

4. **Indizierung** der Objektdatenbank:
   Aktuelle Verfahren verwenden zunehmend Indizierungsstrategien, um schnell auf eine große Objektdatenbank zugreifen zu können. Hierbei wird aus einer Menge von Merkmalen ein Schlüssel erzeugt, mit dessen Hilfe eine Objekthypothese in der Datenbank gefunden werden kann.

5. **Anpassung** des Objektmodells an die Bildmerkmale:
   Es wird nach einer Transformation gesucht, die das interne Objektmodell aus der Datenbank mit dem Objektabbild der Eingabe zur Deckung bringt und somit dessen Bildmerkmale möglichst genau vorhersagt. Üblicherweise wird dabei ein Gütemaß für die Passung von Modell, Transformation und Beobachtung verwendet, welches durch Variation des Modells und der Transformation maximiert wird.

6. **Verifikation** des Anpassungsergebnisses:
   Nach der Modellanpassung liegen im Allgemeinen mehrere Hypothesen für erkannte Objekte und deren Lage vor. Diese Objekt-(Lage-)Hypothesen werden in einem Verifikationsschritt bestätigt oder abgelehnt. Einige Erkennungssysteme fassen Anpassung und Verifikation zu einem iterativen Prozess zusammen oder verwenden zur Verifikation das Gütemaß der Anpassung.

Erkennungssysteme lassen sich grob hinsichtlich folgender Kriterien unterscheiden:

1. **Repräsentation des Objektes**: Durch ein 3D-(Form-)Modell oder durch eine Menge von 2D-Ansichten.

2. **Raum der Modellanpassung**: Der Bildraum, der Transformationsraum oder beide.

1.2 Wissensrepräsentation

Zentral für den Entwurf und die Analyse von Erkennungssystemen ist die Wahl der Wissensrepräsentation. Jedes System ist beschränkt durch den Informationsgehalt der Eingabe und deren interner Darstellung. Hohe Erkennungsleistung kann nur erbracht werden, wenn das Abbild der Szene und das Aussehen der Objekte in der Datenbank adäquat beschrieben werden und beide Darstellungsformen in ausreichendem Maße zueinander kompatibel sind.

Repräsentation von Objekten

Man unterscheidet zwischen folgenden zwei Repräsentationsformen (s. Abbildung 1.2):

- Das *formbasierte* oder *strukturbasierte Objektmodell*: Es beschreibt die 3-dimensionale Form eines Objektes und dessen Oberflächeneigenschaften.
- Das *ansichtsbasierte Objektmodell*: Es beschreibt das Aussehen eines Objektes unter verschiedenen Blickwinkeln. Die 3-dimensionale Form des Objektes ist dadurch nur implizit repräsentiert.

Bei der erstgenannten Form, der *modellbasierten Repräsentation*, wird das Aussehen implizit über die Darstellung der Form gespeichert. Aus den geometrischen Merkmalen (Kanten, Ecken, Flächen) des Modells und den Abbildungseigenschaften der Kamera kann auf das Aussehen des Objektes im Bild geschlossen werden. Bei der Modellanpassung werden die geometrischen Merkmale des Modells an die photometrischen des Bildes angepasst. Die modellbasierte Repräsentation setzt voraus, dass die Form des Objektes explizit bekannt ist.

Die zweite Repräsentationsform, die *ansichtsbasierte Repräsentation*, hat im Allgemeinen einen wesentlich höheren Speicherbedarf, kann jedoch auch ohne Kenntnisse über die Form des Objektes angegeben werden. Sie wird meist durch ein Lernverfahren aus einer großen Menge von Bildern des realen Objektes gewonnen. Vorteilhaft daran ist die Tatsache, dass komplizierte Abbildungseigenschaften wie Beleuchtungseffekte, Reflexion oder Selbstverdeckung nicht modelliert werden müssen, sondern implizit durch die Trainingsdaten gegeben sind. Auf der anderen Seite stellt dies die große Forderung an

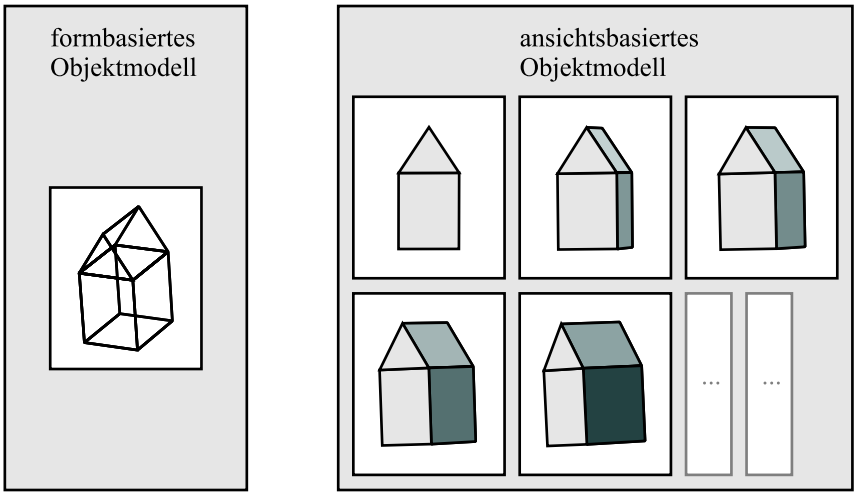

Abbildung 1.2: Repräsentationsformen für Objekte. Links: das formbasierte Objektmodell beschreibt explizit die dreidimensionale Form. Rechts: das ansichtsbasierte Objektmodell besteht aus einer Menge von zweidimensionalen Ansichten.

die Trainingsdaten, repräsentativ für alle möglichen Erscheinungsformen des Objektes zu sein. Außerdem können die Lageparameter nicht frei variiert werden, da nur eine diskrete, endliche Menge von Ansichten vorliegt. Ein wichtiger Vorteil der ansichtsbasierten Repräsentation ist die Möglichkeit, die einzelnen Objektansichten mit denselben Merkmalen wie das Eingabebild zu beschreiben. Man steht dann, im Gegensatz zur modellbasierten Repräsentation, nicht vor dem Problem, Merkmale unterschiedlichem Typs aneinander anpassen zu müssen.

Bildmerkmale und deren Organisation

Es ist von großer Bedeutung, dass die relevanten Bildinformationen in den extrahierten Merkmalen enthalten und leicht zugänglich sind. Anhand zu weniger Merkmale ist eine robuste Erkennung nicht möglich und zu viele wirken negativ auf die Effizienz. In der Literatur werden hauptsächlich folgende Typen von Bildmerkmalen beschrieben:

- *Einfache Merkmale* wie Grauwertgradienten und Kantenelemente.
- *Segmente geometrischer Figuren* wie Geraden, Kreise oder Kurven.
- *Komplexe Merkmale* als Gruppen anderer Merkmale. Dies sind beispielsweise Ecken und Polyeder, welche aus Gruppen von Geradensegmenten gebildet werden.
- *Texturmerkmale* wie die skaleninvarianten SIFT-Merkmale (siehe [Lowe 1999]) oder *local-jet* Merkmale (siehe [Schmid & Mohr 1997]).

- *Geometrische Invarianten* wie z.B. das geometrische Kreuzverhältnis, welches unter perspektivischen Abbildungen erhalten bleibt (siehe [Zisserman, Forsyth, Mundy & al 1995])

- *Globale Merkmale* wie beispielsweise die Eigenvektoren eines Bildes in Vektordarstellung (siehe [Murase & Nayar 1995]).

Bildmerkmale können auf vielfältige Art und Weise repräsentiert und organisiert werden. Neben der Organisation in Listen oder indizierten Datenbanken wird die hierarchische Anordnung in einem Graphen häufig angewandt. In [Swain & Ballard 1991] und [Schiele & Crowley 2000] werden vornehmlich Histogramme von Merkmalen als Zwischenrepräsentation verwendet.

1.3 Modellanpassung

In Abbildung 1.3 sind die im Rahmen der vorliegenden Arbeit betrachteten Erkennungsverfahren gemäß ihrer Zugehörigkeit zu unterschiedlichen Verfahrensklassen dargestellt. Dieser Graph stellt lediglich eine für die hier durchgeführten Betrachtungen geeignete Klassifikation dar. Er ist eine von zahlreichen Möglichkeiten zur Strukturierung der Menge unterschiedlicher Verfahren und erhebt keinen Anspruch auf Vollständigkeit.

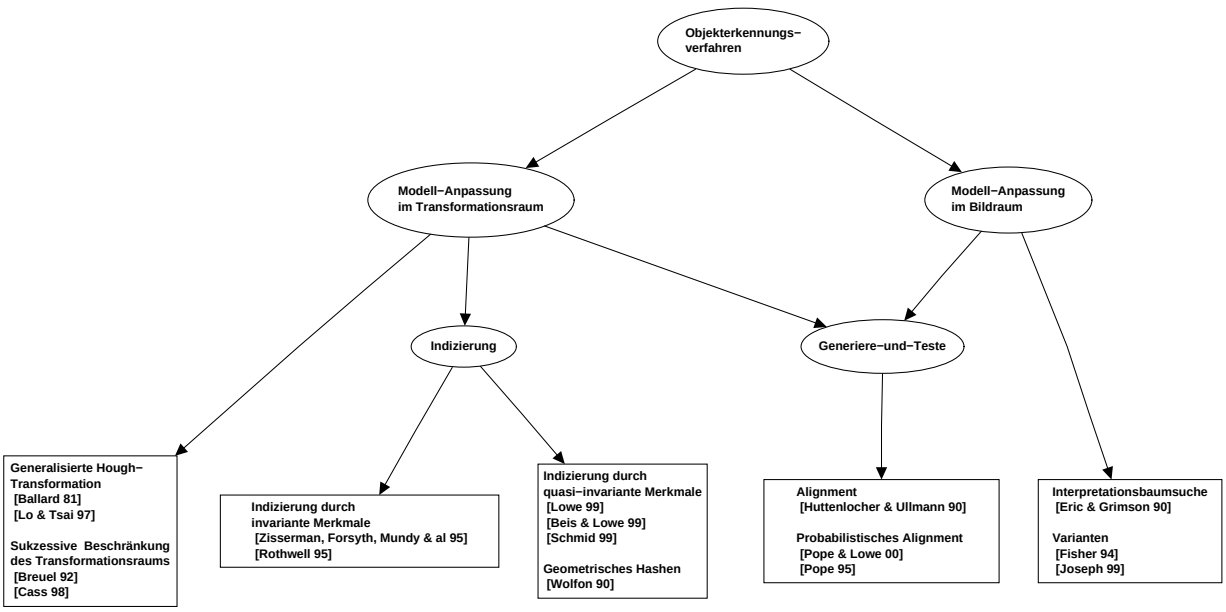

Abbildung 1.3: Einordnung der im Rahmen der vorliegenden Arbeit betrachteten Verfahren aus der Fachliteratur.

Modellanpassung im Bildraum

Unter einer *Interpretation* wird im Kontext von Anpassungsverfahren eine Zuordnung von Bild- und Modellmerkmalen verstanden. Der *Interpretationsbaum* enthält alle derartigen Zuordnungen und soll nach der Besten bezüglich eines Gütekriteriums durchsucht werden. Da es aufgrund der Baumgröße nicht sinnvoll ist, diesen vollständig zu durchsuchen, müssen effiziente Suchstrategieen verwendet werden[1]. In dem in der Literatur oft zitierten Buch [Eric & Grimson 1990] wird dieses Konzept der *Interpretationsbaumsuche* ausführlich beschrieben und mit den wichtigsten Alternativansätzen verglichen. In [Joseph 1999] werden heuristische Erweiterung der Interpretationsbaumsuche diskutiert und experimentell untersucht.

Modellanpassung im Transformationsraum

Die Suche nach der besten Anpassung des Modells an Bildmerkmale kann auch im Raum der Transformationen stattfinden, dem Raum der Abbildungen, die ein Objekt aus seinem Koordinatensystem in das der Szene abbilden. Ein bekanntes Beispiel für dieses Vorgehen ist die *generalisierte Hough-Transformation*, welche auch als Verfahren des *pose clustering* bekannt ist (siehe [Ballard 1981], [Lo & Tsai 1997], und [Häusler & Ritter 1999]). Hier werden einige Bild- und Modellmerkmale einander zugeordnet und entsprechende Transformationshypothesen gebildet. Für jede dieser Hypothesen wird im diskretisierten Transformationsraum ein Zähler erhöht. An den Zählerständen kann somit abgelesen werden, wieviele Merkmalspaarungen mit der jeweiligen Transformation konsistent sind. Mit dem höchsten Zählerstand kann die wahrscheinlichste Transformation gefunden und dadurch das Erkennungsproblem gelöst werden. Problematisch ist dabei der sehr große Transformationsraum[2], in dem ein Maximum nicht gleichzeitig effizient und robust gefunden werden kann.

Einen ähnlichen Ansatz verfolgen die *Indizierungsverfahren*. Hier werden Schlüssel aus wenigen Merkmalszuordnungen für den Zugriff auf konsistente Transformationen erzeugt. Wie bei der Hough-Transformation kann hier das Erkennungsproblem durch das Erhöhen und Auswerten von Zählerständen gelöst werden. In [Beis & Lowe 1999] wird beschrieben, wie beispielsweise die Eigenschaften von Kantenzügen zur Indizierung von Formen eingesetzt werden können. Verfahren des *geometric hashing* (siehe [Wolfson 1990], [Liu 1996]) bilden einen Schlüssel aus der relativen Lage von diskreten Merkmalen zueinander. In [Lowe 1999] werden zur Indizierung so genannte SIFT Schlüssel verwendet. Diese beschreiben die lokale Umgebung eines Bildpunktes und

[1] So z.B. das Abschneiden von inkonsistenten Unterbäumen, das Abbrechen der Suche, wenn im aktuellen Unterbaum keine konsistente Lösung mehr möglich ist oder wenn eine ausreichend konsistente Lösung gefunden wurde.

[2] Sechsdimensional im Falle der räumlichen Objektlokalisierung.

sind nahezu invariant unter einer großen Menge von Transformationen. In [Breuel 1993] werden theoretische Überlegungen zur Komplexität der unterschiedlichen Klassen von Indizierungsverfahren angestellt. Dabei wird unter anderem das Indizieren durch Objektansichten behandelt. [Schmid 1999] und [Olson 1994] entwickeln probabilistische Ansätze zur Indizierung.

In [Breuel 1992] und [Cass 1998] werden Verfahren beschrieben, die den Transformationsraum sukzessive durch geometrische Beschränkungen eingrenzen. Die einfache Form der Beschränkungen, welche aus Merkmalspaarungen gewonnen werden, ermöglicht eine effiziente Suche nach maximal konsistenten Merkmalszuordnungen.

In [Zisserman, Forsyth, Mundy & al 1995] und ausführlich im Buch [Rothwell 1995] werden Indizierungsverfahren vorgestellt, die ausschließlich geometrisch invariante Bildmerkmale verwenden. Geometrisch invariante Merkmale verändern sich nicht unter bestimmten Transformationen. So bleibt beispielsweise das geometrische Kreuzverhältnis unter projektiven Abbildungen erhalten. Da für den allgemeinen Fall der unbeschränkten Punktwolke keine Invarianten angegeben werden können, werden in der Arbeit nur spezielle Objektklassen - wie planare Objekte oder Rotationskörper - behandelt.

Modellanpassung im Bild- und Transformationsraum

Generiere-und-Teste-Verfahren[3] und insbesondere das *Alignment*-Verfahren, welches erstmals in [Huttenlocher & Ullman 1990] beschrieben wird, passen solange Bild- und Modellmerkmale einander an, bis daraus eine Transformationshypothese geschätzt werden kann. Diese wird in einem Verifikationsschritt bestätigt oder verworfen. [Pope & Lowe 2000] und die Dissertation [Pope 1995] zu demselben Themenbereich erweitern dieses Vorgehen um eine probabilistische Sichtweise. Die Nützlichkeit von Merkmalen wird durch ihre Auftretenshäufigkeit in den Trainingsdaten geschätzt und der Anpassungsvorgang entsprechend gesteuert. [Costa & Shapiro 2000] repräsentieren Objektwissen und das Eingabebild durch Merkmalsgraphen und verwenden kleine Teilgraphen als Schlüssel für die Suche nach Übereinstimmungen. Diese Vorgehensweise wird von den Autoren *relationales Indizieren* genannt.

[3] In der englischsprachigen Literatur unter dem Begriff *hypothesize-and-test* bekannt.

1.4 Verfahrensentscheidung

In dieser Arbeit wird das in [Pope 1995] beschriebene ansichtsbasierte Erkennungsverfahren als Ansatz zur räumlichen Lokalisierung von Objekten verwendet. Es besitzt einige vielversprechende Eigenschaften, die gute Ergebnisse in komplexen Szenen erwarten lassen und zu einer näheren Untersuchung anregen:

- Durch die ansichtsbasierte Modellierung wird ein großer Teil des Gesamtaufwands für die Objekterkennung, nämlich die Analyse des Zusammenhangs zwischen Objektform und Aussehen, in einen separaten Lernvorgang ausgelagert. Dadurch kann für den zur Laufzeit relevanten Erkennungsschritt ein relativ effizienter Algorithmus angegeben werden, der das zweidimensionale Objektmodell an das Testbild anpasst. Darüberhinaus erlaubt der ansichtsbasierte Ansatz eine Objekterkennung ohne explizites Formwissen und bietet dem Anwender dadurch mehr Flexibilität.

- An das Repertoire der verwendeten lokalen Bildmerkmale werden nur schwache Anforderungen gestellt, weshalb zum einen vielfältige visuelle Aspekte des Bildes erfasst werden können und zum anderen bei Bedarf die Menge der Bildmerkmale relativ einfach erweitert werden kann.

- Durch den probabilistischen Ansatz und dem daraus resultierenden statistischen Vorgehen wird die Relevanz einzelner Bildmerkmale automatisch ermittelt. Dieser Punkt ist besonders wichtig in Realweltszenen, da solchen Szenen eine hohe Merkmalsanzahl inhärent ist.

- Die konkrete Realisierung der Modellanpassung weist im Detail besondere Vorteile auf, wodurch viele - besonders in Realweltszenen auftauchende - Effekte erfasst und voraussichtlich die damit verbundenen Probleme weitgehend gelöst werden können. Solche Effekte sind z.B. schlechte Sichtbedingungen, komplexer Szenenhintergrund, partielle Verdeckungen und ungünstige Kontraste durch die Beleuchtungsverhältnisse. In diesen Details zeigen sich eine Reihe von zu erwartenden Robustheitsvorteilen gegenüber anderen Verfahren.

Kapitel 2

Ansichtsbasierte Objekterkennung nach A. Pope

In seiner Dissertation [Pope 1995] entwickelte Arthur Pope von der *University of British Columbia* ein Verfahren zur Erkennung von Objekten in Grauwertbildern anhand von ansichtsbasierten Objektmodellen. In diesem Kapitel werden die drei Kernbereiche seiner Arbeit vorgestellt: Wissensrepräsentation, Lernverfahren und Erkennungsverfahren.

2.1 Wissensrepräsentation

A. Pope verwendet die ansichtsbasierte Art der Objektrepräsentation in einer probabilistischen Formulierung, d.h. das variierende Erscheinungsbild eines Objektes wird durch eine Wahrscheinlichkeitsverteilung über dem Raum der 2-dimensionalen Bilder beschrieben. Diese Verteilung gibt also für ein beliebiges 2-dimensionales Bild an, mit welcher Wahrscheinlichkeit es das Abbild des 3-dimensionalen Objektes ist.

Die Komplexität der Objekte, die Abbildungsvorschrift und die Umweltbedingungen, wie Beleuchtungsverhältnisse und Verdeckungen, machen diese Verteilung sehr komplex und nicht exakt berechenbar. Es kann jedoch angenommen werden, dass die Verteilung durch Beispiel-Ansichten (Trainingsbilder) hinreichend genau abgetastet werden kann, um eine brauchbare Näherung schätzen zu können. Die Repräsentation dieser Verteilung ist auf den folgenden zwei Ebenen organisiert (siehe Abbildung 2.1).

- **Grob**: Das ganze Spektrum der Objektansichten wird in eine endliche Menge von Bereichen unterteilt, die so genannten *Modellansichten*.

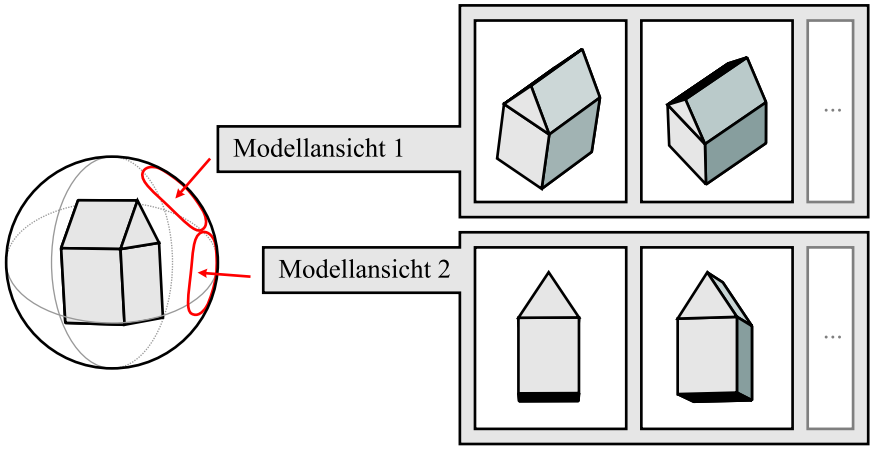

Abbildung 2.1: Die zwei Ebenen des ansichtsbasierten Objektmodells. Zwischen den Modellansichten variiert das Aussehen stark, innerhalb dieser nur schwach.

- **Fein**: Jede Modellansicht beschreibt eine Menge visuell ähnlicher Objektansichten. Es ist dabei nicht nötig, dass die Objektansichten einer Modellansicht zu ähnlichen Blickwinkeln gehören, sie müssen sich lediglich visuell ähnlich sein[1].

Bei der Erzeugung eines Modells aus Trainingsbildern werden ähnliche Objektansichten zu Modellansichten zusammengefasst. Dabei wird angestrebt, möglichst wenige Überlappungen zwischen Modellansichten zu erhalten und dennoch alle Trainings-Ansichten abzudecken.

Um das Objekt in einem Testbild zu erkennen, wird jede einzelne Modellansicht an das Testbild angepasst, d.h. es wird eine 2D-Ähnlichkeitstransformation berechnet, welche Modellansicht und Testbild optimal zur Deckung bringt. Ein Gütemaß für die Anpassung gibt an, mit welcher Wahrscheinlichkeit der gefundene Bildbereich der Modellansicht entspricht. Das Ergebnis der Erkennung ist die Modellansicht mit der größten ermittelten Wahrscheinlichkeit. Wie sich aus diesem 2-dimensionalen Erkennungsergebnis die 3-dimensionale Lage des Objektes bestimmen lässt, ist ein zentraler Beitrag dieser Arbeit und wird in Kapitel 5 beschrieben.

2.1.1 Repräsentation von Objektansichten und Bildern

Zweidimensionale Objektansichten und Kamerabilder werden durch Graphen beschrieben. Die Knoten dieses Graphen sind aus dem Bild extrahierbare lokale Merkmale, die Kanten bezeichnen deren Inklusionsbeziehungen. Eine Inklusionsbeziehung beschreibt,

[1]Man betrachte z.B. Objekte, welche punktsymmetrisch zu ihrem Mittelpunkt sind. Jede ihrer Objektansichten ist gleich der Gegenüberliegenden. Die zwei entsprechenden Blickwinkel sind jedoch konträr, also einander nicht ähnlich.

aus welchen Merkmalen ein höheres Merkmal besteht. Eine solche Beziehung besteht z.B. zwischen einer Ecke und den beiden Kurvensegmenten, durch die sie definiert wird.

Bildmerkmale werden wie folgt beschrieben:

- Merkmals-**Typ**,
- **Lage** im Bild[2]: 2D-Position, Orientierung und Größe,
- **Attribute**: numerische, von der Lage unabhängige Eigenschaften.

Die Erkennungsleistung des Systems hängt maágeblich von der Güte der verwendeten Merkmale ab. Es können lediglich die visuellen Aspekte eines Bildes verarbeitet werden, die hinreichend genau durch Merkmale repräsentiert sind. Der Typ eines Merkmals definiert, ob Inklusionsbeziehungen zu anderen Merkmalen bestehen, d.h. ob das Merkmal aus anderen zusammengesetzt ist und ob deshalb entsprechende Kanten im Bildgraphen existieren.

Das vollständige Repertoire der von A. Pope vorgeschlagenen höheren Merkmale ist in Anhang A aufgeführt. An dieser Stelle werden expemplarisch nur die Merkmale *L-Ecke* und *Kappe* erläutert (siehe Abbildung 2.2).

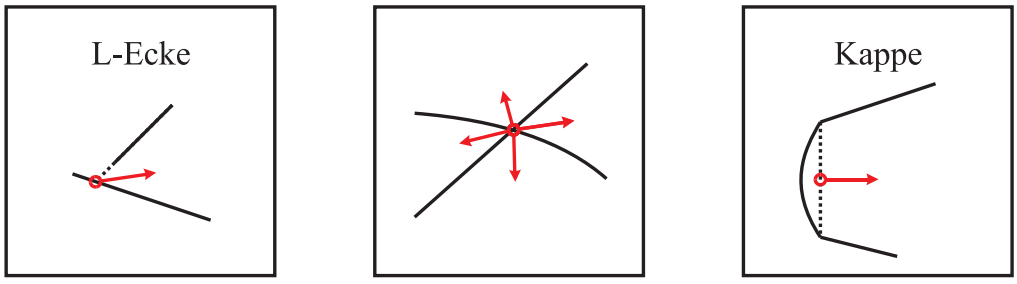

Abbildung 2.2: Links: Merkmal *L-Ecke*. Mitte: Die vier möglichen *L-Ecke*-Merkmale am Schnittpunkt zweier Kurven. Rechts: Merkmal *Kappe*.

Das Merkmal *L-Ecke*

Eine *L-Ecke* tritt an Stellen auf, an denen sich zwei Linien- oder Kurvensegmente mindestens näherungsweise schneiden (siehe Abbildung 2.2 links). Diese beiden Segmente werden *Arme* der Ecke genannt und sind selbst lokale Merkmale. Die Arme werden

[2]Im Folgenden beinhaltet der Begriff *Lage* die zweidimensionale Position, die Orientierung und die Größe eines Merkmals und unterscheidet sich damit von dem Begriff *Position*, welcher nur den Ort angibt.

als Teile der Ecke verstanden und sind deshalb im Bildgraphen durch Kanten mit ihr verbunden. Am Schnittpunkt zweier Segmente können bis zu vier *L-Ecke*-Merkmale auftreten (vgl. Abbildung 2.2 mitte).

Eine *L-Ecke* wird wie folgt beschrieben:

- **Typ**: *L-Ecke*,
- **Lage**: Position = Schnittpunkt der Arme[3], Orientierung = Richtung der Winkelhalbierenden zwischen den Armen und Größe = Summe der Armlängen,
- **Attribute**: Öffnungswinkel der Ecke. Dieser ist gleich dem Winkel zwischen den Armen.

Das Merkmal *Kappe*

Haben zwei *L-Ecken* einen Arm gemeinsam, so entsteht das höhere Merkmal *Kappe* (siehe Abbildung 2.2 rechts). Zwei Kanten im Graphen geben an, welche *L-Ecken* die Teile der *Kappe* sind.

Eine *Kappe* wird wie folgt beschrieben:

- **Typ**: *Kappe*,
- **Lage**: Position = Mittelpunkt der Verbindungsstrecke zwischen den beiden Eckpunkten, Orientierung = Winkel der Senkrechten zur Verbindungsstrecke und Größe = Länge der Verbindungsstrecke,
- **Attribute**: Öffnungswinkel der beiden Ecken.

2.1.2 Repräsentation von Modellansichten

Jede Ansicht eines Objektes ist ein zweidimensionales Bild und wird, wie im vorigen Abschnitt erläutert, als Graph aus lokalen Merkmalen repräsentiert. Da eine Modellansicht einen Bereich von Objektansichten abdeckt, muss ihre Beschreibung eine gewisse Variabilität zulassen. Dies wird durch eine probabilistische Erweiterung der oben beschriebenen Graphenrepräsentation erreicht. Auch für Modellansichten wird also ein Graph aus lokalen Merkmalen und deren Inklusionsbeziehungen zugrunde gelegt. Die Lage- und Attributvektoren werden jedoch nicht durch feste Werte, sondern durch

[3] bzw. Schnittpunkt der verlängerten Arme, falls diese sich nur näherungsweise schneiden.

Wahrscheinlichkeitsverteilungen beschrieben. Im Falle des Lagevektors (2D-Position, Orientierung und Größe) wird eine Normalverteilung angenommen. Dargestellt wird sie durch den Vektor der Mittelwerte und eine entsprechende Kovarianzmatrix.

Da jede Eigenschaft eines Merkmals, die von der Lage unabhängig ist, als Attribut verwendet werden kann, ist die Verteilung der Attributvektoren im Allgemeinen unbekannt. Aus diesem Grund wird die Verteilung empirisch, d.h. durch eine Menge von Beispielvektoren, dargestellt.

Modellmerkmale (d.h. Merkmale von Modellansichten) werden demnach wie folgt beschrieben:

- Merkmals-**Typ**,
- **Lage** im Bild: Vektor der Mittelwerte und Kovarianzmatrix zu einer Normalverteilung über dem Raum von 2D-Position, Orientierung und Größe,
- **Attribute**: Menge von Attributvektoren als Stichproben einer empirischen Verteilung.

Die Graphenstruktur entspricht der eines Bildgraphen mit den folgenden zwei Zusätzen:

- Es wird gespeichert, aus wievielen Trainingsbildern der Graph während des Lernvorgangs erzeugt worden ist.
- Zu jedem Knoten wird gespeichert, in wievielen Trainingsbildern das entsprechende Merkmal aufgetreten ist.

Aus diesen statistischen Angaben können Aussagen über die Relevanz von Merkmalen gewonnen werden.

2.1.3 Repräsentation von Szenenwissen

Objektunabhängiges Wissen über die verwendeten Merkmalstypen wird durch die so genannte *Umgebungsverteilung* repräsentiert. Sie wird aus einer Menge von Bildern gewonnen, die repräsentativ für spätere Testbilder ist. Aus ihr lässt sich die Wahrscheinlichkeit schätzen, mit der ein bestimmtes Merkmal unabhängig von seiner Lage in einem Testbild oder in einer Modellansicht auftritt.

Die Komponenten der Umgebungsverteilung sind:

- Die relativen Häufigkeiten der Merkmalstypen.

- Für jeden Merkmalstyp die Verteilung seiner Attributvektoren. Diese wird als empirische Verteilung durch eine Menge von Stichprobenvektoren dargestellt.

2.1.4 Notation

Für den formalen Umgang mit den in diesem Kapitel beschriebenen Konzepten wird die folgende Notation verwendet. Sie ist angelehnt an die Notation aus [Pope 1995], macht an einigen Stellen jedoch die Zusammenhänge klarer und behebt Mehrdeutigkeiten. Generell werden spitze Klammern für geordnete Mengen fester Größe (Tupel) verwendet. Großbuchstaben stehen für Matrizen, Relationen sowie Mengen und fettgedruckte Buchstaben für Vektoren. Für die sich entsprechenden Teile von Bild- und Modellgraphen werden dieselben Zeichen verwendet. Zur Unterscheidung sind alle Variablen, die zu Modellansichten gehören, mit einer Tilde versehen.

Bildmerkmal m_k:

$$m_k = \langle t_k, \mathbf{b}_k, \mathbf{a}_k \rangle \quad , \quad \begin{array}{l} t_k : \text{Typ des Merkmals,} \\ \mathbf{b}_k : \text{Lagevektor,} \\ \mathbf{a}_k : \text{Attributvektor.} \end{array}$$

Bild g:

$$g = \langle M, R \rangle \quad , \quad \begin{array}{l} M = \{m_1, \ldots, m_n\} : \text{Menge von Bildmerkmalen,} \\ R : \text{Relation über den Elementen von } M. \end{array}$$

Modellmerkmal $\tilde{m}_j$:

$$\tilde{m}_j = \langle \tilde{t}_j, \tilde{\mathbf{b}}_j, \tilde{\mathbf{B}}_j, \tilde{\mathcal{A}}_j, \tilde{h}_j \rangle \quad , \quad \begin{array}{l} \tilde{t}_j : \text{Typ des Merkmals,} \\ \tilde{\mathbf{b}}_j : \text{Mittlerer Lagevektor,} \\ \tilde{\mathbf{B}}_j : \text{Kovarianzmatrix des Lagevektors,} \\ \tilde{\mathcal{A}}_j : \text{Menge der Attributvektoren der Trainingsmerkmale,} \\ \tilde{h}_j : \text{Häufigkeit dieses Merkmals in der Trainingsmenge.} \end{array}$$

Modellansicht $\tilde{g}_i$:

$$\tilde{g}_i = \langle \tilde{M}, \tilde{R}, \tilde{l} \rangle \quad , \quad \begin{array}{l} \tilde{M} = \{\tilde{m}_1, \ldots, \tilde{m}_{\tilde{n}}\} : \text{Menge von Modellmerkmalen,} \\ \tilde{R} : \text{Relation über den Elementen von } \tilde{M}, \\ \tilde{l} : \text{Anzahl der Trainingsbilder, aus denen } \tilde{g}_i \text{ erzeugt wurde.} \end{array}$$

Objekt-Modell $\tilde{o}$:

$$\tilde{o} = \{\tilde{g}_1, \tilde{g}_2, \dots\} : \quad \text{Menge von Modellansichten.}$$

Zuordnungsrelation:

$$\begin{aligned} \tilde{m_j} \mapsto m_k &: \text{Modellmerkmal } \tilde{m_j} \text{ ist dem Bildmerkmal } m_k \text{ zugeordnet,} \\ \tilde{m_j} \mapsto \perp &: \text{Modellmerkmal } \tilde{m_j} \text{ ist keinem Bildmerkmal zugeordnet.} \end{aligned}$$

Zuordnung Z von Modell- und Bildmerkmalen:

$$\begin{aligned} Z &= \langle z_1, z_2\ , \dots z_{\tilde{n}} \rangle, \qquad z_j \in M\ , \\ z_j &= m_k\ , \qquad \text{falls} \quad \tilde{m_j} \mapsto m_k\ . \end{aligned}$$

2.2 Lernverfahren

Aufgabe des Lernverfahrens ist die Erzeugung einer Menge von Modellansichten, welche die wichtigen Aspekte des Objektaussehens in möglichst kompakter Form modellieren. Grundlage dazu bietet eine Stichprobe des Objektaussehens in Form einer Menge von Beispielansichten des Objektes.

Das Lernverfahren kombiniert zwei Vorgänge miteinander, welche die Menge der Beispielansichten schrittweise in eine Menge von Modellansichten überführen:

- Ein **Ballungsverfahren** fasst ähnliche Objektansichten zu Ballungen zusammen.
- Ein **Generalisierungsverfahren** verschmelzt die Objektansichten einer Ballung zu einer Modellansicht.

Die Gesamtkoordination wird dabei von dem Ballungsverfahren übernommen, welches wiederholt das Generalisierungsverfahren verwendet, um zwischen möglichen Ballungsalternativen zu entscheiden und diese zu realisieren.

