



Rheinische      Institut für Informatik  
Friedrich-Wilhelms-      Abteilung VI  
Universität Bonn      Autonome Intelligente Systeme

Diplomarbeit

# Entropiebasierte interessante Regionen in 3D Punktwolken

Torsten Fiolka

Erstgutachter: Prof. Dr. Sven Behnke  
Zweitgutachter: Prof. Dr. Armin B. Cremers

22. November 2012





## Danksagungen

Einen ganz besonderen Dank möchte ich Jörg Stückler und Dominik Alexander Klein aussprechen für die Idee der Diplomarbeit, die Anleitung und die Zeit, die sie aufgebracht haben, um diese Arbeit zu ermöglichen.

Weiter danke ich allen Mitgliedern der Abteilung VI und insbesondere Herrn Dr. Nils Goerke für die Nutzung des Arbeitsplatzes im Praktikumslabor. Außerdem danke ich den anderen Diplomanden der Abteilung, meiner Familie für die fortwährende Unterstützung, Reinhold Kötter für die Arbeit beim Gegenlesen und Susanne und Thomas sowie Carolin und Aaron für die Zuvfügungstellung einiger Spielzeuge als Testobjekte.

## Versionshinweise

Version 1.0, 5.September 2011: Version zur Abgabe beim Prüfungsamt

Version 1.1, September 2011: Rechtschreibfehler korrigiert

Version 1.2, November 2012: Formel 4.5 korrigiert

Version 1.3, November 2012: Rechtschreibfehler korrigiert



## Kurzfassung

Die Wahrnehmung der Umwelt ist für Systeme, die autonom in ihr agieren sollen, ein grundlegender Bestandteil. Die Weiterentwicklung der Sensoren macht bessere Verfahren zur Wahrnehmung möglich, zeitgleich steigen aber auch die Anforderungen an die effiziente Datenverarbeitung.

In dieser Arbeit wird ein neuer Ansatz zur Detektion von Merkmalen und deren Beschreibung auf dreidimensionalen Punktwolken mit Farbinformation entwickelt. Es wird ein Detektor vorgestellt, der in Punktwolken Regionen mit hoher Oberflächenvarianz selektiert und ein Deskriptor, der diese Varianz und zusätzliche Farbeigenschaften in eine distinktive Beschreibung der Region umsetzt.

Als Maß für die Interessantheit einer Region wird die Shannon-Entropie auf lokalen Oberflächenorientierungen verwendet. Die Oberflächenorientierungen werden rotationsinvariant in einem Formdeskriptor erfasst und die Beschreibung wird um Farb- und Helligkeitsinformation in der Region ergänzt. Dabei erfolgt die effiziente Verwaltung der Daten in einem Oktalbaum. In Experimenten werden die Eigenschaften von Detektor und Deskriptor evaluiert.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Verwandte Arbeiten</b>	<b>3</b>
<b>3</b>	<b>Grundlagen</b>	<b>7</b>
3.1	Der Begriff <i>Merkmal</i> . . . . .	7
3.2	Punktwolke . . . . .	8
3.3	Sensoren . . . . .	8
3.4	Entropie . . . . .	11
3.5	Der Oktalbaum . . . . .	12
3.5.1	Motivation . . . . .	13
3.5.2	Struktur . . . . .	13
3.5.3	Aufbau und Effizienz . . . . .	14
3.6	Schätzung der Oberflächennormalen . . . . .	16
3.7	Mean-Shift . . . . .	17
3.8	Point Feature Histogram . . . . .	18
3.9	Earth Mover's Distance . . . . .	19
<b>4</b>	<b>Implementierung</b>	<b>21</b>
4.1	Softwareumgebung . . . . .	21
4.2	Implementierung des Oktalbaums . . . . .	22
4.2.1	Struktur eines Oktalbaumknotens . . . . .	22
4.2.2	Aufbau des Oktalbaums . . . . .	24
4.2.3	Diskretisierung der Punktwolke . . . . .	24
4.2.4	Bereichsanfragen im Oktalbaum . . . . .	26
4.3	Diskretisierung von Normalen . . . . .	28
4.3.1	Gestalt des Histogramms . . . . .	28
4.3.2	Eintragen der Normalen ins Histogramm . . . . .	30
<b>5</b>	<b>Detektion interessanter Regionen</b>	<b>31</b>
5.1	Prinzip des Detektors . . . . .	31
5.1.1	Die Abtastrate . . . . .	31
5.1.2	Die Histogrammskala . . . . .	32
5.1.3	Die Normalenabtastrate . . . . .	33

5.1.4	Die Normalenskala . . . . .	33
5.2	Schätzung von Oberflächennormalen . . . . .	36
5.2.1	Instabile Normalen . . . . .	37
5.2.2	Normalen bei großer Entfernung . . . . .	37
5.3	Entropieberechnung . . . . .	38
5.4	Extraktion von Merkmalen . . . . .	40
5.5	Eckigkeit . . . . .	42
5.6	Lokalisierung der Detektionen . . . . .	44
5.6.1	Subsampling . . . . .	45
5.6.2	Mean-Shift . . . . .	47
<b>6</b>	<b>Beschreibung interessanter Regionen</b>	<b>49</b>
6.1	Formdeskriptor . . . . .	49
6.1.1	Motivation . . . . .	49
6.1.2	Aufbau . . . . .	50
6.1.3	Distanzfunktion . . . . .	51
6.2	Farbdeskriptor . . . . .	57
6.2.1	Motivation . . . . .	57
6.2.2	Entwicklung und Aufbau . . . . .	57
6.2.3	Distanzfunktion . . . . .	58
6.3	Helligkeitsdeskriptor . . . . .	58
6.3.1	Motivation . . . . .	58
6.3.2	Aufbau . . . . .	61
6.3.3	Distanzfunktion . . . . .	61
<b>7</b>	<b>Experimente</b>	<b>63</b>
7.1	Experimente mit künstlich erzeugten Daten . . . . .	63
7.1.1	Beschreibung des Datensatzes . . . . .	63
7.1.2	Vorgehensweise . . . . .	65
7.1.3	Evaluation der Deskriptoren auf dem sechsfarbigem Würfel . . . . .	66
7.1.4	Experimente zur Bestimmung sinnvoller Parameter für den Detektor . . . . .	67
7.1.5	Experimente zum Einfluss von Sensorrauschen . . . . .	68
7.1.6	Einfluss von Eckigkeit und Entropieschwellwert . . . . .	69
7.2	Experimente zum Vergleich von Merkmalen zwischen Einzel- bildern . . . . .	83
7.2.1	Erstellung und Beschreibung der Datensätze . . . . .	83
7.2.2	Vorgehensweise . . . . .	84
7.2.3	Experimente . . . . .	86
7.3	Experimente zum Vergleich von Vollansichten mit Einzelbildern	92
7.3.1	Erstellung von Vollansichten . . . . .	92
7.3.2	Datensätze . . . . .	92
7.3.3	Vorgehensweise . . . . .	92

7.3.4	Experimente . . . . .	93
7.3.5	Diskussion der Ergebnisse . . . . .	93
7.4	Vergleich des Detektors mit dem „NARF Keypoint Detector“	100
7.4.1	Vorgehensweise . . . . .	100
7.4.2	Experimente . . . . .	100
7.4.3	Diskussion der Ergebnisse . . . . .	100
7.5	Laufzeitverhalten . . . . .	104
7.5.1	Diskussion der Laufzeit . . . . .	104
<b>8</b>	<b>Zusammenfassung</b>	<b>105</b>





# Kapitel 1

## Einleitung

In den vergangenen Jahrzehnten hat die Automatisierung in immer mehr Bereiche der Arbeitswelt und des Alltags Einzug gehalten. Dabei waren die Anforderungen auf einen programmierten Ablauf in einer kontrollierten Umgebung beschränkt. Die zukünftige Automatisierung erfordert vielmehr die Fähigkeit von Systemen, autonom mit der Umwelt zu agieren und auf eine Veränderung dieser zu reagieren. Als Beispiel sei hier die fortschreitende Unterstützung des Fahrers eines PKW durch computergestützte Systeme oder der Einsatz von Robotern in für den Menschen gefährlichen oder nicht erreichbaren Umgebungen. Für solche Systeme ist die Wahrnehmung der Umgebung essentiell: Ohne zuverlässige Informationen über die Umwelt sind fehlerfreie Entscheidungen zu Verhaltensweisen unmöglich.

In den letzten Jahren hat es im Bereich der Sensoren deutliche Weiterentwicklungen gegeben, so dass nicht mehr nur Bilddaten verfügbar sind, sondern auch genaue Messungen zur Größe von Objekten und deren Entfernung zum Sensor. Diese Entwicklung erfordert gleichzeitig neue Verfahren, die in der Lage sind, die gestiegene Menge an Daten effizient nach relevanten Informationen zu untersuchen. Der Ansatz dieser Arbeit liegt in der Reduktion aller Daten aus dem Sichtbereich auf relevante Regionen. Dazu werden dreidimensionale Punktwolken nach Regionen untersucht, die sich durch verschiedene Oberflächen auszeichnen. Die Idee ist, dass solche Regionen sehr distinktiv sind und sich daher sowohl zuverlässig finden als auch beschreiben lassen, um verschiedene Regionen unterscheiden zu können. Umgesetzt wird dies durch die Implementierung des Prinzips von Detektor und Deskriptor. Der Detektor wählt anhand eines Maßes Regionen aus der Punktwolke, der Deskriptor nutzt die verfügbaren Daten zur Beschreibung der Region.

Das Maß für den Detektor in dieser Arbeit ist die Shannon-Entropie auf den Oberflächenorientierungen in einer Region. Je höher die Variation in der Oberflächenorientierung ist, desto höher ist die Shannon-Entropie. Der Detektor findet lokale Maxima in diesem Maß und soll dabei in der

Lage sein, auch bei von Sensorrauschen beeinflussten Daten zuverlässige Ergebnisse zu erzielen. Es werden weiterhin verschiedene Techniken zur effizienten Verwaltung der Daten eingesetzt. Sind Regionen vom Detektor bestimmt worden, so gilt es, diese möglichst distinktiv zu beschreiben. Erst eine solche Beschreibung ermöglicht die Wiedererkennung gesehener Objekte oder die Zuordnung von bekannten Objekten. Diese Aufgabe wird durch den Deskriptor realisiert, der Daten zur Beschreibung der Form sowie der Farbe und relativer Helligkeit kombiniert.

## Kapitel 2

# Verwandte Arbeiten

Dieses Kapitel gibt einen Überblick darüber, ob und welche vergleichbaren Arbeiten in den Bereichen Computer Vision und Robotik existieren, und welche Gemeinsamkeiten und Unterschiede zur vorliegenden Arbeit bestehen.

### Bilddaten

Die Verarbeitung von Bilddaten ist ein wichtiger Bereich innerhalb der Computer Vision und eine große Menge an Arbeiten ist in diesem Zusammenhang zu finden. Die ausgewählten Arbeiten aus diesem Bereich zeigen Konzepte und Herangehensweisen an die Aufgabe, Merkmale zu extrahieren und zu beschreiben.

Der SIFT-Algorithmus, den Lowe in [Low99] beschrieben hat, extrahiert und beschreibt Merkmale auf Bilddaten. Der Algorithmus ist sehr weit verbreitet und wird vielfach bearbeitet, zitiert und verwendet. Er extrahiert Merkmale aus Bilddaten in vier Schritten:

- Detektion von interessanten Punkten in verschiedenen Skalen des Bildes. Die Interessantheit wird evaluiert, die Maxima in Ort und Skala werden zur weiteren Verarbeitung ausgewählt.
- Entfernung von instabilen Punkten und Verbesserung der Lokalisation der verbliebenen Punkte.
- Jedem Punkt wird eine Orientierung zugeordnet.
- Ein Deskriptor wird erstellt.

Der Ablauf des in meiner Arbeit vorgestellten Verfahrens gestaltet sich ähnlich zu dem des SIFT-Algorithmus, auch wenn sowohl die grundlegenden Daten als auch die eingesetzten Methoden andere sind.

Kadir und Brady haben in [KB01] die Shannon-Entropie auf Intensitäten im Bildern berechnet und als Grundlage für die Detektion interessanter Regionen verwendet. Dabei haben sie sich explizit mit der Skalenselektion beschäftigt; sie ziehen den Schluss, dass eine Region, die über mehrere Skalen eine hohe Entropie aufweist, nicht interessant sein kann und verwerfen diese (siehe S.90). Dies begründen sie damit, dass solche Regionen sehr selbstähnlich sind, z.B. rufen Fraktale oder Zufallsbilder genau diesen Effekt hervor. Außerdem gehen sie auf die Wahl der Skala ein, was für entropiebasierte Regionen wichtig ist, da viele Regionen nur auf bestimmten Skalen eine hohe Entropie erzeugen. Dies ist auch für diese Arbeit von Bedeutung; selbstähnliche Strukturen in Punktwolken sind dagegen für diese Arbeit interessant, ein einfaches Beispiel ist eine Ecke an einem Karton. In [KZB04] ist die Detektion weiter verbessert worden, insbesondere wurden Veränderungen der Daten bei affinen Transformationen behandelt.

## Punktwolken

Die Zahl der Arbeiten zu diesem Bereich ist geringer als bei Bilddaten, was vor allem auf der schwierigen Verfügbarkeit geeigneter Sensoren gründet. In Kapitel 3.3 werden die in dieser Arbeit verwendeten Sensoren vorgestellt.

In [JH99] beschreiben Johnson und Hebert dreidimensionale Oberflächen durch zweidimensionale Abbilder, genannt „Spin Images“. Allerdings erfolgt keine Detektion, die Ansatzpunkte für die Beschreibung werden frei gewählt, und das Ziel ist die Erkennung bereits bekannter und katalogisierter Objekte in Szenen mit sehr vielen anderen Objekten oder bei teilweiser Verdeckung. Pauly et al. detektieren und beschreiben in [PKG03] linienförmige Merkmale in Punktwolken auf unterschiedlichen Skalen. Sie berechnen mittels Hauptkomponentenanalyse die Kovarianzmatrix von Punktnachbarschaften. Aus den so gewonnenen Eigenvektoren berechnen sie die Krümmung des Nachbarschaftszentrums. Aus den Punkten, deren Krümmung einen Schwellwert übersteigt, erstellen sie Merkmale. Einige der in dem Beitrag verwendeten Methoden wende ich auch an, wenn gleich auch verändert. Die Hauptkomponentenanalyse zur Schätzung von Normalen wird auch in dieser Arbeit angewendet, allerdings nicht an jedem Punkt, sondern nur auf per Abtastrate ausgewählten Punktbereichen. Auch werden unterschiedlich große Punktnachbarschaften genutzt, um verschieden skalierende Merkmale zu extrahieren. Die Unterschiede liegen zum einen in der Art der Merkmale, zum anderen verwenden sie ausschließlich Vollansichten von Modellen.

In [NNS06] und [NN07] beschäftigen sich Novatnack et al. mit der Extraktion von Ecken- und Kantenmerkmalen in Polygonnetzen („Meshes“). Sie legen dabei sehr viel Wert auf die Erfassung der Skala eines Objektes. Die Detektion der Merkmale erfolgt durch Ecken auf Oberflächenkarten der Po-

lygonnetze.

Rusu et al. beschreiben in [RMBB08b] die „Point Feature Histograms“ (PFH) zur Beschreibung lokaler dreidimensionaler Strukturen in Punktwolken. In [RBB09] und [RHBB09] werden diese zur Verbesserung der Laufzeit überarbeitet und als „Fast Point Feature Histograms“ (FPFH) zur Segmentierung und Registrierung von Punktwolken vorgestellt. Diese sind Teil des in Kapitel 6.1 beschriebenen Deskriptors und werden daher auch in den Grundlagen in Kapitel 3.8 erläutert. Eine Weiterentwicklung der FPFHs sind die „View-point Feature Histograms“ aus [RBTH10], die aus dem Blickwinkel zu einem Objekt dessen Pose extrahieren.

Himmelsbach et al. stellen in [HLW09] eine Methode zur Segmentierung, Beschreibung und Verfolgung von Objekten vor. Das System basiert auf LIDAR-Sensoren und ist für Situationen wie die Navigation eines autonomen Systems im Straßenverkehr ausgelegt. Die Detektion erfolgt auf Segmenten in einem Belegungsgitter („occupancy grid“), die Beschreibung erfolgt mittels „Point Feature Histograms“.

Unnikrishnan und Herbert beschreiben in [UH08] eine Methode zur automatischen Skalenwahl auch bei starken Schwankungen in der Dichte der Punktwolke vor. Sie arbeiten dabei auch ausschließlich auf Punktwolken.

In [SRKB10] entwickeln Steder et al. einen Detektor und Deskriptor, genannt „Normal Aligned Radial Feature“ (NARF). Der Detektor sucht nach Oberflächen, die einer gegebenen Skala entsprechen und in ihrer Nähe einen substanziellen Tiefensprung aufweisen. Die erste Eigenschaft soll garantieren, dass das Objekt bei gleicher Skala wiedergefunden wird und eine stabile Normale berechnet werden kann, während die zweite Eigenschaft „Interessantheit“ repräsentieren soll. Ziel sind Merkmale an den Kanten und Ecken von Objekten, da man mit solchen Merkmalen die Form eines Objektes beschreiben kann und diese oft distinktiv ist. Ähnlichkeiten zu dieser Arbeit sind beim Deskriptor zu finden, dieser beschreibt die Umgebung um das Merkmal, orientiert an der Normale. Unterschiedlich ist zum einen die Arbeitsweise, da der komplette Algorithmus auf Tiefenbildern und nicht auf Punktwolken arbeitet, zum anderen konzentriert sich diese Arbeit auf die sichtbare Oberflächenstruktur von Objekten, während NARF die Kontur eines Objektes betrachtet. Während die Arbeit auf Tiefenbildern deutliche Vorteile beim Rechenzeitbedarf bietet, hat man wie bei Bilddaten den Nachteil, dass Punktnachbarschaften nicht dreidimensional ermittelt werden.



## Kapitel 3

# Grundlagen

In diesem Kapitel werden die theoretischen Grundlagen aller Datenstrukturen und Algorithmen, die in dieser Arbeit verwendet werden, eingeführt und kurz erklärt. Daneben wird ebenfalls der Begriff Merkmal definiert, sowie auf die Sensoren eingegangen, mit denen die für diese Arbeit notwendigen Daten erstellt werden. Ausführlich wird auf den Oktalbaum eingegangen, dessen Eigenschaften für die Arbeit von großer Bedeutung sind.

### 3.1 Der Begriff *Merkmal*

Der Begriff *Merkmal* ist, so häufig er im Bereich Computer Vision verwendet, nicht einheitlich definiert. Zum Beispiel definieren Pauly et al. in [PKG03]:

Features are usually defined as entities of an object that are considered important by a human for an accurate description of the object. This definition is highly subjective, however, and very difficult to express in algorithmic form.

Der Begriff *Merkmal* wird sehr häufig in sehr verschiedenen Kontexten und mit unterschiedlichen Bedeutungen verwendet, auch wenn sich diese meistens implizit aus dem Kontext, in dem sie verwendet werden, ergeben. Um Ungenauigkeiten und Zweideutigkeiten in dieser Arbeit zu vermeiden, definiere ich *Merkmal* als ein Tupel aus einer Positionsangabe und einer Beschreibung.

Die Positionsangabe kann zum Beispiel aus Bildkoordinaten in einem Bild bestehen oder aus 3D-Koordinaten in einer Punktwolke, wie es für die Merkmale in dieser Arbeit gilt. Die Position muss eindeutig sein.

Mit Beschreibung ist die formale Beschreibung von Eigenschaften gemeint. Zum Beispiel besteht der Deskriptor des SIFT-Algorithmus aus sechzehn Graustufengradientenhistogrammen, berechnet aus den Bilddaten in der Umgebung der Position.

## 3.2 Punktwolke

Eine Punktwolke ist eine Menge von Punkten im  $\mathbb{R}^3$ , die weitere Eigenschaften wie zum Beispiel Farbe aufweisen können. Die im folgenden verwendeten Punktwolken sind zum großen Teil Produkte von Tiefenkameras, die die Entfernung der Kamera zu ihrer Umgebung punktweise messen und aus diesen Tiefenbilder oder Punktwolken erzeugen. Hat man mehrere Ansichten eines Objektes von verschiedenen Seiten, so ist es möglich, diese zu einer Punktwolke zusammenzufassen und somit eine komplette Repräsentation dieses Objektes als Punktwolke zu erhalten. Dieser Vorgang wird mit „registrieren“ bezeichnet und erfordert einen gewissen Aufwand. Diese Punktwolken werden im folgenden auch „Vollansicht“ bezeichnet. Daneben wird auch die Möglichkeit genutzt, Punktwolken künstlich zu erzeugen, die ein bestimmtes Objekt repräsentieren sollen. Diese haben den Vorteil, dass Struktur und Form genau gewählt werden können. Sie eignen sich damit für Tests, in denen man externe Einflüsse wie technische Beschränkungen der Tiefenkameras ausschließen will.

## 3.3 Sensoren

Zur Erzeugung von Punktwolken gibt es verschiedene Methoden, die darauf beruhen, dass die Zeit gemessen wird, bis die Reflexion einer ausgesandten Welle beim Sensor eintrifft. Üblicherweise wird dabei ein Laserstrahl erzeugt und ein danebenliegender Sensor misst die Zeit, bis die Reflexion des Strahls zu sehen ist. Aus der Zeit lässt sich dann die Entfernung des Objektes berechnen, das den Laserstrahl reflektiert hat.

Eine Art von solchen Sensoren sind Laserentfernungssensoren („Laser Range Scanner“), die einen Laserstrahl per sich drehendem Spiegel in verschiedene Richtungen aussenden und damit die Umgebung vor sich zweidimensional mit einem begrenzten Blickwinkel abtasten. Üblich sind hier  $180^\circ$  bis  $270^\circ$ . Montiert man einen solchen Scanner auf ein Drehgelenk und neigt ihn während des Scan von oben nach unten, so erhält man ein dreidimensionales Bild aus mehreren zweidimensionalen Scans mit verschiedenen Neigewinkeln. Die Auflösung des Bildes hängt dabei sowohl vom Laser Range Scanner als auch von der Geschwindigkeit und Genauigkeit der Neigung ab. Allerdings dauert ein Scan bei guter Auflösung mehrere Sekunden.

Seit Mitte des letzten Jahrzehnts existieren daneben auch sogenannte „Time-



Abbildung 3.2: Time-of-flight Kamera



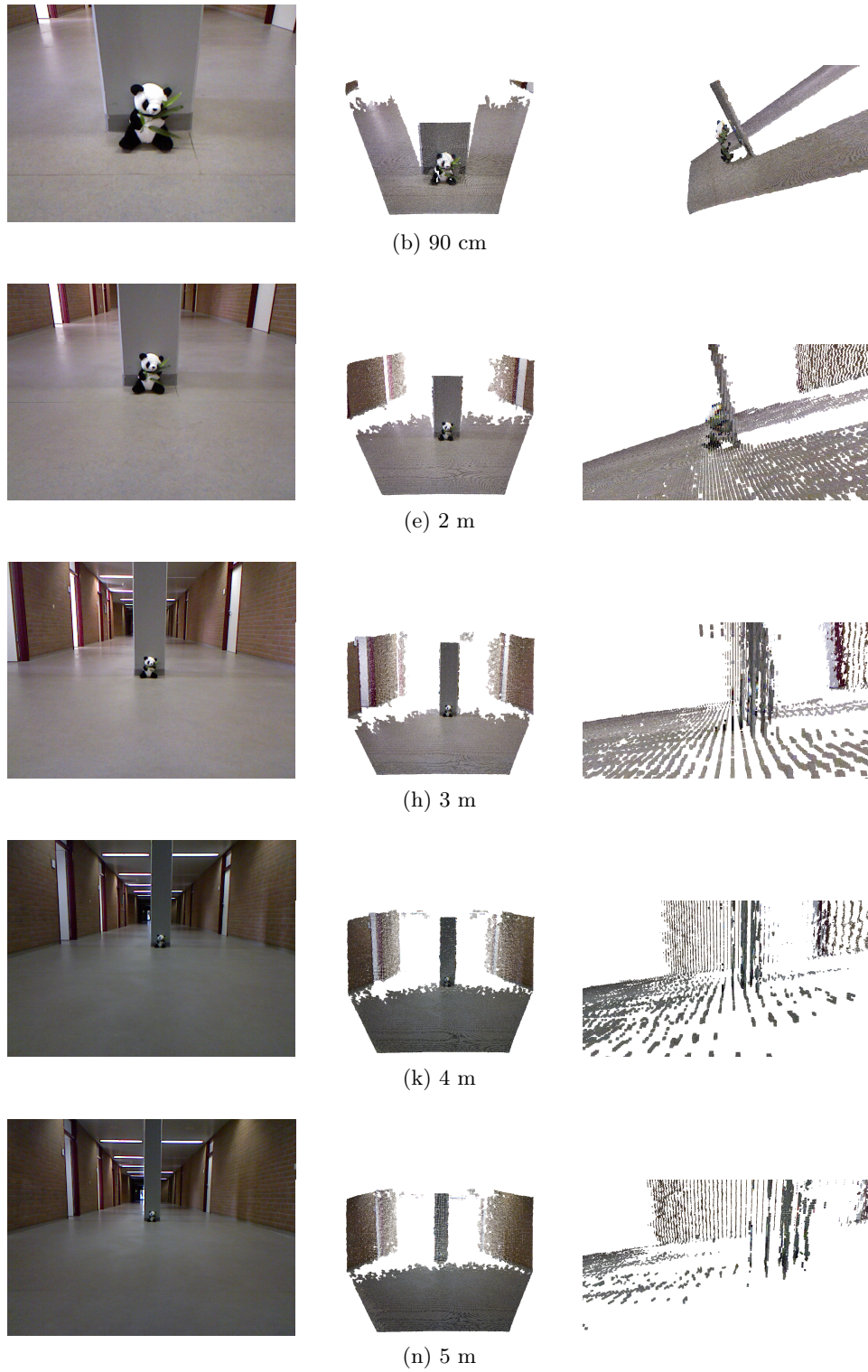


Abbildung 3.1: Aufnahme eines Teddybären aus verschiedenen Entfernungen.



Abbildung 3.3: Microsoft Kinect

of-Flight“ Kameras, die die Umgebung vor sich zunächst pulsweise illuminieren (siehe Abbildung 3.2). Ein CCD-Sensor misst das reflektierte Licht und die Zeit, die das Licht auf einem bestimmten Pixel gebraucht hat und berechnet daraus die Entfernung. Der Vorteil gegenüber Laserentfernungssensoren ist die hohe Bildfrequenz von bis zu 100 Hz, während die Auflösung meist darunter liegt und die Reichweite von der Wellenlänge des verwendeten Lichts begrenzt wird.

Der in dieser Arbeit primär benutzte Sensor ist die Kinect der Firma Microsoft, die auf patentierten Techniken der Firma PrimeSense beruht. Die Kinect ist als Erweiterung für die Konsole XBox 360 gedacht um ohne Eingabegeräte spielen zu können. Der Treiber, um die Kinect am PC nutzen zu können, ist in die Middleware ROS (siehe Kapitel 4.1) integriert und nutzt den Open Source Treiber der Firma PrimeSense als Grundlage. Die Kinect beinhaltet eine herkömmliche Kamera, einen Infrarot-Projektor, eine Infrarotkamera, Mikrofone, einen Motor für die vertikale Ausrichtung und einen Beschleunigungssensor. In dieser Arbeit werden die Kameras und der Infrarotprojektor benutzt.

Das Prinzip hinter der Kinect ähnelt dem der Time-of-Flight Kamera, allerdings wird die Umgebung vom Infrarot-Projektor nicht pulsweise illuminiert, sondern konstant, wobei das Licht strukturiert ist. Die Infrarotkamera misst die Reflexionen und berechnet aus den Störungen der Struktur die Entfernung an einem bestimmten Pixel. Die Daten werden dann als Tiefenbild zurückgegeben, der Treiber erzeugt daraus bei Bedarf eine Punktwolke. Das Bild der Kamera und das Tiefenbild werden registriert, so dass jedem Pixel ein Farbwert und dessen Tiefe zugeordnet werden kann.

Die Vorteile der Kinect gegen über den beschriebenen Sensoren sind: Sie liefert registrierte Tiefen- und Bilddaten, was bei Laserentfernungssensoren und Time-of-Flight Kameras eine zusätzliche Kamera erfordern würde, die zusammen mit dem Sensor entsprechend kalibriert werden müsste. Die Daten werden mit einer Auflösung von 640x480, während gängige Time-of-flight-Kameras im Bereich von 176x144 liegen, wenngleich es auch Modelle mit

Sichtfeld horizontal:	58°
Sichtfeld vertikal:	45°
Auflösung Tiefenbild:	640x480
Räumliche Auflösung (x/y):	3mm (in zwei Meter Abstand)
Tiefenauflösung (z):	1cm (in zwei Meter Abstand)
Maximale Bildwiederholfrequenz:	60Hz
Reichweite:	0,8m - 3,5m
Auflösung Farbbild:	SXGA (1280x1024)

Abbildung 3.4: Technische Daten der Microsoft Kinect

höherer Auflösung gibt. Dazu kommt die Bildwiederholfrequenz von 30Hz, die oben beschriebene Laserentfernungssensoren nicht erreichen können. Ein Kriterium aus wirtschaftlicher Sicht ist der Preis, der derzeit bei ungefähr 100 Euro liegt, während die anderen Sensoren mindestens um den Faktor 10 teurer sind.

Allerdings gibt es auch Einschränkungen: Der Tiefensensor funktioniert nicht bei starken Infrarotquellen, was zum Teil aber auch für die anderen Sensoren gilt. Das bedeutet, dass die Funktion im Freien nur eingeschränkt und bei Sonnenlicht nicht vorhanden ist. Zudem gibt es Probleme bei spiegelnden Oberflächen und solchen, die Infrarotlicht absorbieren. Einen Eindruck, wie gut die Qualität der Punktwolken der Kinect abhängig von der Distanz der Objekte vermittelt Abbildung 3.1.

### 3.4 Entropie

Der Begriff Entropie ist fachübergreifend; unter anderem gibt es den Begriff im Bereich der Thermodynamik in der Physik als auch in der Informationstheorie. Die Definition und Anwendung in der Informationstheorie geht auf Claude Elwood Shannon zurück und wird daher auch oft als Shannon-Entropie bezeichnet. Im Folgenden ist immer die Shannon-Entropie gemeint, wenn der Begriff Entropie benutzt wird. Shannon hat den Begriff in [Sha48] eingeführt. Er zeigte, dass der Informationsfluss auf einem Übertragungskanal durch die Kapazität des Kanals und den Informationsgehalt der Nachrichten beschränkt ist. Allerdings hat er auch auf die Nutzung der Entropie als Maß für Unsicherheit hingewiesen.

Die Definition der Entropie  $H$  auf einer Zufallsvariable  $P$  mit der diskreten Wahrscheinlichkeit  $(p_1, p_2, \dots, p_n)$  lautet:

$$H(P) = -K \sum_{i=1}^n p_i \log p_i \quad (3.1)$$

Im Fall  $p_i = 0$  wird  $p_i \log p_i := 0$  definiert.  $K$  ist eine positive Konstante, die zur Normalisierung auf den Wertebereich  $[0 : 1]$  dient und auch wegge-

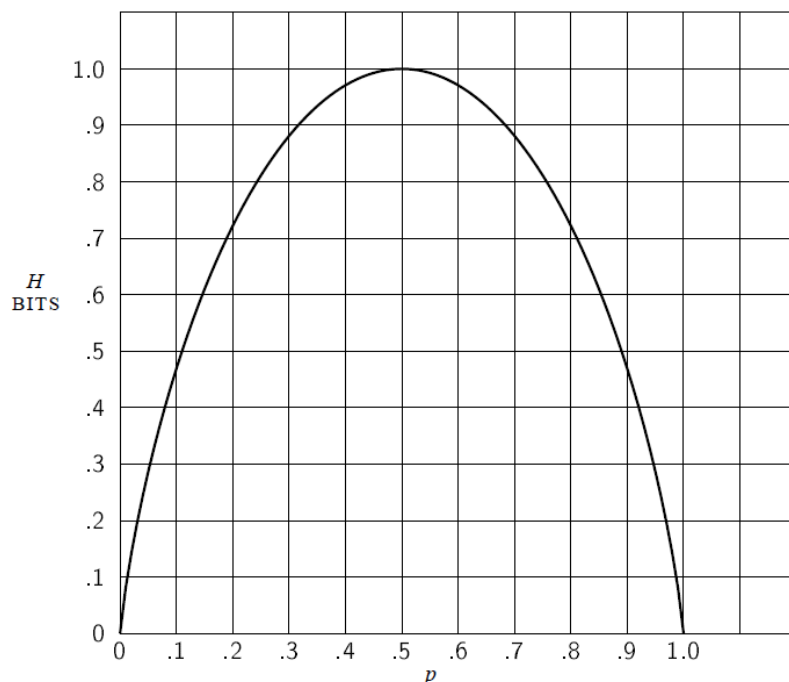


Abbildung 3.5: Entwicklung der Entropie für zwei Variablen mit den Wahrscheinlichkeiten  $p$  und  $1 - p$

lassen werden kann, sollte dies nicht erforderlich sein. Die Entropie  $H$  hat dabei folgende Eigenschaften, die in dieser Arbeit eine Rolle spielen:

$$\exists p_i \in P : p_i = 1 \Leftrightarrow H(P) = 0 \quad (3.2)$$

$$\forall p_i \in P : p_i = \frac{1}{n} \Leftrightarrow H(P) = \log n \quad (3.3)$$

Die Entropie wird in dieser Arbeit als Maß für die Unterschiedlichkeit der Oberflächen innerhalb einer Region im Raum eingesetzt. Dazu werden die vorkommenden Oberflächen geschätzt (siehe Kapitel 3.6), in Form eines Histogramms diskretisiert und die Entropie der Verteilung berechnet (siehe Kapitel 4.3). Verschiedene Oberflächen erzeugen unterschiedliche Einträge im Histogramm. Die Entropie des Histogramms wird größer, je näher die Verteilung im Histogramm der Gleichverteilung ist, und maximiert, falls eine solche vorliegt. Diese bedeutet, dass innerhalb einer Region alle durch die Diskretisierung möglichen Oberflächen in gleicher Anzahl vorliegen.

### 3.5 Der Oktalbaum

In diesem Abschnitt werden die theoretischen Grundlagen zum Oktalbaum vorgestellt, darunter die Struktur, die Funktionsweise des Aufbaus und An-

gaben zur Rechenzeit und Speicherplatzkomplexität. Einige Bemerkungen zu den von mir verwendeten Begriffen: Die Blätter des Baumes werden im folgenden Blattknoten, die „internen Knoten“ des Baumes Zweigknoten genannt.

### 3.5.1 Motivation

Einer der wichtigsten Punkte für diese Arbeit ist die effiziente Verwaltung der Daten. Die Punktwolken der Kinect umfassen 300.000 Punkte pro Aufnahme. Dazu kommen die Daten, die für die Schätzung der Oberflächen und für die anschließende Entropieberechnung notwendig sind. Anschließend muss möglich sein, die berechneten Daten effizient auf lokale Maxima zu untersuchen.

Die Implementierung des Oktalbaums, die mir zur Verfügung stand, löst einige dieser Probleme, so dass der Oktalbaum die grundlegende Datenstruktur dieser Arbeit geworden ist. Die Aufgaben sind im Einzelnen:

1. **Diskretisierung der Punktwolke**

Es ist nicht praktikabel, jeden Punkt in der Punktwolke daraufhin zu untersuchen, wieviel verschiedene Oberflächen es in seiner Umgebung gibt. Einerseits würde dies sehr viel Aufwand bedeuten, andererseits führt es zu Redundanzen, da zwei nah benachbarte Punkte sich in dieser Hinsicht kaum unterscheiden. Der Oktalbaum ermöglicht die Diskretisierung der Punktwolke in verschiedenen Auflösungen, so dass dieselbe Datenstruktur für Untersuchungen in verschiedenen Skalen benutzt werden kann.

2. **Speicherung regionsbezogener Daten**

Die Daten über Oberflächen und Entropie sind stark regionsbezogen. Die verwendete Implementierung des Oktalbaums bietet hier die Möglichkeit, Daten in der zugehörigen Region abzulegen. Für eine genaue Beschreibung siehe Kapitel 4.2.1.

3. **Effiziente Bereichsanfragen**

Der Oktalbaum bietet die Möglichkeit, Daten über die Punkte in Bereichen beliebiger Größen abzufragen. Dies inkludiert auch die in Punkt 2 zusätzlich gespeicherten Daten. Auch dies ist eine Eigenschaft der Implementierung, mehr dazu in Kapitel 4.2.4.

### 3.5.2 Struktur

Die Struktur des Oktalbaums entspricht weitestgehend der Struktur, die Samet in [Sam06] auf Seite 42ff mit dem Namen „PR-Quadtree“ beschrieben hat. Diese Struktur wird im folgenden erläutert, wobei sie anhand des Oktalbaums erklärt wird und Änderungen zur Struktur von Samet beinhaltet.

