

Automatische bildbasierte Segmentierung organischer Objekte einer gleichartigen Gruppe

Abgeleitet vom Problem der Stammschnittflächensegmentierung

Dissertation

zur

Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik

der Universität Rostock



eingereicht am 24. Juni 2014 in Rostock

von Enrico Gutzeit

geboren am 18. April 1979 in Teterow

wohnhaft in Rostock

Verteidigung:

12. Dezember 2014 in Rostock

Gutachter:

Prof. Dr.-Ing. Bodo Urban, Universität Rostock

Prof. Dr.-Ing. Reinhard Koch, Universität Kiel

PD Dr. habil. Ir. Arjan Kuijper, Technische Universität Darmstadt

Schlagwörter:

- Segmentierung, Objektdetektion, Maschinelles Sehen, Bildverarbeitung
- Organische Objekte, Stammschnittflächen, Holzpolter, Vermessung

Keywords:

- segmentation, object detection, computer vision, image processing
- organic objects, stack of wood, wood cut surfaces, measuring

CR-Classification:

- I.4.6: Segmentation (Image Processing and Computer Vision)
- I.4.9: Applications (Image Processing and Computer Vision)
- I.2.10: Vision and Scene Understanding (Artificial Intelligence)
- I.5.1: Models (Pattern Recognition)

Zusammenfassung

Diese Arbeit adressiert die automatische Segmentierung von **organo**-Gruppen, Gruppen gleichartiger organischer Objekte. Der Fokus wird auf das ähnlich schwere Problem der Stammschnittflächensegmentierung gelegt. Die Segmentierung von multiplen Objekten auf unbekanntem Hintergrund in Einzelbildern ist ein schweres und ungelöstes Problem. Es existieren einige spezielle Ansätze für spezifische Objekte, wie Tomaten, Äpfel oder Fische. Dennoch existiert keine robuste und automatische Methode für Stammschnittflächen oder ein allgemeingültiger Ansatz für **organo**-Gruppen. Die Segmentierung einer **organo**-Gruppe ermöglicht Anwendungen zur automatischen Vermessung, Inspektion oder Sortierung.

In dieser Arbeit werden, ausgehend vom Problem der Stammschnittflächen, drei Konzepte zur automatischen Segmentierung entwickelt und quantitativ evaluiert. Die Konzepte bauen aufeinander auf und ermöglichen die automatische Segmentierung von Schnittholz- oder Stammschnittflächen in Bildern mit unbekanntem Hintergrund. Ausgehend von den Konzepten wird eine allgemeinere Lösung für **organo**-Gruppen entwickelt und am Beispiel von Plattfischen, Kartoffeln und Äpfeln evaluiert. In den einzelnen Evaluierungen der Konzepte werden gute bis sehr gute Ergebnisse erreicht. Die Ergebnisse sind in ein praxistaugliches System zur Vermessung von Holzpoltern eingeflossen. Die Evaluierung des Systems zeigt, dass die Konzepte zur Anwendung geeignet sind, wenn gewisse Rahmenbedingungen eingehalten werden.

abstract

This thesis addresses the automatic segmentation of so-called **organo**-groups (groups of organic objects of similar types). It focuses on a similarly hard problem, namely the segmentation of wood cut surfaces. The segmentation of multiple objects on an unknown background in single images is a challenging task. Existing approaches so far are rather special, and are designed for specific objects like tomatoes, apples or fish. Additionally, there is no robust and automatic approach for the segmentation of wood cut surfaces and also none of them is general enough for **organo**-groups. The segmentation of **organo**-groups sets the ground for automatic measurement, inspection or sorting.

Based on the problem of wood cut surfaces three concepts have been developed and quantitatively evaluated. The concepts build upon each other and allow the binary segmentation of sawn wood and multi-object-segmentation of wood cut surfaces in images on an unknown background. Based on the three concepts a more general solution has been developed and evaluated using the examples of fish, potatoes and apples. Good to very good results are achieved in the evaluations of the concepts. These results are integrated into a practical measuring system for woodpiles. An evaluation of the system has shown that the concepts are practically applicable under certain frame conditions.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung und Problemstellung	3
1.3	Organisation der Arbeit	6
1.4	Zusammenfassung der wichtigsten Ergebnisse	7
2	Stand der Forschung zur Segmentierung	9
2.1	Begriffsdefinition	9
2.2	Vorbetrachtung der Segmentierung und Objektdetektion	16
2.2.1	Abgrenzung und Einordnung	17
2.2.2	Zusammenhang und Lösungsraum	18
2.2.3	Fazit und Erarbeitung einer Grobstrukturierung	21
2.3	Basismethoden der Segmentierung und Objektdetektion	21
2.3.1	Klassifikation der Basismethoden	22
2.3.2	Basismethoden der Segmentierung	23
2.3.3	Basismethoden der Objektdetektion	34
2.3.4	Zusammenfassung der Basismethoden	37
2.4	Spezielle Methoden am Beispiel von organo -Gruppen	38
2.4.1	Segmentierung von Stammschnittflächen	38
2.4.2	Segmentierung von Äpfeln und Tomaten	40
2.4.3	Segmentierung von Kiwis, Litschis und anderen Früchten	44
2.4.4	Segmentierung von Fischen, Plankton und Muscheln	45
2.5	Basismethoden versus Segmentierungsaufgabe	47
3	Automatische Segmentierung von Stammschnittflächen und Schnittholz	51
3.1	Problemdiskussion und Herleitung von Konzepten	51
3.1.1	Problemdiskussion	52
3.1.2	Voruntersuchung von Basismethoden der Segmentierung	55
3.1.3	Diskussion und Herleitung verschiedener Konzepte	59
3.2	Schnittholzsegmentierung durch Graph-Cut und Holzfarbe	66
3.2.1	Konzeption der Schnittholzsegmentierung	66
3.2.2	Holzfarbsegmentierung zur Vorder- und Hintergrundabschätzung	68
3.2.3	Erstellung des gewichteten Graphen und Graph-Cut	74
3.2.4	Zusammenfassung und Ergebnisse	81
3.3	Stammschnittflächensegmentierung mit Objektdetektion	84
3.3.1	Konzeption der Segmentierung mit Objektdetektion	84

3.3.2	Wahl des Objektdetektors und Detektion	86
3.3.3	Verifikation der Bildobjekte und Segmentierung	91
3.3.4	Zusammenfassung und Ergebnisse	96
3.4	Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion	98
3.4.1	Konzeption einer kombinierten Segmentierung	98
3.4.2	Ermittlung der Trimap und Graph-Cut	100
3.4.3	Separierung der Stammschnittflächen	103
3.4.4	Zusammenfassung und Ergebnisse	107
3.5	Evaluierung und Diskussion der Konzepte	109
4	Generalisierung für organische Objekte einer gleichartigen Gruppe	111
4.1	Problemdiskussion	111
4.2	Ableitung eines allgemeingültigen Konzeptes	113
4.3	Generalisierung der Objektdetektion und Verifikation	115
4.4	Generalisierung der binären Segmentierung	118
4.5	Generalisierung der Objektseparierung	122
4.6	Zusammenfassung, Evaluierung und Grenzen	124
5	Anwendung und Ergebnisse der Holzpoltervermessung	127
5.1	Forstwirtschaftliche Begriffe	127
5.2	Das Holzpolter-Vermessungssystem AFoRS	128
5.3	Evaluierung von AFoRS und Rückschluss	131
6	Zusammenfassung und Ausblick	135
6.1	Zusammenfassung	135
6.2	Ausblick	137
A	Grundlagen und Ergänzendes zum Maschinellen Sehen	141
A.1	Morphologische Operationen und Schwerpunkt	141
A.2	Aktive Konturen und die Energieterme	142
A.3	Klassifikationen von Segmentierungsmethoden	143
A.4	Evaluierungsmaße der Segmentierung	144
B	Segmentierungsergebnisbilder	147
B.1	Ergebnisbilder mit Otsu’s Methode, Sobel und Canny	147
B.2	Ergebnisbilder regionenbasierter Basismethoden	148
B.3	Ergebnisbilder der verschiedenen Konzepte	149
C	Visualisierung und Evaluierungsdaten	155
C.1	Farbraumvisualisierung von Schnittholz	155
C.2	Evaluierungsbilder und deren Grundwahrheit	157
C.3	Praxisdaten AFoRS	160
D	Entwickelte Algorithmen	165
D.1	GMC -Algorithmus	165

D.2 LSGMC-Algorithmus	166
D.3 GMC-G-Algorithmus	167
Abkürzungs- und Symbolverzeichnis	169
Literaturverzeichnis	173
Eigene Veröffentlichungen	191
Danksagung	195
Selbstständigkeitserklärung	197
Thesen	199

1 Einleitung

Die automatische Segmentierung von Objekten ist ein häufig adressiertes Thema und stellt Wissenschaftler immer noch vor große Herausforderungen. Was der Mensch in einem Bild auf Anhieb erkennt, ist für einen Algorithmus nur unter gewissen Annahmen und Bedingungen zu fassen. Die kognitiven Fähigkeiten eines Menschen prägten sich erst über Millionen von Jahren der Evolution aus. Ermöglicht wird sie durch ein Netzwerk von vererbten Neuronen, die sich innerhalb der Entwicklung des Menschen verbinden und ausprägen. Auf solch einer Grundlage und langen Lern- und Entwicklungszeiten kann ein Algorithmus nicht aufbauen.

Einen Beitrag zur Problemlösung für ein spezielles Anwendungsgebiet leistet diese Dissertation. Es wird eine generelle Methodik zur automatischen Segmentierung von organischen Objekten einer gleichartigen Gruppe (kurz **organo**-Gruppe) vorgestellt. Als Basis zur Lösungsfindung dient das Segmentierungsproblem für Stammschnittflächen von Rohholzabschnitten. Die generelle Lösung wird auf andere **organo**-Gruppen angewandt und die Verallgemeinerung der Lösung bewiesen.

Das entwickelte Segmentierungsverfahren ist flexibel und abstrakt betrachtet zweistufig. In der ersten Stufe werden Repräsentanten einer **organo**-Gruppen detektiert und deren Charakteristiken extrahiert. In der zweiten Stufe werden die Charakteristiken zur Segmentierung der Objekte der **organo**-Gruppen genutzt. Eine optimierungsbasierte Segmentierung mit einer speziellen Wahrscheinlichkeitsberechnung anhand der Objekt-Charakteristiken bildet den Kern der zweiten Stufe.

1.1 Motivation

Im Forschungsbereich des Maschinellen Sehens ist die Segmentierung eine notwendige Voraussetzung zur Klassifikation und Ableitung von weiteren Parametern, wie Größe oder Volumen. Eine robuste Segmentierung der Objekte einer **organo**-Gruppen ermöglicht verschiedene Anwendungen. Es ergeben sich Anwendungen in der Vermessung, Bestimmung von Quantität, Qualitätsprüfung, Inspektion oder auch der automatischen Sortierung. Denkbare Szenarien der Anwendung finden sich bei Früchten, wie Äpfel an einem Baum, Melonen auf einem Förderband oder Weintrauben an einer Rebe. Weitere Szenarien ergeben sich in der Biologie bei gleichartigen Pflanzen- oder Tiergruppen, wie Muscheln, Bienen oder auch Zooplankton. Eine weitere Anwendung ist die Vermessung von Baumstämmen. Die Grundlage für diese Anwendung bildet hier die Segmentierung von Stammschnittflächen. In dieser Dissertation wird zur Herleitung eines allgemeinen Verfahrens die Segmentierung von Stammschnittflächen genauer betrachtet. Die These der Anwendbarkeit des allgemeinen Verfahrens auf andere spezielle **organo**-Gruppen wird durch die zusätzliche Betrachtung von Kartoffeln, Äpfeln und Fischen gestützt.

1 Einleitung

Die Segmentierung der Stammschnittflächen ist ein selten adressiertes Forschungsthema mit großen Herausforderungen. Die Herausforderungen bestehen in den unterschiedlichen Ausprägungen der Stammschnittflächen, der schlechten Abgegrenztheit zum Hintergrund und der Bildaufnahme unter Praxisbedingungen. Der erarbeitete Lösungsansatz in dieser Dissertation besteht in der Nutzung der Charakteristiken der Stammschnittflächen und des Hintergrundes pro Bild. Die Charakteristiken werden zuerst extrahiert und anschließend für die Segmentierung genutzt. Die Stammschnittflächen sind sich zwar in gewisser Weise in Form und Farbe ähnlich, aber oftmals nur innerhalb eines Bildes. Auch die typische runde Form eines Stammes kann nicht vorausgesetzt werden. Die Abbildung 1.1 zeigt die Verschiedenheit der Stammschnittflächen, des Hintergrunds und der Aufnahmebedingungen.



Abbildung 1.1: Im Wald stehende Rundholzabschnitte (Polter) guter Qualität. Der Hintergrund, die Form- und Farbcharakteristiken, sowie die Aufnahmebedingungen sind von Bild zu Bild unterschiedlich.

Als Basis für die Entwicklung der Konzepte und Lösungen dient das Anwendungsszenario der fotometrischen Baumstammvermessung. In der Forstwirtschaft werden Baumstämme nach dem Fällen im Wald zu sogenannten Poltern aufgestapelt. Zur Erfassung der gefällten Menge für den Verkauf oder weiterer Logistik ist eine Vermessung der Rohholzpolter notwendig. Ein bewährtes und in der Praxis oft eingesetztes Vermessungsverfahren, ist das Sektionsverfahren. Dabei wird der Polter manuell in Sektionen eingeteilt und das Volumen von erfahrenen Mitarbeitern sektionsweise geschätzt. Das Verfahren ist allerdings subjektiv, zum Teil zeitaufwändig und birgt Ungenauigkeiten.

Bisherige bildbasierte industrielle Anwendungen unterstützen bereits den Prozess der Holzstammvermessung. Die Verfahren benötigen allerdings unhandliches Equipment, wie das Stereo-Messsystem sScale, oder erfordern zur Segmentierung der Stammschnittflächen Nutzerinteraktion (siehe Produktblatt [Sys08]). Die Anwendung von Stereo-Verfahren zur Vermessung beurteilt F. Fink in seiner Dissertation [Fin04] mit den Worten:

„Aufgrund des erheblich größeren Mess- und Auswerteaufwands, den das dreidimensional foto-optische Vermessungsverfahren erfordert, kommt es zudem als Messverfahren in der forstlichen Praxis kaum in Frage.“

Es ergibt sich folglich ein Bedarf an einer mobilen, genauen und leicht zu handhabenden Lösung, die durch automatische Bildverarbeitung auf Basis einzelner Fotos realisierbar

ist. Die Algorithmen müssen den Anforderungen gerecht werden. Diese Arbeit stellt sich diesem Bedarf.

Weiterhin wird in dieser Dissertation die erarbeitete Lösung auf das allgemeine Problem der Segmentierung einer **organo**-Gruppe angewendet. Zum Beweis der Allgemeingültigkeit werden drei weitere Anwendungsszenario betrachtet, die Segmentierung von Plattfischen, Kartoffeln und Äpfeln. Die Segmentierung von Kartoffeln und Äpfeln ermöglicht Anwendungen zur automatischen Zählung, Vermessung und Positionsbestimmung. Konkrete Anwendungsbeispiele sind die automatische Sortierung, Qualitätsprüfung und Volumenbestimmung von Kartoffeln und Äpfeln nach der Ernte. Ein Anwendungsbeispiel für Fische ist eine automatische Vermessung und Artenbestimmung nach dem Fischfang. Die einzige Bedingung für eine erfolgreiche Anwendung ist, dass die organischen Objekte ähnlich sind und sich kaum bis geringfügig überschneiden. Beispiele sind auf dem Boden oder einem Förderband liegende organische Objekte.



Abbildung 1.2: Drei weitere **organo**-Gruppen, die auf verschiedene Böden liegen.

Die Herausforderungen sind ähnlich dem Problem der Stammschnittflächensegmentierung. Die Bilder in Abbildung 1.2 zeigen einen variierenden Hintergrund und Objekte mit inhomogener Farbe.

1.2 Zielsetzung und Problemstellung

Die Segmentierung von Objekten ist in der Bildverarbeitung ein großes Forschungsfeld. Ein spezielles Segmentierungsproblem benötigt **i.d.R.** eine spezielle Lösung. Es existiert noch keine einheitlich robuste Lösung für beliebige Objekte. Laut [YFR04] ist es auch gar nicht möglich, denn:

„Image segmentation is basically a problem of psycho-physical perception, and therefore, not susceptible to a purely analytical solution.“

Demnach besteht die wissenschaftliche Herausforderung dieser Arbeit in der Entwicklung eines geeigneten Verfahrens für mehrere Objektklassen. Das zu entwickelnde Verfahren soll eine größere Klasse von Objekten und Umweltstrukturen abdecken.

1 Einleitung

Die Zielsetzung dieser Dissertation ist die Entwicklung einer Lösung für das Problem der automatischen Segmentierung einer *organo*-Gruppen. Am Beispiel der Stammschnittflächen soll ein Verfahren entwickelt werden, das robust gegenüber der Varianz von Objekten einer Klasse und der Varianz der Umwelt ist. In Einzelbildern soll jede Stammschnittfläche separiert werden, was begrifflich einer Multi-Objekt-Segmentierung entspricht. Im Ergebnis soll ein Segment genau einer Stammschnittfläche entsprechen, wie es in Abbildung 1.3 illustriert ist.

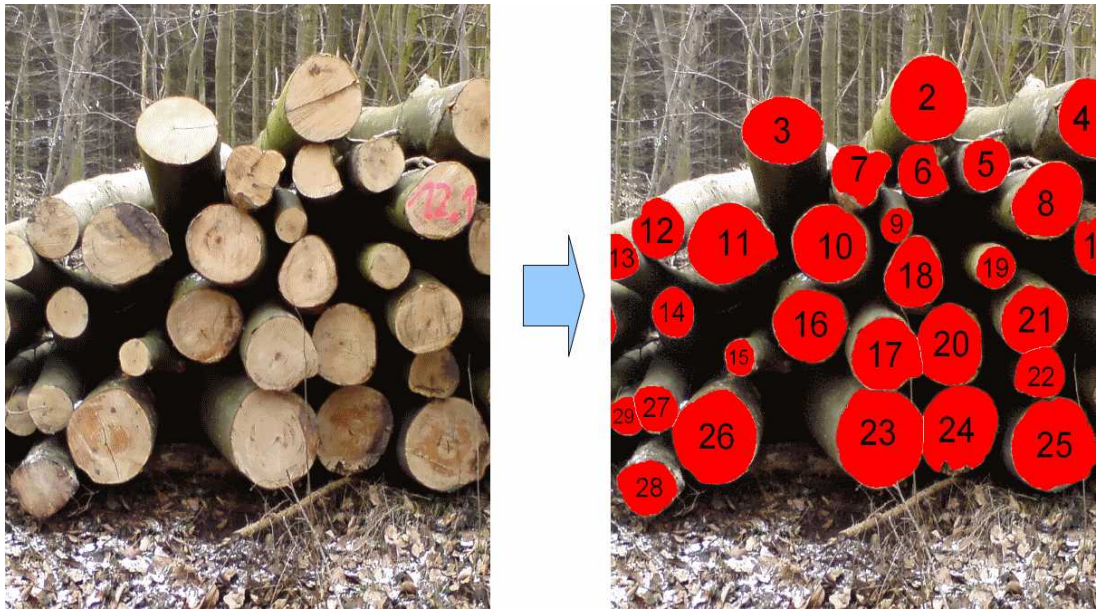


Abbildung 1.3: Die Abbildung zeigt einen Rohholzpolter (links) und das zu erzielende Segmentierungsergebnis (rechts).

Voraussetzung zur Erfüllung der Zielstellung ist eine sichtbare Polterfläche. Das heißt die Stammschnittflächen sind möglichst orthogonal zur Blickrichtung der Kamera ausgerichtet und nur gering bis gar nicht verdeckt. Über- und unterbelichtete Bilder sind nicht Gegenstand der Betrachtung.

Der Schwierigkeitsgrad einer Segmentierungsaufgabe steigt, je weniger Annahmen über das Objekt und den Hintergrund eines Bildes gemacht werden können. Hilfreiche und *i.d.R.* genutzte Annahmen im Bereich der Segmentierung von Objekten sind Form-, Farb- und Texturcharakteristiken. Ähnlich ist es beim Hintergrund, welcher sich aus allen Nicht-Objektpixeln im Bild zusammensetzt. Je homogener der Hintergrund in Farbe und Textur, und je weniger Ähnlichkeit zum Objekt besteht, um so einfacher ist die Segmentierungsaufgabe. Zum Beispiel ist eine Segmentierung eines nicht blauen Objektes auf einem blauen Hintergrund keine große Herausforderung. Die Segmentierung von Stammschnittflächen in Bildern natürlicher Umgebung lässt hingegen nicht viele Annahmen zu.

1.2 Zielsetzung und Problemstellung

Die in Abbildung 1.4 zu sehenden Bilder demonstrieren die Herausforderungen der Segmentierungsaufgabe. Die Stämme weisen nur begrenzt eine runde oder elliptische Form auf. Viele Stämme besitzen keine geometrisch eindeutig klassifizierbare Form. Es können sowohl konkave als auch konvexe Schnittflächen gegeben sein. Eine charakteristische Farbe der Stämme ist ebenfalls nicht gegeben. Aufgrund der verschiedenen Holzsorten und der verschiedenen Zustände, wie nasses, verschmutztes, defektes oder farblich markiertes Holz, variiert die Farbe. Eine weitere Schwierigkeit besteht im unbestimmten Hintergrund. So können bspw. Waldboden, Rasen, Matsch, Blätter oder auch Schnee im vorderen Bereich sein, wohingegen Bäume, Himmel oder auch Berge im hinteren Bereich vorkommen. Eine weitere Herausforderung besteht, wenn sich Farbbereiche der Schnittflächen mit



Abbildung 1.4: Die linke Seite zeigt nicht selten vorkommende Stammschnittflächen aus der forstwirtschaftlichen Praxis. Zu sehen sind verdreckte, markierte, nicht runde und beschädigte Stämme. Die Mitte zeigt verschiedene Hintergründe und die rechte einige Extremfälle von Poltern.

der Farbe des Hintergrunds überschneiden. Zu guter Letzt sind auch Schattenwürfe und grenzwertige Lichtbedingungen nicht ausgeschlossen. Es ergeben sich folglich die nachstehenden Probleme:

- Objekte variieren in der Form, der Farbe und in der Textur
- Hintergrund (nicht Objektpixel) ist in der Farbe und der Textur unbekannt
- Varianzen in Farbe, Form und Textur innerhalb der Objektklasse
- Varianzen in Farbe und Textur im Hintergrund
- Überschneidung der Farbbereiche vom Hintergrund mit den Objekten

Ziel dieser Dissertation ist es, die identifizierten Probleme systematisch zu lösen und eine automatische Multi-Objekt-Segmentierung der Stammschnittflächen von Rundholzabschnitten zu erarbeiten. Die Segmentierung soll zum Aufzeigen einer Verallgemeinerung auf andere *organo*-Gruppen angewendet werden.

1.3 Organisation der Arbeit

Diese Arbeit über die automatische bildbasierte Segmentierung von organischen Objekten einer gleichartigen (**organo**) Gruppe gliedert sich in aufeinander aufbauende Kapitel. Ein Einblick in das Thema wurde bereits in diesem Kapitel 1 gegeben. Das Thema der Arbeit wurde vorgestellt und motiviert, die Ziele definiert und die zu lösenden Probleme aufgezeigt. Eine kurze Zusammenfassung der wichtigsten Ergebnisse wird im nachfolgenden Abschnitt 1.4 noch gegeben.

Im Kapitel 2 werden die notwendigen Grundlagen der bildbasierten Segmentierung beschrieben und essentielle Zusammenhänge herausgearbeitet. Es werden die zum Verständnis der Arbeit erforderlichen Begriffe definiert und erklärt. Die Segmentierung und Objektdetektion werden in Zusammenhang gebracht und es wird die allgemeine Problemstellung erläutert. Die etablierten und von der Segmentierungsaufgabe unabhängigen Basismethoden werden unter Betrachtung unterschiedlichster bestehender Klassifikationen neu klassifiziert und erklärt. Anschließend werden bereits existierende spezielle Methoden zur Segmentierung von Stammschnittflächen und ähnlichen Objekten beschrieben. Die Basismethoden und die speziellen Methoden werden abschließend gegenübergestellt.

Im Kapitel 3 werden unter Betrachtung der Grundlagen Konzepte zur automatischen Segmentierung von Stammschnittflächen erarbeitet. Die zu lösenden Probleme werden herausgestellt, mögliche Konzepte diskutiert, drei grobe Konzepte erarbeitet und verfeinert. Es ergibt sich jeweils ein Konzept zur Schnittholzsegmentierung mit Graph-Cut und zur Segmentierung von Stammschnittflächen durch Objektdetektion. Die Kombination der beiden Konzepte ergibt ein drittes Konzept, nämlich zur automatischen Stammschnittflächensegmentierung. Alle Konzepte werden evaluiert, wobei sich das kombinierte Konzept als genauestes und robustestes Vorgehen erweist.

Im Kapitel 4 wird das kombinierte Konzept zur Segmentierung von Stammschnittflächen generalisiert und angepasst. Die einzelnen Schritte werden an Beispielen von Äpfeln, Kartoffeln und Plattfischen diskutiert und generalisiert. Das Ergebnis ist ein generalisiertes Konzept zur Segmentierung verschiedener **organo**-Gruppen. Das generalisierte Konzept wird mit den drei verschiedenen **organo**-Gruppen und den Stammschnittflächen evaluiert. Weiterhin wird das generalisierte Konzept den speziellen Konzepten zur Segmentierung von Stammschnittflächen gegenübergestellt und die Ergebnisse diskutiert. Abschließend werden die Voraussetzungen und Grenzen des generalisierten Konzeptes dargelegt.

Im Kapitel 5 wird die Anwendung der Konzepte betrachtet. Die Konzepte zur Segmentierung von Stammschnittflächen wurden im **AFoRS**-System umgesetzt. Das **AFoRS**-System wurde zur Vermessung von Holzpoltern entwickelt, evaluiert und ist auf dem Weg in die forstwirtschaftliche Anwendung. Dementsprechend werden zuerst einige forstwirtschaftliche Begriffe geklärt und anschließend die Architektur von **AFoRS** sowie die einzelnen Komponenten vorgestellt. Zu guter Letzt wird die Genauigkeit der Holzpoltervermessung evaluiert und der Einfluss der Segmentierung auf die Ergebnisse diskutiert.

Im letzten Kapitel 6 werden zuerst alle erarbeiteten wissenschaftlichen Fragestellungen, Inhalte, Konzepte und Ergebnisse zusammengefasst. Anschließend werden die sich aus der Arbeit ergebenden noch offenen Forschungsfragen aufgezeigt und ein Ausblick auf zukünftige Arbeiten gegeben.

1.4 Zusammenfassung der wichtigsten Ergebnisse

Der Begriff **organo**-Gruppe wird erstmals in dieser Arbeit im Kontext des Forschungsfeldes Segmentierung eingeführt und betrachtet. Der aktuelle Stand im Forschungsbereich der bildbasierten Segmentierung wird neu aufgearbeitet und klassifiziert. Die Methoden der Objektdetektion und der Segmentierung werden klar voneinander getrennt. Die grundlegenden Methoden der Segmentierung (Basismethoden) und die anwendungsspezifischen Methoden zur Segmentierung von **organo**-Gruppen (speziellen Methoden) werden analysiert und gegenübergestellt. Es wird belegt, dass noch keine geeignete Methode zur automatischen Segmentierung von Stammschnittflächen existiert und ähnliche spezielle Methoden zu unflexibel sind, wie bspw. Methoden für Tomaten und Äpfel.