2.2.1 Generalisierung über einer Menge von Objektansichten

Das Generalisierungsverfahren erzeugt eine Modellansicht aus einer Menge von Objektansichten. Dabei wird sowohl verlangt, dass die Modellansicht allgemein genug ist, um

alle Objektansichten zu beschreiben, als auch, dass durch sie jede einzelne Objektansicht genau genug beschrieben wird.

Die Generalisierung erfolgt schrittweise für die einzelnen Objektansichten, indem zuerst eine Modellansicht aus einer einzelnen Ansicht erzeugt wird und dann sukzessive weitere integriert werden. Die Integration einer weiteren Objektansicht erfolgt mithilfe des gleichen Verfahrens zur Anpassung zweier Merkmalsgraphen, wie es für die Erkennung einer Modellansicht in einem Testbild eingesetzt und in Abschnitt 2.3 vorgestellt wird. Dieses Anpassungsverfahren liefert eine Transformation T, welche die zu integrierende Objektansicht mit der bestehenden Modellansicht zur Deckung bringt sowie eine Zuordnung Z, die angibt, welche Merkmale der Objektansicht den Modellmerkmalen zugeordnet wurden. Anhand der Zuordnung Z wird die Modellansicht folgendermaßen verändert:

1. Ein Modellmerkmal $\tilde{m}_j$, welches durch Z zugeordnet wurde $(\tilde{m}_j \mapsto m_k)$ wird gemäß des zugeordneten Bildmerkmals m_k verändert. Der Lagevektors $\mathbf{b}_j$ wird angepasst, eine neue Attributausprägung wird zu $\tilde{\mathcal{A}}_j$ hinzugefügt und die Auftretenshäufigkeit $\tilde{h_j}$ des Merkmals angepasst.

2. Ein Bildmerkmal der Objektansicht, das nicht zugeordnet werden konnte, wird der Modellansicht hinzugefügt, falls

   - das Bildmerkmal direkt oder indirekt mit einem zugeordneten Bildmerkmal verbunden ist und
   - das Bildmerkmal den bestehenden Abstraktionsbeziehungen der Modellansicht nicht widerspricht. Dies wäre z.B. der Fall, wenn das Merkmal Teil eines zugeordneten höheren Merkmals ist, deren Teile in der Modellansicht schon anderweitig festgelegt sind.

3. Nicht zugeordnete Modellmerkmale werden entfernt, falls sie in zuwenig der bisher integrierten Objektansichten vorhanden waren.

2.2.2 Ballung von Objektansichten

Die Begriffe *Ballung* und *Modellansicht* werden hier synonym verwendet, da eine Modellansicht die durch das Generalisierungsverfahren erzeugte Repräsentation einer Menge von Objektansichten, also eine Ballung, darstellt. An die Menge der Modellansichten werden zwei Anforderungen gestellt:

1. **Einfachheit**: Die Anzahl der Modellansichten und jeweils deren Komplexität soll gering sein, um eine effiziente Objekterkennung zu ermöglichen.

2. **Exaktheit**: Jede Objektansicht muss durch eine der Modellansichten akkurat repräsentiert werden.

Die beiden geforderten Qualitäten werden durch das angewandte *Prinzip der kürzesten Beschreibung*[4] erreicht. Diese Prinzip besagt, dass das beste statistische Modell M für eine Stichprobe S jenes ist, mit dem die kürzeste Beschreibung für M und S zusammen erreicht werden kann. Im vorliegenden Fall ist die Stichprobe S die Menge der Objektansichten, welche durch das Modell M in Form der Menge von Modellansichten beschrieben werden sollen. Nach dem Prinzip der kürzesten Beschreibung wird eine Menge von Modellansichten gesucht, durch welche die Kodierungslänge der Einheit aus Modellansichten und Objektansichten minimal ist. Für jede Modellansicht und die ihr zugeordneten Objektansichten lässt sich eindeutig eine kürzeste Beschreibungslänge durch Unterscheidung der gemeinsamen und der nicht gemeinsamen Merkmale herleiten[5].

Das Ballungsverfahren arbeitet iterativ auf der Menge der zu lernenden Objektansichten. Zu Beginn wird die erste Objektansicht in eine Modellansicht überführt und diese als einzige bestehende Ballung angesehen. Im weiteren Verlauf werden weitere Objektansichten durch die folgenden Schritte bearbeitet:

1. Integriere die Objektansicht testweise in jede einzelne der bestehenden Modellansichten mittels des Generalisierungsverfahrens. Bestimme die resultierende Veränderung der Gesamtkodierungslänge für die Einheit aus Modellansichten und bisher betrachteten Objektansichten und speichere diese.

2. Berechne die Veränderung der Gesamtkodierungslänge für den Fall, dass die Objektansicht keiner bestehenden Ballung hinzugefügt sondern eine neue Ballung erzeugt wird.

3. Wähle aus den Ballungsalternativen der vorherigen Schritte diejenige mit der kürzesten Gesamtkodierungslänge aus und verwirkliche sie.

Nach Integration aller Objektansichten liegt das ansichtsbasierte Objektmodell in Form der entstandenen Menge von Modellansichten vor.

Durch die sukzessive Verarbeitung der einzelnen Objektansichten ist das Ballungsergebnis von deren Reihenfolge abhängig. Untersuchungen haben gezeigt, dass sich durch eine veränderte Reihenfolge beim Lernen deutlich unterschiedliche Modellansichten ergeben, welche aber dennoch zu vergleichbaren Erkennungsleistungen führen. Außerdem

[4] minimum description length principle (MDL) [Rissanen 1978].

[5] Das Kodierungsschema ist hier aus Platzgründen nicht aufgeführt und kann in [Pope 1995] eingesehen werden.

bleiben die wichtigsten Kennzahlen des Ballungsergebnisses wie die Anzahl der Modellansichten und die Verteilung der Ballungsgrößen trotz Umordnung der Trainingsmenge nahezu gleich.

2.3 Erkennungsverfahren

Sobald ein Objekt mit dem in Abschnitt 2.2 beschriebenen Lernverfahren in Form von Modellansichten repräsentiert ist, kann es in Testbildern erkannt werden. Dazu identifiziert das Erkennungsverfahren den Ausschnitt des Testbildes, der mit einer möglichst hohen Wahrscheinlichkeit einem Abbild des gesuchten Objektes entspricht. Durch die ansichtsbasierte Objektrepräsentation reduziert sich das Problem darauf, eine der zweidimensionalen Modellansichten im Testbild zu finden (siehe Abbildung 2.3).

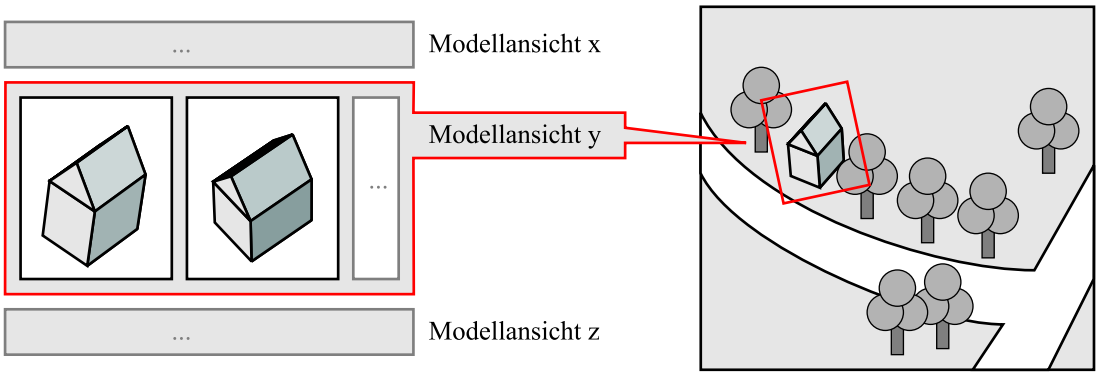

Abbildung 2.3: Erkennung einer Modellansicht in einem Testbild.

Im Folgenden wird beschrieben, wie eine einzelne Modellansicht in einem Testbild identifiziert werden kann und wie sich der Grad der Übereinstimmung messen lässt. Es wird angenommen, dass eine Modellansicht durch eine 2D-Ähnlichkeitstransformation (beschrieben durch Verschiebung, Skalierung und Rotation) mit dem Bildausschnitt zur Deckung gebracht werden kann. Dies entspricht der Annahme, dass es keine Rolle spielt, ob das Objektabbild sich in der Bildmitte oder den Randbereichen des Bildes befindet (da mit Ähnlichkeitstransformationen keine perspektivischen Verzerrungen modelliert werden können, siehe Abbildung 2.4). In Kapitel 6.1.1 wird gezeigt, dass diese vereinfachende Annahme in praxisrelevanten Szenen gerechtfertigt ist.

Das hier beschriebene Erkennungsverfahren bildet sukzessiv Paare aus Modell- und Bildmerkmalen. Mit diesen Paaren wird eine Transformation definiert, welche die Modellansicht und das Objektabbild zur Deckung bringt. Diese Transformation wird probabilistisch durch eine Gaußverteilung über dem Raum der Ähnlichkeitstransformationen modelliert.

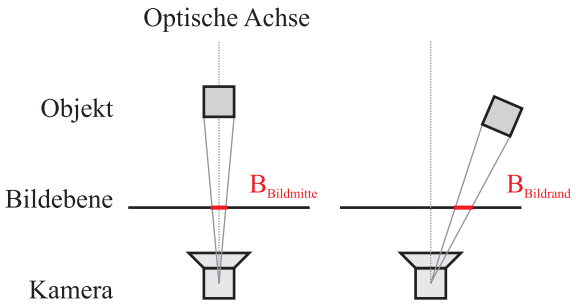

Abbildung 2.4: Voraussetzung an das Objektabbild. Das Abbild $B_{Bildmitte}$ eines Objektes in der Bildmitte ist näherungsweise gleich dem Abbild $B_{Bildrand}$ des Objektes in den Randbereichen.

2.3.1 Gütemaß für Merkmalspaarungen

Die Menge der möglichen Paare aus Modell- und Bildmerkmalen ist im Allgemeinen sehr groß. Um diese Menge effizient durchsuchen zu können, wird jedes Paar durch ein Gütemaß bewertet und dadurch eine vorteilhafte Suchreihenfolge definiert. Die Herleitung des Gütemaßes wird an dieser Stelle formal dargestellt, um dem Leser eine möglichst konkrete Vorstellung davon zu geben, auf welche Art der Erkennungsvorgang durch statistische Daten gesteuert wird.

Notation:

- Z : Zuordnung von Modell- und Bildmerkmalen (siehe Abschnitt 2.1.4),
- H : Hypothese, dass das Objekt im Bild zu sehen und entsprechend der Modellansicht zur Kamera orientiert ist,
- T : 2D-Ähnlichkeitstransformation, welche das Bild auf die Modellansicht abbildet.

Der Wert $\mathrm{P}(H \mid Z,T)$ gibt an, mit welcher Wahrscheinlichkeit eines der durch die Modellansicht repräsentierten Objektabbilder im Bild zu sehen ist, falls sich Modell- und Bildmerkmale gemäß Z zuordnen lassen und die Transformation T angibt, wo sich die Modellansicht im Bild befindet. Durch die Formel von Bayes ergibt sich

$$\mathrm{P}(H \mid Z,T) = \frac{\mathrm{P}(Z \mid T,H)\, P(T \mid H)}{P(Z \wedge T)} P(H) \ .$$

Um die Berechnungen der hochdimensionalen Wahrscheinlichkeitsverteilungen praktikabel zu machen, werden die folgenden Annahmen getroffen:

1. Die Merkmalspaare einer Zuordnung seien paarweise unabhängig voneinander.

Formal: $\mathrm{P}(Z) = \prod_j \mathrm{P}(\tilde{m}_j \mapsto z_j)$.

2. Zuordnungen und Transformationen seien unabhängig voneinander.
   Formal: $\mathrm{P}(Z \wedge T) = \mathrm{P}(Z)\,\mathrm{P}(T)$.

3. Alle Lagen einer Modellansicht im Testbild seien gleich wahrscheinlich.
   Formal: $\mathrm{P}(T \mid H) = \mathrm{P}(T)$.

Durch diese Annahmen lässt sich eine Näherung für die Wahrscheinlichkeit $\mathrm{P}(H \mid Z, T)$ angeben:

$$\mathrm{P}(H \mid Z, T) \;\approx\; P(H) \cdot \frac{\prod_j \mathrm{P}(\tilde{m}_j \mapsto z_j \mid T, H)}{\prod_j \mathrm{P}(\tilde{m}_j \mapsto z_j)} \;.$$

Ziel der Graphenanpassung ist es, eine Zuordnung Z und Transformation T zu finden, bezüglich der die Wahrscheinlichkeit für die Präsenz des Objektes $\mathrm{P}(H \mid Z, T)$ maximal ist. Aus diesem Grund definiert man das Gütemaß

$$\begin{aligned} g(Z,T) &:= \log \mathrm{P}(H \mid Z, T) \\ &\approx \log \mathrm{P}(H) + \sum_j \log \mathrm{P}(\tilde{m}_j \mapsto z_j \mid T, H) - \sum_j \log \mathrm{P}(\tilde{m}_j \mapsto z_j) \;, \end{aligned}$$

welches durch die Graphenanpassung maximiert werden soll. Die Teile dieses Maßes lassen sich aus statistischen Angaben schätzen, die während des Trainings der Modellansicht gewonnen wurden:

- $\mathrm{P}(H)$ berechnet sich aus dem Anteil der Trainingsbilder, die zur Bildung der Modellansicht beigetragen haben.

- $\mathrm{P}(\tilde{m}_j \mapsto z_j \mid T, H)$ ist die Wahrscheinlichkeit einer konkreten Merkmalspaarung. Ihre Berechnung wird im folgenden Abschnitt dargestellt.

- $\mathrm{P}(\tilde{m}_j \mapsto z_j)$ ist die objektunabhängige Wahrscheinlichkeit für die Zuordnung von $\tilde{m}_j$ und z_j. Diese kann aus einer objektunabhängigen Statistik - der Umgebungsverteilung - entnommen werden, die aus einer Menge von zufälligen Beispielbildern der Szene generiert wird (siehe Abschnitt 2.1.3).

2.3.2 Wahrscheinlichkeit einer Merkmalspaarung

Ein Modellmerkmal sollte nur dann einem Bildmerkmal zugeordnet werden, wenn sich die durch sie jeweils repräsentierten Bildstrukturen ähnlich sind. Da lokale Merkmale durch einen Lagevektor und einen davon unabhängigen Attributvektor beschrieben werden, hängt die Wahrscheinlichkeit einer Merkmalspaarung von der Kompatibilität dieser Vektoren ab. Demnach sollten zwei Merkmale einander zugeordnet werden, wenn

- ihre Attributvektoren einander ähnlich sind und
- der Lagevektor des Bildmerkmals nach Anwendung der Transformation T dem Lagevektor des Modellmerkmals ähnlich ist.

Kompatibilität von Lagevektoren

Die Lage eines Bildmerkmals wird durch einen 4-dimensionalen Vektor aus 2D-Position, Orientierung und Größe angegeben, die Lage eines Modellmerkmals durch eine Gaußverteilung über dem entsprechenden Raum. Die Transformation T, welche das Testbild auf die Modellansicht abbildet, wird ebenfalls durch eine 4-dimensionale Gaußverteilung beschrieben. T angewandt auf den Lagevektor des Bildmerkmals ergibt somit eine Gaußverteilung im Raum der Modellansicht, welche mit der Verteilung des Modellmerkmals zu vergleichen ist (siehe Abbildung 2.5).

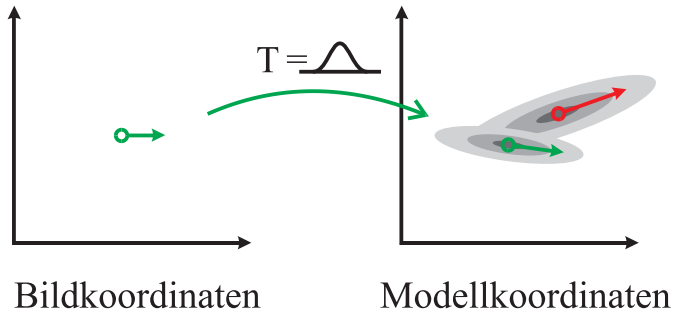

Abbildung 2.5: Die probabilistische Ähnlichkeitstransformation T überführt den Lagevektor eines Bildmerkmals (links, in grün) in eine Gaußverteilung im Raum der Modellansicht. Dort ist sie mit der Lageverteilung eines Modellmerkmals (rechts, in rot) zu vergleichen.

Dieser Vergleich geschieht durch folgende Integration:

$$\mathrm{P}(\tilde{\mathbf{b_j}} = T(\mathbf{b_k}) \mid z_j \neq \perp, T, H) = \int \mathrm{P}(\tilde{\mathbf{b_j}} = \mathbf{r}) \cdot \mathrm{P}(T(\mathbf{b_j}) = \mathbf{r})\, d\mathbf{r}$$

Das Integral läßt sich leicht bestimmen, da es über dem Produkt zweier Gaußverteilungen gebildet wird und somit selbst Gaußform besitzt.

Kompatibilität von Attributvektoren

Die Verteilung der Attribute eines Merkmals wird durch eine Menge von Stichprobenvektoren repräsentiert. In der Arbeit von A. Pope wird ein Verfahren aus [Silverman 1986] verwendet, um aus einer solchen empirischen Verteilung die Wahrscheinlichkeit von Vektoren zu schätzen. Dazu werden um die Stichproben spezielle Kernelfunktionen zentriert und deren Beiträge an der auszuwertenden Stelle summiert.

2.3.3 Probabilistische Korrespondenzsuche

Kern des Erkennungsverfahrens ist der Algorithmus, welcher Modellansicht und Testbild zur Deckung bringt und die zugehörige Ähnlichkeitstransformation schätzt. Ein bekanntes Verfahren für eine solche Aufgabe ist das in Abschnitt 1.3 kurz vorgestellte *Alignment*-Verfahren: wenige Merkmale werden zur Deckung gebracht und dadurch eine Transformation geschätzt. Ein zweiter Schritt verifiziert, ob die erhaltene Transformation mit den übrigen Merkmalen konsistent ist. Dieses Verfahren wurde von A. Pope zum so genannten *probabilistic alignment* erweitert, wobei sowohl die Merkmale als auch die Transformationen als unsicher modelliert werden.

Zu Beginn wird die Menge derjenigen Paare von Modell- und Bildmerkmalen gebildet, deren Lage als stabil extrahierbar angenommen wird[6]. Im Folgenden werden solche Merkmale *Initialmerkmale* und entsprechende Paare *Initialpaare* genannt. Die einzelnen Initialpaare werden durch das oben beschriebene Gütemaß bewertet und in absteigender Reihenfolge bearbeitet:

- Für jedes Initialpaar wird eine Transformation geschätzt.
- Durch weitere, zu dieser Transformation konsistente Paare wird die Transformation verfeinert, bis das Gütemaß nicht mehr verbessert werden kann.

So erhält man für jedes Initialpaar eine Zuordnung Z, eine Transformation T und deren Bewertung $g(Z,T)$. Mittels eines von der Bewertungsfunktion g unabhängigen Verifikationsmaßes, das im folgenden Abschnitt beschrieben wird, lässt sich bestimmen, welche der Zuordnungen tatsächlich eine Instanz des gesuchten Objektes identifizieren.

[6] Beispielsweise wird die Orientierung eines Merkmals *Ellipse* nicht als stabil angenommen, da deren Bestimmung durch die Merkmalsextraktion starken Schwankungen unterworfen ist. Im Extremfall eines Kreises ist die Orientierung einer *Ellipse* nicht definiert.

2.3.4 Verifikation

Das beschriebene Gütemaß $g(Z,T)$ für Zuordnungen ist gut für die Steuerung des Anpassungsvorgangs geeignet, jedoch schlecht für die Entscheidung, ob eine konkrete Zuordnung als Erkennungsergebnis akzeptiert werden sollte. Ein Grund hierfür ist die in der Herleitung gemachte Unabhängigkeitsannahme über die Zuordnung einzelner Merkmale. Würde man beispielsweise aus einer Zuordnung alle bis auf die niedrigsten Merkmale entfernen, so würde sich das Gütemaß deutlich verringern. Die visuelle Beschreibung der Objektansicht bliebe jedoch erhalten, da aus den niedrigsten Merkmalen alle anderen erzeugt worden sind.

A. Pope schlägt aus diesem Grund vor, lediglich die niedrigsten Merkmale, d.h. die Kurvensegmente, zu betrachten. Dazu werden die Anteile der Bildkurven aufsummiert, die nahe an Modellkurven liegen. Diese Summe wird durch die Gesamtlänge der Modellkurven geteilt und als Verifikationsmaß $V(Z,T)$ verwendet. Als Akzeptanzschranke wurde experimentell der Wert $V(Z,T) = 0{,}5$ ermittelt, was anschaulich bedeutet, dass mindestens 50% der Modellkurven im Bild überdeckt werden müssen.

Kapitel 3

Realisierung des ansichtsbasierten Lokalisierungssystems

In diesem Kapitel wird beschrieben, wie der ansichtsbasierte Erkennungsansatz zur räumlichen Lokalisierung von Objekten verwendet werden kann. Es stellt das im Rahmen dieser Arbeit implementierte System vor und macht die Schnittstellen zum Erkennungsverfahren von A. Pope deutlich. Das im vorigen Kapitel vorgestellte Objekterkennungsverfahren leistet die Erkennung und die Lokalisierung von Objektabbildern in Testbildern sowie das Lernen des dazu benötigten ansichtsbasierten Objektmodells. Ziel der vorliegenden Arbeit ist die Erweiterung des Ansatzes auf die räumliche Objektlokalisierung unter Nutzung vorhandener 3D-Formmodelle. Dazu werden primär zwei zusätzliche Vorgänge benötigt:

1. Die Gewinnung einer geeigneten Menge von Objektansichten für ein reales Objekt aus dem Formmodell.
2. Die Auswertung des zweidimensionalen Erkennungsergebnisses zur Schätzung der räumlichen Objektlage.

Im Folgenden wird ein Überblick über die Konzeption und Realisierung des Gesamtsystems gegeben. Die Herleitung der entwickelten Verfahren zur Gewinnung von Objektansichten und zur Schätzung der räumliche Objektlage werden in den anschließenden Kapiteln behandelt.

3.1 Einbettung in das Objektverfolgungssystem

Das in dieser Arbeit entwickelte Lokalisierungssystem soll am Fraunhofer Institut für Informations- und Datenverarbeitung (IITB) im Rahmen des Projekts MQube eingesetzt werden zur Initialisierung eines Objektverfolgungsprozesses, welcher die räumliche Lage von Objekten auf einer Modellbühne schätzt. In Abbildung 3.1 wird schematisch dargestellt, wie sich das Lokalisierungssystem in das Verfolgungssystem einbettet.

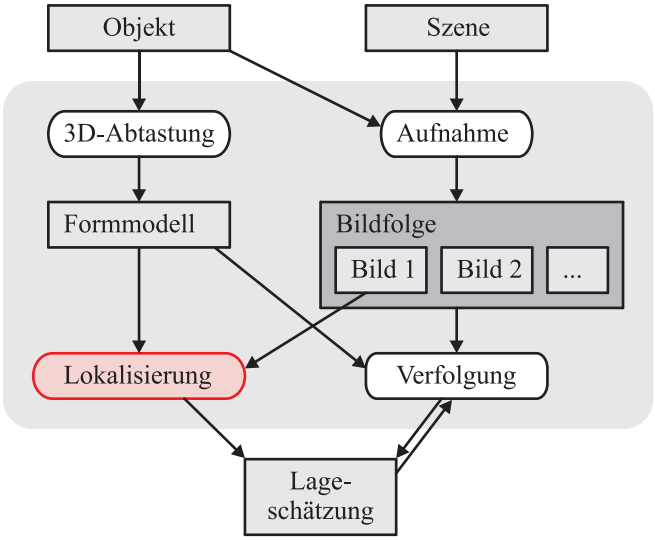

Abbildung 3.1: Die Einbettung des Lokalisierungssystems in das Objektverfolgungssystem. Daten werden in dieser Darstellung durch Rechtecke begrenzt, Verarbeitungseinheiten durch abgerundete Rechtecke.

Eingaben des Verfolgungsystems sind ein Formmodell des Objektes sowie eine Bildfolge der Szene in der das Objekt bewegt wird. Das Formmodell wird in einem Vorverarbeitungsschritt aus dem realen Objekt generiert, ist eine wichtige Grundlage für die Verfolgung und soll auch als Grundlage für die Lokalisierung dienen. Diese Modellierung eines Objektes kann für einfache Objekte manuell geschehen. Im Projekt MQube steht aber auch ein 3D-Scanner zur Verfügung, um automatisch die Form und Struktur von Objekten zu erfassen. Das Lokalisierungssystem leistet die Initialschätzung der Objektlage aufgrund eines einzelnen Bildes der Bildsequenz. Die Verfolgungskomponente verfeinert diese Anfangslage und passt sie an die weiteren Szenenbilder an. Der lokal arbeitende Verfolgungsprozess benötigt dazu eine hinreichend genaue Lageschätzung, um das Objekt erfassen und verfolgen zu können.

3.2 Systemstruktur

In Abbildung 3.2 ist die Struktur des realisierten Lokalisierungssystems mit seinen Verarbeitungseinheiten und Datenrepräsentationen schematisch dargestellt.

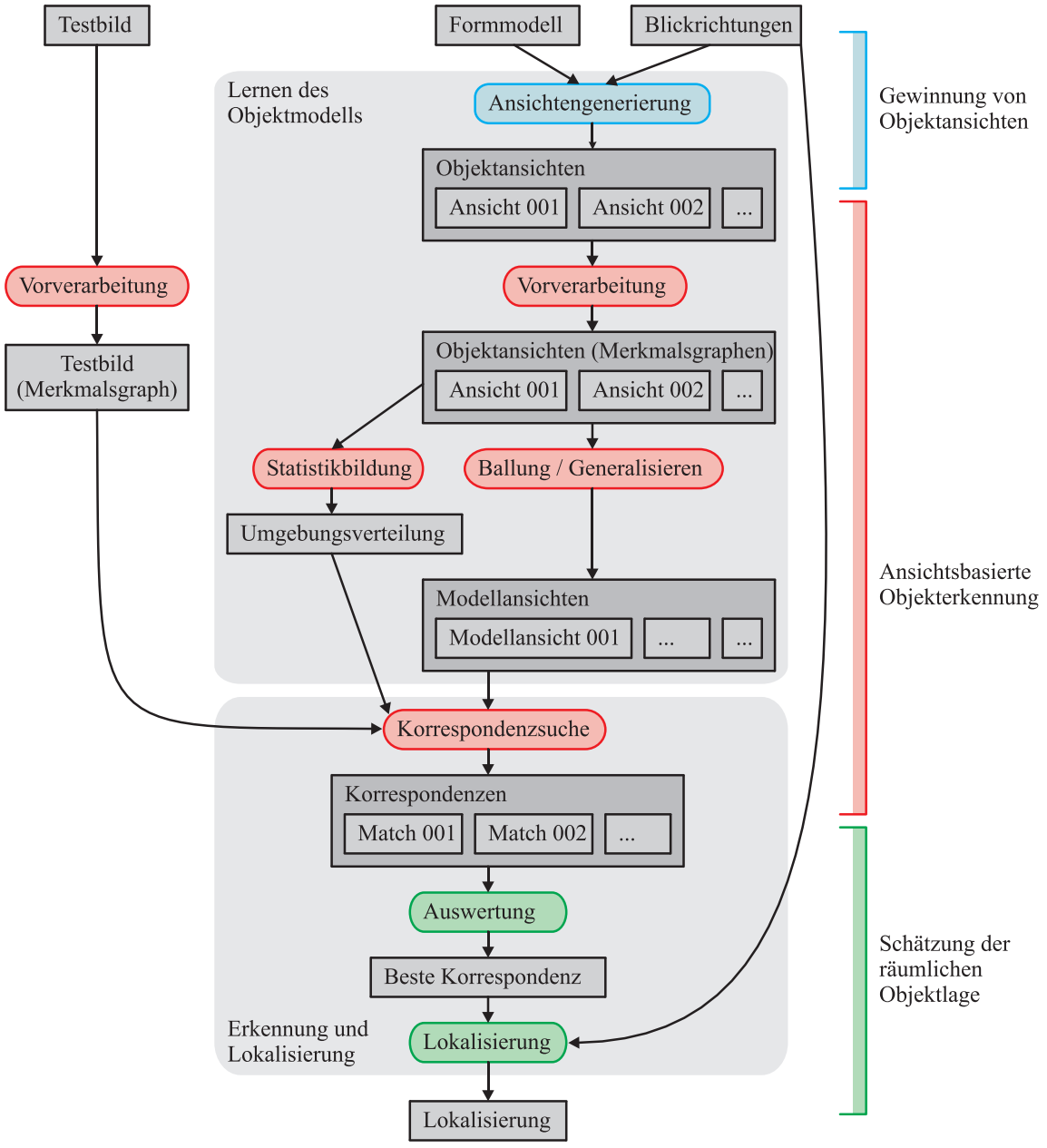

Abbildung 3.2: Schematische Darstellung des Lokalisierungssystems, seiner Verarbeitungseinheiten und der wichtigsten Datenrepräsentationen. Daten werden in der Darstellung durch Rechtecke begrenzt, Verarbeitungseinheiten durch abgerundete Rechtecke.

Im Strukturdiagramm sind die im Folgenden näher erläuterten drei Verarbeitungsstufen farblich voneinander abgegrenzt. Anhand dieser Abgrenzung wird deutlich, auf welchen Teilen in dieser Arbeit aufgebaut werden konnte (rot dargestellt), welche Teile neu entwickelt wurden (blau und grün dargestellt) und wie die Schnittstellen dazwischen aussehen.

Gewinnung von Objektansichten

Aufgabe dieser Verarbeitungsstufe ist die Generierung der Ansichtsrepräsentation aus dem zur Objektverfolgung eingesetzten Formmodell. Die Umsetzung wird in Kapitel 4 ausführlich beschrieben. Es wird untersucht, welche Auswirkungen die gewählte Parametrisierung auf den anschließenden Lernprozess und die Erkennung hat.

Ansichtsbasierte Objekterkennung

Diese Verarbeitungsstufe umfaßt die Verarbeitungseinheiten des Verfahrens von A. Pope sowie die Beziehungen dazwischen. Wie in Kapitel 2 bereits dargestellt, umfasst deren Funktionalität zum einen das Lernen eines ansichtsbasierten Objektmodells und zum anderen dessen Nutzung zur Objekterkennung.

Schätzung der räumlichen Objektlage

Das Ergebnis der ansichtsbasierten Objekterkennung liegt in Form einer Menge von Anpassungshypothesen vor, welche durch ein Verifikationsmaß bewertet wurden. Aus den bei diesen Hypothesen ermittelten 2D-Transformationen wird in dieser Verarbeitungseinheit unter Verwendung der Informationen aus dem Lernvorgang auf die räumliche 6D-Lage des Objektes geschlossen. Die Herleitung dieses Schätzverfahrens wird in Kapitel 5 gegeben.

3.3 Praktische Umsetzung

A. Pope, der Autor des in Kapitel 2 beschriebenen Erkennungsverfahrens, war so freundlich, die Implementierung seiner Verfahren zur Verfügung zu stellen. So war es möglich, im Rahmen dieser Arbeit ein umfangreiches System fertigzustellen, das in der Praxis einsetzbar ist. Der gewählte Ansatz, das System als Zusammenschluss von eigenständigen Programmen zu entwerfen, entspricht dem Vorgehen des Frameworks *Vista* (siehe [Pope & Lowe 1994]) in der Implementierung von A. Pope, welches für einfache Vorverarbeitungsschritte benutzt wird und ein flexibles Datenformat für die Kommunikation der Funktionseinheiten bietet.

Die ergänzten Verarbeitungseinheiten wurden jeweils als eigenständige Programme implementiert, welche durch eine einfache Skriptsprache koordiniert werden. Im Ver-

gleich zu einer möglichen Implementierung des Systems als einzelnes Programm (z.B. nach dem Prinzip der objektorientierten Programmierung) bietet die Zerlegung in eigenständige Einheiten mehrere Vorteile:

- Die Zwischenergebnisse liegen explizit in Form von Dateien vor. Sie können leicht analysiert oder zu Testzwecken verändert werden.
- Die Funktionseinheiten können leicht unabhängig voneinander verwendet und weiterentwickelt werden. So wäre beispielsweise der Einsatz des Programms zur Extraktion lokaler Merkmale in anderen Anwendungen denkbar.
- Die Verarbeitungsstufen können leicht parallelisiert werden. Diese Möglichkeit bietet sich unter anderem bei den zeitaufwändigen Verfahren zur künstlichen Ansichtengenerierung und der Graphenanpassung an.

Das Gesamtsystem ist auf dem Betriebssystem Linux entwickelt worden und mit dem von Dr. T. Müller im Rahmen des Projekts MQube am Fraunhofer Institut IITB in Karlsruhe entwickelten Objektverfolgungssystem kompatibel. Das Framework *Vista* sowie die Implementierung des Erkennungsverfahrens aus Kapitel 2 verwenden ausschließlich die die Programmiersprache C. Bis zur Einsatzfähigkeit dieser Komponenten war ein erheblicher Aufwand zu leisten, da die insgesamt 654 Einzelmodule und 98 Programme zur Benutzerinteraktion zu einem großen Teil aus unveröffentlichten Forschungsimplementierungen stammten und in teilweise inkompatiblen Versionen vorlagen. Einige dieser Versionen waren ursprünglich auf das Betriebssystem Solaris ausgelegt und mußten auf das Betriebsystem Linux portiert werden.

Für die Komponente zur Ansichtengenerierung (siehe Abbildung 3.2) kam die Szenenbeschreibungssprache des Raytracers *Povray*[1] zum Einsatz sowie die Programmiersprache C++ für ein Programm zur Berechnung einer Menge von gleichverteilten Blickwinkeln auf ein Objekt. Die Komponente zur Schätzung der räumlichen Objektlage wurde in C++ entwickelt und steht ebenfalls in Form eines eigenständigen Programms zur Verfügung.

[1] siehe www.povray.org

Kapitel 4

Gewinnung von Objektansichten

Um ein ansichtsbasiertes Objektmodell lernen zu können, wird eine Menge von Beispielansichten benötigt. Diese Menge sollte eine repräsentative Stichprobe der möglichen Objekterscheinungen darstellen und dabei Veränderungen der räumlichen Lage, der Beleuchtungsverhältnisse und anderer Umweltbedingungen berücksichtigen.

4.1 Methoden zur Gewinnung von Objektansichten

Objektansichten lassen sich folgendermaßen gewinnen:

1. **Abbildung des realen Objektes**: Mittels einer Kamera wird das reale Objekt aus verschiedenen Blickwinkeln aufgenommen.
2. **Erzeugung künstlicher Objektansichten**: Aus einem dreidimensionalen Formmodell werden künstliche Abbilder erzeugt.
3. **Formrekonstruktion und Erzeugung künstlicher Ansichten**: Als Kombination der ersten beiden Ansätze wird hier das reale Objekt mittels photometrischer Verfahren in ein dreidimensionales Formmodell überführt und daraus eine Menge künstlicher Abbilder erzeugt.

Im Folgenden werden die drei Arten der Ansichtengewinnung näher beschrieben und einander gegenübergestellt.