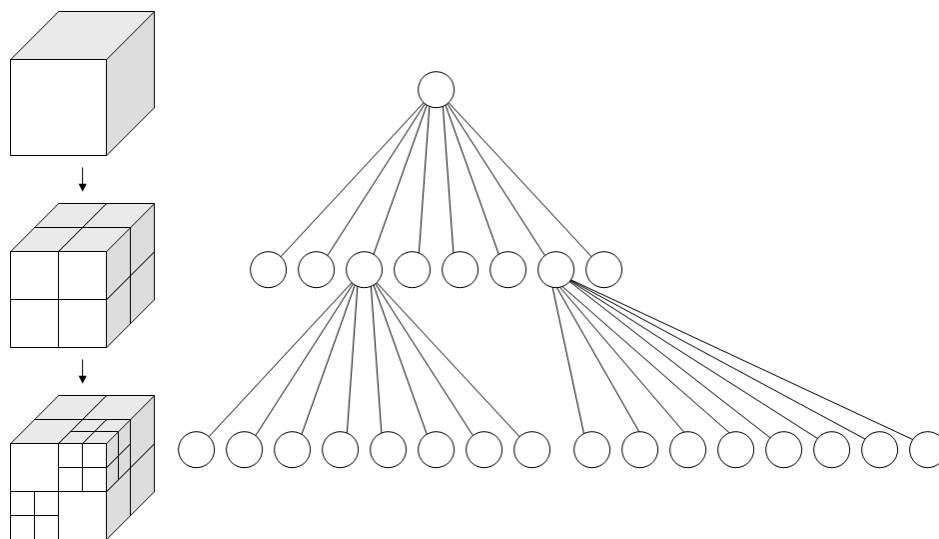


Abbildung 3.6: Schema eines Oktalbaums

Der Oktalbaum ist ein gewurzelter Baum, in dem jeder Zweigknoten bis zu acht Kinder hat. Jeder Knoten im Baum ist ein Kubus, und die Position der Kinder jedes Knoten entspricht der Position seiner Oktanten. Daher sind alle Knoten der gleichen Tiefe gleich groß. Die Größe und Zentrum der Wurzel müssen so gewählt sein, dass diese die Punktwolke komplett enthält. Hier gibt es den ersten Unterschied zu der von Samet beschriebenen Struktur: Bei Samet besitzt jeder Zweigknoten immer alle acht Kinder, auch wenn nicht jedes der Kinder Daten enthält. Die Zweigknoten in der hier verwendeten Implementierung besitzen nur in den Oktanten Kinder, die auch Daten enthalten, die anderen Oktanten sind leer. Die Speicherung der Daten erfolgt blattorientiert, die Daten sind hierbei die Punkte aus der Punktwolke mit Koordinaten im  $\mathbb{R}^3$ .

### 3.5.3 Aufbau und Effizienz

Gegeben sind die Position und das Zentrum der Wurzel. Der Aufbau erfolgt rekursiv ausgehend von der Wurzel. Beim Einfügen des ersten Punktes wird dieser in die Wurzel eingefügt und diese zurückgegeben. Bei jedem weiteren Punkt können folgende Möglichkeiten auftreten:

1. Wird der Punkt an einen Zweigknoten übergeben, so prüft dieser, in welchem seiner Oktanten der Punkt liegt. Existiert in dem Oktanten schon ein Knoten, so wird der Punkt an diesen Knoten übergeben. Ist dort noch kein Knoten vorhanden, so wird ein Blattknoten mit der Position und Größe des Oktanten erstellt, der Punkt in diesen eingefügt und der Blattknoten als Kind des Zweigknotens eingehängt.

2. Wird der Punkt an einen Blattknoten übergeben, so muss dieser Blattknoten aufgeteilt und in einen Zweigknoten umgewandelt werden. Danach werden einmal der ursprünglich im Blattknoten gespeicherte Punkt und der neu hinzugekommene Punkt an den neuen Zweigknoten zum Einfügen übergeben und es geht bei 1 weiter.

Daraus ergeben sich folgende Eigenschaften:

- Werden zwei Punkte mit geringem Abstand zueinander in den Oktalbaum eingefügt, so werden die entstehenden Zweigknoten so lange weiter verschachtelt, bis die beiden Punkte in unterschiedlichen Blattknoten liegen. Abhängig von der Größe der Wurzel erreicht ein Baum auch mit zwei Punkten größere Tiefen.
- Hat ein Zweigknoten nur ein Kind, so ist dieses ebenfalls ein Zweigknoten. Hat ein Zweigknoten ein Blattknoten als Kind, so hat er mindestens noch ein weiteres Kind. Dies folgt aus dem Aufteilen eines Knotens bei 2.
- Ein Sonderfall entsteht bei mir dadurch, dass ich meinem Baum eine Mindestknotengröße vorgebe. Ein Blattknoten mit dieser Größe wird nicht weiter aufgeteilt, auch wenn ein weiterer Punkt eingefügt wird. Die Tiefe des Baumes wird damit begrenzt. Das führt dazu, dass in Blattknoten die Daten mehrerer Punkte enthalten sein können. Diese Situation und der Umgang mit den Daten ist in Kapitel 4.2.1 beschrieben.

Zur Abschätzung der Effizienz der Datenstruktur ist im folgenden der Beweis zur Laufzeit- und Speicherplatzabschätzung aus [dBvKOS00] Seite 296 zusammengefasst. Das Theorem ist für einen Quadbaum aufgestellt, gilt aber auch für den Oktalbaum. Die Änderungen, die sich durch den Oktalbaum ergeben, sind (**hervorgehoben**) dargestellt.

**Theorem 14.2:** Ein Quadbaum (Oktalbaum) der Tiefe  $d$ , in dem eine Menge von  $n$  Punkten gespeichert ist, hat  $O((d+1)n)$  Knoten und kann in  $O((d+1)n)$  Schritten konstruiert werden.

Die Anzahl der Blattknoten wird durch die Anzahl der Zweigknoten abgeschätzt: Falls es keine Zweigknoten gibt, so besteht der Baum nur aus der Wurzel. Gibt es genau einen Zweigknoten, so ist das die Wurzel und kann vier (**acht**) Kinder besitzen. Gibt es zwei oder mehr Zweigknoten, so muss jeder Zweigknoten das Kind eines anderen Zweigknotens sein, ausgenommen die Wurzel. Jeder Zweigknoten, der vier (**acht**) Blattknoten als Kinder besitzt, reduziert damit die Anzahl der möglichen Blattknoten in allen über ihm liegenden Zweigknoten um eins. Damit kann es im Quadbaum maximal  $1 + (\text{Zweigknoten} * 3)$  Blattknoten (**Oktalbaum:  $1 + (\text{Zweigknoten} * 7)$  Blattknoten**) geben. Es genügt also, die Anzahl der Blattknoten mit der

Anzahl der Zweigknoten zu begrenzen. Als Anmerkung dazu sei gesagt, dass auch de Berg et al. einen Baum beschreiben, in dem jeder Zweigknoten alle Kinder besitzt, unabhängig davon, ob diese leer sind oder nicht. Da dies hier nicht der Fall ist, so ist die Anzahl der Knoten gleich oder niedriger. Jeder Zweigknoten hat mindestens einen Punkt in seinem Volumen. Dazu ist die Menge der Knoten einer Ebene disjunkt, das heißt es kann nicht mehr Knoten als Punkte auf einer Ebene geben. Damit ist die Grenze von  $O((d+1)n)$  bewiesen.

Der grösste Rechenzeitverbrauch beim Aufbau entsteht durch die Anzahl der Rekursionsschritte beim Einfügen eines Punktes. Im äußersten Fall muss jeder Punkt  $d$  Schritte durchlaufen, um endgültig einsortiert zu werden. Daher gilt auch hier die Grenze von  $O((d+1)n)$ .

### 3.6 Schätzung der Oberflächennormalen

Im folgenden wird eine Methode vorgestellt, die es ermöglicht, aus den Punkten einer Punktwolke eine lokale Schätzung der von ihnen repräsentierten Oberfläche zu ermitteln. Von einer Schätzung ist daher die Rede, weil zum einen keine weiteren Informationen über die vorliegenden Oberflächen bekannt sind. Ohne genaues Wissen über die Größe dieser Oberflächen kann und wird es passieren, dass die Punkte zwei oder mehrerer Oberflächen zusammen als Grundlage für die Schätzung einer Oberfläche dienen. Hinzu kommen technische Beschränkungen der Tiefenkameras, insbesondere durch Sensorrauschen und unterschiedliche Dichte der Punktwolke.

Gegeben sein eine Menge von Punkten  $P := \{p_1, p_2, \dots, p_n\} \in \mathbb{R}^3$ . Die Menge  $P$  steht stellvertretend für eine Teilmenge aus der Punktwolke, in dem die Normale geschätzt werden soll. Aus dem Mittelwert der Punkte in  $P$

$$\mu = \frac{1}{n} \sum_{i=1}^n p_i \quad (3.4)$$

wird mit dessen Hilfe die Kovarianz

$$Cov(P) = E[PP^T] - (E[P])(E[P])^T = \frac{1}{n} \sum_{i=1}^n p_i p_i^T - \mu \mu^T \quad (3.5)$$

der Koordinaten der Punkte in  $P$  bestimmt. Die Kovarianz wird mit Hilfe des Verschiebungssatzes der Varianz ausgerechnet, da es bei der Implementierung möglich ist, die Summen  $\sum p_i$  und  $\sum p_i p_i^T$  effizient für kleine Bereiche der Punktwolke zu erstellen. Auch können die Summen mehrerer Bereiche integriert werden, siehe Kapitel 4.2.1. Auf der Kovarianzmatrix wird eine Hauptkomponentenanalyse durchgeführt, um die Eigenvektoren der Kovarianzmatrix zu bestimmen.



Der größte Eigenvektor repräsentiert die Richtung der Punkte in  $P$ , in der die Varianz maximal ist. Der zweite Eigenvektor steht senkrecht auf dem ersten und gibt die Richtung der zweitgrößten Varianz der Punkte an. Von diesen beiden Vektoren wird eine Ebene aufgespannt, die den Verlauf der Punktmenge relativ gut wieder gibt (siehe Abbildung 3.7). Der dritte und kleinste Eigenvektor steht senkrecht auf den ersten beiden und damit auch senkrecht auf der Ebene. Dieser wird als Schätzung der Oberflächennormale verwendet.

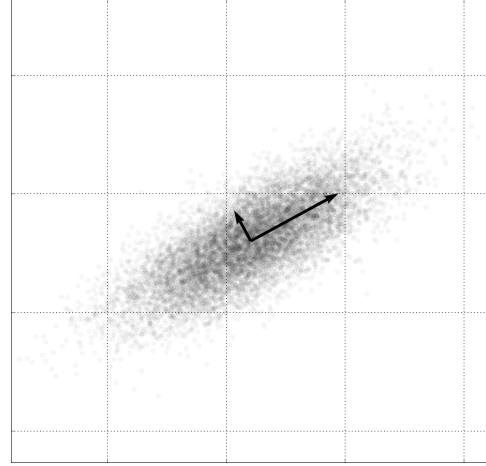


Abbildung 3.7: Eigenvektoren einer zweidimensionalen Punktmenge

### 3.7 Mean-Shift

Der „Mean-Shift Algorithmus“ wurde von Fukunaga und Hostetler in [FH75] zur Schätzung des Gradienten einer Wahrscheinlichkeitsdichtefunktion vorgestellt. Derzeit findet der Algorithmus in abgewandelten Formen Anwendung in Bereichen wie Segmentierung, Clustering oder Tracking. Im folgenden wird die Funktionsweise des Algorithmus erläutert. Grundlage für den Algorithmus ist die Parzen-Fenster-Methode zur Kerndichteschätzung. Gegeben sei ein Kern  $K$ , ein Parameter für die Bandbreite  $h$  und eine Menge von Punkten  $X = \{x_1, x_2, \dots, x_n\}$  in Dimension  $d$ . Dann ist der Kerndichteschätzer definiert als:

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (3.6)$$

Mithilfe dieser Gleichung lässt sich jetzt der Gradient der Dichtefunktion schätzen:

$$\nabla \hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^n K'\left(\frac{x - x_i}{h}\right) \quad (3.7)$$

Setzt man nun die Normalverteilung ein, so ergibt sich:

$$\nabla \hat{f}(x) = \sum_{i=1}^n (x_i - x) \exp\left(-\frac{(x_i - x)^2}{2h^2}\right) \quad (3.8)$$

Der Versatz zwischen dem Zentrum des Fensters  $x$  und  $x_i$  wird gewichtet und der Mittelwert des gewichteten Versatzes ist die Schätzung des Gradienten. Der Algorithmus verschiebt nun das Fenster in Richtung des Gradienten entsprechend dessen Steigung und wiederholt die Berechnung solange,

bis entweder eine festgelegte Anzahl von Iterationen erreicht oder die Steigung des Gradienten einen Schwellwert unterschreitet. Anwendung findet der Algorithmus in dieser Arbeit zur genaueren Bestimmung lokaler Entropiemaxima zur Lokalisation der Merkmale, siehe Abschnitt 5.6.2.

### 3.8 Point Feature Histogram

Das Point Feature Histogram (PFH) dient zur Beschreibung einer Region einer Oberfläche durch das Verhältnis zwischen den Normalen an verschiedenen Punkten in dieser Region. Beschrieben als „Surflet-Pair-Relation Histograms“ in [WHH03] sind diese unter anderem in [RMBB08b] und [RMBB08a] zur Klassifikation von Regionen in Punktwolken verwendet worden. Im folgenden wird die Struktur wie in [WHH03] beschrieben zusammengefasst.

Sei  $\circ$  das Skalarprodukt zwischen zwei Vektoren,  $\| \cdot \|_2$  die euklidische Norm eines Vektors,  $|\cdot|$  der Betrag einer reellen Zahl und ein Surflet ein Paar  $(p, n)$  aus einer Position  $p$  im  $\mathbb{R}^3$  und der Oberflächennormale  $n$  an Punkt  $p$ . Gegeben sei ein Paar von Surflets  $(p_1, n_1)$  und  $(p_2, n_2)$  mit

$$|n_1 \circ (p_2 - p_1)| \leq |n_2 \circ (p_2 - p_1)| \quad (3.9)$$

Dann wird  $p_1$  als Ursprung für ein von  $(u, v, w)$  aufgespanntes Koordinatensystem gewählt:

$$u = n_1 \quad (3.10)$$

$$v = \frac{(p_2 - p_1) \times u}{\|(p_2 - p_1) \times u\|} \quad (3.11)$$

$$w = u \times v \quad (3.12)$$

Schließlich wird definiert:

$$\alpha = \arctan(w \circ n_2, u \circ n_2) \quad (3.13)$$

$$\beta = v \circ n_2 \quad (3.14)$$

$$\gamma = u \circ \frac{p_2 - p_1}{\|p_2 - p_1\|} \quad (3.15)$$

$$\delta = \|p_2 - p_1\| \quad (3.16)$$

Das Tupel  $S = \{\alpha, \beta, \gamma, \delta\}$  spezifiziert das Verhältnis zwischen den beiden Surflets  $(p_1, n_1)$  und  $(p_2, n_2)$ . Durch die Wahl des eigenen Koordinatensystems ist es unabhängig von den tatsächlichen Koordinaten. Das Koordinatensystem selbst mit Ursprung  $p_1$  ist durch die Eigenschaft aus Gleichung 3.9 stabil definiert. Damit soll sichergestellt werden, dass die Winkel zwischen den Normalen gegenüber Rotationen invariant sind.

Die möglichen Wertebereiche der Einträge  $S$  für zwei Surflets  $(p_i, n_i)$  und  $(p_j, n_j)$  sind wie folgt:

$$\alpha \in [-\pi, \pi] \quad (3.17)$$

$$\beta \in [-1, 1] \quad (3.18)$$

$$\gamma \in [-1, 1] \quad (3.19)$$

$$\delta \in [0, \max_{i,j} (\|p_i - p_j\|)] \quad (3.20)$$

Wahl et al. unterteilen die Wertebereiche der einzelnen Komponenten in jeweils fünf Klassen und erstellen so ein Histogramm mit  $5^4 = 625$  Klassen. Nachdem die „Surflet-Pair-Relations“ aller Surflets einer Region paarweise eingetragen sind, wird das Histogramm entsprechend der Punktdichte der Region normalisiert und das „Surflet-Pair-Relation Histogram“ der Region ist fertig.

Rusu et al. ändern in [RBB09] und [RHBB09] folgende Punkte: Der Wert  $\delta$  wird entfernt, da dieser nach ihren Angaben wenig zur Stabilität der Histogramme beiträgt und bei geringerer Punktdichte die Entfernung desselben sogar vorteilhaft ist. Desweiteren kürzen sie die Berechnung der Relationen auf die  $k$ -nächsten-Nachbarn zu jedem Surflet statt der paarweisen Berechnung mit allen anderen Surflets der Menge. Dem Histogramm werden dann noch Anteile der Histogramme der  $k$ -nächsten-Nachbarn aufaddiert. Da dies zu redundanten Berechnungen führen könnte, legen sie einen Cache an, um bereits berechnete Histogramme zwischenspeichern. Außerdem schlagen sie vor, die resultierenden Werte in einzelne Histogramme mit  $n$  Klassen einzutragen und diese einfach zu konkatenieren. Dies führt zu einem Deskriptor mit  $3n$  Einträgen. Die resultierenden Histogramme nennen sie „Fast Point Feature Histograms“ (FPFH). Die Nutzung der FPFHs in dieser Arbeit wird in 6.1 beschrieben.

### 3.9 Earth Mover's Distance

Rubner et al. untersuchen in [RTG00] die Eigenschaften der „Earth Mover's Distance“ ( $EMD$ ) als Distanz zweier Verteilungen im Anwendungsgebiet Content Based Image Retrieval. Der Name rührt daher, dass die  $EMD$  die Arbeit misst, die man braucht, um einen Erdhaufen in einen anderen zu transformieren. Dieses Prinzip kann man auf Histogramme anwenden. Die  $EMD$  gibt dann die minimalen Kosten einer Transformation von einem Histogramm zum anderen wieder.

Gegeben seien zwei Histogramme  $P$  und  $Q$ ,  $f_{ij}$  sei der Fluss zwischen  $P_i$  und  $Q_j$  und  $d_{ij}$  sei die Distanz zwischen den Klassen  $i$  und  $j$ . Dann ist die  $EMD$  definiert als:

$$EMD(P, Q) = \frac{\min_{f_{ij}} \sum_{i,j} f_{ij} d_{ij}}{\sum_{i,j} f_{ij}} \quad (3.21)$$

mit den Einschränkungen:

$$f_{ij} \geq 0 \quad (3.22)$$

$$\sum_j f_{ij} \leq P_i \quad (3.23)$$

$$\sum_i f_{ij} \leq Q_j \quad (3.24)$$

$$\sum_{i,j} f_{ij} = \min(\sum_i P_i, \sum_j Q_j) \quad (3.25)$$

Pele und Werman haben diese in [PW08] und [PW09] zur effizienten Berechnung modifiziert und diese dann  $\widehat{EMD}$  genannt:

$$\widehat{EMD}_\alpha(P, Q) = (\min_{f_{ij}} \sum_{i,j} f_{ij} d_{ij}) + |\sum_i P_i - \sum_j Q_j|^\alpha \max_{i,j} d_{ij} \quad (3.26)$$

Für  $\alpha = 1$  und auf eins normalisierte Histogramme ändert sich die Distanz von  $\widehat{EMD}$  gegenüber  $EMD$  nicht. Zudem schlagen sie eine begrenzte Distanzfunktion vor:

$$d_t(a, b) = \min(d(a, b), t) \text{ mit } t > 0 \quad (3.27)$$

Zusammen mit anderen Änderungen zeigen Pele und Werman, dass sich die  $\widehat{EMD}$  schneller als die  $EMD$  von Rubner et al. berechnen lässt. Zudem ist die nach oben begrenzte Distanzfunktion bei Vergleichen von Farben bzw. Farbdeskriptoren nützlich, da sie nur angibt ob sich zwei Farben ähnlich sind oder ob sie es nicht sind. Ein Grund für die  $\widehat{EMD}$  sind die guten Ergebnisse in der Anwendung von Farbdeskriptoren, wie in [PW09] gezeigt, als auch die effiziente Berechenbarkeit auf großen Histogrammen. In dieser Arbeit wird die Implementierung von Pele verwendet, die als Quellcode frei verfügbar ist<sup>1</sup>.

---

<sup>1</sup><http://www.cs.huji.ac.il/~ofirpele/FastEMD/code/>, zuletzt abgerufen am 24.08.11 14:00 MEST

## Kapitel 4

# Implementierung

Dieses Kapitel beschäftigt sich mit Details der Implementierung der Arbeit. Es wird die benutzte Softwareumgebung kurz eingeführt sowie detailliert auf die Implementierung des Oktalbaums eingegangen, dessen Implementierungsdetails für die Funktion des Detektors wichtig sind. Außerdem wird die Vorgehensweise bei der Diskretisierung von Normalen beschrieben.

### 4.1 Softwareumgebung

Die Arbeit ist in der Programmiersprache C++ unter Linux entstanden und in die Middleware ROS<sup>1</sup> eingebunden. Weitere Abhängigkeiten sind im Folgenden erläutert. ROS steht für „Robot Operating System“ und ist der Name einer Middleware für Robotikanwendungen. Angestoßen von der Firma Willow Garage wird diese von einer weltweiten Gemeinschaft als Open Source weiterentwickelt mit dem Ziel einer gemeinsamen Plattform für Anwendungen in der Robotik. Das System bietet neben Hardwaretreibern verschiedene von der Hardware unabhängig implementierte Algorithmen der Robotik und Computer Vision. Die Abhängigkeiten innerhalb von ROS umfassen den Treiber für die Kinect, diverse Pakete zur Nachrichtenübermittlung zwischen Programmen sowie die „Eigen“-Bibliothek, die Teilprobleme der linearen Algebra implementiert.

Die „Point Cloud Library“, kurz PCL<sup>2</sup>, ist ein Open Source Projekt zur Verarbeitung von Punktwolken. Sie stellt Algorithmen und Datenstrukturen auf Punktwolken zur Verfügung und ist komplett in ROS integriert, für mehr Informationen siehe [RC11]. Aus der PCL werden Datenstrukturen für Punktwolken, Algorithmen zur Hauptkomponentenanalyse und die umfangreiche Werkzeugsammlung zur Visualisierung benutzt. Weitere Abhängigkeiten sind das ROS-Paket „octreelib“<sup>3</sup> von Jörg Stückler, das die Implementie-

---

<sup>1</sup><http://www.ros.org>

<sup>2</sup><http://www.pointclouds.org>

<sup>3</sup><http://www.ros.org/wiki/octreelib>

rung des Oktalbaums enthält, sowie die „Fast Earth Mover’s Distance“ von Ofir Pele.

Entwickelt und getestet wurde die Arbeit auf verschiedenen Computern mit Ubuntu Linux in den Versionen 10.04, 10.10 und 11.04 für die ROS-Version „diamondback“.

## 4.2 Implementierung des Oktalbaums

Die grundlegende Struktur des Oktalbaums wurde in 3.5 beschrieben, in diesem Abschnitt werden die Details der Implementierung desselben beschrieben. Dies umfasst die Arbeitsweise der Bibliothek „octreelib“ von Jörg Stückler als auch die Struktur der Daten, die neben den Punkten der Punktwolke im Oktalbaum gespeichert werden.

### 4.2.1 Struktur eines Oktalbaumknotens

Jeder Oktalbaumknoten besteht aus zwei Komponenten. Einmal der Knoten selbst, der Informationen zu seiner Größe, der Position und seinen Kindern beinhaltet. Zudem werden in jedem Knoten die Koordinaten des Punktes aus dem Volumen des Knotens gespeichert, deren euklidischer Abstand zum Zentrum des Knotens minimal ist. Diese Koordinaten sind der Ausgangspunkt aller Bereichsanfragen von diesem Knoten aus und werden auch, falls aus diesen Knoten ein Merkmal hervorgeht, als Position des Merkmals genutzt. Dies hat den Vorteil, dass alle Berechnungen und die Beschreibung von echten Daten der Punktwolke ausgehen und nicht von diskretisierten Daten aus den Zentren der Oktalbaumknoten. Es kann allerdings zu leichten Ungenauigkeiten kommen, wenn die „nächsten Punkte“ benachbarter Knoten nicht die gleichen Abstände voneinander haben.

Die zweite Komponente ist eine Datenklasse, die jeder Knoten zusätzlich enthält. Zum einen integriert diese Klasse die Eigenschaften aller Punkte, die im Volumen des Knotens enthalten sind, als auch die Daten, die im Laufe der Detektion gewonnen werden und regionsbezogen sind.

Jeder neue Punkt, der in den Oktalbaum eingefügt wird, durchläuft alle Knoten, in deren Volumen er liegt. Dabei werden die Daten des Punktes in die Datenklasse des passierten Oktalbaumknotens integriert. Die folgenden Eigenschaften werden für jeden eingefügten Punkt  $p_i \in P$  in einem Knoten  $K$  gespeichert:

1. Die Summen der einzelnen Koordinaten der Punkte im  $\mathbb{R}^3$  :

$$\sum_{p_i \in P} p_i = \left( \sum_{p_i \in P} x_{p_i} \quad \sum_{p_i \in P} y_{p_i} \quad \sum_{p_i \in P} z_{p_i} \right) \quad (4.1)$$

2. Eine Matrix mit den Summen der miteinander multiplizierten Koordinaten im  $\mathbb{R}^3$  :

$$\sum_{p_i \in P} p_i p_i^T = \begin{pmatrix} \sum x_{p_i} x_{p_i} & \sum x_{p_i} y_{p_i} & \sum x_{p_i} z_{p_i} \\ \sum x_{p_i} y_{p_i} & \sum y_{p_i} y_{p_i} & \sum y_{p_i} z_{p_i} \\ \sum x_{p_i} z_{p_i} & \sum y_{p_i} z_{p_i} & \sum z_{p_i} z_{p_i} \end{pmatrix} \quad (4.2)$$

3. Die Anzahl der Punkte  $P$  in  $K$ .
4. Die Summe der Farbwerte der Punkte im RGB-Farbraum:

$$\sum_{p_i \in P} r_{p_i}, \sum_{p_i \in P} g_{p_i}, \sum_{p_i \in P} b_{p_i}. \quad (4.3)$$

Die Daten der Punkte 1. - 3. werden für die Schätzung der Oberflächennormalen benötigt, wie in Kapitel 5.2 beschrieben. Über die Daten in den Punkten 3. und 4. wird der durchschnittliche RGB-Wert des Knotens ermittelt. Jeder Knoten besitzt damit alle nötigen Informationen zur Schätzung der Normalen auf den Punkten in seinem Volumen, die er ohne diese Daten bei seinen Kindern abfragen müsste.

Desweiteren können für den Bereich jedes Knotens folgende Daten berechnet und dort gespeichert werden:

- Die Oberflächennormale, falls sie geschätzt worden ist.
- Der Status der Normale. Der Status gibt an, ob sie nicht berechnet wurde, ungültig oder gültig ist. Eine Normale kann ungültig sein, wenn die Anzahl der für die Schätzung berücksichtigten Punkte zu niedrig ist. Nur gültige Normalen werden für die Entropieberechnung verwendet.
- Die Normalendiskretisierung. Diese wird für die Normale erstellt, falls sie geschätzt worden ist.
- Das Entropiehistogramm. In diesem Histogramm werden die Normalen der Umgebung des Knotens aufsummiert und daraus die Entropie berechnet.
- Die Entropie des Entropiehistogramms.
- Der Status des Knotens. Dieser beschreibt, ob ein Knoten ein Maximum der Entropie in seiner Umgebung ist, sein könnte oder nicht sein kann. Bei der Suche nach Maxima können viele Knoten als potenzielle Maxima schnell ausgeschlossen werden, das wird dann entsprechend in den Status geschrieben, und die Knoten werden bei der weiteren Suche nicht weiter betrachtet.

Diese Daten werden nur bei Bedarf berechnet und werden nicht beim Aufbau des Baumes gebraucht und verändert. Die Datenspeicherung im Oktalbaumknoten hat den Vorteil, dass diese Daten durch Bereichsanfragen zur Berechnung über die Nachbarschaft eines Knotens im Oktalbaum gefunden werden können. Dadurch lassen sich zum einen redundante Berechnungen vermeiden, zum Beispiel wird jede Normale je nach Wahl der Parameter in vielen Knoten zur Entropieberechnung benötigt. Anstatt nun jedes Mal die Normale neu zu rechnen, wird diese nur beim ersten Mal berechnet und danach nur noch aus dem Oktalbaum ausgelesen.

### 4.2.2 Aufbau des Oktalbaums

Der Wurzel des Oktalbaums ist ein Würfel mit Kantenlänge von  $2^{13}$  cm = 81,92 Metern und dem Zentrum im Ursprung des Koordinatensystems. Die Größe wurde so gewählt, dass die Kantenlängen aller Knoten Zweierpotenzen ausgehend von 1 cm sind. Die Kantenlänge der Wurzel reicht aus, um Punktwolken aus den zur Verfügung stehenden Sensoren aufzunehmen. Um die Tiefe des Baumes zu begrenzen, gibt es eine minimale Kantenlänge von 5mm. Der Wert wurde gewählt als Kompromiss zwischen einer möglichst guten Auflösung für die Berechnungen gegenüber dem erhöhten Rechenzeitbedarf und der Genauigkeit der Punktwolken mit Sensorrauschen.

Der Aufbau des Baums erfolgt wie in Kapitel 3.5 beschrieben. Um den Rechenzeitbedarf beim Einfügen eines Punktes gering zu halten, wird vorher ein passender Oktalbaumknoten erstellt. Die Position des Knotens wird auf die Koordinaten des Punktes eingestellt, und die Datenklasse wird mit den oben beschriebenen Werten mit einem Punkt initialisiert. Jeder Zweigknoten, den der neue Knoten durchläuft, addiert die Datenklasse des neuen Knotens zu seiner eigenen. Ist der neue Knoten auf der korrekten Position angekommen, werden Maße und Position des Knotens richtig gesetzt. Einen Überblick über den Ablauf des Einfügens gibt der Pseudocode-Algorithmus 4.1. Die finale Anzahl der Knoten im Oktalbaum ist abhängig von der Punktwolke, die gespeichert wird. Eine Punktwolke von der Kinect umfasst maximal 307.200 Punkte. Ein Teil der Werte sind ungültig, im Schnitt sind bleiben etwa 250.000 gültige Punkte. Je nach Szene ergibt das einen Oktalbaum mit 150.000 bis 350.000 Knoten.

### 4.2.3 Diskretisierung der Punktwolke

Sind alle Punkte in den Baum eingefügt, werden alle Knoten des Baums entsprechend ihrer Tiefe in Listen eingetragen, im folgenden Tiefenliste genannt. Jede Tiefenliste enthält alle Knoten einer Tiefe. Die Tiefe bestimmt die Kantenlänge eines Knotens, somit ist jede Tiefenliste eine Abtastung der Punktwolke mit einer Abtastrate, die dieser Kantenlänge entspricht. Bei der Verwendung der Tiefenlisten muss auf folgende Sonderfälle geachtet werden:





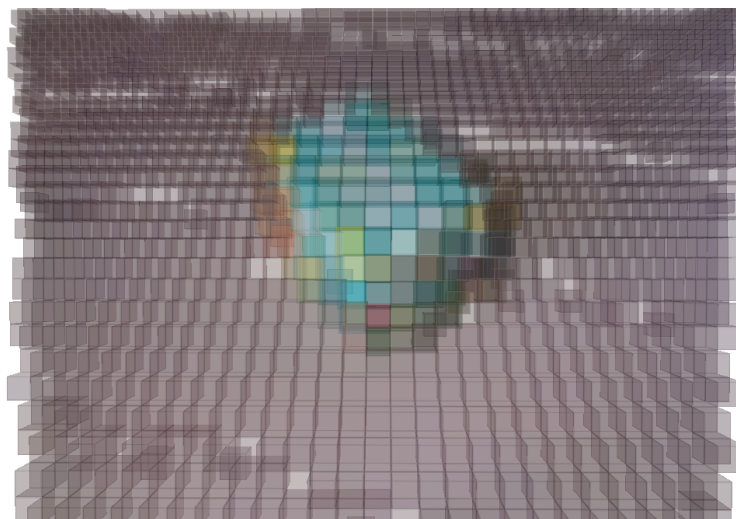
(a) Bild des Kartons



(b) Punktwolke des Kartons



(c) Oktalbaum in Tiefe entsprechend 1cm Auflösung



(d) Oktalbaum in Tiefe entsprechend 4cm Auflösung

Abbildung 4.1: Diskretisierung einer Punktwolke mittels Oktalbaum

---

**Algorithmus 4.1** Einfügen eines Knotens in den Oktalbaum
 

---

**Require:** Knoten  $n$  mit Koordinaten von Punkt  $p_n$ 

```

if aktueller Knoten  $k$  ist Blattknoten then
  addiere die Daten von  $n$  zu den Daten von  $k$ 
  Erstelle neuen Blattknoten  $l$  mit Koordinaten von Punkt  $p_k$ 
  Wandle  $k$  in Zweigknoten um
  Füge  $l$  in  $k$  ein
  Füge  $n$  in  $k$  ein
else
  addiere die Daten von  $n$  zu den Daten von  $k$ 
  for all Kindknoten  $k_i \in k$  do
    if  $p_n \in k_i$  then
      Füge  $n$  in  $k_i$  ein
    end if
  end for
end if

```

---

- Blattknoten kann es in jeder Tiefe des Baumes geben, Es kommt vor, dass einzelne Punkte in einer Tiefe gespeichert werden, die geringer ist, als die des Baumes. Diese Punkte liegen meist am Rand der Punktwolke oder in schlecht aufgelösten Bereichen der Punktwolke, und es trifft nur einen kleinen Teil der Punkte. Diese Blattknoten werden in den Tiefenlisten größerer Tiefe nicht erneut berücksichtigt.
- Je nachdem, wie gut die Auflösung des Sensors ist, geben die Tiefenlisten der kleinsten Abtastraten keine vollständige Abtastung der Punktwolke wieder. Bei einem Laserentfernungssensor mit einer Winkelauflösung von 0.25 Grad zwischen den Laserstrahlen wären zwei Punkte in 10 Metern Entfernung mindestens 4 cm voneinander entfernt und würden nicht in Knoten mit einer Kantenlänge von 5 mm gespeichert werden. Allerdings ist diese Eigenschaft von Vorteil, da schwach aufgelöste Bereiche nicht geeignet sind, um auf kleinen Skalen Normalen zu berechnen oder Merkmale zu suchen.

Die Berechnungen zur Merkmalsextraktion finden ausschließlich auf diesen Listen statt, was die Rechenzeit je nach gewählter Abtastrate deutlich reduziert.

#### 4.2.4 Bereichsanfragen im Oktalbaum

Mithilfe des Oktalbaums lassen sich Daten und Knoten in beliebigen Bereichen abfragen. Die Anfragen verfolgen keine unnötigen Wege im Baum und die Suchtiefe ist auf die Tiefe des Baumes beschränkt, daher sind diese Bereichsanfragen sehr schnell zu berechnen. Weiterhin kann man die Anfrage

auf eine maximale Tiefe der Knoten begrenzen, wenn man den Baum nicht bis in die größte Tiefe durchsuchen möchte. Es sind verschiedene Anfragen implementiert, die im Folgenden erläutert werden:

### **Anfrage über alle Knoten in einem Bereich**

Zum Start übergibt man die Koordinaten des Bereichs an die entsprechende Methode des Startknotens, üblicherweise die Wurzel. Der Rückgabewert ist eine Liste von Knoten, die im gesuchten Bereich liegen. Dazu wird der Baum rekursiv durchlaufen und es werden folgende Bedingungen geprüft:

- Ist der aktuelle Knoten ein Blattknoten, wird geprüft, ob dieser komplett im Suchbereich enthalten ist. Ist er es, so wird er der Menge der gesuchten Knoten hinzugefügt, ist er es nicht, wird abgebrochen.
- Ist der aktuelle Knoten ein Zweigknoten, so wird er zunächst der Rückgabemenge hinzugefügt. Anschließend wird bei allen Kindern geprüft, ob sich die Koordinaten des Suchbereichs und die Koordinaten der Kinder überlappen, das heißt dass evtl. das Kind oder tieferliegende Kinder im Suchbereich enthalten sind oder sein könnten. In diesem Falle wird die Rekursion auf dem Kind fortgesetzt. Andernfalls wird dieser Kindknoten ignoriert.

### **Bereichsanfrage über alle Knoten einer bestimmten Tiefe**

Die Suche nach allen Knoten auf einer bestimmten Tiefe inkludiert mehrere Abfragen:

- Ist der aktuelle Knoten ein Blattknoten, wird geprüft, ob dieser komplett im Suchbereich enthalten ist. Wenn ja, wird geprüft, ob er sich in der gesuchten Tiefe befindet. Ist er es, so wird er der Menge der gesuchten Knoten hinzugefügt, ist er es nicht, wird abgebrochen.
- Ist der aktuelle Knoten ein Zweigknoten, wird geprüft, ob die Tiefe des Knotens gleich der gesuchten Tiefe ist. Falls ja, so wird er zurückgegeben und die Anfrage abgebrochen, da die Kinder in jedem Fall in größerer Tiefe liegen. Falls nein, so ist die Tiefe geringer als die gesuchte Tiefe. Es wird dann bei allen Kindern geprüft, ob sich die Koordinaten des Suchbereichs und die Koordinaten der Kinder überlappen. Falls dies zutrifft, wird auf den Kindern die Anfrage fortgesetzt.