In dieser Arbeit werden **drei neue Konzepte** zur automatischen Segmentierung von Schnittholz und Stammschnittflächen erarbeitet. Die drei Konzepte werden speziell zur Segmentierung der Frontfläche eines Holzpolters entwickelt. Die Konzepte werden generalisiert und ein Vorgehen, das zur Segmentierung verschiedenster **organo**-Gruppen geeignet ist, erarbeitet. Alle Konzepte inklusive der Generalisierung folgen einer Grundidee dieser Arbeit: *„Zuerst Informationen über die Bildobjekte sammeln und dann mit den Informationen segmentieren“*. Die Grundidee basiert auf der Annahme, dass die Bildobjekte einer **organo**-Gruppe in einem Bild ähnlich, von Bild zu Bild aber verschieden sind.

Das erste **Konzept der Schnittholzsegmentierung** segmentiert automatisch die Schnittholzflächen eines Holzpolters in Schnittholz- und Nicht-Schnittholzpixel. Dabei wird, ausgehend von wenigen Restriktionen in der Bildaufnahme und wenigen Farbheuristiken segmentiert. Das Konzept benötigt keinen Objektdetektor und ermöglicht die Volumenbestimmung eines Holzpolters, wenn der Maßstab und die Holzstammlänge bekannt sind.

Das zweite **Konzept der Stammschnittflächendetektion** segmentiert automatisch einzelne Stammschnittflächen auf Basis eines Detektors ohne weitere Bildaufnahmerestriktionen. Es sind keine Farbheuristiken und optimierungsbasierte Segmentierungen im Konzept verankert, wodurch das Konzept eine leichtgewichtige und performante Lösung zur Segmentierung von Stammschnittflächen darstellt. Auf Basis der segmentierten Stammschnittflächen wird zusätzlich zur Volumenbestimmung die Bestimmung der Größenklasse einzelner Holzstämmen ermöglicht.

Das dritte **Konzept der Kombination**, kombiniert die Schnittholzsegmentierung mit der Stammschnittflächendetektion. Das Konzept ermöglicht die Segmentierung einzelner Stammschnittflächen und ist genauer als die vorherigen Konzepte. Da die Schritte der vorherigen Konzepte kombiniert wurden, ist allerdings die Komplexität und folglich auch die benötigte Rechenleistung höher.

In der **Generalisierung** werden die entwickelten Konzepte zur Segmentierung verschiedenster **organo**-Gruppen abstrahiert. Das generalisierte Konzept ermöglicht die Segmentierung von **organo**-Gruppen mit quasi-runden organischen Objekten (Überdeckung zum besten angepassten Kreis $> 60\%$). Das Konzept erweitert das Konzept der Kombination, ist genereller und segmentiert ähnlich genau (in einigen Fällen sogar besser) Stammschnittflächen wie die drei Ausgangskonzepte.

1 Einleitung

Alle Konzepte werden gegenüber der Grundwahrheit evaluiert und führen zu besseren Ergebnissen als bestehende Methoden. Die Konzepte sind in [Gut+10], [Gut+11a], [GV12], [GL13] und die Anwendung in [Foe+12] veröffentlicht.

2 Stand der Forschung zur Segmentierung

Dieses Kapitel betrachtet den Stand der Forschung zur Segmentierung von **organo**-Gruppen. Im ersten Abschnitt 2.1 werden notwendige Begriffe definiert. Im folgenden Abschnitt 2.2 wird die Segmentierung und Objektdetektion verortet, in Zusammenhang gebracht und die allgemeine Problemstellung erläutert. In Abschnitt 2.3 werden die von der Zielstellung unabhängigen Basismethoden klassifiziert und beschrieben. Im Abschnitt 2.4 werden die speziellen Methoden zur Segmentierung von **organo**-Gruppen gezeigt, wobei die Objektdetektion als Mittel zur Segmentierung aufgefasst wird. Es wird zuerst die Segmentierung von Stammschnittflächen betrachtet und folgend auf weitere **organo**-Gruppen wie Äpfel, Tomaten und Kiwis eingegangen. Die Betrachtung ist grob in Segmentierungsaufgaben und fein in Segmentierungsklassen unterteilt. Zur Lösung einer konkreten Segmentierungsaufgabe werden **i.d.R.** Basismethoden erweitert, adaptiert und kombiniert. Die Frage ist: „Welche Basismethoden eignen sich zur Lösung der Zielsetzung?“ Daher wird abschließend in Abschnitt 2.5 die Lösung der Zielsetzung aus Sicht der Basismethoden untersucht und der Kreis der Argumentation geschlossen.

2.1 Begriffsdefinition

In der Forschung werden Begriffe nicht immer einheitlich verwendet und sind durchaus mehrdeutig. Um eine einheitliche Sprache zu schaffen und Kommunikationsprobleme zu vermeiden, werden notwendige Begriffe und Symbole für diese Arbeit nachfolgend definiert.

Bildverarbeitung (Image Processing)

Die **Bildverarbeitung (BV)** ist ein Fachgebiet der Informatik und wird unterschiedlich umschreiben. Nick Efford beschreibt Bildverarbeitung in [Eff00] mit den Worten:

„Image Processing is a general term for the wide range of techniques that exist for manipulating and modifying image in various ways.“

In [Nis+11] beschreiben die Autoren Bildverarbeitung mit:

„... Bildverarbeitung ist die Analyse von Bildern ...“ und *„Grundsätzlich gibt es zwei Zielsetzungen der digitalen Bildverarbeitung: Bildverbesserung und die Extraktion von Informationen.“*

Verallgemeinert umfasst die **BV** alle Algorithmen und Methoden, die auf und mit Bild-daten arbeiten. Im Wesentlichen sind die Erzeugung, Verbesserung, Restaurierung, Segmentierung von Bildern sowie die Extraktion von Merkmalen aus Bildern Bereiche der Bildverarbeitung.

Maschinelles Sehen und Lernen (Computer Vision, Maschine Learning)

Maschinelles Sehen (Computer Vision – **CV**) ist ein Fachgebiet der Informatik und wird im deutschen Sprachgebrauch **i.d.R.** mit dem Gebiet *Computer Vision* sowie mit dem Gebiet *Machine Vision* gleichgesetzt. Beide Bezeichnungen beinhalten den Begriff Vision und umfassen ähnliche Aspekte. Den Begriff Vision beschreibt David Marr in [MN78] mit:

„Vision is a process that produces from images of the external world a description that is useful to the viewer and not cluttered with irrelevant information.“

Das Gebiet Machine Vision wird in [JKS95] beschrieben mit:

„The goal of a machine vision system is to create a model of the real world from images“

Als inverse Computer Graphik wird Computer Vision in [Sze10] mit den Worten beschrieben:

„In computer vision, we are trying to do the inverse, i.e., to describe the world that we see in one or more images and to reconstruct its properties, such as shape, illumination, and color distributions“

Folglich ist verallgemeinert **CV** die Erzeugung eines Modells aus Bilddaten der Umwelt und deren Interpretation. Maschinelles Sehen erweitert die Bildverarbeitung um die Interpretation durch Methoden des Maschinellen Lernens (**ML**). Teilgebiete des Maschinellen Sehens sind bspw. Tracking, 3D-Rekonstruktion und Objekterkennung. **Maschinelles Lernen** (Machine Learning – **ML**) ist ein Fachgebiet und umfasst die Ermittlung von Wissen aus Erfahrung (siehe [Nil98], [Alp10]). Durch Beispieldaten wird ein künstliches System angelernt, welches entsprechende Regeln erkennt, auswertet und neue unbekannte Daten beurteilen kann. In [Alp10] wird **ML** beschrieben mit:

„The goal of machine learning is to program computers to use example data or past experience to solve a given problem.“

Bild, Bildkanal und Pixel

Ein **Bild** I ist eine 2D-Matrix von Pixeln px mit einer Höhe h und einer Weite w . Ein Bild I soll analog auch als Menge von Pixeln aufgefasst werden, wobei gilt:

$$I = \{px_{x,y} | 0 \leq x < w \wedge 0 \leq y < h \wedge x, y, w, h \in \mathbb{N}\} \quad (2.1)$$

Der Koordinatenursprung eines Bildes befindet sich **i.d.R.** links oben und beginnt bei $(0, 0)$. Eine 2D-Bildmatrix soll wie folgt indiziert werden:

$$I = \begin{pmatrix} px_{0,0} & px_{1,0} & \dots & \dots & px_{w-1,0} \\ px_{0,1} & px_{1,1} & \dots & \dots & px_{w-1,1} \\ \dots & \dots & px_{x,y} & px_{x+1,y} & \dots \\ \dots & \dots & px_{x,y+1} & \dots & \dots \\ px_{0,h-1} & px_{1,h-1} & \dots & \dots & px_{w-1,h-1} \end{pmatrix} \quad (2.2)$$

Ein **Pixel** $px = (va_1, \dots, va_i, \dots, va_n)$ ist ein n -Tupel von Werten va_i (values). Die Anzahl n entspricht der Anzahl der **Bildkanäle**. Zum Beispiel hat ein *8 Bit Grauwertbild* einen Kanal und ein *24-Bit RGB-Bild* drei Kanäle $px = (va_1, va_2, va_3) = (R, G, B)$. Für beide Beispiele gilt $va_i \in [0, 255]$ und $va_i \in \mathbb{N}$. Jeder Kanal i bildet analog eine 2D-Matrix I_i^c bestehend aus den Werten va_i . Die Komposition der Kanäle ergibt wieder das Bild I .

$$I = \{(va_1, \dots, va_n) | \forall va_1 \in I_1^c \wedge \dots \wedge va_n \in I_n^c : \text{gleiche Bildposition}\} \quad (2.3)$$

Bereich von Interesse, Teilbild und Pixelsamples

Ein **Bereich von Interesse** (Region of Interest – **RoI**) bezeichnet einen rechteckigen Bildbereich im Bild I . Die Kopie des Bildbereiches der **RoI** soll als **Teilbild** (*subimage*) bezeichnet werden. Das Teilbild I^{sub} erfüllt alle Kriterien eines Bildes I und ist echte Teilmenge des Bildes ($I^{sub} \subset I$). Ein Teilbild I^{sub} beinhaltet alle Pixel vom obersten linken Rand (x_s, y_s) bis zum untersten rechten Rand (x_e, y_e) , somit gilt

$$I^{sub} = \{px_{x,y} | \forall px_{x,y} \in I : x_s \leq x < x_e \wedge y_s \leq y < y_e \wedge x, y \in \mathbb{R}\}; \quad (2.4)$$

In dieser Arbeit wird außerdem der Begriff **Pixelsamples** verwendet. Pixelsamples sind Beispielpixel einer Pixelmenge eines Bildes oder eines Teilbild. Für ein Bild I werden Pixelsamples I^{smp} wie folgt definiert:

$$I^{smp} \subseteq I \quad (2.5)$$

Objekt und Bildobjekt

Ein reales **Objekt** O^r ist ein zusammenhängendes Ganzes aus der realen Umwelt, wie eine Tasse, ein Buch oder ein Fisch. Die Projektion des realen 3D Objektes O^r auf ein Bild führt zu einer Abbildung des Objektes in 2D. Die 2D Abbildung soll als **Bildobjekt** bezeichnet und zur Vereinfachung nur mit O gekennzeichnet werden. Ein Bildobjekt O ist folglich eine Menge von Pixeln px eines Bildes, wobei $O \subset I$ ist.

Organische Objekte

Unter einem **organischen Objekt** soll in dieser Arbeit ein Lebewesen wie ein Fisch oder eine Biene zu verstehen sein. Auch Bestandteile von Lebewesen, wie ein Apfel, ein Blütenblatt oder die Schnittfläche eines Baumstammes, sollen als organische Objekte bezeichnet werden. Zur Unterscheidung von Real und 2D-Abbildung soll das reale organische Objekt mit $O^{r,o}$ und ein organisches Bildobjekt mit O^o gekennzeichnet werden.

Organische Objekte einer gleichartigen Gruppe (organo-Gruppe)

Unter einer **organo-Gruppe** soll eine Gruppe organischer Objekte einer biologisch gleichen Art (*species*) zu verstehen sein. Eine **organo-Gruppen** soll folglich gleichartige Lebewesen oder gleiche Bestandteile von gleichartigen Lebewesen beinhalten. Beispiele einer **organo-Gruppe** sind ein Schwarm Heringe, ein Blütenfeld von Sonnenblumen, ein

2 Stand der Forschung zur Segmentierung

Polter von Kiefernstämmen und Orangen an einem Baum. Die reale **organo**-Gruppe, bestehend aus realen 3D-Objekten, soll mit OG^r und deren Abbildung im Bild mit OG gekennzeichnet werden. Verdeckungen durch die 3D- zu 2D-Projektion (OG^r zu OG) sind möglich. Eine **organo**-Gruppe OG ist eine Menge artgleicher organischer Bildobjekte O^o und soll definiert werden durch:

$$OG = \{O_i^o | i \in \Lambda\}; \forall i, j \in \Lambda : O_i^o \cap O_j^o = \emptyset \wedge O_i^o \subset I \text{ für } i \neq j, \quad (2.6)$$

wobei Λ der Indexmenge entspricht.

Label und Labeling-Problem

Das allgemeine **Labeling-Problem** bezeichnet das Problem der Zuordnung vom **Label** l einer Labelmenge LD zu einem Ort (*Site*) d einer Domain D . Die Domain D ist eine Abstraktion einer diskreten Menge von m Orten:

$$D = \{1, \dots, m\} \quad (2.7)$$

Die Labelmenge LD ist diskret oder kontinuierlich. In dieser Arbeit wird vom diskreten Fall ausgegangen, wie es in der Segmentierung **i.d.R.** üblich ist. Ein **Label** $l \in LD$ ist ein Bezeichner für eine Klassenzugehörigkeit und ordnet folglich einen Ort d einer Klasse zu. Die Zuordnung wird durch die nachfolgende Labelfunktion $f_l(d)$ definiert, wie es auch in [Li95] beschrieben ist:

$$f_l : D \rightarrow LD \quad (2.8)$$

Bildmerkmale

Bildmerkmale sind Eigenschaften von Pixelmengen, wie Segmenten S , Teilbildern I^{sub} oder dem Bild I . Ein Bildmerkmal bm ist **i.d.R.** ein skalarer Wert, der eine Eigenschaft der Pixelmenge kodiert. Sind n Bildmerkmale der selben Pixelmenge gegeben, werden diese auch zusammengefasst zum Merkmalsvektor:

$$\vec{m} = (bm_1, \dots, bm_n) \quad (2.9)$$

Die n Bildmerkmale bilden auch Achsen eines n dimensionalen Raumes, des sogenannten Merkmalsraumes.

Klassifikation, Klassifikator und Trainingsdaten

Eine **Klassifikation** ist die Einteilung einer Menge von Entitäten in Gruppen, in sogenannte Klassen. Die Unterteilung der Entitäten sowie der Prozess, eine Entität einer Klasse einzuordnen, wird als Klassifikation bezeichnet. Beispiele von Entitäten in der Bildverarbeitung sind einzelne Pixel, Bildobjekte, Bildbereiche, aber auch Bildmerkmale. Die Klassifikation einer Entität et in eine Klasse k wird durch einen Klassifikator vorgenommen. Formal soll ein **Klassifikator** durch die eindeutige Funktion $f_k(et)$ wie folgt beschrieben werden:

$$f_k : \text{Entitäten} \rightarrow \text{Klassen}, et \mapsto k; et \in \text{Entitäten}; k \in \text{Klassen} \quad (2.10)$$

Eine Entität et wird folglich genau auf eine Klasse k abgebildet. Im Forschungsbereich der **CV** wird **i.d.R.** eine Beschreibungsform einer Entität et , der Merkmalsvektor \vec{m} zur Klassifikation, genutzt. Ein Klassifikator ist außerdem ein Algorithmus und wird **i.d.R.** unter Zuhilfenahme von **Trainingsdaten** erstellt (angelernt). Trainingsdaten bestehen aus Entitäten, die jeweils eine Klasse repräsentieren.

Interpretation

Der Begriff **Interpretation** bedeutet im allgemeinen Sinne die Deutung einer Aussage. Der Begriff wird innerhalb der **CV** verwendet und unterschiedlich gebraucht. Im Kontext der **CV** und in dieser Arbeit soll unter Interpretation eine Erweiterung der Klassifikation verstanden werden. Eine Klassifikation bildet eine Entität, wie bspw. einen Merkmalsvektor eines Objektes O , auf eine Klasse ab. Die Klassifikation inklusive der Deutung von der Klasse auf das reale Objekt O^r soll als Interpretation bezeichnet werden.

Segmentierung und Segmente

Der Begriff **Segmentierung** bezeichnet die Zerlegung eines Bildes in disjunkte Teilmengen. Jede Teilmenge fasst eine Menge von Pixeln mit **i.d.R.** gemeinsamen Eigenschaften zusammen. Der Vorgang der Segmentierung wird auch als Labeling bezeichnet. Das beschriebene Labeling-Problem lässt sich im Kontext der Segmentierung konkretisieren, wobei ein Bild I auf eine diskrete Labelmenge L abgebildet wird.

$$f_s : I \rightarrow L, px \mapsto l; px \in I; l \in L \quad (2.11)$$

Die Segmentierungsfunktion $f_s(px)$ bildet ein Pixel px auf genau ein **Label** l ab. Die Segmentierung ist folglich eine spezielle Form der Klassifikation. Die Entitäten entsprechen dem Bild I und die Klassen der Labelmenge L . Verbundene Pixel mit identischem Label werden als **Segment** bezeichnet. In dieser Arbeit wird eine **Vierer-Nachbarschaft** (siehe [Jah05]) zu Grunde gelegt. Nachbarn vom Pixel $px_{x,y}$ an der Position x und y sind die Pixel $px_{x+1,y}$, $px_{x-1,y}$, $px_{x,y+1}$ und $px_{x,y-1}$. Ein Segment besteht aus mindestens einem Pixel und soll mit S gekennzeichnet werden, wobei gilt $S \subset I$. Die Menge aller Pixel aller Segmente ergibt wieder das Bild. Ein Beispiel ist in Abbildung 2.1 skizziert.

$$I = \bigcup_{i \in \Lambda} S_i; \forall i, j \in \Lambda : S_i \cap S_j = \emptyset \text{ für } i \neq j \quad (2.12)$$

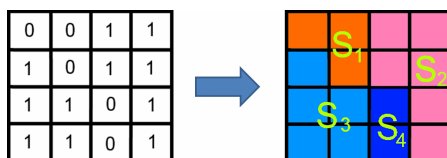


Abbildung 2.1: Die Segmente eines binären Bildes ($L = \{0, 1\}$).

Binäre, Objekt und Multi-Objekt-Segmentierung

Mengentheoretisch betrachtet wird ein Bild bei einer **binären Segmentierung** in zwei disjunkte Mengen A und B von Pixeln px zerlegt, wobei gilt:

$$(A \cup B = I) \wedge (A \cap B = \emptyset) \quad (2.13)$$

Verbundene Pixel gleicher Menge bilden ein Segment S . Die Begriffe Objekt und Multi-Objekt-Segmentierung werden nicht ganz einheitlich in der Forschung verwendet. In dieser Arbeit soll unter **Objekt-Segmentierung** eine binäre Segmentierung verstanden werden. Ziel der Segmentierung ist es, das Bild in eine Bildobjekt O und Nicht-Bildobjekt Pixelmengemenge \bar{O} zu zerlegen, wobei gilt:

$$(\bar{O} = I \setminus O) \wedge (O \hat{=} S) \quad (2.14)$$

Ein Segment S entspricht demzufolge dem Bildobjekt O , was im Gegensatz zur binären Segmentierung keine Bedingung ist. Unter **Multi-Objekt-Segmentierung** soll eine Zerlegung des Bildes in n Bildobjekte ($n > 1$) und eine Nicht-Bildobjekt Menge \bar{O} zu verstehen sein, wobei gilt

$$\bar{O} = I \setminus \{O_i | i \in \Lambda\}; \forall i, j \in \Lambda : (O_i \cap O_j = \emptyset) \wedge (O_i \hat{=} S_i) \text{ für } i \neq j \quad (2.15)$$

Bei einer binären Segmentierung hat die Labelmenge L genau zwei Elemente $L = \{0, 1\}$ und bei einer Multi-Objekt-Segmentierung mit n Objekten $n + 1$ Elemente. Die Ergebnisbilder einer binären Segmentierung und einer Multi-Objekt-Segmentierung sollen unterschiedlich bezeichnet werden:

- Ergebnisbild einer binären Segmentierung: I^b
- Ergebnisbild einer Multi-Objekt-Segmentierung: I^m

Segmentmenge

Die Segmentierung eines Bildes mit mehr als einem Pixel ($|I| > 1$) ist nicht eindeutig. Es gibt mehrere Möglichkeiten, Bild I in Segmente S zu zerlegen. Die Vereinigung der Segmente aller möglichen Zerlegungen soll mit SI gekennzeichnet werden. SI wird in dieser Arbeit als **Segmentmenge** bezeichnet und definiert durch:

$$SI = \{a | \forall a \in I : a \subset I \wedge a \hat{=} S\} \quad (2.16)$$

Automatische, halbautomatische und manuelle Segmentierung

Der Begriff **automatische Segmentierung** soll die Segmentierung eines Bildes ohne zusätzliche Nutzerinteraktion bezeichnen. Das heißt, das Bild wird lediglich von einem Algorithmus segmentiert, welcher das Eingangsbild automatisch zerlegt. Dem entgegen soll unter **halbautomatische Segmentierung** eine Zerlegung des Bildes mit Hilfe von Nutzerinteraktion verstanden werden. Beispiele dafür sind das Setzen von Saatpunkten oder einer groben Kontur. Unter **manuelle Segmentierung** ist die Segmentierung des Bildes durch den Nutzer zu verstehen ohne Algorithmus.

Vorwissen und Modell

Im Kontext der Bildverarbeitung soll unter dem Begriff **Vorwissen** (A-Priori) Wissen über den Bildinhalt zu verstehen sein. Vorwissen beinhaltet Wissen über die Objekte O^r und deren Bildobjekte O sowie den Nicht-Objekt \bar{O} Bildinhalten (Hintergrund). Konkretes Vorwissen ist bspw. die Position, Form oder Farbe eines Objektes. Ein **Modell** hingegen ist formal spezifiziertes Vorwissen. Ein Modell soll abstrakt mit Θ gekennzeichnet werden. Grob soll ein Modell in Objektmodell (Θ^O), Vordergrundmodell (Θ^{FG}) und Hintergrundmodell (Θ^{BG}) unterschieden werden. Θ^O spezifiziert Eigenschaften genau eines Objektes, Θ^{FG} aller Objekte und Θ^{BG} aller Nicht-Objekt-Bildinhalte.

Segmentierungsaufgabe

Die **Segmentierungsaufgabe** soll die gewünschte Zerlegung des Bildes definieren und entspricht analog der Zielsetzung der Segmentierung. Die Segmentierungsaufgabe definiert die Pixelmengen der Bildobjekte O und die Nicht-Bildobjekt-Pixel \bar{O} . Damit definiert die Segmentierungsaufgabe implizit, ob eine Objekt, Multi-Objekt oder Binäre-Multi-Objekt-Segmentierung die Zielstellung ist. Die Aufgabe einer Segmentierung kann das Bild binäre segmentieren mit der Forderung, dass das Bild in eine Himmel- und eine Nicht-Himmel Pixelmenge zerlegt wird, sein. Ein weiteres Beispiel einer Segmentierungsaufgabe ist eine Multi-Objekt-Segmentierung von Fischen. Das Bild soll dabei in n Fischsegmente und eine Nicht-Fisch-Pixelmenge zerlegt werden.

Basismethoden und spezielle Methoden der Segmentierung

Die **Basismethoden** sollen die Methoden der Segmentierung zusammenfassen, die unabhängig von der Segmentierungsaufgabe sind. Hingegen sollen die **speziellen Methoden** der Segmentierung abhängig von einer Segmentierungsaufgabe sein. Basismethoden sind **i.d.R.** allgemeingültiger und spezielle Methoden oft angepasste, kombinierte und/oder erweiterte Basismethoden.

Über- und Untersegmentierung

Durch eine Segmentierung wird ein Bild in n Segmente S zerlegt. Wenn viele kleine Segmente S das zu segmentierende Bildobjekt O überdecken, spricht man von einer **Übersegmentierung**. Gegenteilig verhält es sich bei der **Untersegmentierung**, bei der ein Segment S größer als das zu segmentierende Bildobjekt O ist. Verallgemeinert wird auch die Zerlegung eines Bildes in zu kleine Segmente als Übersegmentierung und in zu große Segmente als Untersegmentierung bezeichnet. Die Über- und Untersegmentierung ist ein bekanntes Problem der Segmentierung und ist oft bei den Basismethoden der regionenbasierten Segmentierung vorzufinden.

Objektdetektion

Als **Objektdetektion** wird die Bestimmung eines Segmentes S bezeichnet, das einem Bildobjekt O von O^r entspricht ($S \hat{=} O$) oder ein Teilbild ist ($S = I^{sub}$) und O beinhaltet

2 Stand der Forschung zur Segmentierung

($O \subset S$). Die Detektion von S im Bild I erfolgt mit Vorwissen über das Objekt, wie bspw. durch ein Objektmodell. Strategien der Klassifikation und der Suche sind notwendig, um die Segmente S zu ermitteln. Gesucht und klassifiziert wird direkt im Bild I oder in einer Transformation des Bildes, wie bspw. im Farb- oder Merkmalsraum. Bei der Objektdetektion wird im Gegensatz zur Segmentierung die Segmentmenge SI und auf die Labelmenge L abgebildet. Die zugehörige Objektdetektionsfunktion $f_{od}(S)$ sei definiert mit:

$$f_{od} : SI \rightarrow L, S \mapsto l; S \in SI; l \in L \quad (2.17)$$

Ein Pixel kann bei der Objektdetektion Bestandteil mehrerer Segmente sein. Es ist folglich keine disjunkte Zerlegung des Bildes und folglich **keine Segmentierung**. Die diskrete Labelmenge L zu einer Objektdetektion ist zumeist binär ($L = \{0, 1\}$). Die Segmente S der Segmentmenge SI erhalten das Label 1 für Bildobjekt oder 0 für Nicht-Bildobjekt. Eine mächtigere Labelmenge ($|L| > 2$) zur Detektion von verschiedenen Objekten ist möglich, aber **i.d.R.** untypisch. Im Fall der Detektion verschiedener Objekte werden **i.d.R.** mehrere Objektdetektionen eines Objektes hintereinander durchgeführt.

Kontur

Eine **Kontur** (Contour) C ist eine geschlossene Kurve, welche im diskreten Bildraum durch verbundene Pixel repräsentiert wird. Eine Kontur zu einem Segment S sind alle $px \in S$, welche am Bildrand oder benachbart mit $px \notin S$ sind. In einigen Fällen ist es notwendig eine Kontur durch die Funktion

$$c(s) : [0, 1] \rightarrow \mathbb{R}^2 \text{ mit } c(s) = \begin{pmatrix} x(s) \\ y(s) \end{pmatrix} \quad (2.18)$$

in expliziter Parameterdarstellung zu beschreiben (siehe [KWT88], [OF03]). Die x-Position $x(s)$ und y-Position $y(s)$ im Bild wird in Abhängigkeit des Parameters s definiert. Speziell bei aktiven Konturen [KWT88] wird $0 < x, y < 1$ vorausgesetzt.

2.2 Vorbetrachtung der Segmentierung und Objektdetektion

In dieser Arbeit wird die Kombination von Segmentierung und Objektdetektion einen Kernbestandteil bilden. Da beide Begriffe eng verzahnt und auch in der Forschung unterschiedlich aufgefasst werden, werden die Begriffe in diesem Abschnitt detaillierter betrachtet. Die Begriffe werden zuerst im Abschnitt 2.2.1 von einander abgegrenzt und in die entsprechenden Fachgebiete eingeordnet. Der Zusammenhang der Forschungsfelder wird anschließend im Abschnitt 2.2.2 betrachtet und die Größe des Lösungsraumes aufgezeigt. Es wird sich zeigen, dass die Objektdetektion auch ein Mittel zur Objekt-Segmentierung ist. Abschließend wird in Abschnitt 2.2.3 ein Fazit gezogen und eine Methodik zur strukturellen Bearbeitung des Forschungsstandes herausgestellt.