4.1.1 Abbildung des realen Objektes

Zur systematischen Erfassung realer Objekte lässt sich eine Versuchsanordnung mit Drehtisch und höhenverstellbarer Kamera verwenden (siehe Abbildung 4.1).

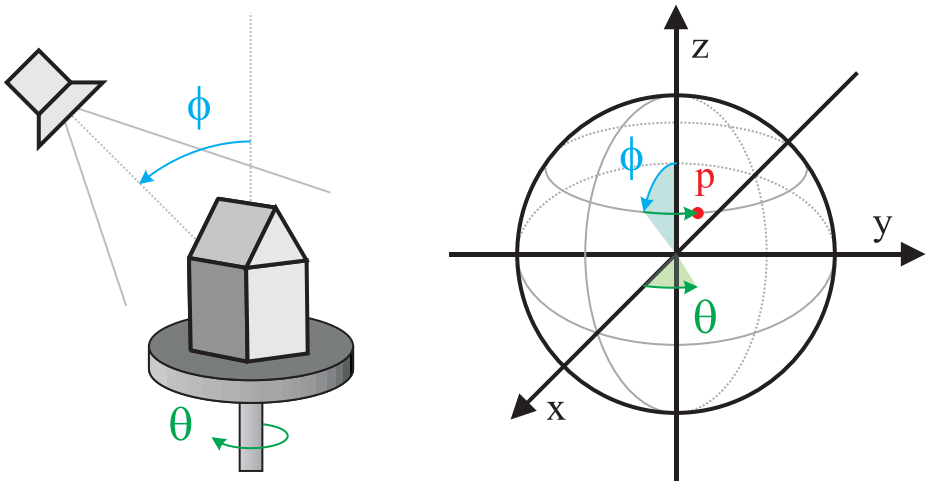

Abbildung 4.1: Versuchsanordnung zur Erfassung realer Objekte. Links: Eine im Neigungswinkel verstellbare Kamera ist auf das drehbare Objekt gerichtet. Rechts: Die Parameter Neigungswinkel ϕ und Drehwinkel θ definieren einen Punkt p auf einer Kugeloberfläche.

Eine Ansicht, bzw. der zugehörige Blickwinkel, wird durch folgende zwei Parameter charakterisiert[1]:

- der *Polarwinkel* ϕ: Neigungswinkel der Kamera,
- der *Azimuthwinkel* θ: Drehwinkel des Objektes um die vertikale Achse.

Die beiden Parameter ϕ und θ definieren den Blickwinkel auf das Objekt eindeutig. Eine Drehung der Kamera um deren Blickrichtung würde lediglich in einer gedrehten aber sonst identischen Ansicht resultieren, was für das Erkennungsverfahren unerheblich ist.

Die Verwendung von realen Objekten zur Bildung der Trainingsmengen birgt den großen Vorteil in sich, dass kein explizites Wissen über die dreidimensionale Struktur der Objekte benötigt wird. Zudem sind sich Trainings- und Testbilder strukturell weitestgehend ähnlich, da sie auf dieselbe Art aufgenommen werden und mit denselben Verfahren verarbeitet werden können.

[1] Auf einer Kugel wird der Wert $90° - \phi$ auch *Breitengrad* genannt und θ mit *Längengrad* bezeichnet.

4.1.2 Erzeugung künstlicher Objektansichten

Die Abbildung eines Objektes kann, sofern dessen Form und Oberflächeneigenschaften bekannt sind, simuliert werden. Ein einfacher Ansatz hierzu ist, die Kanten eines polyedrischen Objektmodells auf eine Ebene zu projizieren, um eine bestimmte Darstellung des Objektaussehens zu erhalten. Werden Oberflächeneigenschaften wie Farbe und Textur hinzugenommen, so erhöht sich die Realitätsnähe des Ergebnisses.

Ein dem vorzuziehender, komplexerer Ansatz zur Erzeugung künstlicher Bilder, der heute weitverbreitet ist und eindrucksvolle Ergebnisse liefert, folgt dem Prinzip der *Strahlenrückverfolgung*[2]. Solche Verfahren nutzen ein Prinzip der Strahlenoptik aus, nach dem Lichtquelle und -senke stets ausgetauscht werden können. Für jeden Bildpunkt einer virtuellen Bildebene wird ein Lichtstrahl durch die Szene *zurückverfolgt*, bis sein Ursprung in einer Lichtquelle gefunden wird oder er die modellierte Szene in den freien Raum verlässt. Moderne Verfahren berechnen auf diese Art physikalisch korrekte Schatten, Spiegelungen, Transparenzen und Brechungen.

Für die ansichtsbasierte Objekterkennung und -lokalisierung bietet die Erzeugung künstlicher Objektansichten folgende Vorteile:

- Durch den geringen Zeitaufwand für die Gewinnung einzelner Ansichten kann ein großes Spektrum an Bildmaterial erzeugt werden. Beleuchtungsänderungen, Störungen und andere Umwelteinflüsse können modelliert und zufällig variiert werden. Dadurch lässt sich, bei niedrigerem Aufwand, eine deutlich größere Trainingsmenge als bei realen Aufnahmen erreichen.

- Die Parameter des Abbildungsverfahrens können so eingestellt werden, dass irrelevante Bildmerkmale abgeschwächt oder entfernt werden. So ist es z.B. möglich, bei Bedarf nur die Objekt-Silhouette zu erzeugen und daraus einfachere Modelle zu lernen.

- Es können beliebige Blickwinkel auf das Objekt realisiert werden. Beim in 4.1.1 beschriebenen physikalischen Versuchsaufbau kann dagegen durch die Auflageflächen des Objektes und Einschränkungen der Kamerabeweglichkeit stets nur ein Teil der Objektansichten aufgenommen werden.

- Es besteht ein präziser Zusammenhang zwischen den Blickwinkel-Parametern ϕ, θ und der resultierenden Objektansicht. Bei der physikalischen Objekterfassung sind dagegen Meßfehler (in der Lage des Objektes auf dem Drehtisch, im Drehwinkel des Tisches und im Neigungswinkel der Kamera) zu berücksichtigen.

[2] Im Englischen: *Raytracing*

4.1.3 Formrekonstruktion und Erzeugung künstlicher Ansichten

Als Kombination der beiden genannten Vorgehensweisen ist der Ansatz zu verstehen, ein reales Objekt dreidimensional zu erfassen, ein entsprechendes Formmodell zu berechnen und auf dieser Grundlage künstliche Ansichten zu generieren. In dieser Arbeit wird gezeigt, dass der Ansatz praktikabel ist und durch ihn auch komplexe Objekte gelernt und lokalisiert werden können.

Um Formmodelle aus realen Objekten zu erstellen, wurde der optische 3D-Scanner *DigiScan 2000* der RSI GmbH verwendet. Die dreidimensionale Erfassung läuft in folgenden Schritten ab:

1. Das Objekt wird auf einem Drehtisch plaziert.
2. Ein Projektor bildet ein optisches Muster auf die Objektoberfläche ab.
3. Eine Kamera nimmt das, durch die Oberfläche entsprechend verzerrte, Muster auf.
4. Die Auswertungssoftware erstellt aus den aufgenommenen Mustern eine dreidimensionale Punktwolke und ergänzt diese zu einem Formmodell aus Dreiecksflächen.

Das so entstandene Formmodell kann direkt zur Generierung künstlicher Objektansichten herangezogen werden.

4.2 Gleichmässige Verteilung von Blickwinkeln

Unabhängig von der Art der Datengewinnung stellt sich die Frage, wieviele und welche Objektansichten als Trainingsmenge verwendet werden sollten. Da das Objekt a-priori in Form und Aussehen unbekannt ist, sollte es möglichst gleichmässig von allen Seiten erfasst werden. Wie diese Gleichmäßigkeit erreicht werden kann, wird im Folgenden beschrieben. Die optimale Größe der Trainingsmenge wird in Abschnitt 6.2.5 experimentell bestimmt.

Jeder Blickrichtung der Kamera auf das Objekt kann eindeutig ein Punkt auf der Einheitskugel zugeordnet werden[3] Es stellt sich also das Problem, eine bestimmte Anzahl

[3]Im Folgenden ist mit *Kamera* auch eine virtuelle Kamera im Kontext der künstlichen Bilderzeugung gemeint. Im Englischen wird die Einheitskugel in diesem Zusammenhang auch *view sphere* genannt.

von Punkten gleichmässig auf der Oberfläche der Einheitskugel zu verteilen (siehe Abbildung 4.2).

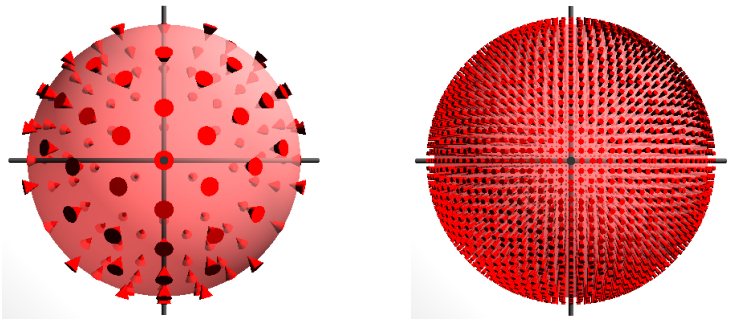

Abbildung 4.2: Gleichmäßige Verteilung von Punkten auf der Einheitskugel. Jeder Punkt auf der Einheitskugel entspricht einer Blickrichtung auf das Objekt. Links: 130 Punkte, rechts: 4098 Punkte.

Jeder Punkt auf der Kugeloberfläche sei durch seinen Ortsvektor beschrieben (siehe Abbildung 4.3). Ein solcher Vektor wird im Folgenden *Blickrichtungsvektor* genannt. Die *Nachbarn* eines Vektors seien die Vektoren, die zu ihm den kleinsten Winkelabstand besitzen. Für eine gleichmässige Verteilung von Blickrichtungsvektoren gilt die Forderung:

Der Winkel zwischen allen benachbarten Blickrichtungsvektoren ist gleich.

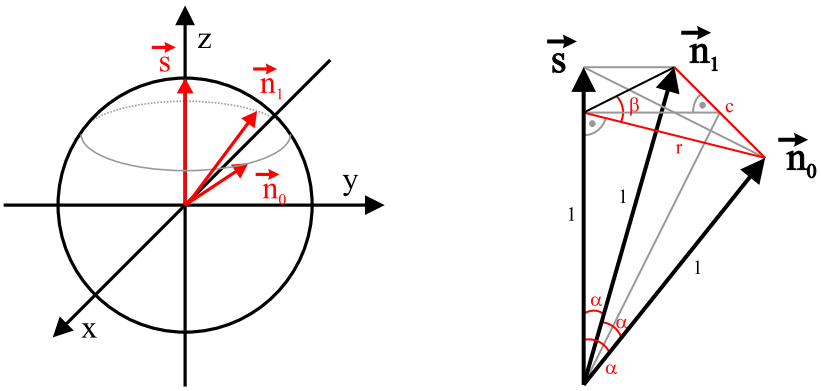

Abbildung 4.3: Gleichverteilung von Blickrichtungsvektoren. Dargestellt ist der Blickrichtungsvektor $\mathbf{s}$ und zwei seiner Nachbarn $\mathbf{n}_0$ und $\mathbf{n}_1$. Man beachte, dass zwischen je zwei der drei Vektoren der Winkel α besteht und dadurch β eindeutig definiert ist.

Seien $\mathbf{s}$ ein Blickrichtungsvektor, $\mathbf{n}_0$ und $\mathbf{n}_1$ zwei seiner Nachbarn, die selbst benachbart sind, und α der konstante Winkel zwischen jeweils zwei Vektoren. Für den Winkel β gelte der Zusammenhang aus Abbildung 4.3. β lässt sich wie folgt berechnen:

$$\begin{aligned} r &:= \sin\alpha \ , \\ c &:= 2 \cdot \sin\frac{\alpha}{2} \ , \\ \beta &= 2 \cdot \sin^{-1}\left(\frac{c}{2r}\right) \ . \end{aligned}$$

Aus dem Winkel β lässt sich nun bestimmen, wieviele benachbarte Blickrichtungsvektoren $\mathbf{s}$ besitzt:

$$\begin{aligned} k \cdot \beta &= 2\pi \quad (\text{die } k \text{ Nachbarn decken zusammen } 360^\circ \text{ ab}) \\ \Longrightarrow \quad k &= \frac{\pi}{\sin^{-1}\left(\frac{\sin(\frac{\alpha}{2})}{\sin\alpha}\right)} \ . \end{aligned}$$

Der Zusammenhang zwischen dem Winkel α und der Zahl der Nachbarn k ist in Abbildung 4.4 aufgetragen. Da nicht-ganzzahlige k ausgeschlossen werden müssen, erhält man folgende drei Lösungen für die gleichmässige Verteilung von Blickrichtungsvektoren:

1. $k = 3 \implies \alpha \approx 109{,}47^\circ$
2. $k = 4 \implies \alpha = 90^\circ$
3. $k = 5 \implies \alpha \approx 63{,}43^\circ$

Die Lösungen entsprechen drei der fünf platonischen Körper (siehe Abbildung 4.5): *Tetraeder* (3 Nachbarn), *Oktaeder* (4 Nachbarn) und *Ikosaeder* (5 Nachbarn). Die platonischen Körper *Würfel* und *Dodekaeder* werden nicht als Lösung erhalten, da ihre Oberflächenstücke vier- bzw. fünfeckig sind und die Nachbarschaft zwischen Vektoren hier jedoch so definiert wurde, dass maximal dreieckige Oberflächen zwischen den Vektoren entstehen können. Würfel und Dodekaeder sind als Lösungen nicht wünschenswert, da lediglich die Ortsvektoren der Mittelpunkte ihrer Oberflächen gleichverteilt sind (und diese Verteilungen schon durch Oktaeder und Ikosaeder gegeben sind), nicht aber die Ortsvektoren der Eckpunkte.

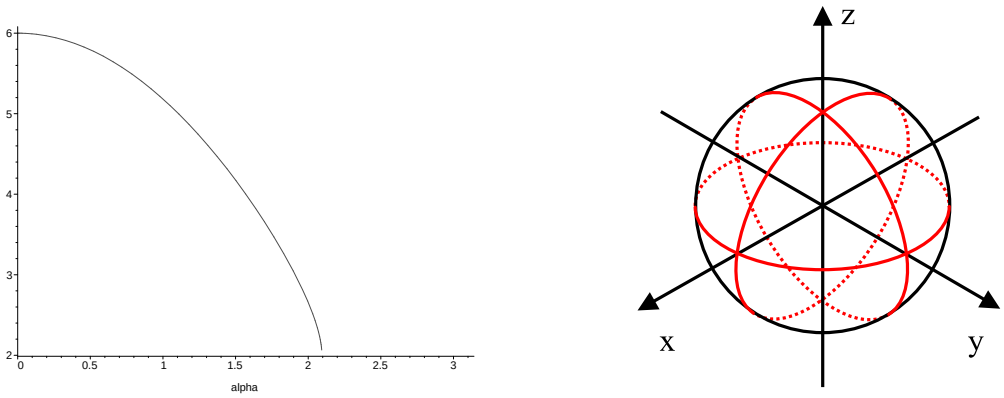

Abbildung 4.4: Zahl der Nachbarn in Abhängigkeit vom eingeschlossenen Winkel. Links: Zusammenhang zwischen dem Winkel α (Angabe im Bogenmaß) und der Anzahl k an Nachbarn von $\mathbf{s}$ (nach oben aufgetragen). Rechts: Ergebnis für $k = 4$: Jeder Knoten des roten Graphen repräsentiert eine Blickrichtung, die Kanten geben die Nachbarschaftsbeziehungen an. Der Winkel $\alpha = 90°$ zwischen den Blickrichtungen ist hier besonders leicht abzulesen.

Abbildung 4.5: Drei der platonischen Körper. Von links nach rechts: Tetraeder, Oktaeder, Ikosaeder.

4.3 Unterteilung sphärischer Dreiecksnetze

Durch die drei ganzzahligen Lösungen für k lassen sich Mengen aus 4, 6 oder 12 gleichverteilten Blickrichtungen angeben (aus den Ecken von Tetraeder, Oktaeder oder Ikosaeder). Diese Zahlen sind jedoch um Größenordnungen zu klein, um komplexe Objekte genau genug erfassen zu können. Eine größere Menge an Blickrichtungen, die zumindest näherungsweise gleichverteilt ist, lässt sich durch Weiterverarbeitung der drei Lösungen erreichen.

Die polyedrischen Körper zu den oben genannten Lösungen liegen in Form von sphäri-

schen Dreiecksnetzen vor (siehe Abbildung 4.4, rechts), d.h. Netzen auf einer Kugeloberfläche, deren Flächen genau drei Begrenzungskanten besitzen. Im Folgenden werden zwei Algorithmen vorgestellt, um sphärische Dreiecksnetze zu unterteilen und dadurch eine feinere Abtastung zu ermöglichen.

4.3.1 Unterteilung über die Seitenmitten

Der erste Algorithmus ersetzt jedes Dreieck des Netzes durch vier kleinere, indem er drei neue Punkte, jeweils in der Mitte der Seiten, einfügt (siehe Abbildung 4.6).

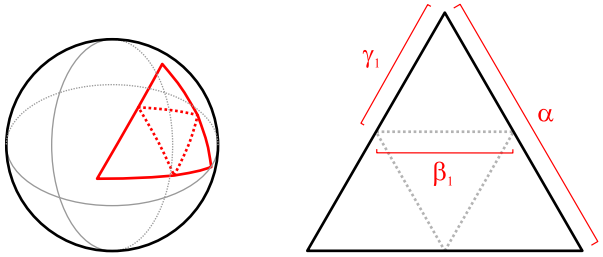

Abbildung 4.6: Unterteilung über die Seitenmitten. Links: Jedes sphärische Dreieck des Netzes wird in vier kleinere unterteilt. Rechts: Projektion des sphärischen Dreiecks auf die Ebene. Man beachte, dass der Bogen β_1 auf der Kugeloberfläche länger ist als der Bogen γ_1, obwohl dies in der dargestellten Projektion nicht der Fall ist.

Unterteilt man ein *ebenes*, gleichseitiges Dreieck nach dieser Vorschrift, so erhält man vier ebenfalls gleichseitige Dreiecke. Bei einem sphärischen Dreieck ist dies nicht der Fall. Hier entstehen vier Dreiecke durch die Unterteilung, deren Seitenlängen nicht exakt gleich lang sind. Jede Seite eines sphärischen Dreiecks ist auf natürliche Weise einem Winkel zugeordnet: dem Winkel zwischen den Ortsvektoren ihrer Eckpunkte. Da sich das sphärische Dreieck auf einer Einheitskugel befindet, ist die Länge der Seite gleich ihrem zugeordneten Winkel im Bogenmaß. Aus diesem Grund werden Winkel und Seitenlängen sphärischer Dreiecke im Folgenden synonym verwendet und mit denselben Symbolen bezeichnet. Die Winkel der Dreiecke, die durch das Unterteilungsverfahren entstehen, lassen sich wie folgt berechnen:

$$\begin{aligned} \beta_1 &= 2 \cdot \sin^{-1}\left(\frac{1}{2}\tan\left(\frac{\alpha}{2}\right)\right) \\ \gamma_1 &= \frac{\alpha}{2} \\ (\forall\alpha &: \beta_1 > \gamma_1) \end{aligned}$$

4.3.2 Unterteilung über die Schwerpunkte

Eine weitere Möglichkeit der Unterteilung besteht darin, die Schwerpunkte der Dreiecke jeweils als neuen Punkt einzufügen und aus der neuen Punktmenge ein Dreiecksnetz zu bilden (siehe Abbildung 4.7). Der *Schwerpunkt* eines Dreiecks ist eindeutig durch den Schnittpunkt der Seitenhalbierenden bestimmt.

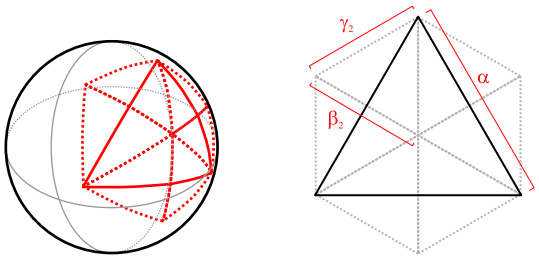

Abbildung 4.7: Unterteilung über die Schwerpunkte. Links: Der Schwerpunkt jedes Dreiecks wird als neuer Punkt eingefügt und die resultierende Punktmenge zu einem neuen Dreiecksnetz verbunden. Rechts: Projektion des sphärischen Dreiecks auf die Ebene. Man beachte, dass der Bogen β_2 auf der Kugeloberfläche länger ist als der Bogen γ_2, obwohl dies in der dargestellten Projektion nicht der Fall ist.

Auch hier entstehen nur näherungsweise gleichseitige Dreiecke, deren zugehörige Winkel sich berechnen lassen:

$$\begin{aligned} \beta_2 &= 2 \cdot \sin^{-1}\left(\frac{1}{\sqrt{3}} \tan\left(\frac{\alpha}{2}\right)\right) \\ \gamma_2 &= \sin^{-1}\left(\frac{1}{\sqrt{3}} \sin\left(\frac{\alpha}{2}\right)\right) \\ (\forall \alpha &: \beta_2 > \gamma_2) \end{aligned}$$

4.3.3 Vergleich und Kombination der Unterteilungsverfahren

Zum Vergleich der beiden Unterteilungsverfahren betrachte man die Verhältnisse $\frac{\beta_1}{\gamma_1}$ und $\frac{\beta_2}{\gamma_2}$. Da $\frac{\beta_2}{\gamma_2}$ für alle α kleiner als $\frac{\beta_1}{\gamma_1}$ ist (vgl. Abbildung 4.8), folgt aus der Unterteilung über die Schwerpunkte eine bessere Gleichverteilung der Punkte. Da beide Verhältnisse jedoch nahe am Idealverhältnis 1,0 liegen und bei sinkendem α gegen diesen Wert konvergieren, liefern beide Verfahren gute Ergebnisse.

Die Unterteilung über die Seitenmitten vermehrt die Anzahl der Dreiecke um den Faktor 4, die Unterteilung über die Schwerpunkte dagegen nur um den Faktor 3. In der

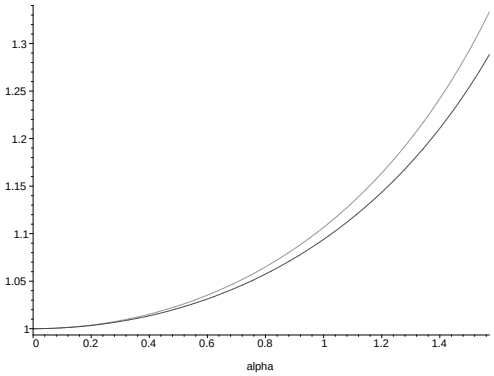

Abbildung 4.8: Gegenüberstellung der Winkelverhältnisse der zwei Unterteilungsalgorithmen. Für jedes α (Angabe im Bogenmaß) ist das Verhältnis $\frac{\beta_1}{\gamma_1}$ in grau und $\frac{\beta_2}{\gamma_2}$ in schwarz aufgetragen.

Anwendung liegt es daher nahe, die beiden Unterteilungsverfahren zu kombinieren, um möglichst nahe an eine gewünschte Anzahl von Trainingsansichten zu gelangen. Welche Konfigurationen ein Dreiecksnetz durch sukzessive Anwendung der Verfahren annehmen kann, wird im Folgenden untersucht. Die Topologie eines sphärischen Dreiecksnetzes ist durch die Angabe der Knoten-, Kanten und Flächenzahl eindeutig bestimmt.

Es sei

$$\begin{aligned} p &: \text{Anzahl der Knoten ,} \\ k &: \text{Anzahl der Kanten ,} \\ f &: \text{Anzahl der Flächen .} \end{aligned}$$

Mittels der *Formel von Euler* ($p - k + f = 2$) lässt sich berechnen, wie die beiden Unterteilungsalgorithmen U_1 (Unterteilung über die Seitenmitten) und U_2 (Unterteilung über die Schwerpunkte) auf die Anzahl der Knoten, Kanten und Flächen eines Graphen wirken:

$$\begin{aligned} U_1 &: \begin{pmatrix} p \\ k \\ f \end{pmatrix} \mapsto \begin{pmatrix} 2k - f + 2 \\ 2k + 3f \\ 4f \end{pmatrix} , \\ U_2 &: \begin{pmatrix} p \\ k \\ f \end{pmatrix} \mapsto \begin{pmatrix} p + f \\ p + 4f - 2 \\ 3f \end{pmatrix} . \end{aligned}$$

Hieraus lässt sich ableiten, welche Graphenkonfigurationen aus den drei Startgraphen (Tetraeder, Oktaeder und Ikosaeder) konstruiert werden können. Die Tabelle in Abbildung 4.10 gibt an, welche Graphenkonfigurationen durch maximal drei Unterteilungsschritte erzeugbar sind. Eine Näherung des Winkels α in der letzten Spalte wurde über den Flächeninhalt der sphärischen Dreiecke berechnet. Er gibt an, wie nahe ein Blickrichtungsvektor im Mittel seinen Nachbarn ist.

4.4 Praktische Umsetzung

Die Startgraphen sowie das Verfahren zur Unterteilung über die Seitenmitten wurden in der Programmiersprache C++ implementiert und getestet. Es wurden Blickrichtungsmengen der Größen 4 bis 4096 erzeugt und als Datenbasis in das Lokalisierungssystem integriert. Wie Experimente zeigen, können beispielsweise mit einer Trainingsmenge aus 130 Objektansichten im Falle des Objektes *Hantel* gute Erkennungsergebnisse erzielt werden (vgl. Abschnitt 6.2.5). In Abbildung 4.9 sind zwei typische, künstlich erzeugte Objektansichten des Objektes *Hantel* dargestellt.

Abbildung 4.9: Objektansicht 2 (links) und Objektansicht 4 (rechts) der synthetischen Trainingsmenge *Hantel-Ansichten-130*.

Zur Erzeugung photorealistischer Objektansichten wird im Lokalisierungssystem der frei erhältliche *Raytracer POV-Ray*[4] in der Version 3.5 eingesetzt. Einfache Formmodelle, wie das in Abbildung 4.9 dargestellte, wurden manuell erstellt, komplexere Objekte mittels des in 4.1.3 beschriebenen optischen 3D-Scanners automatisch erfasst. Folgende zwei Arten der Ansichtenerzeugung sind in das System integriert worden und haben sich als erfolgreich herausgestellt:

- Extrem helle, schattenlose Beleuchtung des Objekts von allen Seiten: Alle Partien des Objektes erhalten eine gleichmäßige, weiße Färbung. Dadurch wird nur die Objektsilhouette abgebildet.

[4]Persistance of Vision Raytracer, www.povray.org

- Schattenlose, moderate Beleuchtung von wenigen Seiten. Diese bildet auch Selbstverdeckungen und feine Objektstrukturen ab, die nicht zur Silhouette gehören.

Generell wurden die künstlichen Beleuchtungsbedingungen dahingegen optimiert, mit dem nachgeschalteten Kantenextraktionsverfahren und der Merkmalsextraktion kompatibel zu sein.

Startgraph:	p	k	f	p	k	f	p	k	f	p	k	f	alpha
Tetraeder (k=3)	**4**	6	4										109,47
			U2:	**8**	18	12							77,87
			U1:	**10**	24	16							69,56
						U2, U2:	**20**	54	36				49,02
						U2, U1:	**26**	72	48				42,95
						U1, U2:	**26**	72	48				42,95
						U1, U1:	**34**	96	64				37,53
								U2, U2, U2:		**56**	162	108	29,22
								U2, U2, U1:		**74**	216	144	25,40
								U2, U1, U2:		**74**	216	144	25,40
								U1, U2, U2:		**74**	216	144	25,40
								U1, U1, U2:		**98**	288	192	22,07
								U1, U2, U1:		**98**	288	192	22,07
								U2, U1, U1:		**98**	288	192	22,07
								U1, U1, U1:		**130**	384	256	19,16
Oktaeder (k=4)	**6**	12	8										90,00
			U2:	**14**	36	24							58,68
			U1:	**18**	48	32							51,69
						U2, U2:	**38**	108	72				35,49
						U2, U1:	**50**	144	96				30,92
						U1, U2:	**50**	144	96				30,92
						U1, U1:	**66**	192	128				26,90
								U2, U2, U2:		**110**	324	216	20,83
								U2, U2, U1:		**146**	432	288	18,07
								U2, U1, U2:		**146**	432	288	18,07
								U1, U2, U2:		**146**	432	288	18,07
								U1, U1, U2:		**194**	576	384	15,68
								U1, U2, U1:		**194**	576	384	15,68
								U2, U1, U1:		**194**	576	384	15,68
								U1, U1, U1:		**258**	768	512	13,59
Ikosaeder (k=5)	**12**	30	20										63,43
			U2:	**32**	90	60							38,69
			U1:	**42**	120	80							33,75
						U2, U2:	**92**	270	180				22,78
						U2, U1:	**122**	360	240				19,77
						U1, U2:	**122**	360	240				19,77
						U1, U1:	**162**	480	320				17,16
								U2, U2, U2:		**272**	810	540	13,24
								U2, U2, U1:		**362**	1080	720	11,47
								U2, U1, U2:		**362**	1080	720	11,47
								U1, U2, U2:		**362**	1080	720	11,47
								U1, U1, U2:		**482**	1440	960	9,94
								U1, U2, U1:		**482**	1440	960	9,94
								U2, U1, U1:		**482**	1440	960	9,94
								U1, U1, U1:		**642**	1920	1280	8,62

Abbildung 4.10: Graphenkonfigurationen nach maximal drei Unterteilungsschritten. Die Spalten p, k und f geben die Knoten-, Kanten- und Flächenzahl eines Graphen an. Der Winkel α in der letzten Spalte gibt den mittleren Winkelabstand eines Blickrichtungsvektors zu seinen in Grad Nachbarn an. U_1 bezeichnet das Unterteilungsverfahren über die Seitenmitten und U_2 jenes über die Schwerpunkte.

Kapitel 5

Schätzung der räumlichen Objektlage

Das in Kapitel 2 beschriebene Erkennungsverfahren leistet die Lokalisierung einer zweidimensionalen Modellansicht in einem Testbild[1]. In diesem Kapitel wird beschrieben, wie sich aus einem solchen ansichtsbasierten Lokalisierungsergebnis die dreidimensionale Lage des Objektes bestehend aus drei Translations- und drei Rotationsparametern relativ zur Kamera bestimmen lässt. Abbildung 5.1 veranschaulicht den Abbildungsvorgang der Kamera bei der Erzeugung eines Testbildes.

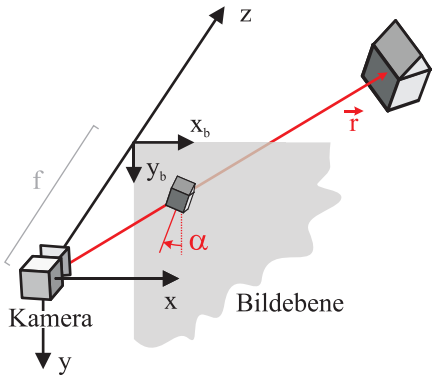

Abbildung 5.1: Abbildungsvorgang der Kamera zur Erzeugung eines Testbildes. Das Objekt mit dem Ortsvektor **r** wird auf die Bildebene projiziert. Sein Abbild ist gegenüber der gelernten Modellansicht um den Winkel α auf der Bildebene gedreht.

Gesucht werden der Ortsvektor **r** des Objektes im Kamerakoordinatensystem sowie die drei Eulerwinkel e_0, e_1 und e_2, welche die Orientierung des Objektes im Raum angeben.

[1] Im Folgenden wird der Begriff *Testbild* für ein Bild verwendet, in welchem ein Objekt lokalisiert werden soll.

Grundlage der Berechnung sind folgende Daten:

- Die Lage $\mathbf{L} = (s, \alpha, x_b, y_b)$ des Objektabbildes im Testbild, wobei s die Skalierung α die Rotation und $(x_b,\, y_b)$ die Position in der Bildebene bezeichnen.
- Die Kalibriermatrix K_L der Kamera, die zum Lernen der Modellansicht verwendet wurde.
- Die Kalibriermatrix K_T der Kamera, die bei der Aufnahme des Testbildes verwendet wurde.
- Die Parameter ϕ und θ der Objektansicht, die angeben, aus welcher Richtung das Objekt während des Lernverfahrens aufgenommen worden ist.

In den nächsten beiden Abschnitten wird dargestellt, welches Kameramodell der Lokalisierung zugrundegelegt wird und wie sich die oben genannte Lage $\mathbf{L}$ aus dem ansichtsbasierten Erkennungsergebnis berechnen lässt. Anschließend wird darauf aufbauend eine Transformation hergeleitet, die das Koordinatensystem des Formmodells in das Kamerakoordinatensystem überführt und dadurch die Objektlokalisierung ermöglicht.

5.1 Kameramodell

Die projektive Abbildung, welche die Raumkoordinaten (x, y, z) in die zugehörigen Bildkoordinaten (x_b, y_b) überführt, wird durch folgende Gleichung definiert:

$$w \cdot \begin{pmatrix} x_b \\ y_b \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} s_x f & s_{xy} f & c_x & 0 \\ 0 & s_y f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{K} \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{T} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}.$$

Dabei beschreibt die Matrix T die räumliche Lage der Kamera bezüglich des Szenenkoordinatensystems. Die Matrix K wird auch *Kalibriermatrix* der Kamera genannt und enthält die Brennweite f, den Hauptpunkt (c_x, c_y) und die Verzerrungsfaktoren s_x, s_y und s_{xy} der Kamera. Man beachte, dass bei einer projektiven Abbildung stets ein Faktor w unbestimmt bleibt. Ein anschaulicher Grund hierfür ist z.B. die Tatsache, dass ein doppelt so großes und gleichzeitig doppelt so weit entferntes Objekt zum gleichen Abbild führt. Für die Herleitung des Lokalisationsverfahrens wird von folgenden Voraussetzungen ausgegangen:

1. Szenen- und Kamerakoordinatensystem sind identisch: $T = I_{4\times4}$.
2. Die Bildebene der Kamera ist rechtwinklig und der Abstand der horizontalen Bildpunkte entspricht dem der vertikalen: $s_y = s_x$, $s_{xy} = 0$.

Voraussetzung 1 schränkt die Allgemeinheit nicht ein, da die Lage eines bezüglich der Kamera lokalisierten Objektes leicht in ein anderes Koordinatensystem transformiert werden kann[2]. Voraussetzung 2 ist bei hochwertigen Kameras gegeben[3]. Im Falle einer verzerrenden Kamera kann das Kamerabild vor der Verarbeitung durch eine affine Transformation entzerrt werden, so dass auch dann Voraussetzung 2 erfüllt ist. Durch die Voraussetzungen vereinfacht sich die Abbildungsvorschrift auf

$$w \cdot \begin{pmatrix} x_b \\ y_b \\ 1 \end{pmatrix} = \begin{pmatrix} s_x f & 0 & c_x & 0 \\ 0 & s_x f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \tag{5.1}$$

Siehe [Armangué, Salvi & Batlle 2000] für einen Vergleich gängiger Kalibrierverfahren für reale Kameras. Wie die Kalibriermatrix einer virtuellen Kamera bestimmt werden kann, wird in Anhang B.1 beschrieben.