Der Rückgabewert ist auch hier eine Liste von Knoten.

### **Anfrage über die integrierten Daten eines Bereiches**

Die Anfrage nach den integrierten Daten aller Punkte in einem Bereich liefert eine Instanz der Datenklasse zurück. Der Vorgang ist wie folgt:

- Der aktuelle Knoten wird daraufhin geprüft, ob er komplett im Suchbereich enthalten ist. Wenn ja, wird die Datenklasse des Knotens auf die Rückgabedatenklasse addiert. Ist der Knoten ein Zweigknoten, so werden die Kinder auf Überlappung mit dem Suchbereich geprüft und die Anfrage bei ihnen fortgesetzt, ansonsten wird die Suche abgebrochen.

Bei der Anfrage nach den integrierten Daten ist zu beachten, dass keine Knoten integriert werden, die nicht vollständig im Suchbereich liegen. An den Rändern des Suchbereichs kann es daher vorkommen, dass Knoten verworfen werden, obwohl einer oder mehrere Punkte dieses Knotens im Suchbereich liegen würden.

## 4.3 Diskretisierung von Normalen

Die Diskretisierung der Oberflächennormalen sollte zum einen ihre Ausrichtung möglichst gut wiedergeben als auch effizient berechenbar sein. Im Laufe der Arbeit sind einige Ansätze ausprobiert und verworfen worden, diese werden im Folgenden vorgestellt.

### 4.3.1 Gestalt des Histogramms

#### Zwei eindimensionale Histogramme

Die erste Überlegung bestand aus zwei eindimensionalen Histogrammen, die die Winkel der Normale einmal gegenüber der z-Achse und gegenüber der x-Achse bilden. Der Polarwinkel  $\theta \in [0, \pi]$  ist der Winkel eines Vektors mit der positiven z-Achse. Der Azimutwinkel  $\phi \in [-\pi, \pi]$  ist der Winkel eines Vektors mit der positiven x-Achse, gegen den Uhrzeigersinn ermittelt. Problematisch ist hier, dass abhängig davon, wie nah eine Normale den Polen liegt, der Azimutwinkel eine unterschiedlich starke Aussagekraft über die Lage einer Normale hat. Zwei Normalen, die sich bei einem Polarwinkel von  $\frac{\pi}{2}$  gegenüberliegen, sind bei einem Polarwinkel von Null identisch. Daher könnten recht ähnliche Normalen, so sie denn in der Nähe eines der Pole lägen, in unterschiedliche Klassen bezüglich ihres Azimutwinkels eingetragen werden. Diese Eigenschaft würde das Histogramm erheblich verzerren, daher wurde der Ansatz nicht weiter verfolgt.

#### Ein platonischer Körper als Grundlage

Eine weitere Möglichkeit, die untersucht wurde, war die Verwendung eines platonischen Körpers, des Ikosaeders (zwanzigseitiger Würfel), zur Einteilung der Klassen. Die Ecken des Ikosaeders waren hierbei die Zentren der Histogrammklassen, die Entfernung einer Klasse zu ihren nächsten Nachbarklassen ist damit überall gleich. Das bedeutet, dass jede Klasse von der

Oberfläche der Einheitskugel einen gleich großen Bereich abdeckt. Diese Eigenschaft ist wünschenswert, da somit ähnliche Normalen auch ähnliche Einträge im Histogramme zur Folge haben. Die Berechnung der Histogrammeinträge ist ebenfalls unkompliziert, hierzu wird paarweise das Skalarprodukt des Normalenvektors und der Vektoren der Klassenzentren ausgerechnet, um den Winkel zwischen den Vektoren zu erhalten und damit die Klassen mit minimalem Winkelabstand zu wählen. Problematisch ist die mit zwanzig geringe Anzahl von Klassen. Zwar erfolgen die Einträge gewichtet, aber je nach Größe des Gewichts zeigen sich unterschiedliche Ergebnisse. Entspricht das maximale Gewicht der Winkeldistanz zweier Klassen, so werden die Normalen, die nahe eines Klassenzentrums liegen, sehr stark in diese Klasse eingetragen und sehr schwach in die benachbarten. Normalen, die zwischen die Klassenzentren fallen, werden dagegen gleich in die benachbarten Zentren eingetragen. Das führt zu Instabilitäten der Entropie auch bei leichten Schwankungen der Normalen. Ein Lösungsansatz ist, die maximale Winkeldistanz für einen Eintrag in eine Histogrammklasse zu erhöhen, z.B. auf die doppelte Distanz zwischen benachbarten Klassen. Jede Normale wird damit auf mehrere Klassen verteilt, was die Unterschiede zwischen den erzeugten Verteilungen im Histogramm deutlich reduziert, unabhängig davon, ob eine Normale nun nah bei einem Klassenzentrum liegt oder nicht. Unter dieser Glättung leidet jedoch die Genauigkeit, die mit nur zwanzig Histogrammklassen nicht sonderlich hoch ist. Dies führt bei vielen Normalen schnell zu Histogrammen, die einer Gleichverteilung recht ähnlich sehen. Die wiederum sorgt überall für hohe Werte der Entropie und macht es schwer, einen Grenzwert zu ziehen, ab dem lokale Maxima ermittelt werden. Dies kann dazu führen, dass die späteren Merkmale instabil sind und setzt die Qualität des Detektors sehr herab. Auch dieser Ansatz wurde daher nicht weiter verfolgt.

### Die Approximation gleichgroßer Bereiche auf der Einheitskugel

Das schließlich verwendete Verfahren stammt aus dem Appendix von [Sha06]. Dort ist beschrieben, wie eine Menge von Einheitsvektoren im  $\mathbb{R}^3$  erzeugt wird, so dass jeder der Vektoren eine ungefähr gleichgroße Fläche auf der Einheitskugel repräsentiert. Dazu legt man die Anzahl der Vektoren fest, die man entlang des Polarwinkels haben möchte. Die Anzahl der Vektoren entlang des Azimuthwinkels  $\phi$  hängt dann vom Polarwinkel  $\theta$  ab. Sei  $n_\theta$  die Anzahl der Vektoren entlang des Polarwinkels. Abhängig vom Winkel  $\theta_i$  mit  $i \in [0, n_\theta]$  ist die Anzahl  $n_{\phi i}$  der Vektoren entlang des Azimuthwinkels wie folgt definiert:

$$n_{\phi i} = 2n_\theta \sin \theta_i + 1 \quad (4.4)$$

Die resultierenden Vektoren sind derart verteilt, dass der Abstand eines Vektors zu allen seinen nächsten Nachbarn ungefähr gleich ist. Je kleiner die Zahl  $n_\theta$  ist, desto größer ist allerdings der Fehler der Approximation, so dass die



Abbildung 4.2: Klassen des Normalenhistogramms.

repräsentierten Bereiche bei kleinen  $n_\theta$  unterschiedlich groß sind. Die Festlegung der Zahl  $n_\theta = 7$  ist ein Kompromiss zwischen der Größe des Fehlers auf der einen Seite und des Rechenzeitbedarfs auf der anderen. Die resultierenden 66 Vektoren der Klassenzentren werden bei der Initialisierung des Programms einmalig berechnet und statisch gespeichert.

#### 4.3.2 Eintragen der Normalen ins Histogramm

Die Normalen werden abhängig von der Entfernung zum Zentrum einer Histogrammklasse gewichtet eingetragen. Das Gewicht einer Normale in einer bestimmten Histogrammklasse lässt sich nun über das Skalarprodukt der Normale mit dem der Klasse zugehörigen Vektor bestimmen. Da beide Vektoren die Länge 1 besitzen, entspricht das Skalarprodukt dem Cosinus des zwischen den vektoren eingeschlossenen Winkels. Legt man nun eine maximale Winkeldistanz fest, so lässt sich das Gewicht wie folgt bestimmt:

Sei  $H$  das Histogramm und  $V_H$  die Menge aller Vektoren des Histogramms. Entsprechend seien  $h_i \in H$  und  $v_i \in V_H$  die  $i$ -te Histogrammklasse bzw. der  $i$ -te Histogrammvektor für  $i \in \{1, n_H\}$ . Weiter sei  $\alpha$  die maximale Winkeldistanz. Dann ist

$$w_i := \begin{cases} 0, & \text{falls } \arccos(n * v_i) > \alpha, \\ 1 - \frac{\arccos(n * v_i)}{\alpha}, & \text{sonst,} \end{cases} \quad (4.5)$$

das Gewicht der Histogrammklasse  $h_i$  für die Normale  $n$ .

Je nach Wahl der Maximaldistanz ist die Gesamtsumme des Histogramms ungleich eins. Daher wird das Histogramm nach Eintragen der Normale normalisiert. Diese Diskretisierung einer Normale wird in dem Knoten des Oktalbaums gespeichert, von dem ausgehend sie berechnet worden ist.

## Kapitel 5

# Detektion interessanter Regionen

Diese Kapitel widmet sich der Detektion von interessanten Regionen in der Punktwolke. Es wird die Vorgehensweise beschrieben, die für die Detektion relevanten Parameter sowie die Funktionsweise der einzelnen Komponenten. Außerdem wird auf Herausforderungen eingegangen, die im Laufe der Arbeit zusätzlich gelöst werden mussten, unter anderem die Lokalisierung der Merkmale und die Unterdrückung instabiler Merkmale.

### 5.1 Prinzip des Detektors

Zu Beginn sind eine Punktwolke und der auf dieser erzeugte Oktalbaum gegeben. Durch den Oktalbaum ist eine Diskretisierung der Punktwolke in Größe der Abtastrate möglich. Ausgehend von einem Abtastungsschritt dieser Diskretisierung werden in einem durch die Histogrammskala festgelegten Bereich die Oberflächennormalen geschätzt und in einem Histogramm integriert. Von diesem Histogramm wird die Entropie berechnet. Ist dies für alle Punkte der Abtastrate geschehen, so wird nach lokalen Maxima der Entropie in diesen Punkten gesucht. Im Gegensatz zu Merkmalen in zweidimensionalen Bilddaten ist im dreidimensionalen die Größe der dargestellten Objekte gegeben. Damit entfällt eine Skalenselektion, wie sie in Bilddaten nötig ist, und die Festlegung auf eine gewünschte Größe der Merkmale ist möglich.

#### 5.1.1 Die Abtastrate

Die Abtastrate bezeichnet die Genauigkeit, mit der die Punktwolke für die Suche nach Maxima in der Entropie aufgelöst wird. Dazu wird ausgehend von jedem Knoten, dessen Kantenlänge der Abtastrate entspricht, die Entropie der Oberflächenrichtungen in einem Bereich der Punktwolke um den Knoten herum berechnet. Die Abtastrate hat sowohl Einfluss auf die Qualität der

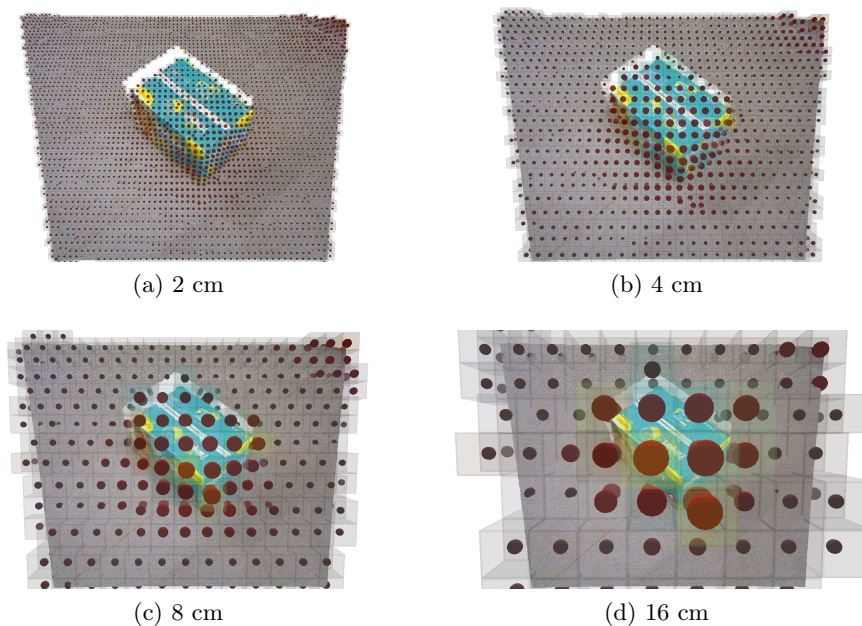


Abbildung 5.1: Berechnung der Entropie auf verschiedenen Abtastraten. Größere Kugeln und stärkeres Rot bedeuten höhere Entropiewerte. Die entsprechenden Oktaalbaumknoten sind als transparente Quader angedeutet.

Ergebnisse als auch auf die Laufzeit.

Je kleiner die Abtastrate ist, desto höher ist die Auflösung der diskretisierten Punktwolke und desto feiner besser wird die Struktur der Oberfläche ausgewertet (siehe Abbildung 5.1). Allerdings steigt mit einer feineren Abtastrate auch der Rechenzeitbedarf der Detektion. Ist die Abtastrate zu grob eingestellt, desto wahrscheinlicher ist, dass ein Maximum der Entropie im Grenzbereich zwischen zwei oder Oktaalbaumknoten liegt und nicht aufgelöst werden kann. Dies kann zu verpassten Maxima führen. Bei großen Abtastraten tritt verstärkt der Fall auf, dass ein Maximum durch die Diskretisierung versetzt vom tatsächlichen Maximum gefunden wird. Mehr dazu in Kapitel 5.6.

### 5.1.2 Die Histogrammskala

Die Histogrammskala bezeichnet die Größe der Region um einen Punkt, in der alle Normalen eingesammelt und in das Entropiehistogramm eingetragen werden. Ausgehend von einem Knoten in der Abtastrate wird zur Entropieberechnung eine Bereichsanfrage im Oktaalbaum gestellt, die alle Knoten der Größe der Normalenabtastrate findet. Diese Bereichsanfrage ist ein Kubus mit Kantenlänge entsprechend des Histogrammbereichs, zentriert auf den „nächsten Punkt“ der Punktwolke zum Zentrum des Knotens der Abtastra-



te. Die Histogrammskala bestimmt die Größe der interessanten Regionen und ist damit entscheidend für die Größe der Objekte, die vom Detektor als interessant wahrgenommen werden (siehe Abbildung 5.2). Sie sollte mindestens doppelt so groß sein wie die Abtastrate. Dadurch überlappen sich benachbarte Histogrammskalen. Weiter kann eine im Vergleich zur Histogrammskala große Normalenabtastrate dazu führen, dass die Entropie nur auf wenigen Normalen berechnet wird. Dies führt zu instabilen Maxima. Ist dagegen die Histogrammskala im Verhältnis zu den anderen Werten sehr groß, so erhöht sich der Rechenzeitbedarf.

### 5.1.3 Die Normalenabtastrate

Diese bezeichnet die Abtastrate, auf der die Normalen geschätzt werden (siehe 5.3). Dies wird umgesetzt, indem die Normalen auf Oktalbaumknoten mit Kantenlänge der Abtastrate geschätzt werden. Der Wert spiegelt die Genauigkeit wieder, mit der die Oberfläche der Punktwolke diskretisiert wird. Die Wahl der Größe der Normalenabtastrate hängt sehr mit der des Histogrammbereichs zusammen. Ist die Normalenabtastrate im Vergleich sehr hoch, so werden nur wenige Normalen in das Entropiehistogramm eingetragen; dies bedeutet, dass numerische Ungenauigkeiten und Sensorrauschen in der Punktwolke und daraus resultierende „schlechte“ Normalenschätzungen sehr viel Gewicht in der Entropie bekommen und so das Ergebnis verfälschen. Eine im Vergleich kleine Normalenabtastrate bedeutet, dass sehr viele Normalen ins Histogramm eingetragen werden, was den Einfluss einzelner Normalen auf die Entropie stark mindert. Allerdings erhöht dies den Rechenzeitbedarf.

### 5.1.4 Die Normalenskala

Die Normalenskala bezeichnet die Kantenlänge des Kubus in der Punktwolke, in dem die Normale geschätzt wird (siehe Abbildung 5.4). Entspricht diese der Normalenabtastrate, so wird die Normale einfach mit den im entsprechenden Oktalbaumknoten gespeicherten Daten geschätzt. Ansonsten wird eine Bereichsanfrage im Oktalbaum gestellt und die Daten aller Punkte der Punktwolke in diesem Bereich werden zur Schätzung der Normale genutzt. Während ersterer Fall der deutlich performantere ist, gibt es bei kleinen Normalenskalen das Problem, dass die geschätzten Normalen sehr wenig Kontext in der Punktwolke haben. Das führt an Tiefenkanten dazu, dass die Normalen instabil werden und sich teilweise in Richtung der Kante neigen, siehe Abbildung 5.5.

Je größer die Normalenskala ist, desto mehr Grundlage in der Punktwolke hat die Schätzung einer Normale und solche Instabilitäten werden vermieden. Auch werden Unregelmäßigkeiten in der Punktwolke, zum Beispiel



(a) 4 cm



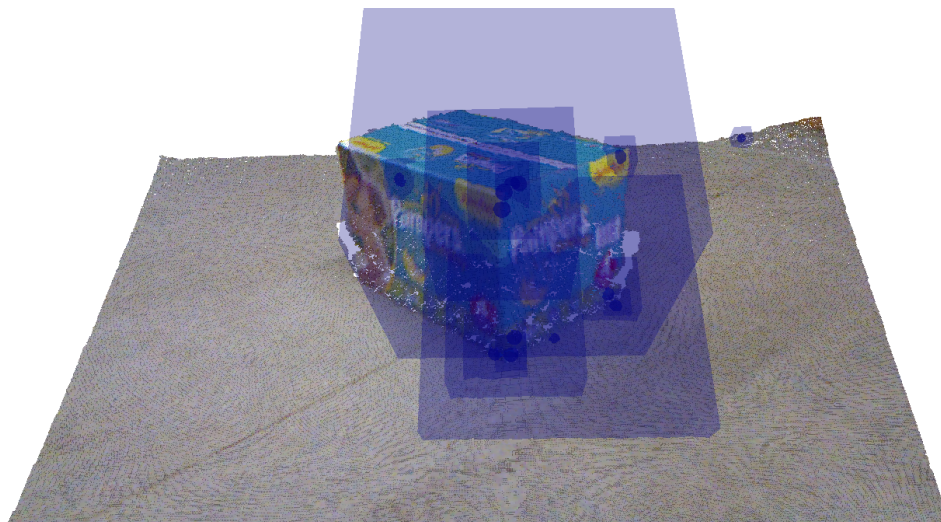
(b) 8 cm



(c) 16 cm

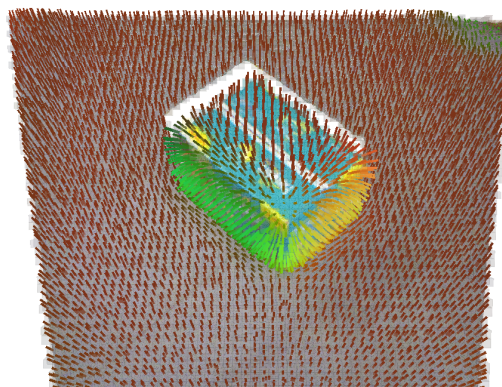


(d) 32 cm

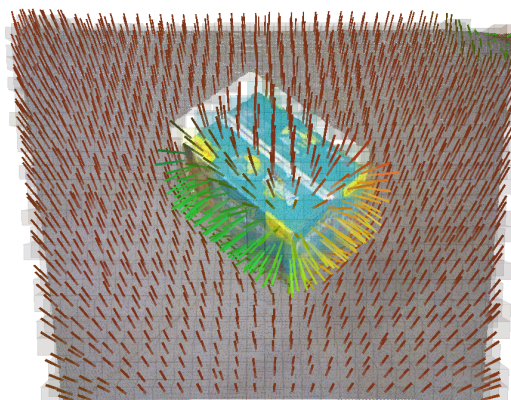


(e) 4 cm - 32 cm gleichzeitig

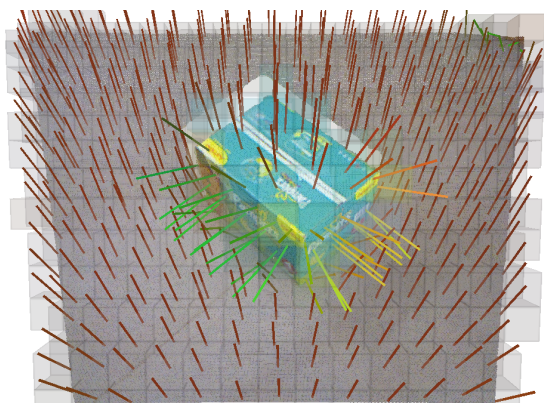
Abbildung 5.2: Merkmale des Kartons auf verschiedenen Skalen.



(a) 2 cm



(b) 4 cm



(c) 8 cm

Abbildung 5.3: Schätzung der Oberflächennormalen auf verschiedenen Normalenabtastraten mit einer Normalenskala von 16 cm. Die entsprechenden Oktalbaumknoten sind als Quader angedeutet.

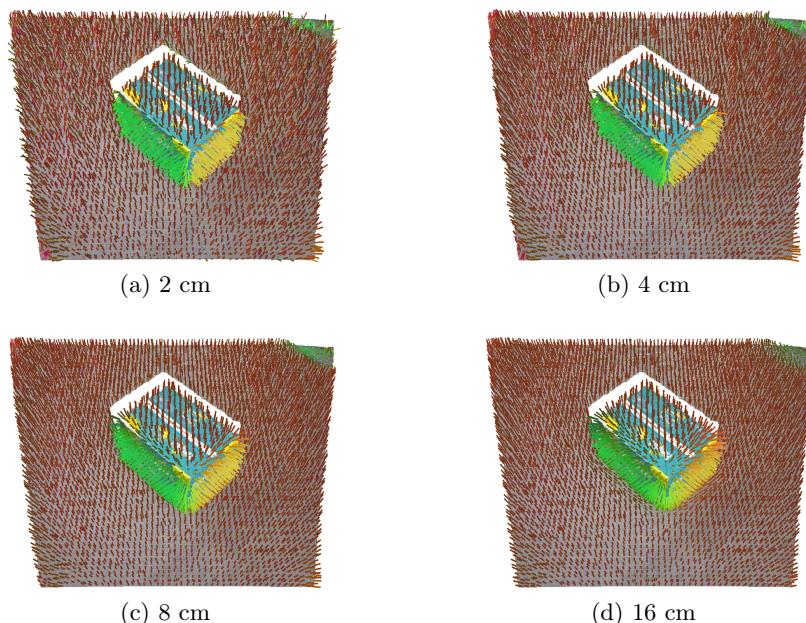


Abbildung 5.4: Schätzung der Oberflächennormalen in verschiedenen Normalenskalen und einer Normalenabtastrate von 2cm.

durch Sensorrauschen, geglättet, was die Qualität der Normalen weiter erhöht. Nachteilig ist zum einen der hohe Rechenzeitbedarf, zum anderen werden eventuell relevante Oberflächenänderungen zu sehr geglättet und nicht mehr erfasst. Daher sollte die Normalenskala immer in Bezug zur Histogrammskala und Normalenabtastrate stehen.

## 5.2 Schätzung von Oberflächennormalen

Die Theorie der Schätzung der Oberflächennormalen ist in Kapitel 3.6 beschrieben, die effiziente Vorhaltung der benötigten Daten in Kapitel 4.2.1. Im folgenden wird die genaue Vorgehensweise erläutert, mit der Normalen schließlich im Programm geschätzt werden.

Soll auf einem Knoten eine Normale berechnet werden, so wird geprüft, ob die Normalenskala kleiner oder größer als die Kantenlänge des Knotens ist. Ist sie größer als die Kantenlänge, so wird eine Bereichsanfrage in Größe der Normalenskala auf der Position des Knotens an den Oktalbaum gestellt. Die Bereichsanfrage liefert die integrierten Daten über alle Punkte im Bereich zurück, unter anderem die Summen  $\sum p_i$ ,  $\sum p_i p_i^T$  (siehe die Formeln 4.1 und 4.2) und die Anzahl  $n$  der Punkte. Ist die Anzahl der Punkte kleiner als fünf, so wird die Berechnung abgebrochen und der Normalenstatus im Knoten auf ungültig gesetzt. Für eine Ebene im  $\mathbb{R}^3$  braucht es mindestens

drei Punkte, um diese zu beschreiben, der Wert von fünf hat den Zweck, instabile Normalen, die wenig Kontext in der Punktwolke haben, zu unterdrücken.

Sind genug Punkte vorhanden, so wird die Kovarianz der Punkte berechnet und eine Hauptkomponentenanalyse durchgeführt. Diese liefert die Eigenwerte nach Größe sortiert und die dazugehörigen Eigenvektoren. Der Eigenvektor zum kleinsten Eigenwert wird als Normale verwendet.

Von der Normale und dem Blickvektor auf den Oktalbaumknoten wird dann das Skalarprodukt gebildet. Ist das Ergebnis negativ, so wird das Vorzeichen der Normale invertiert. Damit werden alle berechneten Normalen in Richtung des Betrachters orientiert, was aufgrund der Eigenschaften der „Point Feature Histograms“ im Deskriptor wichtig ist. Im Anschluss wird die Normale zurückgegeben.

### 5.2.1 Instabile Normalen

Sind die einer Normalen zugrunde liegenden Punkte kollinear, so kann im dreidimensionalen Raum keine Ebene geschätzt werden. Ohne zweite Richtung kann die Normale noch um die Linie herum gedreht werden. Diese Eigenschaft kann auch in Punktwolken auftreten, insbesondere am Rand des Sichtbereichs und an Abrisskanten (siehe Abbildung 5.5) oder wenn die Tiefenauflösung deutlich unter der räumlichen Auflösung liegt.

Das Problem tritt vor allem in Abhängigkeit von kleinen Normalenskalen auf und kann dazu führen, dass aufgrund solcher Normalen die Entropie eines Knotens steigt, so dass er als lokales Maximum der Entropie detektiert wird. Mit steigender Größe der Normalenskala sinkt die Wahrscheinlichkeit, dass Normalen an solchen Randbereichen geschätzt werden.

### 5.2.2 Normalen bei großer Entfernung

Die Kinect diskretisiert die Tiefe der Punkte ausgehend von den Werten, die aus dem Bild der Infrarotkamera berechnet werden. Alle Punkte derselben Diskretisierungsstufe bekommen die gleiche z-Koordinate. Dies führt zu Ebenen in der Tiefe, die mit steigender Entfernung von der Kinect weiter auseinander liegen. Ist die Normalenskala kleiner als der Abstand zwischen den Ebenen, so werden die Normalen auf diesen Ebenen geschätzt, wie in Abbildung 5.6 zu sehen. Einerseits sind damit alle weiteren Berechnungen überflüssig, da aufgrund dieser Normalenschätzungen keine aussagekräftigen Ergebnisse erzielt werden. Andererseits führt diese Eigenschaft nicht zu Problemen im Ablauf, da die betroffenen Normalen nahezu alle parallel zur Tiefenachse sind. Die Entropie in Regionen mit derartigen Normalen ist niedrig und die damit Anzahl der Fehlmessungen. Es kann aber in dem Bereich der Punktwolke, in dem die Entfernung zwischen den Tiefenstufen in der Punktwolke gleich der Normalenskala ist, zu fehlerhaften Detektionen kommen.



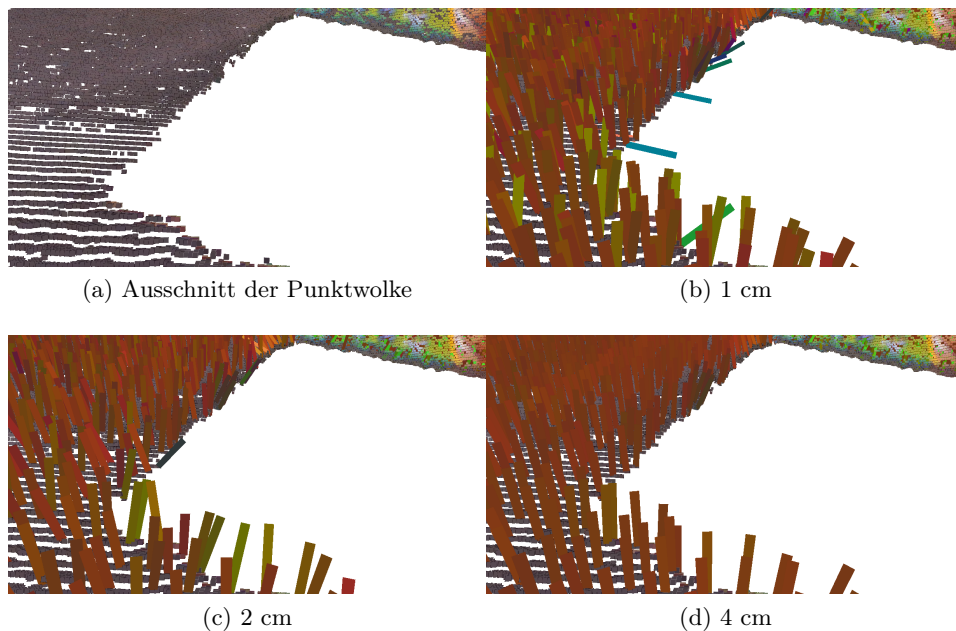


Abbildung 5.5: Instabile Normalen an Kanten der Punktwolke bei kleinen Normalenskalen.

### 5.3 Entropieberechnung

Nachdem der Oktalbaum aufgebaut ist, kann die Berechnung der Entropie in Regionen der Punktwolke erfolgen. Dazu werden dem Detektor die gewünschten Parameter zu Skalen und Abtastraten übergeben und die Berechnung gestartet.

Der Detektor durchläuft nur die Tiefenliste der Knoten, deren Kantenlänge der Abtastrate entspricht. Für jeden dieser Knoten startet er eine Bereichsanfrage, deren Größe der Histogrammskala entspricht. In dieser werden alle Knoten eingesammelt, deren Kantenlänge der Normalenabtastrate entspricht.

Dabei wird in jedem dieser Knoten geprüft, ob eine Normale schon berechnet worden ist. Es werden folgende drei Fälle unterschieden:

- Es ist keine Normale geschätzt worden. Die Normalenschätzung wird gestartet und deren Ergebnis wird als Status der Normale in den Knoten eingetragen. Ist eine Normale geschätzt worden, so wird deren Diskretisierung ins Entropiehistogramm addiert.
- Es existiert bereits eine Normale. Die entsprechende Diskretisierung

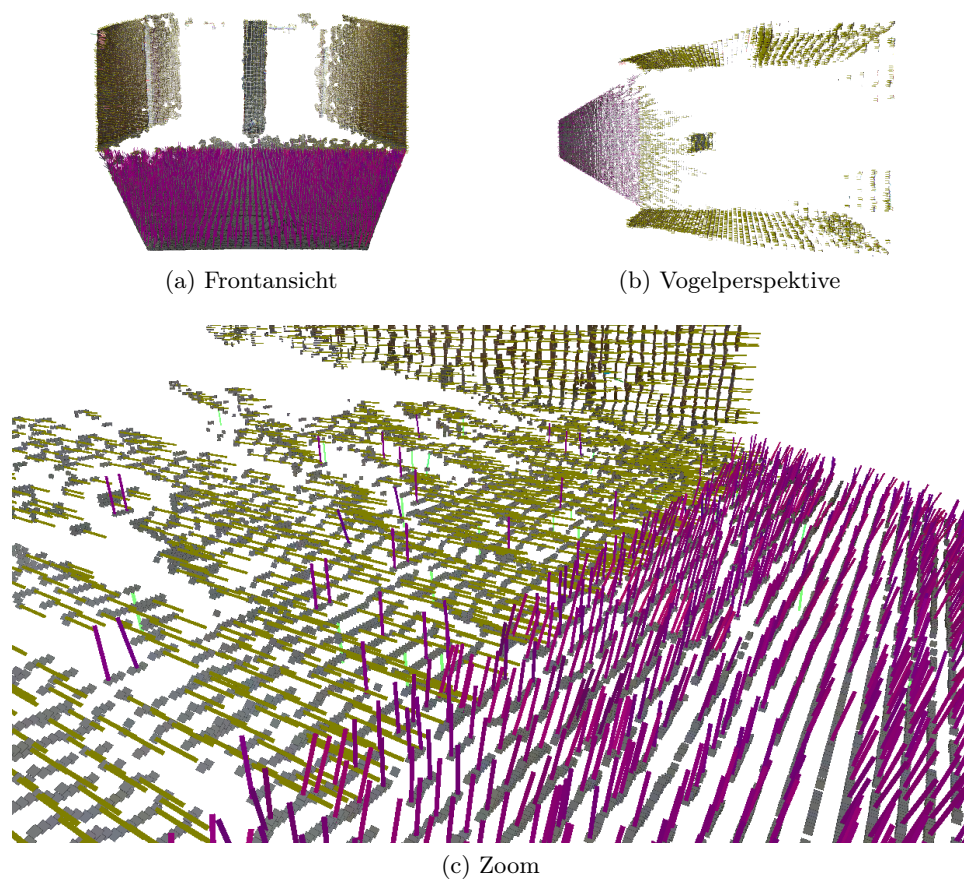


Abbildung 5.6: Die ebenenweise Diskretisierung der Punktwolke in der Tiefe erfordert die Anpassung der Normalenskala zur Schätzung korrekter Normalen.

wird auf das Entropiehistogramm des Ausgangsknotens addiert.

- Der Status der Normale ist „ungültig“. Dies ist der Fall, wenn sich zuwenig Punkte im Volumen entsprechend der Normalenbereichsgröße befinden. Der Knoten wird übergangen.

Sind alle Knoten aus der Bereichsanfrage bearbeitet worden, so wird das Entropiehistogramm entsprechend der Anzahl der eingetragenen Normalen normalisiert und die Entropie auf diesem berechnet und gespeichert. Die wichtigsten Schritte der Entropieberechnung sind in Abbildung 5.7 zu sehen. Die aufwändigsten Teile der Berechnung sind die Bereichsanfragen sowie die Entropieberechnung.

## 5.4 Extraktion von Merkmalen

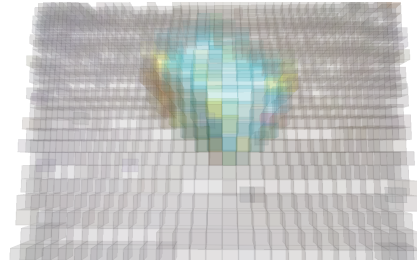
Die Merkmalsextraktion sucht nach lokalen Maxima in der Entropie, für diese wird dann der Deskriptor erstellt, Position und Deskriptor ergeben das Merkmal. Für die Extraktion gibt es Parameter, die für das Ergebnis relevant sind:

- Der Entropieschwellwert. Dieser ist der Grenzwert, ab dem ein Knoten als Maximum in Betracht gezogen wird. In Knoten mit einer Entropie unterhalb des Schwellwertes wird der Status entsprechend geändert, dann werden sie übergangen und auch in den weiteren Schritten ignoriert. Das dient dazu, schwache lokale Maxima wie Fehlmessungen oder Unregelmäßigkeiten in ansonsten uninteressanten Regionen auszuschließen. Ist der Wert sehr niedrig, so werden viele Merkmale extrahiert; darunter auch solche, die kaum Chancen haben wiedererkannt zu werden. Ist er zu hoch eingestellt, so werden eventuell gute Maxima verworfen.
- Die Eckigkeit. Die Eckigkeit dient zum Ausschluss von Entropiemaxima an Kanten und wird in Kapitel 5.5 beschrieben. Der Wert legt fest, welche Eckigkeit ein Knoten mindestens haben muss, um als Maximum in Betracht gezogen zu werden.
- Der Dominanzbereich: Dieser bestimmt die Größe des Bereichs, in der ein Maximum dominant sein muss, dass heißt es darf innerhalb des Bereichs kein weiterer Knoten eine gleiche oder höhere Entropie aufweisen. Der Wert ist üblicherweise gleich der Histogrammskala. Größere Werte senken die Anzahl der gefundenen Merkmale. Dieser Wert ist während der Entstehung der Arbeit nützlich gewesen, um zu prüfen, welche Maxima der Entropie die stärksten waren. Zum Beispiel ließ sich überprüfen, dass Merkmale an den Ecken eines Würfels Merkmale an Kanten dominieren. Der Parameter ist nur von Bedeutung, wenn man die Anzahl der Merkmale künstlich beschränken will.