2.2.1 Abgrenzung und Einordnung

Die Forschungsfelder Segmentierung und Objektdetektion sind essentielle Bestandteile des Maschinellen Sehens (CV) und teilweise Bestandteil der Bildverarbeitung (BV). Die BV umfasst Manipulationen, Verbesserungen und Aufbereitungen von Bilddaten. Auch die Extraktion von Merkmalen wird der BV zugeordnet. Die CV schließt die BV ein und ergänzt die BV um die Interpretation und die Nutzung von Vorwissen. Durch die Interpretation werden Aussagen über den Bildinhalt und folglich die aufgenommene Umwelt ermöglicht. Eine Einordnung der Segmentierung und Objektdetektion ist in Abbildung 2.2 illustriert.

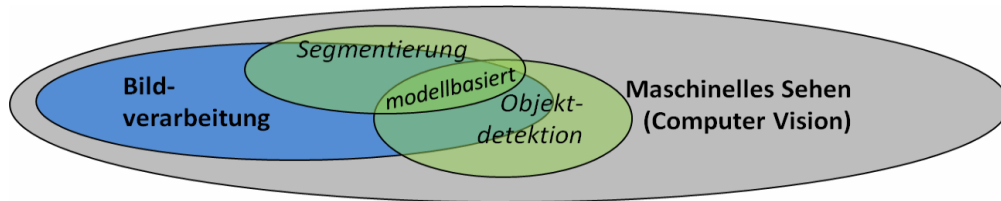


Abbildung 2.2: Einordnung der Forschungsfelder Segmentierung und Objektdetektion in die übergeordneten Fachgebiete.

Beide Forschungsfelder schneiden die BV und sind Bestandteil der CV. Die Segmentierung wird zum größten Teil in die BV eingeordnet, da Bilddaten aufbereitet und noch nicht interpretiert werden. Allerdings gibt es Segmentierungsmethoden, die Vorwissen nutzen, weshalb ein Teil der Segmentierung keine BV sondern CV ist. Einige Bereiche der Objektdetektion werden häufig als modellbasierte Segmentierung bezeichnet. Die modellbasierte Segmentierung (siehe [Jah05], [Ton05], [FP11]) sowie die Nutzung der Objektdetektion zur Segmentierung bilden die Schnittmenge von Segmentierung und Objektdetektion. Da Segmentierung sowie Objektdetektion auf Bilddaten operieren, gibt es auch eine Schnittmenge zwischen BV, Segmentierung und Objektdetektion. Die Objektdetektion im Vergleich zur Segmentierung ist zum größten Teil nur CV, da Vorwissen über die zu detektierenden Segmente verfahrenstechnisch benötigt wird. Der Interpretationsgedanke (Erkennung, Klassifikation) ist Grundbestandteil der CV und bei der Objektdetektion werden Segmente S anhand von Modellen über Bildobjekt O und/oder Objekt O^r detektiert. Die verwendeten Modelle beinhalten Vorwissen, welche Interpretationen über das Bild ermöglichen.

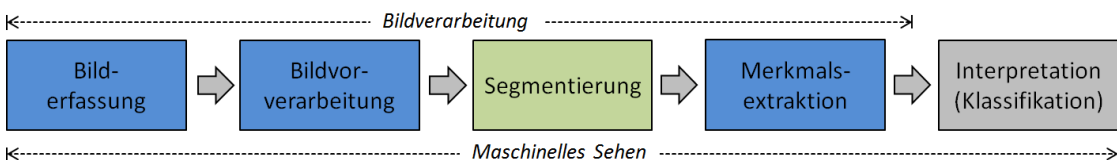


Abbildung 2.3: Die Segmentierung in der Pipeline der Bildanalyse.

2 Stand der Forschung zur Segmentierung

Bei der Analyse eines Bildes kommt *i.d.R.* die Pipeline der Bildanalyse (siehe [Han09] [Ste06] und [Abm94]) zur Anwendung, welche in Abbildung 2.3 illustriert ist. Die Pipeline der Bildanalyse¹ enthält Komponenten der **BV** und **CV**. In der Pipeline ist die Segmentierung ein fester Bestandteil. Ziel der Pipeline ist die Interpretation eines Bildes I mittels Bildvorverarbeitung, Segmentierung und Merkmalsextraktion. Die Komponenten der Pipeline haben folgende Bedeutung:

1. **Bilderfassung:** Aufnahme der realen Umgebung und Erzeugung des Bildes I
2. **Bildvorverarbeitung:** Die Verbesserung, Rekonstruktion und/oder Aufbereitung des Bildes I zum vorverarbeiteten Bild I'
3. **Segmentierung:** Zerlegung des Bildes I' in sinnvolle Segmente $I' = \{S_1, \dots, S_n\}$
4. **Mermalsextration:** Berechnung eines Merkmalsvektor \vec{m} für jedes Segment
5. **Interpretation:** Deutung der Objekte O^r mit Hilfe der Klassifikation $k = f_c(\vec{m})$ der Segmente S anhand des Merkmalsvektors \vec{m}

Der Schwerpunkt dieser Dissertation liegt in der Segmentierung (Abbildung 2.3), speziell der Schnittmenge zwischen Segmentierung und Objektdetektion ein (Abbildung 2.2).

2.2.2 Zusammenhang und Lösungsraum

Segmentierung und Objektdetektion sind zwei unterschiedliche Forschungsfelder mit unterschiedlicher Herangehensweise und dennoch ähnlicher Problemstellung. Nachfolgend wird zuerst ein Zusammenhang zwischen der Segmentierung und der Objektdetektion hergestellt und anschließend deren Lösungsräume² aufgezeigt. Im Abschluss werden die wesentlichen Erkenntnisse zusammengefasst.

Zusammenhang zwischen Segmentierung und Objektdetektion

Bei der Segmentierung wird ein Bild in disjunkte Segmente zerlegt. Die Segmente werden anschließend in der Pipeline der Bildanalyse interpretiert. Bei der Objektdetektion werden hingegen die Segmente im Bild anhand von Vorwissen detektiert. Die Richtung des Schließens ist bei der Objektdetektion im Vergleich zur Segmentierung invers. Vorwissen über das Objekt ist der Ausgangspunkt einer Objektdetektion, wohingegen das Erlangen von Wissen über das Objekt das Ziel der Segmentierung ist. Bei der Objektdetektion ist durch das Modell bekannt, was detektiert werden soll, aber das Segment ist unbekannt. Bei der Segmentierung ist das Segment auch unbekannt, aber auch seine Interpretation. Formal

¹Die Pipeline der Bildanalyse wird je nach Kontext auch als Pipeline der Mustererkennung oder Pipeline der Bildverarbeitung bezeichnet.

²Der Lösungsraum kennzeichnet in diesem Zusammenhang die Menge der möglichen Lösungen einer Objektdetektion oder Segmentierung.

2.2 Vorbetrachtung der Segmentierung und Objektdetektion

betrachtet ähneln sich die Funktionen der Objektdetektion f_{od} und der Segmentierung f_s . Sie unterscheiden sich lediglich in der Definitionsmenge wie folgt:

$$f_s : I \rightarrow L, px \mapsto l; px \in I; l \in L \text{ (Segmentierung)} \quad (2.19)$$

$$f_{od} : SI \rightarrow L, S \mapsto l; S \in SI; l \in L \text{ (Objektdetektion)} \quad (2.20)$$

Die Funktionen sind formal betrachtet spezielle Klassifikationen, die sich in den Entitäten unterscheiden. Bei der Objektdetektion werden Segmente S und bei der Segmentierung Pixel px klassifiziert. Bei der Segmentierung sind die Pixel in den meisten Methoden nicht unabhängig von einander. Nachbarschaften und Beziehungen von Pixeln werden der Segmentierungsmethode entsprechend in $f_s(px)$ einbezogen und Äquivalenzklassen von Pixeln (Segmente) gebildet. Durch eine Segmentierung und Objektdetektion ergeben sich die Korrespondenzen ϕ und Relationen R :

$$R_s = \{(px, l) \in I \times L | l \in \phi_S(px)\}; R_s \subseteq I \times L \text{ (Segmentierung)} \quad (2.21)$$

$$R_{od} = \{(S, l) \in SI \times L | l \in \phi_{od}(S)\}; R_{od} \subseteq SI \times L \text{ (Objektdetektion)} \quad (2.22)$$

Die Objektdetektion ist in der Segmentierung einsetzbar, wenn die Ergebnisrelation R_{OD} innerhalb der Segmentierungsfunktion f_s Verwendung findet. Durch die Verwendung von R_{od} zur Segmentierung ergibt sich die Funktion $f_{s|od}$ ³:

$$f_{s|od} : I \xrightarrow{R_{od}} L, px \mapsto l; px \in I; l \in L \quad (2.23)$$

Lösungsräume der Segmentierung und Objektdetektion

Beide Herangehensweisen eröffnen einen großen und ähnlichen Lösungsraum. Der Lösungsraum der Segmentierung lässt sich mit dem Labeling-Problem aufzeigen und abschätzen. Nach [Li95] ergibt sich die Menge M aller Möglichkeiten⁴ einer Abbildung der Domain D auf die Labelmenge LD , wie folgt:

$$M = LD^n = \underbrace{LD \times LD \dots \times LD}_{n\text{-fach}} \quad (2.24)$$

Im Fall der Segmentierung wird ein Bild I auf die diskrete Labelmenge L abgebildet, wobei ein Pixel px auf ein Label l abgebildet wird. Es ergibt sich im speziellen Fall der Segmentierung eines Bildes I :

$$M = L^n = \underbrace{L \times L \dots \times L}_{n\text{-fach}} \quad (2.25)$$

Ein Pixel px im Bild I mit der Weite w und Höhe h kann genau auf eines von m Labels abgebildet werden. Die Anzahl m der möglichen Abbildungen ergibt sich durch:

$$m = |M| = |L|^{|I|}; |I| = w * h \quad (2.26)$$

³($\xrightarrow{R_{od}}$) und ($s|od$) soll kennzeichnen, dass die Relation R_{od} in der Abbildung verwendet wird.

⁴Im Zusammenhang mit Markov Random Fields (MRF) ist der Begriff Konfigurationen üblich.

2 Stand der Forschung zur Segmentierung

Das bedeutet, je größer das Bild I und die Anzahl von Labels $|L|$ ist, desto mehr Möglichkeiten m der Abbildung gibt es. Bei einer binären Segmentierung mit $L = \{0, 1\}$ gibt es folglich genau

$$m = 2^{|I|} \quad (2.27)$$

Möglichkeiten der Segmentierung. Allerdings gilt dies bei einer Objekt-Segmentierung nicht, weil das Bild in die Bildobjekte O und in die Nicht-Bildobjekt Pixelmenge \bar{O} zerlegt wird. Ein Bildobjekt O ist ein Segment und ein Segment besteht aus benachbarten Pixeln. Unter dieser Bedingung sind die Labels l nicht mehr unabhängig voneinander und m wird geringer ($m \leq |L|^{|I|}$). Die Anzahl der Möglichkeiten m_{os} im Fall der binären Objekt-Segmentierung ist in Abhängigkeit von der Segmentmenge⁵ SI berechenbar mit:

$$m_{os} = 2^{|SI|} \quad (2.28)$$

Für eine Labelmenge $|L| \neq 2$ gilt dies nicht, da sich Segmente nicht überschneiden können. Eine Objektdetektion ist im Gegensatz zur Segmentierung die Abbildung einer Segmentmenge auf eine diskrete Labelmenge ($SI \rightarrow L$). Die Anzahl der Möglichkeiten m_{od} bei der Objektdetektion ergibt sich durch die Potenzmenge \mathcal{P} , weil alle möglichen Segmente S in I einem Label zugeordnet werden. Auch sich überschneidende Segmente werden betrachtet, was im Unterschied zur Segmentierung nicht der Fall ist. Die Anzahl m_{od} ergibt sich durch:

$$m_{od} = |L|^{\mathcal{P}(SI)} \quad (2.29)$$

Unter der Voraussetzung des gleichen Bildes I und einer binären Labelmenge $|L| = 2$ ergibt sich:

$$m_{so} \leq m_{od} \quad (2.30)$$

Da eine Potenzmenge größer gleich der Kardinalität derselben Menge ist, ist auch die Anzahl der Möglichkeiten m_{od} einer Objektdetektion größer gleich der Anzahl der Möglichkeiten m_{os} einer Objekt-Segmentierung.

Zusammenfassung

Die Segmentierung bildet Pixel und die Objektdetektion Segmente auf Labels ab. Die Objektdetektion nutzt immer Vorwissen über ein Objekt. Hingegen wird bei der Segmentierung nur in einigen Fällen Vorwissen genutzt. Es hat sich gezeigt, dass die Objektdetektion ein Mittel zur Segmentierung ist. Die eingeführte Segmentierungsfunktion $f_{s|od}$ kennzeichnet eine Segmentierung durch Nutzung der detektierten Objekte.

Weiterhin hat sich gezeigt, dass der Lösungsraum der binären Segmentierung und Objektdetektion in Abhängigkeit der Mächtigkeit der Segmentmenge $|SI|$ exponentiell ansteigt. Die Mächtigkeit von $|SI|$ ist Abhängig von der Größe des Bildes. Je mehr Pixel das Bild enthält, umso größer der Lösungsraum.

⁵Die Berechnung der Anzahl der Elemente von SI in Abhängigkeit von L und I ist nicht trivial, wird hier aber nicht weiter betrachtet.

2.2.3 Fazit und Erarbeitung einer Grobstrukturierung

In diesem Abschnitt 2.2 wurden Vorbetrachtungen der Segmentierung und der Objektdetektion vorgenommen. Es wurde gezeigt, dass eine Schnittmenge zwischen Segmentierung und Objektdetektion existiert. Die Schnittmenge umfasst die Segmentierung von Objekten mit der Nutzung von Vorwissen. Es wurde auch herausgestellt, dass die Objektdetektion als Mittel zur Segmentierung genutzt werden kann. Weiterhin hat sich bei der Segmentierung und Objektdetektion ein exponentiell wachsender Lösungsraum herausgestellt.

Die Forschungsfelder Segmentierung und Objektdetektion sind sehr breit, wie in [Ton05] formuliert wird:

„Segmentierung ist ein Gebiet von enormer Breite und bei weitem nicht abgeschlossen. Bücher zur Bildsegmentierung behandeln das Thema in unterschiedlichem Umfang und setzen sehr verschiedene Schwerpunkte.“

Es existiert bereits eine große Anzahl an Publikationen zur Segmentierung und Objektdetektion, die stetig wächst. Ursache sind die vielen offenen Forschungsfragen und auch zwei unterschiedliche Herangehensweisen an die Problemstellung. Zum einen wird an den Grundlagen geforscht und zum anderen an anwendungsbezogenen Lösungen. In der **Grundlagenforschung** ist die Anwendung zweitrangig. Die grundlegenden Methoden werden in dieser Arbeit als **Basismethoden** bezeichnet und sind von der Segmentierungsaufgabe unabhängig. Einige Übersichten über Basismethoden sind bspw. in [PP93] [Rau+09] und [SK94] zu finden. Im Gegensatz zur Grundlagenforschung ist in der **anwendungsorientierten Forschung** eine Segmentierungsaufgabe definiert und die Anwendung steht im Mittelpunkt. Es gibt viele Anwendungen und folglich viele offene Fragestellungen und auch Lösungen für definierte Segmentierungsaufgaben. Die Methoden mit definierter Segmentierungsaufgabe werden in dieser Arbeit als **spezielle Methoden** bezeichnet. Beispiele für spezielle Methoden sind die Buchstabensegmentierung in [DW92], die spezielle Buchstabensegmentierung auf Autokennzeichen in [Ana+08] oder die Segmentierung von Haut in [VSA03].

Zur Erfassung des wesentlichen Standes der Forschung werden die Methoden im Folgenden unterteilt betrachtet. Die wesentlichen Basismethoden aus der Grundlagenforschung der Segmentierung und der Objektdetektion werden in Abschnitt 2.3 getrennt betrachtet. In Abschnitt 2.4 werden spezielle Methoden der Segmentierung von **organo**-Gruppen beschrieben. Da die Objektdetektion auch ein Mittel zur Segmentierung ist, wird bei den speziellen Methoden keine Unterscheidung mehr vorgenommen. Die Objektdetektion wird als Unterklasse der Segmentierung zugeordnet und der Abschnitt wird aus Sicht der Anwendung strukturiert. Welche Basismethoden bei den speziellen Methoden Anwendung finden, wird abschließend in Abschnitt 2.5 herausgearbeitet.

2.3 Basismethoden der Segmentierung und Objektdetektion

In diesem Abschnitt werden die wesentlichen Basismethoden der Segmentierung und Objektdetektion beschrieben. Es wird in Abschnitt 2.3.1 zuerst eine Klassifikation der Basismethoden vorgenommen. Die Basismethoden der Objektdetektion werden getrennt

2 Stand der Forschung zur Segmentierung

von der Segmentierung betrachtet. In Abschnitt 2.3.2 werden die herausgearbeiteten Klassen der Segmentierung und in Abschnitt 2.3.3 die Klassen der Objektdetektion erörtert. Die beschriebenen Klassen mit den entsprechenden Basismethoden werden abschließend in Abschnitt 2.3.4 zusammengefasst.

2.3.1 Klassifikation der Basismethoden

Die Segmentierung ist ein noch nicht abgeschlossenes Forschungsfeld ohne einheitliche Klassifikation. Es existieren stattdessen viele verschiedene Klassifikationen. So sind bspw. die Klassifikationen A-H aus den Büchern A: [Nal93], B: [Eff00], C: [Jah05], D: [Ton05], E: [Umb05], F: [Ste06], G: [Sze10] und H: [FP11] verschieden und dennoch in gewisser Weise ähnlich (Details siehe Anhang A.3). In den verschiedenen Klassifikationen A-H der Segmentierung werden verschiedene Abstraktionsebenen und Schwerpunkte gesetzt. Betrachtet man alle Klassifikationen zusammen als Menge und verbindet die Unterklassen einer Oberklasse der Menge miteinander, so ergibt sich die Abbildung 2.4.

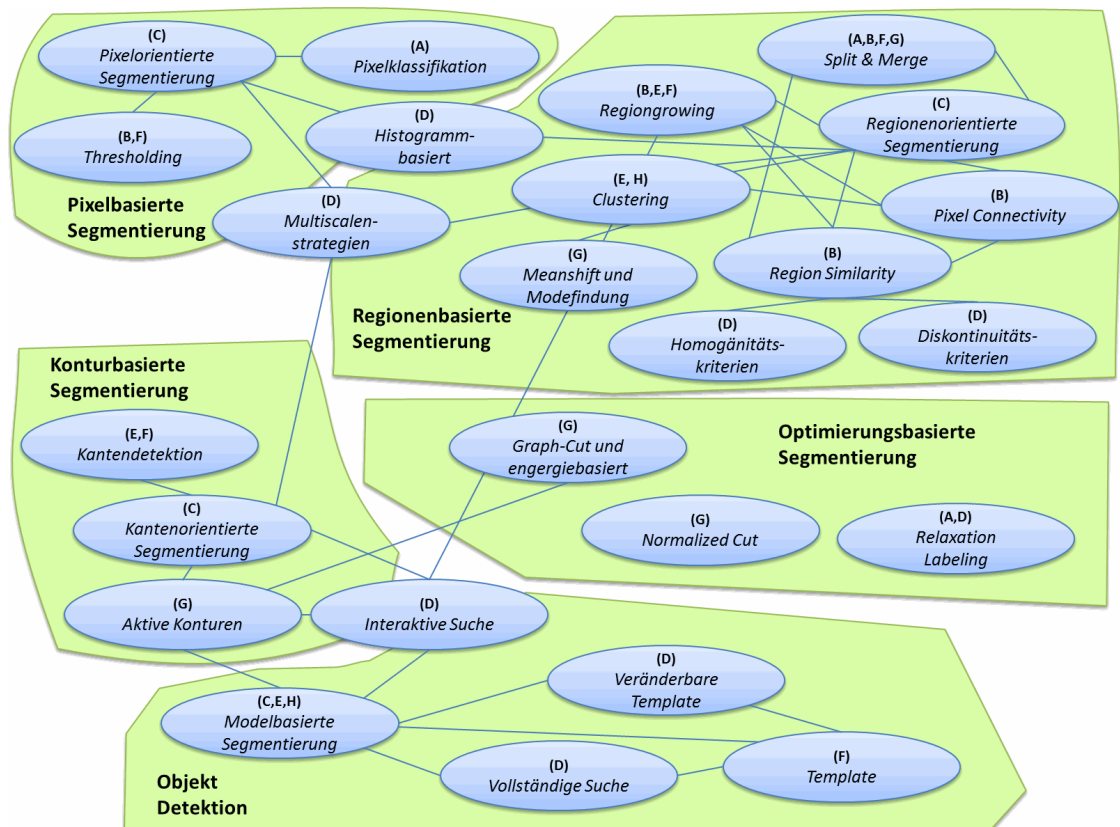


Abbildung 2.4: Die Abbildung zeigt die Klassen der Segmentierung aus den Büchern A-H (hellblaue Ellipsen), deren Beziehungen zueinander und die eigene Klassifikation (grüne Polygone).

In der Abbildung 2.4 sind die in den Lehrbüchern A-H aufgeführten Klassen mit hellblauen Ellipsen markiert. In den bestehenden Klassifikationen A-H wird die Objektdetektion der Segmentierung untergeordnet. Die Linien in Abbildung 2.4 verbinden eine Oberklasse mit den entsprechenden Unterklassen. Welches die Ober- und welches die Unterklasse ist, ist abhängig von der Sichtweise und daher nicht hervorgehoben. Die eigene Klassifikation dieser Arbeit ist mit grünen Polygonen gekennzeichnet. Im Ergebnis ergeben sich die Klassen

1. Pixelbasierte Segmentierung
2. Konturbasierte Segmentierung
3. Regionenbasierte Segmentierung
4. Optimierungsbasierte Segmentierung

Die Klassen werden im Folgenden genauer betrachtet. Die **Objektdetektion** wird separat betrachtet und im späteren Verlauf weiter unterteilt.

2.3.2 Basismethoden der Segmentierung

Im Folgenden werden die Basismethoden der Segmentierung nach der im Abschnitt 2.3.1 vorgenommenen Klassifikation beschreiben.

2.3.2.1 Pixelbasierte Segmentierung

Die pixelbasierten Segmentierungsmethoden ordnen jedem Pixel separat ein Label zu. Es wird *i.d.R.* eine binäre Segmentierung durchgeführt, womit die Labelmenge $L = \{0, 1\}$ nur zwei Elemente besitzt. Die Zuordnung beruht auf einer Bedingung pro Pixel, welche durch die nachfolgende Pixelfunktion f_s^{px} verdeutlicht werden soll.

$$f_s^{px}(px) = \begin{cases} 1, & \text{Pixelbedingung erfüllt} \\ 0, & \text{sonst} \end{cases} \quad (2.31)$$

Nachbarschaften werden meist nicht betrachtet und wenn doch, dann nur indirekt durch die Pixelbedingung. Die pixelbasierten Segmentierungsmethoden lassen sich in Schwellwertverfahren und Pixelklassifikation einteilen. Zusätzlich soll auch die Hintergrundsubtraktion in die Klasse der pixelbasierten Segmentierung eingeordnet werden, da pro Pixel binär segmentiert wird.

Schwellwertverfahren

Die Schwellwertverfahren (*Thesholding*) sind die bekanntesten und *i.d.R.* einfachsten Segmentierungsverfahren. Das Ergebnis ist stets eine binäre Segmentierung. Der Grauwertwert va jedes Pixels $px = (va)$ eines Grauwertbildes wird mit einem Schwellwert T wie folgt verglichen:

$$f_s^{px}(px) = \begin{cases} 1, & va < T \\ 0, & va \geq T \end{cases} \quad (2.32)$$

2 Stand der Forschung zur Segmentierung

Wird T überschritten, so erhält das Pixel das Label 0, andernfalls das Label 1. Der Schwellwert wird vom Nutzer oder automatisch durch lokale oder globale Bildanalyse bestimmt. Verschiedene Methoden zur Ermittlung eines Schwellwertes sind in [SS04] beschrieben. Das bekannteste globale Verfahren ist **Otsu's Methode** [Ots79]. Mit Otsu's Methode wird ein Schwellwert durch eine Varianzanalyse im Histogramm ermittelt. Schwellwertverfahren können auch auf Mehr-Kanal-Bildern angewandt werden. In diesem Fall wird entweder das Bild auf einen Kanal reduziert oder es wird pro Kanal ein Schwellwert angewandt. Es gibt auch Verfahren zur Bestimmung mehrerer Schwellwerte, wie es bspw. in [LCC01] beschrieben wird.

Pixelklassifikation

Bei der Pixelklassifikation wird jedes Pixel unabhängig der benachbarten Pixel durch einen Klassifikator einer Pixelklasse zugeordnet. Der Klassifikator wird entweder empirisch ermittelt oder mit Trainingsdaten durch Methoden des überwachten Lernens (*supervised learning*) (siehe [HTF01]) erstellt. Ein Pixelklassifikator kommt zumeist bei der Objekt-Segmentierung zur Anwendung und klassifiziert ein Pixel binär als Objektpixel oder Nicht-Objektpixel. Ein Farb- oder Grauwertbild wird durch den Klassifikator $f_k(px)$ wie folgt segmentiert:

$$f_k^{px}(px) = \begin{cases} 1, & \text{wenn } f_k(px) = \text{Objektpixel} \\ 0, & \text{sonst} \end{cases} \quad (2.33)$$

Eine Variante der Klassifikation sind explizite Regeln (siehe [KPS03], [Nal+07]). In [KPS03] wird ein Pixel anhand der Regeln als Haut oder Nicht-Haut klassifiziert. Die Regeln in [KPS03] entsprechen Schwellwertberechnungen über die Farbkanäle und sind empirisch ermittelt worden. In [Nal+07] werden verschiedene Methoden der Pixelklassifikation speziell zur Hautsegmentierung gegenübergestellt. Als Klassifikatoren werden *Multi-Layer-Perceptron* (MLP)⁶, Bayes Klassifikator (BK) und Gauß Klassifikator (GK) aus [PCB01] verwendet. Als Pixelklassifikatoren sind theoretisch alle Klassifikatoren des überwachten Lernens geeignet, wie Support Vector Maschine (SVM), Entscheidungsbäume (*Decision Trees* – DT) und Neuronale Netze (NN). Die Klassifikatoren können in [HTF01] nachgelesen werden. Theoretisch kann ein Pixelklassifikator auch mehr als zwei Klassen betrachten. Dieser Fall ist möglich, aber untypisch.

Hintergrundsubtraktion

Die Hintergrundsubtraktion (*background subtraction*) wird vorwiegend zur Segmentierung von bewegenden Objekten in Videos bei fixierter Kamera-Position angewandt. Der Grundgedanke besteht in der Annahme eines statischen Hintergrundes hinter sich bewegenden Objekten. Bei der Segmentierung in Videos baut sich das Hintergrundbild I_{bg} oder ein entsprechendes Hintergrundmodell über die Zeit auf. Bei der **Segmentierung in Einzelbildern** wird i.d.R. I_{bg} manuell erstellt. Für jedes Pixel px wird anschließend

⁶MLP ist eine Form der Neuronalen Netze (NN)

die Distanz zum Hintergrund ermittelt. Bei der einfachen Hintergrundsabstraktion wird das Bild wie folgt segmentiert:

$$f_{hs}^{px}(px) = \begin{cases} 1, & \text{wenn } d(px, px_{bg}) > T; px_{bg} \in I_{bg} \\ 0, & \text{sonst} \end{cases} \quad (2.34)$$

T ist ein Schwellwert und $d(px, px_{bg})$ eine Distanzfunktion vom Pixel px zum Hintergrundpixel px_{bg} . Die Methoden der Hintergrundsabstraktion werden nicht weiter betrachtet. Einige Methoden sind in [Pic04] und [Ben+08] zu finden.