5.2 Lage des Objektabbildes auf der Bildebene

Als Erkennungsergebnis liefert das ansichtsbasierte Verfahren diejenige Modellansicht, die am Besten an das Testbild angepasst werden konnte sowie deren vierdimensionale Lage $\mathbf{L_1}$ in der Bildebene. Diese Modellansicht besteht aus einer Menge von Objektansichten (vgl. Kapitel 2), welche jeweils eine bestimmte Relativlage zu ihr besitzen. Die Objektlokalisierung soll bezüglich einer dieser Objektansichten erfolgen. Deren Relativlage zur Modellansicht sei mit $\mathbf{L_0}$ bezeichnet[4].

Um die Lage $\mathbf{L}$ des Objektabbildes in der Bildebene zu bestimmen, müssen $\mathbf{L_1}$ und $\mathbf{L_0}$ miteinander verknüpft werden (vgl. Abbildung 5.2).

Notation: Die *Lage* eines zweidimensionalen Musters in der Bildebene wird durch den Vektor $\mathbf{L} = (s, \alpha, x_b, y_b)$ beschrieben (s: Skalierung, α: Rotation, (x_b, y_b): Position in

[2] Im Projekt MQube ist diese Voraussetzung unmittelbar erfüllt, da bei der Verfolgung die Objektlage relativ zur Kamera geschätzt wird.

[3] Dies ist ebenfalls im Projekt MQube der Fall.

[4] Die Richtungen der Lagevektoren $\mathbf{L_1}$ und $\mathbf{L_0}$ wurden konsistent zu dem in Kapitel 2 beschriebenen Verfahren gewählt.

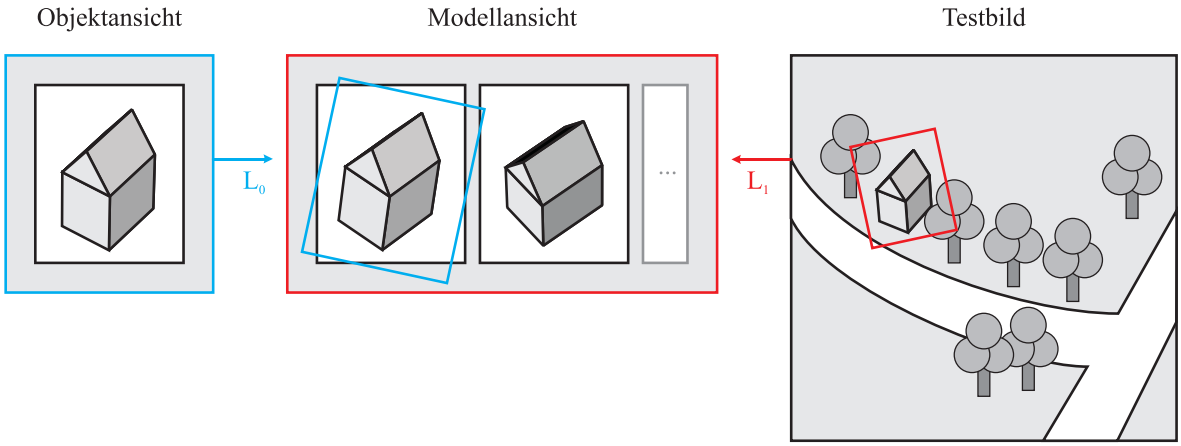

Abbildung 5.2: Bestimmung der Lage des Objektabbildes im Testbild. Der Lagevektor $\mathbf{L_1}$, bzw. die zugehörige Transformation L_1, beschreibt die Lage der Modellansicht im Testbild während $\mathbf{L_0}$ (bzw. L_0) für die Relativlage einer Objektansicht zur Modellansicht steht.

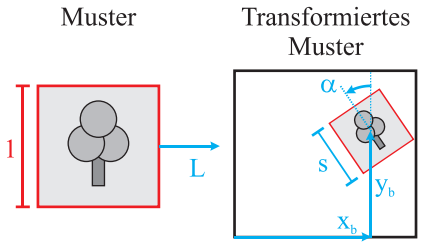

Abbildung 5.3: Die Lage eines Bildmusters wird beschrieben durch Skalierung s, Rotation α und Position (x_b, y_b). Jedem Lagevektor ist eindeutig eine Transformation L zugeordnet, welche die entsprechende Koordinatentransformation ausführt.

der Bildebene, siehe Abbildung 5.3). Die Transformation, die ein Muster in diese Lage bringt, hat die Form

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} u & -v & x_b \\ v & u & y_b \\ 0 & 0 & 1 \end{pmatrix}}_{L} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \qquad \begin{aligned} u &= s\,\cos(\alpha), \\ v &= s\,\sin(\alpha). \end{aligned}$$

Die Transformation und der Einfachheit halber auch die Matrix zum Lagevektor $\mathbf{L}$ sei mit L bezeichnet.

Die Lage $\mathbf{L_1}$ der Modellansicht und die Relativlage $\mathbf{L_0}$ der Objektansicht sind folgendermaßen miteinander zu verknüpfen (siehe Abbildung 5.2):

$$L = \left(L_0^{-1}\right) \circ L_1.$$

Die Transformation $\mathbf{L}$ überführt ein Muster des Testbildes in das Koordinatenssystem des Objektabbildes. L^{-1} gibt demnach direkt die Lokalisierung des Objektabbildes im Testbild an. In Anhang B.2 wird beschrieben, welche konkrete Form die Matrix L besitzt und wie man unterschiedlichen Koordinatensystemen für Objektansicht und Testbild begegnen kann.

5.3 Transformationskette vom Objektabbild in das Koordinatensystem des Formmodells

Ein anschaulicher Weg zur Bestimmung der räumlichen Objektlage ist die schrittweise Überführung des Kamerakoordinatensystems in das Koordinatensystem des Formmodells durch eine lineare Kette einzelner Teiltransformationen (siehe Abbildung 5.4).

Im Folgenden werden die einzelnen Transformationen hergeleitet und daraus schließlich die räumliche Objektlage bestimmt.

5.3.1 Verschiebung des Objektes auf den Ursprung

Um das Objekt auf den Ursprung des Kamerakoordinatensystems verschieben zu können, muss der dreidimensionale Ortsvektor $\mathbf{r}$ des Objektes bestimmt werden. Dazu berechnet man, welchen Abstand r das Objekt von der Kamera hat und in welcher Richtung $\mathbf{r_0}$ es liegt. Diese beiden Informationen zusammen ergeben direkt den Ortsvektor $\mathbf{r}$.

Durch die Lokalisation der Objektansicht im Testbild ist bekannt, um welchen Faktor s sich die Längen einer Strecke B der Objektansicht und der entsprechenden Strecke $\overline{B}$ des Objektabbildes im Testbild unterscheiden. Es gilt also der Zusammenhang

$$B = s\,\overline{B}. \tag{5.2}$$

Die Kamera, die zum Lernen der Objektansicht eingesetzt wurde, besitze die Brennweite f und den Verzerrungsfaktor s_x, die Kamera, mit der das Testbild aufgenommen wurde, die Parameter $\overline{f}$ und $\overline{s_x}$. Dann lässt sich über die Kalibriermatrizen der Kameras berechnen, welche Längen die realen Strecken b und $\overline{b}$ auf den Sensormatrizen der Kameras besitzen:

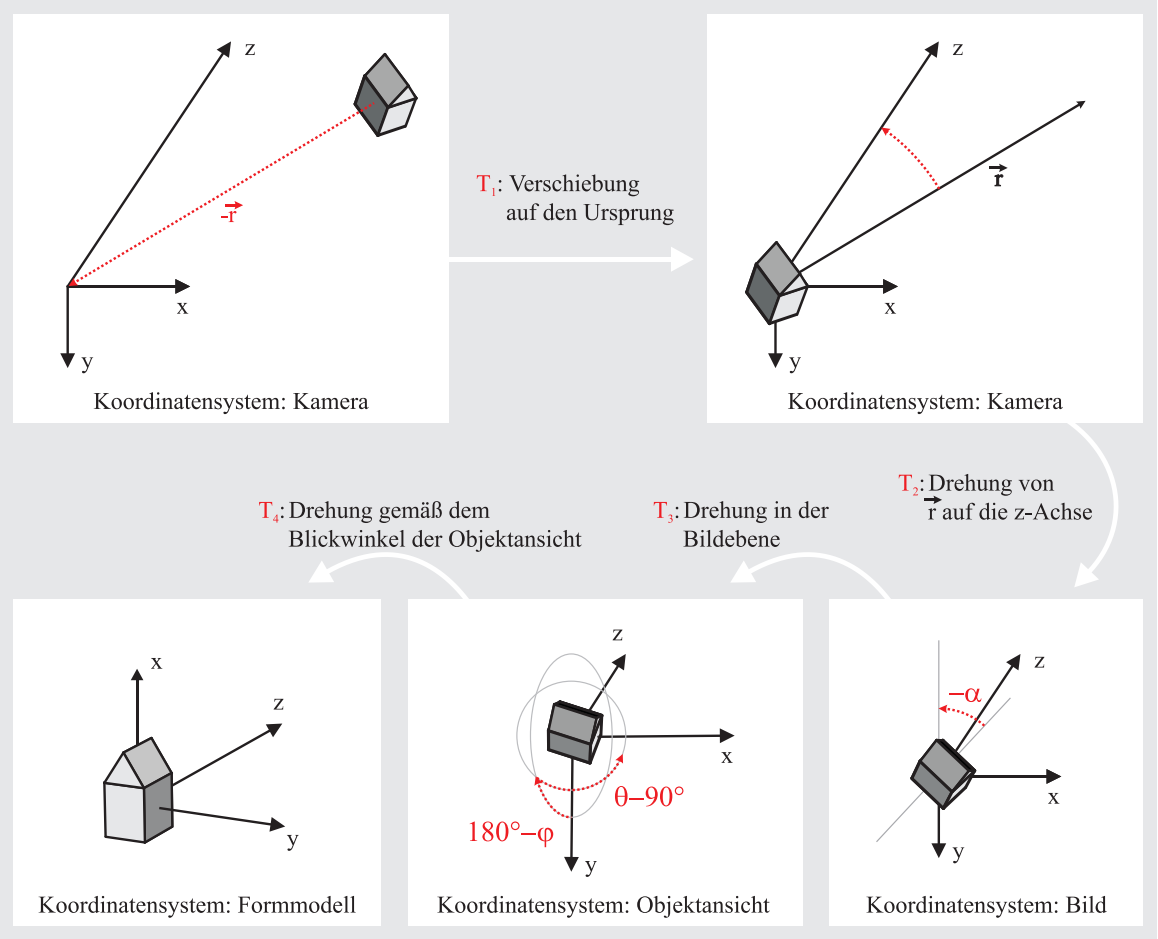

Abbildung 5.4: Übergang vom Kamerakoordinatensystem zum Koordinatensystem des Formmodells durch vier Teiltransformationen.

$$b = \frac{B}{s_x}, \qquad \bar{b} = \frac{\overline{B}}{\overline{s_x}}.$$

Abbildung 5.5 zeigt links den Abbildungsvorgang, der während des Lernverfahrens zur Modellansicht führt und rechts die Aufnahme eines Testbildes.

Da beim Lernen der Modellansicht alle relevanten Parameter bekannt und beide Kameras kalibriert sind, lässt sich mithilfe des zweiten Strahlensatzes die Größe r folgendermaßen berechnen:

$$\frac{l}{c} = \frac{f}{b}, \qquad \frac{r}{c} = \frac{\overline{f}}{\overline{b_0}} \approx \frac{\overline{f}}{\overline{b}} \tag{5.3}$$

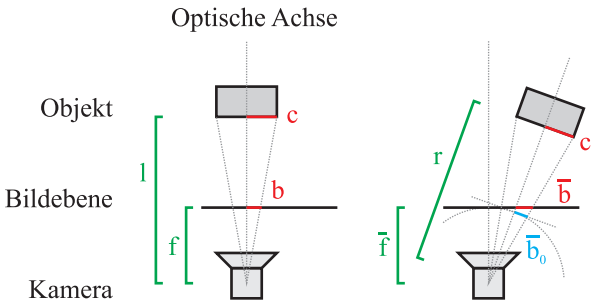

Abbildung 5.5: Bestimmung der Objektentfernung durch den Vergleich zweier Abbilder. Links: Der Abbildungsvorgang beim Lernen der Modellansicht. Eine senkrecht zur optischen Achse stehende Strecke c beliebiger Länge wird unter Berücksichtigung von Kamerabrennweite f und Objektentfernung l auf eine Strecke b abgebildet. Rechts: Der Abbildungsvorgang bei der Aufnahme des Testbildes. Die Strecke c wird bei unbekannter Objektentfernung r und bekannter Kamerabrennweite $\overline{f}$ auf die Strecke $\overline{b}$ abgebildet. Es wird angenommen, dass die perspektivische Verzerrung klein ist, also $\overline{b} \approx \overline{b_0}$ gilt.

$$\Longrightarrow \qquad r \approx l \cdot \frac{\overline{f}\, b}{f\, \overline{b}} \,. \tag{5.4}$$

Wählt man O.B.d.A. die Urbildstrecke c auf dem Objekt derart, dass ihr Bild B beim Lernverfahren genau 1 Pixel lang wird, so ergeben sich wegen der Kamerakalibrierung und Gleichung 5.2 die folgenden Längen der Bildstrecken:

$$b = \frac{B}{s_x} = \frac{1}{s_x}, \qquad \overline{b} = \frac{\overline{B}}{\overline{s_x}} = \frac{B}{s\ \overline{s_x}} = \frac{1}{s\ \overline{s_x}} \,.$$

Zusammen mit Gleichung 5.4 erhält man als Abstand zwischen Objekt und Kamera

$$r = l\, s \cdot \frac{\overline{f}\ \overline{s_x}}{f\ s_x} \,. \tag{5.5}$$

Man beachte, dass die Produkte $\overline{f}\ \overline{s_x}$ und $f\, s_x$ direkt aus den zugehörigen Kalibriermatrizen abgelesen werden können. Die Richtung des Objektortsvektors $\mathbf{r}$ lässt sich über die Position (x_b, y_b) der Objektansicht auf der Bildebene berechnen. Aus Projektionsgleichung 5.1 erhält man den Ortsvektor $\mathbf{r_0} = (x, y, z)$ des Objektabbildes auf der Bildebene, indem man $w = s_x f$ einsetzt und nach (x, y, z) auflöst:

$$
\begin{aligned}
s_x\, f \begin{pmatrix} x_b \\ y_b \\ 1 \end{pmatrix} &= \begin{pmatrix} s_x f & 0 & c_x & 0 \\ 0 & s_x f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \\
\Longrightarrow \quad \mathbf{r_0} &= \begin{pmatrix} x_b - c_x \\ y_b - c_y \\ s_x\, f \end{pmatrix}.
\end{aligned}
$$

Als Ortsvektor des Objektes ergibt sich

$$
\mathbf{r} = r\, \frac{\mathbf{r_0}}{|\, \mathbf{r_0}\, |}
$$

und als entsprechende Transformationsmatrix

$$
T_1 = \begin{pmatrix} 1 & 0 & 0 & -\mathbf{r}_x \\ 0 & 1 & 0 & -\mathbf{r}_y \\ 0 & 0 & 1 & -\mathbf{r}_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad \mathbf{r} = \begin{pmatrix} \mathbf{r}_x \\ \mathbf{r}_y \\ \mathbf{r}_z \end{pmatrix}.
$$

5.3.2 Übergang zum Koordinatensystem der Objektansicht

Die Transformationen T_2 und T_3 überführen das Koordinatensystem der Testbildkamera in das der Kamera zum Lernen der Modellansicht. Dazu dreht T_2 den Vektor $\mathbf{r}$ auf die z-Achse, was der Tatsache Rechnung trägt, dass die Kamera beim Lernen der Modellansicht genau auf das Objektzentrum gerichtet war (wohin der Vektor $\mathbf{r}$ zeigt). T_3 dreht das Objekt in der xy-Ebene um genau den Winkel α, um den sich das Objektabbild auf der Bildebene von der Objektansicht unterscheidet.

Es ist

$$
T_2 = T_{\mathbf{r},\hat{\mathbf{z}}}, \qquad \hat{\mathbf{z}} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}
$$

und

$$T_3 = T_{\hat{\mathbf{z}},\alpha}, \qquad \hat{\mathbf{z}} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} .$$

Definition: $T_{\mathbf{a},\mathbf{b}}$ drehe hierbei den Vektor $\mathbf{a}$ auf den Vektor $\mathbf{b}$ und $T_{\mathbf{a},\alpha}$ drehe um die Achse durch den Ursprung mit Richtung $\mathbf{a}$ um dem Winkel α. Die Matrizen dieser Transformationen sind in Anhang B.3 angegeben.

5.3.3 Von der Objektansicht in das Formmodell

Die Tranformation T_4 überführt das Koordinatensystem der Objektansicht in das des Formmodells. Abbildung 5.6 veranschaulicht die Abhängigkeiten der verschiedenen Koordinatensysteme bei der Erzeugung einer Objektansicht.

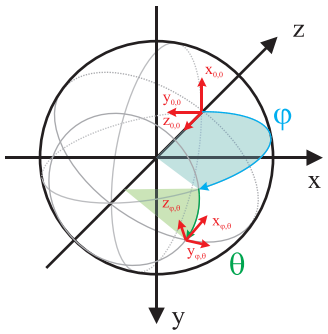

Abbildung 5.6: Die Koordinatensysteme bei der Erzeugung einer Modellansicht. Das Formmodell ist bezüglich des xyz-Koordinatensystems definiert. Die Kamera wird abhängig von den Winkelparametern ϕ und θ auf der Einheitskugel um den Ursprung positioniert.

Das Formmodell ist bezüglich des xyz-Koordinatensystems definiert und um dessen Ursprung zentriert. Die Lage der Kamera sei durch das Koordinatensystem $[\mathbf{x}_{\phi,\theta}, \mathbf{y}_{\phi,\theta}, \mathbf{z}_{\phi,\theta}]$ beschrieben. Sie ist durch die zwei Winkelparameter ϕ und θ mit den folgenden Forderungen eindeutig bestimmt:

- Der Ursprung des Kamerakoordinatensystems liegt auf der Einheitskugel um den Ursprung.
- Die Achse $\mathbf{z}_{\phi,\theta}$ zeige in Richtung des Ursprungs.

- Der Ursprung von $[\mathbf{x}_{\phi,\theta}, \mathbf{y}_{\phi,\theta}, \mathbf{z}_{\phi,\theta}]$ wird gemäß der üblichen Definition der sphärischen Winkel ϕ (Polarwinkel) und θ (Azimuthwinkel) bestimmt (siehe Anhang B.4 für diese Definition).
- Es verbleibt ein Freiheitsgrad: die Drehung des Kamerakoordinatensystems um die Achse $\mathbf{z}_{\phi,\theta}$. Dieser wird durch die Forderung eliminiert, dass die Achse $\mathbf{y}_{\phi,\theta}$ in der Ebene liegen soll, die durch die z-Achse und $\mathbf{z}_{\phi,\theta}$ aufgespannt wird. Dies hat zur Folge, dass mit $\phi = \theta = 0$ ein aufrechtes Bild des Objektes entsteht, falls das Formmodell wie in Abbildung 5.4 (unten links) definiert ist.

Die Transformation T_4, welche das Koordinatensystem $[\mathbf{x}_{\phi,\theta}, \mathbf{y}_{\phi,\theta}, \mathbf{z}_{\phi,\theta}]$ auf das des Formmodells $[x, y, z]$ überführt, kann in folgende zwei Teile zerlegt werden. Es sei

$$\begin{aligned} T_4 &= T_{4,2} \circ T_{4,1} \ , \\ T_{4,1} &= T_{\mathbf{x},(180^\circ - \phi)} \ , \\ T_{4,2} &= T_{T_{4,1}(\mathbf{z}),(\theta - 90^\circ)} \ . \end{aligned}$$

$T_{4,1}$ dreht das Koordinatensystem um die $\mathbf{x}$-Achse mit $180 - \phi$ Grad. $T_{4,2}$ dreht um die resultierende $\mathbf{z}$-Achse mit $\theta - 90$ Grad.

5.4 Lokalisierung

Die beschriebenen Transformationen lassen sich kombinieren und invertieren:

$$T = (T_4 \circ T_3 \circ T_2 \circ T_1)^{-1} \ . \tag{5.6}$$

So erhält man eine Transformation T, welche Punkte des Formmodells an die entsprechenden Stellen im Kamerakoordinatensystem überführt:

$$\begin{pmatrix} x_{kamera} \\ y_{kamera} \\ z_{kamera} \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{T} \begin{pmatrix} x_{modell} \\ y_{modell} \\ z_{modell} \\ 1 \end{pmatrix}$$

Die räumliche Position $\mathbf{r}$ des Objektes lässt sich direkt angeben zu

$$\mathbf{r} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}.$$

Die drei Eulerwinkel e_0, e_1 und e_2 der räumlichen Orientierung des Objektes lassen sich aus der Rotationsmatrix $R := ((r_{11}, \ldots, r_{33}))$ berechnen (siehe hierzu [Shoemake 1994]).

Kapitel 6

Evaluierung des Lokalisierungssystems

In diesem Kapitel wird das realisierte Lokalisierungssystem auf Stärken und Schwächen hin untersucht, um seinen Anwendungsbereich zu definieren und Hinweise auf Verbesserungsmöglichkeiten zu bekommen. Zu Beginn werden theoretische Überlegungen zur Leistungsfähigkeit des Systems angestellt und anschließend die Ergebnisse praktischer Untersuchungen mit künstlichen und realen Daten präsentiert.

Für die experimentellen Untersuchungen wurden Arbeitsplatzrechner mit Intel-Prozessoren der Taktfrequenzen 1,7 GHz bis 3 GHz verwendet. Die Experimente fanden ausschließlich unter dem Betriebssystem Linux statt.

6.1 Theoretische und praktische Voruntersuchungen

6.1.1 Einfluss der projektiven Verzerrung

Das verwendete ansichtsbasierte Erkennungsverfahren setzt voraus, dass die projektive Verzerrung durch den Abbildungsvorgang gering genug ist, um Objektansichten unabhängig von ihrer Lage auf der Bildebene zu erkennen (vgl. Kapitel 5.3.1). Dass diese Annahme für praxisrelevante Szenen gerechtfertigt ist, zeigt die folgende Überlegung.

In Anhang Anhang B.5 wird hergeleitet, wie sich der projizierte Schnittwinkel γ zweier Geraden ändert, wenn deren Schnittpunkt um den Ursprung rotiert wird. Dieser Zusammenhang gibt Aufschluss über die perspektivische Verzerrung einer Objektansicht, die sich bei Drehung des Objektes um die Kamera einstellt (siehe Abbildung 6.1).

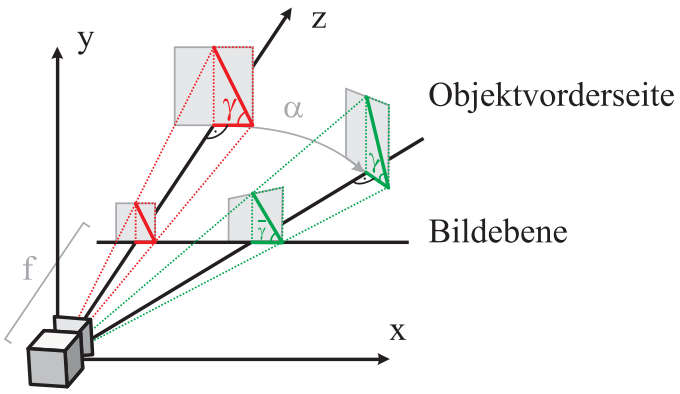

Abbildung 6.1: Rotation eines Objektes um die Kamera. Die Objektvorderseite ist grau dargestellt, ein sich darauf befindender Winkel γ rot und die beiden Winkel nach der Objektrotation grün.

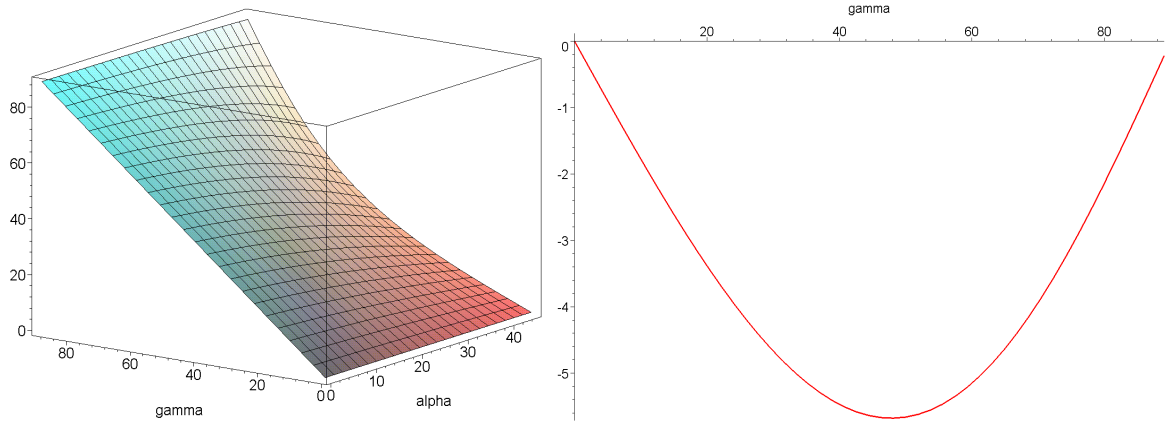

Abbildung 6.2: Projektive Verzerrung eines Winkels auf der Objektoberfläche bei Drehung des Objektes um die Kamera. Links: Nach oben ist der projizierte Winkel $\overline{\gamma}$ aufgetragen, auf den ein Winkel γ einer Struktur auf der Objektoberfläche abgebildet wird, wenn das Objekt mit dem Winkel α um die Kamera rotiert wird. Rechts: Winkelveränderung $\overline{\gamma}-\gamma$ bei Rotation um $\alpha = \frac{41}{2}°$, aufgetragen über γ.

Ein Objekt von 1 Meter Durchmesser befinde sich im Abstand von 1,5 Metern vor der Kamera[1], so dass sich zwei seiner Bildkanten im Winkel γ schneiden. Das linke Diagramm der Abbildung 6.2 zeigt, welcher Schnittwinkel $\overline{\gamma}$ in der Bildebene auftritt, wenn das Objekt um die Kamera mit dem Winkel α rotiert wird. Nach links ist dabei der Schnittwinkel γ der Kanten aufgetragen, nach rechts der Winkel α mit dem um die Kamera gedreht wird. Darüber ist der resultierende projizierte Winkel $\overline{\gamma}$ aufgetragen.

Zur Ermittlung der maximalen Verzerrung eines Winkels durch den Abbildungsvor-

[1]Dies sei die ungünstigste Szene, mit der das Erkennungssystem konfrontiert werden soll.

gang setzt man für α den halben Öffnungswinkel der verwendeten Kamera ein und betrachtet den Zusammenhang zwischen der Veränderung $\overline{\gamma} - \gamma$ und γ. Auf der rechten Seite von Abbildung 6.2 ist dieser Zusammenhang für den Kameraöffnungswinkel 41° dargestellt[2]. Die maximale Veränderungen von 5,6° stellt sich mit dem Winkel $\gamma = 47°$ ein. Selbst dieser maximale Unterschied ist so gering, dass das Erkennungsverfahren damit problemlos umgehen kann.

6.1.2 Voraussetzungen des Verfahrens zur Graphenanpassung

Wichtigster Bestandteil des in Kapitel 2 beschriebenen Erkennungsverfahrens ist der Algorithmus zur Anpassung von Graphen aus Bildmerkmalen. Er kommt nicht nur bei der Erkennung von Modellansichten zum Einsatz, sondern stellt auch das zentrale Hilfsmittel für das Lernen von Modellansichten aus einer Trainingsmenge dar. Aus der Herleitung des Anpassungsverfahrens und seiner Implementierung ergeben sich folgende Aussagen über die zu erwartende Leistungsfähigkeit.

Eine zentrale Bedeutung kommt denjenigen Bildmerkmalen zu, die stabil bezüglich ihrer vier Lagedimensionen (2D-Position, Orientierung und Skalierung) lokalisiert sind. Nur bei ihnen kann die Korrespondenzsuche ansetzen und eine initiale Transformationshypothese schätzen. Beispielsweise eignen sich Paare von Geradensegmenten nicht zur initialen Schätzung einer Transformation, da deren 2D-Position und Skalierung (im Gegensatz zur Orientierung) nicht als stabil angesehen werden können. Die Bildmerkmale, deren vier Dimensionen als stabil angesehen werden, erhalten im Folgenden die Bezeichnung *Initialmerkmale*, um ihre Bedeutung für den Beginn der Korrespondenzsuche zu unterstreichen.

Die Menge der *Initialmerkmale* besteht aus

- *Kappe*, *Kappenpaar*, *Kurbel-LR*, *Kurbel-RL* und *Region*.

Keine Initialmerkmale sind

- *Geradensegment*, *Kurvensegment*, *Kreis*, *L-Ecke* und *Parallelen*.

Paare aus Initialmerkmalen werden als *Initialpaare* bezeichnet. Zu Beginn der Graphenanpassung werden alle Initialpaare zwischen Testbild und Modellansicht ermittelt und daraus erste Transformationshypothesen geschätzt. Aus diesem Vorgehen folgt eine starke Abhängigkeit der Erkennungsleistung von der Qualität der Initialmerkmale.

[2]Die im Projekt MQube vorrangig eingesetzte Kamera besitzt diesen Öffnungswinkel.

Ideal für den Erkennungsvorgang ist eine kleine Menge von präzise extrahierten Initialpaaren. Problematisch sind folgende Situationen:

- In Bild oder Modellansicht sind zu wenige Initialmerkmale enthalten, die aufgrund ihrer Attributausprägungen zu geeigneten Initialpaaren zusammengefasst werden können. In diesem Fall kann der Anpassungsvorgang nicht beginnen.
- In Bild und Modellansicht werden sehr viele Initialmerkmale identifiziert, so dass sich eine übermäßig große Menge an Initalpaaren ergibt. Hierunter leidet die Effizienz des Verfahrens, da unter Umständen ein großer Teil der Paare betrachtet werden muss, bevor eine geeignete Transformation gefunden wird. Darüberhinaus bricht das Anpassungsverfahren nach einer bestimmten Anzahl von erfolglosen Anpassungshypothesen die Suche ab und liefert ein negatives Ergebnis zurück[3]. Durch eine zu große Menge an Initialpaaren sinkt also auch die Wahrscheinlichkeit eines Erkennungserfolgs.
- Die Initialmerkmale können, bedingt durch schlechte Bildqualität oder ungünstige Parametrisierung der Vorverarbeitung, nur ungenau extrahiert werden. Unter Umständen unterscheidet sich dann die geschätzte Transformationen zu stark von der tatsächlichen, so dass die Erkennung mißlingt.

Aufgrund dieser Vorüberlegungen wurde in den praktischen Untersuchungen besonderes Augenmerk auf die Qualität und den Einfluss der Initialmerkmale gelegt.

6.1.3 Leistungsfähigkeit des Merkmalsrepertoirs und der Graphendarstellung

Die Qualität der Merkmalsextraktion hat einen bedeutenden Einfluss auf die Leistung eines Erkennungssystems.In der verwendeten Implementierung wird ein Eingabebild durch die folgenden Schritte in eine Graphendarstellung überführt (siehe auch Kapitel 2):

1. **Extraktion von Kantenelementen**: *Kantenelemente* sind Stellen im Bild, an denen der Grauwertgradient ein Maximum in Gradientenrichtung besitzt, also ein Grauwertübergang vorliegt. Charakterisiert werden Kantenelemente durch ihre Position im Bild (in Pixeln) und oft durch Orientierung und Stärke des Grauwertübergangs.

[3]Ein Schwellwert von 50 erfolglosen Anpassungshypothesen hat sich als praktikabel erwiesen.

2. **Gruppierung von Kantenelementen zu Kurvensegmenten**: ausgehend von kleinen, quadratischen Bildbereichen werden aus den Kantenelementen Kurvensegmente gebildet und diese sukzessive erweitert.

3. **Gruppierung von Kurvensegmenten zu komplexeren Einheiten**: die Kurvensegmente werden zu komplexeren Beschreibungen der Grauwertstrukturen zusammengefasst und in einem Graph repräsentiert.

Tests mit praxisrelevanten Bildern haben ergeben, dass die extrahierten Merkmalsgraphen die für das Erkennungsverfahren relevanten Strukturen robust abbilden, sofern die Auflösung der Strukturen hoch genug und das Ausmaß an Störungen wie Schattenwurf und Bildrauschen hinreichend niedrig ist. Die Auswirkung zu niedriger Bildauflösung soll hier beispielhaft an einem Experiment mit dem Testbild *M3-Hantel-10* demonstriert werden. Abbildung 6.3 zeigt das Bild mit den identifizierten Geradensegmenten sowie mit den für die Erkennung wichtigen Initialmerkmalen des Typs *Kappe*.

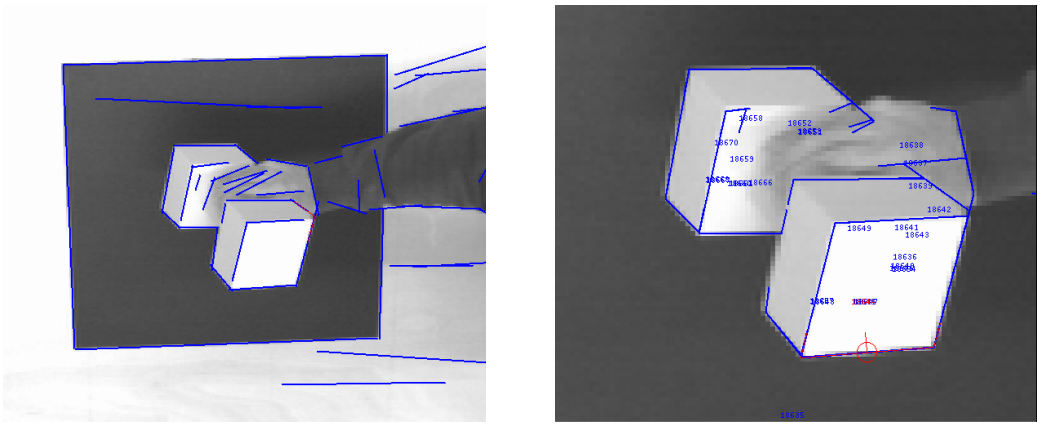

Abbildung 6.3: Das Testbild *M3-Hantel-10* nach der Vorverarbeitung. Links: Die extrahierten Geradensegmente sind überlagert dargestellt. Rechts: Die Initialmerkmale des Typs *Kappe* sowie ihre systeminternen Bezeichnungen sind überlagert dargestellt. Zur Veranschaulichung wurde eine spezielle *Kappe* rot eingefärbt.

Im Testbild besitzt das Objektabbild eine Größe von ca. 60 Pixeln. Aufgrund der verdeckenden Hand sind nur 29 Ecken des Hantelobjektes sichtbar (von 43 möglichen in dieser Objektlage). Durch das Extraktionsverfahren wurden 20 dieser Ecken als Bildmerkmale des Typs *L-Ecke* identifiziert. Wären zwei Linien der Hantel nicht beleuchtungsbedingt unerkannt geblieben, hätten acht weitere *L-Ecke*-Merkmale leicht identifizert werden können. Die gefundene Menge von 20 *L-Ecke*-Merkmalen reicht aus, um daraus 20 Initalmerkmale (*Kappe*, *Kappenpaar*, *Kurbel-LR*, *Kurbel-RL* und *Region*) zu erzeugen, an denen der Erkennungsprozess ansetzen kann. Aufgrund dieser hohen Zahl an Initialmerkmalen verläuft die Erkennung problemlos.