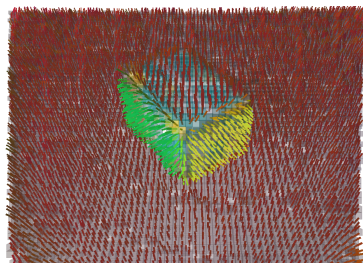




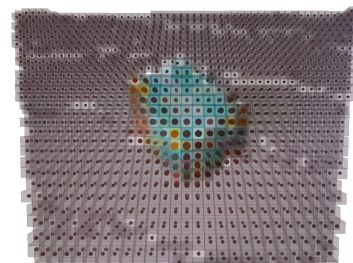
(a) Punktwolke



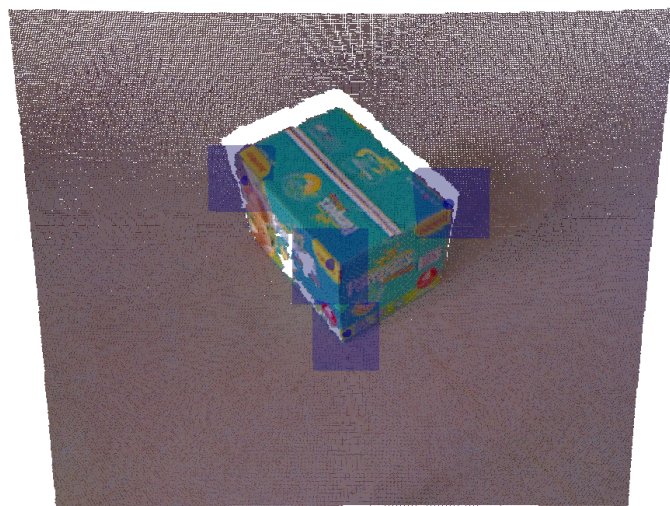
(b) Oktalbaum der Punktwolke



(c) Schätzung der Normalen



(d) Berechnung der Entropie



(e) Extraktion der Merkmale

Abbildung 5.7: Ablauf der Detektion

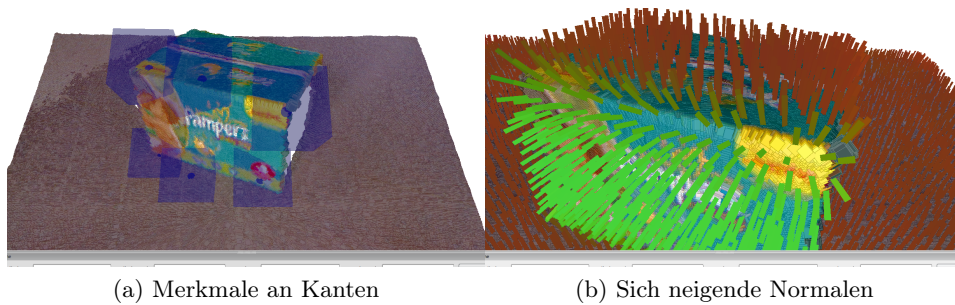


Abbildung 5.8: Merkmale an Kanten und sich neigende Normalen.

Der Detektor durchläuft nun die Tiefenliste der Knoten, deren Kantenlänge der Abtastrate entspricht, und die alle einen Entropiewert gespeichert haben. Jeder Knoten muss drei Tests durchlaufen:

1. Die Entropie wird mit der Mindestentropie verglichen. Liegt die Entropie darunter, so wird der Knoten entsprechend markiert und übergangen.
2. Es wird eine Bereichsanfrage in Größe des Dominanzbereiches gestellt, in der alle Knoten mit Kantenlängen entsprechend der Abtastrate eingesammelt werden. Mit diesen Knoten wird zunächst die Eckigkeit des Knotens berechnet. Liegt diese unter dem Schwellwert, so wird der Knoten entsprechend markiert und übergangen.
3. Die Entropie der gültigen Knoten der Bereichsanfrage aus Punkt 2 werden mit der Entropie des aktuellen Knotens verglichen. Findet sich eine höhere Entropie, so wird der Knoten markiert und übergangen.

Falls ein Knoten alle diese Tests besteht, so wird er als Maximum markiert und in die Liste der zukünftigen Merkmale eingetragen. Nach Durchlauf der kompletten Abtasttiefe werden aus den Knoten in der Liste die Merkmale erstellt. Die aufwändigsten Schritte sind bei der Extraktion sowohl die Bereichsanfragen im Oktalbaum als auch die Berechnung der Eckigkeit.

## 5.5 Eckigkeit

Die Entropie an Kanten ist höher als in Flächen, was auch an der Art und Weise der Normalenschätzung liegt. Wird eine Normale in der Nähe einer Kante geschätzt, so fließen beide Seiten in die Schätzung mit ein und die Normale neigt sich je nach Einfluss der Seiten, siehe Abbildung 5.8. Die führt zu einer hohen Entropie, da dadurch verschiedene Normalen im Bereich der Kante geschätzt werden.

Das Problem von Merkmalen an Kanten ist, dass sie nicht entlang der Kante lokalisiert sind und bereits durch geringes Sensorrauschen ihre Position ändern. Sollten nun auch die Deskriptoren dieser verschobenen Merkmale ähnlich genug sein, dass diese einander zugeordnet werden, so würde sich das negativ auf eventuelle Anwendungen auswirken, die eine eindeutige Positionsbestimmung der Merkmale benötigen.

Eine Möglichkeit zur Minderung des Problems ist, Merkmale an Kanten aufgrund ihrer geometrischen Eigenschaften herauszufiltern. Die Idee dazu kam aus den Eigenschaften der Eigenwerte der Normalenschätzung, genauer gesagt aus der Frage, wie das Verhältnis von kleinstem zu größtem Eigenwert ist. An Kanten und in Ebenen ist das Verhältnis zwischen den Eigenwerten um ein Vielfaches höher als an Ecken. Daraus entstand der Begriff der Eckigkeit eines Merkmals, der an den englischen Begriff der „Cornersness“ angelehnt ist, der in Publikationen im Bereich der Merkmalerkennung in Bilddaten verbreitet ist, siehe [MS04], Seite 66.

Unter Berücksichtigung der Entropie im Umkreis eines Merkmals lässt sich daraus mit Hilfe der bedingten Kovarianz folgende Berechnung festlegen: Gegeben seien die Menge  $K = k_1, k_2, \dots, k_n$  von  $n$  Knoten im Histogrammbereich um den Referenzknoten  $r$ .  $H(k)$  bezeichnet die für den Knoten  $k$  berechnete Entropie. Dann wird zunächst der nach Entropie gewichtete Mittelwert  $m_H$  der Koordinaten aus den Knoten in  $K$  berechnet:

$$m_H = \frac{1}{\sum_{k_i \in K} H(k_i)} \begin{pmatrix} \sum_{k_i \in K} x_{k_i} H(k_i) \\ \sum_{k_i \in K} y_{k_i} H(k_i) \\ \sum_{k_i \in K} z_{k_i} H(k_i) \end{pmatrix} \quad (5.1)$$

Für alle Knoten  $k_i \in K$  wird dann die Abweichung  $d_{k_i}$  vom Mittelwert  $m_H$  berechnet:

$$d_{k_i} = \begin{pmatrix} x_{m_H} - x_{k_i} \\ y_{m_H} - y_{k_i} \\ z_{m_H} - z_{k_i} \end{pmatrix} \quad (5.2)$$

Die Abweichung eines Knotes gewichtet nach seiner Entropie wird dann in der Kovarianzmatrix über alle Knoten aufsummiert:

$$Cov_E = \frac{1}{\sum_{k_i \in K} H(k_i)} \sum_{k_i \in K} H(k_i) d_{k_i} d_{k_i}^T \quad (5.3)$$

Es werden dann die Eigenwerte  $\lambda_1$ ,  $\lambda_2$  und  $\lambda_3$  der Kovarianzmatrix  $Cov_E$  berechnet, wobei  $\lambda_3$  der größte und  $\lambda_1$  der kleinste Eigenwert ist. Die Eckigkeit  $E$  des Referenzknotens  $r$  wird wie folgt definiert:

$$E(r) = \frac{\lambda_1}{\lambda_3} \quad (5.4)$$

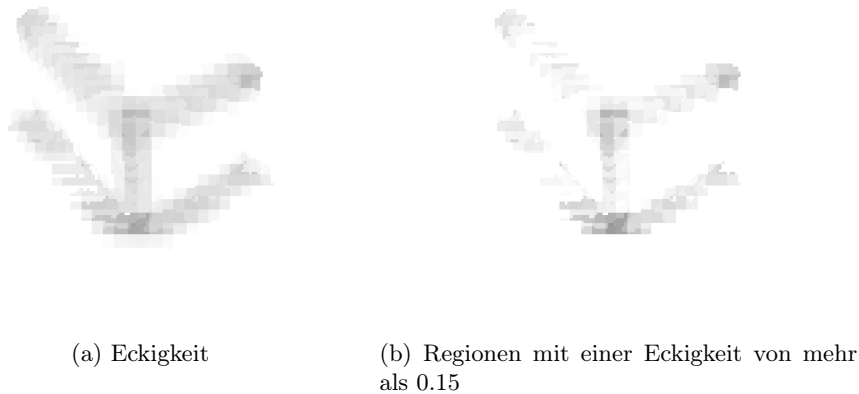


Abbildung 5.9: Eckigkeit auf einer Punktwolke eines Kartons. Dunkle Punkte markieren eine höhere Eckigkeit

In Abbildung 5.9 sieht man die Eckigkeit visualisiert. Sie weist im Verhalten Ähnlichkeiten zur Entropie auf. Die Berechnung ist wesentlich aufwändiger, daher wird sie nur als Ausschlusskriterium in der Merkmalsextraktion eingesetzt.

## 5.6 Lokalisierung der Detektionen

Die Position einer Detektion stimmt teilweise nicht mit der erwarteten Position überein. Die Gründe hierfür liegen zum einen beim Oktalbaum und der Diskretisierung der Punktwolke, mit folgenden Auswirkungen:

- Wird ein Oktalbaumknoten als lokales Maximum der Entropie identifiziert, wird das Merkmal an dem Punkt der Punktwolke erstellt, der am nächsten zum Zentrum des Oktalbaumknotens liegt. Je nach Lage des Knotens zu Punktwolke kann dies eine Abweichung in Größenordnung der Abtastrate zur Folge haben.
- Die Bereichsanfragen verwerfen Knoten, die nicht komplett im Bereich der Anfrage liegen. Würden Knoten am Rand einer Bereichsanfrage mit in diese einbezogen, würde dies auch dieselben Probleme hervorrufen.
- Der Ausgangspunkt für alle Bereichsanfragen von einem Knoten aus ist der nächste Punkt der Punktwolke zum Zentrum des Knotens. Dies kann an den Rändern der Bereichsanfrage zum Verwerfen von Knoten führen, die nicht vollständig im Bereich liegen. Wählt man die Alternative, die Anfragen vom Zentrum des Knotens aus zu stellen, kommt es vor, dass je nach Lage der Punktwolke zum Knoten die Bereichsanfrage Bereiche sehr stark abdeckt, in denen sich keine Punkte aus der

Punktwolke befinden. Die kann dazu führen dass die Entropieberechnung bei benachbarten Knoten in Bereichen mit unterschiedlich gutem Kontext aus der Punktwolke stattfindet. Dies kann auch zu schlecht lokalisierten Merkmalen führen. Die erste Variante ist gewählt worden, weil die Bereichsanfragen damit näher an den tatsächlichen Daten aus der Punktwolke bleiben und weniger von der Diskretisierung des Oktalbaums abhängen.

Weitere, wenn auch geringfügigere Ursachen für das Problem sind:

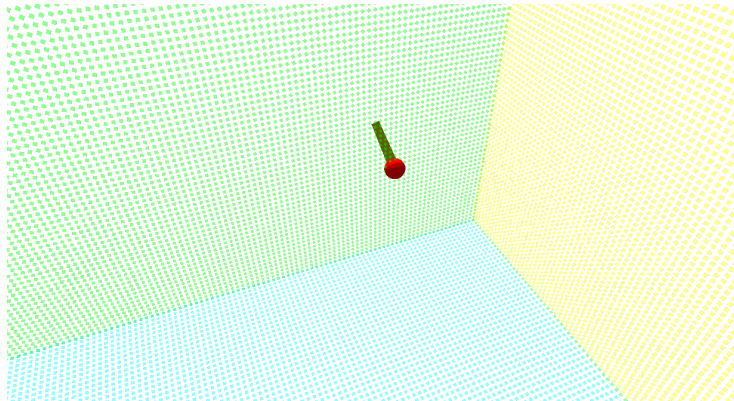
- Die Diskretisierung im Normalenhistogramm. Da die Diskretisierung nur eine Approximation ist, treten auch hier Fehler auf, die sich auf die Entropie auswirken.
- Ungenauigkeiten des Sensors führen zu verrauschten Punktwolken, diese wiederum zu ungenauen Schätzungen der Normalen und damit zu Ungenauigkeiten in der Entropieberechnung. Dies lässt sich allerdings nicht vermeiden, daher müssen Detektor und Deskriptor entsprechend robust gegenüber diesen Einflüssen sein, zum Beispiel durch die Wahl genügend großer Skalen.
- Ungenauigkeiten in der Rechnung mit Fließkommazahlen. Auch diese lassen sich nicht vermeiden, müssen also einkalkuliert werden.

In der Summe führt dies zu Ungenauigkeiten, die einer Korrektur bedürfen, da ohne Korrektur die Merkmalsposition von gleichen Merkmalen so stark schwankt, dass eine erfolgreiche Zuordnung der Deskriptoren nicht mehr möglich ist. Folgende Lösungsansätze wurden umgesetzt:

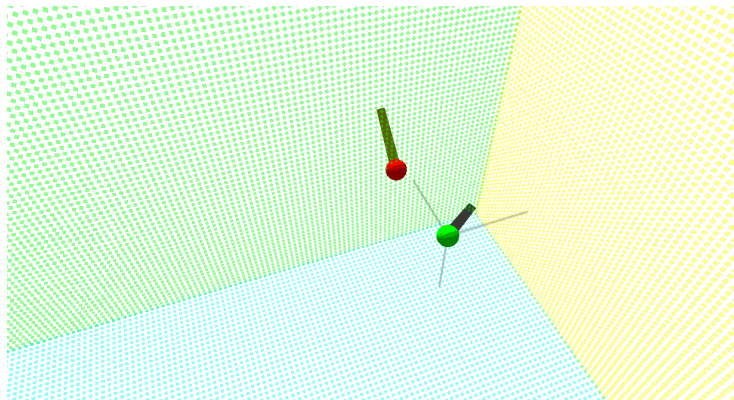
### 5.6.1 Subsampling

Jede Detektion wird im Umkreis um die gefundene Position mit höherer Auflösung wiederholt. Das heißt, in einem Bereich in Größe der Histogrammskala um die Detektion herum werden Normalen- und Entropieberechnung mit gleicher Histogramm- und Normalenskala, aber geringerer Abtastrate und Normalenabtastrate durchgeführt. Das lokale Maximum der hierbei ermittelten Entropie wird als verbesserte Detektion verwendet. Dies lässt sich je nach Auflösung des Baumes und verwendeten Abtastraten mehrfach mit kleiner werdenden Abtastraten wiederholen. Problematisch ist die Tatsache, dass die Punktwolke ungleichmäßig aufgelöst ist und daher ein höheres Auflösen nicht immer möglich beziehungsweise sinnvoll ist. Der Versuch der besseren Lokalisierung kann dann auch schlechtere Ergebnisse liefern. Weiter problematisch ist der Rechenzeitbedarf, der je nach Anzahl der Detektionen und verwendeten Parametern sehr viel länger sein kann als die Berechnung der Detektionen selbst. Vorteilhaft ist, dass die neu berechneten Positionen immer Bezug zur Punktwolke haben, siehe Abbildung 5.10(c).

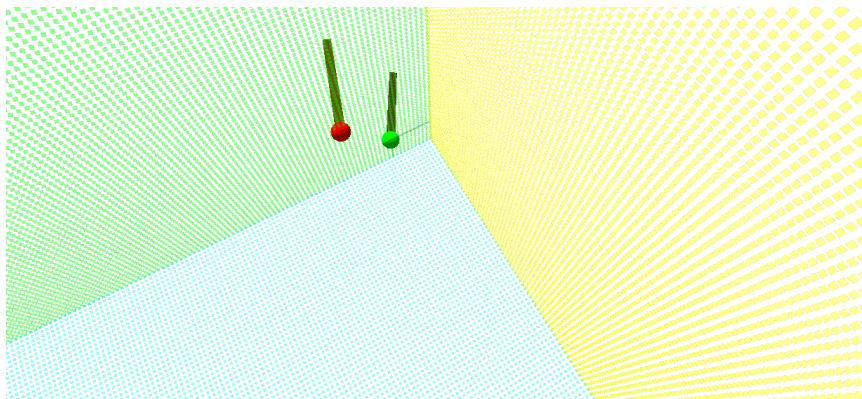




(a) Schlecht lokalisiertes Merkmal (rot) an einer Ecke eines Würfels



(b) Verbesserte Lokalisation mittels Mean-Shift (grün)



(c) Verbesserte Lokalisation mittels Subsampling (grün)

Abbildung 5.10: Lokalisation der Merkmale

### 5.6.2 Mean-Shift

Die Idee hierbei ist die Interpretation der Entropie auf der Punktwolke als Dichtefunktion. Diese lässt sich nicht in geschlossener Form angeben, sondern nur mit den Detektionen als Stichproben schätzen. Der Mean-Shift Algorithmus berechnet aus diesen Stichproben einen Gradientensanstieg zum lokalen Maximum der Dichtefunktion im dreidimensionalen Raum, was den lokalen Maxima der Entropie entspricht. Als initiale Position für die Schätzung dient die Detektion, die Größe des Fensters um die initiale Position wird durch die Histogrammskala gegeben. Innerhalb des Fensters werden der Mittelwert der Entropie und die Standardabweichung gebildet. Die Abweichung der Position jedes im Fenster liegenden Knotens vom Mittelwert Fensters wird nun - mit der Abweichung gewichtet - aufsummiert. Das Resultat ist eine Position, die entlang des stärksten Gradienten der Dichtefunktion verschoben ist. Diese Position ersetzt nun die initiale Position und der Algorithmus wird mit der neuen Position wiederholt. Die Zahl der Wiederholungen ist fest auf drei begrenzt, da bei Tests weitere Wiederholungen nicht oder nur geringfügig verschobene Positionen ergeben haben. Die Lokalisation mittels Mean-Shift führt dazu, dass die Merkmale in konvexe Strukturen, zum Beispiel Ecken wie in Abbildung 5.10(b), hinein verschoben werden. Dies wirkt sich aber nicht nachteilig aus, da die Verschiebung nur innerhalb der Histogrammskala erfolgen kann und damit der Kontakt zu Punktwolke nicht verloren geht. Zudem erfolgt diese Verschiebung unabhängig davon, wie gut oder schlecht das Merkmal vor Anwendung des Algorithmus lokalisiert war anhand der Verteilung der Entropie in der Histogrammskala. Dies führt zu ähnlichen Positionen aller Merkmale im Bezug zur Punktwolke, was sich positiv auf die Wiedererkennbarkeit auswirkt.





## Kapitel 6

# Beschreibung interessanter Regionen

Dieses Kapitel dient zur Beschreibung des Deskriptors. Der Deskriptor besteht aus drei getrennten Deskriptoren, einer beschreibt die Form des Merkmals, eine weitere die Farbe, der Dritte dient zur Beschreibung von Helligkeitsverläufen in detektierten Regionen. Erläutert werden Motivation, Struktur und Art und Weise der Distanzberechnung zwischen den Deskriptoren.

### 6.1 Formdeskriptor

#### 6.1.1 Motivation

Die Überlegung hinter dem Formdeskriptor ist, dass die vom Detektor als interessant markierten Regionen eine hohe Anzahl unterschiedliche Oberflächenstruktur besitzen können. Es sollte daher möglich sein, verschiedene Regionen durch eine Beschreibung der vorkommenden Oberflächen zu unterscheiden. Eine Möglichkeit zur Beschreibung ist durch die „Point Feature Histograms“ (siehe Kapitel 3.8) schon erfolgreich vorgestellt worden. Es lag nahe, diese als Grundlage für die Beschreibung der Form zu nutzen. Weiterhin sollte der Deskriptor invariant gegenüber Blickwinkeländerungen sein, was entweder eine Orientierung des Deskriptors im Raum erforderlich gemacht hätte, oder der Deskriptor muss so angelegt werden, dass die Elemente des Deskriptors invariant gegenüber Blickwinkeländerungen sind. Eine Orientierung eines Merkmals im dreidimensionalen Raum erfordert die Wahl von zwei orthogonalen Achsen. Die erste ist die „Hauptnormale“ des Merkmals, diese wird mittels aller in der Histogrammskala liegenden Punkte geschätzt. Die Wahl einer zweiten Achse, die gegenüber Änderungen des Blickwinkels invariant ist, stellte sich als schwierig heraus und wurde zugunsten eines Deskriptors verworfen, der gegenüber Rotationen um die Hauptnormale invariant ist.

### 6.1.2 Aufbau

Der Deskriptor besteht aus mehreren Histogrammen, in denen drei Winkel zwischen den in der Histogrammskala vorkommenden Normalen und der Hauptnormale des Merkmals eingetragen werden. Die Erstellung erfolgt ähnlich der von „Fast Point Feature Histogrammen“, es gibt aber einige Unterschiede, die im Folgenden erklärt werden. Gegeben sei ein Merkmal  $M$  mit Position  $p_M$  und Hauptnormale  $n_M$ . Weiterhin sei  $t$  die Anzahl der Knoten mit gültiger Normale, die in die Entropieberechnung für  $M$  mit eingeflossen sind, diese liegen innerhalb eines Bereiches der Größe der Histogrammskala um  $p_M$ . Sei  $S = \{(n_1, p_1), (n_2, p_2), \dots, (n_t, p_t)\}$  die dazugehörige Menge an Surflets. In den PFHs wird für die Berechnung der Winkel zwischen zwei Normalen ein Koordinatensystem definiert, dessen Ursprung wird anhand des Winkels gewählt, den die Normalen gegenüber der Strecke zwischen den Normalen haben. In unserem Ansatz ist der Ursprung durch die Merkmalsposition  $p_M$  gegeben. Für ein Surflet  $(n_i, p_i) \in S$  sieht das Koordinatensystem wie folgt aus:

$$u = n_M \quad (6.1)$$

$$v = \frac{(p_i - p_M) \times u}{\|(p_i - p_M) \times u\|} \quad (6.2)$$

$$w = u \times v \quad (6.3)$$

Die drei Winkel sind die folgenden:

$$\alpha = \arctan(w \circ n_i, u \circ n_i) \quad (6.4)$$

$$\beta = v \circ n_i \quad (6.5)$$

$$\gamma = u \circ \frac{p_i - p_M}{\|p_i - p_M\|} \quad (6.6)$$

Im Gegensatz zu den PFHs und FPFHs werden die Winkel nur zwischen den Normalen und der Hauptnormalen berechnet. Diese Winkel werden nun in ein Histogramm eingetragen. Für jeden Winkel gibt es hierbei zwei Histogramme mit jeweils elf Klassen:  $H_{\alpha,1}, H_{\alpha,2}, H_{\beta,1}, H_{\beta,2}, H_{\gamma,1}, H_{\gamma,2}$ . In das erste Histogramm werden die Winkel aller Surflets eingetragen, deren Distanz zur Merkmalsposition kleiner als die Hälfte der Histogrammskala ist, in das zweite alle anderen. Als Distanz wird die L1-Distanz verwendet, da der Bereich der Histogrammskala ein Kubus ist und die euklidische Distanz spherenförmige Bereich darin erzeugt hätte. Grund ist, dass die Histogrammskala ein Würfel ist, und damit auch der Bereich, dessen Normalen zur Erstellung des Formdeskriptors verwendet werden. Bereichsanfragen in Sphärenform im Oktalbaum müssen durch den die Sphäre umfassenden Würfel substituiert werden und die Entfernung zum Zentrum anschließend für jeden Knoten der Anfrage berechnet werden. Aus Effizienzgründen wird darauf verzichtet und stattdessen würfelförmige Bereiche mit L1-Distanz verwendet. Dies bedeutet

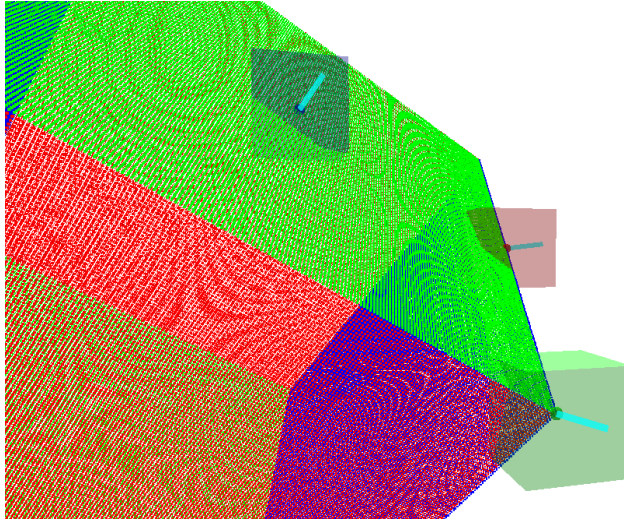


Abbildung 6.1: Drei manuell platzierte Merkmale an einer Ecke, einer Kante und auf der Ebene eines Würfels. Die Histogramme der Merkmale sind als Beispiele in den Abbildungen 6.2 bis 6.5 zu sehen.

auch, dass das Volumen des Bereichs für das zweite Histogramm siebenmal so groß ist wie der Bereich für das erste Histogramm. Insgesamt besteht der Formdeskriptor aus sechs Histogrammen mit zusammen 66 Klassen. Um einen Eindruck zu gewinnen, wie die Histogramme aussehen, wurden drei Merkmale an einer Ecke, einer Kante und auf einer Ebene erstellt (siehe Abbildung 6.1) und deren Deskriptoren in den Abbildungen 6.2, 6.3 und 6.4 dargestellt. Deutlich zu sehen ist, dass sich der Formdeskriptor des Merkmals auf der Ebene deutlich von den anderen beiden abhebt. Dagegen ist bei den Deskriptoren für das Kantenmerkmal und das Eckenmerkmal die Unterscheidung schwieriger, lediglich bei der  $\beta$ -Komponente in 6.3 fällt der Unterschied deutlich aus.

### 6.1.3 Distanzfunktion

Die Distanz zwischen zwei Formdeskriptoren ist das arithmetische Mittel aus der euklidischen Distanz zwischen den sechs Histogrammen der Deskriptoren

$$d_{Form}(D_1, D_2) = \frac{1}{6} \sum_{i=1}^6 d(H_i, I_i) \quad (6.7)$$

für  $H_i \in D_1$  und  $I_i \in D_2$ . Die euklidische Distanz wird auch bei den PFHs und FPFHs verwendet. Abbildung 6.6 zeigt einen Vergleich zwischen der  $\widehat{EMD}$  und der euklidischen Distanz.

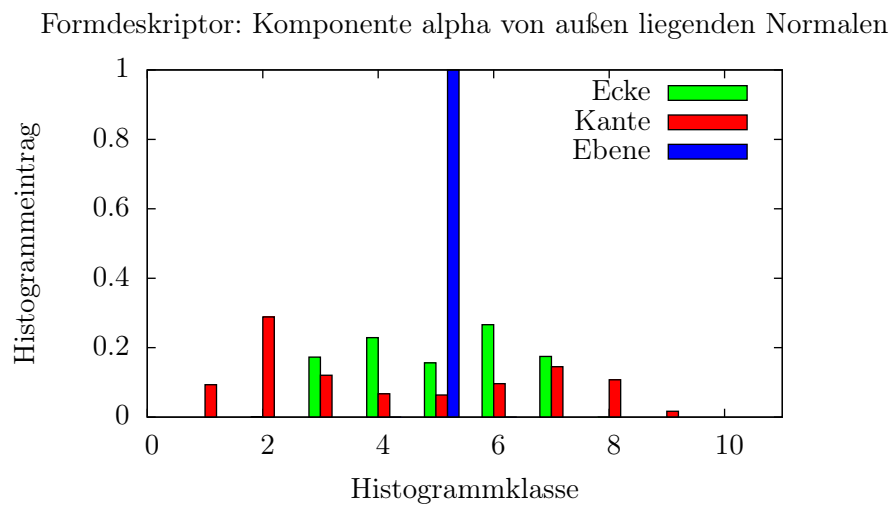
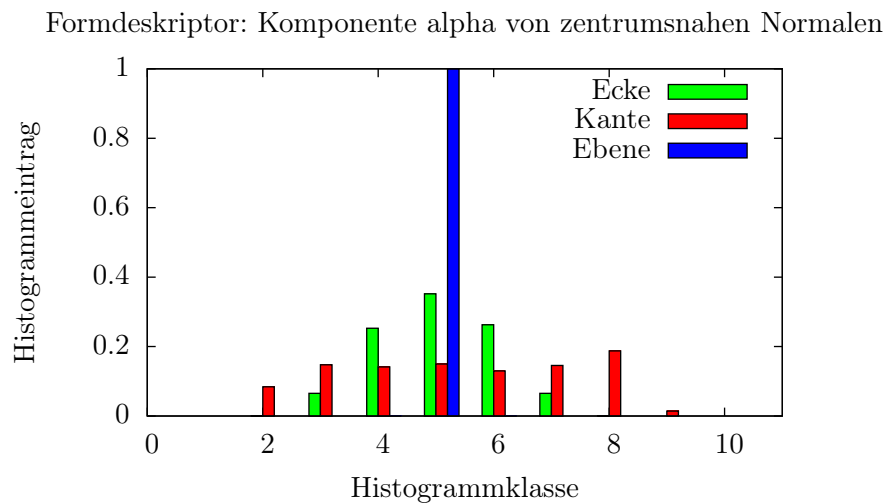


Abbildung 6.2: Die Histogramme der Komponente  $\alpha$  des Formdeskriptors für die drei Merkmale aus Abbildung 6.1. Im oberen Histogramm werden die Werte für Normalen nahe des Merkmalszentrums eingetragen, im unteren für die entfernt liegenden Normalen.

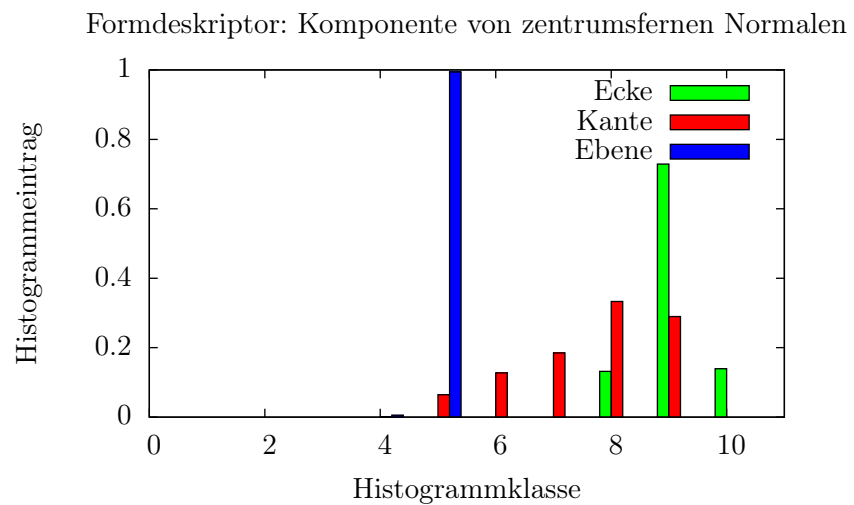
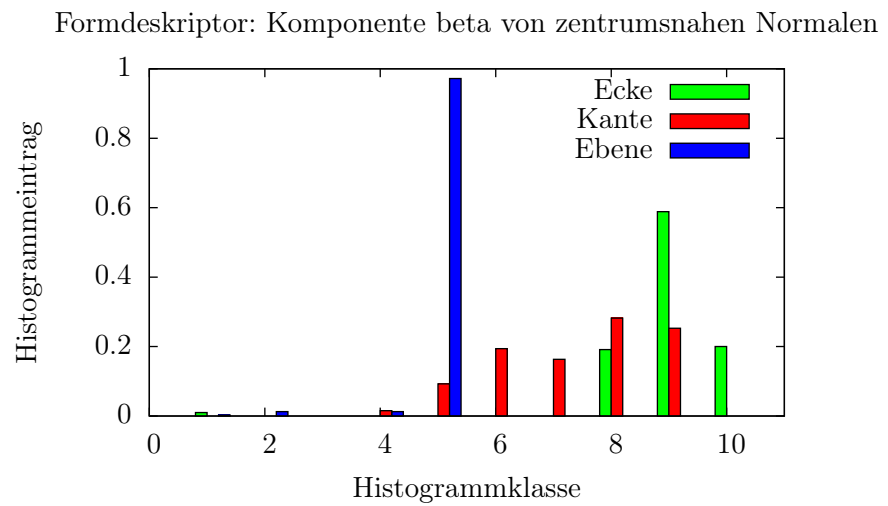
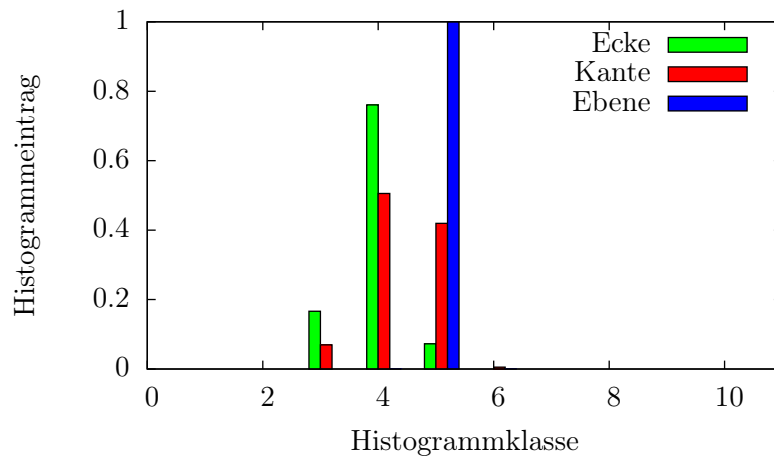


Abbildung 6.3: Die Histogramme der Komponente  $\beta$  des Formdeskriptors für die drei Merkmale aus Abbildung 6.1. Im oberen Histogramm werden die Werte für Normalen nahe des Merkmalszentrums eingetragen, im unteren für die entfernt liegenden Normalen.

Formdeskriptor: Komponente gamma von zentrumsnahen Normalen



Formdeskriptor: Komponente gamma von zentrumsfernen Normalen

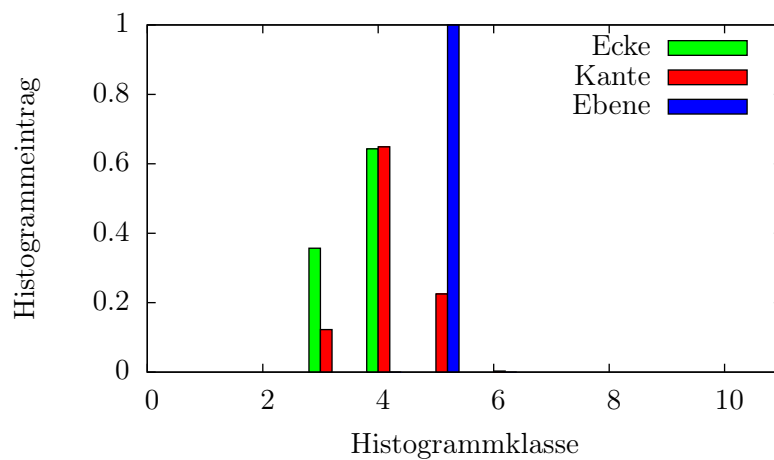


Abbildung 6.4: Die Histogramme der Komponente  $\gamma$  des Formdeskriptors für die drei Merkmale aus Abbildung 6.1. Im oberen Histogramm werden die Werte für Normalen nahe des Merkmalszentrums eingetragen, im unteren für die entfernt liegenden Normalen.