Vor- und Nachteile der pixelbasierten Segmentierung

Der Vorteil der pixelbasierten Segmentierung liegt in der hohen Geschwindigkeit der Segmentierung. Zudem sind speziell die Schwellwertverfahren sehr einfach umzusetzen. Einfache Verfahren der Segmentierung können durchaus für eine Lösung einer Segmentierungsaufgabe ausreichend sein und sollten als erstes getestet werden. Nachteilig an pixelbasierten Segmentierungsmethoden ist, dass Salz- und Pfeffer-Rauschen (*Salt and Pepper*)⁷ zu Löchern im Segmentierungsergebnis führt. Die Löcher entstehen durch die Betrachtung eines Pixels unabhängig von seinen Nachbarn. Weiterhin werden Texturen und Gradienten nicht betrachtet, wodurch Objektgrenzen unscharf sein können.

2.3.2.2 Konturbasierte Segmentierung

Die Methoden der Segmentierung, die primär mit Konturen operieren, sollen als konturbasierte Segmentierungsmethoden aufgefasst werden. Eine Kontur (*Contour*) ist eine geschlossene Kurve und wird im Bildraum durch Konturpunkte diskretisiert (siehe Begriffsdefinition in Abschnitt 2.1). Die Segmentierung ergibt sich durch die berechneten Konturen ($C_1, \dots, C_i, \dots, C_n$) wie folgt:

$$f_s^{co}(px) = \begin{cases} 0, & \text{außerhalb aller Konturen} \\ i, & \text{innerhalb der Kontur } C_i \end{cases} \quad (2.35)$$

Die von einer Kontour C_i eingeschlossenen Pixel bilden das gesuchte Segment S_i . Die Konturen zu berechnen, ist auf unterschiedliche Weise möglich. Im Folgenden werden statische und dynamische Möglichkeiten der konturbasierten Segmentierung beschrieben.

Statische Konturen (Edge Linking, Hough-Transformation)

Statische Konturen ergeben sich durch die Analyse von Bildkanten. Die Methoden sind **i.d.R.** zweistufig und werden auch als **Edge Linking** bezeichnet. In der ersten Stufe werden Kanten im Bild extrahiert und in der zweiten Stufe geschlossene Konturen unter Hilfenahme der Kanten gesucht. **I.d.R.** wird im ersten Schritt ein Kantendetektor angewandt. Kanten im Bild lassen sich durch die erste Ableitung ermitteln. Approximiert und in 2D betrachtet wird die erste Ableitung mit dem Sobel oder Priwitt-Operator

⁷Hochfrequentes Rauschen, als ob Salz und Pfeffer über das Bild gestreut wurde (siehe auch [AJE10])

umgesetzt. Diese Operatoren (Filter-Masken) separieren starke Gradienten im Bild. Allerdings werden viele einzelne Kanten extrahiert, welche unterschiedliche Breiten aufweisen können. Eine Möglichkeit, dem entgegen zu wirken, ist der Canny-Kanten-Detektor (siehe [Can86]). Der Canny-Kanten-Detektor separiert in mehreren Stufen ein Pixel breite Kanten. Weitere Verfahren zur Kantendetektion sind in [SFM02] zu finden. Nach der Extraktion von Kanten werden die Kanten analysiert und zusammengefügt. Eine Variante der Realisierung ist die Verfolgung der Konturen. Ausgehend von einem Startpunkt wird entlang der Kanten eine geschlossene Kontur gesucht. Die Kanten, die eine geschlossene Kontur bilden, umschließen das gesuchte Segment. Zur Lösung der Mehrdeutigkeiten werden mitunter auch Algorithmen zur Bestimmung des kürzesten Pfades eingesetzt, wie zum Beispiel der A^* Algorithmus (siehe [DP85]). Weitere Ansätze des *Edge Linking* sind in [Hat+96] und [Hsu+10] zu finden.

Das Zusammenfügen von Bildkanten mit *Voting* ist Grundgedanke der **Hough-Transformation**. Die Hough-Transformation ist ein Klassiker zur Segmentierung von parametrierbaren geometrischen Figuren (siehe [Hou62] und [DH72]). Der Grundgedanke ist die Transformation der Parameter in den sogenannten **Houghraum**. Im Houghraum findet anschließend ein **Voting** statt. Mögliche Parameter einer Geraden sind bspw. die Paare (m, n) oder (d, α) , der Gleichungen:

$$f(x) = m * x + n \quad \text{oder} \quad d = x * \cos(\alpha) + y * \sin(\alpha) \quad (2.36)$$

Nach einer Kantenextraktion werden die Kantenpixel in den Houghraum überführt. Für jedes Kantenpixel werden für die Position (x, y) alle möglichen Parameterkombinationen zum Houghraum addiert. Durch das sogenannte Voting (Wert wird um 1 erhöht) ergeben sich Häufigkeiten. Eine Häufung (hohe Werte) kennzeichnet die Parameter der geometrischen Figur. Die Hough-Transformation ist in der ursprünglichen Form sehr rechenintensiv, ist nicht robust gegenüber Rauschen und benötigt ein gutes Kantenbild.

Dynamische Konturen

Dynamische Konturen hingegen ändern die Form, bis eine Bedingung erfüllt ist. Ein Verfahren dieser Kategorie sind die **aktiven Konturen** oder analog auch *Snakes*. Ausgehend von einer initialen Kontur wird die Kontur iterativ neu bestimmt, bis die Energiefunktion E_{snake}^* minimal wird. In [KWT88] wird die aktive Kontur definiert durch

$$E_{snake}^* = \int_0^1 E_{snake}(c(s)) = \int_0^1 E_{int}(c(s)) + E_{image}(C(s)) + E_{con}(c(s)) ds, \quad (2.37)$$

wobei $c(s)$ die Kontur in expliziter Parameterdarstellung mit dem Parameter s ist (siehe Abschnitt 2.1). Die Energiefunktion E_{snake}^* besteht aus drei Energietermen. Die interne Energie E_{int} beschreibt die Krümmungskräfte der Kontur, die externen Energien E_{image} die Bildkräfte und E_{con} sonstige externe Zwänge (*Constrains*). E_{int} wird i.d.R. minimal, wenn die von $c(s)$ überlagerten Pixel hohe Gradienten aufweisen, E_{int} wird zumeist durch die 1. und 2. Ableitung von $c(s)$ beschrieben. Die konkrete Definition der Energieterme

ist in der Literatur unterschiedlich und Gegenstand der Forschung. Einige typische Definitionen sind im Anhang A.2 beschrieben. Eine Variante der aktiven Konturen ohne Bestimmung der Gradienten ist in [CV01] beschrieben. Eine weitere Methode der dynamischen Konturen ist die **level-set** Methode. Das Problem wird in den 3D-Raum auf Oberflächen übertragen. Der Schnitt der Oberfläche mit der Bildebene ergibt die Kontur. Weiteres zu level-set kann in [OF03] nachgelesen werden.

Vor- und Nachteile der konturbasierten Methoden

Die konturbasierten Methoden betrachten die Objektgrenzen und lassen Farbe und Textur der Objekte außer Acht. Bei den statischen Konturen sind Mehrdeutigkeiten bei der Kontursuche möglich. Problematisch sind auch Lücken in den Konturen. Ein Lösungsansatz bzgl. Konturlücken wird in [Tsu+05] beschrieben. Die Hough-Transformation hat den Nachteil, dass nur nahezu geometrisch perfekte Kreise, Linien oder Ellipsen detektiert werden können. Die Kanten des Objektes müssen scharf und klar abgegrenzt sein. Dynamische Konturen sind robuster gegenüber Lücken in den Konturen. Allerdings sind die dynamischen Methoden halbautomatisch, benötigen eine initiale Kontur und dem Objekt angepasste Energieterme.

2.3.2.3 Regionenbasierte Segmentierung

Die vermutlich umfangreichste Klasse der Segmentierung ist die regionenbasierte Segmentierung. Regionenbasierte Methoden zerlegen ein Bild oft in wenigen Schritten in n Segmente, die sogenannten Regionen. Eine Region wird manchmal auch als Superpixel bezeichnet. In [Ach+12] werden Superpixel beschrieben mit:

„superpixel algorithms group pixels into perceptually meaningful atomic regions“

In dieser Arbeit wird weiterhin der Begriff Segment verwendet. Ein Segment soll zugleich ein analoger Begriff zu Superpixel und Region sein. Regionenbasierte Methoden gruppieren benachbarte Pixel anhand gewisser Kriterien zu einem Segment. Die Kriterien sind angelehnt an Ähnlichkeiten der Pixel, der Homogenität oder Diskontinuität eines Segments, wie in [LM98] beschrieben:

„Region-based image segmentation techniques make use of similarity in intensity, color and texture to determine the partitioning of an image.“

Die regionenbasierte Segmentierung lässt sich nicht klar getrennt in wenige Klassen einteilen, da die Methoden und Herangehensweisen sehr verschieden sind. Es gibt aber bewährte Verfahren, die im Weiteren erläutert werden. Im Nachfolgenden werden die Klassiker der Segmentierung **Region Growing**, **Watershed**, **Split & Merge** sowie **Clustering-** und **sonstige Methoden** beschreiben.

Region Growing

Ein Klassiker der Segmentierung ist das Wachsen von Regionen, genannt *Region-Growing*. Das *Region Growing* (siehe [AB94]) ist eine halbautomatische Methode und benötigt

2 Stand der Forschung zur Segmentierung

einen sogenannten **Saatpunkt** (*seed point*). Ausgehend vom Saatpunkt werden iterativ benachbarte Pixel zu einem Segment gruppiert. Das Segment wächst so lange, bis ein gewisses Ähnlichkeitskriterium in Bezug auf die benachbarten Pixel nicht mehr erfüllt ist. Eine triviale Variante von *Region Growing* ist *Floodfill*. Bei *Floodfill* wird **i.d.R.** von einem Saatpunkt ausgegangen. Alle Pixel, deren Differenz zum Saatpunkt einen Schwellwert nicht überschreiten, werden zu dem Segment hinzugefügt. Dieses Verfahren separiert allerdings nur ein Segment und lässt die anderen Pixel unbetrachtet.

Watershed

Ein weiterer Klassiker der Segmentierung ist die Wasserscheiden-Segmentierung (*Watershed*). Die Grundidee von *Watershed* (siehe [BL79], [Beu91]) ist, dass die Pixelwerte eines Bildes als Höhenwerte betrachtet werden und das Gebirge mit Wasser geflutet wird. Während des iterativen Prozesses bilden sich wachsende Seen. Stoßen zwei Seen aneinander, so bilden sich Segmentgrenzen (Wasserscheiden). Der Prozess wird solange durchgeführt, bis das Bild komplett geflutet ist. Verschiedene Algorithmen zu *Watershed* sind in [RM00] beschrieben. Viele Varianten der Wasserscheiden-Segmentierung sind halbautomatisch und benötigen Saatpunkte. Die Saatpunkte bestimmen die Anzahl und ungefähre Lage eines Segments, wodurch der Bildung von zu vielen kleinen Segmenten entgegengewirkt wird. Bei *Watershed* Varianten mit Saatpunkten wird nicht selten ein *Region Growing* Ansatz zu Grunde gelegt.

Split & Merge

Eine automatische Segmentierungsmethode und ebenfalls ein Klassiker ist *Split & Merge* (siehe [HP74], [HS79], [KG08]). Mit der *Split & Merge*-Methode wird das Bild erst in kleine Segmente zerlegt (*Splitting*). Anschließend werden die Segmente nach der Zerlegung wieder zusammengeführt (*Merging*). Der Standardansatz beinhaltet die Nutzung einer Quadtree-Datenstruktur. Das Bild wird solange in rechteckige Segmente zerlegt bis ein Homogenitätskriterium erfüllt ist. Ähnliche Segmente werden anschließend wieder zu einem Segment verschmolzen.

Clustering-Methoden

Zu den regionenbasierten Segmentierungsmethoden soll auch das *Clustering* zugeordnet werden. Beim *Clustering* werden Pixel mit ähnlichen Eigenschaften durch Methoden des *Clusterings* gruppiert. Cluster werden von Pixeln im Bild-, Merkmals- und/oder Farbraum berechnet.

Ein Klassiker und leicht umzusetzendes Verfahren ist *K-Mean* (siehe [Ste56], [HW79], [Llo82]). Ausgehend von k Startpunkten wird jedem Startpunkt das dichteste Pixel zugeordnet. Die Zuordnung erfolgt anhand eines Abstandmaßes. Es wird nach der ersten Zuordnung der Mittelwert jedes Clusters bestimmt. Der Mittelwert ergibt ein neues Clusterzentrum. Iterative werden die Zentren bis zur Konvergenz neu bestimmt. Der Standard *K-Mean* hat eine Komplexität von $O(kNI)$, wobei k die Anzahl der Cluster, N die Anzahl der Pixel und I die Anzahl der Iterationen ist. Eine beschleunigte Abwandlung

von *K-Mean* ist die **SLIC**-Methode (*Simple Linear Iterative Clustering*). In der **SLIC**-Methode (siehe [Ach+12]) wird das Bild zuerst in regelmäßige Blöcke eingeteilt. In jedem Block befindet sich in der Mitte einer von k Startpunkten. Um Startpunkte auf Kanten zu vermeiden, wandern die Startpunkte zum Pixel mit dem geringsten lokalen Gradienten hin. Jedes Pixel wird anhand eines Distanzmaßes einem Cluster zugeordnet und die Cluster werden neu bestimmt. Iterativ wird der Prozess (Pixelzuordnung und Clustering) wiederholt bis ein Fehlermaß unterschritten ist (im Durchschnitt ca. 10 Iterationen). Bei der **SLIC**-Methode ist der Suchraum im Bild reduziert, wodurch laut [Ach+12] **SLIC** eine lineare Komplexität von $O(N)$ hat. Bei **SLIC** nach [Ach+12] wird der CIELAB Farbraum betrachtet. Von [Dun73] wurde das Verfahren *Fuzzy C-Mean* (**FCM**) entwickelt (siehe auch [Bez81]). Es wird bei der **FCM** keine scharfe Zuordnung der Pixel vorgenommen, sondern eine unscharfe (Fuzzy). Zugehörigkeiten (Gewichte) eines Pixels werden zu jedem Cluster ermittelt. Der gewichtete Abstand zu allen Clusterzentren wird iterativ optimiert. Ein Verfahren, das keine Startpunkte benötigt, ist *Mean Shift* (siehe [FH75] und [CM02]). Es wird bei *Mean Shift* ein Fenster (Parzen Window) um jedes Pixel aufgespannt. Der Durchschnitt (Mean) der Pixel im Fenster wird bestimmt und das Fenster solange verschoben, bis eine Konvergenz erreicht wird. Die Pixel, deren Fenster sich überlappen, werden zu einer Region zusammengefügt.

Texturbasierte Methoden

Die texturbasierten Methoden sind nicht scharf von anderen Methoden trennbar und sehr verschieden. Die allgemeine Herangehensweise gliedert sich in:

1. Extraktion von lokalen Texturmerkmalen
2. Segmentierung durch Auswertung der Merkmale

Methoden zur Bestimmung von lokalen Texturmerkmalen sind bspw. der Gaborfilter (siehe [Mov08], eine fensterbasierte Fouriertransformation), *Local Binary Pattern* (**LBP**, siehe [GZZ10]), Merkmale der *Gray Level Co-occurrence Matrix* (**GLCM**, siehe [HSD73]) und Bildstatistiken der ersten und zweiten Ordnung (siehe [PCR10]). Die Auswertung und folglich Segmentierung des Bildes mit Hilfe der Texturmerkmale erfolgt durch *Clustering*, Klassifikation oder Kombination mit einer anderen Segmentierungsmethode. Beim *Clustering* werden Cluster im Merkmalraum ermittelt. Das Label eines Pixels ergibt sich **i.d.R.** durch den geringsten Pixel-zu-Cluster-Abstand. Bei der Klassifikation ist eine Möglichkeit, das Bild in gleichförmige Teilbilder zu zerlegen und anschließend jedes Teilbild anhand der Texturmerkmale zu klassifizieren. Dieses Vorgehen ist nach der Definition in Abschnitt 2.1 keine Segmentierung mehr, sondern eine Objektdetektion (siehe auch Abschnitt 2.3.3). Es gibt vielfältige Kombinationen mit anderen Methoden. So wird **z.B.** in [ZB02] Textur mit Farbe kombiniert und in [OP97] Textur mit einem Split & Merge-Ansatz. Da Texturen sich typischerweise durch Inhomogenität auszeichnen, kann eine pixelweise Zuordnung von Pixel zu Label zu unscharfen und verrauschten Ergebnissen führen. Um dem entgegenzuwirken, werden **MRF** in die Segmentierung einbezogen. Bei **MRF** sind die Zustände (Labels) von den Zuständen der benachbarten Pixel abhängig.

2 Stand der Forschung zur Segmentierung

Modelliert wird dies durch bedingte Wahrscheinlichkeiten. In dieser Arbeit sind **MRF** auch in die optimierungsbasierten Methoden eingeordnet.

Sonstige Methoden

Regionenbasierte Segmentierung ist auch durch Multi-Skalen-Analyse möglich. Eine Multi-Skalen-Methode ist *Pyramid Linking* (siehe [PRD96]). Das Bild wird in verschiedenen Auflösungen betrachtet und eine Bildpyramide aufgebaut. Eine Standardvariante ist der Aufbau einer Gaußpyramide wie in [Bur83] vorgeschlagen. Der Mittelwert von vier Pixeln bildet eine Ebene höher ein Pixel. Die Pixel der Bildpyramide werden beim *Pyramid Linking* verbunden. Die Verbindungen werden anhand verschiedenen Kriterien iterativ ausbalanciert. Zusammenfallende Verbindungen innerhalb der Pyramide ergeben ein Segment. Es gibt noch viele weitere, weniger bekannte regionenbasierte Methoden, als die bereits beschriebenen. Eine Betrachtung weiterer Methoden wird hier nicht vorgenommen, da die Vielzahl der Methoden den Rahmen der Arbeit sprengen würde. Zu den weiteren Methoden zählen bspw. *Quickshift* in [VS08] oder die Turbopixel Methode in [ZK07].

Vor- und Nachteile der regionenbasierten Methoden

Die Vor- und Nachteile der Methoden sind verschieden, sehr individuell und kaum zu abstrahieren. Meistens führen die automatischen Methoden zu einer Über- oder Untersegmentierung, wodurch erst eine Nachverarbeitung, wie eine Regionenzerlegung oder Regionenvermelzung, zum gewünschten Ergebnis führt. Nachteilig an den halbautomatischen Verfahren (z.B. *Region Growing*) ist, dass Saatpunkte benötigt werden, und folglich Vorwissen über die Objekte benötigt wird. Vorteilhaft an den meisten Verfahren ist, dass **i.d.R.** Bildkanten die Regionen begrenzen und eine Farbähnlichkeit innerhalb einer Region besteht. Allerdings sind die meisten Verfahren sensible gegenüber Bildrauschen.

2.3.2.4 Optimierungsbasierte Segmentierung

In der **CV** gibt es Methoden der Segmentierung, die die Zerlegung eines Bildes optimieren. Nachbarschaften und Pixelwerte gehen **i.d.R.** gemeinsam in die Optimierung ein. Die Bezeichnung „optimierungsbasiert“ ist nicht allgemein üblich. Sie wird in dieser Arbeit verwendet, weil sie das Verständnis erleichtert, die Methoden gut zusammenführt und aus Sicht der Eingabedaten eine Optimierung errechnet wird. Die Eingabedaten sind ein bereits segmentiertes Bild und/oder ein bis zwei Wahrscheinlichkeitsbilder I^p . Ein Wahrscheinlichkeitsbild I^p ist in dieser Arbeit ein Ein-Kanal-Bild. I^p enthält zu jedem Pixel eine korrespondierende Wahrscheinlichkeit p , wie bspw. Vorder- oder Hintergrundwahrscheinlichkeit. Die Methoden der optimierungsbasierten Segmentierung legen ein statistisches Modell zugrunde, die Markov Random Fields (**MRF**) (siehe [BK04], [FH04], [RB05], [Lan+06], [AF07]). Sie untergliedern sich in **Graph-Cut**, **Belief Propagation** und **weitere Methoden** (siehe [Lan+06]). Weiterhin gibt es zwei theoretische Sichtweisen der Problemstellung, die Sicht der Wahrscheinlichkeitstheorie und der Graphentheorie. In dieser Arbeit wird das Thema primär aus Sicht der **Graphentheorie** betrachtet. Im

folgenden werden zuerst Graphen, Cliques und **MRF** erklärt. Anschließend werden die Methodenklassen einzeln beschrieben.

Graphen, Cliques und Markov Random Fields

Ein **Graph** $G = (V, E)$ besteht aus Knoten (vertices) V und Kanten (edges) E . Die Knoten V sind über die Kanten E verbunden. Gibt es zu den Kanten $e \in E$ korrespondierende Gewichte $w \in W$, so spricht man von einem **gewichteten Graphen**. Je nachdem, ob die Kanten gerichtet sind oder nicht, spricht man von einem gerichteten oder ungerichteten Graphen. Wird ein Bild I als Graph betrachtet, so entspricht ein Knoten $v \in V$ einem Pixel $px \in I$ und eine Kante e verbindet zwei benachbarte Knoten v . Abhängig von der Segmentierungsmethode ergeben sich ergänzende Knoten v und/oder weitere Kanten e . In der Segmentierung werden **i.d.R.** ungerichtete Graphen betrachtet.

Eine **Clique** Cl eines Graphen G ist eine Teilmenge von Knoten $Cl \subseteq V$ mit der Bedingung, dass für alle zwei Knoten $v_1, v_2 \in Cl$ eine Kante e existiert. Eine spezielle Form der Clique ist die **maximale Clique**. Eine maximale Clique ist eine Clique, die durch hinzufügen von benachbarten Knoten nicht erweitert werden kann, da bereits alle möglichen Knoten enthalten sind.

Ein **Markov Random Field MRF** ist (siehe [Li95], [Bou95], [Fig98], [Lan+06], [Sri11], [Kim+11]) ein statistisches Modell. Graphentheoretisch ist ein **MRF** ein ungerichteter Graph $G = (V, E)$. Jeder Knoten des Graphen korrespondiert mit einer Zufallsvariable X_v der Menge

$$X = (X_v)_{v \in V}, \quad (2.38)$$

wobei v die Menge der Knoten V indiziert.

Eine Kante e im Graph verbindet zwei Knoten, deren korrespondierenden Zufallsvariablen voneinander stochastisch bedingt abhängig sind. Ein **MRF** erfüllt die drei **Markov Eigenschaften** (siehe [Sri11]) und modelliert in der **CV** die Wahrscheinlichkeit eines Bildes I . Nach [Lan+06] kann ein **MRF** durch das **Produkt aller maximalen Cliques**

$$P(I) = \frac{1}{Z} \prod_{Cl} \psi(I_{Cl}) \quad (2.39)$$

beschrieben werden, wobei I_{Cl} die Bildregion (**RoI**) korrespondierend zu der Clique Cl , ψ eine positive Potenzialfunktion und Z ein Normalisierungsfaktor ist. **MRF** unterteilen sich in **Paarweise-MRF** (*pairwise MRF*) und in **MRF höherer Ordnung** (*higher-order MRF*). Im paarweisen Fall verbinden Kanten e benachbarte Knoten v in horizontaler und vertikaler Richtung. Das heißt im Bildraum wird eine Vierer-Nachbarschaft betrachtet. In einem **MRF** höherer Ordnung hingegen sind alle Knoten v aller möglichen quadratischen $n \times n$ **RoIs** eines Bildes I durch eine Kante e miteinander verbunden.

Graph-Cut

Graph-Cut ist eine graphenbasierte Segmentierungsmethode (siehe [BJ01], [BK04] und [FH04]) und ist eine Variante der approximierten Schlussfolgerung (*probabilistic inferenz*)

2 Stand der Forschung zur Segmentierung

auf **MRF** Modellen. Das als Graph-Cut bekannte Segmentierungsverfahren basiert auf der Teilung eines über dem Bild aufgespannten Graphen $G = (V, E)$. In der Graphentheorie wird die Teilung **Cut** (Schnitt) genannt. Ein *Cut* ist die Zerlegung der Knotenmenge V in disjunkte Teilmengen. Der **Min-Cut** (minimaler Schnitt) ist gegeben, wenn die Anzahl der geschnittenen Kanten nicht größer als die eines anderen Schnittes ist. Im Fall eines gewichteten Graphen wird die Summe der Gewichte w der geschnittenen Kanten minimiert (siehe Abbildung 2.5).

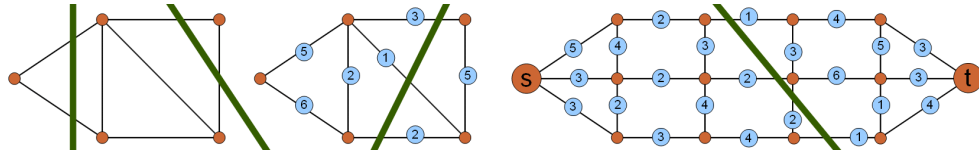


Abbildung 2.5: Links: *min-cuts*; Mitte: gewichteter *min-cut*; Rechts: *s/t-cut*

Der typische Graph-Cut in der Bildverarbeitung entspricht einem **s/t-cut** eines gewichteten Graphen (siehe Abbildung 2.5). Ein *s/t-cut* ist der minimale Schnitt eines Graphen $G = (V, E)$, welcher zusätzlich die Terminal-Knoten $s \in V$ (source) und $t \in V$ (sink) besitzt. In der Graphentheorie spricht man dann auch von einem Netzwerk. Der minimale s/t Schnitt eines gewichteten Graphen ist auf das Problem des maximalen Flusses zurückzuführen, was auch als **Min-Cut/Max-Flow** Theorem bekannt ist (siehe [EFS56]). Ausgehend von s hin zu t wird der maximale Fluss im Netzwerk bestimmt. Dort, wo sich der maximale Fluss staut, befindet sich ebenfalls der minimale Schnitt. Demzufolge haben die Kanten auch verschiedene Typen von Gewichten (siehe Abbildung

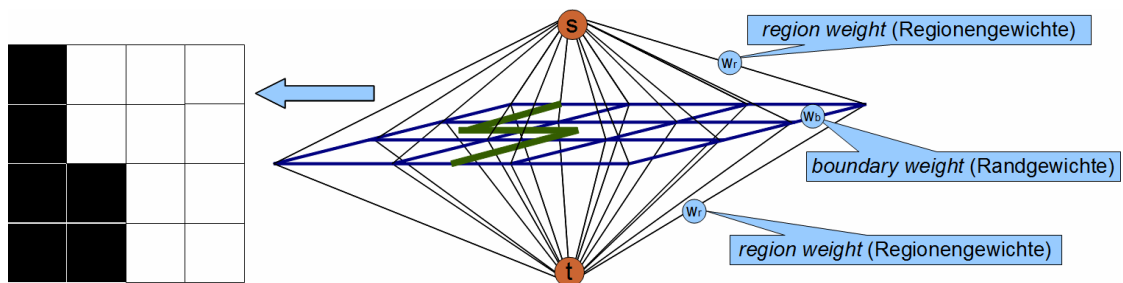


Abbildung 2.6: Veranschaulichung der Graph-Cut Segmentierung; Links: segmentiertes Bild I^b ; Mitte: der Graph G und der *min-cut*; Rechts: die Typen der Gewichte der Kanten E von G .

2.6). Sie werden auch als *region* (w_r) und *boundary weights* (w_b) bezeichnet, wobei $w_r, w_b \in W$. w_b steht für die Pixel-zu-Pixel-Nachbarschaft und w_r für die Beziehung eines Pixel zu Objekt und Nicht-Objekt. Alle Gewichte w_r eines Typs sollen auch zu einem Wahrscheinlichkeitsbild I^p zusammengefasst werden. Prinzipiell ergibt sich genau ein Wahrscheinlichkeitsbild für Objekt (Vordergrund) I_{fg}^p und eines für Nicht-Objekt

(Hintergrund) I_{bg}^p . Graph-Cut ist in ursprünglicher Form ein halbautomatisches Segmentierungsverfahren. Anhand von markierten Objekt und Nicht-Objekt-Pixeln werden Farbmodelle ermittelt, wodurch die Wahrscheinlichkeitsbilder I^p erstellt werden.