Abbildung 6.4 zeigt die Extraktionsergebnisse für künstlich verringerte Bildauflösungen. Für das auf 75% verkleinerte Testbild (in der Abbildung links) werden nur noch 10 *L-Ecke*-Merkmale gefunden. Aus dieser Menge können lediglich vier Initialmerkmale gebildet werden, weshalb die Erkennung nur noch bei mässig komplexem Hintergrund möglich ist. Bei dem auf 50% verkleinerten Testbild (in der Abbildung rechts) werden nur noch 3 Merkmale des Typs *L-Ecke* gefunden. Die Gruppierung dieser Merkmale ist nicht möglich, weshalb die Erkennung scheitert.

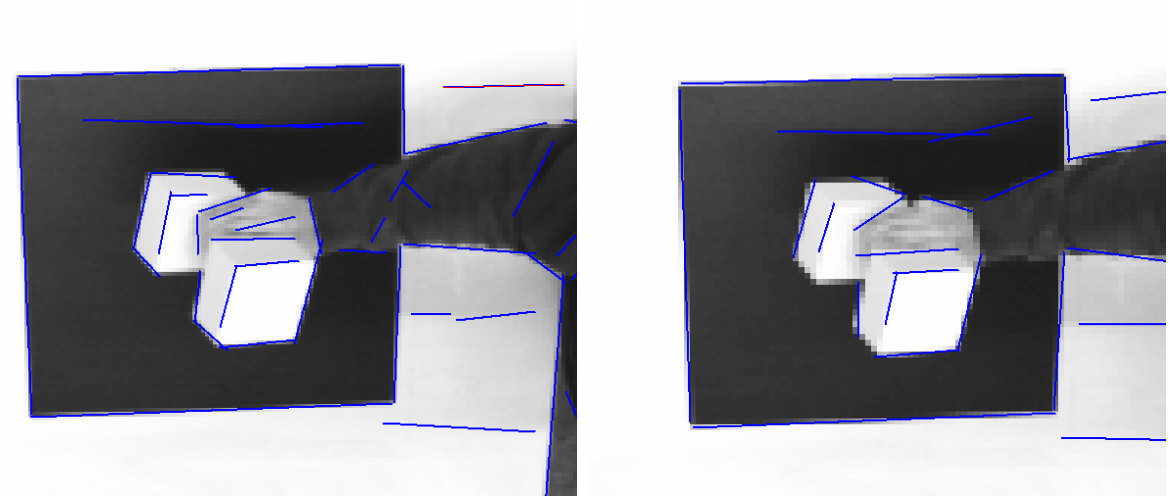

Abbildung 6.4: Verkleinerte Versionen des Testbilds *M3-Hantel-10* nach der Vorverarbeitung. Links: Das auf 75% verkleinerte Bild mit überlagerten Geradensegmenten. Rechts: Das auf 50% verkleinerte Bild mit überlagerten Geradensegmenten.

Die zusätzliche Extraktion von Kreissegmenten hatte bei Freiformobjekten in den durchgeführten Experimenten keine Steigerung der Erkennungsleistung zur Folge. Abbildung 6.5 zeigt zwei typische Extraktionsergebnisse für ein Objekt mit deutlich gekrümmter Silhouette. Obwohl die Kontur des Objektes augenscheinlich besser durch Kreissegmente beschrieben wird als allein durch Geradensegmente, ergibt sich eine niedrigere Erkennungsrate. Dies kann darauf zurückgeführt werden, dass die Schnittstellen zwischen Kreissegmenten nicht robust gefunden werden und so nur verhältnismäßig wenige Initialmerkmale gebildet werden können.

Eine Schwachstelle des eingesetzten Verfahrens zur Merkmalsextraktion ist die lange Laufzeit. Für die Überführung eines Bildes der Größe 384 × 288 Pixel[4] mit ca. 30000 Kantenelementen in die Graphenrepräsentation werden auf einem Testrechner mit 3 GHz Taktfrequenz durchschnittlich 35 Sekunden benötigt. Dabei entfallen etwa 32 Sekunden auf die Gruppierung von Kantenelementen zu Geraden- und Kurvensegmenten (obiger zweiter Vorverarbeitungsschritt).

[4]Dies entspricht der Größe von Kamerabildern im Projekt MQube.

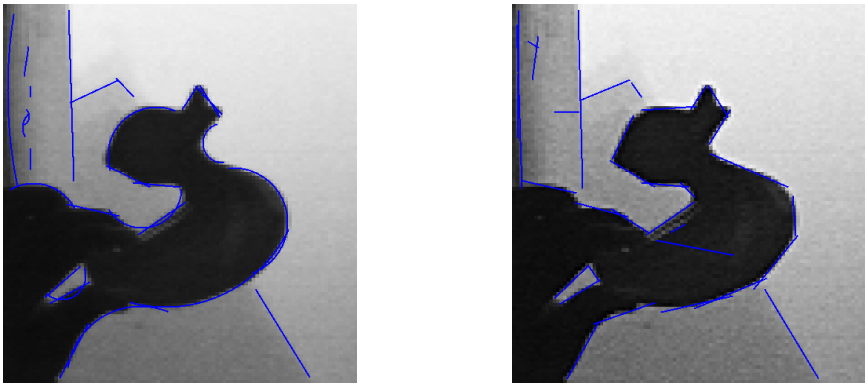

Abbildung 6.5: Ergebnis der Vorverarbeitung eines Bildes mit deutlich gekrümmten Strukturen. Links: Das Extraktionsergebnis mit Kreissegmenten. Rechts: Das Extraktionsergebnis unter Ausschluss von Kreissegementen.

Im Rahmen dieser Arbeit wurde eine Alternative für die Extraktion von Geradensegmenten entwickelt und implementiert, die durchschnittlich um den Faktor 27 schneller arbeitet. Sie basiert auf dem Ansatz der Hough-Transformation und nutzt die wertvolle Information der Orientierung von Kantenelementen. Konkret werden für die in Abbildung 6.6 dargestellten Ergebnisse mit dem Hough-basierten Ansatz auf einem Testrechner mit 3 GHz Taktfrequenz nur 0,27 Sekunden benötigt im Vergleich zu 7,4 Sekunden bei dem anderen Verfahren. Aus Zeitgründen wurde auf die Integration des Hough-basierten Verfahrens in das Lokalisierungssystem verzichtet.

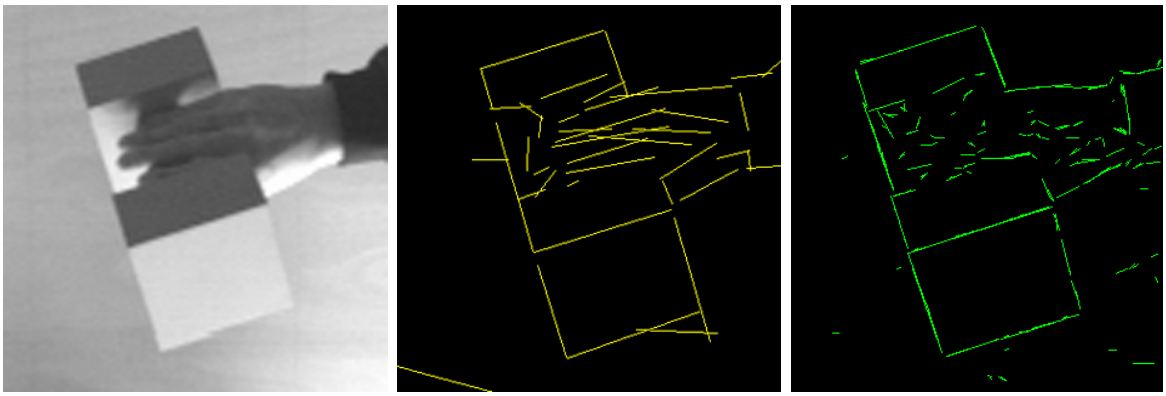

Abbildung 6.6: Gruppierung von Kantenelementen. Links: Originalbild in der Auflösung 307×319 Pixel. Mitte: Gruppierungsergebnis nach der Methode von A. Pope. Rechts: Gruppierungsergebnis durch das schnellere, Hough-basierte Verfahren.

6.1.4 Einfluss der Umgebungsverteilung

Zur Schätzung von statistischen Größen während der Graphenanpassung wird eine so genannte *Umgebungsverteilung* verwendet (vgl. Abschnitt 2.1.3). Sie enthält eine objektunabhängige Statistik über die Häufigkeit und Ausprägung von Bildmerkmalen und wird aus Beispielbildern der Szene aufgebaut. Eine starke Abhängigkeit der Erkennungsergebnisse von der Umgebungsverteilung wurde nicht festgestellt. Anderen denkbaren Varianten leicht überlegen ist die Verwendung aller zur Verfügung stehenden Bilder zur Erzeugung der Umgebungsverteilung. Dies beinhaltet auch die zur Generierung der Modellansichten synthetisierten Objektansichten.

6.2 Untersuchungen anhand künstlicher Daten

6.2.1 Testautomatisierung

Neben Tests mit realen Aufnahmen wurden in dieser Arbeit zahlreiche vollautomatische Lokalisierungstests durchgeführt. Dazu wurde die Bühnensituation des Projektes MQube einschließlich der Beleuchtungsverhältnisse, der Objektoberflächen und einiger Störeinflüsse modelliert und Programme zur Erzeugung zufälliger Szenen entwickelt. Die Verwendung vollautomatischer Tests bietet folgende Vorteile:

- Im Vergleich zu manuellen Tests lassen sich deutlich größere Testserien verwirklichen und somit statistisch aussagekräftigere Kennzahlen messen.
- Die Grundwahrheit über die räumliche Lage des zu lokalisierenden Objektes ist exakt bekannt. Zusätzliche Messfehler aufgrund ungenauer manueller Positionierung entfallen.
- Systemparameter lassen sich automatisch optimieren. Dazu werden die Erkennungsergebnisse der verschiedenen Parameterwerte miteinander verglichen und ein Optimum ermittelt.

Bei diesem Vorgehen muss jedoch sichergestellt werden, dass die mit künstlichen Daten erzielten Ergebnisse auf reale Daten übertragbar sind. Deshalb sind Gegenproben mit Bildern eines realen Versuchsaufbaus erfolgt und belegen, dass die Vergleichbarkeit von künstlichen und realen Daten gewährleistet ist und die mit den synthetischen Daten gewonnenen Erkenntnisse auf die realen übertragbar sind. Die Ergebnisse der Untersuchungen zeigen, dass künstlich erzeugte Objektansichten insbesondere zum Lernen von ansichtsbasierten Modellen gut geeignet sind. Daneben hat sich herausgestellt, dass

die relevanten Störeinflüsse, welche die Objekterkennung in realen Bildern schwer machen, in künstliche Bilder integriert werden können. Abbildung 6.8 zeigt zwei typische synthetische Testbilder, wie sie für einige der Untersuchungen erzeugt worden sind. Folgende Einflüsse sind dabei konkret modelliert worden:

- Zufällige Positionierung des Testobjektes.
- Schattenwurf durch mehrere Lichtquellen.
- Eine variierende Anzahl von zufällig positionierten Störobjekten. Im Falle des Testobjektes *Hantel* wurden bewusst quaderförmige Störobjekte verwendet, da diese dem Objekt visuell ähnlich sind und so die Erkennung erschweren.
- Die Aufnahmequalität der Kamera durch additives Rauschen und eine gauß'sche Tiefpassfilterung. Dies hat zum Teil erheblichen Einfluss auf die Erkennung von Kanten und insbesondere Ecken.

Zur Steuerung des Testablaufs dient ein einfaches Programm, das neben der Protokollierung von Kennzahlen (wie der Geschwindigkeit der Zwischenschritte) die Gesamtergebnisse zusammenstellt und automatisch Auswertungsdiagramme erstellt.

Maß für die Lokalisierungsgenauigkeit

Um das Erkennungsergebnis automatisch bewerten zu können, ist ein Vergleich des sechsdimensionalen Lokalisierungsvektors mit dem tatsächlichen Lagevektor nötig. Dies geschieht durch Differenzbildung und Gewichtung der einzelnen Dimensionen innerhalb einer Gütefunktion. In der konkreten Anwendung dieser Arbeit, der Initialisierung eines Verfolgungsprozesses, wurde die Gütefunktion derart gewählt, dass sie die Eignung der Lokalisierungsergebnisse für den Verfolgungsprozess widerspiegelt. Die Gewichtung der Dimensionen wurde gemäß ihrer Relevanz für die Verfolgung gewählt.

Die folgenden zwei Gütefunktionen wurden implementiert und getestet. $\mathbf{L}$ bezeichnet dabei den geschätzten und $\mathbf{T}$ den tatsächlichen Lagevektor. δ_z steht für den Abstand der beiden Lagen in Richtung der z-Achse, δ_{xy} für ihren Abstand senkrecht dazu (parallel zur Bildebene). δ_α bezeichnet die Differenz der beiden Orientierungswinkel im Gradmaß. f_z, f_{xy} und f_α sind Gewichtungsfaktoren:

$$\begin{aligned} \text{linear} \quad &: \quad g_l(\mathbf{L},\mathbf{T}) := \max(0, 1 - (f_z\,\delta_z + f_{xy}\,\delta_{xy} + f_\alpha\,\delta_\alpha)) \;, \\ \text{quadratisch} \quad &: \quad g_q(\mathbf{L},\mathbf{T}) := \max(0, 1 - (f_z\,\delta_z^2 + f_{xy}\,\delta_{xy}^2 + f_\alpha\,\delta_\alpha^2)) \;. \end{aligned}$$

Das quadratische Maß g_q hat sich im Kontext der Objektverfolgung als geeigneter erwiesen, da hier die starke Abweichung einer einzelnen Dimension deutlicher zum Tragen kommt.

Wie im nächsten Abschnitt gezeigt wird, führt die durch den Erkennungsprozess am höchsten bewertete Modellansicht nicht notwendigerweise zum besten Lokalisierungsergebnis. Aus diesem Grund sollen in der konkreten Anwendung vorsichtshalber die fünf besten Erkennungsergebnisse an den Verfolgungsprozess weitergereicht werden. So ergibt sich eine Steigerung der Sicherheit für den unwahrscheinlichen Fall, dass die Verfolgung nicht am besten Ergebnis ansetzen kann. Als Bewertung der Lokalisierungsergebnisse von automatischen Tests wird der jeweils höchste Wert g_q dieser fünf Modellansichten verwendet.

Die Gewichtungsfaktoren der quadratischen Gütefunktion g_q wurden folgendermaßen gewählt:

$$\begin{aligned} f_z &= \frac{1}{250000}\,, \\ f_{xy} &= \frac{1}{50000}\,, \\ f_\alpha &= \frac{1}{10000}\,. \end{aligned}$$

In dieser Wahl spiegelt sich die Relevanz der einzelnen Dimensionen für die Objektverfolgung wieder. Der Versatz parallel zur Bildebene wird stärker gewichtet als der in Blickrichtung. Beispielsweise ergibt sich für die Abweichungen $\delta_z = 45$ cm, $\delta_{xy} = 8$ cm und $\delta_\alpha = 20°$ ein Wert von $g_q = 0{,}022$, der als gerade noch akzeptabel angesehen werden kann.

Testbildmengen

Die Untersuchungen der folgenden Abschnitte wurden anhand der zwei folgenden künstlich erzeugten Datensätze durchgeführt:

1. *M3-Hantel-Synth-1*: 100 Bilder von typischen Lokalisierungssituationen aus dem Projekt MQube. Eine einzelne Hantel liegt ca. 2 Meter vor der Kamera entfernt auf einer beleuchteten Ebene. Zwei Beispielbilder aus dieser Testbildmenge sind in Abbildung 6.7 dargestellt.
2. *M3-Hantel-Synth-2*: 100 Bilder von komplexeren Szenen. Die Hantel wird zufällig im Raum positioniert und gedreht. Im Hintergrund werden quaderförmige Störobjekte verteilt. Zwei Beispielbilder aus dieser Testbildmenge sind in Abbildung 6.8 dargestellt.

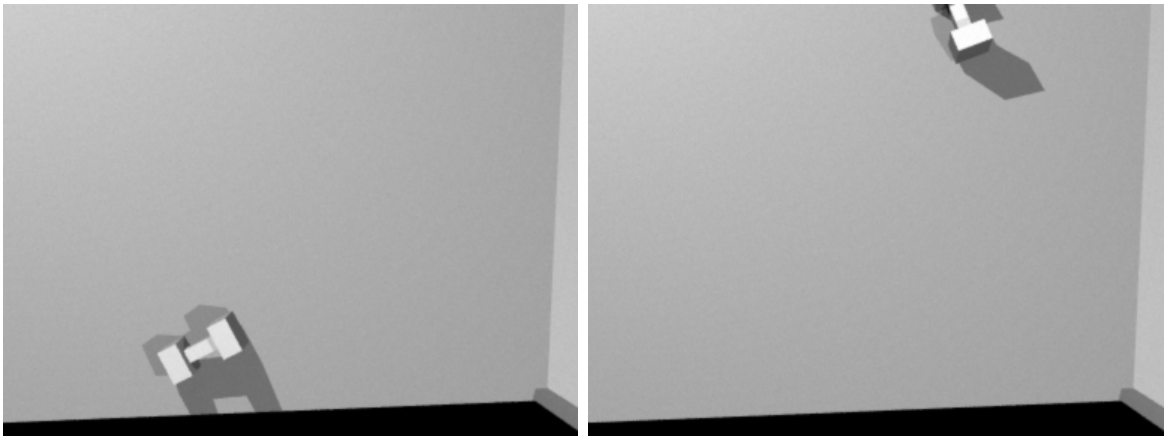

Abbildung 6.7: Testbild 2 (links) und Testbild 75 (rechts) der Testbildmenge *M3-Hantel-Synth-1*.

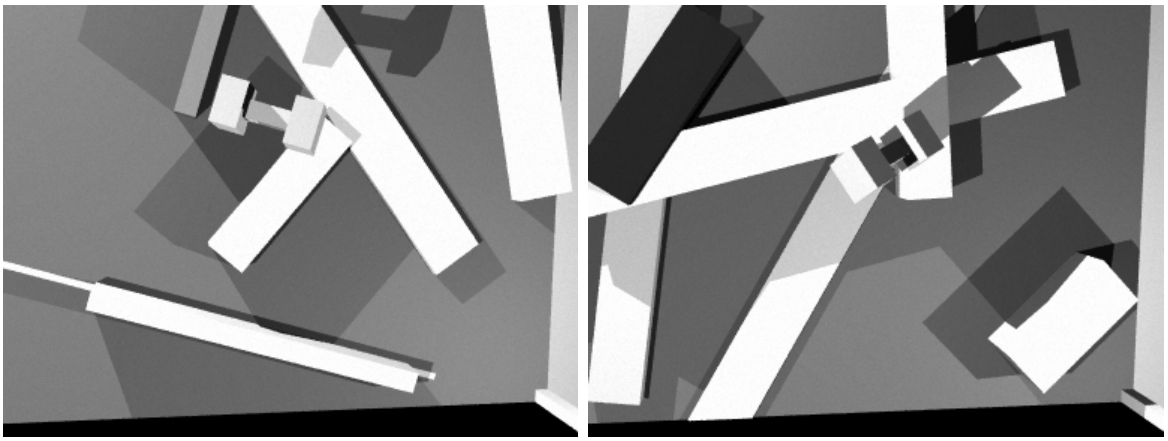

Abbildung 6.8: Testbild 68 (links) und Testbild 83 (rechts) der Testbildmenge *M3-Hantel-Synth-2*.

6.2.2 Verlässlichkeit des Verifikationsmaßes

In Kapitel 2 wird beschrieben, wie anhand eines Verifikationsmaßes angegeben werden kann, wieviel Vertrauen in ein konkretes Erkennungsergebnis zu setzen ist. Mit automatischen Tests, wo die Grundwahrheit über die Lage des Objektes präzise bekannt ist, lässt sich die Leistung dieses Verifikationsmaßes überprüfen. Bei Testbildmenge *M3-Hantel-Synth-2* ist in 20% der Fälle keine Erkennung möglich. Abbildung 6.9 veranschaulicht den Zusammenhang zwischen dem Verifikationsmaß (blaue Rauten) und dem in Kapitel 6.2.1 eingeführten Lokalisierungsmaß g_q (rote Quadrate). Zur deutlichen Hervorhebung des Zusammenhangs wurden die Testergebnisse absteigend nach dem Lokalisierungsmaß sortiert.

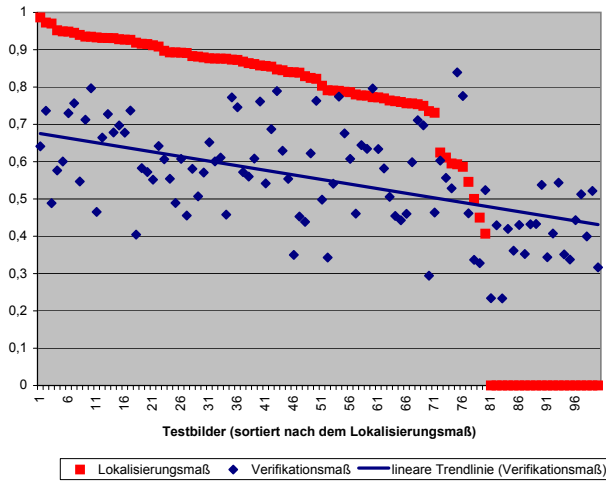

Abbildung 6.9: Untersuchungsergebnisse zur Verlässlichkeit des Verifikationsmaßes.

Von einer *erfolgreichen Lokalisierung* soll gesprochen werden, wenn das Lokalisierungsmaß g_q einen Wert größer als 0 besitzt. Verwendet man die von A. Pope vorgeschlagene Verifikationsmaß-Schwelle von 0,5 für die Entscheidung, ob die Erkennung erfolgreich war oder nicht, so ist diese Aussage in 76% der Fälle konsistent mit der Erfolgsaussage durch das Lokalisierungsmaß. Für eine Schwelle von 0,45 erhöht sich diese Quote auf 88%.

Kritisch ist über das Verifikationsmaß anzumerken, dass in den Untersuchungen dieser Arbeit keine starke Trennschärfe nachgewiesen werden konnte, wie sie in den Experimenten von Pope genannt wird. Der Grund hierfür dürfte in der höheren Auflösung liegen, die Pope für seine Test- und Trainingbilder verwendet. Bei höherer Auflösung wirken sich Lageveränderungen stärker auf das Verifikationsmaß aus, da dieses aus der Überlappung von Modell- und Bildkurven gebildet wird.

An der liniearen Trendlinie in Abbildung 6.9 lässt sich erkennen, dass das Verifikationsmaß mit dem Lokalisierungsmaß korreliert ist. Allerdings ist der Korrelationskoeffizient (0,5806) zu niedrig, um verlässlich aus dem Verifikationsmaß auf das Lokalisierungsmaß schließen zu können. Da aus diesem Grund auch Modellansichten mit niedrigerem Verifikationsmaß zu besseren Lokalisierungsergebnissen führen können, werden in der akutellen Systemimplementierung und den Untersuchungen dieses Kapitels die besten fünf Modellansichten (gemessen am Verifikationsmaß) als potentielle Erkennungsergebnisse angesehen und in absteigender Reihenfolge ausgegeben.

Da das Lokalisierungsmaß aus der bekannten Grundwahrheit über die Objektlage gebildet wird, ist seine Aussagekraft deutlich höher als die des Verifikationsmaßes. Für die Auswertung von Testergebnissen wird daher das Lokalisierungsmaß verwendet. In der Anwendung hingegen ist sein Einsatz aufgrund der unbekannten Objektlage nicht möglich und das Verifikationsmaß bietet eine akzeptable Alternative.

6.2.3 Lokalisierungsleistung und Genauigkeit

Um die Lokalsierungsgenauigkeit des Systems zu überprüfen, werden automatische Tests mit den beiden Testbildmengen *M3-Hantel-Synth-1* und *M3-Hantel-Synth-2* durchgeführt. Auf den Diagrammen der Abbildung 6.10 ist für jedes der insgesamt 200 Testbilder das Lokalisierungsmaß des Erkennungsergebnisses aufgetragen.

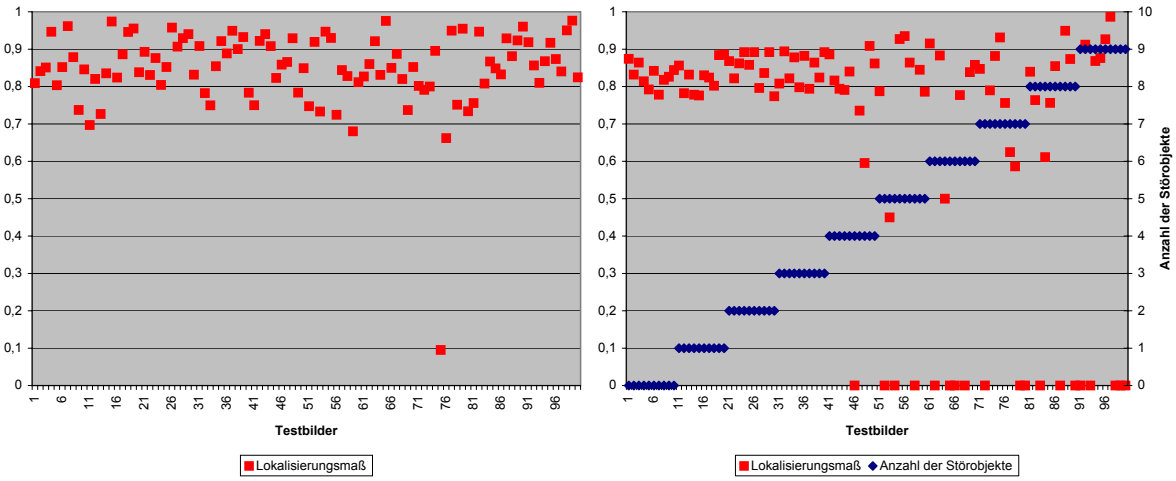

Abbildung 6.10: Untersuchungsergebnisse zur Lokalisierungsleistung und Genauigkeit. Links: Ergebnisse mit der Testbildmenge *M3-Hantel-Synth-1*. Rechts: Ergebnisse mit der Testbildmenge *M3-Hantel-Synth-2*. Die Anzahl der zufällig generierten Störobjekte ist mit blauen Rauten aufgetragen.

Die aus denselben Testergebnissen generierten Diagramme in Abbildung 6.11 demonstrieren den Zusammenhang zwischen dem Lokalisierungsmaß und seinen Komponenten δ_z, δ_{xy}, δ_α. Der Übersichtlichkeit halber wurden die Testbilder absteigend nach dem Lokalisierungsmaß sortiert und die Werte der Komponenten in Form der zugehörigen linearen Trendlinien eingezeichnet.

Abbildung 6.12 verdeutlicht speziell den Zusammenhang zwischen dem Lokalisierungsmaß und seinen Komponenten δ_z und δ_{xy}. In dieser Darstellung wurden die Testbilder absteigend nach dem Lokalisierungsmaß sortiert.

Die Ergebnisse führen zu folgenden Beobachtungen und Interpretationen:

- Gemäß der Erwartungen nimmt mit zunehmender Komplexität der Szene die Anzahl der gescheiterten Erkennungsversuche sowie die Varianz der Lokalisierungsgenauigkeit zu. Betrachtet man die Erkennungsergebnisse in ihrer Gesamtheit, ist die Erkennungsleistung als gut einzustufen.
- Die Lokalisierung auf der Testbildmenge *M3-Hantel-Synth-1* schlägt nur in einem

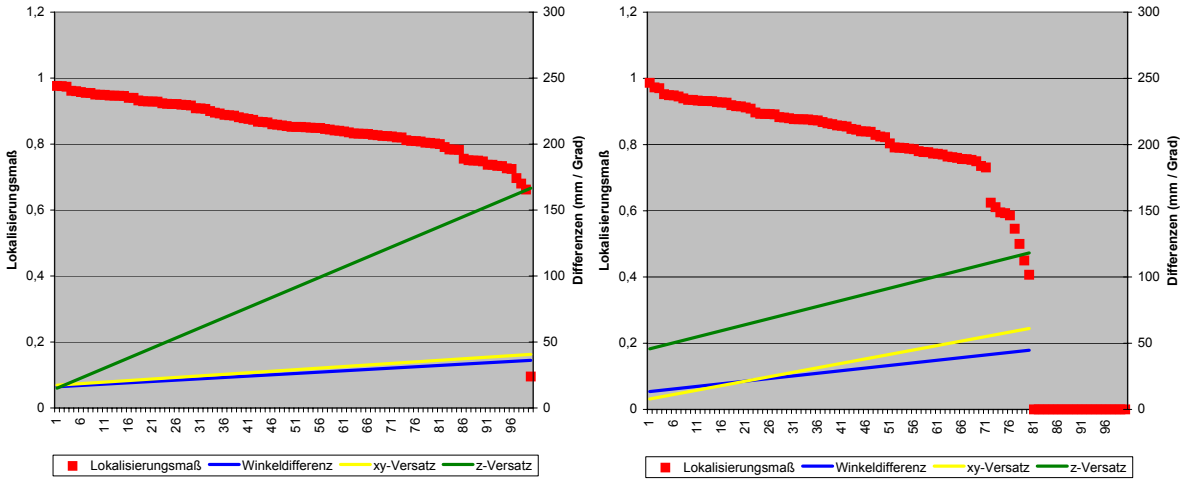

Abbildung 6.11: Zusammensetzung des Lokalisierungsmaßes. Links: Ergebnisse auf der Testbildmenge *M3-Hantel-Synth-1*. Rechts: Ergebnisse auf der Testbildmenge *M3-Hantel-Synth-2*.

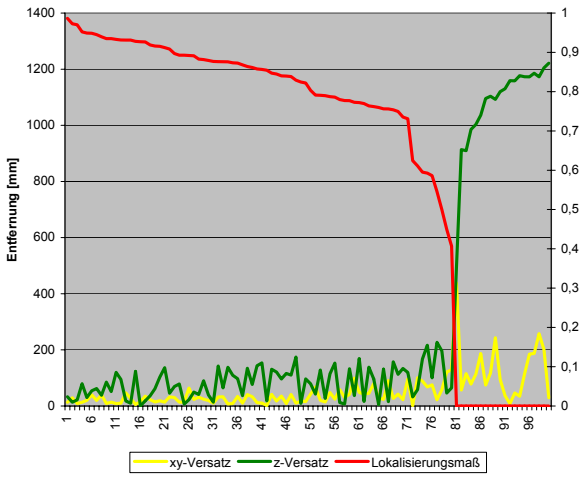

Abbildung 6.12: Entfernungskomponenten des Lokalisierungsmaßes bei auf der Testbildmenge *M3-Hantel-Synth-2*.

Bild fehl (dieses ist in Abbildung 6.7 rechts dargestellt). Auf diesem Bild ist zufallsbedingt nur ein kleiner Teil des Objektes sichtbar, da es am äußeren Rand der Szene positioniert wurde.

- Die Lokalisierungsgenauigkeit auf der Testbildmenge *M3-Hantel-Synth-1* ist stärker gestreut (Mittelwert 0,8489 und Standardabweichung 0,1059) als auf der ersten Hälfte der Testbildmenge *M3-Hantel-Synth-2*, wo die Lokalisierung robust möglich ist (Mittelwert 0,8357 und Standardabweichung 0,0383). Grund hierfür ist die in der ersten Testbildmenge deutlich größere Entfernung der Testobjekte

von der Kamera, was Erkennung und Lokalisierung schwerer macht.

- Bei den 20% der Testbilder aus *M3-Hantel-Synth-2*, bei denen die Erkennung nicht möglich ist, liegt die geschätzte Objektposition nur wenige Zentimeter von der Kamera entfernt, was sich in großen Werten für δ_z in Abbildung 6.12 äußert. Diese Fehlerkennungen könnten voraussichtlich durch Einbringen von Vorwissen über die Objektlage deutlich eingeschränkt werden.

- Der Versatz δ_{xy} parallel zur xy-Ebene hat dieselbe Größenordnung wie die Differenz δ_α zwischen der tatsächlichen und der geschätzten Orientierung. Daher werden die entsprechenden Gewichtungsfaktoren der Gütefunktion für weitere Tests zugunsten der Winkeldifferenz verändert, da beispielsweise eine Abweichung von $\delta_{xy} = 5\ cm$ für das Verfolgungssystem viel besser zu kompensieren ist als die dem bisher entsprechende Winkelabweichung von $\delta_\alpha = 50°$.

6.2.4 Zeitkomplexität

Bei den oben beschriebenen Tests auf einem Testrechner mit der Taktfrequenz 3 GHz wurden die in Abbildung 6.13 dargestellten Zeiten für die einzelnen Verarbeitungsschritte gemessen. Deutlich zu erkennen ist die Abnahme der Erkennungsgeschwindigkeit bei zunehmender Zahl von Hintergrundobjekten. Grund hierfür ist die stark anwachsende Zahl der Initialpaare sowie bei scheiternder Erkennung die Tatsache, dass eine hohe Zahl von Hypothesen betrachtet wird, da die Suche nicht vorzeitig mit einer gefundenen Lösung terminieren kann. Zum hohen Zeitaufwand des Vorverarbeitungsschrittes siehe Abschnitt 6.1.3.

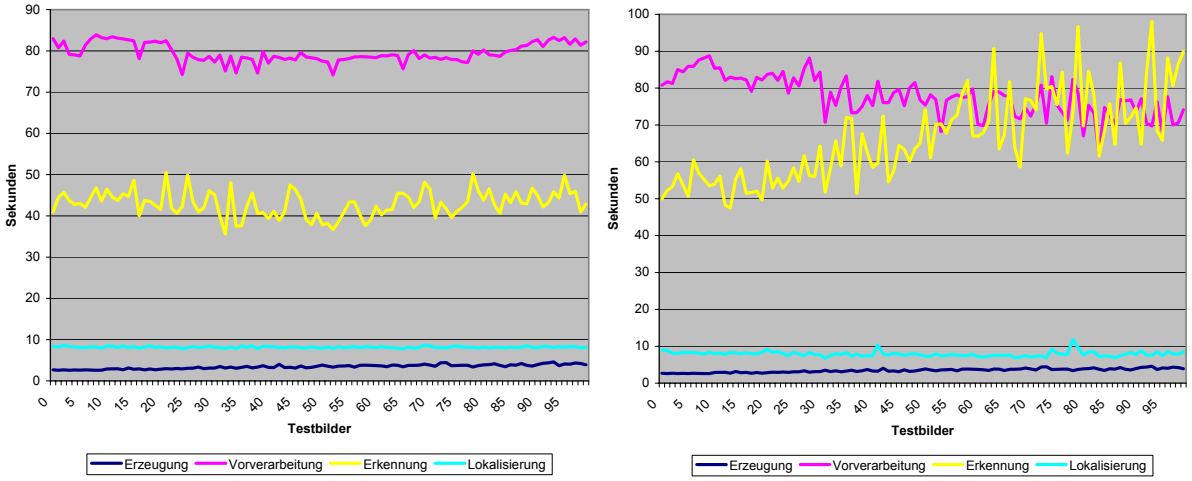

Abbildung 6.13: Zeitbedarf der einzelnen Erkennungsschritte auf einem Testrechner der Taktfrequenz 3 GHz. Links: Zeitbedarf zur Erkennung bei der Testbildmenge *M3-Hantel-Synth-1*. Rechts: Zeitbedarf zur Erkennung bei der Testbildmenge *M3-Hantel-Synth-2*.