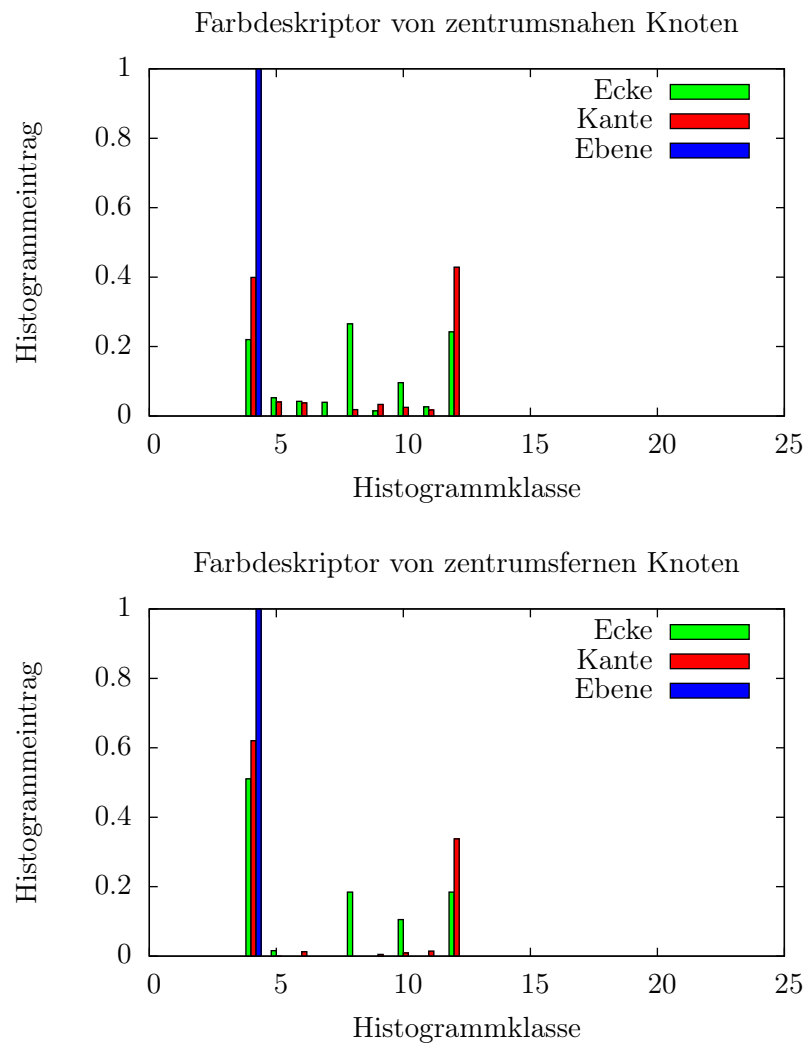
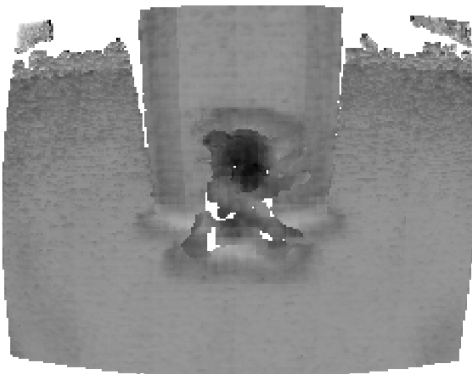
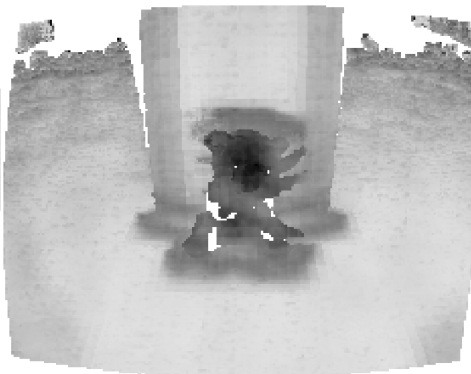


Abbildung 6.5: Die Histogramme der Farbdeskriptors für die drei Merkmale aus Abbildung 6.1. Im oberen Histogramm werden die Werte für Knoten nahe des Merkmalszentrums eingetragen, im unteren für die entfernt liegenden Knoten.



(a) Punktwolke mit Teddy

(b)  $\widehat{EMD}$ 

(c) euklidische Distanz

Abbildung 6.6: Vergleich der  $\widehat{EMD}$  und der euklidischen Distanz auf dem Formdeskriptor mittels Ähnlichkeitsbildern. Auf der Punktwolke in Abbildung (a) werden auf jedem Knoten der Abtastrate Merkmale erstellt und mit dem Merkmal an der Position des blauen Punktes verglichen, in diesem Fall wird nur der Helligkeitsdeskriptor verglichen. Abbildung (b) zeigt die Ähnlichkeit mittels  $\widehat{EMD}$ , Abbildung (c) die Ähnlichkeit bezüglich der euklidischen Distanz. Dunklere Punkte stehen für niedrige Distanzen, helle Punkte für sehr hohe bzw. unendlich, wenn keine gültigen Punkte vorliegen. Die  $\widehat{EMD}$  zeigt hier das schlechteren Ergebnis, vor allem die Distanzen von der Nase des Teddy zum Boden und zum Pfeiler sind bei der euklidischen Distanz sehr viel größer.



## 6.2 Farbdeskriptor

### 6.2.1 Motivation

Für die Implementierung eines Farbdeskriptors gibt es mehrere Gründe: Der Detektor reagiert auf sehr spezielle Regionen, daher war anzunehmen, dass diese Regionen untereinander auch in der Form ähnlich sein können, was zum Beispiel bei Ecken der Fall ist. Weitere Deskriptoren der Form betreffend hinzuzufügen wäre daher nicht unbedingt erfolgversprechend. Ein weiterer Grund, einen Farbdeskriptor zu benutzen, sind Ergebnisse von Farbdeskriptoren gegenüber intensitätsbasierten Deskriptoren, zum Beispiel aus [vdSGS10]. Vorteilhaft ist, dass die Kinect als primär verwendeter Sensor mit Farben registrierte Punktwolken erzeugt. Weiter wurde entschieden, dass ein anderer als der RGB-Farbraum verwendet werden sollte, da Farbähnlichkeiten im RGB-Farbraum schwierig zu bestimmen sind. Hinzu kommt, dass sich die Werte der Farben bei unterschiedlicher Beleuchtung ändern und dies bei einer wiederholbaren Beschreibung kontraproduktiv wäre.

### 6.2.2 Entwicklung und Aufbau

Die erste Idee war ein Deskriptor im YUV-Farbraum. Dieser diskretisiert auf der YUV-Farbebene die Distanz und die Richtung zwischen zwei Farbpunkten in einem Histogramm. Die Punkte sind einmal der YUV-Farbwert im Zentrum des Merkmals und die Punkte der Surflets, die auch für die Formdeskription verwendet werden. Problematisch ist die Abhängigkeit der UV-Ebene von der Luminanz und den damit verbundenen Einfluss auf die Distanz zwischen den Farben. Die ersten Ergebnisse waren nicht so distinktiv wie erhofft und der Ansatz wurde nicht weiter verfolgt.

Der nächste Ansatz war ein Deskriptor im HSL-Farbraum (Hue, Saturation, Lightness)). Der Wert von Lightness (Helligkeit) wird im Helligkeitsdeskriptor in Kapitel 6.3 benutzt und daher hier nicht weiter betrachtet. Der Farbwert  $H$  wird in einem Histogramm diskretisiert. Für jeden Punkt aus den Surflets  $S$  wird nun der Farbwert im HSL-Farbraum berechnet und an der zum Hue (Farbton) korrespondierenden Stelle wird die Saturation (Sättigung) eingetragen. So bietet das Histogramm zum einen eine Verteilung der Farbe im Merkmal, als auch einen Ansatzpunkt, wie gesättigt die Farben waren. Die Histogramme waren dann nicht normiert, was einerseits die Abschätzung eines Grenzwertes bei der maximalen Distanz zwischen den Histogrammen schwierig machte, zum anderen variierte die Höhe der Einträge auch mit der Anzahl der eingetragenen Punkte. Desweiteren wurde kaum zwischen unterschiedlich gesättigten Farben unterschieden. Da im HSL-Farbraum eine geringe Sättigung den Farbton grauer werden lässt, hatte das zur Folge, dass die Farbe Grau nicht gut von anderen Farben unterschieden werden konnte. Als Lösung wurde im Farbhistogramm eine weitere Klasse hinzugefügt, in die zu jedem Eintrag der Wert 1–Sättigung eingetragen wird. Damit

kann das Histogramm normiert werden und unterschiedlich stark gesättigte Farben werden berücksichtigt. Auch der Farbdeskriptor besteht wie der Formdeskriptor aus zwei Histogrammen, die je nach Entfernung zum Merkmalszentrum befüllt werden. Damit wird einmal der Farbverlauf im Zentrum des Merkmals als auch im äußeren Bereich des Merkmals dokumentiert. Abbildung 6.5 zeigt den Farbdeskriptor für die Merkmale aus Abbildung 6.1. Das auf der Ecke liegende Merkmal beinhaltet die Farben Rot, Grün und Blau, das Merkmal auf der Kante die Farben Grün und Blau und das Merkmal auf der Ebene nur die Farbe Grün. Alle Farben sind voll gesättigt. Das einfarbige Merkmal zeigt wieder den deutlichsten Unterschied, aber auch der Vergleich der beiden anderen zeigt größere Unterschiede als beim Formdeskriptor.

### 6.2.3 Distanzfunktion

Aufgrund der guten Ergebnisse im Bereich der Farbhistogramme (siehe [RTG00]) wurde die Variante  $\widehat{EMD}$  der „Earth Mover’s Distance“ aus [PW09] als Distanzfunktion für das Farbhistogramm gewählt. Die maximale Distanz zwischen zwei Klassen ist auf 2 begrenzt. Dadurch liefert die  $\widehat{EMD}$  eine geringe Distanz für ähnliche Histogramme und eine hohe Distanz für unähnliche Histogramme, unabhängig von tatsächlich beteiligten Farben.

Die Distanz für die Klasse, in der 1-Sättigung eingetragen ist, ist 2 zu allen anderen Klassen. Dies erhöht die Distanz zwischen Histogrammen, in denen gleiche, aber unterschiedlich gesättigte Farben eingetragen sind. Die Distanzmatrix für das Farbhistogramm ist in Abbildung 6.7 zu sehen, ein Beispiel für die Distanz zwischen unterschiedlichen Farben und unterschiedlichen Sättigungen ist in Abbildung 6.8 zu sehen. Da die Histogramme auf 1 normiert sind und die Maximaldistanz zwischen zwei Klassen 2 beträgt, ist die maximale Distanz zweier Histogramme 2. In der Abbildung sieht man, dass die Distanz zwischen Farben mit steigender Ähnlichkeit sinkt und für alle übrigen Farben gleich ist. Dies war die Intention der Verwendung der „Earth Mover’s Distance“. Außerdem zeigt die Abbildung, dass unterschiedlich gesättigte Farben ebenfalls unterschiedliche Distanzen erzeugen. In Abbildung 6.9 wird die  $\widehat{EMD}$  mit der euklidischen Distanz bei Anwendung auf dem Farbdeskriptor verglichen.

## 6.3 Helligkeitsdeskriptor

### 6.3.1 Motivation

Da der Farbdeskriptor Graustufen nur schlecht sowie Schwarz und Weiß nicht voneinander unterscheiden kann, wurde ein weiterer Deskriptor für die Helligkeit implementiert.

$$\begin{pmatrix}
 0 & 1 & 2 & \dots & 2 & 1 & 2 \\
 1 & 0 & 1 & 2 & \dots & 2 & \vdots \\
 2 & 1 & 0 & 1 & \ddots & \dots & \vdots \\
 \vdots & 2 & 1 & \ddots & \ddots & \ddots & \vdots \\
 \vdots & \cdot & \ddots & \ddots & \ddots & \ddots & 2 \\
 2 & \dots & \ddots & \ddots & \ddots & 1 & \vdots \\
 1 & 2 & \dots & 2 & 1 & 0 & 2 \\
 2 & \dots & \dots & \dots & 2 & 0
 \end{pmatrix}$$

Abbildung 6.7: Distanzmatrix für die Klassen des Farbhistogramms. Die letzte Zeile und Spalte markiert den Eintrag für 1-Sättigung

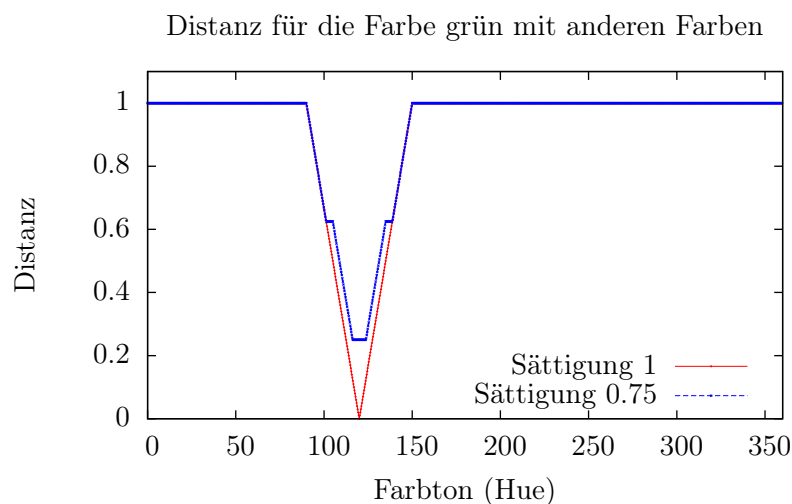
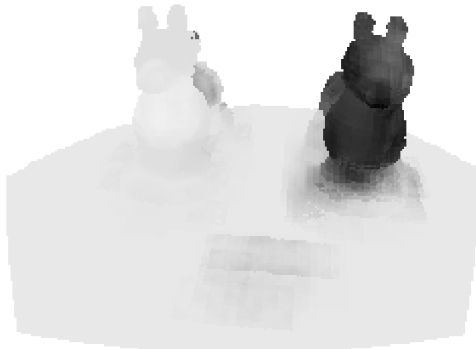
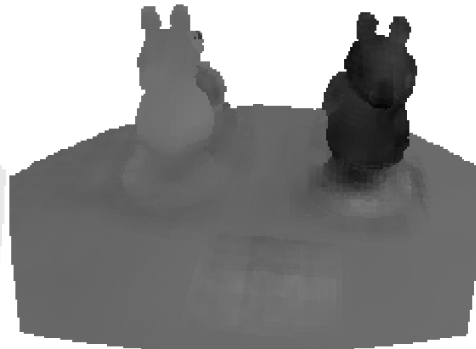


Abbildung 6.8: Die Abbildung zeigt die Distanz zwischen der Farbe grün (Farbton  $120^\circ$ , Helligkeit 1) und anderen Farben im Farbraum. Die Messung erfolgte einmal mit Sättigung 1 und einmal mit Sättigung 0.75 für die Farbe grün. Sättigung und Helligkeit der anderen Farben betragen 1, der Farbton wurde in Schritten von 0.3 inkrementiert. Zu sehen ist, dass unterschiedliche Sättigung auch bei gleichen Farben zu unterschiedlichen Distanzen führt.



(a) Punktwolke mit zwei Schaukelpferden

(b)  $\widehat{EMD}$ 

(c) euklidische Distanz

Abbildung 6.9: Vergleich der  $\widehat{EMD}$  und der euklidischen Distanz auf dem Farbdeskriptor mittels Ähnlichkeitsbildern. Auf der Punktwolke in Abbildung (a) werden auf jedem Knoten der Abtastrate Merkmale erstellt und mit dem Merkmal an der Position des blauen Punktes verglichen, in diesem Fall wird nur der Farbdeskriptor verglichen. Abbildung (b) zeigt die Ähnlichkeit mittels  $\widehat{EMD}$ , Abbildung (c) die Ähnlichkeit bezüglich der euklidischen Distanz. Dunklere Punkte stehen für niedrige Distanzen, helle Punkte für sehr hohe bzw. unendlich, wenn keine gültigen Punkte vorliegen. Beide Distanzen zeigen die Ähnlichkeit der Farbe des gelben Schaukelpferdes, die  $\widehat{EMD}$  erzeugt dabei die größeren Abstände. Das Checkerboard wird vom Farbdeskriptor allerdings ignoriert.

### 6.3.2 Aufbau

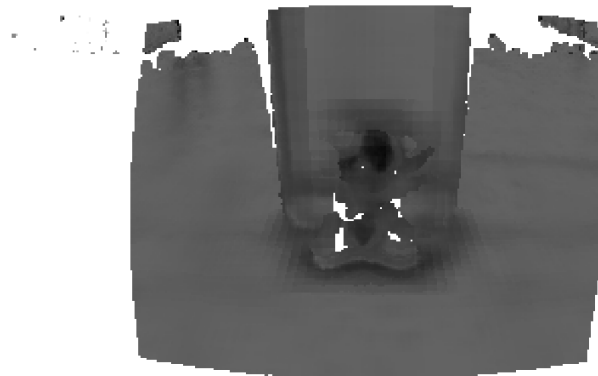
Der erste Helligkeitsdeskriptor wurde im YUV-Farbraum erstellt und ist ein Histogramm der Luminanz Y. Mit der Änderung des Farbraums zugunsten von HSL ist auch der Helligkeitsdeskriptor auf die Helligkeit L umgestellt worden. Es zeigt sich allerdings, dass die einfache Abbildung der Helligkeit in einem Histogramm nicht zur Distinktivität des Deskriptors beiträgt, da Merkmale desselben Objektes unterschiedliche Distanzen erzeugen, je nachdem wie die Beleuchtung des Objektes sich zwischen den Aufnahmen ändert. In der endgültigen Fassung wird die relative Helligkeit zum Zentrum des Merkmals gespeichert. Das Histogramm besteht aus zehn Klassen mit einem Wertebereich von  $[-1, 1]$ . Der Einfluss der Helligkeit einer Szene auf den Deskriptor ist damit gesenkt worden, allerdings kann es immer noch zu Diskrepanzen kommen, wenn Teile des Merkmals stärker ausgeleuchtet sind als andere. Das führt zu unterschiedlichen Deskriptionen, die aber nur von der aktuellen Helligkeit abhängen und nicht vom Aussehen des Objektes selber.

### 6.3.3 Distanzfunktion

Als Distanzfunktion wurde die modifizierte „Earth Mover’s Distance“ ( $\widehat{EMD}$ ) gewählt. Es wurde ein kurzer Vergleich zwischen euklidischer Distanz und der  $\widehat{EMD}$  durchgeführt, dessen Ergebnisse für die  $\widehat{EMD}$  sprechen. Dies ist in Abbildung 6.10 zu sehen.



(a) Punktwolke mit Teddy

(b)  $\widehat{EMD}$ 

(c) euklidische Distanz

Abbildung 6.10: Vergleich der  $\widehat{EMD}$  und der euklidischen Distanz auf dem Helligkeitsdeskriptor mittels Ähnlichkeitsbildern. Auf der Punktwolke in Abbildung (a) werden auf jedem Knoten der Abtastrate Merkmale erstellt und mit dem Merkmal an der Position des blauen Punktes verglichen, in diesem Fall wird nur der Helligkeitsdeskriptor verglichen. Abbildung (b) zeigt die Ähnlichkeit mittels  $\widehat{EMD}$ , Abbildung (c) die Ähnlichkeit bezüglich der euklidischen Distanz. Dunklere Punkte stehen für niedrige Distanzen, helle Punkte für sehr hohe bzw. unendlich, wenn keine gültigen Punkte vorliegen.

## Kapitel 7

# Experimente

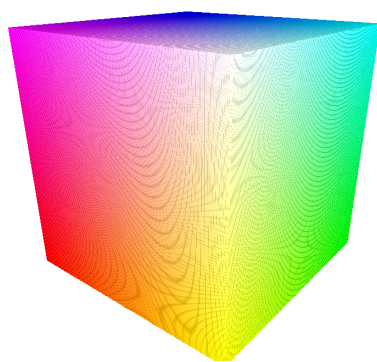
Dieses Kapitel widmet sich der Auswertung von Detektor und Deskriptor. Es werden verschiedene Experimente in Aufbau und Durchführung erklärt. Die Experimente dienen sowohl dem Finden guter Werte für die verschiedenen Parameter des Detektors als auch der Evaluation der Parameter zur Unterdrückung unerwünschter Merkmale. Dazu werden Experimente von Detektor und Deskriptor auf verschiedenen Objekten und Szenen erklärt und diskutiert.

### 7.1 Experimente mit künstlich erzeugten Daten

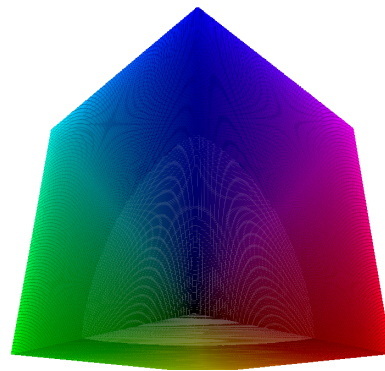
Tests mit künstlichen Punktwolken sind von Anfang an Bestandteil der Arbeit gewesen, da diese sowohl einfach zu handhaben als auch frei von externen Störfaktoren sind. Sie sind vor allem bei der Fehlersuche hilfreich gewesen, da Größe und Form der Punktwolke bekannt und somit Abweichungen der Ergebnisse von den Erwartungen sehr schnell aufgefallen sind. Mit Fortschreiten der Programmentwicklung sind die künstlichen Daten weiter variiert worden, um bestimmte Eigenschaften von auf realen Daten basierenden Punktwolken zu simulieren, wie zum Beispiel Sensorrauschen.

#### 7.1.1 Beschreibung des Datensatzes

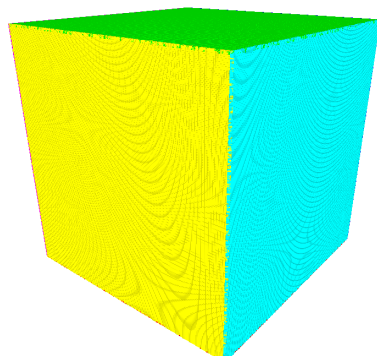
Die Punktwolke repräsentiert einen Kubus mit einem Meter Kantenlänge in 5mm Auflösung. Das Zentrum des Kubus liegt im Ursprung des Koordinatensystems, die Koordinaten der acht Ecken sind damit bekannt. Der Kubus kann beliebig um alle drei Koordinatenachsen gedreht und verschoben werden. Damit werden Nebeneffekte ausgeschlossen, die entstehen, wenn Seiten des Kubus mit Seiten von Oktalbaumknoten parallel oder identisch sind. Für Tests des Deskriptors wird der Würfel eingefärbt, und zwar entweder als RGB-Farbwürfel (siehe Abbildung 7.1b) oder seitenweise mit den sechs



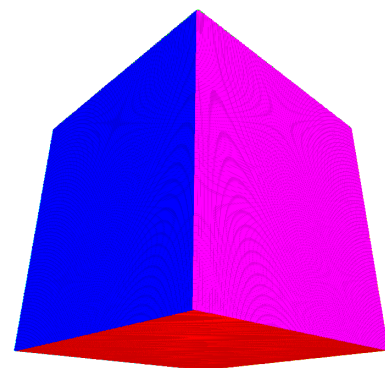
(a) RGB-Würfel (Frontseite)



(b) RGB-Würfel (Rückseite)



(c) Sechsfarbiger Würfel (Frontseite)



(d) Sechsfarbiger Würfel (Rückseite)

Abbildung 7.1: Zwei der Würfel, die für Experimente genutzt wurden. Die Würfel sind nicht texturiert, die Aliasingartefakte stammen aus der Visualisierung



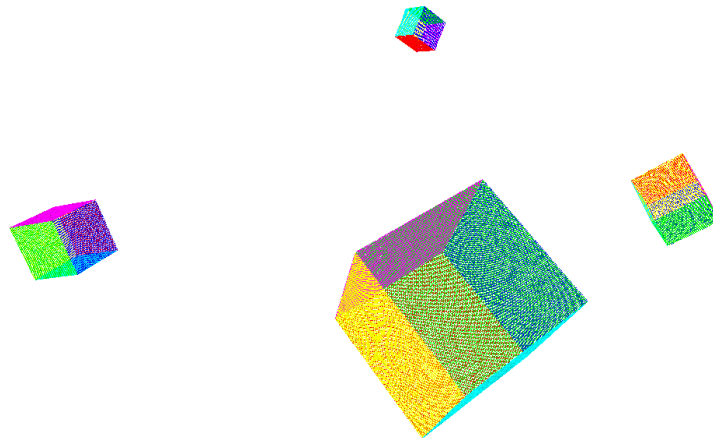


Abbildung 7.2: Vier unterschiedlich affin-transformierte Würfel

Farben Rot, Grün, Blau, Gelb, Violett und Cyan (siehe Abbildung 7.1d).

### 7.1.2 Vorgehensweise

Die Experimente laufen wie folgt ab: Der Detektor und der Deskriptor werden konfiguriert und die Punktwolke wird generiert. Die Punktwolke wird ausgehend vom Ursprung um alle drei Koordinatenachsen gedreht und danach in alle drei Koordinatenrichtungen verschoben (siehe Abbildung 7.2). Die Drehung und die Verschiebung werden durch gleichverteilte Zufallszahlen ermittelt. Der Zufallszahlengenerator wird vor der Generierung der Transformationen mit einem festen Anfangswert initialisiert. Detektor und Deskriptor werden auf der Punktwolke angewendet, die Ergebnisse werden gespeichert. Um die räumlichen Entfernungen zwischen Merkmalen zu messen, die derselben Ecke des Würfels zugeordnet werden, werden die Koordinaten der Merkmale zurück ins ursprüngliche Koordinatensystem transformiert. Um eine ausreichende Datenbasis für die Auswertung herzustellen, werden die Transformation und Berechnung der Ergebnisse über 250 Iterationen durchgeführt. Es wird sowohl die Wiederholpräzision des Detektors evaluiert als auch die Distanzen der Farb- und Formdeskriptoren. Mit Wiederholpräzision ist die Entfernung der Merkmale zueinander gemeint, die zu derselben Ecke des Würfels gehören. Da die Punktwolke zufällig transformiert wird, ändert sich die Struktur des erzeugten Oktalbaums. Dies führt zu unterschiedlich guten Lokalisationen der Merkmale (siehe Abbildung 7.3). Diese sollten aber so nah wie möglich zusammenliegen, damit die Deskriptoren der Merkmale möglichst gleich sind. Liegen sie zu weit auseinander, so steigt die Distanz zwischen den Deskriptoren eventuell derart, dass die Merkmale als unter-

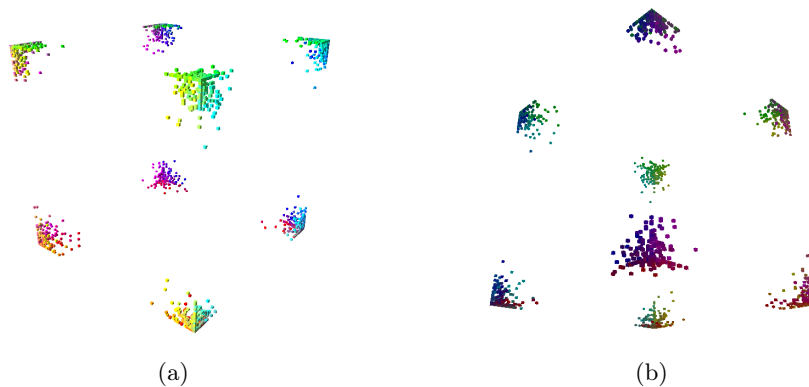


Abbildung 7.3: Die Positionen der gefundenen Merkmale für die Ecken des sechsfarbenen Würfels in 250 Iterationen, gesehen von zwei Seiten. Die Histogrammskala für die Aufnahmen betrug 48 cm, damit die Entfernungen zwischen den Positionen deutlicher zu sehen sind.

schiedlich deklariert werden, obwohl sie dasselbe Objekt beschreiben.

Da die Merkmale alle auf Ecken liegen, sollte der Formdeskriptor unabhängig von der jeweiligen Ecke gleiche Ergebnisse erzeugen. Dagegen ermöglicht die unterschiedliche Farbgebung des Würfels eine differenzierte Auswertung: Während die Distanzen zwischen den Farbdeskriptoren bei gleichen Ecken auch gleich sein sollten, so sollten sie sich bei anderen Ecken unterscheiden, und das unterschiedlich stark. Zu jeder Referenzecke gibt es drei Ecken, die zwei Farben mit der Referenzecke gemeinsam haben, weitere drei Ecken, die nur eine Farbe mit der Referenzecke gemeinsam haben und die der Referenzecke gegenüberliegende Ecke, die keine Farbe mit der Referenzecke gemeinsam hat.

### 7.1.3 Evaluation der Deskriptoren auf dem sechsfarbenen Würfel

Dieses erste Experiment dient zur Überprüfung der grundlegenden Funktionsweise der Deskriptoren. Diese ist vor den Experimenten mit dem Detektor erforderlich, da in den folgenden Experimenten die Ergebnisse der Deskriptoren zur Bestimmung der Qualität des Detektors herangezogen werden. Die Ergebnisse der Deskriptorevaluation sind in Abbildung 7.4 auf Seite 70 zu sehen.

### Diskussion der Ergebnisse

Der Formdeskriptor zeigt keinerlei Differenzen, unabhängig davon, ob er mit Merkmalen derselben Ecke verglichen wird oder mit allen anderen Ecken.

Der Farbdeskriptor ist bei Merkmalen derselben Ecke hingegen sehr ähnlich, beim Vergleich mit allen anderen Ecken sind drei Gruppen von Merkmalen auszumachen, deren Farbdeskriptoren zur Referenzecke deutlich unterschiedliche Distanzen aufweisen. Die Ergebnisse zeigen, dass die Deskriptoren kein Verhalten aufweisen, welches nicht intendiert ist.

#### **7.1.4 Experimente zur Bestimmung sinnvoller Parameter für den Detektor**

In Kapitel 5.1 sind die verschiedenen Parameter und die Zusammenhänge untereinander erklärt worden, und während der Arbeit am Detektor entstanden auch grobe Einschätzungen, welche Werte bei den Parametern zu sinnvollen Ergebnissen führen. Diese Einschätzungen sollen mittels der folgenden Experimente durch Testergebnisse evaluiert werden.

Ziel der Experimente ist einmal der Einfluss der verschiedenen Parameter auf die Qualität der Ergebnisse. Mit den Evaluationen der Parameter wird dann geprüft, ob sich feste Relationen zwischen den Parametern festlegen lassen und welche optimal sind. Folgende Parameter werden evaluiert:

1. Die Abtastrate. Die Ergebnisse finden sich in den Abbildungen 7.5 und 7.6 auf Seite 71 und 72.
2. Die Normalenabtastrate. Die Ergebnisse finden sich in den Abbildungen 7.7 und 7.8 auf Seite 73 und 74.
3. Die Normalenskala. Die Ergebnisse finden sich in den Abbildungen 7.9 und 7.10 auf Seite 75 und 76.
4. Die Relation zwischen den Parametern Abtastrate, Histogrammskala, Normalenskala und Normalenabtastrate. Für die Evaluation sind sowohl die Ergebnisse der Punkte 1. - 3. relevant, als auch das Ergebnis in Abbildung 7.11 auf Seite 77.

#### **Diskussion der Experimente zur Abtastrate**

Die Abtastrate bestimmt den Grad der Diskretisierung der Punktwolke zur Suche nach lokalen Maxima der Entropie. Je kleiner die Abtastrate, desto besser sollten die Ergebnisse sein, je größer, desto schneller ist die Detektion. Die Ergebnisse aus Abbildung 7.5 zeigen, dass die Wahl der Abtastrate die Qualität der Merkmale beeinflusst, dies sich aber erst bei der größten Abtastrate deutlich zeigt.

#### **Diskussion der Experimente zu Normalenabtastrate und Normalenskala**

Die Wahl der Normalenabtastrate und Normalenskala ist nicht unwichtig, weil unterschiedliche Werte Einfluss auf die Geschwindigkeit der Detektion

haben. Es ist also interessant zu wissen, welche Werte erforderlich sind, um sowohl den Laufzeitbedarf als auch den Einfluss auf die Qualität zu minimieren. Die Grafiken in Abbildung 7.7 zeigen auch unter Rauschen kaum Unterschiede für eine variierende Normalenabtastrate, lediglich bei 8 cm sinkt die Wiederholpräzision sichtbar. In Abbildung 7.9 sehen wir hingegen, dass eine kleine Normalenskala gegenüber verrauschten Daten anfällig ist. Dies gilt sowohl für die Wiederholpräzision als auch für die Deskriptoren, deren Qualität deutlich sinkt, mit Ausnahme des Formdeskriptors bei 2 cm Normalenskala und 1 cm Rauschen. Den Einfluss der Normalenabtastrate kann man als gering bezeichnen. Dass die Werte schlechter werden, wenn Normalenabtastrate und Normalenskala gleich sind, war zu erwarten, hier geht je nach Lage des Oktalbaums zur Punktwolke Information verloren, zum Beispiel signifikante Oberflächenänderungen an der Grenze zwischen zwei Knoten der Normalenabtastrate. Die Normalenskala beeinflusst das Ergebnis negativ, sollte sie im Vergleich zum Rauschen zu klein gewählt sein. Zusammenfassend kann man sagen, dass eine Normalenskala von 4 cm eine sinnvolle Untergrenze darstellt, und da die Normalenabtastrate etwas kleiner ausfallen sollte, sollte diese höchstens die Hälfte der Normalenskala betragen.

## Diskussion zu Relationen zwischen den Parametern

Abbildung 7.11 zeigt eine leichte Bevorzugung von kleinen Abtastraten im Vergleich zur Histogrammskala. Dies ist auch in den Experimenten zur Abtastrate zu sehen, aber hier für eine variierende Histogrammskala bestätigt. Zusammen mit den Ergebnissen der Normalenabtastrate und Normalenskala ergibt sich folgendes Bild: Die Abtastrate sollte möglichst um den Faktor 3 kleiner sein als die Histogrammskala, die Normalenabtastrate sollte höchstens die Hälfte der Normalenskala und der Abtastrate betragen und die Normalenskala sollte 4 cm nicht unterschreiten. Folgende Tabelle fasst die Restriktionen zu einem Richtwert der Verhältnisse zusammen:

Optimales Verhältnis der Parameter			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
1	3	1	0.5

### 7.1.5 Experimente zum Einfluss von Sensorrauschen

Da die Punktwolken der Kinect ein Sensorrauschen aufweisen, soll der Einfluss des Sensorrauschens anhand von künstlich verrauschten Punktwolken des Würfels untersucht werden. Die Koordinaten der Punkte werden um einen normalverteiltem Zufallswert verändert, der Mittelwert ist 0, die Standardabweichung kann festgelegt werden. Die Ergebnisse sind in Diagramm 7.12, 7.14 und 7.13 auf den Seiten 78, 80 und 79 zu sehen.

### **Einfluss von Rauschen**

Die Ergebnisse zeigen einen deutlichen Einfluss von Sensorrauschen sowohl auf die Wiederholpräzision als auch auf die Deskriptoren. Allerdings bewirkt die verbesserte Lokalisation durch Mean-Shift eine deutliche Verbesserung der Ergebnisse selbst bei sehr starkem Rauschen.

#### **7.1.6 Einfluss von Eckigkeit und Entropieschwellwert**

Nachdem Eckigkeit in 5.5 und der Entropieschwellwert in 5.4 definiert worden sind, dienen die folgenden Experimente zur Evaluation der Funktion beider Konzepte als auch zur Einschätzung, welcher Wertebereich für den Einsatz optimal ist. Dazu wurden zwei Experimente durchgeführt: Im ersten wird ermittelt, wieviele lokale Maxima auf dem Würfel in Abhängigkeit der beiden Variablen gefunden werden. Der Wert sollte im Idealfall acht betragen, da der Würfel entsprechend viele Ecken besitzt und andere Positionen auf dem Würfel ungeeignet sind, da sie potenziell instabil sind. Im zweiten Experiment wird betrachtet, wie zuverlässig die acht Ecken des Würfels erkannt werden. Ziel ist ein Wertebereich, in dem die Anzahl der ungewünschten Merkmale minimal und die Zuverlässigkeit bei den gewünschten Merkmalen maximal ist. Die Ergebnisse sind in den Abbildungen 7.15 auf Seite 81 und 7.16 auf Seite 7.16 zu sehen.

#### **Diskussion der Experimente zur Eckigkeit**

Verrauschte Daten führen hier zu deutlichen Unterschieden in den Ergebnissen; die Anzahl der Detektionen verdoppelt sich mit Sensorrauschen und fällt deutlich langsamer. Ab einem Wert von 0.2 riskiert man allerdings einen Rückgang in der Zuverlässigkeit. Da dieser Wert nur für die Ecken des Würfels ermittelt worden ist, kann dieser bei andersartigen Merkmalen abweichen. Als Richtwert ist während der Experimente meistens 0.15 verwendet worden, die Zahl der unerwünschten Merkmale wird deutlich reduziert, ohne negative Auswirkungen auf gute Merkmale fürchten zu müssen. Allerdings ist die Anzahl der unerwünschten Merkmale bei einem Wert von 0.15 immer noch hoch, insbesondere bei verrauschten Daten.

#### **Diskussion der Experimente zum Entropieschwellwert**

Die Ergebnisse zum Entropieschwellwert sind einfach zu interpretieren. Zwar ist auch hier die Zahl der Detektionen bei verrauschten Daten sehr hoch, sie sinkt aber auch sehr viel stärker mit steigendem Schwellwert. Die Zuverlässigkeit bleibt bis zu einem Schwellwert von 0.6 sehr hoch, sinkt dann aber rapide auf Null bei einem Wert von ungefähr 0.8. Beides zusammen spricht für einen Entropieschwellwert im Wertebereich von 0.5 - 0.7.