Eine Nutzervereinfachung zur Markierung der Objekt und Nicht-Objekt Pixel wird in [RKB04] mit dem sogenannten *Grab-Cut* vorgestellt. Bei *Grab-Cut* werden die Wahrscheinlichkeiten (Gewichte) lediglich auf Basis eines vom Nutzer um das Objekt gesetzten Rechtecks ermittelt. Mit den Pixeln innerhalb und außerhalb des Rechtecks werden *Gaussian Mixture Models* (**GMM**) iterativ berechnet. Die Abstände (Mahalanobis-Distanz) jedes Pixels zu den **GMM** ergeben die Wahrscheinlichkeiten (Gewichte).

Beliefpropagation

Beliefpropagation, auch bekannt als *sum-produkt message passing*, ist eine weitere Variante der approximierten Schlussfolgerung (*probabilistic inferenz*) auf **MRF** Modellen. Die Grundidee ist der Austausch von sogenannten Nachrichten (*messages*) μ zwischen benachbarten Knoten eines **Faktor-Graphen** G_f über das angenommene (*belief*) Label des Empfängerknotens. Ein Faktor-Graph ist ein **bipatiter Graph** $G_f = (F, V, E)$ und besteht aus den disjunkten Knotenmengen F (Faktoren) und V (Variablen), wobei eine Kante $e \in E$ einen Knoten aus F mit einem in V verbindet. Nach dem **Hammersley-Clifford Theorem** (siehe [HC71]), kann ein **MRF** durch einen Faktor-Graph $G_f = (F, V, E)$ beschrieben werden. Aufgrund der zwei Knotenmengen (F, V) ergeben sich zwei Typen von Nachrichten, die Nachricht $\mu_{f \rightarrow v}$ von einem Faktor $f \in F$ an eine Variable $v \in V$ und der umgekehrte Fall $\mu_{v \rightarrow f}$. Die Nachricht $\mu_{v \rightarrow f}$ ergibt sich aus dem Produkt aller benachbarten Faktor-Knoten $(N(v) \setminus \{f\})$ außer f selbst, und $\mu_{f \rightarrow v}$ ergibt sich aus der Summe aller benachbarten Variablen-Knoten $(N(f) \setminus \{v\})$ außer v selbst. Die Nachrichten werden, bis zur Konvergenz oder bis ein Abbruchkriterium erreicht ist, iterativ ausgetauscht. Es gibt verschiedene Methodiken des Nachrichtenaustausches, die hier nicht weiter ausgeführt werden und in [SKS01], [YFW03], [FH06] und [Lan+06] zu finden sind.

Weitere Methoden

Eine weitere graphenbasierte Methode ist **Normalized-Cut** (**N-Cut**; siehe [SM00b]). Im Unterschied zum Graph-Cut, ist der Graph beim **N-Cut** nicht durch feste Nachbarschaften definiert und nicht regulär. Der Graph wird bei **N-Cut** durch die *Laplacian-Matrix* definiert. Weiterhin gibt es Methoden, die **MRF** Modelle (siehe [GR12]) anhand von Trainingsdaten anlernen, wie bspw. **Fields of Expert** (**FoE**) in [RB05]. Eine weitere optimierungsbasierte Methode ist **Relaxation Labeling** (siehe [HZ83]).

Vor- und Nachteile der optimierungsbasierten Methoden

Der Vorteil der optimierungsbasierten Methoden liegt in der Genauigkeit der Segmentierung. Allerdings benötigen die Methoden Vorwissen in Form von Modellen, Wahrscheinlichkeitsbildern und/oder bereits segmentierten Bildern. Erst in Kombination mit einem anderen Segmentierungsverfahren und/oder von Nutzerinteraktion wird eine Segmentierung ermöglicht. Auch ist die benötigte Rechenzeit **i.d.R.** recht hoch.

2.3.3 Basismethoden der Objektdetektion

Die Objektdetektion in Bildern hat sich mittlerweile zu einem eigenständigen jungen Forschungsfeld im maschinellen Sehen entwickelt. Im Folgenden werden die Klassiker **Template Matching**, **Haarkaskaden**, **Eigenobjekte** und **HoG** beschrieben. Weiterhin werden **Texturbasierte Methoden** und ausgewählte **sonstige Methoden** kurz dargelegt.

Bei der Objektdetektion wird nach der Definition in Abschnitt 2.1 das Objekt anhand eines Modells im Bild gesucht. Das Modell enthält Vorwissen über Farbe, Form und/oder Textur des Bildobjektes O und/oder dem realen Objekt O^r . Das Modell wird **i.d.R.** mit Hilfe von Trainingsdaten erstellt oder angelernt. Das prinzipielle Vorgehen der meisten Methoden der Objektdetektion kann in die folgenden drei Schritte gegliedert werden:

1. **Modellerstellung**: Erstellung eines Modells von O und/oder O^r
2. **Bildsuche oder Transformation**
 - a) **Bildsuche**: Traversieren des Bildes und Bestimmung von **RoIs** (S^{win})
 - b) **Transformation** des Bildes in einen Merkmalsraum
3. **Klassifikation** von S^{win} oder Analyse des Merkmalsraumes anhand des Modells

Alle Schritte sind je nach Methode sehr verschieden. Schritt 2.a) ist in vielen Methoden fensterbasiert (siehe [LBH08]). Ein Fenster (*sliding window*) wird über das Bild geschoben, und der rechteckige Bereich von Interesse im Fenster in Objekt (Obj) oder Nicht-Objekt ($\neg Obj$) klassifiziert. Alle Bereiche von Interesse **RoIs** sind auch rechteckige Segmente S^{win} und sollen durch die Menge SI^{win} zusammengefasst werden. Die Menge SI^{win} ist eine Teilmenge der Segmentmenge SI . Für den Fall des fensterbasierten Vorgehens soll die Objektdetektionsfunktion f_{od} spezialisiert werden zu:

$$f_{od}^{win} : SI^{win} \rightarrow L, S^{win} \mapsto l \quad (2.40)$$

Für die Variablen der Funktion $l = f_{od}^{win}(S^{win})$ gelten folgende Definitionen:

$$S^{win} \in SI^{win}; l \in L; SI^{win} \subset SI; L = \{Obj, \neg Obj\} \quad (2.41)$$

Template Matching

Das *Template Matching* ist ein Paradebeispiel für das fensterbasierte Vorgehen. Es wird ein Fenster über das Bild geschoben und ein *Template*-zu-Fenster-Vergleich vorgenommen. Das *Template* ist **i.d.R.** ein Graustufenbild und repräsentiert das Modell des Bildobjektes O . Das Graustufenbild wird mit S^{win} direkt Pixel für Pixel verglichen. Zur Auswertung des Vergleichs werden Ähnlichkeitsmaße eingesetzt, wie bspw. die Summe der absoluten Differenzen. Ist eine gewisse Ähnlichkeit gegeben, so wird S als Objekt (Obj) klassifiziert, ansonsten als Nicht-Objekt ($\neg Obj$). Es gibt viele Abwandlungen des *Template Matching*. Einige können in [Bru09] nachgelesen werden. Die Standardversion des *Template Matching* ist nicht rotations- und skalierungsinvariant. Zur Schaffung einer Skalierungsinvarianz wird im Bild mit *Templates* verschiedener Skalierungen gesucht.

Eigenobjekte

Bei Eigenobjekten oder auch *Eigen Images* ist die Reduktion des Bildinhaltes der Grundgedanke. Der Bildinhalt wird durch eine Transformation in den sogenannten **Eigenraum** auf das Wesentliche reduziert. Im Gegensatz zum *Template Matching* ist kein Bild das Modell, sondern ein sogenanntes Eigenobjekt. Das Modell wird aus mehreren Trainingsbildern gleicher Größe (w, h) erstellt. Jede Pixelposition bildet ein Merkmal und folglich eine Achse im Merkmalsraum. Es ergibt sich ein Merkmalsraum mit $w * h$ Dimensionen, welcher auf die wesentlichsten Informationen reduziert wird (siehe auch [MK82]). Zur Reduktion ist die **PCA** oder auch Karhunen-Loeve-Transformation (siehe [AF92]) üblich. Das Vorgehen beruht auf der Ermittlung von Eigenwerten und -vektoren, weshalb das erstellte Modell auch als **Eigenobjekt** bezeichnet wird. Nach der Erstellung eines Eigenobjektes (Schritt 1) wird **i.d.R.** fensterbasiert im Bild gesucht (Schritt 2). Zur Klassifikation von S^{win} wird S^{win} in den Eigenraum überführt und mit dem Eigenobjekt verglichen (Schritt 3). Anwendung findet das Vorgehen in der Detektion und Erkennung von Gesichtern (*Eigenfaces*), wie in [YKA02], [PMS94] und [SK87] beschrieben.

Haarkaskaden (Boosting)

Haarkaskaden (siehe [VJ01]) sind ursprünglich zur Detektion von Gesichtern entwickelt worden und finden zunehmenden Einsatz in der Objektdetektion. Die Grundidee ist das Hintereinanderschalten von schwachen Klassifikatoren (Haar-Templates) zu einer Kaskade, dem sogenannten Boosting. Die Haar-Templates (siehe Abbildung 2.7) bestehen aus wenigen rechteckigen Regionen, womit die Summe der Pixel verglichen wird. Zum

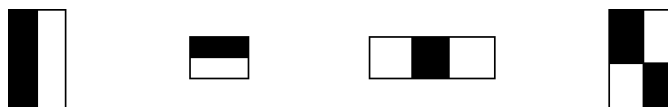


Abbildung 2.7: Schwache Klassifikatoren (Haar-Templates) der Haarkaskaden.

Vergleich wird ein sogenanntes Integralbild verwendet. Ein Pixel enthält im Integralbild die Summe der Vorgängerpixel. Anhand des Integralbildes ist ein Vergleich mit nur wenigen Operationen möglich. Der damit echtzeitfähige Ansatz detektiert ein Objekt, wenn alle Haar-Templates passen. Das Modell in dieser Methode entspricht der Haarkaskade. In der Modellerstellung (Schritt 1) wird eine Haarkaskade mit mehreren positiven und negativen Samples erstellt. Aus den Samples werden Ähnlichkeiten in Form und Intensität ermittelt. Die verschiedenen Ähnlichkeiten werden als Haar-Templates in die Haarkaskade eingespeichert. Gesucht wird **i.d.R.** fensterbasiert (Schritt 2) und S^{win} wird mit der Haarkaskade im Integralbild klassifiziert (Schritt 3).

Histogramm der orientierten Gradienten (HoG)

Histogramme der orientierten Gradienten **HoG** verschlüsseln Formmerkmale eines Objektes durch ein Histogramm und ermöglichen die Detektion anhand der Objektform.

2 Stand der Forschung zur Segmentierung

Das Bild wird in Blöcke eingeteilt und die Blöcke weiter in Zellen. Für jede Zelle wird ein Histogramm über die Gradienten oder die Kantenpixel ermittelt. Die Kombination der Histogramme ergeben einen *Descriptor*. Der *Descriptor* enthält die Merkmale und repräsentiert *i.d.R.* einen Block. Der *Descriptor* entspricht in diesem Ansatz dem Modell (Schritt 1) und wird mit Trainingsbildern erstellt. Es wird über alle Blöcke gesucht (Schritt 2). Ein Block entspricht S^{win} und wird in Objekt oder Nicht-Objekt klassifiziert (Schritt 3). Die Wahl des Klassifikators (*SVM*, *NN*, *BK*, usw.) ist abhängig von der Anwendung. Ein Beispiel der Anwendung von *HoG* zur Personendetektion ist in [DT05] und [Zhu+06] beschrieben. Eine abgewandelte Variante (*Pro-HoG*) wird in [PWT12] vorgestellt.

Texturbasierte Methoden

Eine weitere Methodenklasse bilden die texturbasierten Methoden. Kennzeichen ist die Nutzung von Texturmerkmalen wie *LBP* und *GLCM* (siehe auch texturbasierte Segmentierung in Abschnitt 2.3.2.3). Eine Möglichkeit ist die fensterbasierte Extraktion von Texturmerkmalen und anschließende Klassifikation. Um die rechenaufwändige Bestimmung von Texturmerkmalen zu beschleunigen, gibt es die Möglichkeit, das Bild zu transformieren und auf dem transformierten Bild zu Klassifizieren. Ein Beispiel wird in [PGC10] beschrieben. Durch *Integral Local Binary Pattern* wird die Berechnung von *LBP* beschleunigt.

Weitere Methoden

Eine nicht fensterbasierte Methode, Objekte zu detektieren, ist die Nutzung von **lokalen Bildmerkmalen**. Robuste lokale Bildmerkmale sind bspw. *SURF* (siehe [BTG06]) oder *SIFT* (siehe [Low99]). Ein Objekt ist durch mehrere lokale Bildmerkmale gekennzeichnet. Zur Detektion des Objektes werden die lokalen Bildmerkmale und deren Anordnung im Bild gesucht und ausgewertet.

Eine weitere Methode ist der **Histogramm-Abgleich** (*Histogram Matching*) (siehe [Meh+10]). In der Methode stellt ein Histogramm Farbhäufigkeiten des Bildobjektes O dar. Das Histogramm ist ein Farbmodell von O . Es wird *i.d.R.* fensterbasiert gesucht und die Ähnlichkeit werden mit speziellen Metriken ermittelt. Eine Metrik ist bspw. die *Earth Mover Distanz* aus [RTG00].

Eine weitere ergänzende Möglichkeit ist, das Objekt als Summe von **Teilobjekten** aufzufassen. Die Teilobjekte werden detektiert und die Struktur der Teilobjekte zueinander analysiert, wie bspw. in [HWZ11] beschrieben.

Vor- und Nachteile der Methoden

Die Methoden der Objektdetektion ermöglichen die Detektion eines beliebigen Objektes. Allerdings muss bei allen Methoden erst ein geeignetes Modell vom Objekt erstellt werden. Genereller Nachteil der fensterbasierten Methoden (*Templates*, *Eigenobjekte*, *Haarkaskaden*, *HoG*, *Histogramm-Abgleich*) ist, dass nicht das Bildobjekt detektiert wird. Es wird stattdessen eine **RoI** (S^{win}) detektiert, in welchem sich das Bildobjekt O befindet. Das Bildobjekt muss im Nachgang erst in S^{win} segmentiert werden. Das

Template Matching ist sehr rechenintensiv und bei kleinen Abweichungen vom *Template* instabil. Außerdem sind *Template Matching* und Haarkaskaden nicht invariant gegenüber Rotation und Skalierung. Das Verwenden von **HoGs** und Eigenobjekten führt zu einer Invarianz gegenüber Rotation. Des Weiteren sind **HoGs** verfahrenstechnisch abhängig von Kanten und Gradienten sowie der Bildzerlegung in Blöcke und Zellen. Die Genauigkeit der Detektion aller Methoden ist abhängig davon, wie gut das erstellte Modell ist.

2.3.4 Zusammenfassung der Basismethoden

In diesem Abschnitt 2.3 wurden die Basismethoden der Segmentierung und Objektdetektion strukturiert, klassifiziert und beschrieben. Die Objektdetektion wurde, entgegen der in der Forschung gebräuchlichen Vermischung, von der Segmentierung klar getrennt. Es wurden die Klassen pixelbasierte, konturbasierte, regionenbasierte und optimierungsbasierte Methoden der Segmentierung vorgestellt. In Abbildung 2.8 sind alle beschriebenen Klassen der Segmentierung und Objektdetektion verdeutlicht.

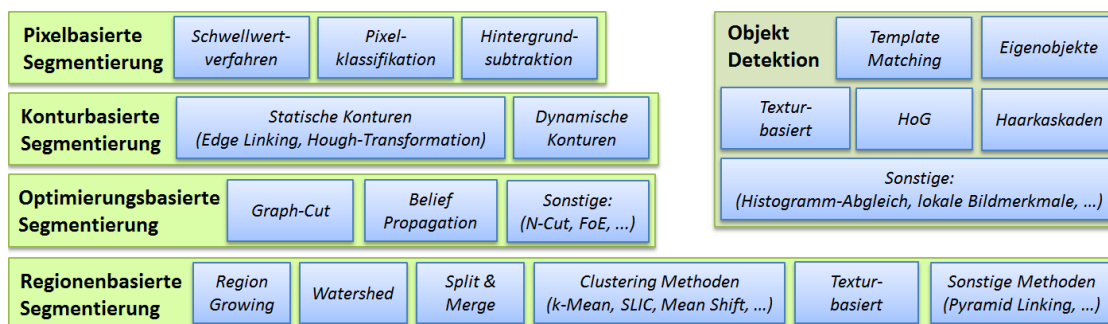


Abbildung 2.8: Klassifikation der vorgestellten Basismethoden der Segmentierung und Objektdetektion.

Die pixelbasierten Methoden ordnen jedem Pixel, unabhängig von den benachbarten Pixeln, ein Label zu. Die konturbasierten Methoden operieren auf Bildkanten oder Gradienten und ermitteln Konturen. Alles, was von der Kontur eingeschlossen wird, entspricht dem gesuchten Segment S . Die regionenbasierten Methoden zerlegen ein Bild in Segmente S gleicher Eigenschaften. Sie stellen die größte Methodenklasse der Segmentierung dar und sind in der Methodik der Segmentierung sehr verschieden. Die optimierungsbasierten Methoden optimieren die Segmentierung eines bereits segmentierten Bildes oder segmentieren ein Bild anhand von Vorwissen. Alle genannten Klassen und deren Basismethoden haben verschiedene Vor- und Nachteile. Grundsätzlich ist die Eignung einer Basismethode von der Segmentierungsaufgabe und dem Bildmaterial abhängig. Ähnliches gilt für die Basismethoden der Objektdetektion. Die Objektdetektion detektiert Objekte anhand eines Modells. Die Eignung der Basismethoden der Objektdetektion ist abhängig von den Eigenschaften des zu detektierenden Bildobjektes.

Zur Erfüllung einer Segmentierungsaufgabe werden *i.d.R.* Basismethoden ausgewählt, adaptiert, erweitert und kombiniert. Welche Basismethoden zur Segmentierung von *organo*-Gruppen geeignet sind, wird im nachfolgenden Kapitel betrachtet.

2.4 Spezielle Methoden am Beispiel von *organo*-Gruppen

Spezielle Methoden der Segmentierung sind abhängig von der Anwendung, und das Ziel der Segmentierung ist *i.d.R.* durch eine Segmentierungsaufgabe definiert. In dieser Arbeit soll eine automatische Multi-Objekt-Segmentierung von Stammschnittflächen entwickelt und die erarbeitete Lösung verallgemeinert werden. Zur Erarbeitung eines geeigneten Konzeptes werden in diesem Abschnitt bereits existierende spezielle Methoden der Segmentierung von *organo*-Gruppen herausgearbeitet.

Eine generelle Lösung zur Segmentierung von *organo*-Gruppen ist bisher nicht Bestandteil der Forschung, weshalb in diesem Abschnitt Einzellösungen untersucht werden. Es werden zuerst spezielle Methoden der Segmentierung von Stammschnittflächen in Abschnitt 2.4.1 beschrieben. Da nur wenige Methoden existieren und werden zur Erfassung eines breiteren Spektrums von speziellen Methoden anschließend auch ähnliche *organo*-Gruppen betrachtet. Die Abschnitte sind nach der Ähnlichkeit der *organo*-Gruppe zu einer Stammschnittfläche sortiert. Zuerst wird in Abschnitt 2.4.2 auf die Segmentierung von Äpfeln und Tomaten eingegangen, welche visuell einer Stammschnittfläche in Form und teilweise auch in der Farbe ähnlich sind. Anschließend werden in Abschnitt 2.4.3 Kiwis, Litchis und andere Früchten betrachtet. Danach wird in Abschnitt 2.4.4 auf Fische und Plankton eingegangen, welche kaum Ähnlichkeiten zu Stammschnittfläche aufweisen. Die genannten Abschnitte 2.4.1, 2.4.2, 2.4.3, 2.4.4 werden weiter nach verwendeter Basismethode und/oder Methodenklasse unterteilt. Die herausgestellten Basismethoden werden im Abschnitt 2.5 zusammengefasst und diskutiert.

Die Objektdetektion wurde bisher klar von der Segmentierung abgegrenzt. Sie ist ein Mittel zur Segmentierung und wird in der Literatur der speziellen Methoden zumeist der Segmentierung untergeordnet. Zur besseren Übersicht und zur Vereinfachung der Verständlichkeit wird nachfolgend auch die **Objektdetektion der Segmentierung untergeordnet**.

2.4.1 Segmentierung von Stammschnittflächen

Viele wissenschaftliche Arbeiten bewegen sich im Anwendungsfeld der Holzinspektion, wie beispielsweise in [AW92] und [REP05]. Beide Beiträge beinhalten Methoden zur Detektion von defekten und unerwünschten Stellen in Holzbrettern. Demgegenüber sind im Bereich der automatischen Segmentierung von Stammschnittflächen aktuell sehr wenige Forschungsarbeiten vorhanden. Die aktuell existierenden Verfahren werden nachfolgend beschrieben und in Schwellwertverfahren und regionen- und konturbasierte Segmentierung eingeteilt. Ein kurzes Fazit über die Methoden wird abschließend gezogen.

Schwellwertverfahren zur Stammschnittflächensegmentierung

Die erste spezielle Methode einer Segmentierung für Baumstämme ist in [MFD92] zu finden. Der Beitrag beschreibt eine binäre Segmentierung für Graustufenbilder anhand eines adaptiven Schwellwertes. Mit adaptivem Schwellwert meinen die Autoren eine Kurve, die durch das Grauwert-Histogramm verläuft und für jeden Grauwert einen speziellen Schwellwert festlegt. Zur Bestimmung der Kurve wird das Bild zuerst mit einem Gauß-Kernel gefiltert und ein Histogramm erzeugt. Nach der ersten Ableitung im Histogramm, werden Histogramm-Merkmale extrahiert wie Null-Zonen, lokale Minima und Maxima. Die Merkmale werden zur Berechnung markanter Grauwerte genutzt. Die Verbindung der markanten Grauwerte im Histogramm ergeben die gesuchte Schwellwertkurve. Der beschriebene Algorithmus wurde allerdings auf nur drei Bildern mit Werkzeug und drei mit Stammschnittflächen getestet. Die Werkzeuge liegen auf einer farblich homogenen Unterlage und die Baumstämme auf einem Truck unter klarem Himmel, womit optimale Bedingungen gegeben sind. Die Abweichung der genutzten Baumstammbilder zur Grundwahrheit liegt bei 0,09 %, 0,65 % und 1,15 %.

Eine weitere Betrachtung der Schnittflächensegmentierung zur Bestimmung des Volumens von Baumstämmen wird in [Sil+05] behandelt. Das beschriebene Verfahren ist eine simple globale Schwellwertsegmentierung für Graustufenbilder von Eukalyptus-Stammschnittflächen. Der globale Schwellwert wird durch ein Histogramm ermittelt. Nach der Segmentierung muss der Nutzer interaktiv ein Polygon über den Holzbereich legen, um den Hintergrund vom Polter zu trennen. Das halbautomatische Verfahren wurde anhand von 15 Bildern evaluiert, wobei eine Volumenabweichung unter 5 % aufgetreten ist.

Regionen- und konturbasierte Stammschnittflächensegmentierung

Eine aus forstwirtschaftlicher Sicht ausführliche Betrachtung der bildbasierten (foto-optischen) Vermessung von Nadelrundholzabschnitten ist in der Dissertation [Fin04] gegeben. Der Fokus der Arbeit liegt nur teilweise in der Bildverarbeitung. Stattdessen steht die Anwendbarkeit und Zweckmäßigkeit von Segmentierungsmethoden zur Bestimmung des Volumens und der Stammdurchmesser im Vordergrund. Dennoch werden manuelle, halbautomatische und vollautomatische Konzepte zur Segmentierung von Stammschnittflächen vorgestellt, wobei lediglich die manuellen final umgesetzt und zur Evaluierung genutzt werden. Der Autor schreibt dazu: „Die Auswertung der Fotos wurde in dieser Arbeit ausschließlich mit den dazu entwickelten manuellen Routinen durchgeführt...“. Die manuelle Methode entspricht einer manuellen Markierung der Schnittflächen mit Unterstützung von *Adobe Photoshop D1-4.0.1*-Algorithmen. Weiterhin werden Passpunkte im Bild gesetzt, die mit eingemessenen realen Stammschnittflächenpunkten korrespondieren. Das Konzept der halbautomatischen Segmentierung basiert auf aktiven Konturen [KWT88], wobei der Nutzer den Stamm zuvor manuell umkreisen muss. Dahingegen entspricht das Konzept der automatischen Segmentierung einer Wasserscheiden Segmentierung [Beu91]. Es wird allem Anschein nach von frischen und sauberen Schnittflächenbildern ohne weiteren Hintergrund wie Boden, Himmel oder Wald ausgegangen. Das Bild der Schnittflächen

2 Stand der Forschung zur Segmentierung

wird zuerst in Graustufen konvertiert und mehrmals mittels Gauß gefiltert. Anschließend wird es invertiert, eine Wasserscheiden-Segmentierung (*Watershed*) durchgeführt und regionenweise mehrfach schwellwertsegmentiert. Anhand vom Nutzer gewählten Formen und einigen Nachverarbeitungsschritten werden die entsprechenden Segmente ermittelt. Weiterhin wird in [Fin04] die Eignung von Stereo-Bildverarbeitung geprüft, aber nicht detaillierter untersucht.

In einem Poster [DGM06] werden zwei automatische Ansätze beschrieben. Der Beitrag baut auf Daten von sScale (siehe Produktblatt [Sys08]) auf und geht von einer bereits bekannten Positionen der Polterfrontfläche im Bild aus. Im ersten Ansatz wird das Bild in einen Multi-Skalen-Raum (*Scale Space*, siehe [Lin96]) überführt und auf allen Skalen geglättet. Die lokalen Maxima der Laplacian-Filterung und ein normalisiertes Stärkemaß werden zur Selektion des entsprechenden Skalenraums benutzt. Dadurch, dass die lokalen Maxima langsam über die Skalen wandern, lässt sich im idealen Fall die Position und Größe des Stammes ableiten. Im zweiten Ansatz wird eine Watershed-Segmentierung [Beu91] des Bildes vorgenommen, was zu einer Übersegmentierung führt. Anhand eines Kriteriums werden Regionen verschmolzen. Beide Algorithmen in [DGM06] werden kurz diskutiert, aber nicht evaluiert.

Fazit der Stammschnittflächensegmentierung

Hauptsächlich werden Watershed ([Fin04], [DGM06]) und spezielle Schwellwertverfahren ([MFD92], [Sil+05]) zur Segmentierung adaptiert. Ein regionenbasierter Multi-Skalen-Ansatz wurde in [DGM06] und ein konturbasierter Ansatz mit aktiven Konturen in [Fin04] adaptiert. Alle Verfahren operieren auf Graustufenbildern und wurden bisher ungenügend unter optimalen Bedingungen bis gar nicht evaluiert. Da wichtige Farbinformationen verloren gehen und nicht von optimalen Bedingungen ausgegangen werden kann, sind die Verfahren für die Zielsetzung dieser Arbeit nur bedingt geeignet.

2.4.2 Segmentierung von Äpfeln und Tomaten

Äpfel und Tomaten sind in Form und Farbe den Stammschnittflächen sehr ähnlich. Das Thema Segmentierung ist vorwiegend im chinesischen Umfeld für die automatische Ernte von Interesse. Im Nachfolgenden werden wissenschaftliche Beiträge zu dem Thema nach Datum geordnet beschrieben und in pixelbasierte, regionenbasierte und konturbasierte Segmentierung und Objektdetektion unterteilt. Ein kurzes Fazit wird abschließend gezogen.