6.2.5 Einfluss der Trainingsmenge auf das Objektmodell

Anhand der Testbildmenge *M3-Hantel-Synth-1* wird untersucht, welchen Einfluss unterschiedlich große Trainingsmengen auf die Menge der Modellansichten und somit auf das Erkennungsergebnis haben. Zwei typische Objektansichten, wie sie für die folgenden Trainingsmengen generiert wurden, sind in Abbildung 4.9 auf Seite 45 dargestellt. Es wurden folgende Mengen von Objektansichten erzeugt und daraus Modellansichten gelernt:

1. *Hantel-Ansichten-34*: Aus 34 Ansichten des Objektes *Hantel* wurden 11 Modellansichten gelernt. Diese sind in Anhang C.1 abgebildet. Besonders auffällig ist eine entartete Modellansicht (das 10. Bild in der Abbildung von Anhang C.1) die aufgrund einer suboptimalen Anpassung zustande gekommen ist. Eine solche Modellansicht kann aufgrund schlechter Passung ihrer Beispielansichten leicht identifiziert und aus der Menge entfernt werden. In keiner Untersuchung wurden jedoch an solchen entarteten Modellansichten hohe Verifikationsmaße beobachtet, weshalb die Erkennungsergebnisse durch deren Existenz nicht beeinflusst wurden.

2. *Hantel-Ansichten-66*: Aus 66 Objektansichten ergaben sich 7 Modellansichten (siehe Anhang C.1). Diese Zahl liegt deutlich unter der bei einer Trainingsmenge von 34 Ansichten, da hier die Symmetrie des Objektes aufgrund der speziellen Verteilung der Blickrichtungen auf das Objekt stark zum Tragen kommt. Die 66 Objektansichten lassen sich in 7 Gruppen von exakt gleichen Ansichten einteilen, welche unmittelbar zu Modellansichten zusammengefügt werden.

3. *Hantel-Ansichten-130*: Aus 130 Objektansichten ergaben sich 18 Modellansichten (siehe Anhang C.1). Diese Menge von Modellansichten ergab auf künstlichen und realen Daten die besten Ergebnisse.

Abbildung 6.14 zeigt die Lokalisierungsgenauigkeiten, die mit den drei Sätzen von Modellansichten auf der Testbildmenge *M3-Hantel-Synth-1* erzielt werden konnten. Die Testbilder wurden dabei absteigend nach dem Lokalisierungsmaß bei 130 Trainingsansichten sortiert. 130 Trainingsansichten erzielen die beste und stabilste Lokalisierung (Median 0,8522 und Standardabweichung 0,1059), gefolgt von 66 Trainingsansichten (Median 0,8184 und Standardabweichung 0,2234) und 34 Trainingsansichten (Median 0,7306 und Standardabweichung 0,2388).

Mit Mengen von Modellansichten, die aus mehr als 130 Trainingsansichten gelernt werden, wurde keine weitere Steigerung der Lokalisierungsleistung erreicht. Bei deutlich erhöhtem Aufwand während der Lernphase liegen die Ergebnisse im Bereich der mit 130 Trainingsansichten erzielten. Selbstverständlich hängt die optimale Anzahl von Trainingsansichten vom speziellen Objekt ab. Bei komplexeren Objekten werden mehr

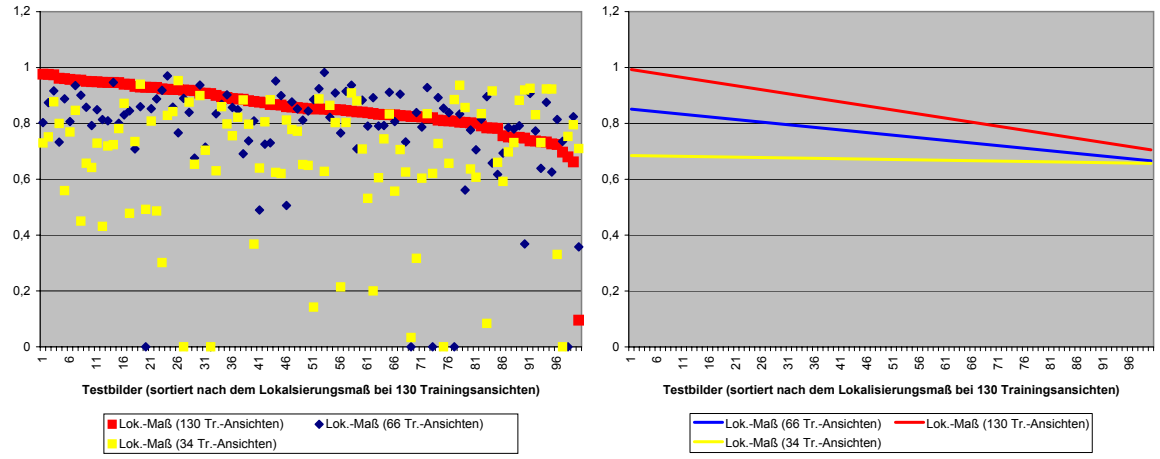

Abbildung 6.14: Vergleich der Lokalisierungsleistungen bei den drei gewählten, unterschiedlich großen Trainingsmengen. Das rechte Diagramm zeigt die linearen Trendlinien zu den Datenpunkten des linken Diagramms. Rot: 130 Trainingsansichten. Blau: 66 Trainingsansichten. Gelb: 34 Trainingsansichten.

Objektansichten benötigt, um eine akkurate Repräsentation durch Modellansichten zu erhalten. Zur Ermittlung des Bedarfs an Trainingsansichten für ein spezielles Objekt ließe sich der oben beschriebene automatische Test gut verwenden.

Abbildung 6.15 stellt den Zeitbedarf für die Erkennung mit den unteschiedlichen Objektmodellen dar. Man beachte, dass die Erkennung mittels des Modells aus 34 Ansichten mehr Zeit in Anspruch nimmt als bei 66 Ansichten. Dies liegt an der höheren Zahl von Modellansichten bei den 34 Trainingsansichten, was einen direkten Einfluss auf die Laufzeit hat.

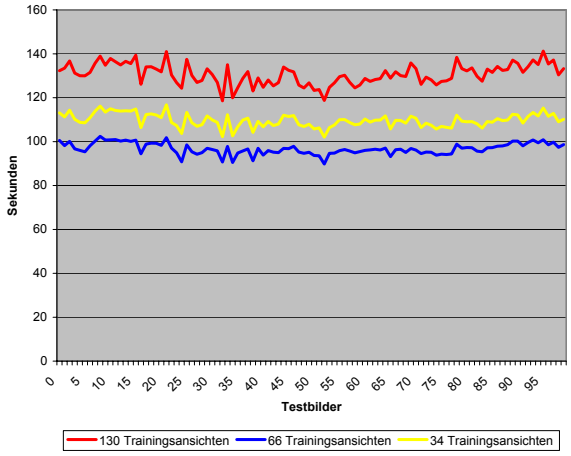

Abbildung 6.15: Zeitbedarf für die Erkennung bei unterschiedlich großen Trainingsmengen.

6.3 Ergebnisse bei realen Daten

Um den Anwendungsbereich des Systems zu definieren werden Untersuchungen mit verschiedenen realen Objekten durchgeführt. Zu Beginn wird eine quantitative Untersuchung der Lokalisierungsleistung anhand des im Projekt MQube besonders wichtigen Objektes *Hantel* dargestellt.

6.3.1 Lokalisierungsleistung, Genauigkeit und Ergebnisse beim Objekt *Hantel*

Die Testbildmenge *M3-Hantel-Real* (siehe Anhang C.2) enthält 13 typische Testbilder des Objektes *Hantel* aus dem Projekt MQube. Auf neun Bildern ruht die Hantel in derselben Lage und wird durch einströmenden Nebel teilweise eingehüllt. Auf drei weiteren Bildern verdeckt die führende Hand das Objekt partiell. Zur Abschätzung der Lokalisierungsgenauigkeit wurde die in Kapitel 6.2.3 vorgeschlagene Änderung der Gewichtungsfaktoren umgesetzt und folgende Werte bei der quadratischen Gütefunktion g_q verwendet:

$$\begin{aligned} f_z &= \frac{1}{3 \cdot 200^2} , \\ f_{xy} &= \frac{1}{3 \cdot 100^2} , \\ f_\alpha &= \frac{1}{3 \cdot 25^2} . \end{aligned}$$

Die Lokalisierungsergebnisse sind in Abbildung 6.16 aufgetragen. Die berechneten Lagevektoren hängen mit den tatsächlichen folgendermaßen zusammen:

- Versatz der Position in z-Richtung: Median 58,92 mm, Standardabweichung 43,07 mm.
- Versatz der Position in xy-Richtung: Median 36,49 mm, Standardabweichung 35,40 mm.
- Differenz der Orientierungswinkel: Median 23,70°, Standardabweichung 10,88°.

Abbildung 6.17 zeigt die beiden Testbilder 11 und 13 mit der jeweils höchstbewerteten Modellansicht. Erwartungsgemäß decken sich die Modellansichten dabei, auch trotz der

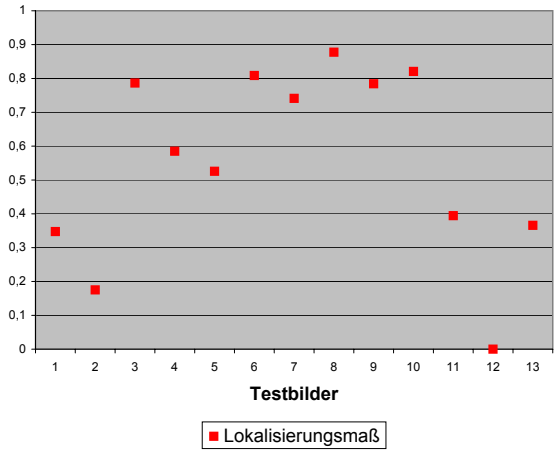

Abbildung 6.16: Die Lokalisierungsleistung auf der Testbildmenge *M3-Hantel-Real*.

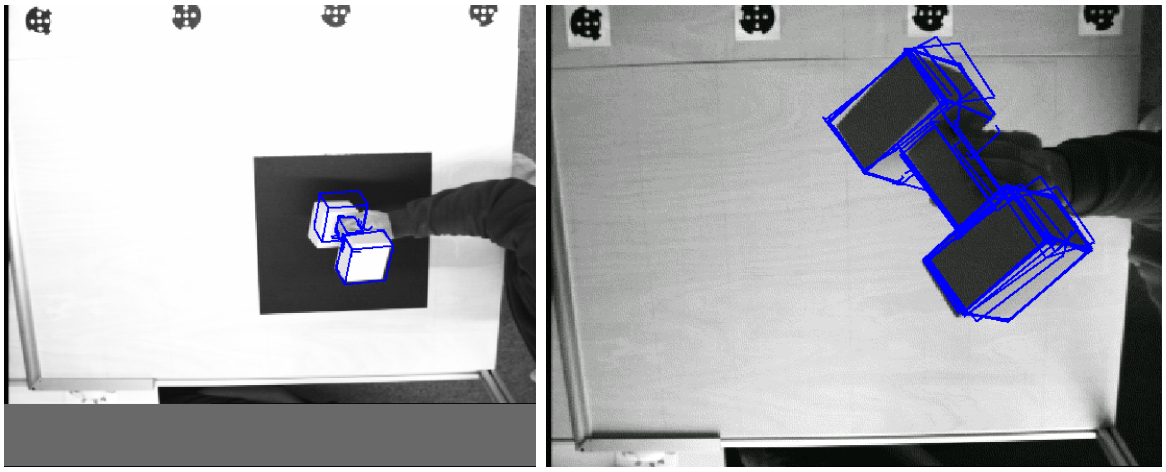

Abbildung 6.17: Ergebnisse des ansichtsbasierten Erkennungsverfahrens für Testbild 11 (links) und 13 (rechts) der Testbildmenge *M3-Hantel-Real*. Den Testbildern sind die höheren Merkmale der angepassten Modellansichten in blau überlagert.

erfolgreichen Erkennung, nicht exakt mit den jeweiligen Objektabbildern. Die aus diesem Erkennungsergebnis berechnete räumliche Lage der Objekte ist jedoch hinreichend genau, um dem Verfolgungsprozess die Erfassung der Objekte zu ermöglichen.

Anhand der Testbilder 2 und 10 (vgl. Abbildung 6.18) lässt sich deutlich machen, dass die Lokalisierungsgenauigkeit nicht nur von der Bildqualität und der Stärke der Störungen abhängt, sondern auch von der Qualität derjenigen Modellansicht, die das höchste Verifikationsmaß erhält und somit als Erkennungsergebnis betrachtet wird. Beispielsweise kann die Hantel in Bild 10 genauer lokalisiert werden (g_q=0,82, siehe Abbildung 6.18 rechts) als in Bild 2 (g_q=0,18, siehe Abbildung 6.18 links), obwohl Bild 10 durch Nebel stark gestört ist. Die unterschiedliche Lokalisierungsgenauigkeit

beruht auf der Tatsache, dass sich trotz derselben tatsächlichen Objektlage in Bild 2 und Bild 10 unterschiedliche Modellansichten als Erkennungsergebnis ergeben haben. Die an Bild 10 angepasste Modellansicht passt besser auf die tätsächliche Objektlage und resultiert dadurch in einer genaueren Lokalisierung. Man beachte in diesem Zusammenhang, dass zur Wahl der besten Modellansicht allein das Verifikationsmaß der Graphenanpassung verwendet wird, dessen Betrag keinen zuverlässigen Aufschluss über die Lokalisierungsgenauigkeit gibt, wie Abschnitt 6.2.2 gezeigt hat.

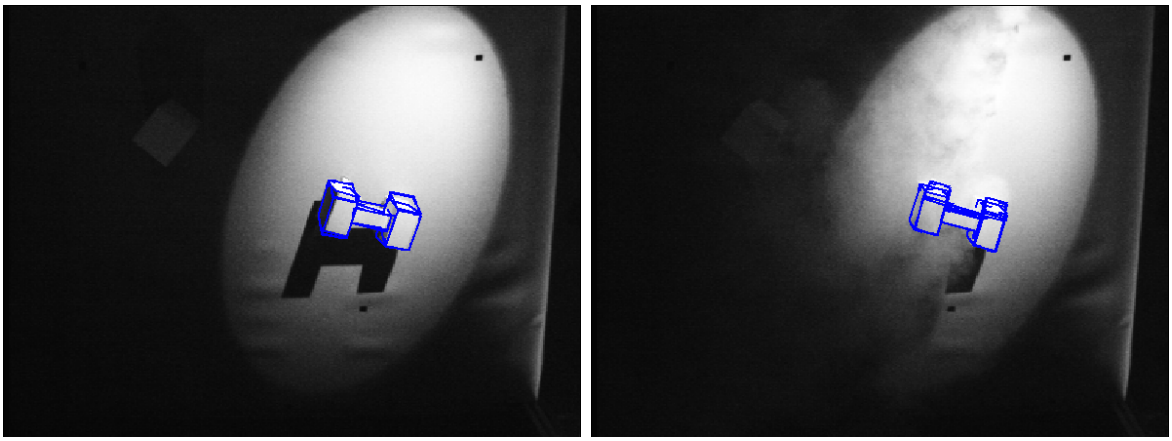

Abbildung 6.18: Ergebnisse des ansichtsbasierten Erkennungsverfahrens für Testbild 2 (links) und 10 (rechts) der Testbildmenge *M3-Hantel-Real*. Den Testbildern sind die höheren Merkmale der angepassten Modellansichten in blau überlagert.

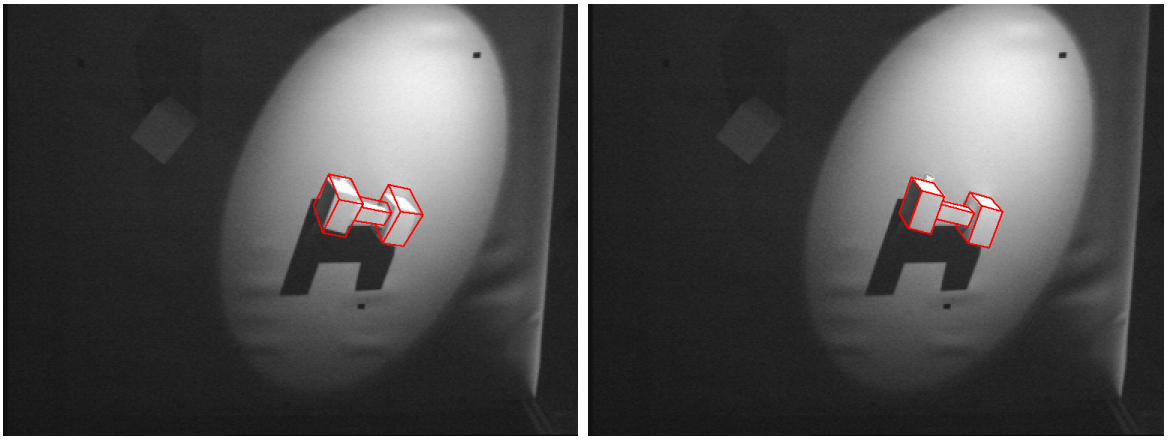

Abbildung 6.19: Berechnete Objektlage und deren Verbesserung durch den Verfolgungsprozess für Testbild 2 der Testbildmenge *M3-Hantel-Real*. Links: die durch das Lokalisierungssystem berechnete räumliche Objektlage. Das 3D-Formmodell ist dem Testbild in rot überlagert. Rechts: die räumliche Objektlage nach Anwendung des Verfolgungsprozesses.

Die erzielten Lokalisierungsergebnisse sind hinreichend genau für die Initialisierung des Verfolgungsprozesses aus dem Projekt MQube. Der Verfolgungsprozess kann die

durch die ansichtsbasierte Lokalisierung nur grob berechnete Objektlage verfeinern, wie durch Abbildung 6.19 veranschaulicht wird. In dieser Abbildung ist auf der linken Seite das Testbild 2 mit der durch das Lokalisierungssystem berechneten Objektlage dargestellt. Auf der rechten Seite sind Testbild und Objektlage nach Anwendung des Verfolgungsverfahrens zu sehen. Die durch den Verfolgungsprozess erreichte Verbesserung der Lageschätzung ist dabei deutlich zu erkennen.

6.3.2 Ergebnisse bei Freiformobjekten

Eine besondere Herausforderung für Objekterkennungssysteme stellen Objekte aus vorherrschend gekrümmten Strukturen dar. Im Allgemeinen lassen sich geradlinige Strukturen leichter extrahieren, repräsentieren und zuordnen. Für den hier gewählten Erkennungsansatz ist die robuste Identifikation von Ecken Voraussetzung für eine hohe Erkennungsleistung. Ecken, d.h. Stellen maximaler Krümmung, sind in gekrümmten Strukturen schwerer zu identifizeren und meist nur unsicher lokalisierbar. Die Abbildungen 6.20 und 6.21 zeigen die zwei Objekte *Dinosaurier* und *Ente* aus voherrschend gekrümmten Strukturen, mit denen das System getestet wurde. Es sind jeweils zwei künstlich erzeugte Objektansichten unterschiedlicher Auflösung, ein künstlich erzeugtes Silhouettenbild und eine gelernte Modellansicht dargestellt. Die Modellansichten wurden aus jeweils 15 Trainingsbildern gelernt, deren Blickrichtungen um ca. 10-20 variieren.

Wie in Kapitel 6.1.3 dargestellt wurde, führt die Verwendung von Kreissegmenten bei dieser Klasse von Objekten zu keiner Steigerung der Erkennungsleistung, weshalb auch hier ausschließlich mit geradlinigen Strukturen gearbeitet wird. Mit dem Objekt *Dinosaurier* konnten deutlich bessere Erkennungsergebnisse erzielt werden als mit dem Objekt *Ente*. Ein Grund hierfür ist die höhere Zahl an extrahierbaren Initialmerkmalen, welche ja für die Erkennung von zentraler Bedeutung sind. Trotz komplexem Hintergrund konnte das Objekt *Dinosaurier* im Testbild (in Abbildung 6.22 links) lokalisiert werden (Ergebnis in der Abbildung rechts). Abbildung 6.23 zeigt eine typische Situtation, in der das Objekt *Ente* trotz kontrastreicher Kontur nicht identifiziert werden konnte.

6.3.3 Ergebnisse bei weiteren Objekten

Objekt *Rohrkombination*

Mit dem in Abbildung 6.25 links dargestellten Objekt aus dem industriellen Umfeld wurden für eine konkrete Anfrage eines Fahrzeugteile-Herstellers Erkennungsversuche

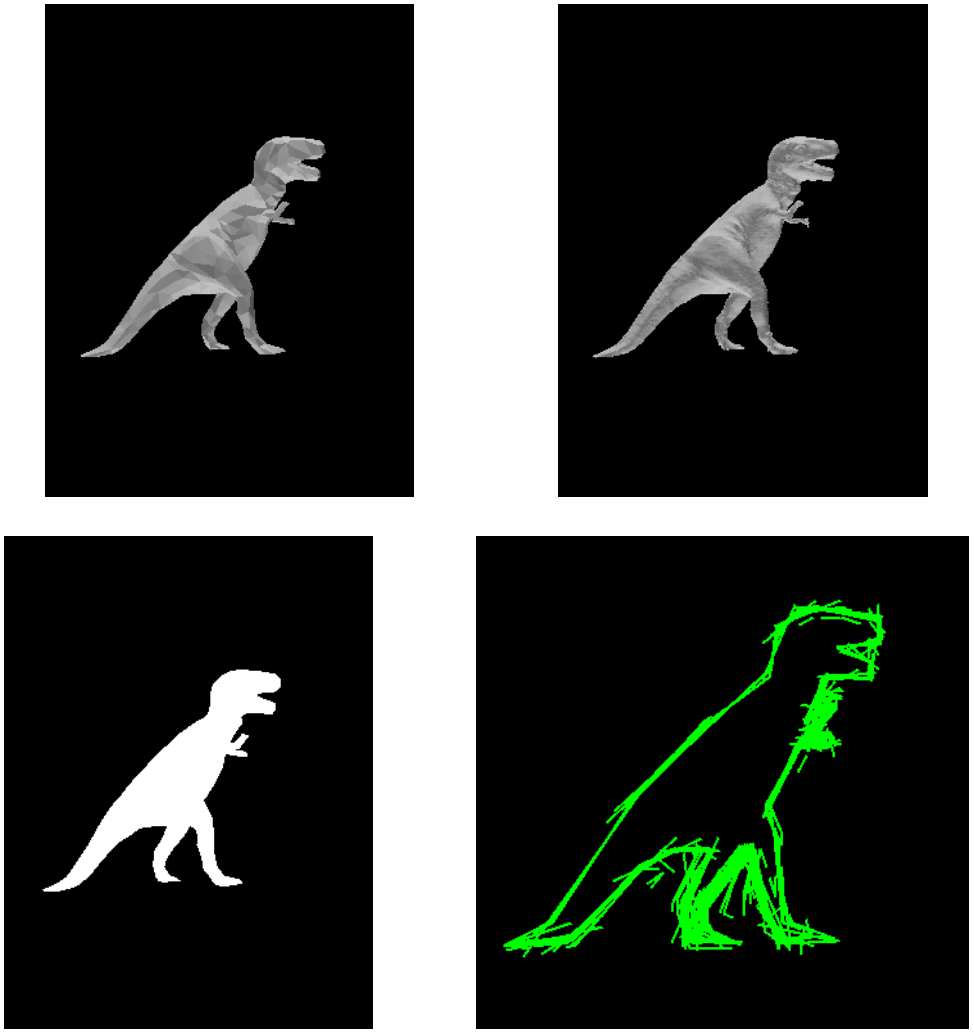

Abbildung 6.20: Das Objekt *Dinosaurier*. Von links oben nach rechts unten: Trainingsansicht mit niedriger Objektauflösung, Trainingsansicht mit hoher Objektauflösung, Trainingsansicht mit Objektsilhouette, gelernte Modellansicht (grün: Initialmerkmale).

mit gutem Ergebnis durchgeführt[5]. Eine Schwierigkeit dieses Objektes liegt in der reflektierenden Oberfläche, welche die robuste Extraktion von Bildmerkmalen erschwert. Desweiteren wird die Erkennung durch die Tatsache erschwert, dass bei der gegebenen Rohrlänge, dem Kameraabstand und dem Öffnungswinkel der Kamera perspektivische Verzerrungen auftreten, wenn das Objekt zum Bildrand bewegt wird.

In einem Versuch wurde die Rohrkombination unter verschiedenen Rotationen mit einer Kamera aufgenommen und daraus eine Modellansicht gelernt (siehe Abbildung 6.24). Als Vorverarbeitungsschritt zur Abschwächung der Reflexionsstörungen wurde die Ob-

[5]In einem möglichen Projekt geht es um die Lokalisierung von Rohrteilen für die Fahrzeugindustrie auf einem Fließband, um sie mit einem Roboter greifen zu können.

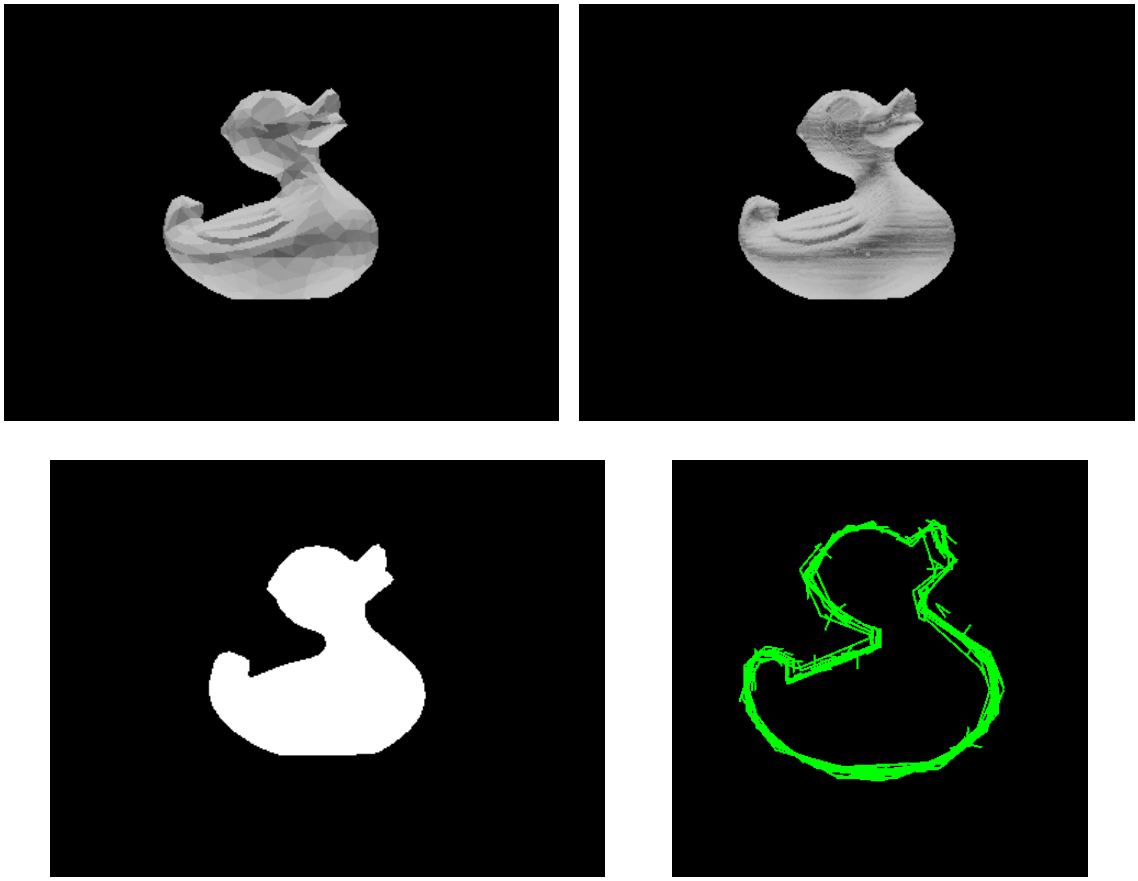

Abbildung 6.21: Das Objekt *Ente*. Von links oben nach rechts unten: Trainingsansicht mit niedriger Objektauflösung, Trainingsansicht mit hoher Objektauflösung, Trainingsansicht mit Objektsilhouette, gelernte Modellansicht (grün: Initialmerkmale).

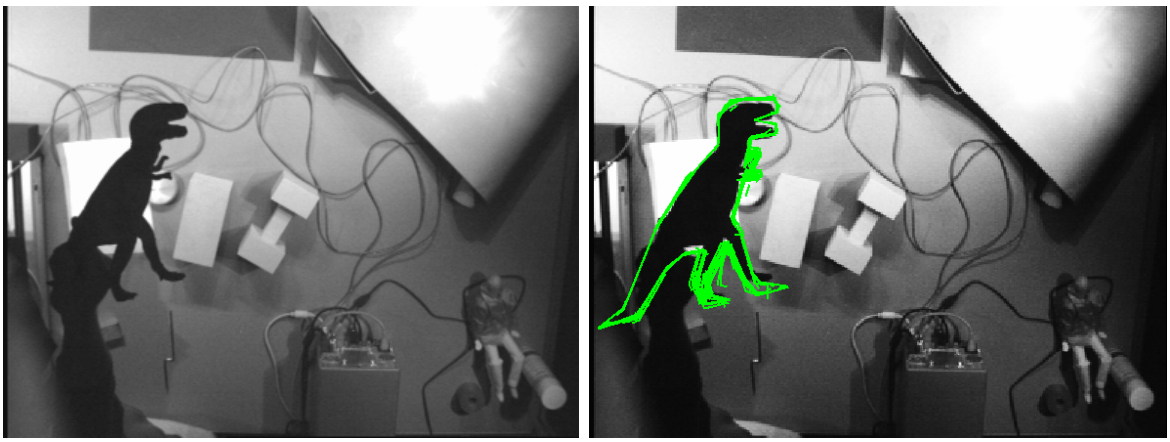

Abbildung 6.22: Links: Testbild mit Abbild des Objektes *Dinosaurier*. Rechts: Testbild mit überlagertem Erkennungsergebnis (grün: Initialmerkmale der Modellansicht).

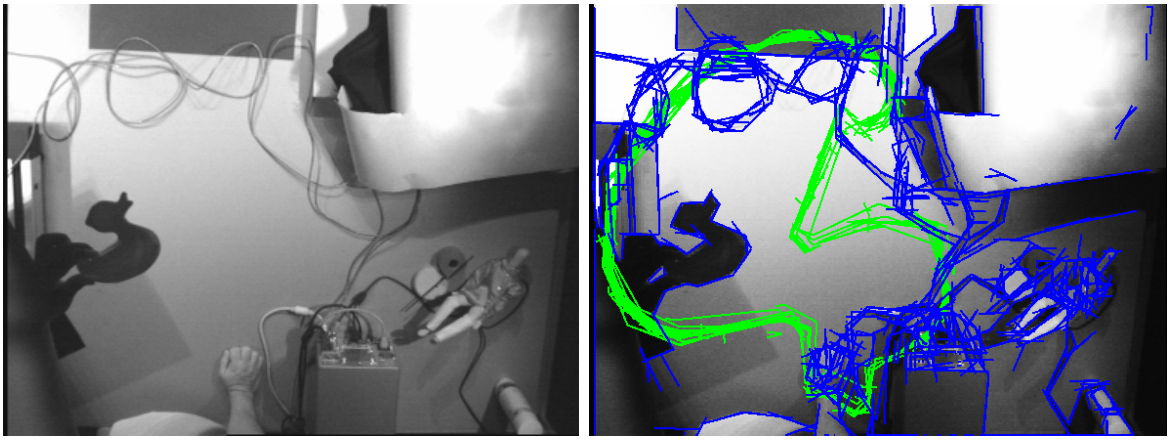

Abbildung 6.23: Links: Testbild mit Abbild des Objektes *Ente*. Rechts: Testbild mit überlagertem Erkennungsergebnis (grün: Initialmerkmale der Modellansicht, blau: Initialmerkmale des Testbildes).

jektkontur extrahiert und ein entsprechendes Binärbild erstellt.

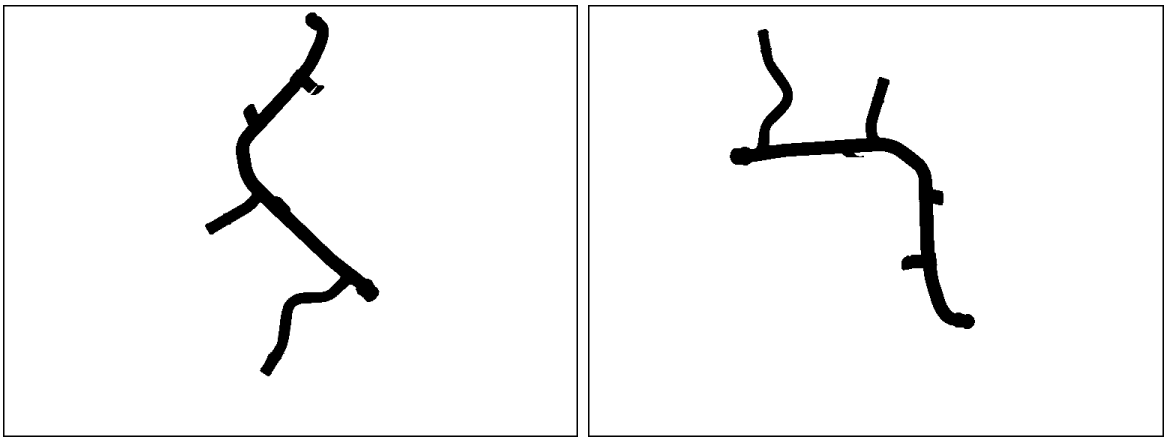

Abbildung 6.24: Mit einer Kamera aufgenommene Trainingsbilder des Objektes *Rohrkombination*. Zur Elimination von Reflexionsstörungen wurde die Objektkontur extrahiert und ein entsprechendes Binärbild erstellt.

Trotz oben genannter Erschwernisse konnte die Erkennungsaufgabe mit gutem Erfolg bewerkstelligt werden. Abbildung 6.25 zeigt auf der rechten Seite das Erkennungsergebnis in einem Testbild, auf dem das Objekt zum rechten Bildrand verschoben wurde. Die Erkennung war auch ohne den oben genannten Vorverarbeitungsschritt auf dem ursprünglichen Grauwertbild möglich. Die perspektivische Verzerrung des Objektabbilds durch die veränderte Lage ist in der Abbildung sichtbar. Trotz der Ungenauigkeit der Modellansicht durch die veränderte Perspektive ist das als gut zu bewertende Erkennungergebnis zustande gekommen.