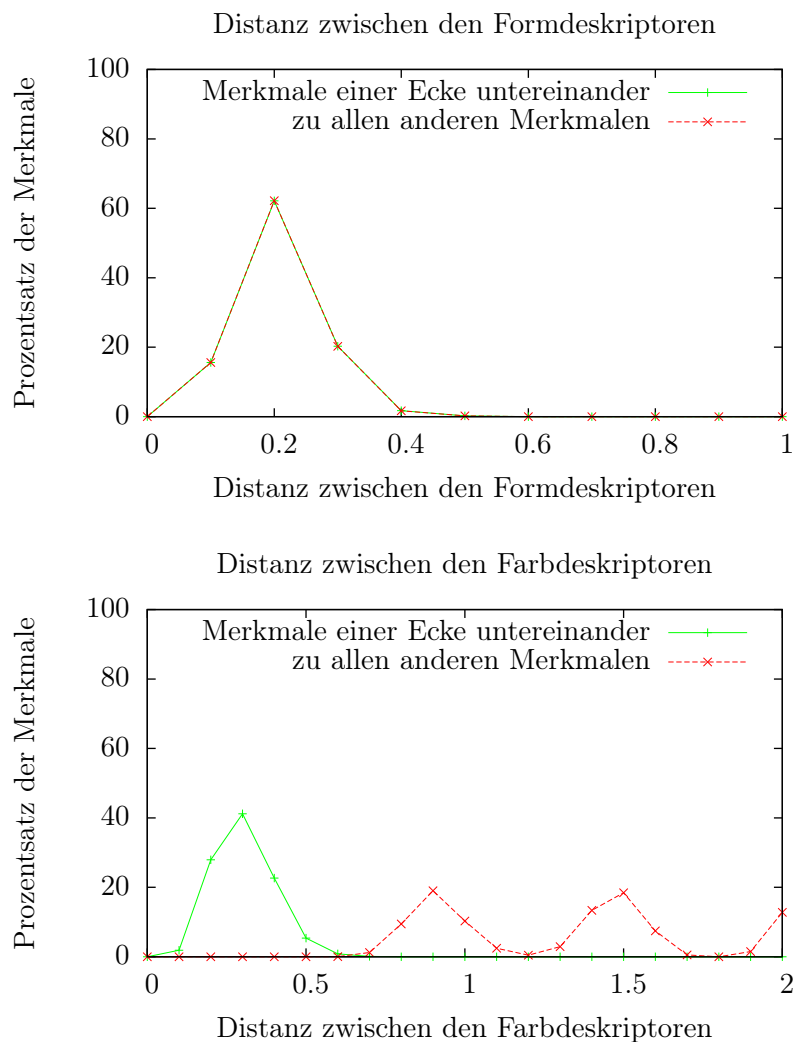


Abbildung 7.4: Durchschnittsdistanzen der Form- und Farbedeskriptoren der Merkmal an den Ecken des Würfels. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören. Dann werden die Merkmale einer Ecke mit den Merkmalen aller anderen Ecken verglichen. Durch die unterschiedliche Einfärbung des Würfels sind die Farbedeskriptoren teilweise sehr unterschiedlich, abhängig davon, wieviele gemeinsame Seiten die verglichenen Merkmale haben.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	32 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	nein

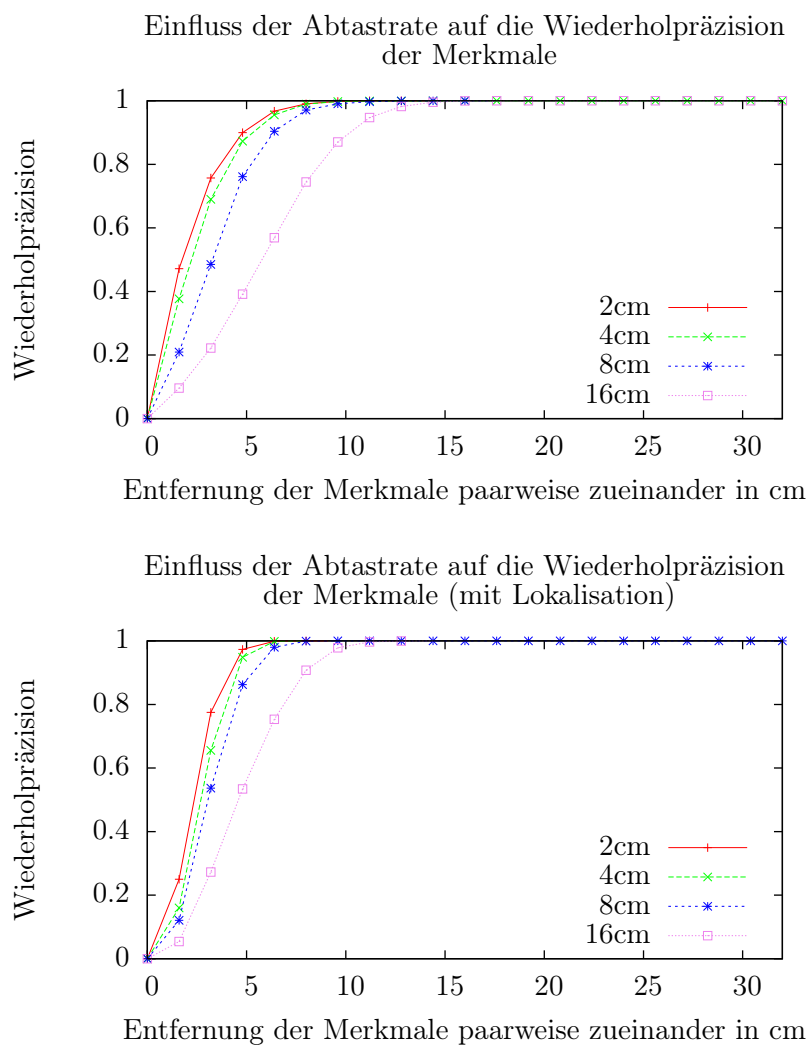


Abbildung 7.5: Die Diagramme zeigen die Entwicklung der Wiederholpräzision bei steigender Entfernung der Merkmale untereinander. Beim unteren Diagramm wurden die Merkmale nachträglich mit Mean-Shift lokalisiert.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
variabel	32 cm	4 cm	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nur unten	nein

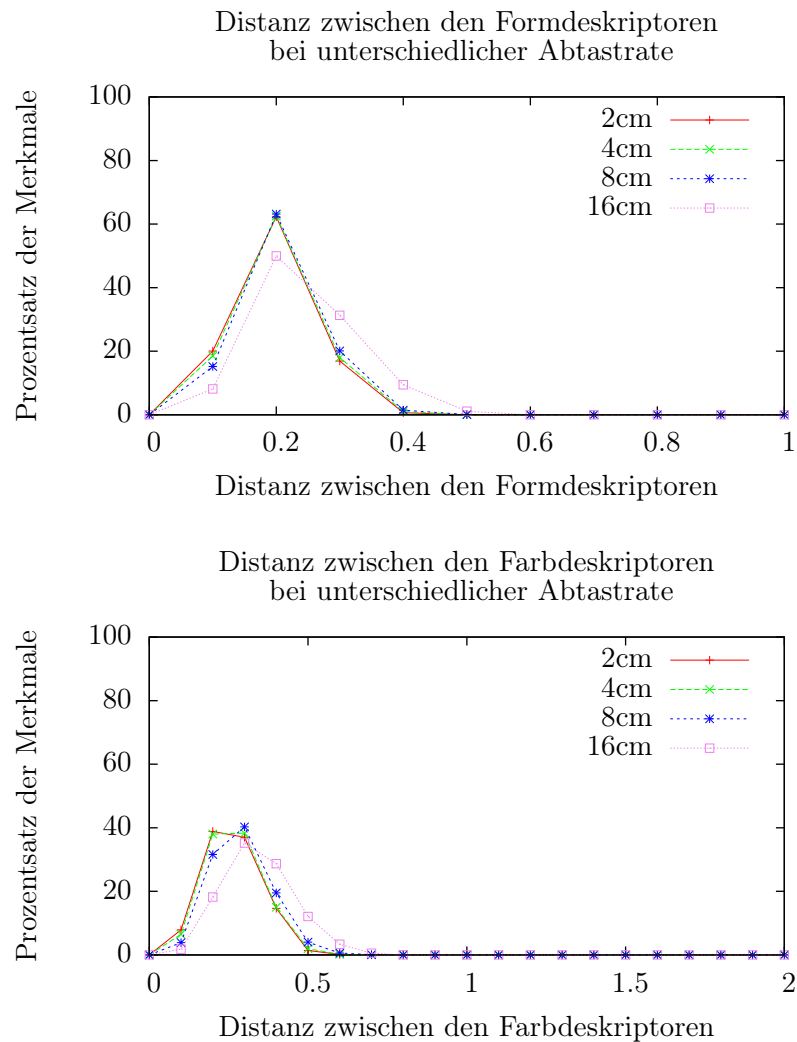
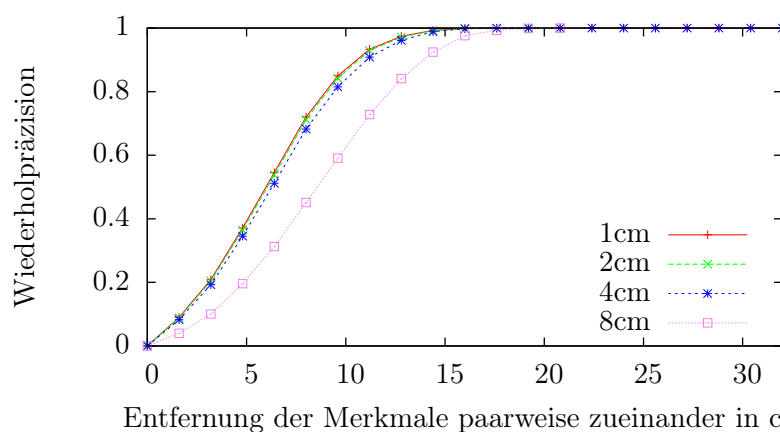


Abbildung 7.6: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmale an den Ecken des Würfels bei variabler Abtastrate. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören. Die Abtastrate hat kaum einen Einfluss auf die Deskriptoren, lediglich bei 16 cm sind Veränderungen auszumachen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
variabel	32 cm	4 cm	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	nein



Einfluss der Normalenabtastrate auf die Wiederholpräzision der Merkmale



Einfluss der Normalenabtastrate auf die Wiederholpräzision der Merkmale (mit Rauschen)

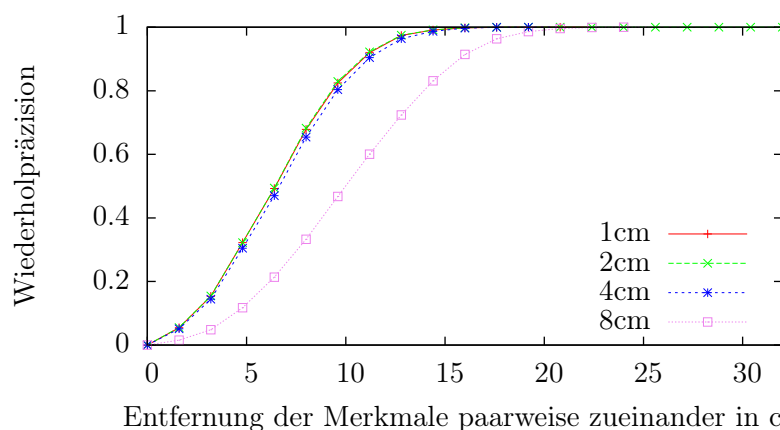


Abbildung 7.7: Die Diagramme zeigen den Einfluss der Normalenabtastrate auf die Präzision des Detektors. Nur bei einem Wert von 8 cm zeigen sich Verschlechterungen der Wiederholpräzision

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
16 cm	32 cm	8 cm	variabel
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	nur unten, 1 cm

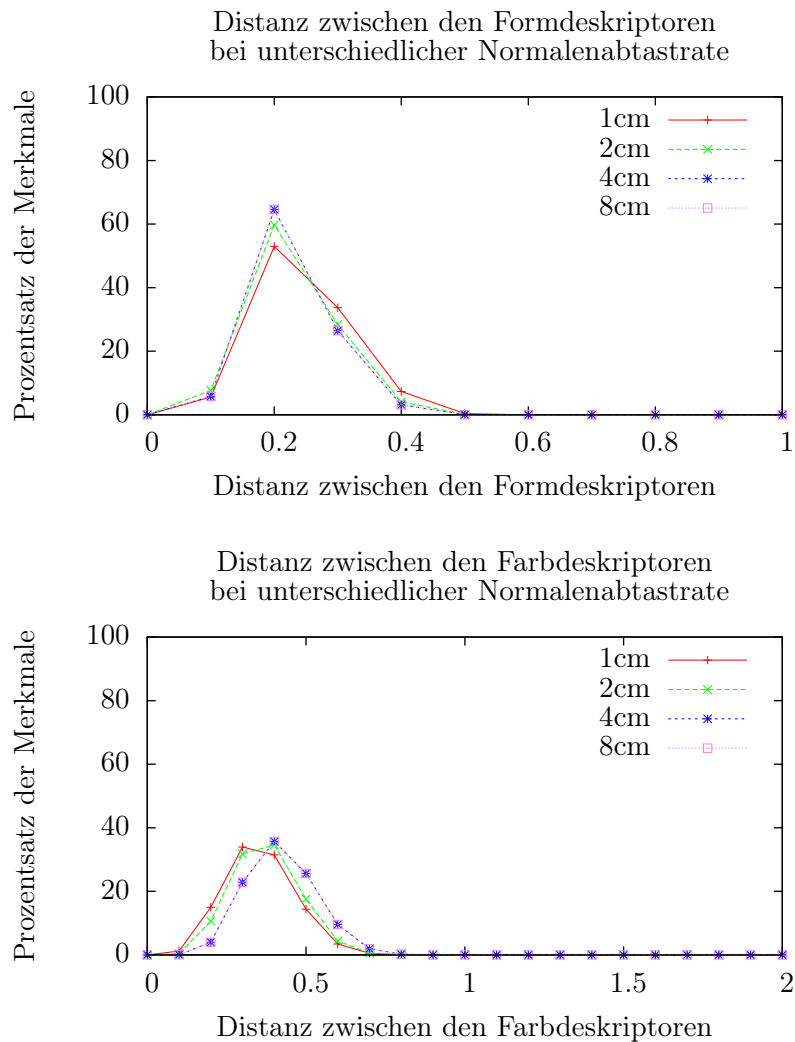


Abbildung 7.8: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmale an den Ecken des Würfels bei variabler Normalenabtastrate. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören. Leichte Veränderungen gibt es bei großen Abtastraten, hier sinkt allerdings dann auch die Zahl der für die Deskriptoren verwendeten Normalen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
16 cm	32 cm	8 cm	variabel
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	1 cm

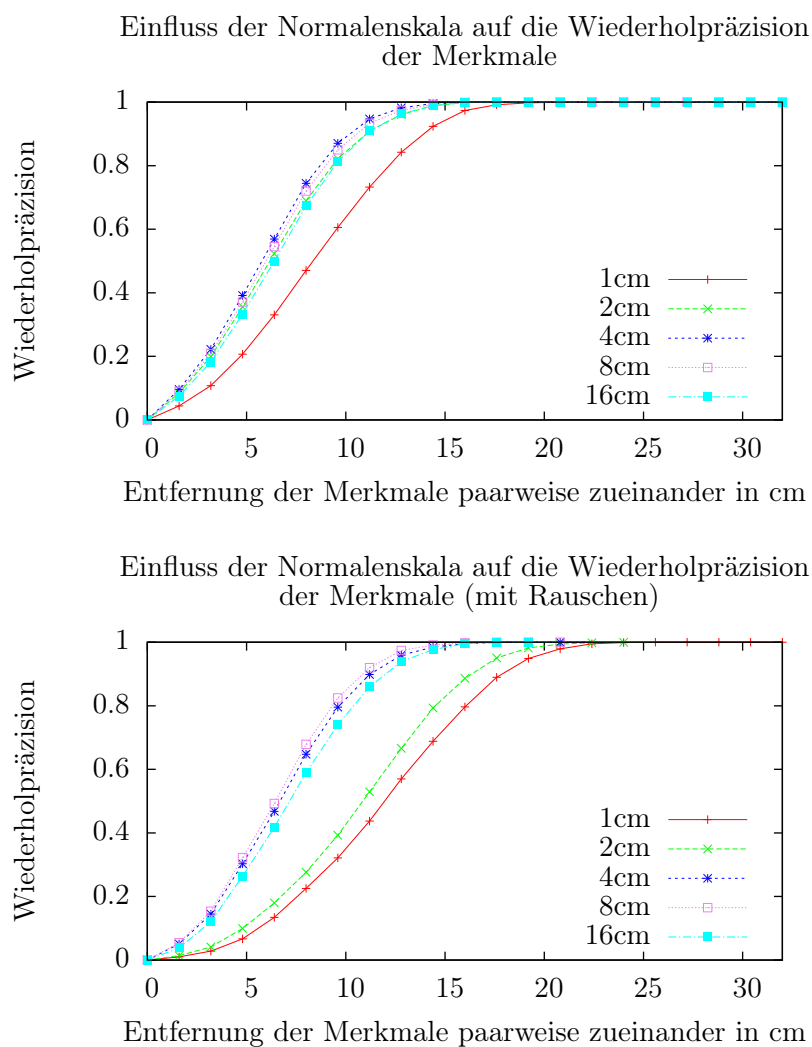


Abbildung 7.9: Die Diagramme zeigen den Einfluss der Normalenskala auf die Präzision des Detektors. Normalen einer kleinen Normalenskala reagieren mangels genügend Kontext in der Punktwolke stark auf Sensorrauschen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
16 cm	32 cm	variabel cm	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	nur unten, 1 cm

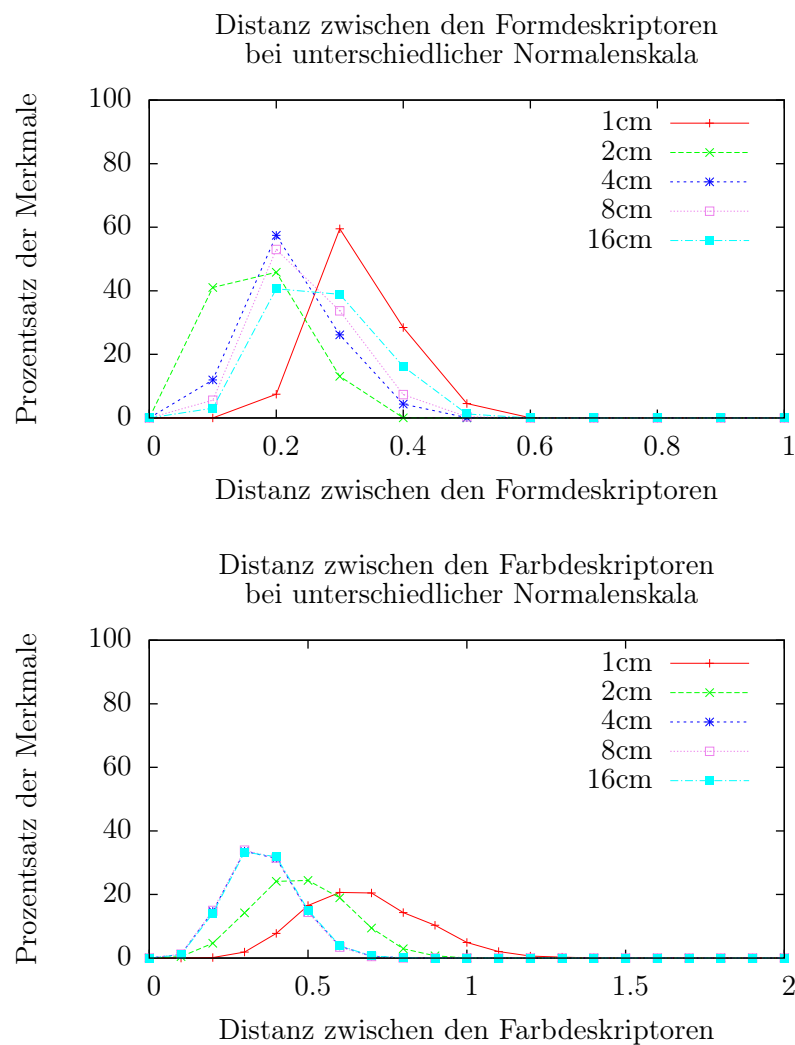


Abbildung 7.10: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmale an den Ecken des Würfels bei variabler Normalenskala. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
16 cm	32 cm	variabel	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	1 cm

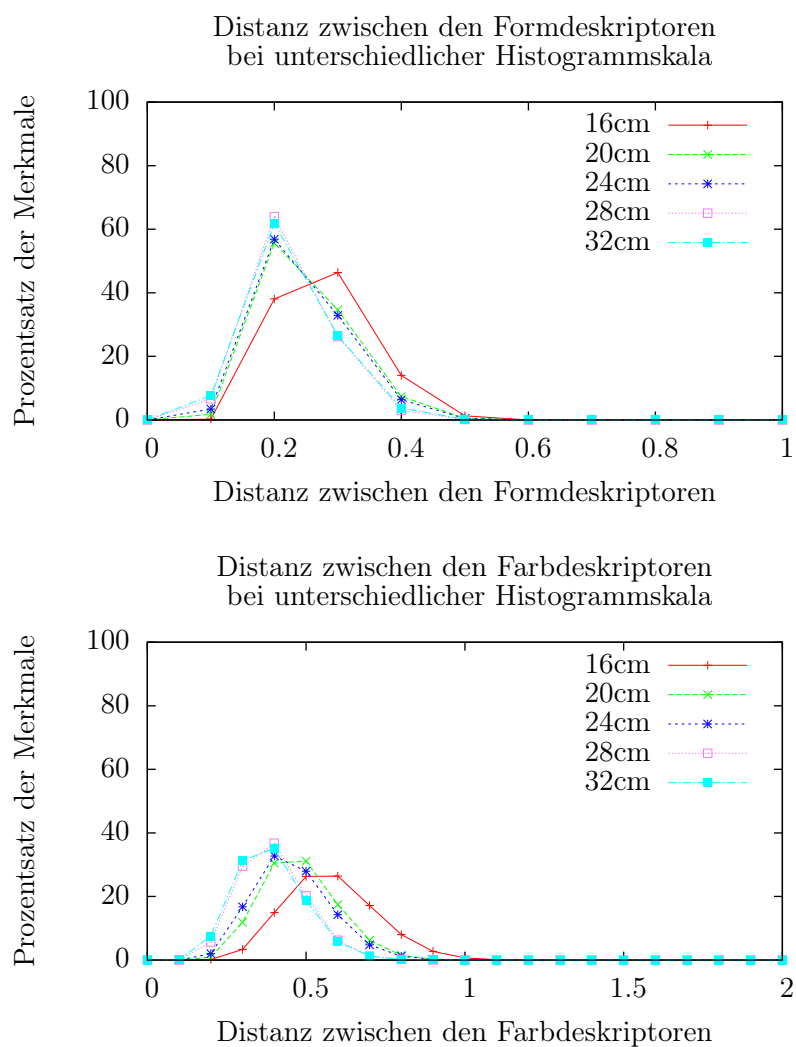


Abbildung 7.11: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmale an den Ecken des Würfels bei variabler Histogrammskala. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
8 cm	variabel	8 cm	4 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nein	1 mm

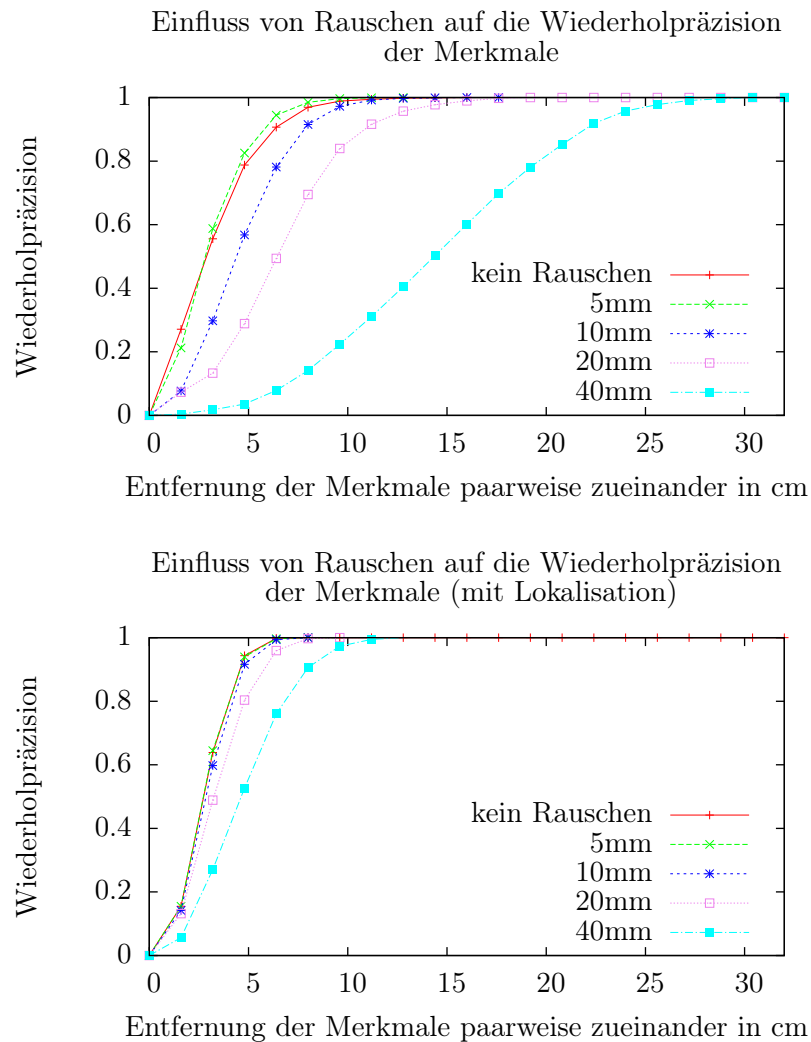


Abbildung 7.12: Die Grafiken zeigen die Entwicklung der Wiederholpräzision bei steigender Entfernung der Merkmale untereinander. Im Gegensatz zu Abbildung 7.5 sind die Punkte in der Punktwolke mit einem normalverteilten Rauschen versehen. Die Kurven zeigen die Ergebnisse für unterschiedliche Standardabweichungen. In der unteren Grafik wurden die Ergebnisse nachträglich lokalisiert.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	32 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nur unten	variabel

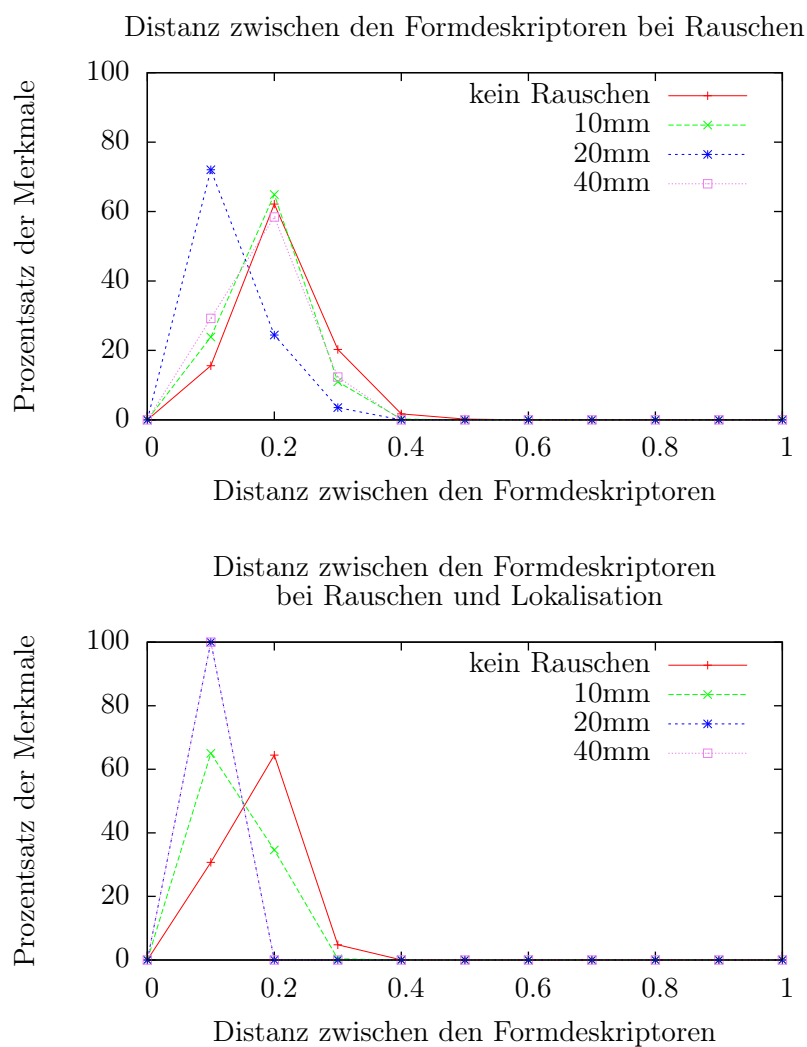


Abbildung 7.13: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmal an den Ecken des Würfels bei künstliche hinzugefügtem Rauschen. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	32 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nur unten	variabel

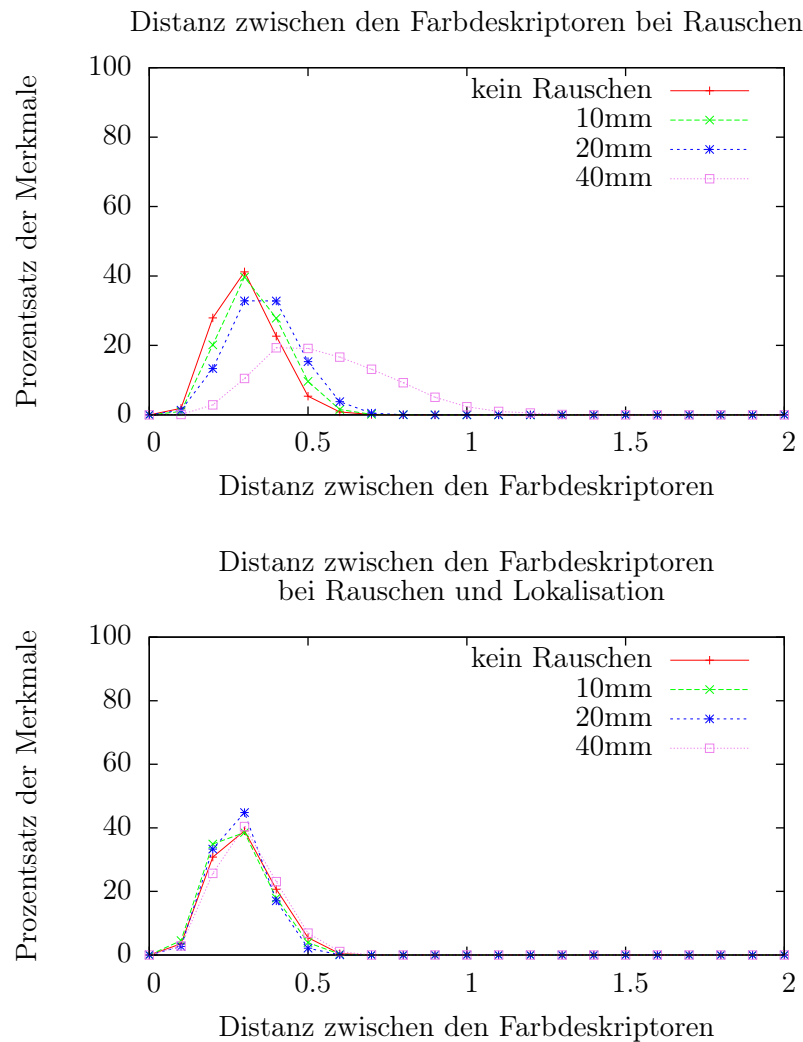


Abbildung 7.14: Durchschnittsdistanzen der Form- und Farbdeskriptoren der Merkmal an den Ecken des Würfels bei künstliche hinzugefügtem Rauschen mit nachträglicher Lokalisation der Merkmale. Es werden alle Merkmale untereinander verglichen, die zu derselben Ecke gehören.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	32 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	nur unten	variabel



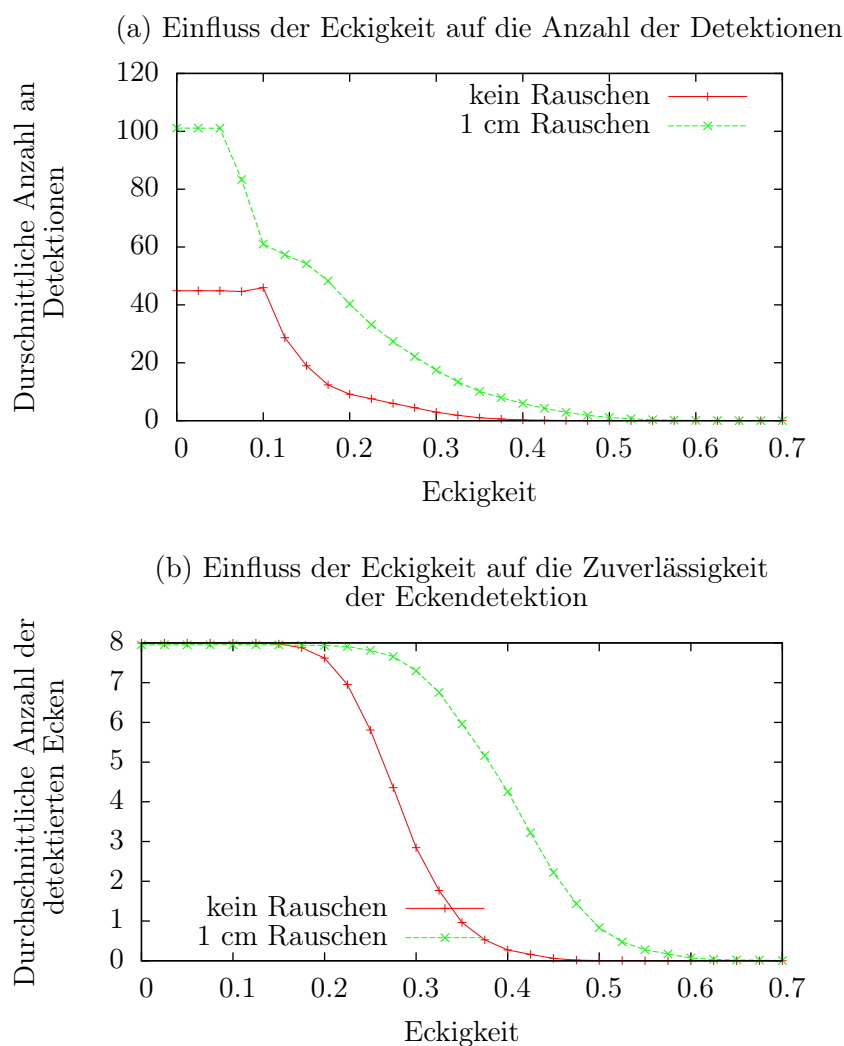


Abbildung 7.15: Grafik (a) zeigt die Anzahl der detektierten Regionen im Verhältnis zur Eckigkeit. Mit steigender Eckigkeit sinkt die Anzahl der Detektionen, wobei diese bei verrauschten Punktwolken sehr hoch bleibt. Grafik (b) zeigt, wie sich die Eckigkeit auf die Zuverlässigkeit der Detektion der Ecken des Würfels auswirkt. Ein hoher Wert in der Eckigkeit bewirkt einerseits eine Reduktion der falschen Detektionen, andererseits senkt er auch die Zahl der richtigen Detektionen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	16 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.5	variabel	nein	variabel

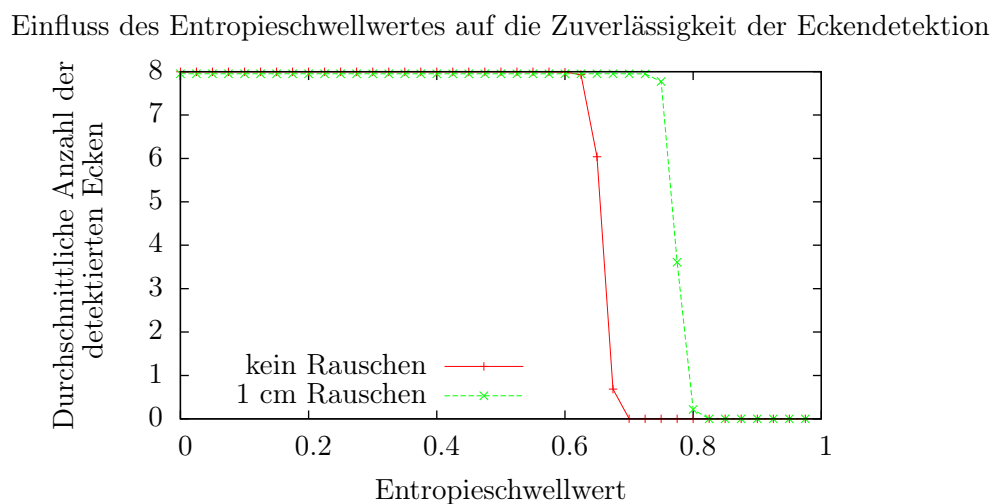
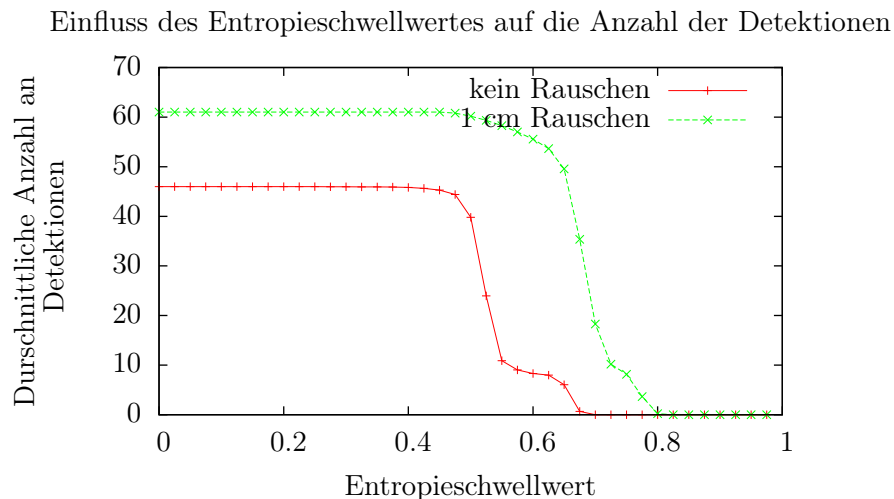


Abbildung 7.16: Die obere Grafik zeigt die Anzahl der detektierten Regionen im Verhältnis zum Entropieschwellwert. Die Anzahl der Detektionen bleibt ungefähr bis zum Wert von 0.5 konstant und fällt dann ab. Die untere Grafik zeigt die Auswirkungen des Schwellwertes auf die Zuverlässigkeit der detektierten Ecken. Bis zu einem Wert von 0.6 werden die acht Ecken des Würfels zuverlässig erkannt, danach fällt die Kurve steil ab.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	16 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
variabel	0.15	nein	variabel

## 7.2 Experimente zum Vergleich von Merkmalen zwischen Einzelbildern

In diesen Experimenten werden die Ergebnisse der Detektion und Deskription auf realen Datensätzen ausgewertet. Die Experimente umfassen dabei mehrere Objekte, die aus verschiedenen Blickwinkeln mit der Kinect aufgenommen wurden. Die entstandenen Punktwolken wurden in einem Referenzkoordinatensystem registriert.