Pixelbasierte Segmentierung von Äpfeln und Tomaten

Eine Methode zur Segmentierung von Tomaten wird in [LZW08] beschrieben. Ein aus Rot und Grün $(R + G)/2$ erzeugtes Grauwertbild wird mit einem Hochpass gefiltert und durch Nutzung einer Schwellwertsegmentierung iterativ segmentiert. Nach zwei morphologischen Opening-Operationen ergeben sich die gesuchten Segmente, die Tomaten, im Bild. Der Algorithmus ist zu 74,3 % genau, was anhand von 70 Bildern evaluiert wurde.

Vier Segmentierungsmethoden für Tomaten, Äpfel, Erdbeeren, Kakis und Orangen in natürlichen Bildern werden in [YMX08] vorgestellt. Alle vier Methoden arbeiten mit einer Farbheuristik und nutzen ein Grauwertbild, erzeugt durch die Differenz vom Rot- und Grünwert ($R - G$). Die Erste entspricht einer Otsu-Segmentierung [Ots79], die Zweite erweitert Otsu's Methode und die Dritte kombiniert eine verbesserte Otsu Methode mit genetischer Arithmetik (siehe genetische Algorithmen in [MTK96]). Die vierte und letzte Methode klassifiziert die Pixel mittels LVQ (Learning Vector Quantisation) in Frucht oder Hintergrund. Im Ergebnis soll die letzte Methode am genauesten sein, aber am langsamsten, was anhand weniger Beispiele belegt ist.

Eine Methode zur Segmentierung von grünen und reifen roten Tomaten zur automatischen Ernte wird in [XXB11] beschrieben. Zur Segmentierung werden ausgewählte normalisierte Farbkanäle genutzt. Für rote Tomaten wird Hue des HIS und für grüne I_2 des $I_1I_2I_3$ Farbraums verwendet. Nach einer abgewandelten Otsu Segmentierung werden die reifen Tomaten zu 98 % und die Grünen zu 89 % richtig erkannt. Der Ansatz wurde anhand von 100 Bildern evaluiert.

Die Segmentierung von Granatäpfeln wird in [Aki+12] anhand eines speziellen Farbraumes vorgenommen. Von den im Beitrag untersuchten Farbräumen erweist sich die modifizierte Differenz $mR - mG$ und $mR - mB$ als geeignet. Nach einer nicht genau beschriebenen Binärisierung und der Anwendung von Kreismerkmalen werden die Granatäpfel mit einem maximalen Fehler von 9,1 % auf 20 Bildern lokalisiert.

Der sehr ausführliche Journal-Beitrag von [LCN12] beschreibt eine Vier-Schritte-Methodik zur Segmentierung von grünen Äpfeln. Im ersten Schritt werden charakteristische Pixel durch Klassifikation segmentiert. Als Klassifikator dient der KNN (*K Nearest Neighborhood*) Klassifikator mit Suche im KD-Baum. Die genutzten Merkmale sind die Glattheit innerhalb eines quadratischen Bildbereiches und die Intensität. Im zweiten Schritt werden benachbarte Pixel mit gewisser Ähnlichkeit zu Segmenten (Blobs) zusammengefügt. Die Segmente werden im dritten Schritt in Bogensektionen und Sonstiges zerlegt. Die Bogensektionen werden im letzten Schritt zu kreisähnlichen Objekten, den gesuchten Äpfeln, gruppiert. Der Ansatz soll zu 85 % grüne Äpfel richtig erkennen, was anhand von 10 Bildern ausgewertet wurde.

Regionenbasierte Segmentierung von Äpfeln und Tomaten

Eine Methode zur Erkennung von Tomaten-Clustern in Gewächshäusern zur automatischen Ernte wird in [Yan+07] beschrieben. Der Bildaufnahme dient die Stereo-Farbkamera BumbleBee2. Die Tomaten werden durch den im Beitrag beschriebenen CLG-Algorithmus (Color Layer Growing) vom Hintergrund einer 3D-rekonstruierten Szene getrennt. Kernpunkte des CLG sind eine Übersegmentierung des Bildes in kleine Regionen und eine anschließende Verbindung derer anhand von Farbähnlichkeiten. Die Evaluierung der ermittelten 3D-Position anhand von 12 Tomaten-Clustern zeigt nur wenig Abweichungen von Soll zu Ist.

Die Anwendung von FCM-LS (Fuzzy C-Mean Clusterung mit Level Set) zur Segmentierung von Früchten in natürlichen Bildern wird in [Xie09] betrachtet. Der Schwerpunkt liegt in der Erweiterung des FCM-Algorithmus. Zur Fruchtsegmentierung wird der FCM-

LS im HSV-Farbraum angewendet. Die Ergebnisse sind anhand von vier Bildern mit Kirschen und Äpfeln illustriert und sehen zufriedenstellend aus.

Die Lokalisierung von Äpfeln zum automatischen Ernten ist Thema in [Mao+09]. Dabei wird für einen Ernte-Roboter, ausgestattet mit einer Stereokamera, die 3D-Position der Äpfel berechnet. Die vorgestellte Segmentierung der Äpfel basiert auf einer Vorder- und Hintergrund-Clustering im 2D-Farbraum von $R - G$ und $R - B$. Ein Pixel wird anhand des geringsten Abstandes im Farbraum dem Vorder- oder Hintergrund zugeordnet. Nach einer anschließenden Rauschbeseitigung im resultierenden Binärbild werden zusätzlich verdeckte Äpfel durch einen zeilenweisen Scann-Algorithmus segmentiert. Im Ergebnis ermittelt das Verfahren die Anzahl der Äpfel mit einem maximalen Fehler von 9,4 %, was anhand von 10 Bildern evaluiert wurde.

Eine Erkennung von Äpfeln mit LS-SVM (Least Squares Support Vector Machine) zur automatischen Ernte wird in [Kon+10] beschrieben. Die Bilder werden von einer Stereokamera am Roboter unter natürlichen Bedingungen aufgenommen. Hauptsächlich wird die Klassifikation der Segmente mit LS-SVM nach der Segmentierung unter Nutzung von Form- und Farbmerkmalen beschrieben. Die eigentliche Segmentierung wird nur kurz dargelegt, genauer gesagt wird Region Growing mit einer Schwellwertsegmentierung durchgeführt. Die Bilder werden mit einer Vektor Median Filterung (siehe [AHN90]) vorverarbeitet und mit den morphologischen Operationen Opening und Closing nachverarbeitet. Die Klassifikation der Segmente ist laut [Kon+10] zu 90 % korrekt.

In [ZZK10] wird eine K-Mean Clustering Methode mit Einbeziehung der Entropie zur Segmentierung von Äpfeln vorgestellt. Auf das Bild wird erst eine gewichtete Median Filterung angewandt und anschließend in den HSI Farbraum transformiert. Die H und S Kanäle werden zum Clustern genutzt. Das entwickelte Entropy Clustering erzielt eine Genauigkeit von 98,9 % gegenüber dem K-Mean mit 91 % auf einer nicht erwähnten Anzahl an Bildern.

Konturbasierte Segmentierung und Objektdetektion von Äpfeln

In [MD92] wird eine Segmentierung für maschinelle Qualitätsprüfungen und Sortierungen von Äpfeln vorgenommen. Der aus dem Bereich des Routings bekannte A* (A-Stern) Algorithmus von N. J. Nilsson in [Nil71] wird in [MD92] adaptiert und zur heuristischen Findung der Äpfel-Konturen genutzt. Der entwickelte Algorithmus arbeitet auf Grauwertbildern und sucht nach den Konturen der Äpfel. Die Position der Äpfel ist in [MD92] bereits bekannt und wird zur Bestimmung des Startpunktes genutzt. Ausgehend vom Startpunkt wird die Äpfelkontur anhand von acht Nachfolgeregeln und acht Eckentemplates mit dem Ziel der Bestimmung der Kosten ermittelt. Die Genauigkeit des Algorithmus ist lediglich anhand von vier Beispielen illustriert.

In [FLB93] wird eine Detektion von Äpfeln für das automatische Pflücken durch einem Roboter vorgestellt. Der Roboter ist mit einem Projektor ausgestattet und erhellt die Szene, wodurch Glanzlichter auf den Äpfeln erzeugt werden. Anhand der GLCM werden Schwellwerte bestimmt, womit das Grauwertbild auf lediglich 20 Werte abgebildet wird. Das Resultat wird anschließend durch eine Maske und spezielle Regeln in ein Binärbild umgewandelt (0 oder 255). Die Maske setzt ein Pixel auf 255, wenn der Absolutbetrag

der Differenz zu jeden der 12 Nachbarn kleiner als ein gewisser Schwellwert ist. Der Algorithmus soll die Äpfel laut [FLB93] in 92 % der Fälle richtig detektieren.

Zur Segmentierung von grünen und roten Äpfeln wird in [ZTK05] ein kombinierter Ansatz aus Textur und Farbe vorgestellt. Zuerst wird das RGB-Bild in ein Grauwertbild mit einem hoch gewichteten Rotanteil konvertiert. Das Bild wird auf 0-16 Werte quantisiert und innerhalb eines 5×5 Bereiches für jedes Pixel der Kontrast der GLCM ermittelt. Auf dem sich ergebenden Kontrastbild werden mit Canny (siehe [Can86]) Kanten detektiert und geschlossene Konturen extrahiert. Aufgrund der genutzten Texturmerkmale, dem Kontrast der GLCM, enthalten die geschlossenen Konturen Segmente homogener Farbe. Liegen die sich ergebenden Segmente homogener Farbe innerhalb eines speziellen Farbberreiches, so werden diese weiter auf Größe geprüft und ggf. auch gesplittet. Dazu wird eine Kreiseinpassung (*circle fitting*) genutzt. Der Soll-Radius-Bereich des Kreises ergibt sich aus der genutzten Stereobildaufnahme und den ermittelten Tiefen. Das Verfahren wird lediglich anhand eines Bilder demonstriert. Auf dem Bild wurden 18 von 20 Äpfeln richtig segmentiert.

Eine Detektion von Äpfeln wird in [Wan+09] mittels SVM-Klassifikation vorgenommen. Das Bild wird mit einer Median-Vektor-Filterung vorverarbeitet und dann mit Region Growing segmentiert. Die notwendigen Saatpunkte werden mit einer 5×5 Maske gesucht, wobei ein Saatpunkt gefunden wurde, wenn mindestens 20 Pixel kleiner als ein gewisser Schwellwert sind. Nach anschließender Extraktion von Farb- und Formmerkmalen wird das Segment in Apfel oder Nicht-Apfel klassifiziert. Das Verfahren zeigt eine Genauigkeit von 93,3 % mit Nutzung eines RBF-Kerns (Radial-Basis-Funktion) bei der SVM-Klassifikation. Ein nahezu identischer Ansatz, nur ohne SVM-Klassifikation, wird auch in dem Vision System des Apfel-Ernte-Roboters beschrieben und in [Lv+11] angewendet.

Eine weitere Klassifikationsmethode wird in [RGY10] zur Detektion von Äpfeln adaptiert. Es wird der Hue-Kanal des HSV-Farbraumes verwendet und das Bild in 8×8 Blöcke zerlegt. Für jeden Block wird die Energie der GLCM ermittelt. Die Energie wird als Merkmal genutzt und ein Modell mittels EM (Expectation Maximization) für Äpfel und Hintergrund erstellt. Das Bild wird mithilfe des Modells und der Merkmale der Blöcke segmentiert. Auf 113 Bildern mit insgesamt 776 Äpfeln wurden mit dem Verfahren 85,33 % richtig erkannt.

Fazit zur Segmentierung von Äpfeln und Tomaten

Die Ansätze der Segmentierung von Äpfeln und Tomaten sind vielfältig. Häufige adaptierte und kombinierte Methoden sind Schwellwertsegmentierung ([LZW08], [YMX08], [Kon+10], [XXB11], [Aki+12], [Aki+12]) und Clustering-Methoden ([Yan+07], [Xie09], [Mao+09], [Xie09], [ZZK10]). Weiterhin werden Texturmerkmale, speziell die Merkmale der GLCM ([FLB93], [ZTK05], [RGY10]), zur Detektion von homogenen Bereichen verwendet. Zur Segmentierung werden bei fast der Hälfte der Ansätze spezielle Farbbereiche und/oder Farbräume verwendet ([ZTK05], [YMX08], [LZW08], [Mao+09], [ZZK10], [XXB11], [Aki+12]). Die Methoden sind folglich **sehr speziell** und auf die Segmentierung von Äpfeln und/oder Tomaten angepasst. Für andere *organo*-Gruppen mit anderen Farb- oder Textur-Charakteristiken sind die Methoden nicht geeignet.

2.4.3 Segmentierung von Kiwis, Litschis und anderen Früchten

In diesem Abschnitt werden spezielle Methoden, geordnet nach Datum der Veröffentlichung, zur Segmentierung von Kiwis, Litschis und sonstige Früchte beschrieben. Die Methoden werden nachfolgend unterteilt in pixelbasierte und regionen- und konturbasierte Methoden. Ein kurzes Fazit über die adaptierten Methoden wird abschließend gezogen.

Pixelbasierte Segmentierung von Kiwis, Litschis und Zitronen

Eine Klassifikationsmethode zur Unterscheidung von Kiwi-Pixel und Nicht-Kiwi-Pixel wird in [Wij+09] vorgestellt. Anhand eines Neutralen Netzwerks NN mit SOM (Self Organized Maps) wird ein Klassifikator erstellt. Ermittelte Farbmerkmale aus dem CIELab-Farbraum werden zur Klassifikation genutzt. Die sich durch die Klassifikation ergebenden segmentierten Bilder sind nicht evaluiert.

Die Segmentierung von Litschis und deren Stängel wird in [DLZ11] betrachtet. Der Algorithmus arbeitet in drei Schritten. Zuerst wird der Cr-Kanal vom YCbCr-Farbraum schwellwertsegmentiert und morphologisch nachbearbeitet, was Litschi inklusive Stängel vom natürlichen Hintergrund trennt. Im zweiten Schritt wird die Frucht durch K-Mean Clustering im RGB-Raum separiert und durch Bild-Subtraktion wird Frucht und Stängel im letzten Schritt getrennt. Von 30 getesteten Bildern sind die Litschis in 24 erfolgreich segmentiert worden.

Die Segmentierung von Zitronen ist zum Teil Thema in [Alo+12]. Genau betrachtet wird eine Pixelklassifikation angewendet, wobei Zitronen- oder Nicht-Zitronen Pixel unterschieden werden. Verwendet wird ein Naive-Bayes-Klassifikator, welcher zuvor mit Zitronen- und Hintergrundpixel trainiert wurde. Das Vorgehen ist bzgl. Segmentierung nicht evaluiert.

Regionen- und konturbasierte Segmentierung von sonstigen Früchten

In [HWJ05] wird eine spezielle Watershedsegmentierung vorgestellt. Der Kern des neuen Ansatzes ist die Nutzung von Multi-Skalen-Gradienten zur Berechnung von morphologischen Gradienten. Der Algorithmus soll zur Multi-Objekt-Segmentierung von Trauben, Kirschen, Bananen und Äpfeln geeignet sein. Die Ergebnisse sind aber nur anhand eines Bildes von Weintrauben auf homogenen Hintergrund visuell dargelegt.

Zur automatischen Zählung von grünen und goldenen Kiwis wird in [Wij+08] eine entsprechende Segmentierung vorgestellt. Die R-, G- und B-Kanäle der RGB-Bilder werden in dem Ansatz speziell transformiert. Anschließend wird das Bild in den Lab-Farbraum konvertiert und die a- und b-Komponenten im Histogramm analysiert. Peaks im Histogramm werden in Form von Clustern angelernt. Der Abstand zu den Clustern wird in der finalen Segmentierung genutzt und eine abschließende Opening-Operation durchgeführt. Die Anzahl der lokalen Maxima nach einer Distanztransformation wird ermittelt und stellt die gesuchte Anzahl der Kiwis dar. Das Vorgehen soll zu 90 % genau bei goldenen Kiwis (78 Bilder) und zu 60 % genau bei grünen Kiwis (42 Bilder) sein.

Die Segmentierung von Kaffeekirschen unter kontrollierten Bedingungen auf homogenen Hintergrund wird in [BP08] adressiert. Ziel ist eine echtzeitfähige kostengünstige Lösung,

2.4 Spezielle Methoden am Beispiel von *organo*-Gruppen

umgesetzt auf DSP oder FPGA, für kolumbianische Arbeiter. Die Segmentierung arbeitet grob in drei Schritten. Zuerst werden Cluster mit FCM im RGB-Farbraum erstellt. Von den Clustern werden sieben dem Vordergrund und drei dem Hintergrund zugeordnet. Die Zuordnung der Cluster wurde zuvor anhand der Grundwahrheit experimentell ermittelt. Die Vordergrund-Cluster werden im finalen Schritt als Ausgangspunkt für Fuzzy-Aktive-Konturen genutzt.

Die automatische Segmentierung von Litschis zur automatischen Ernte ist Thema in dem Beitrag [Lia+11]. Es wird von einem Stereosetup ausgegangen und auf einem Kamerabild segmentiert. Segmentiert wird in zwei Schritten im Hue-Kanal des HSV-Farbraums, welcher allerdings rotiert betrachtet wird. Im ersten Schritt wird anhand gewisser Regeln (*Fuzzy Rules*) ein optimaler Schwellwert ermittelt und somit eine initiale evolutionäre Kurve ermittelt. Die Kurve passt sich im zweiten Schritt mit *Sparse Field Level Set* dem Objekt genauer an. Der Algorithmus soll zu 90,48 % richtig segmentieren.

Eine Methode zur Trennung von quasi-runden Objekten wird in [LT11] beschrieben, wobei das Verfahren nur anhand dreier Bilder illustriert ist. Die Früchte liegen auf einem homogenem Hintergrund und es wird anscheinend auch schon von segmentierten Früchten ausgegangen. Durch ein internes Radius-Template werden die Objekte getrennt.

Fazit zur Segmentierung von Kiwis, Litschis und sonstigen Früchte

Die speziellen Methoden zur Segmentierung von Kiwis, Litschis und anderen Früchten sind teilweise denen von Äpfeln und Tomaten ähnlich. Primär werden pixel- und regionenbasierte Methoden adaptiert, speziell sind dies Schwellwertverfahren [DLZ11], Pixelklassifikationen ([Wij+09], [Alo+12]), Clustering Methoden ([BP08], [Wij+08], [DLZ11]) und Watershed [HWJ05]. Auch dynamische konturbasierte Verfahren werden adaptiert und kombiniert ([BP08], [Lia+11]). Die Methoden sind sehr speziell zugeschnitten auf die Objekte. Außerdem sind die meisten Methoden nicht evaluiert, womit keine Aussage über die Genauigkeit getroffen werden kann. Es ergibt sich folglich, dass die Methoden **nicht ohne Anpassung** auf andere *organo*-Gruppen **übertragbar** sind.

2.4.4 Segmentierung von Fischen, Plankton und Muscheln

In diesem Abschnitt werden spezielle Methoden zur Segmentierung von Fischen, Plankton und Muscheln beschrieben. Die betrachteten *organo*-Gruppen sind Unterwasserlebewesen, die wenig Ähnlichkeit zu Stammschnittflächen aufweisen. Die Methoden werden nachfolgend in Hintergrundsubtraktion und sonstige Methoden unterteilt. Die sonstigen speziellen Methoden unterteilen sich in pixel-, regionen- und konturbasierte Verfahren. Ein kurzes Fazit über die speziellen Methoden der Unterwasserlebewesen wird abschließend gezogen.

Hintergrundsubtraktion zur Segmentierung von Fischen und Plankton

Die Segmentierung von **Fischen** in Einzelbildern unter Wasser wird selten adressiert, da sich Fische kaum vom Hintergrund abgrenzen. Auch variieren Fische stark in Form, Farbe und Textur, womit eine robuste Segmentierung in Einzelbildern ganz spezielle

Rahmenbedingungen benötigt. Das Problem der Einzelbildsegmentierung unter Wasser wird zumeist umgangen und die Bild-zu-Bild Kohärenz in Videos genutzt, wie bspw. mit der Hintergrundsubtraktion in [Zhu+12] und [LWG10].

In [Mor+05] werden Fische zum Zwecke der Verfolgung (*tracking*) in Videoaufnahmen detektiert. Zur Detektion wird je ein Vorder- und ein Hintergrundfarbmodell erstellt. Als Modelle kommen **GMMs** zum Einsatz, welche mit Bildern vom leeren Hintergrund und einzelnen Fischen angelernet werden. Der Ansatz basiert auf dem Verfahren in [Sul+99].

Ein System zur Vermessung von Fischen über Wasser wird in [WSS06] beschrieben. Die Fische liegen auf einem Förderband mit homogener Farbe. In einer Initialisierung von 20 Sekunden wird die mittlere Farbe des Förderbandes bestimmt. Die Fische werden anschließend anhand des Unterschiedes zur mittleren Förderbandfarbe segmentiert.

Eine weitere Klasse der **organo**-Gruppe ist **Plankton**. Plankton sind Kleinstlebewesen im Wasser und werden **i.d.R.** mit Mikroskop-Kameras Unterwasser oder im Labor aufgenommen. Plankton hebt sich im Bild **i.d.R.** sehr gut vom Hintergrund ab, womit einfache Segmentierungsverfahren **i.d.R.** ausreichend sind. Eine Laborvariante ist ZooScan, ein wasserfester Flachbett-Scanner mit der Software ZooImage. In ZooImage (siehe [Gro+04], [Gor+10]) wird eine Hintergrundsubtraktion vorgenommen. Das heißt, das Bild vom leeren Scanner (Hintergrundbild) wird vom Bild mit Plankton subtrahiert. Ist die Differenz eines Pixels vom Hintergrundbild zum Bild mit Plankton größer als ein gewisser Schwellwert, so wird das Pixel als Plankton segmentiert.

Sonstige Segmentierungen von Fischen, Plankton und Muscheln

In [MS08] wird eine regionenbasierte Segmentierung von **Fischen** beschrieben, wobei speziell eine Kombination aus *Mean Shift* und *Median Cut* vorgestellt wird. Die Farbe wird erst mit einer Histogramm-Equalisierung korrigiert. Anschließend wird eine *Mean Shift* Segmentierung im LUV-Farbraum durchgeführt und die Farbe mit *Median Cut* quantisiert. Nach einer Kantendetektion im Ergebnisbild sind einzelne Fische von einer Kontur umschlossen. Das Verfahren wird in [MS08] *K-Mean* und *Mean Shift* gegenübergestellt und erweist sich als genauer.

In [Bla+05] werden zwei Segmentierungsmethoden von **Plankton** für das FlowCam-System vorgestellt. Die erste Methode segmentiert Plankton mit aktiven Konturen (siehe [KWT88]). Die zweite Methode ist eine Schwellwert-Segmentierung anhand des Histogramms der Grauwerte. Der Schwellwert wird ermittelt, indem zwei **GMMs** in das Histogramm mit **EM** eingepasst werden (siehe [DLR77]).

Ein System zur Segmentierung und Klassifikation von Zooplankton auf sehr großen Bilddaten ($\approx 80.000 \times 80.0000$ Pixel) ist ZooCount (siehe [Rus+13], [Gut+14]). In ZooCount wird das Bild mittels einem Quadtree in Teilbilder zerlegt und anschließend in jedem Teilbild mit der Methode von Otsu segmentiert. Nach der Segmentierung werden die Konturen extrahiert und aneinandergrenzende Konturen von Teilbild zu Teilbild verschmolzen. Das Ergebnis sind Konturen von Zooplankton und Nicht-Zooplankton (Algen, Eier, Schmutz, ...). Die Konturen werden weiterhin zur Bestimmung von Zooplankton gefiltert, wobei spezielle Formmerkmale Verwendung finden. Das Verfahren wurde auf fünf sehr großen Bildern, bestehend aus insgesamt 5027 Teilbildern (2048×1536 Pixel),

evaluiert. Die Evaluierung zeigt eine *precision* von 0,96 und einen *recall* von 0,44.

Eine Segmentierung von **Muscheln** auf statischem und homogenem Hintergrund wurde in der Diplomarbeit [Fri13] untersucht. Die schwarzen Muscheln liegen separiert auf einem weißen Hintergrund und wurden mit der Methode von Otsu segmentiert. Die Segmentierungsaufgabe ist infolge der Rahmenbedingungen sehr einfach. Es wurde eine mittlere Messgenauigkeit in 15 Bildern von 98 % ermittelt.

Fazit zur Segmentierung von Fischen, Plankton und Muscheln

In den speziellen Methoden zur Segmentierung von Fischen, Plankton und Muscheln wurden spezielle Rahmenbedingungen an die Bildaufnahme gesetzt, so dass einfache Methoden bereits gute Ergebnisse erzielen. Viele Methoden verwenden eine Hintergrundsubtraktion in Videos ([Zhu+12], [LWG10]) oder auf Bildern ([Gro+04], [Gor+10], [Mor+05], [WSS06]). Weiterhin werden Clustering-Methoden [MS08], dynamische Konturen [Bla+05] und pixelbasierte Methoden ([Fri13], [Gut+14]) adaptiert. Zur Erfüllung der Zielstellung in dieser Arbeit sind die Methoden einzeln kaum geeignet, da von wenig Varianzen im Hintergrund (\bar{O}) ausgegangen wird. Außerdem sind die Methoden, außer der pixelbasierten Methoden in [WSS06] und [Fri13], [Gut+14], nicht evaluiert.

2.5 Basismethoden versus Segmentierungsaufgabe

In Abschnitt 2.4 wurden spezielle Methoden zur Segmentierung von **organo**-Gruppen beschrieben. Die Methoden sind sehr verschieden. Unterschiedlichste Basismethoden wurden adaptiert, erweitert und kombiniert. Auch wurden neue Algorithmen entwickelt, um die jeweilige Segmentierungsaufgabe zu erfüllen. Es ist auch zu erkennen, dass **spezielle Farbbereiche** und/oder Farbräume genutzt werden. Die Farbbereiche werden **i.d.R.** empirisch ermittelt und sind kennzeichnend für spezielle Methoden. Bei den speziellen Methoden kommen primär folgende Basismethoden (siehe Zusammenfassung in Abschnitt 2.3.4) zur Anwendung:

- **F**: Empirisch ermittelte spezielle Farbräume
- **P**: Pixelbasierte Segmentierung
 - **P_S**: Schwellwert, **P_K**: Pixelklassifikation, **P_H**: Hintergrundsubtraktion
- **K**: Konturbasierte Segmentierung
 - **K_S**: statische Konturen, **K_D**: dynamische Konturen
- **R**: Regionenbasiert Segmentierung
 - **R_R**: Region Growing, **R_W**: Watershed
 - **R_S**: Split & Merge, **R_C**: Clustering-Methoden
- **O**: Objektdetektion
 - **O_T**: Texturbasiert, **O_M**: Template Matching

2 Stand der Forschung zur Segmentierung

In der nachfolgenden Tabelle 2.1 werden für spezielle Segmentierungsaufgaben die verwendeten Basismethoden (BM) aufgezeigt und bewertet. In Spalte **BM** sind die Basismethoden (Abkürzungen siehe vorangestellter Liste) zugeordnet. Basismethoden, die sehr spezielle Algorithmen enthalten und keiner Unterklasse zugeordnet werden können, sind nur der Methodenklasse (P, K, R oder O) zugeordnet. In Spalte **Kernalgorithmus** ist die Kernaussage des Ansatzes benannt und in der Spalte **E** die Qualität der Evaluierung mit Werten von 0 bis 5 beurteilt⁸. Außerdem ist die Tabelle gegliedert in verschiedenen Kategorien von **organo**-Gruppen.