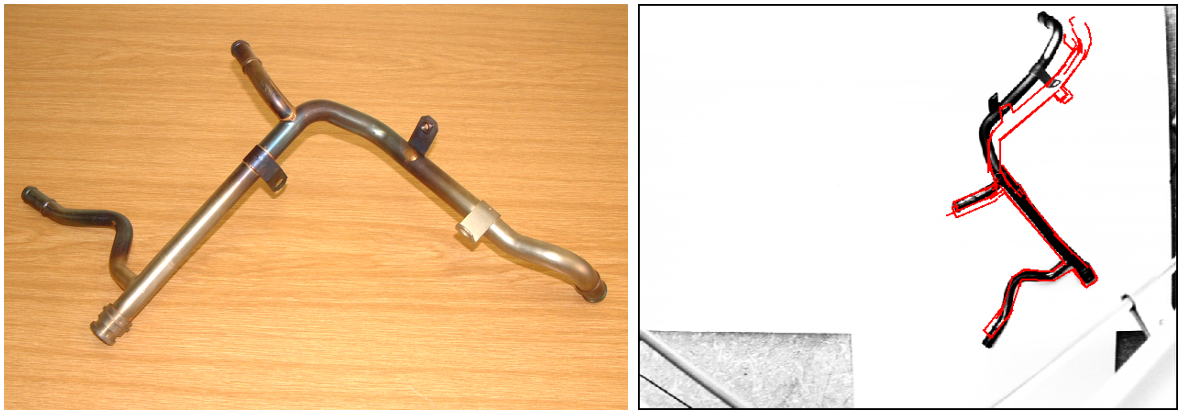

Abbildung 6.25: Links: Das Objekt *Rohrkombination*. Rechts: Erkennungsergebnis in einem lediglich kontrastverstärkten Testbild. Die Initialmerkmale der Modellansicht sind in rot überlagert. Das Objekt wurde auf einer homogenen, weißen Oberfläche positioniert. An den Bildrändern sind Teile des Versuchsaufbaus sichtbar.

Objekt *Kaffeekocher*

Das in Abbildung 6.26 links dargestellte Objekt *Kaffekocher* besitzt zwar gerade Kanten und eine große Anzahl von Ecken, deren Erkennung wird jedoch durch die stark reflektierende Oberfläche erschwert. Wie das rechte Bild in Abbildung 6.26 demonstriert, ist die Erkennung innerhalb einer mäßig komplexen Szene dennoch möglich. Bei der Analyse des Erkennungsergebnisses fällt auf, dass trotz der hohen Komplexität des Objektes nur 12 Initalpaare zur Erkennung beigetragen haben (diese sind dem Testbild in Abbildung 6.27 überlagert). Grund hierfür ist die Instabilität der Initalmerkmale aufgrund der reflektierenden Oberfläche unter Lage- und Beleuchtungsänderungen. Sowohl in Testbild als auch in der Modellansicht sind sehr viele Initalmerkmale vorhanden, deren Schnittmenge ist jedoch klein.

Objekt *Hammer*

Der Einfluss von Beleuchtungsverhältnissen wird an einem Experiment mit dem von der Form her einfachen Objekt *Hammer* deutlich. Verwendet man das in Abbildung 6.28 links dargestellte Bild des Hammers als Trainingsbild, so kann eine erfolgreiche Erkennung realisiert werden (Testbild mit Erkennungsergebnis siehe Abbildung 6.29 links). Wird dagegen das rechts in Abbildung 6.28 dargestellte Bild des Hammers als Trainingsbild verwendet, kann die daraus gelernte Modellansicht nicht erkannt werden. In diesem Trainingsbild behindern die leichten Schatten, welche die Objektkontur umgeben, die hinreichend genaue Erkennung der wichtigen Bildmerkmale.

Auf der rechten Seite von Abbildung 6.29 sind die Initialpaare dem Testbild überlagert,

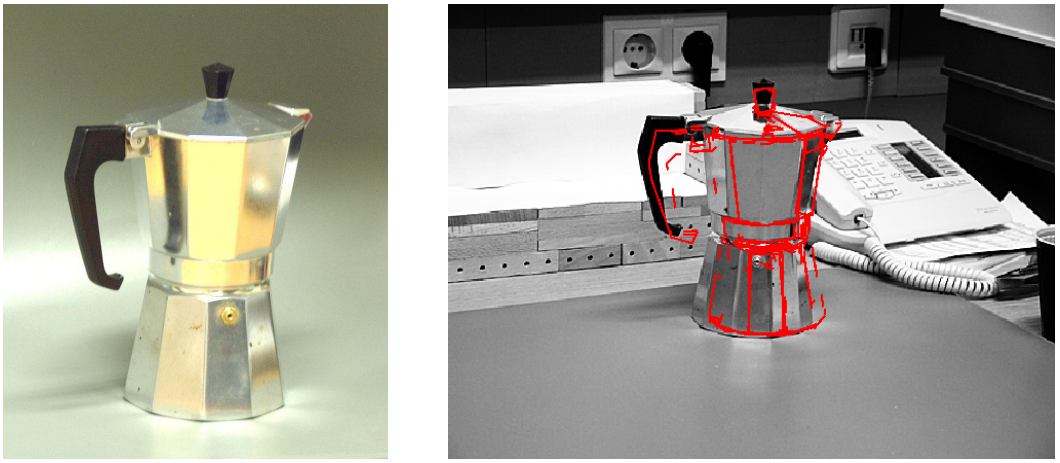

Abbildung 6.26: Links: Ein Trainingsbild des Objektes *Kaffekocher*. Rechts: Erkennungsergebnis (Die Merkmale der Modellansicht sind dem Testbild in rot überlagert).

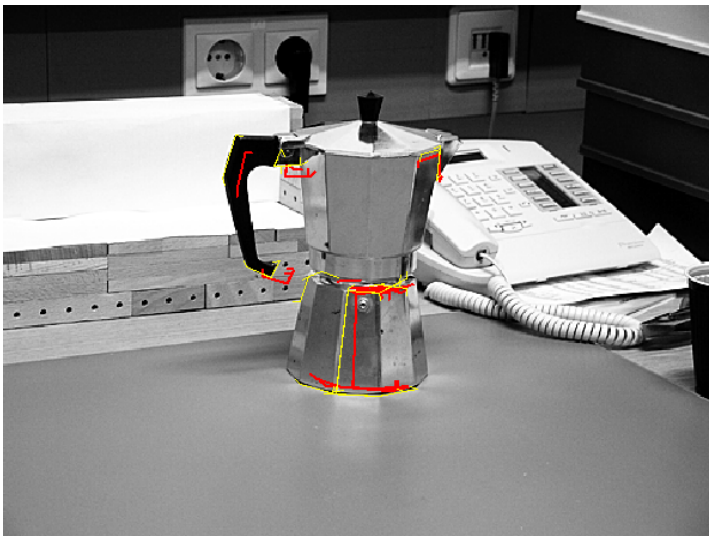

Abbildung 6.27: Das Testbild mit überlagerten Initialpaaren. Initialmerkmale des Testbildes sind gelb, jene der Modellansicht rot dargestellt.

welche zur Erkennung mittels des erstgenannten Trainingsbildes geführt haben. Die Erkennung konnte nur an zwei Initialmerkmalen (*Kappe* und *Kurbel-RL*) ansetzen. Wären diese nicht zuverlässig identifiziert worden, wäre die Erkennung gescheitert.

Objekt *Honigglas*

Mit dem in Abbildung 6.30 dargestellten Honigglas konnte aufgrund der karierten Textur keine Erkennung realisiert werden. Durch das sich häufig wiederholende Muster

Abbildung 6.28: Trainingsbilder des Objektes *Hammer*. Links: Aus diesem Trainingsbild kann eine Modellansicht gelernt werden, mit der eine Erkennung möglich ist. Rechts: Durch den Schattenwurf ist die robuste Extraktion der relevanten Bildmerkmale nicht möglich und eine Erkennung mit der damit erzeugten Modellansicht scheitert.

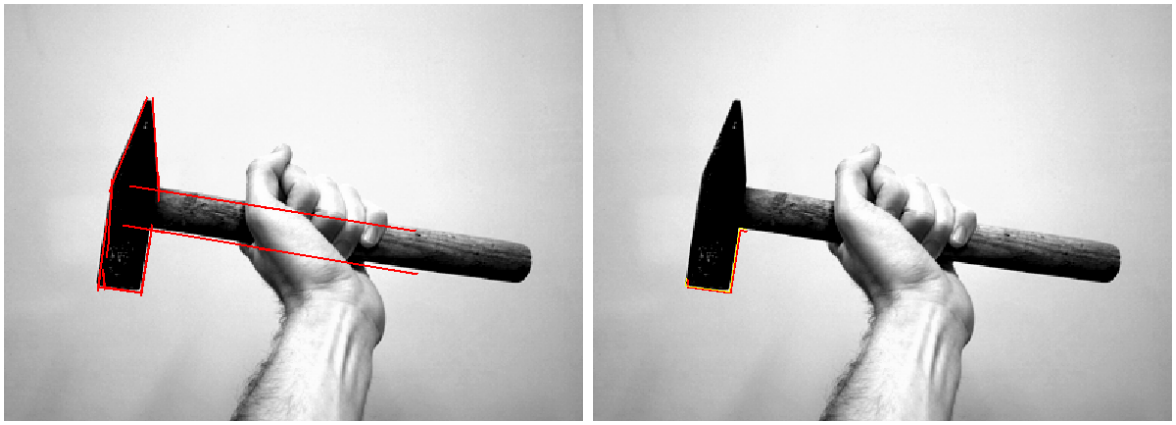

Abbildung 6.29: Erkennungsergebnisse mit dem Objekt *Hammer*. Links: Erfolgreiche Erkennung (die Initialmerkmale der Modellansicht sind dem Testbild in rot überlagert). Rechts: Überlagerung der Initialpaare, an denen die Erkennung angesetzt hat (Initialmerkmale des Testbildes sind gelb, jene der Modellansicht rot dargestellt).

werden mehr als 4000 Initialmerkmale mit nahezu denselben Attributwerten extrahiert. Die Korrespondenzsuche für ein Testbild, welches dasselbe karierte Muster wie das Trainingsbild enthält, müsste demnach im ungünstigsten Fall mehr als 4000^2 Initialpaare auswerten. Dies ist bei vertretbarem Zeitaufwand nicht möglich.

Objekt *Biene*

In Kapitel 6.1.3 wurde das Objekt *Ente* behandelt, bei dem zu wenige Ecken extrahiert werden und somit keine geeigneten Initialpaare gebildet werden können. Ähnlich

Abbildung 6.30: Objekt *Honigglas*.

verhält es sich bei einigen anderen Objekten aus vorherrschend gekrümmten Strukturen. Beispielsweise bei der in Abbildung 6.31 dargestellten Knetfigur *Biene* können zwar zahlreiche Initialmerkmale gefunden werden, diese variieren jedoch zu stark bei Veränderung der Objektlage oder der Beleuchtung. Dieses Problem wurde bei mehreren Objekten mit weicher oder diffuser Oberfläche beobachtet.

Abbildung 6.31: Objekt *Biene*.

6.4 Diskussion der Ergebnisse

Die Tabelle in Abbildung 6.32 gibt einen Überblick über die untersuchten Objekte, deren Eigenschaften und die erzielte Erkennungsleistung.

In Anbetracht der Erkennungsergebnisse unter schweren Szenenbedingungen und der

			Form			Erkennung / Lokalsierung		
Objekt	Siehe Abb.	Textur und Oberfläche	Kanten	Ecken	Initialprimitive	Einfache Szene	Komplexe Szene	Anhand der Silhouette
Hantel	6.17	einfarbig, matt	gerade	viele	viele	gut	gut	gut
Hammer	6.28	rau, komplex	gerade	wenige	wenige	möglich	möglich	gut
Kaffeekocher	6.26	glänzend	gerade	viele	sehr viele	möglich	möglich	gut
Rohrkombination	6.25	glänzend	gekrümmt	wenige	wenige	gut	möglich	gut
Dinosaurier	6.20	einfarbig, matt	gekrümmt	viele	viele	möglich	möglich	gut
Ente	6.21	einfarbig, matt	gekrümmt	wenige	wenige	möglich	nicht möglich	möglich
Knetfigur (Biene)	6.31	matt	gekrümmt	viele	sehr viele	möglich	nicht möglich	möglich
Honigglas	6.30	kariert	gerade / gekr.	wenige	sehr viele	nicht möglich	nicht möglich	nicht möglich

Abbildung 6.32: Die in den Untersuchungen verwandten Objekte, ihre Eigenschaften und erzielten Erkennungsergebnisse.

Vielfalt an betrachteten Objekten ist insgesamt die Erkennungs- und Lokalisierungsleistung als gut zu beurteilen.

Trotz der Robustheit des Erkennungsverfahrens gegenüber Störungen wurde eine deutliche Abhängigkeit der Erkennungsergebnisse von der Qualität der extrahierten Bildmerkmale beobachtet. Insbesondere die Anzahl und Stabilität der extrahierten Initialmerkmale ist entscheidend für eine erfolgreiche Erkennung. Es besteht die Forderung, dass mindestens ein Initialpaar robust aus Objektansicht und Testbild extrahiert werden kann. Problematisch sind somit einerseits Objekte, deren Gesamtzahl an höheren Merkmalen aufgrund ihrer einfachen Form und Textur zu gering ist. Andererseits stellen Objekte ein Problem dar, deren Merkmale nicht zuverlässig exrahiert werden können. Solche Objekte zeichnen sich meist durch eine besonders zerklüftete oder glänzende Oberfläche aus. Die Untersuchungen mit dem Objekt *Dinosaurier* (siehe Kapitel 6.1.3) haben gezeigt, dass die Erkennung trotz komplexer Form möglich ist, wenn zumindest an einigen Stellen robuste Initialmerkmale vorhanden sind.

Es können keine Bilder erfolgreich verarbeitet werden, auf denen ungünstige regelmäßige Muster, wie z.B. Karomuster, abgebildet sind.

Ist die Auflösung des Objektabbildes im Testbild zu gering oder weicht diese zu stark von der Auflösung der Modellansicht ab, so werden die für die Erkennung relevanten Bildstrukturen in Testbild und Modellansicht unter Umständen nicht durch zueinander kompatible Merkmale beschrieben, so dass die Korrespondenzsuche scheitert. Einem solchen Fall kann vorgebeugt werden, indem die Modellansichten in der zu erwartenden Auflösung des Objektabbildes erzeugt werden oder das Testbild selbst skaliert wird. In zahlreichen Testbildern der Untersuchungen waren die Objektabbilder bis zu viermal kleiner als die entsprechenden Modellansichten, ohne dass ein Absinken der Erkennungsleistung beobachtet werden konnte. Es kann jedoch davon ausgegangen werden, dass ein noch größerer Unterschied, besonders bei niedrigen Auflösungen, zu inkompatiblen Merkmalen führt.

In den Experimenten konnten zwar meistens gute Lokalisierungsergebnisse durch dieje-

nigen Modellansichten erreicht werden, die mit dem höchsten Verifikationsmaß bewertet wurden, einer algorithmischen Entscheidung mittels Schwellwertvergleich, ob die Erkennung gelungen ist, sind jedoch Grenzen gesetzt.

Kapitel 7

Abschlussbetrachtungen

In dieser Arbeit wurde ein System realisiert und getestet, das die räumliche Lokalisierung von Objekten unter praxisrelevanten Bedingungen leistet.

Arbeitsergebnisse

Zusammmenfassend sind im Rahmen der Arbeit folgende Ergebnisse erzielt worden:

- Auswahl, Beschaffung, Portierung und Ausbau eines im Hinblick auf die gestellte Lokalisierungsaufgabe geeigneten ansichtsbasierten Erkennungssystems.
- Erweiterung des Erkennungssystems auf die Berechnung des sechsdimensionalen Lagevektors eines Objektes aus dem Ergebnis der ansichtsbasierten Erkennung.
- Nutzung von Formmodellen zur Erzeugung künstlicher Objektansichten als Erweiterung und Vereinfachung des Lernprozesses.
- Das anfangs gesteckte Ziel einer hinreichenden Erkennungsleistung und Lokalisierungsgenauigkeit zur Initialisierung des Objektverfolgungsprozesses im komplexen Szenario des Projektes MQube wurde erreicht. Darüber hinaus konnte die angestrebte Flexibilität des Verfahrens für den Einsatz in verschiedenen anderen Anwendungen erreicht werden.
- Die detailierte Evaluierung des Lokalisierungssystems sowie des zugrundeliegenden ansichtsbasierten Erkennungsverfahrens. Hierbei konnten wichtige Erkenntnisse durch die Verwendung von sowohl künstlichen als auch realen Daten gewonnen werden. Durch Tests mit einer Reihe von Objekten z.T. sehr unterschiedlicher Objektklassen konnten die Möglichkeiten und Grenzen des Verfahrens noch weiter präzisiert werden.

Anwendungsgebiete

Es wurde gezeigt, dass das Lokalisierungssystem in Realweltszenen einsetzbar ist, wenn bestimmte Mindestanforderungen an Objektform und -oberfläche erfüllt sind: Können aus der Objektabbildung in Trainingsmenge und Testbild robuste Initialpaare (d.h. Paare aus Modell- und Bildmerkmalen, die stabil bezüglich ihrer Lage im Bild sind) extrahiert werden, liefert das System auch unter schweren Bedingungen wie partiellen Verdeckungen, komplexem Hintergrund und ungünstiger Beleuchtung gute Ergebnisse. Im Folgenden werden einige konkrete Anwendungsmöglichkeiten genannt.

Initialisierung eines Verfolgungsprozesses

Das Lokalisierungssystem kann im Rahmen des in der Einleitung beschriebenen Projekts MQube am Fraunhofer Institut IITB in Karlsruhe eingesetzt werden. Es hat sich gezeigt, dass in diesem Umfeld trotz starker Störeinflüsse wie Nebel und Verdeckungen gute Lokalisierungsergebnisse erzielt werden können. Versuche haben ergeben, dass die erreichten Lokalisierungsergebnisse zur Aufnahme des Verfolgungsprozesses ausreichen. Innerhalb weniger Sekunden kann das von Dr. Thomas Müller entwickelte Verfolgungsverfahren die Lageschätzung präzisieren.

Industrielle Anwendung

Durchgeführte Versuche haben die Anwendbarkeit des Lokalisierungssystems in einer industriellen Anwendung demonstriert. Es wurden verschweißte Rohrstücken trotz erheblicher Störeinflüsse durch ihre glänzende Oberfläche erkannt und deren räumliche Lage geschätzt. Solche Lageschätzungen sind im industriellen Umfeld von großer Bedeutung, da durch sie oft eine automatische Weiterverarbeitung ermöglicht wird. Im Gegensatz zu anderen Anwendungsgebieten ergibt sich hier die Möglichkeit, die Szenenverhältnisse bei Bedarf zu beeinflussen. So kann gegebenenfalls die Beleuchtung auf die Bedürfnisse der Merkmalsextraktion hin eingerichtet werden. Die Ergebnisse stimmen optimistisch im Hinblick auf die mögliche Einsetzbarkeit des Verfahrens in anderen industriellen Anwendungen.

Objektidentifikation

Im Rahmen ihrer Dissertation entwickelte Astrid Laubenheimer am Fraunhofer Institut IITB in Karlsruhe ein System zur Identifikation von Flugzeugen im Bereich der Luftbildauswertung. Deformierbare Formmodelle werden durch ein iteratives Verfahren an ein Grauwertbild angepasst und so die genaue Objektausprägung und die räumliche

Lage ermittelt (siehe [Laubenheimer & Link 2003]). Zur Initialisierung der iterativen Modellausprägung werden zwei Dinge benötigt:

- Die Angabe der Objektklasse des im Bild sichtbaren Objektes.
- Eine Schätzung der räumlichen Objektlage unter Berücksichtigung bestimmter Genauigkeitsschranken.

Das in dieser Arbeit vorgestellte Lokalisierungssystem wäre für beide Aufgaben erwartungsgemäß gut geeignet, da es tolerant gegenüber den dort auftretenden Modellabweichungen ist und trotz der Abweichungen eine robuste Lageschätzung liefern dürfte. Aufgrund der Art der verwendeten Objekte und der Szenenverhältnisse (Luftbildaufnahmen von Flugzeugen auf Flugplätzen) wird erwartet, dass eine hinreichend genaue Beschreibung durch lokale Bildmerkmale möglich ist und gute Erkennungsergebnisse erzielt werden können.

Grenzen

Die Erkennungsleistung ist in natürlicher Weise durch die Qualität der zugrundeliegenden Merkmalsextraktion beschränkt. Können aufgrund ungünstiger Bildmerkmale die relevanten visuellen Strukturen nur unzureichend im Objektmodell oder in der Darstellung des Testbildes erfasst werden, kann das Erkennungsverfahren kein korrektes Ergebnis liefern. Ungünstige Darstellungen durch Bildmerkmale treten konkret in den folgenden Fällen auf:

- Aufgrund einer zu einfachen Objektform und -oberfläche können keine höheren Bildmerkmale extrahiert werden (z.b. bei einem einfarbigen Ball).
- Bestimmte sich wiederholende Muster (z.B. Karomuster) führen zu einer übermäßig großen Menge von höheren Bildmerkmalen.
- Die extrahierten Merkmale variieren zu stark bei Lage- oder Beleuchtungsveränderungen (z.B. bei stark zerklüfteter oder glänzender Objektoberfläche).

Ausblick

Die Integration von Vorwissen über die Objektlage würde sowohl die Laufzeit drastisch reduzieren als auch die Fehlerrate verringern. Ebenso könnten die Ergebnisse vergangener Erkennungsvorgänge als Vorwissen eingebracht werden. Beispielsweise können

vorrangig die Modellansichten mit einem Testbild verglichen werden, die in der Vergangenheit häufig erkannt wurden. Unwahrscheinliche oder entartete Modellansichten müssten dann erst spät im Erkennungsprozess bzw. meist gar nicht betrachtet werden.

Eine Erweiterung des Merkmalsrepertoirs sowie eine Verbesserung des Extraktionsverfahrens hätte großes Potential zur Leistungssteigerung des Systems. Hier bieten sich beispielsweise texturbeschreibende Merkmale an, da im aktuellen System nur kantenbasierte Merkmale verwendet werden. Es sollten besonders Merkmale integriert werden, die bezüglich ihrer vier Lagedimensionen stabil sind, also zur Bildung von Initialpaaren geeignet sind und damit besonders zur Erkennungsleistung beitragen[1] Ein vorgeschalteter Verarbeitungsschritt zur Identifikation von Texturen würde den Umgang mit Objekten erlauben, die sich häufig wiederholende ungünstige Muster, z.B. karierte Texturen, enthalten. Die relativ leicht zu realisierende Verbesserung der Eckenidentifikation im Falle der Verwendung von Kreissegmenten dürfte die Erkennungsleistung bei Objekten mit gekrümmten Strukturen erhöhen.

Durch den statistischen Ansatz wird die Relevanz von Merkmalen automatisch gewichtet. Aus diesem Grund ist zu erwarten, dass zusätzliche Objektmerkmale positiv auf das Erkennungsergebnis wirken und nur einen geringen Einfluss auf die Laufzeit haben. Beispielsweise könnte eine Modellansicht weitere Modellmerkmale erhalten, indem ihre Objektansichten jeweils in mehreren Auflösungen der Trainingsmenge hinzugefügt werden. So könnte für jeden Merkmalstyp die optimale Skala verwendet werden, da er in der entsprechenden Auflösung die höchste Stabilität besitzt.

Die Lokalisierungsgenauigkeit ließe sich, alternativ zur Nachschaltung eines Verfolgungsschritts, auch durch ein iteriertes Vorgehen erhöhen. Für jede Modellansicht wäre eine weitere Menge von Modellansichten zu generieren, welche den entsprechenden Bereich der Trainingsansichten detailierter abbildet.

Die Laufzeit nimmt linear in der Anzahl der gesuchten Objekte und der Anzahl der jeweiligen Modellansichten zu. Eine Indizierungskomponente, die anhand der Ausprägung einiger höherer Merkmale den selektiven Zugriff auf vielversprechende Modellansichten erlaubt, würde die Laufzeit, insbesondere bei eine großen Objektdatenbank, vorraussichtlich deutlich reduzieren.

Durch Ausbau der in den Experimenten benutzten automatischen Tests ließe sich durch relativ geringen Aufwand ein Werkzeug schaffen, das automatisch die optimale Anzahl von Trainingsansichten für ein spezielles Objekt bestimmt.

[1] In Anhang A ist die Skizze eines denkbaren neuen Initialmerkmals *T-Ecke* zu finden, das in vielen Fällen zu einer Steigerung der Erkennungsleistung führen könnte.

Anhang A

Merkmalsrepertoire

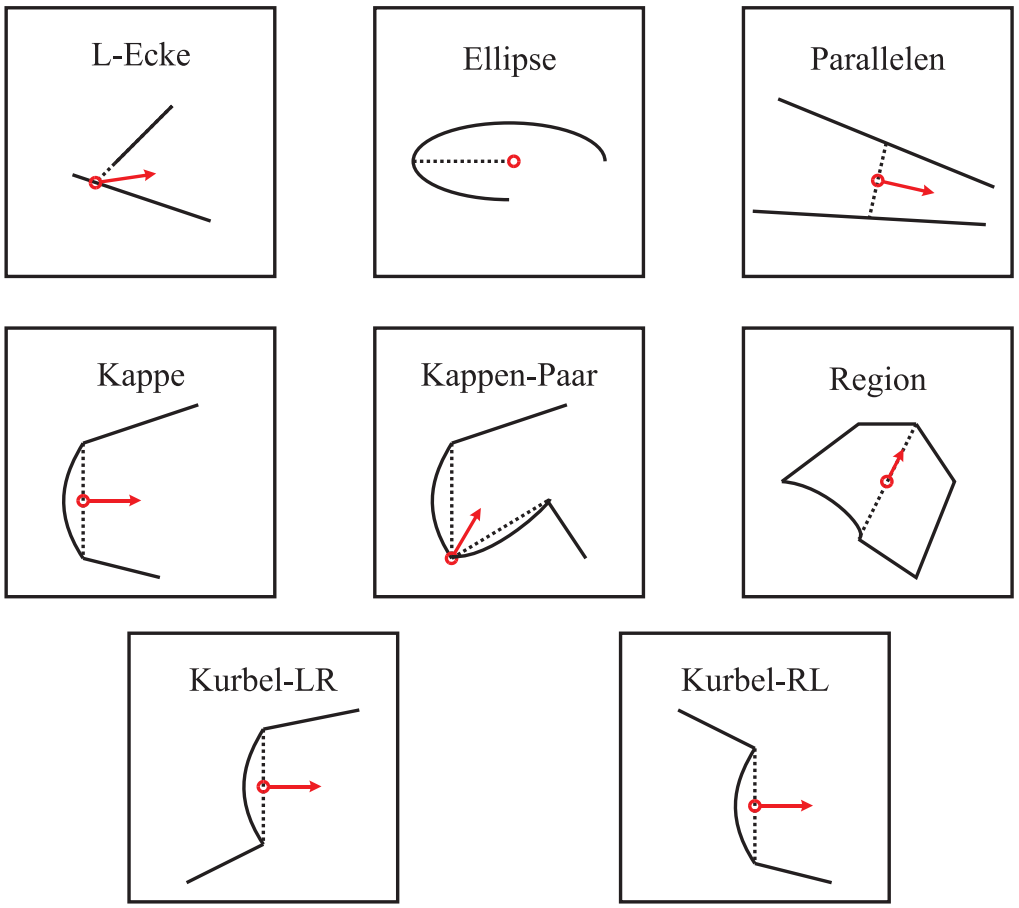

Abbildung A.1: Repertoire der im Lokalisierungsystem verwendeten höheren Merkmale. Initialmerkmale: *Kappe*, *Kappenpaar*, *Region*, *Kurbel-LR*, *Kurbel-RL*, keine Initialmerkmale: *L-Ecke*, *Ellipse*, *Parallelen*.

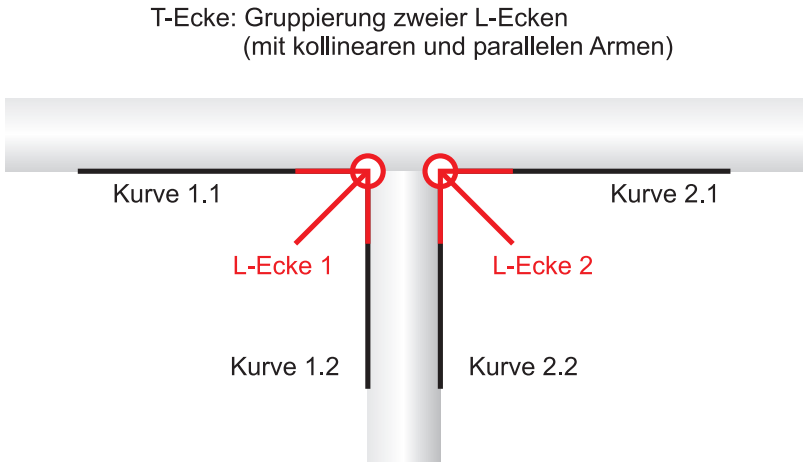

Abbildung A.2: Als Erweiterung des Repertoirs vorgeschlagenes Initialmerkmal *T-Ecke*. Es ist stabil bezüglich der vier Lagedimensionen Position, Orientierung und Größe.

Anhang B

Ergänzende Berechnungen

B.1 Kalibrierung einer virtuellen Kamera

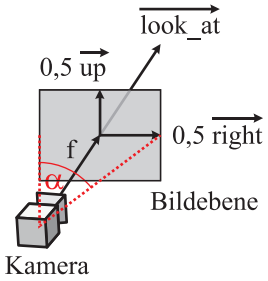

Abbildung B.1: Die Notation der virtuellen Kamera des Softwarepakets *Povray*.

Die Kamera der Software zur Erzeugung photorealistischer Bilder *Povray* wird durch folgende Parameter beschrieben (vgl. Abbildung B.1):

- Position $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$,
- Blickrichtungsvektor **look_at**,
- Öffnungswinkel α,
- Ausrichtung der Bildebene **up** und **right**,
- Höhe h und Breite b des Bildes in Pixeln.

Daraus kann die Kalibriermatrix der Kamera in der Notation von Abschnitt 5.1 bestimmt werden:

$$\begin{aligned} f &= \frac{\frac{1}{2} \mid \mathbf{right} \mid}{\tan \frac{\alpha}{2}} \ , \\ s_x &= \frac{h}{\mid \mathbf{up} \mid} = \frac{b}{\mid \mathbf{right} \mid} \ . \end{aligned}$$

Die Experimente wurden mit folgenden Werten durchgeführt:

$$\begin{aligned} \mathbf{up} &= \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \ , \\ \mathbf{right} &= \begin{pmatrix} \frac{4}{3} \\ 0 \\ 0 \end{pmatrix} \ , \\ h &= 288 \ , \\ b &= 384 \ , \\ \alpha &= 11^\circ \ , \\ f &\approx 6{,}92 \ , \\ s_x &\approx 288 \ , \\ s_x f &\approx 1994 \ . \end{aligned}$$

Zum Lernen der Ansichten wurden die virtuellen Objekte $l = 2000$ mm vor der Kamera positioniert.

Durch die oben genannten Gleichungen lässt sich auch umgekehrt aus einer gegebenen Kalibriermatrix der Öffnungswinkel der Kamera bestimmen:

$$\alpha = 2 \cdot \tan^{-1} \left(\frac{\frac{1}{2} b}{s_x f} \right) \ .$$

Für die im Projekt MQube eingesetzten Kamera ergibt sich der Winkel $\alpha = 41^\circ$.

B.2 Kombination von Lagetransformationen

Seien $\mathbf{L_0}$ und $\mathbf{L_1}$ die Lagevektoren

$$\begin{aligned} \mathbf{L_0} &= (s, \alpha, x_b, y_b) \ , \\ \mathbf{L_1} &= (\tilde{s}, \tilde{\alpha}, \tilde{x_b}, \tilde{y_b}) \end{aligned}$$

mit den entsprechenden Transformationen

$$L_0 = \begin{pmatrix} u & -v & x_b \\ v & u & y_b \\ 0 & 0 & 1 \end{pmatrix} , \qquad \begin{aligned} u &= s \ \cos(\alpha) \ , \\ v &= s \ \sin(\alpha) \ , \end{aligned}$$

$$L_1 = \begin{pmatrix} \tilde{u} & -\tilde{v} & \tilde{x_b} \\ \tilde{v} & \tilde{u} & \tilde{y_b} \\ 0 & 0 & 1 \end{pmatrix} , \qquad \begin{aligned} \tilde{u} &= \tilde{s} \ \cos(\tilde{\alpha}) \ , \\ \tilde{v} &= \tilde{s} \ \sin(\tilde{\alpha}) \ . \end{aligned}$$

Dann lässt sich die Verkettung $\left(L_0^{-1}\right) \circ L_1$ der beiden Transformationen wie folgt berechnen:

$$L = \left(L_0^{-1}\right) \circ L_1 = \frac{1}{u^2 + v^2} \begin{pmatrix} u\tilde{u} + v\tilde{v} & -u\tilde{v} + v\tilde{u} & u\tilde{x_b} + v\tilde{y_b} - vy_b - x_b u \\ -v\tilde{u} + u\tilde{v} & u\tilde{u} + v\tilde{v} & -v\tilde{x_b} + u\tilde{y_b} - u\tilde{y_b} - x_b v \\ 0 & 0 & 1 \end{pmatrix} .$$

Berücksichtigung unterschiedlicher Koordinatensysteme

Das Erkennungsverfahren von Kapitel 2 verwendet intern die Notation des Frameworks *Vista*. Für Objekt- und Modellansichten wird dagegen eine zentrierte, rechtshändige Notation verwendet und für Testbilder ein rechtshändiges Koordinatensystem mit Ursprung in der linken, oberen Ecke. Siehe Abbildung B.2.

Um die Erkennungsverfahren des Kapitel 2 verwenden zu können, müssen die Bilder also transformiert werden (siehe Abbildung B.3):

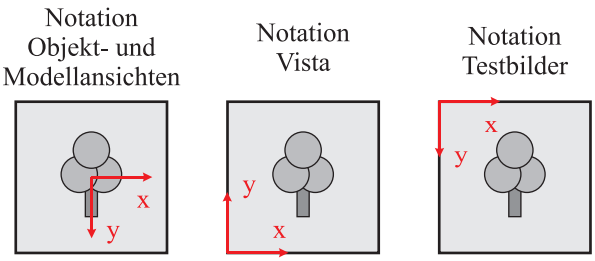

Abbildung B.2: Die verschiedenen Koordinatenssysteme im Lokalisierungssystem.