### 7.2.1 Erstellung und Beschreibung der Datensätze

Der Aufbau der Experimente ist wie folgt: Eine Kinect wird auf einem Stativ befestigt. Das Stativ ist in der Höhe verstellbar, auch kann man die Neigung der Kinect in Längs- und Querachse verändern. Die Kinect wird dann auf eine Szene ausgerichtet, die sowohl Objekte als auch Blätter mit ausgedruckten Schachbrettmustern („Checkerboards“) enthält. Dann werden mit der Kinect Einzelbilder der Szene sowohl von der Kamera als auch von der Tiefenkamera aufgenommen. Zwischen den Einzelbildern wird die Kinect bewegt. Verschiedene Bewegungsmuster kamen hier zum Einsatz:

- Die Kinect wird ungefähr auf einer Kreisbahn um die Szene herum bewegt, wobei die Kinect immer auf die Mitte des Kreises ausgerichtet ist. Die Höhe und die Neigungswinkel der Kinect bleiben konstant, der Abstand zur Szene sollte nicht allzu stark variieren. Die Aufnahmen dienen zur Bestimmung der Blickwinkelabhängigkeit der Merkmale.
- Die Kinect wird entlang einer Geraden von der Szene wegbewegt. Der Nickwinkel muss eventuell korrigiert werden, damit die Szene komplett im Bild bleibt. Die Aufnahmen liefern Aufschluss darüber, inwieweit die Merkmale unter Entfernungsänderung stabil bleiben. Dies ist insbesondere aufgrund der unterschiedlichen Auflösung der Punktwolke in Abhängigkeit von der Entfernung zur Kinect von Bedeutung.

Im ersten dieser Einzelbilder wird anhand des Kamerabildes die Pose des Checkerboards berechnet und alle folgenden Aufnahmen werden anhand der sichtbaren Checkerboards in das Koordinatensystem der ersten Aufnahme transformiert. Dies stellt sicher, dass sich die Objekte in der Szene unabhängig von der Bewegung der Kinect immer ungefähr an denselben Koordinaten befinden. Damit das funktioniert, muss immer ein Checkerboard im Kamerabild komplett zu sehen sein. Um dies trotz Verdeckung durch Objekte der Szene zu gewährleisten, ist es auch möglich, mehrere unterschiedliche Checkerboards zu verwenden, hier muss allerdings darauf geachtet werden, dass mindestens in einer Aufnahme zwei Checkerboards zu sehen sind, um die Transformation zwischen den Checkerboards zu ermitteln.

In diesem Experiment wurden vier Objekte aufgenommen wie in 7.2.1 beschrieben: ein Karton, ein Ball, ein Zauberwürfel und ein Teddybär (siehe Abbildung 7.17). Jedes Objekt wurde dabei einmal in einer Entfernung von ungefähr einem Meter und einmal in einer Entfernung von ungefähr zwei Metern aufgenommen.

Entstanden sind acht Zusammenstellungen aus 60 - 150 Einzelaufnahmen, wobei die Aufnahmen für den Ball und Rubiks Zauberwürfel nur aus einem Halbkreis heraus gemacht worden sind. Die Struktur dieser Objekte ist von allen Seiten gleich und mehr Aufnahmen hätten daher keine neuen Erkenntnisse gebracht. In den Zusammenstellungen sind jeweils folgende Daten für jede Einzelaufnahme enthalten:

- Das normale Kamerabild als Graustufenbild in VGA-Auflösung.
- Die Punktwolke der Szene mit RGB-Farbinformationen im Koordinatensystem des ersten gefundenen Checkerboard im ersten Einzelbild.
- Die Transformation der Szene im Bezug zum ersten gefundenen Checkerboard im ersten Einzelbild

Insgesamt umfassen die Daten der Zusammenstellungen eine Größe von 12 Gigabyte.

### 7.2.2 Vorgehensweise

Die Experimente auf den Datensätzen liefen wie folgt ab:

1. Aus einem Datensatz werden ein Einzelbild und die Parameter für den Detektor ausgewählt und der Detektor auf der entsprechenden Punktwolke berechnet. Die so gefundenen Merkmale sind die Referenzmerkmale in den folgenden Schritten.
2. Für alle anderen Bilder aus dem Datensatz werden die Merkmale berechnet und es wird geprüft, ob die Distanz des Merkmalszentrums zu einem der Referenzmerkmale kleiner als die Histogrammskala ist. Dieses Merkmal wird dann dem Referenzmerkmal zugeordnet. Aus den in jedem Einzelbild enthaltenen Transformationen wurde nun berechnet, wie sich der Blickwinkel auf das Merkmal im Vergleich zum Referenzmerkmal verändert hat. In einer Tabelle wurde nun abhängig vom Blickwinkel für alle Szenen festgehalten, wieviele Merkmale gefunden worden sind
3. Die Schritte 1. und 2. wurden für jedes Einzelbild des Datensatzes durchgeführt.

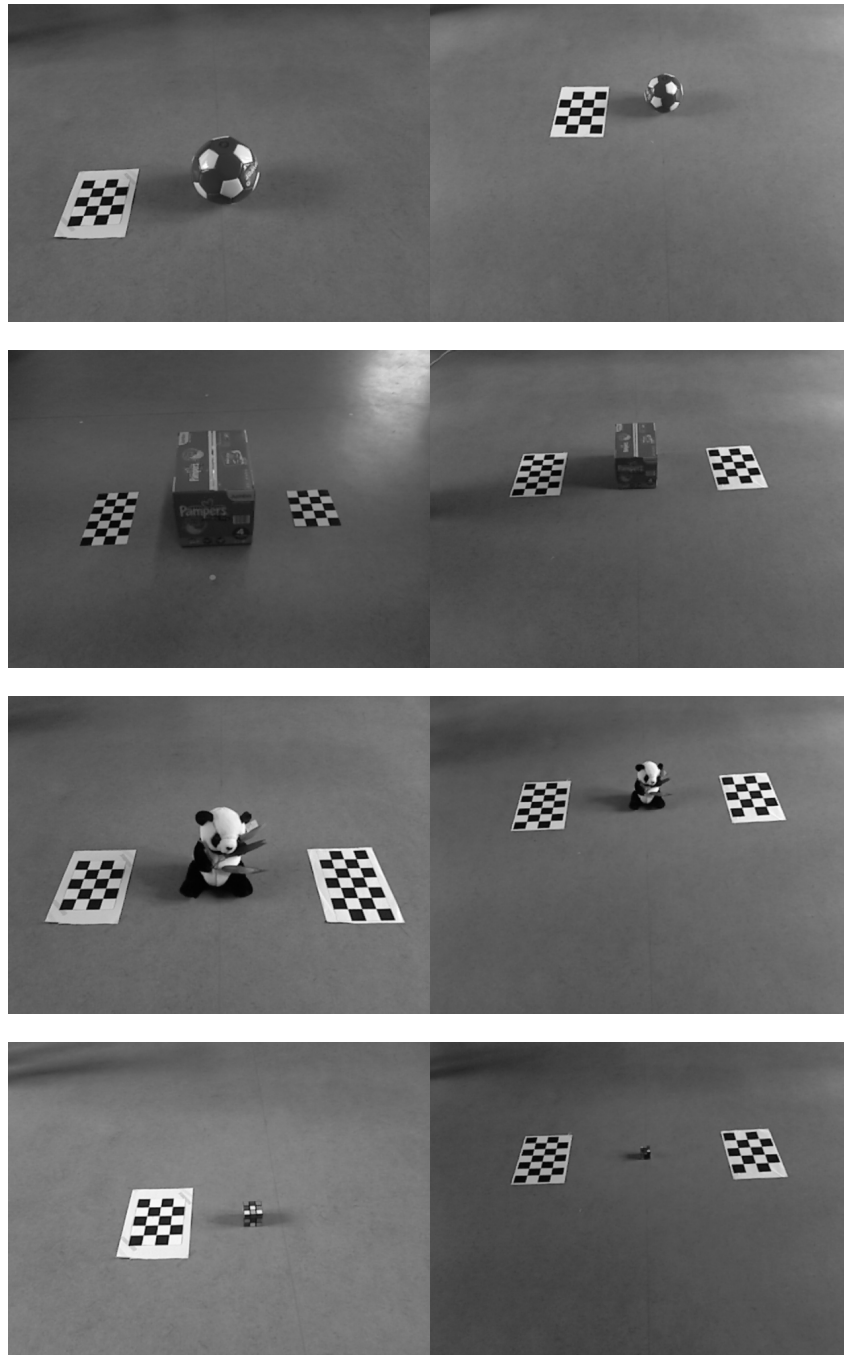


Abbildung 7.17: Graustufenbilder der ersten Einzelbilder aus den Szenen aus einem Meter Entfernung (links) und zwei Metern Entfernung (rechts)

In der Auswertung ließ sich nun die Wiederholbarkeit der Merkmale als Funktion des Blickwinkels darstellen.

Zusätzlich dazu sind Durchläufe durchgeführt worden, in denen die Wiederholbarkeit nicht in Abhängigkeit vom Winkel zum Referenzbild berechnet wurde, sondern in Abhängigkeit vom mittleren Sichtwinkel auf ein Merkmal. Dazu wurde zunächst ermittelt, aus welchen Blickwinkeln ein Merkmal zu sehen war. Anschließend wurde der Mittelwert aus diesen Blickwinkeln gebildet, wobei die äußeren zehn Prozent nicht berücksichtigt worden sind. Die Sichtbarkeit des Merkmals wurde nun ausgehend vom entstandenen mittleren Blickwinkel evaluiert.

Dies hat den Grund, dass jedes Einzelbild als Referenzbild verwendet wird und somit auch Merkmale am Rande ihres Sichtbereiches als Referenzmerkmale verwendet werden. Dann reichen meist kleine Veränderungen im Winkel aus, und das Merkmal verschwindet, so dass das Endergebnis für kleine Winkeländerungen dementsprechend negativ beeinflusst wird. Dieses Experiment soll diesen Faktor minimieren.

### **7.2.3 Experimente**

1. Die Wiederholbarkeit des Detektors auf Szenen der vier vorgestellten Objekte unter Blickwinkeländerungen, siehe Abbildung 7.18.
2. Die Wiederholbarkeit des Kartons und des Teddybären ausgehend vom mittleren Blickwinkel, siehe Abbildung 7.19.
3. Die Wiederholbarkeit des Zauberwürfels bei variierenden Histogrammskalen, siehe Abbildung 7.20.
4. Die Wiederholbarkeit des Teddybären bei variierenden Histogrammskalen, siehe Abbildung 7.21.

### **Diskussion der Wiederholbarkeit des Detektors in den einfachen Szenen**

Der Ball und der Zauberwürfel sind bei einem Meter Entfernung nahezu immer und bei zwei Metern noch sehr gut zu erkennen, wobei der Zauberwürfel hier schon an die Grenze des Detektors stößt. Die Wiederholbarkeit des Kartons in einem Meter Entfernung fällt relativ schnell ab, ein Ergebnis, dass daran liegt, dass Ecken nur noch sporadisch erkannt werden, sobald nicht mehr alle drei Seiten zu sehen sind. Auch werden hier Merkmale auf den Kanten des Kartons erkannt, welche nicht gut wiedererkannt werden. Der Teddy zeigt diese Probleme nicht, was zum einen an den eher runden Formen liegt, die deutlich länger sichtbar bleiben als Ecken als auch daran, dass auf dem Teddy keine Kantenmerkmale erkannt werden. Interessanterweise fällt die Wiederholbarkeit sowohl des Teddys als auch des Kartons bei zwei

Metern nicht so stark, wobei der Teddy schon keine guten Werte für geringe Blickwinkeländerungen zeigt.

### **Diskussion der Wiederholbarkeit des Detektors bei mittlerem Blickwinkel**

Hier zeigt sich beim Karton, dass die Sichtbarkeit bei geringen Blickwinkeländerungen sehr hoch ist, aber bei mehr als  $30^\circ$  in eine Richtung stark abfällt. Das ist eine Folge davon, dass ab diesem Winkelbereich bei den Ecken des Kartons eine Seite wegfällt. Die Werte beim Teddy fallen nicht so stark ab, was für eine längere Sichtbarkeit runder Formen spricht. Die Werte sind deutlich niedriger als bei den Experimenten davor, das liegt aber auch an der kleineren Skala des Teddys.

### **Diskussion der Wiederholbarkeit des Zauberwürfels bei variabler Histogrammskala**

Dieses Experiment dient zur Verdeutlichung der Wirkung der Histogrammskala. Bei einer kleinen Histogrammskala und einer Entfernung von einem Meter wird ein Merkmal an einer der Ecken oder der Seiten des Würfels erkannt, was bei starken Blickwinkeländerungen aus dem Sichtbereich fällt. Auch die verbesserte Lokalisation schafft hier keine Abhilfe, da diese das Merkmal nur leicht innerhalb des Histogrammbereichs verschiebt. Bei einer Histogrammskala von 12 cm und 18 cm liegt das Merkmal dank der Lokalisation komplett im Würfel selbst und wird daher von allen Seiten wiedererkannt. Eine Histogrammskala von 24 cm ist dagegen schon zu groß. Bei zwei Metern Entfernung werden diese Effekte durch Sensorrauschen und Tiefendiskretisierung noch verstärkt, wobei deutlich wird, dass auf die Entfernung ein Objekt von der Größe des Zauberwürfels zu klein ist.

### **Diskussion der Wiederholbarkeit des Teddys bei variabler Histogrammskala**

In diesem Experiment sieht man deutlich, dass ein Objekt auf mehreren Histogrammskalen Merkmale erzeugt. Die kleinen Skalen sind nicht so stabil wie die Großen, wobei die großen Skalen schon einen Großteil des Teddys oder ihn ganz umfassen. Interessanterweise profitiert die größte Skala von 24 cm von einer Erhöhung der Entfernung auf zwei Meter. Hier scheinen Sensorrauschen und verbesserte Lokalisierung zur Positionierung des Merkmals im Teddy zu führen, wo es von allen Seiten erkannt werden kann.

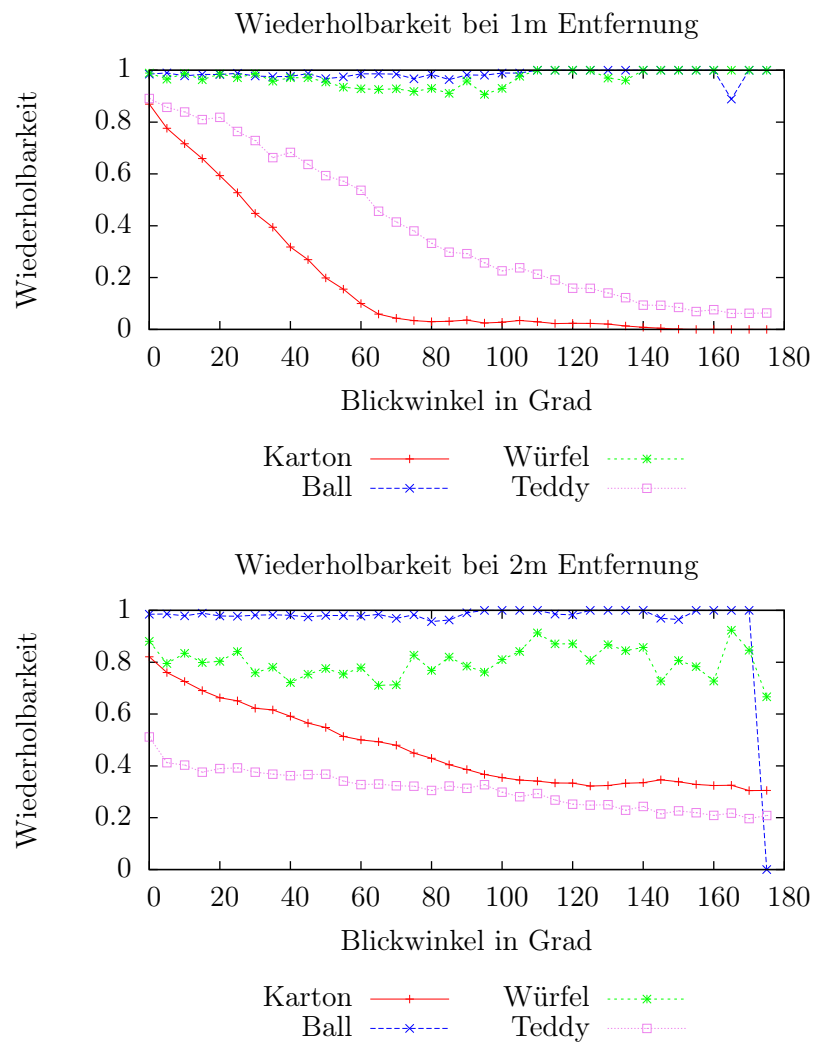


Abbildung 7.18: Wiederholbarkeit des Detektors

Parameter des Experiments				
Objekt	Abtastrate	Hist.-skala	Norm.-skala	Norm.-abtastrate
Karton	4 cm	12 cm	4 cm	2 cm
Ball	8 cm	24 cm	4 cm	2 cm
Würfel	2 cm	12 cm	4 cm	1 cm
Teddy	4 cm	12 cm	4 cm	2 cm
Objekt	min. Entropie	Eckigkeit	Lokalisation	Rauschen
Karton	0.6	0.15	ja	nein
Ball	0.6	0.15	ja	nein
Würfel	0.6	0.15	ja	nein
Teddy	0.6	0.15	ja	nein



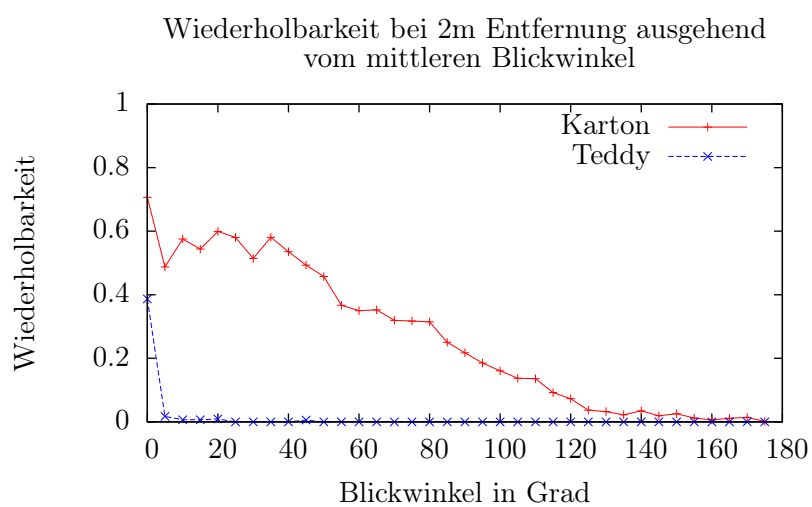
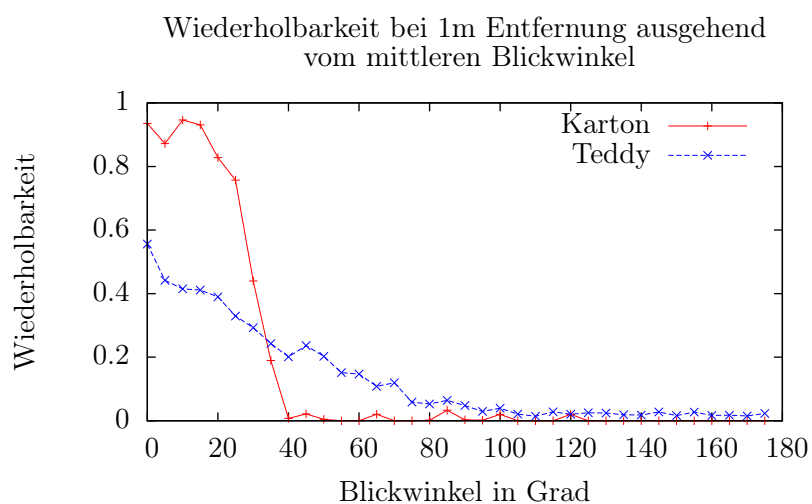


Abbildung 7.19: Wiederholbarkeit des Detektors

Parameter des Experiments				
Objekt	Abtastrate	Hist.-skala	Norm.-skala	Norm.-abtastrate
Karton	2 cm	8 cm	4 cm	1 cm
Teddy	2 cm	8 cm	4 cm	1 cm
Objekt	min. Entropie	Eckigkeit	Lokalisation	Rauschen
Karton	0.6	0.15	ja	nein
Teddy	0.6	0.15	ja	nein

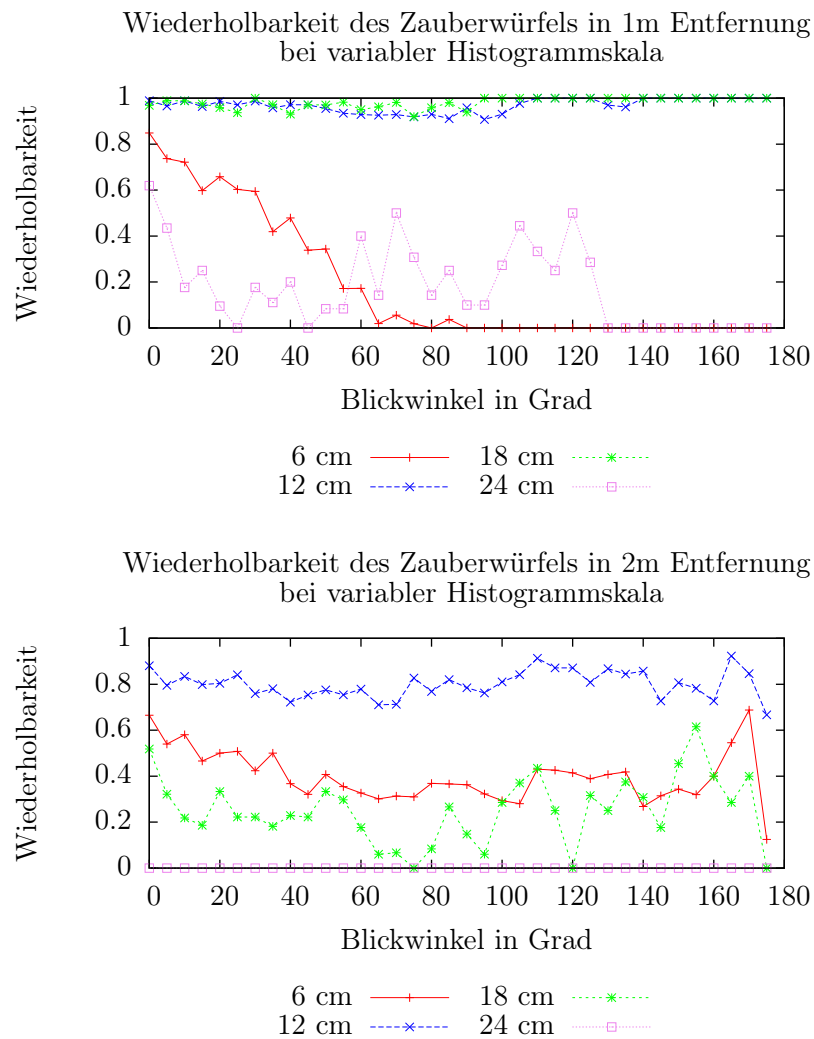


Abbildung 7.20: Wiederholbarkeit des Detektors in der Zauberwürfel-Szene mit verschiedenen Histogrammskalen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
2 cm	variabel	4cm cm	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.5	0.05	ja	nein

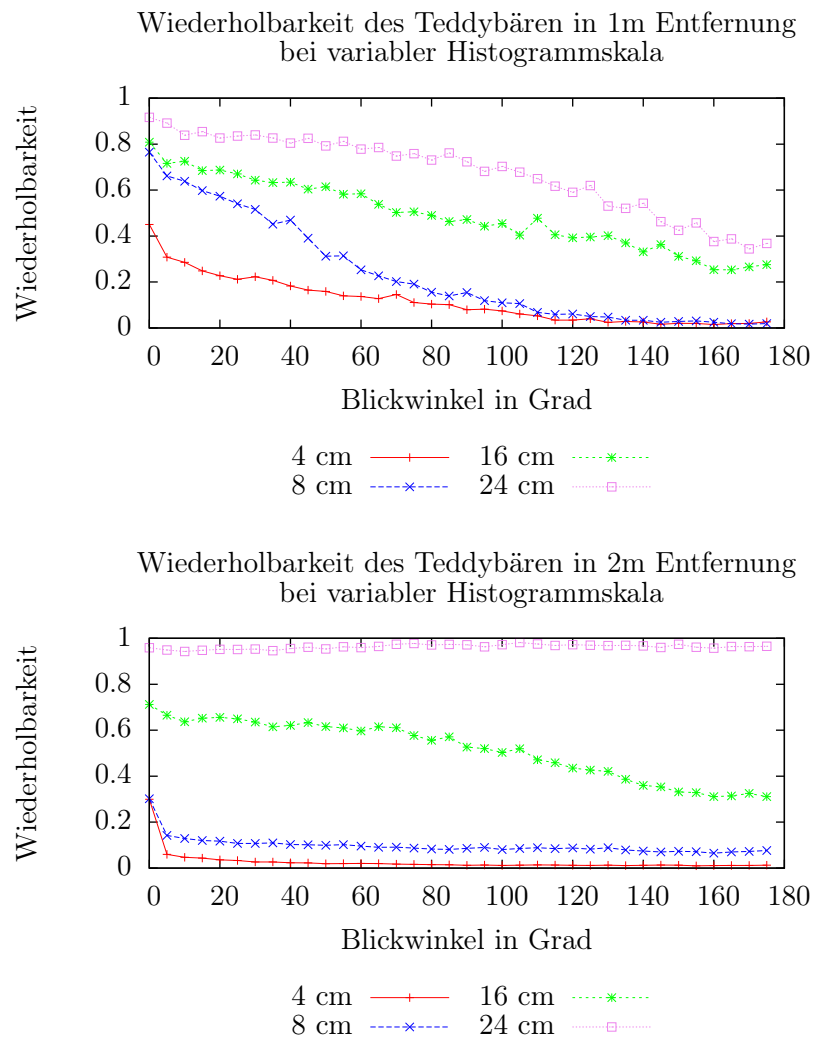


Abbildung 7.21: Wiederholbarkeit des Detektors in der Zauberwürfel-Szene mit verschiedenen Histogrammskalen.

Parameter des Experiments			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
2 cm	variabel	4cm cm	1 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.5	0.05	ja	nein

## 7.3 Experimente zum Vergleich von Vollansichten mit Einzelbildern

Dieses Experiment dient zur Evaluation der Fähigkeiten des Deskriptors, Merkmale desselben Objektes korrekt zuzuordnen. Die Idee ist, die aus Einzelbildern generierte Vollansichten zu nutzen, um auf diesen Merkmale zu generieren. Sollte ein solches Merkmal auch aus einem der Einzelbilder zu sehen sein, so sollte die Distanz zwischen den Deskriptoren klein genug sein, um eine korrekte Zuordnung der Merkmale zu ermöglichen.

### 7.3.1 Erstellung von Vollansichten

Der grundlegende Aufbau erfolgt wie in Kapitel 7.2.1. Aus den registrierten Einzelbildern der erstellten Szenen sind mittels eines Oktalbaums Vollansichten generiert worden. Dazu sind alle Einzelbilder einer Szene in einen Oktalbaum eingefügt worden. Der Oktalbaum ist dabei in der Tiefe begrenzt gewesen, um den Arbeitsspeicherbedarf der Erstellung in Grenzen zu halten. Die minimale Auflösung der erstellten Vollansichten liegt bei 1 cm.

### 7.3.2 Datensätze

Da die bisher verwendeten Szenen nur relativ simple Objekte enthielten, sind zusätzliche Aufnahmen erstellt worden. Diese umfassen zwei Szenen, in denen mehrere Objekte zu sehen waren, darunter der Karton, der Teddy, der Zauberwürfel und der Ball aus Kapitel 7.2.2 sowie eine Kaffeekanne, eine Kaffeetasse und ein Modellauto. Diese wurden an einer Wand platziert. In der ersten Szene wurde die Kinect im Halbkreis um die Szene herumgeführt, siehe Abbildung 7.22(a) bis (d). In der zweiten Aufnahme wurde sie mittig vor der Szene in kurzer Entfernung platziert und dann schrittweise von der Szene wegbewegt, siehe Abbildung 7.22(e) bis (h).

Die dritte Szene umfasst einen Arbeitsplatz mit PC. Die Kinect war auf einem Stativ befestigt und wurde nicht bewegt, nur in kleinen Schritten von Aufnahme zu Aufnahme gedreht, bis zu einer Drehung von etwa  $180^\circ$ . Die Ausrichtung der Szene erfolgte anhand von vier Checkerboards, die so positioniert waren, dass immer mindestens eins zu sehen war und an den Übergängen von einem Checkerboard zum nächsten zwei. Abbildung 7.23 zeigt die Vollansicht des Arbeitsplatzes.

### 7.3.3 Vorgehensweise

Detektion und Merkmalsextraktion werden auf der Vollansicht durchgeführt, die erstellten Merkmale sind im Folgenden die Referenzmerkmale. Dann werden für jedes Einzelbild der Szene Merkmale extrahiert. Aus diesen Merk-

malen werden diejenigen ausgewählt, deren Position mit einem Referenzmerkmal übereinstimmt. Dann wird die Distanz der Deskriptoren zwischen den zugeordneten Merkmalen berechnet. Zusätzlich dazu wird die Distanz zwischen dem Referenzmerkmal und allen im Einzelbild gefundenen Merkmalen berechnet. Für eine korrekte Zurodnung muss die Distanz zwischen dem Referenzmerkmal und dem zugeordneten Merkmal kleiner sein als die Distanzen zwischen dem Referenzmerkmal und den anderen Merkmalen. Ist dies nicht der Fall, so wird ermittelt, wieviele der anderen Merkmale eine kleinere Distanz als das korrekte Merkmal haben. Dieser Rang wird gespeichert. Zusätzlich zur Auswertung des Rangs wird ermittelt, wie hoch die Wiederholbarkeit des Detektors in den Einzelszenen gegenüber den Referenzmerkmalen aus der Vollansicht ist.

### 7.3.4 Experimente

Die folgenden Experimente werden evaluiert:

1. Qualität des Deskriptors bei einer halbkreisförmigen Bewegung um eine Szene, siehe Abbildung 7.24.
2. Qualität des Deskriptors bei schrittweise steigender Entfernung zur Szene, siehe Abbildung 7.25.
3. Qualität des Deskriptors in der Arbeitsplatzszene, siehe Abbildung 7.26.

### 7.3.5 Diskussion der Ergebnisse

#### Diskussion der Vollansicht mit Einzelbildern bei Blickwinkeländerung

Bei der Wiederholbarkeit sieht man, dass Merkmale auf großen Histogrammskalen sehr viel häufiger wiedererkannt werden als solche auf kleinen Histogrammskalen. Dabei muss aber auch berücksichtigt werden, dass in der kleinsten Histogrammskala in der Vollansicht 404 Merkmale detektiert werden, auf der größten Skala aber nur drei. Dabei werden in der kleinsten Skala aber auch viele Merkmale auf Kanten detektiert, die kaum auseinandergehalten werden können, und somit viel häufiger für falsche Zuordnungen beim Vergleichen zweier Merkmale sorgen. Allerdings sieht man auch, dass bei den Histogrammskalen von 12 cm und 24 cm einige der 10 bzw. 37 Merkmale der Vollansicht sichtbar sind und auch im Schnitt die Hälfte korrekt vom Deskriptor korrekt zugeordnet werden.

**Vergleich der Vollansicht mit Einzelbildern bei Entfernungsänderung**

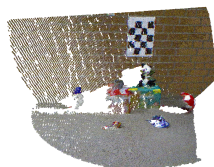
Hier sieht man an der Wiederholbarkeit der Merkmale, dass auf den beiden kleinen Skalen bei steigender Entfernung von der Szene keine Merkmale mehr detektiert werden können. Dagegen werden die beiden großen Skalen mit steigender Entfernung besser, was man bei der Wahl der Skalen berücksichtigen sollte, will man Merkmale auch auf große Entfernungen erkennen. Auch die Anzahl der korrekt zugeordneten Merkmale ist gut.

**Vergleich der Vollansicht mit Einzelbildern in der Arbeitsplatzszene**

Diese Szene ist für die Auswertung schwieriger als die Szene in den vorangegangenen Experimenten. Es sind in jedem Einzelbild nur wenig Merkmale der Vollansicht zu sehen, wobei es von diesen deutlich mehr gibt als in den einfachen Szenen: 22 auf der größten Histogrammskala, 54 auf 24 cm, 120 auf 12 cm und 437 auf 6 cm. Der Deskriptor zeigt akzeptable Ergebnisse wobei deutlich wird, dass die kleinste Skala für solche Zwecke eher schlecht geeignet ist. Der Anstieg der Wiederholbarkeit in den letzten Bildern der Aufnahme liegt daran, dass die Kinect über den Tisch auf der linken Seite schauen kann und dort die Szene bis in eine Entfernung von vier Metern weitergeht.