Tabelle 2.1: Spezielle Methoden zur Segmentierung von **organo**-Gruppen

Beitrag	BM	Kernalgorithmus	E
Segmentierung von Stammschnittflächen			
[MFD92]	P _S	Adaptive Schwellwertkurve	2
[Sil+05]	P _S	Globale Schwellwertsegmentierung mit Nutzerinteraktion	3
[Fin04]	K _D	Aktive Konturen mit Nutzerinteraktion	1
[Fin04]	R _W ,P _S	Watershed kombiniert mit Schwellwertsegmentierung	1
[DGM06]	R	Multi-Skalen-Raum Segmentierung	1
[DGM06]	R _W	Watershed mit Regionenverschmelzung	1
Segmentierung von Äpfeln und Tomaten			
[MD92]	K _S	Adaptiver A*	0
[FLB93]	O _T ,O _M	R-GLCM und 12'er Nachbarschaftsmaske	1
[ZTK05]	O _T ,K _S ,F	Kombiniert Kontrast der lokalen GLCM mit Farbe	1
[Yan+07]	R _C	Color Layer Growing, Segmentverschmelzen	0
[LZW08]	F,P _S	Schwellwert auf Grauwert von $(R - G)/2$	4
[YMX08]	F,P _S ,P _K	Erweiterte Otsu Methode, Learning Vector Quantisation	0
[Mao+09]	F,R _C ,P _K	Vorder- und Hintergrundcluster von D_{rg} und D_{rb}	2
[Wan+09]	R _R ,O	Saatpunktdetektion 5x5 Maske, Region Growing, SVM	1
[Xie09]	R _C ,K _D	FCM-LS (Fuzzy C-Mean - Clusterung mit Level Set)	0
[RGY10]	O _T	Energie-Klassifikation der GLCM mit GMM Cluster	5
[ZZK10]	F,R _C	K-Mean Entropy Clusterung	1
[Kon+10]	R _R ,P _S	Region Growing, Schwellwert, Least Square SVM	1
[XXB11]	F,P _S	Abgewandelter Otsu, H von HIS , I_2 von $I_1I_2I_3$	4
[Aki+12]	F,P _S	Schwellwert auf modifizierter Rot-, Grün-, Blau-Differenz	3
[LCN12]	P _K ,R _S	Pixelklassifikation, Split & Merge, Kreissegmentfindung	2
Segmentierung von Kiwis, Litschis und sonstigen Früchten			
[HWJ05]	R _W	Spezieller Watershed	0
[BP08]	F,R _C ,K _D	FCM, Fuzzy basierte Konturanpassung	0
[Wij+08]	F,R _C	Clustering auf Lab nach Intensitätstransformation	4
[Wij+09]	P _K	Pixel Klassifikation mit NN (ähnlich wie [Lv+11])	0

⁸E: 0: Evaluierung nicht bzgl. Segmentierung oder nur visuell; 1: Genauigkeit nicht unterlegt; 2: weniger gleich 10 Bilder; 3: 11 – 50 Bilder; 4: 51 – 100 Bilder; 5: mehr als 100 Bilder

Beitrag	BM	Kernalgorithmus	E
[Lia+11]	P _S ,K _D	Initiale Kurve durch Schwellwert, SFLS	1
[LT11]	O _M ,K _D	Internes Radius-Template zur Segmenttrennung	0
[DLZ11]	P _S ,R _C	Schwellwert auf C_r , K-Mean Clustering von RGB	3
[Alo+12]	P _K	Naive Bayes Pixelklassifikation	0
Segmentierung von <i>Fischen, Plankton und Muscheln</i>			
[Gro+04]	P _H	Hintergrundsubtraktion (Bild)	0
[Mor+05]	P _H ,P _K	Pixelklassifikation durch GMM Farbmodelle	0
[Bla+05]	K _D	Aktive Konturen	0
[Bla+05]	P _S	Schwellwert mit GMM -Einpassung im Histogramm	0
[WSS06]	P _H	Hintergrundsubtraktion (Bild)	5
[MS08]	R _C	Kombination <i>Mean Shift</i> und <i>Median Cut</i>	0
[LWG10]	P _H	Hintergrundsubtraktion (Video) u. Schattenentfernung	0
[Gor+10]	P _H	Hintergrundsubtraktion (Bild)	0
[Zhu+12]	P _H	Hintergrundsubtraktion (Video)	0
[Fri13]	P _S	Otsu-Methode auf statischem Hintergrund	3
[Gut+14]	P _S , R _S	Otsu-Methode und konturbasiertes Split & Merge	5

Es zeigt sich in Tabelle 2.1, dass die verwendeten Basismethoden (BM) sehr verschieden sind und oft in Kombination verwendet werden. Auch wurden die Basismethoden speziell angepasst und sind zum Teil der originalen Methode kaum noch ähnlich. Die wenigen bereits existierenden Methoden zur Segmentierung von Stammschnittflächen sind kaum evaluiert. Die geringfügig evaluierten Methoden sind Schwellwertverfahren ([MFD92], [Sil+05]), angewandt auf Bildern mit wenig Varianzen im Hintergrund \bar{O} . Die Recherche über spezielle Methoden ähnlicher **organo**-Gruppen (Früchte, Fische, ...) ergab ein breiteres Spektrum an adaptierten Basismethoden. Die Methoden für Äpfel und Tomaten in Naturbildern sind vielfältig. Die meisten Methoden benutzen spezielle empirisch ermittelte Farbbereiche und/oder gehen von einer homogenen Textur der Objekte O aus. Die speziellen Methoden für andere Früchte (Kiwis, Litschis, Zitronen, ...) hingegen sind sehr unterschiedlich und auf die spezifische **organo**-Gruppe zugeschnitten. Auch werden Annahmen in der speziellen Farbcharakteristik gesetzt. Die Methoden für Fische, Plankton und Muscheln setzen zumeist Restriktionen in der Bildaufnahme, wie bspw. einen homogenen Hintergrund. Folglich basieren die meisten Methoden auf einer Hintergrundsubtraktion.

Ein Indikator für eine geeignete spezielle Methode aus Tabelle 2.1 ist eine aussagekräftige Evaluierung mit guten Ergebnissen. Auch müssen in der Methode **wenig Annahmen** über die **organo**-Gruppe und **kaum Restriktionen** in der Bildaufnahme gegeben sein. Die Evaluierungen der recherchierten speziellen Methoden sind sehr verschieden und die Genauigkeiten kaum vergleichbar. Bei Betrachtung der Beiträge mit einer Evaluierung mit mehr als 20 Bildern bleiben nur noch wenige Beiträge übrig, welche folgende Genauigkeiten aufweisen:

- **Beitrag: Evaluierungsziel; Genauigkeit (Anzahl der Bilder)**
- [WSS06]: Fischklassifikation nach Segmentierung auf Förderband; 99,8 % (100)

2 Stand der Forschung zur Segmentierung

- [LZW08]: Erkennung von Tomaten am Strauch; 74,3 % (70)
- [Wij+08]: Zählung von goldenen und grünen Kiwis; 90 % (78), 60 % (42)
- [RGY10]: Erkennung von Äpfeln am Baum; 85,33 % (113)
- [XXB11]: Erkennung von reifen und grünen Tomaten; 98 %, 89 % (100)
- [Gut+14]: (Vor-)Erkennung von Zooplankton in Zählkammern; f -score: 0,7 (5027)

Im Beitrag [WSS06] werden Fische auf einem Förderband klassifiziert. Der Ansatz ist bei der Unterscheidung von sieben Fischarten, welche signifikante Unterschiede in der Form aufweisen, zu 99,8 % genau. Die Segmentierung ist nicht mit Maßzahlen evaluiert und ist keine besondere Herausforderung, da das Förderband eine homogene Farbe besitzt. Der Beitrag [LZW08] präsentiert einen Ansatz, der zu 74,3 % Äpfel und Tomaten richtig erkennt. Unter Erkennung ist zu verstehen, dass die ungefähre Position eines Apfels im Bild erkannt wird. Die Segmentierung ist ebenfalls nicht mit Maßzahlen hinterlegt und der Ansatz geht von einer speziellen Apfel oder Tomatenfarbe aus. In [Wij+08] werden Kiwis in Bildern automatisch gezählt. Die spezielle Methode zählt zu 90 % goldene und zu 60 % grüne Kiwis richtig. Auch in diesem Ansatz werden spezielle Farbeigenschaften der Früchte verwendet, und die Segmentierung ist nicht mit Maßzahlen hinterlegt. In [RGY10] werden Äpfel an einem Baum zu 85,33 % richtig erkannt. “Erkannt” bedeutet in dem Beitrag, dass ein ermitteltes Rechteck ungefähr den Apfel umschließt. Der Ansatz geht ebenfalls von speziellen Farbeigenschaften (Hue-Bereich im HSV-Farbraum) und Textureigenschaften (Energie der GLCM) von Äpfeln aus. In [XXB11] werden reife Tomaten zu 98 % und grüne Tomaten zu 60 % genau erkannt. Die Segmentierung ist nicht mit Maßzahlen hinterlegt, sondern stattdessen die Erkennung. Mit “Erkennung” ist dieses Mal gemeint, dass ein Segment ungefähr mit einer Tomate übereinstimmt. Auch in diesem Beitrag werden spezielle Farbcharakteristiken angenommen (H von HIS, I_2 von $I_1I_2I_3$). Der eigene Ansatz in [Gut+14] erkennt vorsortiertes Zooplankton in partikelreichem Wasser in sehr großen Bildern. Die Bilder sind Graustufenbilder und die Evaluierung wurde bzgl. der umschließenden Box vorgenommen.

Zusammenfassend muss festgestellt werden, dass zur Erfüllung der Zielstellung **keine geeigneten speziellen Methoden existieren**. Die speziellen Heuristiken in der Farbe der meisten Methoden können nicht auf Stammschnittflächen übertragen werden. Weiterhin sind die Methoden nicht mit den tatsächlichen Segmenten (Grundwahrheit) evaluiert. Auch fehlen die entsprechenden Maßzahlen, wie bspw. *precision*, *recall* oder *f-score* (siehe Anhang A.4). Zur weiteren Problemlösung müssen folglich die Basismethoden genauer untersucht, erweitert, kombiniert und an die Segmentierung von Stammschnittflächen und andere organo-Gruppen adaptiert werden. Die Analyse der Basismethoden wird in Abschnitt 3.1 vorgenommen und zur Ableitung eines Konzeptes weiter herangezogen.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

In diesem Kapitel werden Verfahren zur automatischen Multi-Objekt-Segmentierung von **Stammschnittflächen** (f_s^{st}) und binären Segmentierung von **Schnittholz** (f_s^h) konzipiert, diskutiert und evaluiert. Für die entsprechenden Segmentierungsfunktionen (f_s^{st} , f_s^h) ergeben sich verschiedene Labelmengen (L^{st} , L^h), die wie folgt formalisiert werden sollen:

$$f_s^{st} : I \rightarrow L^{st}; L^{st} = \{Stamm_1, \dots, Stamm_n, \neg Stamm\} := \{a | \forall a \in L^{st} : a \hat{=} S\} \quad (3.1)$$

$$f_s^h : I \rightarrow L^h; L^h = \{Schnittholz, \neg Schnittholz\} \quad (3.2)$$

Ausgehend vom Stand der Forschung, wie in Kapitel 2 beschrieben, werden die speziellen Probleme und Herausforderungen der Segmentierung von Stammschnittflächen im Unterkapitel 3.1 diskutiert. Außerdem werden im Unterkapitel 3.1 mögliche Konzepte grob hergeleitet und die Erfolg versprechenden Konzepte vertieft. Zur Herleitung werden im Vorfeld einige Experimente mit einfachen Basismethoden der Segmentierung herangezogen. Die Konzepte werden einzeln in den Unterkapiteln 3.2, 3.3 und 3.4 verfeinert, beschrieben und evaluiert. Im Unterkapitel 3.2 wird eine binäre Segmentierung von Schnittholz anhand der Holzfarbe und in 3.3 ein allgemeines Konzept der Segmentierung von Stammschnittflächen durch Objektdetektion erarbeitet. Ein kombiniertes Konzept bestehend aus einer Objektdetektion, Graph-Cut-Segmentierung und Objektseparierung wird im Unterkapitel 3.4 betrachtet. Die Evaluierungen der einzelnen Konzepte werden im Abschnitt 3.5 gegenübergestellt und die Konzepte diskutiert.

3.1 Problemdiskussion und Herleitung von Konzepten

Generelle Probleme und Herausforderungen bei der Segmentierung von Stammschnittflächen wurden bereits kurz in Abschnitt 1.2 genannt. In diesem Abschnitt werden die Probleme detaillierter untersucht und mögliche Konzepte abgeleitet. Zuerst werden in Abschnitt 3.1.1 die Probleme detailliert aufgeschlüsselt und abhängig von der Bildqualität und dem Stammschnittflächenzustand unterschiedliche Schwierigkeitsgrade aufgezeigt. Zur Untermauerung der Konzeption werden in Abschnitt 3.1.2 einige Experimente mit einfachen Basismethoden zur Segmentierung von Stammschnittflächen oder Schnittholz durchgeführt. In Abschnitt 3.1.3 werden abschließend unter Betrachtung des aktuellen Standes der Forschung und der Experimente grobe Konzepte hergeleitet.

3.1.1 Problemdiskussion

Die Zielstellung dieses Kapitels ist die Konzipierung, Untersuchung und Evaluierung von automatischen Segmentierungen von Stammschnittflächen aus Rundholzabschnitten. **Rundholzabschnitte** sind aufeinander gestapelte Baumstämme. Im Wald oder an der Straße befindliche Rundholzabschnitte werden auch als **Polter** bezeichnet. In Abbildung 3.1 sind einige typische Polter zu sehen.



Abbildung 3.1: Frontflächenabschnitte von verschiedenen Poltern aus der Praxis, aufgenommen mit einem mobilen Endgerät.

Bei der Konzipierung einer Segmentierung ist es notwendig, die zu segmentierenden Objekte genauer zu betrachten. Durch die Betrachtung können Eigenschaften abgeleitet werden, die als Vorwissen für die Konzeption gelten. Wird ein Segmentierungsalgorithmus ohne Vorwissen konzipiert und ohne Vorwissen angesteuert (ohne Modell), so sind die Ergebnisse dem Zufall überlassen. Je weniger Vorwissen vorhanden ist, umso weniger Annahmen können getroffen werden und umso schwieriger ist die Problemstellung.

Die zu segmentierenden Objekte sind Stammschnittflächen von Baumstämmen. Vorwissen über Stammschnittflächen ergibt sich aus dem biologischen Aufbau eines Baumstammes und der Abbildung des Stammschnittflächenobjektes $O^{r,st}$ zum Bildobjekt O^{st} . Aus biologischer Sicht besteht ein Baumstamm aus **Rinde**, **Jahresringen**, **Splint-** und **Kernholz** (siehe [Mat+10]). In Abbildung 3.2 ist der typische Aufbau eines Baumstammes illustriert.

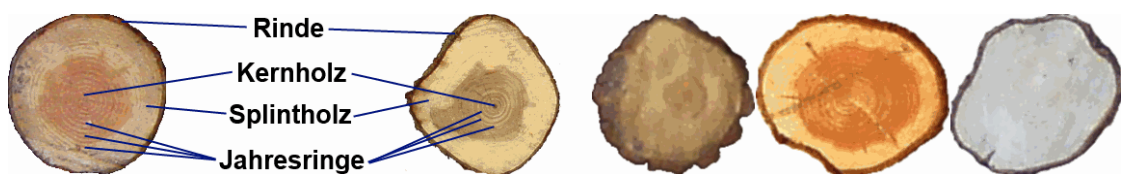


Abbildung 3.2: Aufbau des Stammquerschnittes (links) und Stammquerschnitte mit verschiedenen Verhältnissen von Splint- zu Kernholz (rechts).

Die Rinde ist die äußere Schicht eines Baumstammes und schützt vor äußeren Einflüssen. Nach [Mat+10] wird: „das nicht mehr wasserleitende Holz als Kernholz bezeichnet, das hellere, wasserleitende Holz, welches das Kernholz umgibt, als Splintholz.“ Durch

3.1 Problemdiskussion und Herleitung von Konzepten

klimatisch bedingte Unterschiede im Sommer und Winter wächst das Holz unterschiedlich. Es entstehen Jahresringe, die in der Breite und Farbgebung verschieden sind (Details siehe [Sch83]).

Bedingt durch die Bildaufnahme in der Praxis mit einfachen Kameras und mobilen Endgeräten (siehe Projekt AFoRS in [Foe+12]) ist die Bild- (I) und Stammschnittflächenqualität (O^{st}) sehr unterschiedlich. Durch den Fakt, dass viele Baumstämme aufeinander gestapelt in einem Bild zu sehen sind, ist die Auflösung pro Stammschnittfläche (O^{st}) gering, wodurch Jahresringe selten zu erkennen sind. In Abschnitt 1.2 wurde bereits herausgestellt, dass Varianzen in Form, Farbe und Textur in den Stammschnittflächen und im Hintergrund vorliegen. Außerdem sind die Form, Farbe und Textur der Stammschnittflächen unbekannt, und Überschneidungen mit der Farbe des Hintergrundes sind möglich. Auf dieser Basis können kaum Annahmen zur Konzeption einer Segmentierung von Rundholzabschnitten gemacht werden. Polter aus der Praxis sind von unterschiedlichster Qualität. Die benannten Probleme leiten sich aus der Menge aller vorkommenden Qualitätsstufen (Bild- und Stammschnittflächenqualität) ab. Allerdings ergeben sich bei genauer Betrachtung der Qualitätsstufen mehr Annahmen, die über die Stammschnittflächen gemacht werden können. Im Folgenden werden deshalb verschiedene Qualitätsstufen¹ aufgestellt und betrachtet. Die Qualität soll in zwei Typen (Q_{st} , Q_I) mit je drei Qualitätsstufen (A, B, C) aufgeschlüsselt und wie folgt notiert werden:

- Q_{st} - Qualität der Stammschnittflächen
- Q_I - Qualität der Bildaufnahme und/oder des Bildes
- $Q_{st}, Q_I \in \{A, B, C\}$; $A \hat{=}$ gut, $B \hat{=}$ mittelmäßig, $C \hat{=}$ schlecht

Die vorkommende Stammschnittflächenqualität Q_{st} (siehe Abbildung 3.3) soll sich durch die Form, Holzzustand und Häufigkeit von farblichen Markierungen kennzeichnen. In Abbildung 3.3 sind verschiedene Beispiele pro Kategorie von Q_{st} dargestellt.

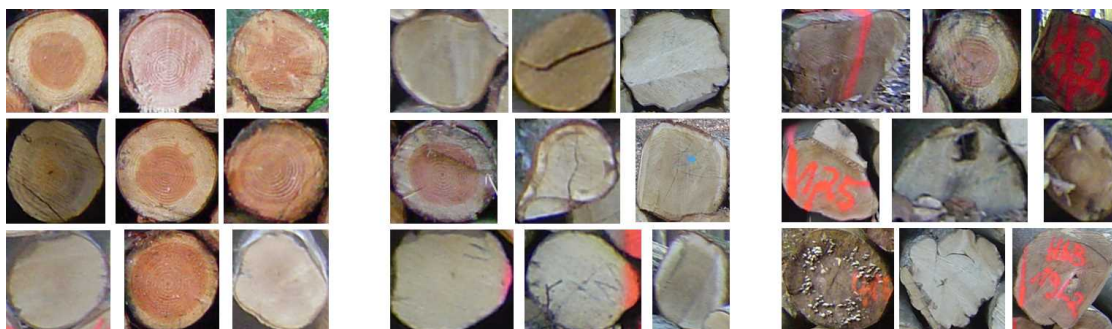


Abbildung 3.3: Von links nach rechts die verschiedenen Qualitätsstufen der Stammschnittflächen A, B und C von Q_{st} .

¹Die aufgestellten Qualitätsstufen sind subjektiv und sollen den Schwierigkeitsgrad der Segmentierungsaufgabe kennzeichnen.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Die Stufen der Qualität Q_{st} sollen folgende Bedeutung widerspiegeln:

- A: Annähernd runde Form, kaum Holzdefekte und keine Markierungen
- B: Geschlossene Form, wenig Holzdefekte oder wenige Markierungen
- C: Kaum erkennbare Form, grobe Holzdefekte oder viele Markierungen

Die Qualität der Bildaufnahme und/oder des Bildes soll hingegen durch die Qualität der Beleuchtung, der Bildschärfe und Verdeckungen der Stammschnittflächen gekennzeichnet sein. In Abbildung 3.3 sind verschiedene Beispiele pro Kategorie von Q_I dargestellt.



Abbildung 3.4: Von links nach rechts die verschiedenen Qualitätsstufen der Bildaufnahme A, B, C von Q_I .

Die Stufen der Qualität Q_I sollen folgende Bedeutung widerspiegeln:

- A: Gleichmäßige Beleuchtung, scharfes Bild und kaum Verdeckungen
- B: Ungleichmäßige Beleuchtung, mittelmäßige Bildschärfe oder wenig Verdeckungen
- C: Schlagschatten, unter- oder überbelichtet, Verwaschen oder viele Verdeckungen

Ausgehend von der besten Qualitätsstufe A bei Q_{st} und Q_I , sowie dem biologisch typischen Querschnitt eines Baumes, ist es möglich Vorwissen abzuleiten. Typische Stämme haben annähernd eine runde Form, welche durch die Rinde begrenzt wird. Es gibt bei Stammschnittflächen innen liegendes und dunkles Kernholz, das von hellerem Splintholz umgeben ist, oder nur Splintholz mit einer homogenen Farbtextur (siehe Abbildung 3.2). Weiterhin sind die Baumstämme $O^{r,st}$ aufeinander gestapelt, wodurch sich im Bild ein Cluster (**organo**-Gruppen - OG^{st}) von Bildobjekten O^{st} ergibt.

Ausgehend vom Vorwissen sollen im Fall der Segmentierung von Stammschnittflächen OG^{st} folgende **Annahmen** getroffen werden:

1. Alle O^{st} im Bild haben ähnliche Eigenschaften und bilden ein Cluster OG^{st}
2. $O^{st} \in OG^{st}$ hat annähernd eine rundliche oder elliptische Form
3. $O^{st} \in OG^{st}$ hat ein bis zwei charakteristische Farbbereiche (Splint/Kernholz)

Die zweite und dritte Annahme sind nur für $Q_{st} = Q_I = A$ haltbar, da defekte Stammschnittflächen, farbliche Markierungen, Verdeckungen oder inhomogene Beleuchtung zu Form und Farbveränderungen der Stammschnittflächen im Bild führt. Für $Q_{st} = B$ oder $Q_I = B$ ist die erste Annahme noch eingeschränkt sinnvoll. Die Varianz der Ähnlichkeiten sollte größer werden, aber dennoch sollten Ähnlichkeiten vorhanden sein. Hingegen sind für $Q_{st} = C$ oder $Q_I = C$ alle drei Annahmen unsicher.

Trotz der Annahmen verbleiben noch einige Herausforderungen. So ist die Farbverteilung des Hintergrundes (Wald, Wiese, Straße, Himmel, ...) vollkommen unbekannt, weshalb Überschneidungen der Farbe zum Vordergrund möglich sind. Rein auf Farbe basierende Segmentierungsalgorithmen sind folglich nicht geeignet und würden die Teile des Hintergrundes mit ähnlicher Farbe auch als Stammschnittfläche segmentieren. Außerdem ist die Farbgebung der Stammschnittflächen innerhalb eines Bildes ähnlich, aber von Bild zu Bild sehr unterschiedlich (siehe Abbildung 3.1).

3.1.2 Voruntersuchung von Basismethoden der Segmentierung

Bestehende Verfahren zur Segmentierung von Stammschnittflächen adaptieren zumeist Schwellwertverfahren ([MFD92], [Sil+05]) oder *Watershed* ([Fin04], [DGM06]). Beide Verfahren sind Klassiker der Segmentierung, für die es einige frei verfügbare Implementierungen gibt. Um die Eignung der Verfahren besser beurteilen zu können, werden in diesem Abschnitt einige Experimente mit Schwellwertverfahren, *Watershed* und auch mit anderen einfach zu testenden Basismethoden vorgenommen. Zur Beurteilung der Ergebnisse werden zwei Bilder (siehe Abbildung 3.5) mit unterschiedlichen Qualitätsstufen Q_{st} und Q_I herangezogen. Weitere Beispielbilder von *organo*-Gruppen und deren Ergebnisse sind im Anhang B.2 zu finden.



Abbildung 3.5: Beispielbilder mit $Q_{st} = Q_I = A$ (links) und $Q_{st} = Q_I = B$ (rechts).

Die Segmentierung eines Bildes mit einem Schwellwert ist trivial und soll als erstes untersucht werden. Zur Segmentierung werden die RGB-Bilder I konvertiert in Grauwert-

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Bilder I' mit $px \in [0, 255]$ und $px \in \mathbb{N}$. Die Bilder I' werden anschließend segmentiert und der benötigte Schwellwert T zum einen manuell und zum anderen automatisch durch die Otsu-Methode [Ots79] bestimmt. Die Ergebnisse der Schwellwertverfahren sind in Abbildung 3.6 zu sehen.

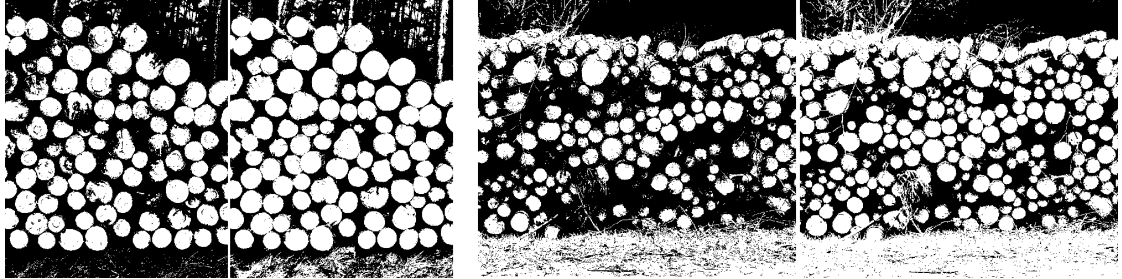


Abbildung 3.6: Ergebnisse der Schwellwertverfahren mit $T = 150$ (links) und Otsu-Methode (rechts). Weiße Pixel entsprechen dem segmentierten Schnittholz.

Die pixelbasierte Segmentierungsfunktion $f_s^{px} = f_s^h$ zerlegt das Bild in Schnittholz- und nicht Schnittholzpixel, wodurch die Stammschnittflächen O^{st} nicht einzeln segmentiert werden. In Abbildung 3.6 segmentiert kein Schwellwert die Beispielbilder optimal. Es ist Rauschen im Ergebnis und viele Teile des Hintergrunds sind falsch segmentiert. Ausgehend von diesen Ergebnissen sind Schwellwertverfahren ohne Erweiterung und/oder spezieller Vorverarbeitungsschritte, wie Farbtransformationen, nicht zur Segmentierung von Stammschnittflächen geeignet.

Als nächstes wird die Eignung von konturbasierter Segmentierungen untersucht. Konturbasierte Segmentierungen sind angebracht, wenn die Gradienten im Bild entsprechend ausgeprägt sind. Genauer betrachtet, müssen die Gradienten am Rande der Stammschnittflächen O^{st} hoch und möglichst eindeutig sein. In Abbildung 3.7 sind die Gradienten und extrahierten Kanten zu sehen.

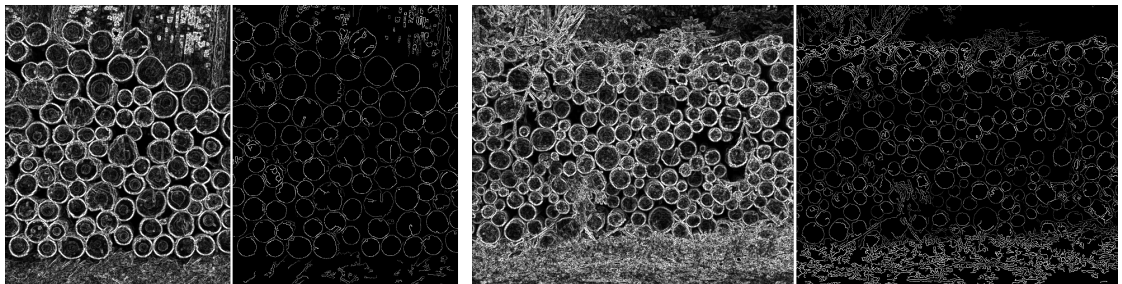


Abbildung 3.7: Gradientenbild mit dem *Sobel Operator* (links) und das Kantenbild mit dem Canny-Kanten-Detektor (rechts).