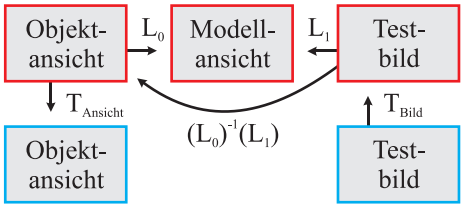

Abbildung B.3: Transformation eines Lagevektors über verschiedene Bildformate hinweg. Rote Rahmen stehen für Bilder in *Vista*-Notation, blaue für die anderen beiden Notationen aus Abbildung B.2.

$$L = T_{Ansicht} \circ \left(L_0^{-1}\right) \circ L_1 \circ T_{Bild} \ .$$

T_{Bild} transformiert das Testbild in die Vista-Notation und $T_{Ansicht}$ bringt das Ergebnis in die Notation für Objektansichten.

$$T_{Ansicht} = \begin{pmatrix} 1 & 0 & -c_x \\ 0 & -1 & c_y \\ 0 & 0 & 1 \end{pmatrix}, \qquad (c_x, c_y) : \text{Mittelpunkt der Ansicht} \ ,$$

$$T_{Bild} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & h \\ 0 & 0 & 1 \end{pmatrix}, \qquad h : \text{Höhe des Testbildes} \ .$$

B.3 Transformationen

Die Drehung um eine beliebige Achse $\mathbf{a} = (x, y, z)$ (mit $\|\mathbf{a}\| = 1$) mit dem Winkel α lässt sich wie folgt darstellen:

$$T_{\mathbf{a},\alpha} = \begin{pmatrix} x^2 + \cos\alpha\,(1-x^2) & xy\,(1-\cos\alpha) + z\sin\alpha & xz\,(1-\cos\alpha) - y\sin\alpha \\ xy\,(1-\cos\alpha) - z\sin\alpha & y^2 + \cos\alpha\,(1-y^2) & yz\,(1-\cos\alpha) + x\sin\alpha \\ xz\,(1-\cos\alpha) + y\sin\alpha & yz\,(1-\cos\alpha) - x\sin\alpha & z^2 + \cos\alpha\,(1-z^2) \end{pmatrix} .$$

Eine Drehung um die x-Achse ergibt sich für $\mathbf{a} = (1, 0, 0)$

$$T_{\hat{\mathbf{a}},\alpha} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{pmatrix} .$$

Eine Drehung um die y-Achse ergibt sich für $\mathbf{a} = (0, 1, 0)$

$$T_{\hat{\mathbf{a}},\alpha} = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix} .$$

Eine Drehung um die z-Achse ergibt sich für $\mathbf{a} = (0, 0, 1)$

$$T_{\hat{\mathbf{a}},\alpha} = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} .$$

Die Händigkeit eines Koordinatensystems kann mit folgender Transformation geändert werden:

$$T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} .$$

Die Drehung $T_{\mathbf{a},\mathbf{b}}$ eines Vektors $\mathbf{a}$ auf einen Vektor $\mathbf{b}$ berechnet sich wie folgt:

$$
\begin{aligned}
\mathbf{s} &:= \mathbf{a} \times \mathbf{b} \ , \\
\mathbf{s_0} &:= \frac{\mathbf{s}}{\|\mathbf{s}\|} \ , \\
\omega &:= \measuredangle(\mathbf{a}, \mathbf{b}) = \cos^{-1}\left(\frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\|\,\|\mathbf{b}\|}\right) \ , \\
T_{\mathbf{a},\mathbf{b}} &:= T_{\mathbf{s_0},\omega} \ .
\end{aligned}
$$

B.4 Sphärische Winkel

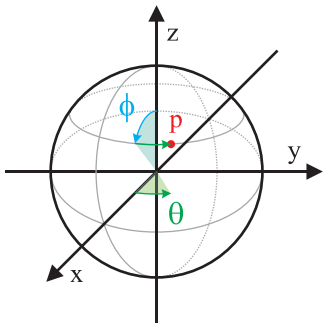

Abbildung B.4: Die sphärischen Winkel ϕ und θ definieren den Punkt $p = p_{\phi,\theta}$ eindeutig.

Die sphärischen Winkel ϕ (Polarwinkel) und θ (Azimuthwinkel) definieren einen Punkt $p = p_{\phi,\theta}$ auf der Oberfläche der Einheitskugel um den Ursprung eindeutig (vgl. Abbildung B.4). In der Literatur ist folgende Definition gebräuchlich:

- $p_{0,0}$ liegt bei $(0, 0, 1)$
- $p_{\phi,0}$ geht aus $p_{0,0}$ durch Drehung um die y-Achse mit dem Winkel $-\phi$ hervor.
- $p_{\phi,\theta}$ geht aus $p_{\phi,0}$ durch Drehung um die z-Achse mit dem Winkel $-\theta$ hervor.

B.5 Projektive Verzerrung von Objektansichten

Unter projektiven Abbildungen bleiben Winkel nicht erhalten. Im Folgenden wird hergeleitet, wie sich diese Aussage für den praxisrelevanten Fall der Verzerrung von Objektansichten quantifizieren lässt. Sei γ ein Winkel auf der Objektoberfläche und $\overline{\gamma}$ das

Bild dieses Winkels nach Rotation des Objektes um die Kamera mit Winkel α (vgl. Abbildung B.5). Man betrachte einen Bildvektor (b_x, b_y), der durch den Winkel γ definiert wird und den entsprechenden verzerrten Bildvektor $(\bar{b}_x, \bar{b}_y)$ (aus den Abbildungen B.5 und Abbildungen B.6 lässt sich die Notation leicht erschließen).

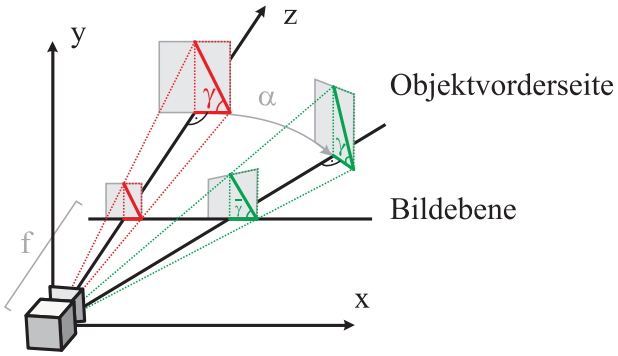

Abbildung B.5: Veränderung des Bildes eines Winkels γ auf der Objektoberfläche bei Rotation des Objektes um die Kamera.

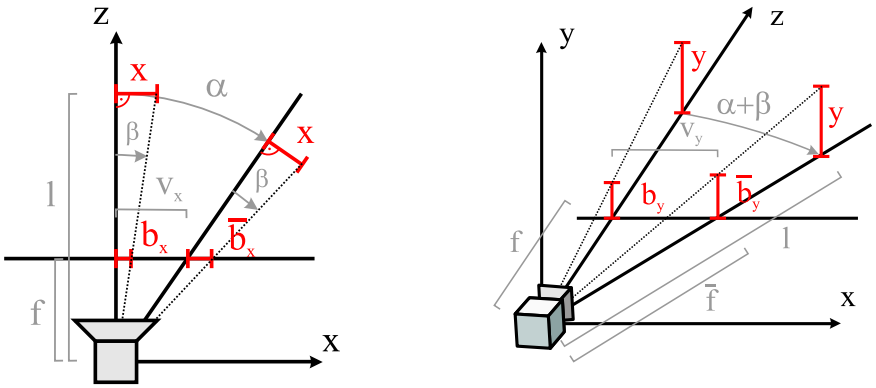

Abbildung B.6: Notation zur Herleitung der Winkeländerung. Links: Notation zur die Herleitung der x-Komponente. Rechts: Notation zur Herleitung der y-Komponente.

Die x-Komponente $\bar{b}_x$ des verzerrten Bildvektors lässt sich wie folgt berechnen (vgl. Abbildung B.6 links):

$$
\begin{aligned}
b_x &= f\,\frac{x}{l} \\
\tan\beta &= \frac{x}{l}, \qquad \beta = \tan^{-1}\frac{x}{l} \\
\tan\alpha &= \frac{v_x}{f}, \qquad v_x = f\,\tan\alpha \\
\tan(\alpha+\beta) &= \frac{v_x+\overline{b}_x}{f}, \qquad \overline{b}_x = f\,\tan(\alpha+\beta) - v_x \\
\overline{b}_x &= f\,\tan\left(\alpha+\tan^{-1}\frac{x}{l}\right) - f\,\tan\alpha
\end{aligned}
$$

Die y-Komponente $\overline{b}_y$ des verzerrten Bildvektors lässt sich wie folgt berechnen (vgl. Abbildung B.6 rechts):

$$
\begin{aligned}
\tan\alpha &= \frac{v_y}{f}, \qquad v_y = f\,\tan\alpha \\
\sin\alpha &= \frac{v_y}{\overline{f}}, \qquad \overline{f} = \frac{v_y}{\sin\alpha} = f\,\frac{1}{\cos\alpha} \\
\overline{b}_y &= \overline{f}\,\frac{y}{l} = f\,\frac{1}{\cos\alpha}\,\frac{y}{l}
\end{aligned}
$$

Aus der x-Komponente $\overline{b}_x$ und der y-Komponente $\overline{b}_y$ des verzerrten Bildvektors lässt sich der verzerrte Bildvektor $\overline{\gamma}$ wie folgt bestimmen (vgl. Abbildung B.5):

$$
\begin{aligned}
\gamma &= \tan^{-1}\frac{y}{x} \\
y &= x\,\tan\gamma \\
\overline{\gamma} &= \tan^{-1}\frac{\overline{b}_y}{\overline{b}_x} \\
&= \tan^{-1}\frac{f\,\frac{1}{\cos\alpha}\,\frac{y}{l}}{f\,\tan\left(\alpha+\tan^{-1}\frac{x}{l}\right) - f\,\tan\alpha} \\
&= \tan^{-1}\frac{\frac{1}{\cos\alpha}\,\frac{x\,\tan\gamma}{l}}{\tan\left(\alpha+\tan^{-1}\frac{x}{l}\right) - \tan\alpha}
\end{aligned}
$$

Dieses Ergebnis lässt sich folgendermaßen auf das Problem der perspektivischen Verzerrung von Objektansichten übertragen: Ein Objekt habe den Abstand l von der Kamera und besitze auf der kamerazugewandten Seite eine Struktur, welche im Bild zu einem

Merkmal des Typs *L-Ecke* der Größe x und dem Winkel γ führt. Dann verändert eine Drehung des Objekts um die Kamera mit dem Winkel α den Winkel des Bildmerkmale auf $\overline{\gamma}$. Setzt man für α den halben Öffnungswinkel der Kamera ein, für x die maximale Größe einer *L-Ecke* (z.B. die maximale Objektgröße) und für l den minimalen Abstand eines Objektes zur Kamera, so erhält man für jeden Winkel γ die maximal mögliche Veränderung $\gamma - \overline{\gamma}$. Das Diagramm in Abbildung B.7 zeigt diesen Zusammenhang für $x = 50$ cm, $l = 150$ cm und $\alpha = \frac{41}{2}^\circ$.

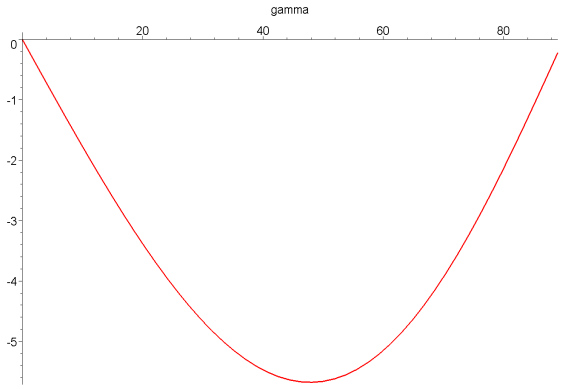

Abbildung B.7: Zusammenhang zwischen dem Winkel γ (waagrecht aufgetragen) auf der Objektoberfläche und der Winkeländerung $\gamma - \overline{\gamma}$ (senkrecht aufgetragen) bei ungünstigster räumlicher Lage des Objektes.

Anhang C

Modellansichten und Testbilder

C.1 Modellansichten

C.1.1 Modellansichten aus 130 Trainingsbildern

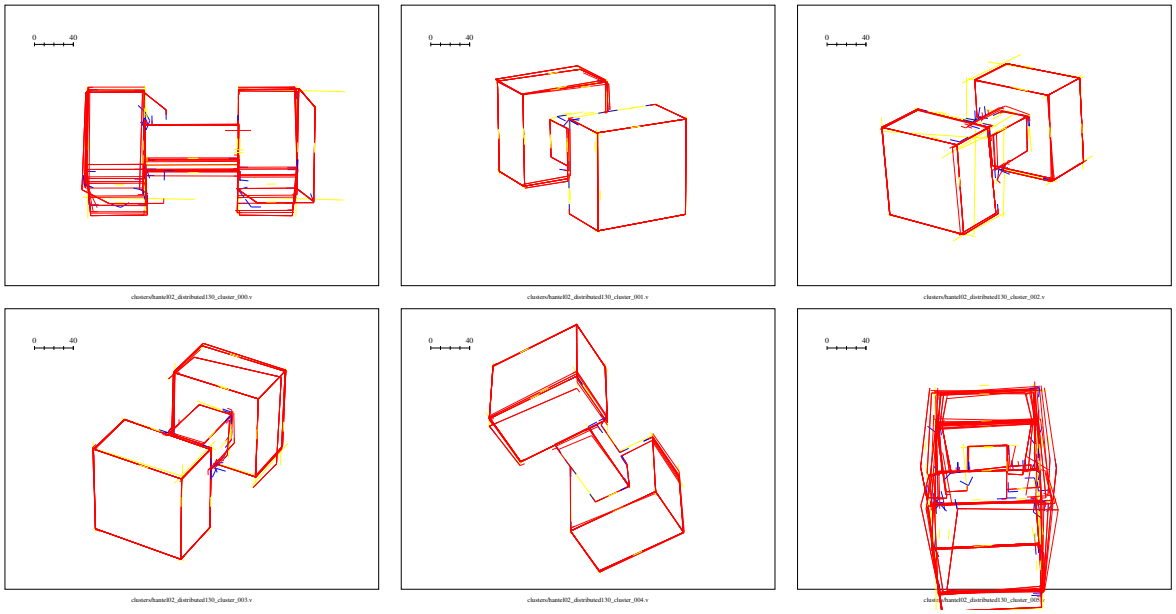

Abbildung C.1: Modellansichten 1 bis 6 (zeilenweise nummeriert) des Objektes *Hantel* gelernt aus den 130 Objektansichten der Menge *Hantel-Ansichten-130*. Die einzelnen Modellmerkmale sind farblich gekennzeichnet (blau: *L-Ecke*, gelb: *Geradensegment*, rot: *Kappe* und *Kappenpaar*).

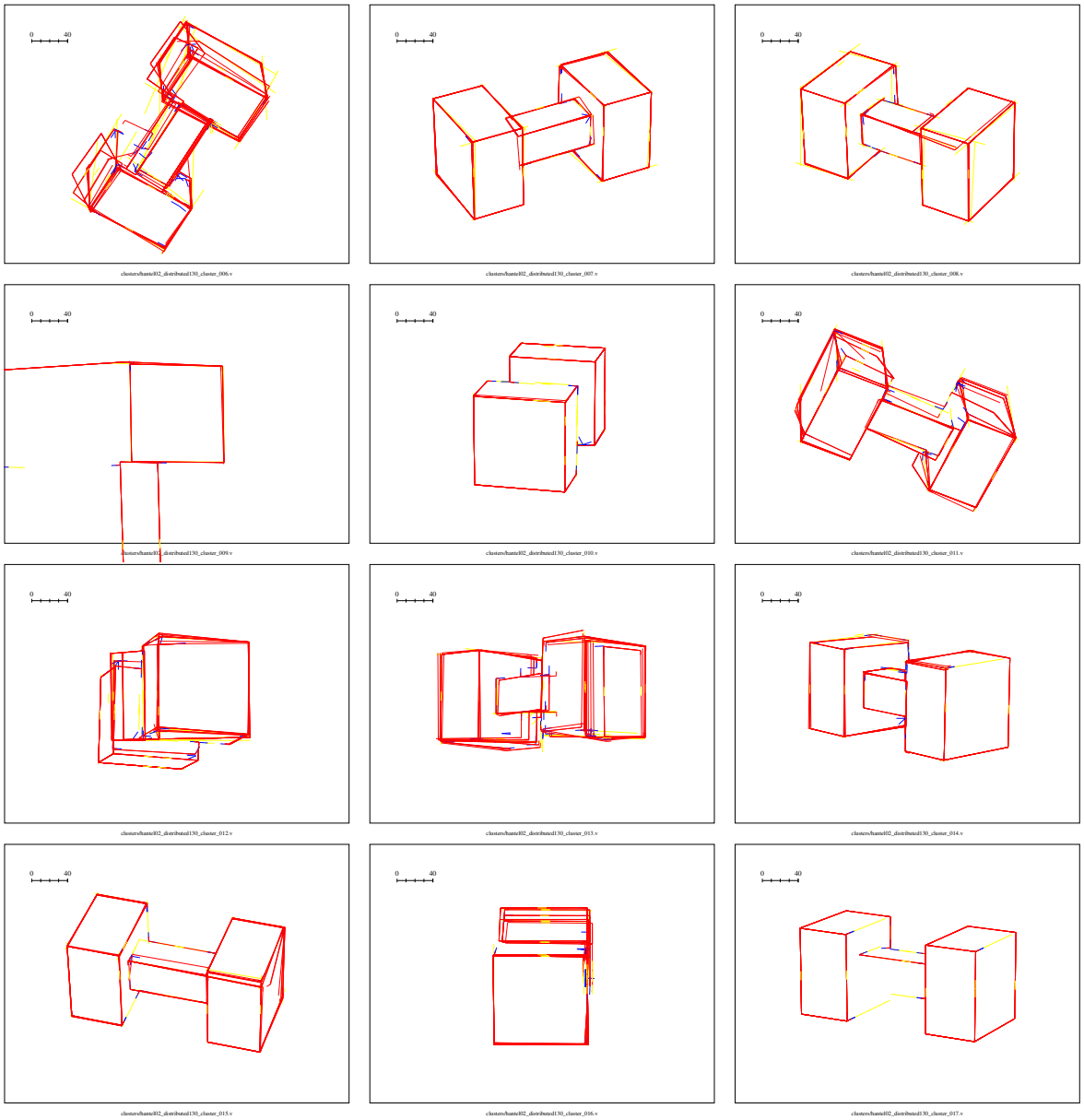

Abbildung C.2: Modellansichten 7 bis 18 (zeilenweise nummeriert) des Objektes *Hantel* gelernt aus den 130 Objektansichten der Menge *Hantel-Ansichten-130*. Die einzelnen Modellmerkmale sind farblich gekennzeichnet (blau: *L-Ecke*, gelb: *Geradensegment*, rot: *Kappe* und *Kappenpaar*)

C.1.2 Modellansichten aus 66 Trainingsbildern

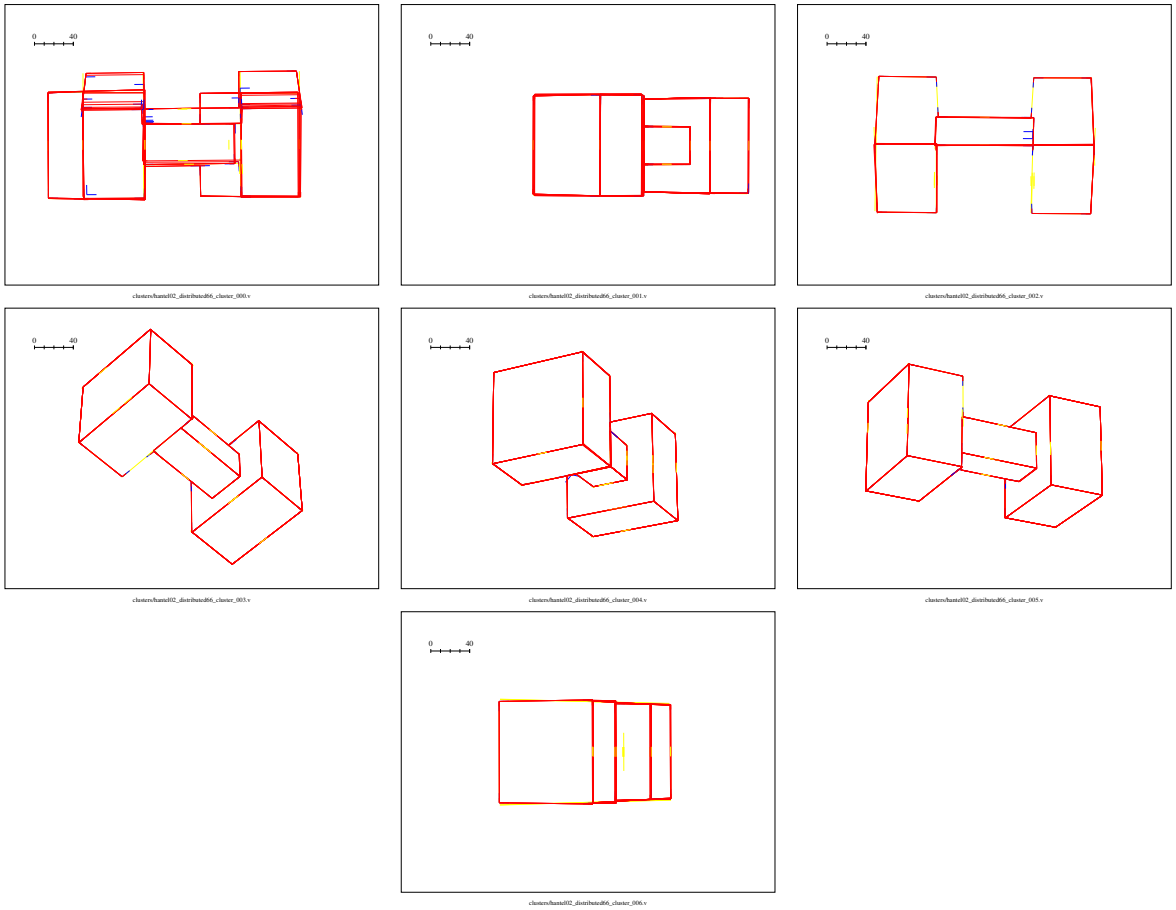

Abbildung C.3: Modellansichten 1 bis 7 (zeilenweise nummeriert) des Objektes *Hantel* gelernt aus den 66 Objektansichten der Menge *Hantel-Ansichten-66*. Die einzelnen Modellmerkmale sind farblich gekennzeichnet (blau: *L-Ecke*, gelb: *Geradensegment*, rot: *Kappe* und *Kappenpaar*)

C.1.3 Modellansichten aus 34 Trainingsbildern

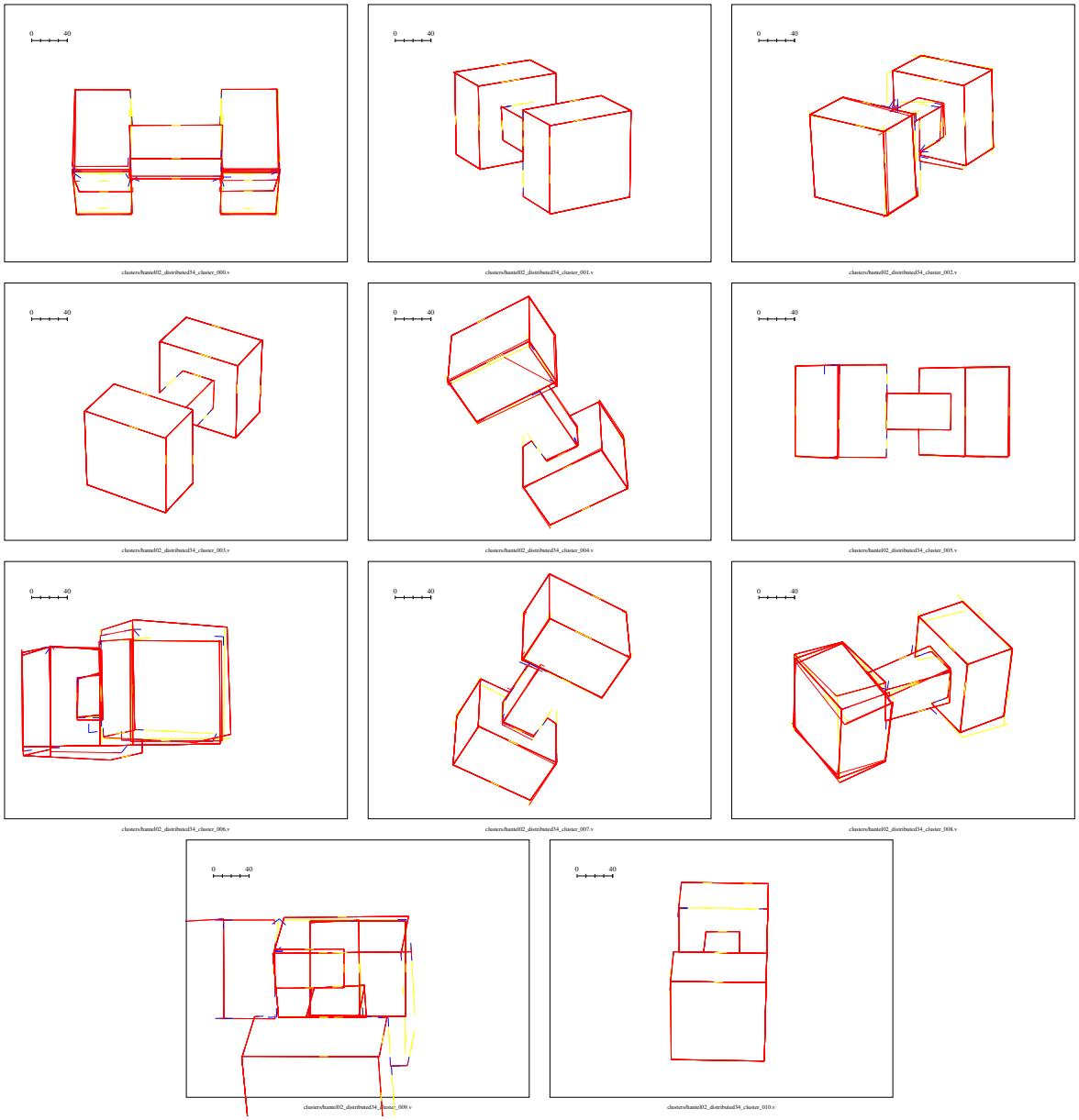

Abbildung C.4: Modellansichten 1 bis 11 (zeilenweise nummeriert) des Objektes *Hantel* gelernt aus den 34 Objektansichten der Menge *Hantel-Ansichten-34*. Die einzelnen Modellmerkmale sind farblich gekennzeichnet (blau: *L-Ecke*, gelb: *Geradensegment*, rot: *Kappe* und *Kappenpaar*)

C.2 Testbildmenge *M3-Hantel-Real*

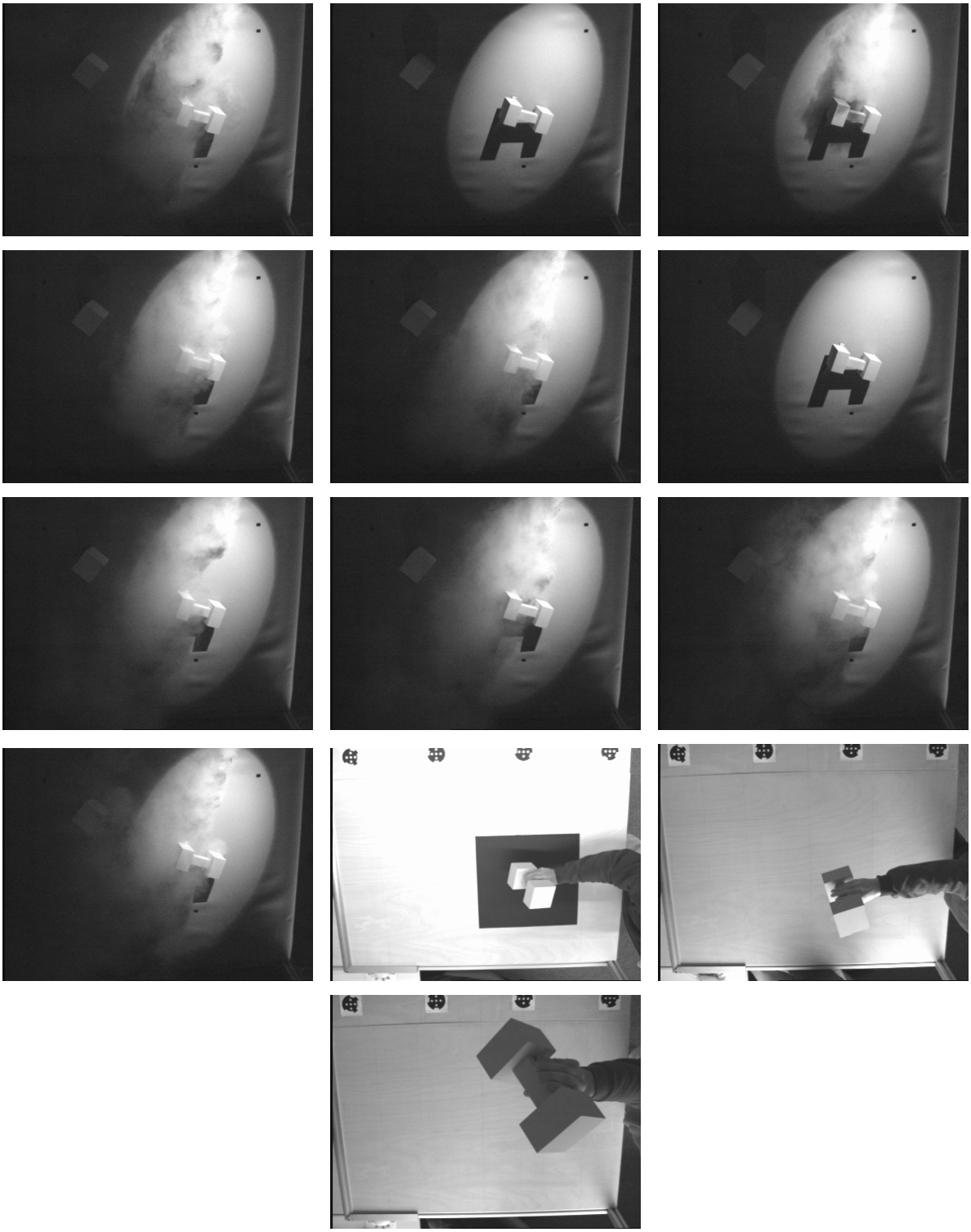

Abbildung C.5: Die 13 Testbilder der Testbildmenge *M3-Hantel-Real*. Die Nummerierung erfolgt zeilenweise, links oben beginnend mit Testbild 1.

Literaturverzeichnis

[Armangué, Salvi & Batlle 2000] X. Armangué, J. Salvi, J. Batlle: *A Comparative Review Of Camera Calibrating Methods with Accuracy Evaluation.* Institut de Informática i Aplications, Universitat de Girona, Spanien, 2000.

[Ballard 1981] D.H. Ballard: *Generalizing the Hough transform to detect arbitrary shapes.* Pattern Recognition, Vol. 13, 1981, pp. 111-122.

[Beis & Lowe 1999] J.S. Beis, D.G. Lowe: *Indexing without Invariants in 3D Object Recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21 (10), 1999, pp. 1000-1015.

[Breuel 1992] T.M. Breuel: *Fast recognition using adaptive subdivisions of transformation space.* Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1992, pages 445-451.

[Breuel 1993] T.M. Breuel: *The 3D Indexing Problem.* Memo 93-08, IDIAP, Martigny, Switzerland, 1993.

[Cass 1998] T.A. Cass: *Robust Affine Structure Matching for 3D Object Recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20 (11), 1998, pp. 1265-1274.

[Costa & Shapiro 2000] M.S. Costa, L.G. Shapiro: *3D Object Recognition and Pose with Relational Indexing.* Computer Vision and Image Understanding, Vol. 70, 2000, pp. 364-407.

[Eric & Grimson 1990] W. Eric, L. Grimson: *Object Recognition by Computer: The Role of Geometric Constraints.* The MIT Press, 1990, Cambridge, Massachusetts, USA.

[Häusler & Ritter 1999] G. Häusler, D. Ritter: *Feature-Based Object Recognition and Localization in 3D-Space, Using a Single Video Image.* Computer Vision and Image Understanding CVIU, Vol. 73 (1), 1999, pp. 64-81.

[Huttenlocher & Ullman 1990] D.P. Huttenlocher, S. Ullman: *Recognizing solid objects by alignment with an image.* International Journal of Compu-

ter Vision, Vol. 2 (5), 1990, pp. 195-212.

[Joseph 1999] S.H. Joseph: *Analysing and Reducing the Cost of Exhaustive Correspondence Search.* Image and Vision Computing 17, 1999, pp. 815-830.

[Laubenheimer & Link 2003] A. Laubenheimer, N. Link: *Towards Adaptive Models for Classification of Technical Objects.* Proc. Vision Modelling and Visualization, München, 2003, pp. 319-327.

[Liu 1996] Jyh-Jong Liu: *A Model-Based 3-D Object Recognition System using Geometric Hashing with Attributed Features.* PhD thesis, New York University, Department of Computer Science, New York, 1996.

[Lo & Tsai 1997] R.C. Lo, W.H. Tsai: *Perspective-Transformation-Invariant Generalized Hough Transform for Perspective Planar Shape Detection and Matching.* Pattern Recognition, Vol. 30 (3), 1997, pp. 383-396.

[Lowe 1999] D.G. Lowe: *Object Recognition from Local Scale-Invariant Features.* Proc. 7th IEEE International Conference on Computer Vision ICCV, Vol. 2, 1999, pp. 1150-1157.

[Murase & Nayar 1995] H. Murase, S.K. Nayar: *Visual Learning and Recognition of 3D Objects from Appearance.* International Journal of Computer Vision, Vol. 14, 1995, pp. 5-24.

[Olson 1994] C.F. Olson: *Probabilistic indexing: A new method of indexing 3d model data from 2d image data.* Proceedings of the Second CAD-Based Vision Workshop, 1994, pp. 2-8.

[Pope & Lowe 1994] A.R. Pope, D.G. Lowe: Vista: A software environment for computer vision research. Proceedings IEEE Conference on Computer Vision and Pattern Recognition Recognition, 1994, pp. 768-772.

[Pope 1995] A.R. Pope: *Learning to recognize objects in images: acquiring and using probabilistic models of appearance.* PhD thesis, University of British Columbia, Canada, 1995.

[Pope & Lowe 2000] A.R. Pope, D.G. Lowe: *Probabilistic models of appearance for 3-D object recognition.* International Journal of Computer Vision, Vol. 40 (2), 2000, pp. 149-167.

[Rissanen 1978] J. Rissanen: *Modeling by shortest data description.* Automatica 14, pp. 465-471.

[Rothwell 1995] C.A. Rothwell: *Object Recognition Through Invariant Indexing.* Oxford University Press, Oxford, 1995.

[Schiele & Crowley 2000] B. Schiele, J.L. Crowley: *Recognition without Correspondence using Multidimensional Receptive Field Histograms.* Interna-

tional Journal of Computer Vision, Vol. 36 (1), 2000, pp. 31-52.

[Schmid & Mohr 1997] C. Schmid, R. Mohr: *Local Greyvalue Invariants for Image Retrieval.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19 (5), 1997.

[Schmid 1999] C. Schmid: *A structured probabilistic model for recognition.* Proc. Computer Vision and Pattern Recognition, Vol. 2, 1999, pp. 485-490.

[Shoemake 1994] K. Shoemake: *Euler Angle Conversion.* Graphics Gems IV, Academic Press, Editor: Paul Heckbert, 1994, pp. 222-229.

[Silverman 1986] B.W. Silverman: *Density Estimation for Statistics and Data Analysis.* Monographs on Statistics and Applied Probability, Chapman and Hall, London, 1986.

[Swain & Ballard 1991] M.J. Swain, D.H. Ballard: *Colour Indexing.* International Journal of Computer Vision, Vol. 7 (1), 1991, pp. 1132.

[Wolfson 1990] H.J. Wolfson: *Model-Based Object Recognition by Geometric Hashing.* Proceedings on the First European Conference on Computer Vision, Springer-Verlag, 1990, Berlin, pp. 526-536.

[Zisserman, Forsyth, Mundy & al 1995] A. Zisserman, D. Forsyth, J. Mundy et al: *3D object recognition using invariance.* Artificial Intelligence, Vol. 78, 1995, pp. 239-288.