(a) Vollansicht aufgenommen im Halbkreis



(b) Erstes Bild



(c) Mittleres Bild



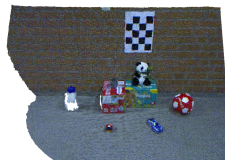
(d) Letztes Bild



(e) Vollansicht aufgenommen aus unterschiedlichen Distanzen



(f) Erstes Bild



(g) Mittleres Bild



(h) Letztes Bild

Abbildung 7.22: Vollansichten und ausgewählte Einzelbilder aus deren Entstehung



(a) Vollansicht von schräg oben gesehen



(b) Vollansicht von der Seite gesehen



Abbildung 7.23: Vollansicht des Arbeitsplatzes und ausgewählte Einzelbilder aus dessen Entstehung



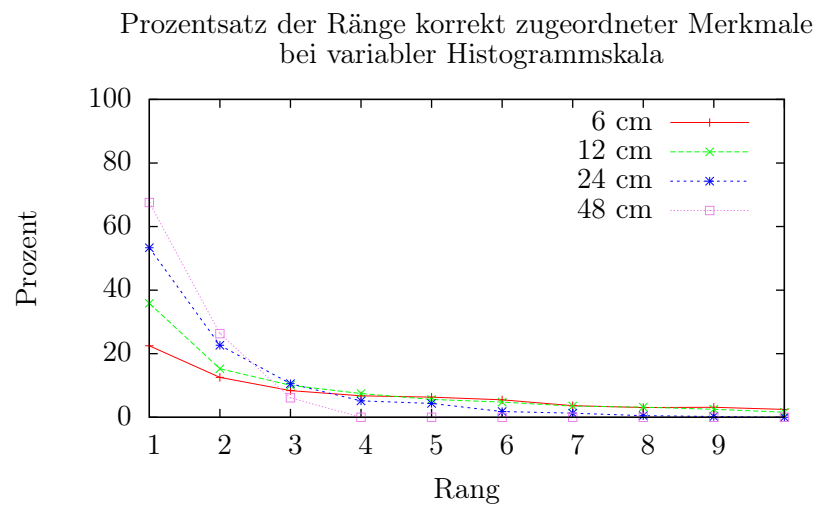
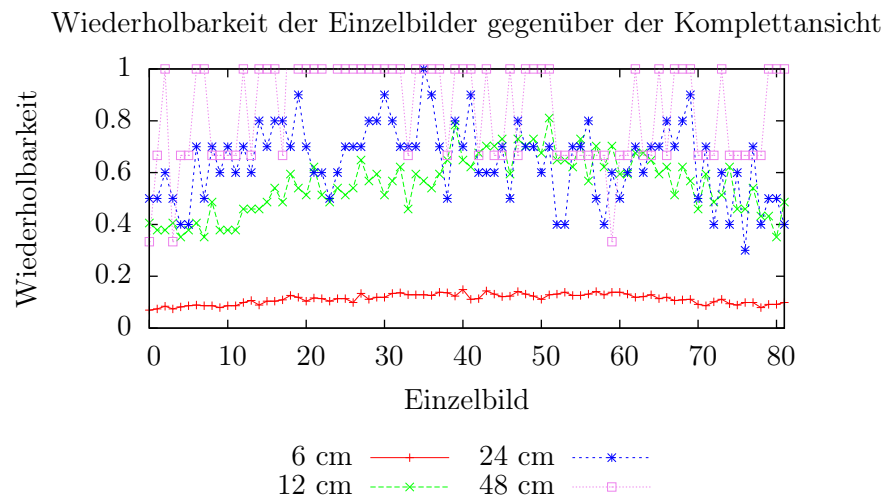


Abbildung 7.24: Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht bei Blickwinkeländerung

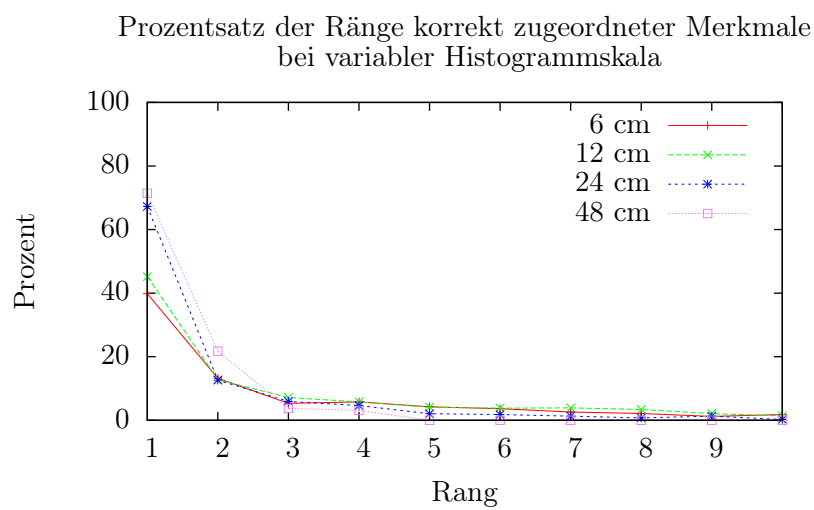
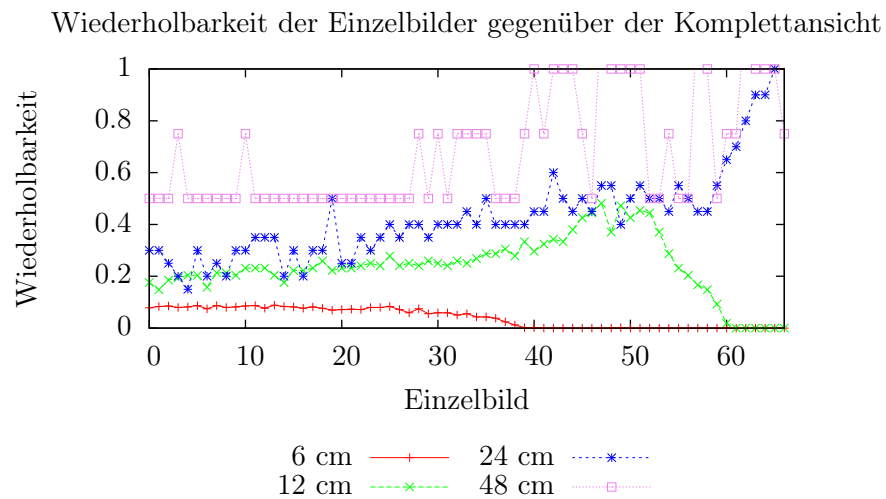


Abbildung 7.25: Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht bei Entfernungsänderung

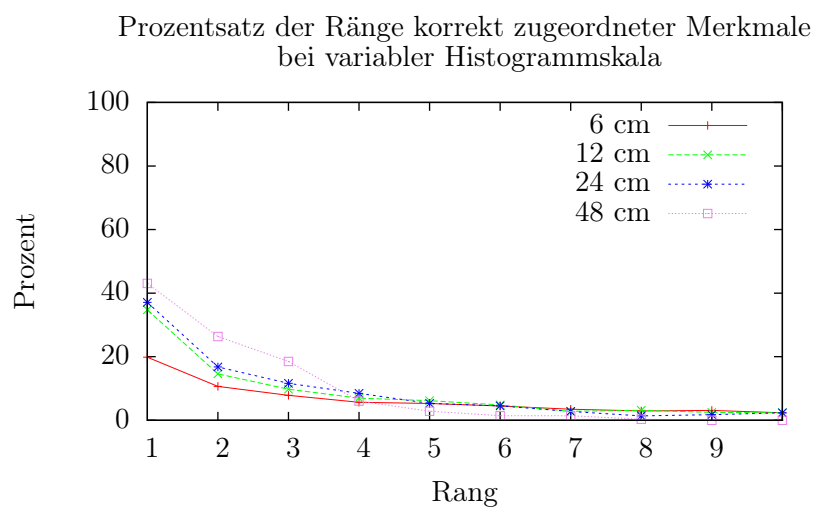
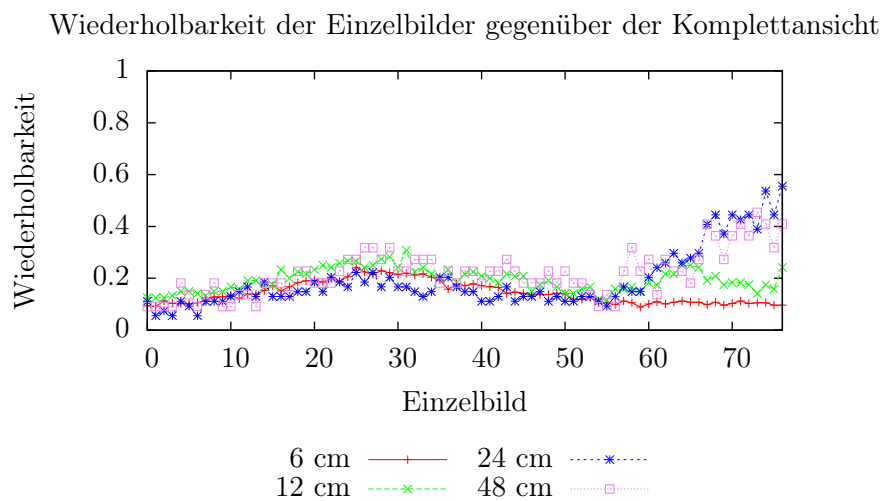


Abbildung 7.26: Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht in der Arbeitsplatzzsene

## 7.4 Vergleich des Detektors mit dem „NARF Key-point Detector“

Zum Abschluss noch ein kurzer Vergleich des in dieser Arbeit vorgestellten Detektors mit dem bereits vorgestellten NARF Detektor aus [SRKB11]. Der Algorithmus von NARF ist in der „Point Cloud Library“ (siehe Kapitel 4.1) implementiert, was einen direkten Vergleich möglich macht. Der NARF-Algorithmus verwendet als Interessantheitsmaß eine Mischung aus der Richtung von Tiefsprungkanten und der Hauptrichtung der lokalen Oberflächenkrümmung. Dies führt zu Detektionen auf Oberflächen, in deren Nähe stark unterschiedliche Krümmungsrichtungen auftreten. Vorteil ist, dass solche Detektionen trotz Abrisskanten gefunden werden, wohingegen der Detektor aus dieser Arbeit Merkmale nur noch selten findet, sobald Oberflächen durch Blickwinkeländerungen aus dem Sichtbereich fallen.

### 7.4.1 Vorgehensweise

Die Vorgehensweise entspricht der aus Kapitel 7.2.2, allerdings wurde aus jeder Szene ein Bild als Referenz ausgewählt und mit den anderen Bildern des Datensatzes verglichen. Die „Support Size“ des NARF-Detektors wurde entsprechend der Histogrammskala gewählt.

Parameter aller Experimente			
Abtastrate	Histogrammskala	Normalenskala	Normalenabtastrate
4 cm	12 cm	4 cm	2 cm
min. Entropie	Eckigkeit	Lokalisation	Rauschen
0.6	0.15	ja	nein

### 7.4.2 Experimente

Der Vergleich ist auf zwei Datensätzen vorgenommen worden, der erste ist der Datensatz des Kartons bei einem Meter Entfernung, der bereits in Kapitel 7.2 verwendet worden ist. Der zweite ist eine Szene mit zwei Schaukelpferden, deren Form und Größe identisch sind, aber sich in Farbe unterscheiden.

Die Ergebnisse und das verwendete Referenzbild sind in den Abbildungen 7.28 und 7.29 zu sehen.

### 7.4.3 Diskussion der Ergebnisse

In der Szene mit dem Karton hat der in dieser Arbeit entwickelte Ansatz auf dem Referenzbild die hervorstehenden Ecken des Kartons detektiert und kann bei leichten Blickwinkeländerungen diese Merkmale sehr gut wiederholt detektieren. Dieses Verhalten ist auch schon in den ersten Experimenten sichtbar, siehe Abbildung 7.19 auf Seite 89 und die Diskussion des Ergebnisses auf Seite 87. Sobald aber eine der Seiten einer Ecke wegfällt, sinkt

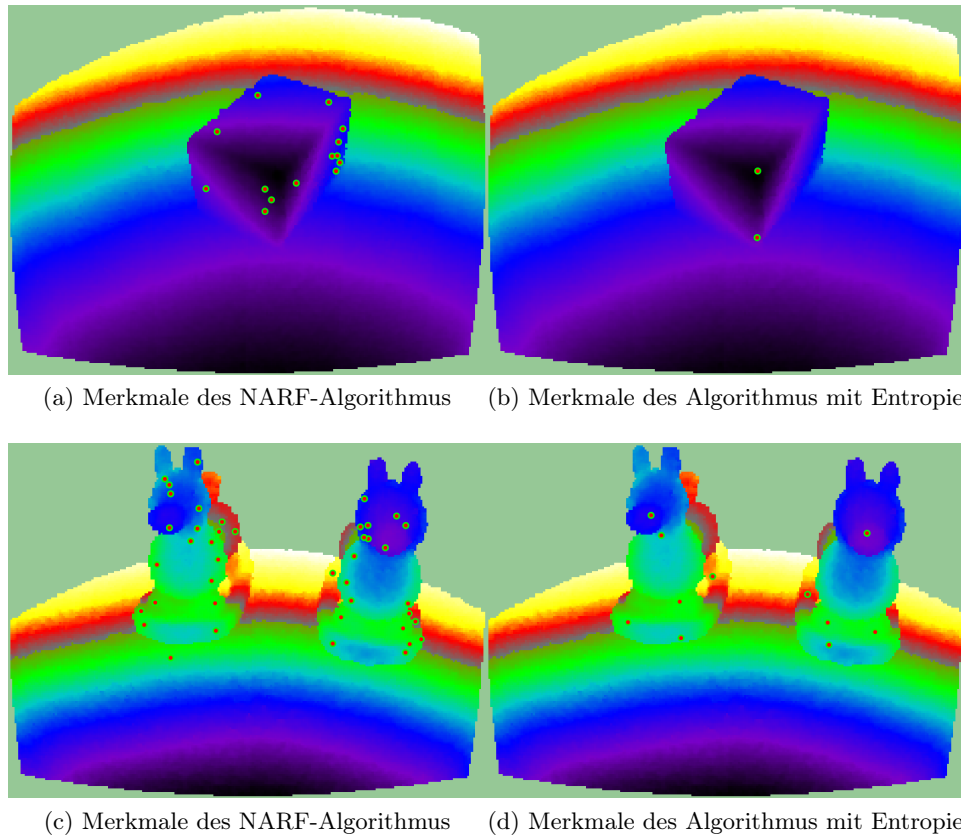


Abbildung 7.27: Die Referenztiefenbilder für NARF und Entropie-Detektor mit entsprechenden Merkmalen

die Wiederholbarkeit sehr schnell. Der NARF-Algorithmus detektiert sehr viele Merkmale auf verschiedenen Oberflächen in der Nähe von Kanten und Tiefsprünge. Auf den Oberflächen, die nicht aufgrund von Blickwinkeländerungen wegfallen, werden diese auch zuverlässig erkannt. Bei der Schaukelpferdszene bestätigt sich die Einschätzung, dass die NARF-Detektion etwas stabiler bezüglich Blickwinkeländerungen sind.



(a) Referenzpunktwolke aus dem Karton-Datensatz

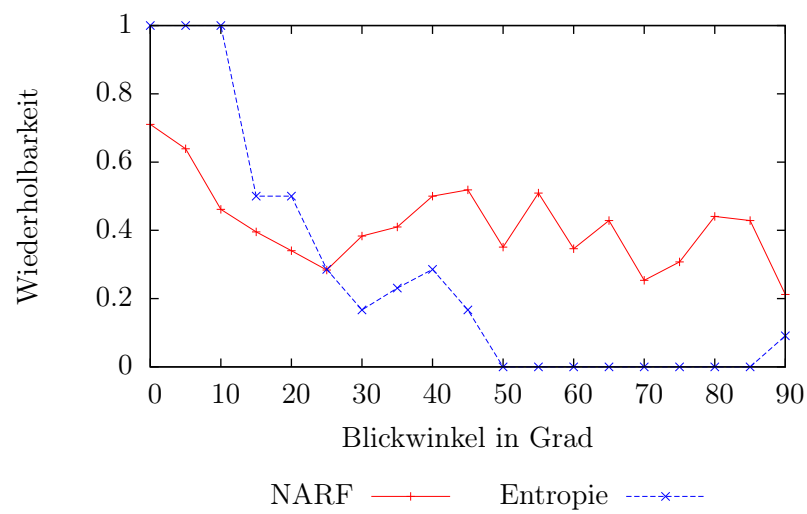


Abbildung 7.28: Vergleich der Wiederholbarkeit des NARF-Algorithmus mit dieser Arbeit in einer Szene mit einem Karton.



(a) Referenzpunktwolke aus dem Schaukelpferd-Datensatz

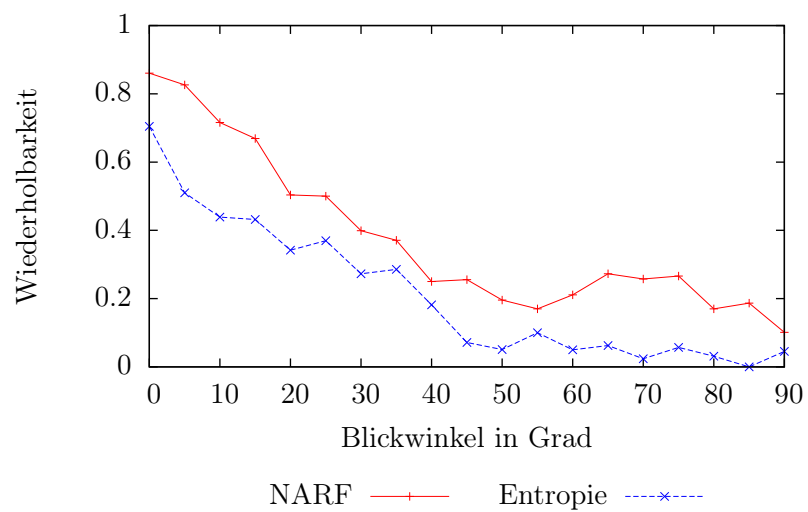


Abbildung 7.29: Vergleich der Wiederholbarkeit des NARF-Algorithmus mit dieser Arbeit in einer Szene mit zwei Schaukelpferden.

## 7.5 Laufzeitverhalten

In diesem Abschnitt wird die Laufzeit der Bestandteile des Verfahrens evaluiert. Dazu wurde der Datensatz der Arbeitsplatzszene als Referenz benutzt. Die Szene wird deshalb gewählt, weil sie Einzelbilder mit unterschiedlicher Komplexität und maximaler Sichtweite enthält. Dies sollte einen Überblick geben, wie effizient das Verfahren bei unterschiedlichen Bedingungen ist. Das Verfahren durchläuft die komplette Szene von 76 Einzelbildern und die Laufzeit der einzelnen Komponenten wird gemessen. Anschließend werden aus den Zeiten Mittelwert und Standardabweichung berechnet. Als System kommt ein Intel Pentium Core 2 Duo P7350 mit 2 GHz Taktfrequenz und 3 Gigabyte Arbeitsspeicher zum Einsatz.

Die Zeiten für die Entropieberechnung und die Merkmalsextraktion wurden für zwei verschiedenen Abtastraten gemessen, einmal 4 cm und einmal 8 cm. Die anderen Parameter sind entsprechend der Tabelle 7.1.4 auf Seite 68 angepasst worden. Die durchschnittliche Anzahl der Merkmale liegt bei 31 für 4cm und 14 für 8 cm. Die Zeiten für die verbesserte Lokalisierung wurden für eine Abtastrate von 4 cm gemessen.

	Mittelwert	Standardabweichung
Aufbau des Oktalbaums	0.51 s	0.06 s
Entropieberechnung 4 cm	0.34 s	0.12 s
Entropieberechnung 8 cm	0.15 s	0.06 s
Merkmalsextraktion 4 cm	0.02 s	0.01 s
Merkmalsextraktion 8 cm	0.02 s	0.01 s
Lokalisierung per Mean-Shift	0.04 s	0.01 s
Lokalisierung per Subsampling	0.21 s	0.06 s

### 7.5.1 Diskussion der Laufzeit

Dominierend ist die Zeit, die der Aufbau des Oktalbaums benötigt. Die Zeit für die Entropieberechnung ist sehr stark abhängig von den gewählten Parametern, während die Merkmalsextraktion davon kaum beeinflusst wird. Bei der verbesserten Lokalisierung ist der Ansatz mittels Mean-Shift dem Subsampling deutlich überlegen, wobei das Subsampling bei kleinen Abtastraten problematisch ist, weil es dann sehr durch Sensorrauschen beeinflusst werden kann.



## Kapitel 8

# Zusammenfassung

Die vorliegende Arbeit präsentiert eine Methode zur Detektion und Deskription interessanter Regionen in dreidimensionalen Punktwolken. Als interessant werden solche Regionen bezeichnet, die sich durch eine komplexe Oberflächenstruktur auszeichnen. Als Maß für die Interessantheit einer Region wird die Shannon-Entropie verwendet. Mit einem Oktalbaum wird eine effiziente Datenstruktur zur Verwaltung und Diskretisierung der Punktwolke benutzt. Im Oktalbaum werden Oberflächennormalen der Punktwolke geschätzt und diskretisiert. Interessante Regionen werden anhand der Shannon-Entropie der vorkommenden Oberflächennormalen gewählt, je unterschiedlicher die Normalen in einer Region desto höher ist die Entropie. Aus den detektierten Regionen werden die instabilen aussortiert, auf den verbleibenden wird ein Deskriptor erstellt. Dieser Deskriptor beinhaltet eine Form-, eine Farb- und eine Helligkeitskomponente. Der Algorithmus ist in der Lage, Einflüsse durch Sensorrauschen zu behandeln und die Zahl instabiler Merkmale zu reduzieren. Die Komponenten des Deskriptors sind gegenüber Blickwinkeländerungen, Sensorrauschen und Helligkeitsänderungen invariant. In der Arbeit werden anschließend sowohl der Detektor als auch der Deskriptor evaluiert.

## Diskussion und Ausblick

Die Idee der Arbeit beruht auf der Annahme, dass Punktwolken gewöhnlicher Szenen viele gleichförmige Oberflächen vom Boden, Wänden oder der Decke sowie solche von Möbelstücken wie Schränken oder Tischen enthalten. Eine Region mit vielen unterschiedlichen Oberflächen hingegen sticht aus diesen Strukturen heraus und es sollte sowohl möglich sein, solche Regionen zu identifizieren als auch zu beschreiben. Solche Ansätze sind in Bilddaten schon erfolgreich umgesetzt worden, in Tiefenbildern und Punktwolken hingegen noch nicht, was die Neuartigkeit dieser Arbeit ausmacht. Die Entscheidung, Punktwolken zu verwenden, wurde getroffen, weil Punkt-

wolken explizit Informationen zur Größe eines Objektes enthalten. Tiefenbilder enthalten diese Information zwar auch, aber nur implizit; betrachtet man feste Größen von Pixelnachbarschaften im Tiefenbild, so ist die Größe eines Objektes durch den Aufnahmeprozess vom Bildbereich und der Distanz zum Objekt abhängig. Die Skala des Merkmals variiert dadurch unter Blickposenänderungen.

Mit dem Oktalbaum ist gezeigt worden, wie Punktwolken effizient und effektiv auf spezielle Eigenschaften untersucht werden können. Allerdings ist der Aufbau des Baums sehr zeitintensiv. Hier besteht noch Potenzial für Verbesserungen. Eine davon ist die Parallelisierung des Aufbaus in einer GPU, wie in [ZGHG11] beschrieben.

Bei der Detektion gibt es einige Punkte, die einer Diskussion bedürfen. Der erste ist die Wahl der Skala. Diese entspricht nicht dem Skalenbegriff aus der Bildverarbeitung, sondern beschreibt die Größe der Regionen, in denen die Entropie berechnet wird. Es wurde gezeigt, dass die Wahl der Skala wichtig für eine korrekte Detektion ist. Optimal wäre daher eine Skalenselektion, die automatisch auswählt, auf welcher Skala eine Region das beste Ergebnis liefert. Das beste Ergebnis ist hier allerdings nicht unbedingt ein Maximum in der Entropie, sondern das Ergebnis, dass eine spätere Beschreibung des Objektes möglichst distinktiv werden lässt. Stehen zum Beispiel mehrere Objekte in kurzer Entfernung nebeneinander, so wäre ein Maximum der Entropie meist in einem Bereich, der alle diese Objekte und ihre Oberflächen erfasst. Dies würde allerdings eine Wiedererkennung eines der Objekte nahezu unmöglich machen.

Der nächste wichtige Punkt der Detektion ist die Stabilität der Detektionen. Es ist leider nicht gelungen, instabile Detektionen auf ein verwertbares Maß zu begrenzen, Merkmale auf Kanten sind hierbei das größte Problem. Das entwickelte Konzept der Eckigkeit kann die Zahl zwar reduzieren, ist aber nur schwer regulierbar. Ist der Wert zu hoch, werden auch stabile Merkmale verworfen, ist er zu niedrig, sinkt die Wiederholbarkeit der Detektionen. Diese Regulation müsste zur Laufzeit des Verfahrens erfolgen und ist stark von der Größe und Form der gewünschten Merkmalen abhängig.

Ein weiterer Punkt ist die Lokalisation der Detektionen. Die starke Diskretisierung der Daten zur effizienten Berechnung hat ihren Preis in Form von Detektionen, die die eigentlichen Maxima der Entropie teilweise sehr deutlich verfehlen. Dies ist sowohl der Wiederholbarkeit der Detektion als auch der Wiedererkennbarkeit eines Merkmals abträglich. Hier ist es gelungen, mit dem Mean-Shift-Algorithmus ein Verfahren zu implementieren, dass effizient und effektiv in der Lage ist, die Lokalisierung der Detektionen stark zu verbessern. Selbst bei ungünstigen Umständen sind Wiederholbarkeit und Wiedererkennbarkeit akzeptabel.

Der letzte Punkt zur Detektion ist die Laufzeit. Die Detektion hat nach dem Aufbau des Oktalbaums den größten Einfluss auf das Laufzeitverhalten, hier wären Verbesserungen auch wünschenswert. Solche Verbesserungen könnten

durch die Anpassung des Verfahrens auf Tiefenbilder erzielt werden. Die Nutzung von Integralbildern würde die Normalenberechnung beschleunigen und durch Nachbarschaften im Tiefenbild entfielen die Bereichsanfragen im dreidimensionalen. Hier würde dann aber ein Mechanismus zur Skalenselektion erforderlich.

Mögliche Anwendungen des Detektors sind unter anderem die Posenschätzung von Objekten oder die Wiedererkennung von Orten durch einen Bag-of-Words-Ansatz. Zur Beschreibung der interessanten Regionen ist ein Deskriptor entwickelt worden, der mehrere Aspekte einer Region beschreibt. Der erste davon ist die Beschreibung der Form. Ziel ist die Beschreibung von durch den Detektor ermittelten Regionen, die eine variierende Oberflächenstruktur besitzen und man annehmen kann, dass diese auch ein gewisses Maß an Distinktivität besitzen. Zudem sollte der Deskriptor möglichst invariant gegenüber Posenänderungen des Betrachters sein. Dies ist durch Implementation der „Point Feature Histograms“ gelungen, die effizient distinktive Beschreibungen der Form erzeugen. Hervorzuheben ist hier zudem, dass die Berechnung der Daten auf bereits vorhandenen Daten wie den Normalen, die auch zur Entropieberechnung verwendet werden, beruht. Allerdings hat die Distinktivität der Regionen Grenzen, einerseits durch viele selbstähnliche Objekte wie Ecken, deren Deskription über verschiedene Skalen identisch ist, als auch durch äußere Effekte wie Sensorrauschen.

Der zweite beschriebene Aspekt eines Objektes ist die Farbe, die zusätzlich zur Form Aufschluss über das Aussehen einer Region gibt. Der entwickelte Farbdeskriptor ist invariant gegenüber Blickwinkel, Beleuchtung und Sensorrauschen. Die Struktur des Deskriptors ist dabei simpel gehalten und die Berechnung ist effizient durchführbar. Auch hier werden die benötigten Daten im Oktalbaum bereitgestellt und müssen nicht neu erhoben werden. Der letzte Aspekt ist die Beschreibung der Helligkeitsverläufe in den interessanten Regionen, die als Ergänzung des Farbdeskriptors gedacht ist. Der Helligkeitsdeskriptor konnte leider nicht mehr intensiv ausgewertet werden. Der erstellte Deskriptor funktioniert unabhängig vom Detektor und ist daher auch für andere Arbeiten nutzbar. Als Anwendungen sind hier die Objektwiedererkennung einmal gesehener Objekte durch ein Lernverfahren oder die Zuordnung von Vollansichten von Objekten aus einer Datenbank zu Objekten im aktuellen Sichtbereich des Sensors.

# Abbildungsverzeichnis

3.2	Time-of-Flight Kamera. <i>Quelle: Wikimedia Commons, Autor: Captaindistance, Lizenz CC-BY-SA 3.0</i> . . . . .	8
3.1	Aufnahme eines Teddybären aus verschiedenen Entfernungen. . . . .	9
3.3	Microsoft Kinect. <i>Quelle: Wikimedia Commons, Autor: Evan-Amos, Lizenz: CC0 1.0</i> . . . . .	10
3.4	Technische Daten der Kinect. <i>Quellen: <a href="http://www.primesense.com/?p=514">http://www.primesense.com/?p=514</a> und <a href="http://openkinect.org/wiki/Hardware_info">http://openkinect.org/wiki/Hardware_info</a>, zuletzt abgerufen am 24.08.11, 14:00 MEST</i> . . . . .	11
3.5	Entwicklung der Entropie. <i>Quelle: [Sha48], S.11)</i> . . . . .	12
3.6	Schema eines Oktalbaums. <i>Quelle: Wikimedia Commons: Autor: WhiteTimberwolf / Nü, Lizenz: CC-BY-SA 3.0</i> . . . . .	14
3.7	Eigenvektoren in 2D. <i>Quelle: Wikimedia Commons, Autor: Ben FrantzDale, Lizenz: CC-BY-SA 3.0</i> . . . . .	17
4.1	Diskretisierung einer Punktwolke mittels Oktalbaum . . . . .	25
4.2	Klassen des Normalenhistogramms. . . . .	30
5.1	Entropie auf verschiedenen Abtastraten . . . . .	32
5.2	Merkmale des Kartons auf verschiedenen Skalen. . . . .	34
5.3	Normalenschätzung auf verschiedenen Normalenabtastraten . . . . .	35
5.4	Schätzung der Oberflächennormalen in verschiedenen Normalenskalen . . . . .	36
5.5	Instabile Normalen . . . . .	38
5.6	Die ebenenweise Diskretisierung der Punktwolke in der Tiefe erfordert die Anpassung der Normalenskala zur Schätzung korrekter Normalen. . . . .	39
5.7	Ablauf der Detektion . . . . .	41
5.8	Merkmale an Kanten und sich neigende Normalen. . . . .	42
5.9	Eckigkeit auf einer Punktwolke eines Kartons . . . . .	44
5.10	Lokalisation der Merkmale . . . . .	46
6.1	Manuell platzierte Merkmale auf einem Würfel . . . . .	51
6.2	Beispiel für die $\alpha$ -Komponente des Formdeskriptors . . . . .	52

6.3	Beispiel für die $\beta$ -Komponente des Formdeskriptors . . . . .	53
6.4	Beispiel für die $\gamma$ -Komponente des Formdeskriptors . . . . .	54
6.5	Beispiel für den Farbdeskriptor . . . . .	55
6.6	Vergleich von Distanzen auf dem Formdeskriptor . . . . .	56
6.7	Distanzmatrix des Farbhistogramms . . . . .	59
6.8	Distanz zwischen verschiedenen Farben im Farbdeskriptor . .	59
6.9	Vergleich von Distanzen auf dem Farbdeskriptor . . . . .	60
6.10	Vergleich von Distanzen auf dem Helligkeitsdeskriptor . . . .	62
7.1	Punktwolken von Würfeln . . . . .	64
7.2	Vier unterschiedlich affin-transformierte Würfel . . . . .	65
7.3	Positionen der Merkmale bei 250 Detektionen im Testwürfel .	66
7.4	Durchschnittsdistanzen der Deskriptoren unter den Ecken ei- nes Testwürfels . . . . .	70
7.5	Einfluss der Abtastrate auf die Wiederholpräzision der Merk- male . . . . .	71
7.6	Durschnittsdistanzen der Deskriptoren bei variabler Abtastrate	72
7.7	Einfluss der Normalenabtastrate auf die Wiederholpräzision der Merkmale . . . . .	73
7.8	Durchschnittsdistanzen der Deskriptoren bei variabler Nor- malenabtastrate . . . . .	74
7.9	Einfluss der Normalenskala auf die Wiederholpräzision der Merkmale . . . . .	75
7.10	Durchschnittsdistanzen der Deskriptoren bei variabler Nor- malenskala . . . . .	76
7.11	Durchschnittsdistanzen der Deskriptoren bei variabler Histo- grammskala . . . . .	77
7.12	Einfluss von Rauschen auf die Wiederholpräzision der Merkmale	78
7.13	Durchschnittsdistanzen der Deskriptoren unter Einfluss von Rauschen . . . . .	79
7.14	Durchschnittsdistanzen der Deskriptoren unter Einfluss von Rauschen und Lokalisation . . . . .	80
7.15	Einfluss der Eckigkeit auf den Detektor . . . . .	81
7.16	Einfluss des Entropieschwellwertes auf den Detektor . . . . .	82
7.17	Graustufenbilder der Experimente . . . . .	85
7.18	Wiederholbarkeit des Detektors . . . . .	88
7.19	Wiederholbarkeit des Detektors . . . . .	89
7.20	Wiederholbarkeit des Detektors in der Zauberwürfel-Szene mit verschiedenen Histogrammskalen. . . . .	90
7.21	Wiederholbarkeit des Detektors in der Zauberwürfel-Szene mit verschiedenen Histogrammskalen. . . . .	91
7.22	Vollansichten und ausgewählte Einzelbilder aus deren Entste- hung . . . . .	95

7.23	Vollansicht des Arbeitsplatzes und ausgewählte Einzelbilder aus dessen Entstehung . . . . .	96
7.24	Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht bei Blickwinkeländerung . . . . .	97
7.25	Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht bei Entfernungsänderung . . . . .	98
7.26	Wiederholbarkeit und Auswertung der Ränge der Deskriptoren im Vergleich von Einzelbildern zur Vollansicht in der Arbeitsplatzszene . . . . .	99
7.27	Die Referenztiefenbilder für NARF und Entropie-Detektor mit entsprechenden Merkmalen . . . . .	101
7.28	Vergleich der Detektoren . . . . .	102
7.29	Vergleich der Detektoren . . . . .	103

# Literaturverzeichnis

- [dBvKOS00] Mark de Berg, Marc van Krefeld, Mark Overmars, and Otfried Schwarzkopf. *Computational Geometry - Algorithms and Applications*. Springer-Verlag, 2000.
- [FH75] Keinosuke Fukunaga and Larry D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *Transactions on Information Theory (IEEE)*, 21(1):32–40, 1975.
- [HLW09] Michael Himmelsbach, Thorsten Luettel, and Hans-Joachim Wuensche. Real-time object classification in 3d point clouds using point feature histograms. In *International Conference on Intelligent Robots and Systems*, 2009.
- [JH99] Andrew E. Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21:433–449, May 1999.
- [KB01] Timor Kadir and Michael Brady. Saliency, scale and image description. *International Journal of Computer Vision*, Volume 45 Issue 2:83–105, 2001.
- [KZB04] Timor Kadir, Andrew Zisserman, and Michael Brady. An affine invariant salient region detector. In *8th European Conference on Computer Vision*, pages 228–241. Springer-Verlag, 2004.
- [Low99] David G. Lowe. Object recognition from local scale-invariant features. In *Seventh International Conference of Computer Vision*, pages 1150–1157, 1999.
- [MS04] Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision No. 60*, 1:63–86, 2004.
- [NN07] John Novatnack and Ko Nishino. Scale-dependent 3d geometric features. In *International Conference of Computer Vision*, 2007.

- [NNS06] John Novatnack, Ko Nishino, and Ali Shokoufandeh. Extracting 3d shape features in discrete scale-space. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 946–953, June 2006.
- [PKG03] Mark Pauly, Richard Keiser, and Markus Gross. Multi-scale feature extraction on point-sampled surfaces. In *Eurographics 03*, 2003.
- [PW08] Ofir Pele and Michael Werman. A linear time histogram metric for improved sift matching. In *ECCV*, 2008.
- [PW09] Ofir Pele and Michael Werman. Fast and robust earth mover’s distances. In *ICCV*, 2009.
- [RBB09] Rusu, Blodow, and Beetz. Fast point feature histograms (fpfh) for 3d registration. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2009.
- [RBTH10] Radu Bogdan Rusu, Gary Bradski, Romain Thibaux, and John Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *Proceedings of the 23rd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, 2010.
- [RC11] Radu Bogdan Rusu and Steve Cousins. 3d is here: Point cloud library (pcl). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.
- [RHBB09] Rusu, Holzbach, Blodow, and Beetz. Fast geometric point labeling using conditional random fields. In *In Proceedings of the 22nd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2009.
- [RMBB08a] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, and Michael Beetz. Learning informative point classes for the acquisition of object model maps. In *Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2008.
- [RMBB08b] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, and Michael Beetz. Persistent point feature histograms for 3d point clouds. In *Proceedings of the 10th International Conference on Intelligent Autonomous Systems (IAS-10)*, 2008.
- [RTG00] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover’s distance as a metric for image retrieval. *International Journal of Computer Vision*, 40:99–121, 2000.



- [Sam06] Hanan Samet. *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann Publishers, 2006.
- [Sha48] Claude Elwood Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
- [Sha06] Tahir Rabbani Shah. Automatic reconstruction of industrial installations using point clouds and images. *Publications on Geodesy*, 62, 2006.
- [SRKB10] Bastian Steder, Radu Bogdan Rusu, Kurt Konolige, and Wolfram Burgard. Narf: 3d range image features for object recognition. In *International Conference on Intelligent Robots and Systems*, 2010.
- [SRKB11] Bastian Steder, Radu Bogdan Rusu, Kurt Konolige, and Wolfram Burgard. Point feature extraction on 3D range scans taking into account object boundaries. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2011.
- [UH08] Ranjith Unnikrishnan and Martial Hebert. Multi-scale interest regions from unorganized point clouds. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.
- [vdSGS10] Koen E. A. van de Sande, Theo Gevers, and Cees G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.
- [WHH03] Eric Wahl, Ulrich Hillenbrand, and Gerd Hirzinger. Surflet-pair-relation histograms: A statistical 3d-shape representation for rapid classification. In *Proceedings of Forth International Conference on 3-D Digital Imaging and Modeling (3DIM 2003)*, 2003.
- [ZGHG11] Kun Zhou, Minmin Gong, Xin Huang, and Baining Guo. Data-parallel octrees for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 17(5):669–681, 2011.