Die Ergebnisse in Abbildung 3.7 zeigen, dass sich die Stammschnittflächen O^{st} durch starke Gradienten gut abgrenzen. Auch die erzeugten Bilder mit dem Canny-Kanten-Detektor (siehe [Can86]) weisen klare Stammschnittflächen-Konturen mit einigen Lücken auf. Im Kantenbild sind dennoch viele Mehrdeutigkeiten (siehe Abbildung 3.8) vorhanden, welche durch den Hintergrund (Gras, Bäume, ...) und aneinanderliegende sowie defekte Stämme entstehen. Es kann demzufolge nicht sichergestellt werden, dass eine Kontur O^{st} umschließt. Nichtsdestotrotz sollte die Kombination von Kontur mit Farbe und/oder Textur aller Stammschnittflächen OG^{st} eine gute Segmentierung ermöglichen.



Abbildung 3.8: Mehrdeutigkeiten des Kantenbildes vom Canny-Kanten-Detektor.

Letztendlich werden die Eignung einiger Basismethoden der regionenbasierten Segmentierung (Watershed, SLIC, Mean Shift) untersucht. Watershed ist eine angewandte Basismethode zur Segmentierung von Stammschnittflächen in [Fin04] und [DGM06]. Die errechneten Ergebnisse der *Watershed*-Segmentierung mit der CV Bibliothek VIGRA² (siehe [Kot00]) und OpenCV³ (siehe [Lag11]) sind in Abbildung 3.9 zu sehen.

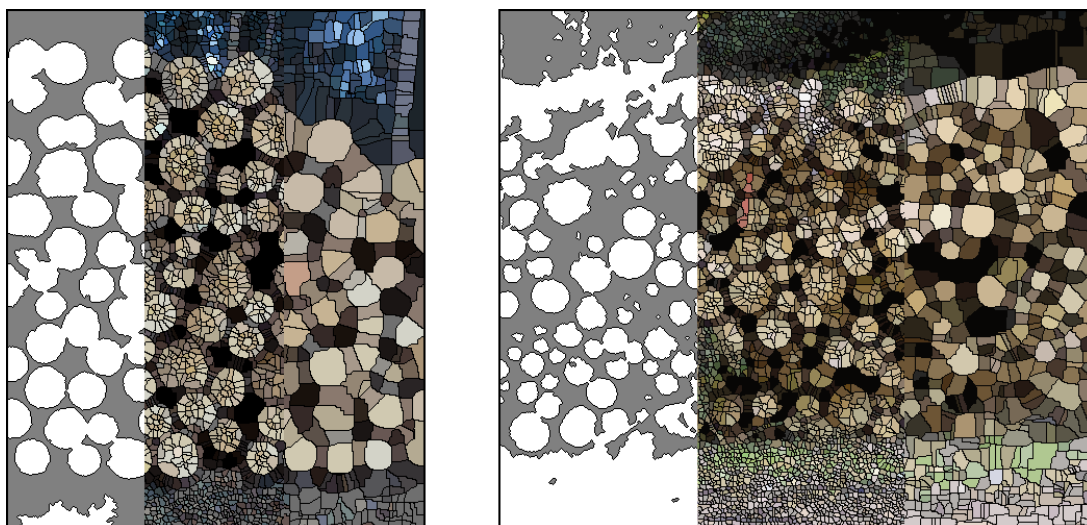


Abbildung 3.9: Ergebnisse verschiedener *Watershed*-Umsetzungen in einem Bild, wobei OpenCV links, VIGRA mit $scale = 2$ in der Mitte und mit $scale = 8$ rechts zu sehen sind. Benachbarte Pixel gleicher Farbe bilden ein Segment.

²VIGRA: Vision with Generic Algorithms, Version 1.9.0, hci.iwr.uni-heidelberg.de/vigra

³OpenCV: Open Source Computer Vision Library, Version 2.4.7, docs.opencv.org

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Die Ergebnisse in Abbildung 3.9 zeigen, dass je nach Umsetzung von *Watershed* und Parameterwahl die Ergebnisse verschieden sind. Trotz der unterschiedlichen Zerlegungen der Bilder in Segmente S ist keine Segmentierung optimal. Die Stammschnittflächen O^{st} sind zumeist über- oder untersegmentiert. Es müssen Parameter (z.B. *scale* zur Skalierung der Gradienten) pro Bild angepasst werden. Weiterhin werden in vielen Umsetzungen Saatpunkte benötigt, von denen der typische simultane Flutvorgang aus startet. Die Saatpunkte müssen manuell definiert oder anhand lokaler Minima errechnet werden. Folglich ist *Watershed* nur bei optimaler Parameterwahl mit geeigneten Saatpunkten und Adaption an die Segmentierungsaufgabe geeignet.

Zur weiteren Untersuchung der regionenbasierten Segmentierung sollen *Mean Shift* aus OpenCV und *SLIC*⁴ (siehe [Ach+12]) betrachtet werden. Beide Basismethoden sind *Clustering*-Methoden und clustern im Farb- und Bildraum. Ergebnisse der Basismethoden sind in Abbildung 3.10 zu sehen.

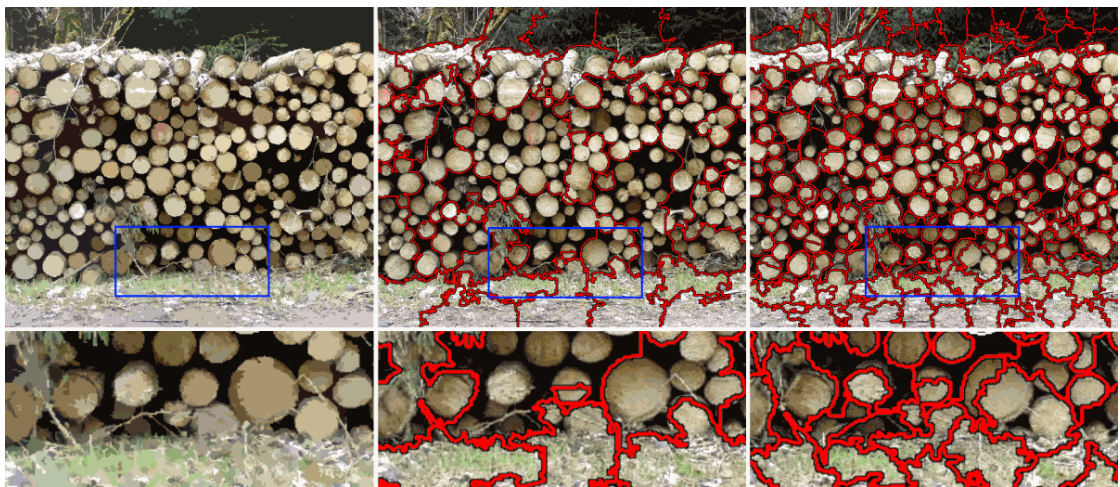


Abbildung 3.10: Ergebnisse von *Mean Shift* mit einem Suchabstand von 50 im Farb- und Bildraum (links), *SLIC* mit $N = 50, C = 20$ (Mitte) und mit $N = 200, C = 20$ (rechts). Bei *Mean Shift* bilden benachbarte Pixel gleicher Farbe ein Segment. Bei *SLIC* zeigen rote Linien die Segmentkonturen.

Die Ergebnisse in Abbildung 3.10 zeigen keine optimale Segmentierung sowie Über- oder Untersegmentierungen. Es müssen mehrere Parameter gut gewählt werden, wie Suchabstände im Farb- und Bildraum, die *Compactness* C und die gewünschte Anzahl der Segmente $N = |S|$. Bei allen Methoden der regionenbasierten Segmentierung ergibt sich zusätzlich das Problem, dass die korrespondierenden Stammschnittflächen zu den Segmenten ($O^{st} \hat{=} S$) des zerlegten Bildes I erst ermittelt werden müssen.

Die Voruntersuchungen in diesem Abschnitt haben gezeigt, dass keine dieser Basismethoden in unveränderter Form zur Segmentierung der Stammschnittflächen OG^{st} geeignet

⁴SLIC: verwendet wurde der frei verfügbare C++ Quellcode zum Beitrag [Ach+12]

ist. Schwellwertverfahren haben sich als ungeeignet herausgestellt, da Farbvarianzen und Überschneidungen im Vorder- (OG^{st}) und im Hintergrund ($\overline{O^{st}}$) bestehen. Weiterhin hat sich gezeigt, dass sich OG^{st} sich durch ausgeprägte Gradienten abhebt, aber durch andere Gradienten (Defekte, Schmutz, Bäume, Gras, ...) viele Mehrdeutigkeiten entstehen. Eine Konturverfolgung entlang der Gradienten wird folglich mehrdeutig. Regionenbasierte Methoden führen zur Über- und Untersegmentierung und sind sehr sensibel bei Parameteränderungen. Im Ergebnis der Untersuchungen kann folglich nur eine Kombination, Erweiterung und/oder Adaption der untersuchten Basismethoden eine Erfüllung der Zielstellung leisten.

3.1.3 Diskussion und Herleitung verschiedener Konzepte

Ausgehend vom aufgearbeiteten aktuellen Stand der Forschung in Kapitel 2, den diskutierten Problemen in Abschnitt 3.1.1 und den Voruntersuchungen im vorangegangenen Abschnitt 3.1.2 sollen nun Konzepte zur Lösung hergeleitet und diskutiert werden. Im Abschnitt 2.5 hat sich gezeigt, dass zur Segmentierung von **organo**-Gruppen die verschiedensten Basismethoden adaptiert und kombiniert werden. Es wird stets Vorwissen über die zu segmentierenden Objekte OG und OG^r verwendet, wie **bspw.** durch Verwendung von speziellen Farbräumen und Farbbereichen. Die Basismethoden der Segmentierung sind sehr vielfältig und es ergeben sich entsprechend viele Kombinationen. Die Basismethoden und deren Unterklassen werden im Folgenden hinsichtlich der Eignung zur Multi-Objekt-Segmentierung (f_s^{st}) der Stammschnittflächen OG^{st} und binären Segmentierung von Schnittholz (f_s^{st}) diskutiert.

Spezielle Farbräume (F)

Die speziellen Farbräume werden relativ häufig zur Segmentierung von **organo**-Gruppen verwendet. Der Grundgedanke besteht darin, dass spezielle organische Objekte O^o eine charakteristische Farbgebung aufweisen. Die spezielle Farbgebung wird experimentell ermittelt oder durch Beispieldaten (*Ground-Truth*) angelernt. Anhand der ermittelten Farbgebung wird in unterschiedlichster Weise (Farbcluster, Regeln, usw.) segmentiert. Da Schnittholz eine charakteristische Farbgebung hat, ist diese Methode eine Möglichkeit zur binären Segmentierung (f_s^h). Allerdings variiert die Farbe von Schnittholz je nach Zustand des Holzes (nass, morsch, etc.) und Baumart. Auch ist eine entsprechende Methode zur Segmentierung notwendig, um anhand der Holzfarbe zu segmentieren, wie **bspw.** ein Pixelklassifikator (P_K). Ein weiterer Nachteil besteht darin, dass die Holzfarbe auch im Hintergrund ($\overline{O^{st}}$) vorkommen kann.

Pixelbasierte Segmentierung (P)

Die pixelbasierte Segmentierung wird trotz der Einfachheit und vieler Nachteile unerwartet häufig zur Segmentierung von **organo**-Gruppen adaptiert. Zumeist wird die Segmentierung zusätzlich mit einer weiteren Basismethode kombiniert. **Schwellwertverfahren (P_S)** wurden bereits im vorherigen Abschnitt untersucht und erzielten schlechte Ergebnisse auf Graustufenbildern. Es ergeben sich aber andere Ausgangsbedingungen,

wenn das Verfahren mit der Holzfarbe (F) kombiniert wird. So lässt sich durch **bspw.** empirisch ermittelte Holzfarbcluster und ein entsprechendes Abstandsmaß ein Wahrscheinlichkeitsbild von Holz ermitteln I_{st}^p . Die Segmentierung von I_{st}^p führt voraussichtlich zu bedeutend besseren Ergebnissen. Ähnlich verhält es sich mit der **Pixelklassifikation** ($\mathbf{P_K}$), für die ein Holzpixelklassifikator erstellt werden muss. Durch geeignete Trainingsdaten und den entsprechenden Klassifikator ist eine binäre Segmentierung der Holzfarbe (f_s^h) denkbar. Die **Hintergrundsubtraktion** ($\mathbf{P_H}$) findet bei der Segmentierung von Fischen Anwendung. Im Fall der Stammschnittflächen ist die Basismethode ungeeignet, da weder Videos noch statische Hintergründe gegeben sind.

Konturbasierte Segmentierung (K)

Für eine konturbasierte Segmentierung werden Gradienten benötigt. Je eindeutiger sich Bildobjekte O durch starke Gradienten abgrenzen, um so besser die Segmentierung. In den Voruntersuchungen hat sich gezeigt, dass an den Objektgrenzen von O^{st} hohe Gradienten verlaufen, aber auch weitere hohe Gradienten in O^{st} und im Hintergrund sind. Die Methoden der **statischen Konturen** ($\mathbf{K_S}$) *Edge Linking* und Hough-Transformation ermitteln Konturen auf Basis der Gradienten. Bei *Edge Linking* werden es die benannten Mehrdeutigkeiten schwierig bis unmöglich machen, die richtigen Konturen von OG^{st} zu finden. Weiterhin bilden die Objektgrenzen keine geometrisch parametrierbare Figuren wie einen Kreis oder eine Ellipse, wodurch auch die Hough-Transformation ungeeignet ist. Die **dynamischen Konturen** ($\mathbf{K_D}$) hingegen passen sich automatisch einem Objekt an und benötigen eine initiale Kontur. Einzeln ist die Basismethode ungeeignet, weil keine initiale Kontur verfügbar und Nutzerinteraktion nicht Gegenstand dieser Arbeit ist. In Kombination mit einer Objektdetektion sind dynamische Konturen eine Möglichkeit O^{st} zu segmentieren, da der Umriss des detektierten Bereiches S^{win} eine initiale Kontur bereitstellt.

Regionenbasierte Segmentierung (R)

Die regionenbasierte Segmentierung zerlegt ein Bild in Segmente S mit unterschiedlicher Methodik. Angenommen, die Methodik hat das Bild gut segmentiert, so ist noch nicht bekannt, welches S eine Stammschnittfläche O^{st} ist. Das impliziert, dass im Nachgang die entsprechenden Segmente ($S \hat{=} O^{st}$) ermittelt werden müssen. Eine Ausnahme bildet **Region Growing** ($\mathbf{R_R}$), da ausgehend von Saatpunkten (Vorwissen) die Regionen, bis ein gewisses Stopp-Kriterium erfüllt ist, wachsen. Das bedeutet, dass zur Segmentierung in einem Vorschritt je ein Saatpunkt innerhalb von O^{st} ermittelt werden muss. Auch wenn die Saatpunkte bekannt sind, bleibt noch das Problem des Stopp-Kriteriums. Je nach Baumart, Zustand und Alter ist O^{st} verschieden texturiert. Die Rinde, das Kern- und Splintholz sind in der Farbgebung verschieden und weisen unterschiedliche Größenverhältnisse auf. Das Stopp-Kriterium, was für eine bestimmte Stammschnittfläche gut ist, wird nicht auf alle anderen übertragbar sein. Ein weiteres Problem ist das sogenannte *Leaking*, wodurch bildlich gesprochen, Regionen an undichten Stellen (Gradientenarmut) auslaufen können. Aufgrund der vielen Nachteile wird **Region Growing**

zur Segmentierung von OG^{st} ausgeschlossen. Auch **Watershed (R_W)** und **Clustering Methoden (R_C)** wird an dieser Stelle ausgeschlossen. Beide Methodenklassen wurden bereits in den Voruntersuchungen betrachtet und erwiesen sich als zu problembehaftet. Weiterhin wird auch **Split & Merge (R_S)** ausgeschlossen, da entsprechende Kriterien für den *Split* und für den *Merge* Prozess notwendig sind. Ähnlich wie bei **Region Growing** werden geeignete Kriterien für O^{st} (bspw. Homogenität bei *Split* und Ähnlichkeit bei *Merge*) nicht auf alle anderen OG^{st} übertragbar sein. Die **texturbasierte Segmentierung (R_T)** basiert auf lokalen Texturmerkmalen und deren Beziehungen zueinander. Da O^{st} verschieden texturiert ist und auch viele Mehrdeutigkeiten im Hintergrund ($\overline{O^{st}}$) vorkommen werden, wird diese Methode ebenso ausgeschlossen. Damit sind alle regionbasierten Segmentierungsmethoden ausgeschlossen. Da die Probleme der Methoden ähnlich sind, werden auch die **sonstige Methoden (R_X)** nicht weiter betrachtet.

Optimierungsbasierte Segmentierung (OP)

Die optimierungsbasierte Segmentierung optimiert ein bereits segmentiertes Bild oder segmentiert anhand von Wahrscheinlichkeitsbildern I^P . Es gehen sowohl Gradienten als auch Farben in die Optimierung ein, wodurch entsprechend genaue Ergebnisse erzielt werden. Die Ermittlung des globalen Optimums führt **i.d.R.** zu einer erhöhten Rechenzeit. Die Rechenzeit ist in dieser Arbeit zweitrangig. Höchste Priorität hat die Segmentierung.

Die bekannteste Methode ist **Graph-Cut (OP_G)**. Graph-Cut ermittelt das Optimum durch den minimalen Schnitt (*min-cut*) eines über dem Bild aufgespannten Graphen. Der *min-cut* korrespondiert invers zum maximalen Fluss (*max-flow*). Man stelle sich den Graphen als Rohrleitungssystem vor, wobei der Rohrdurchmesser invers zu den Gewichten korrespondiert. Wird Wasser durch das Rohrleitungssystem gepumpt, so staut sich das Wasser in einigen Rohren (**z.B.** bei kleinen Durchmessern/hohen Gradienten). Entlang der Staustellen verläuft der gesuchte *min-cut*. Da OG^{st} sich durch hohe Gradienten abgrenzt und Graph-Cut mit Mehrdeutigkeiten, Lücken in den Konturen und Varianzen der Farbe sowie Textur umgehen kann, ist Graph-Cut das bisher beste Verfahren. Graph-Cut benötigt eine Vorsegmentierung oder Wahrscheinlichkeitsbilder I^P , die aber auf Basis einer angelernten Holzfarbe (F) berechenbar ist.

Beliefpropagation (OP_B) ist ähnlich wie Graph-Cut eine Variante der approximierten Schlussfolgerung (*probabilistic inferenz*) auf **MRF**-Modellen. Dennoch ist die Schlussfolgerung algorithmisch kaum mit Graph-Cut vergleichbar, obwohl zumeist ein Graph (Faktor-Graph) dem Algorithmus zugrunde gelegt wird. Die Pixel tauschen untereinander Nachrichten über ihren Zustand aus, wie die Disparität oder deren Label. Die Wahrscheinlichkeit eines Zustandes steigt, wenn die benachbarten Pixel ähnliche Zustände haben. Beliefpropagation wird oft in der Stereobildanalyse zur optimierten Berechnung von Disparitäten eingesetzt. Die Disparitätsberechnung ist eine ähnliche Problemstellung, wie ein Label aus dem Bereich der Segmentierung. Die Anwendung in der Segmentierung ist folglich theoretisch möglich. Es besteht aber der Nachteil, dass die Methode vorwiegend für andere Probleme erforscht ist und an Segmentierungsprobleme erst adaptiert werden muss. Dennoch ist Beliefpropagation theoretisch, ähnlich wie Graph-Cut, zur Segmentierung von Stammschnittflächen geeignet, wobei zuvor auch eine

Vorsegmentierung oder ein Wahrscheinlichkeitsbild I^p ermittelt werden muss.

Die sonstigen (OP_X) optimierungsbasierten Methoden *Relaxation Labeling*, *Normalized-Cut (N-Cut)* und *Fields of Expert (FoE)* sind sehr verschieden. *Relaxation Labeling* optimiert iterativ die Labels der Pixel anhand von Wahrscheinlichkeiten der Nachbarlabels. Das Verfahren ist durchaus ergänzend zu einer anderen Basismethode geeignet, wenn zuvor die Wahrscheinlichkeit eines Labels bestimmt werden kann. **N-Cut** ist ähnlich wie Graph-Cut eine graphenbasierte Segmentierungsmethode. Im Unterschied zu Graph-Cut wird bzgl. der Segmentgröße normalisiert, wobei zu große Segmente in der Optimierung bestraft (*penalize*) werden. Nach [Hoi10] wird **i.d.R. N-Cut** zur Übersegmentierung eines Bildes verwendet, und die Anzahl der Segmente muss vor der Segmentierung bekannt sein. Da die Anzahl der Stammschnittflächen nicht bekannt ist, ist **N-Cut** allein ohne ein Verfahren zur Berechnung der Anzahl ungeeignet. Die **FoE** ermöglichen das Anlernen von **MRF**-Modellen anhand von Trainingsbildern. Erforschte Einsatzgebiete der Methode sind *denoising* und *inpainting*. Für eine Segmentierungsaufgabe muss folglich **FoE** erst untersucht und angepasst werden. Alle beschriebenen sonstigen Methoden haben ihre Vor- und Nachteile. Die Eignung der Methoden kann ohne tiefgreifende Anpassungen und Experimente nicht genau bestimmt werden.

Objektdetektion (O)

Die Methoden der Objektdetektion detektieren Objekte anhand eines Modells vom Objekt O und/oder O^r . Je nach Modell, Suchstrategie und Methode der Klassifikation eines Bildbereiches S^{win} in O oder $\neg O$ sind die Methoden unterschiedlich gut zur Detektion von O^{st} geeignet. Das **Template Matching (O_M)** mit statischen Templates ist gegenüber Rotation und Skalierung nicht invariant. Auch sind statische Templates bei geringen Abweichungen vom Template zum Objekt sehr sensibel. Die **texturbasierten (O_T)** Methoden detektieren Gebiete mit gewissen Textureigenschaften. Die Texturen der Stammschnittflächen O^{st} sind zwar charakteristisch, aber dennoch verschieden und von der Baumart und der Genauigkeit der Bildaufnahme abhängig. Das Histogramm der orientierten Gradienten **HoG (O_{Ho})** hingegen detektiert Objekte anhand der Gradientenverteilung, womit die Farbe unbetrachtet bleibt. **HoG** ist eine Möglichkeit zur Detektion von O^{st} , wobei aufgrund der vielen Mehrdeutigkeiten der Gradienten mit einer erhöhten Falsch-Positiv-Detektion gerechnet werden muss. **Eigenobjekte (O_E)** betrachten implizit alle Merkmalsklassen (Form, Farbe, Textur) durch den entsprechenden Eigenraum. Der Eigenraum ist n -dimensional und eine Transformation aller m Pixel auf n Dimensionen mit hohem Informationsgehalt ($n < m$). Folglich sind Eigenobjekte gut zur Detektion von O^{st} , wie auch **Haarkaskaden (O_{HC})** geeignet. Haarkaskaden bilden das Problem der Objektdetektion auf viele linear verkettete Haartemplates ab. Durch die Kaskadierung werden in der Summe Form, grobe Texturen und Farbunterschiede erfasst. Die **sonstige Methoden (O_X)** der Objektdetektion sind verschieden und einige bieten ebenso eine adäquate Möglichkeit, O^{st} zu detektieren. Das Grundproblem aller umrissenen Verfahren ist aber, dass zumeist ein rechteckiger Bereich S^{win} als Objekt klassifiziert wird. Eine detektierte Stammschnittfläche O^{st} ist dann noch nicht segmentiert. Anhand von S^{win} muss O^{st} erst durch ein Segmentierungsverfahren segmentiert werden,

wie **bspw.** durch dynamische Konturen (K_D) oder Region Growing (R_R), ausgehend vom Zentrum in S^{win} .

Zusammenfassung und Fazit

In diesem Abschnitt wurden bisher alle Basismethodenklassen der Segmentierung und Objektdetektion aus dem Grundlagenkapitel 2.3 hinsichtlich der Eignung zur Segmentierung von Stammschnittflächen untersucht. Diskutiert wurden die Klassen:

- **F**: Empirisch ermittelte spezielle Farbräume
- **P**: Pixelbasierte Segmentierung
 - **P_S**: Schwellwert, **P_K**: Pixelklassifikation, **P_H**: Hintergrundsubtraktion
- **K**: Konturbasierte Segmentierung
 - **K_S**: Statische Konturen, **K_D**: Dynamische Konturen
- **R**: Regionenbasiert Segmentierung
 - **R_R**: Region Growing, **R_W**: Watershed, **R_S**: Split & Merge
 - **R_C**: Clustering Methoden, **R_T**: Textur, **R_X**: Sonstige Methoden
- **OP**: Optimierungsbasierte Segmentierung
 - **OP_G**: Graph-Cut, **OP_B**: Beliefpropagation, **OP_X**: Sonstige
- **O**: Objektdetektion
 - **O_T**: Textur, **O_M**: Template Matching, **O_{Ho}**: **HoG**
 - **O_E**: Eigenobjekte, **O_{HC}**: Haarkaskaden, **O_X**: sonstige

Aus der Diskussion der Basismethoden haben sich verschiedene Vor- und Nachteile ergeben, wobei die Basismethoden zur Lösung der Segmentierungsaufgabe verschieden gut geeignet sind. In der nachfolgenden Tabelle 3.1 wird die Eignung der Basismethoden (**BM**) veranschaulicht und anhand einiger Kriterien bewertet.

Tabelle 3.1: Bewertung der Basismethoden zur Segmentierung von **organo**-Gruppen

BM	S	U	W	Begründung (Kernargument/e)
F	6	n	4	Stammschnittflächen haben charakteristische Holzfarben
P _S	6	j	3	Schwellwert nach Farbtransformation ($BM = F$) denkbar
P _K	6	n	3	Pixelklassifikator für angelernte Holzfarbe denkbar
P _H	3	n	0	Keine Videodaten gegeben und/oder statische Hintergründe
K _S	1	j	1	Anwendbare Gradienten, aber mehrdeutig und nicht elliptisch
K _D	4	n	2	Startkontur notwendig, aber mit Kantenbild ermittelbar
R _R	1	n	1	Saatpunkte innerhalb der Stammschnittflächen notwendig

BM	S	U	W	Begründung (Kernargument/e)
R _W	2	j	1	Saatpunkte notwendig; sensibel bzgl. Parameteränderungen
R _S	1	n	1	Farben von OG^{st} inhomogen und texturiert
R _C	5	j	2	Variierende Farben und Texturen in OG^{st}
R _T	0	n	2	Texturen in OG^{st} , aber mehrdeutig bzgl. $\overline{O^{st}}$
R _X	0	n	2	$S \hat{=} O^{st}$ muss ermittelt und I gut segmentiert werden
OP _G	0	n	5	Adaption komplex, aber Gradienten und Farbe nutzbar
OP _B	0	n	3	Verfahren kaum in der Segmentierung angewandt
OP _X	0	n	3	Verfahren verschieden u. komplex, Eignung kaum abschätzbar
O _M	3	n	1	Statische Templates bzgl. Rotation und Skalierung invariant
O _T	4	n	3	Texturen in OG^{st} , aber mehrdeutig bzgl. $\overline{O^{st}}$
O _{Ho}	0	n	3	Gradienten um O^{st} charakteristisch; Mehrdeutigkeiten in I
O _E	0	n	4	Gradienten und Farben reduziert durch Eigenobjekt trainierbar
O _{HC}	0	n	4	Gradienten, Farbe und Textur von O^{st} trainierbar
O _X	0	n	3	Verschieden gut geeignet, Grundproblem Mehrdeutigkeiten in I

In der Tabelle 3.1 kennzeichnet die Spalte **BM** die jeweilige Basismethode. Die Spalte **S**⁵ gibt die Häufigkeit der Klasse in recherchierten Beiträgen zur Segmentierung von **organo**-Gruppen an. Die Spalte **U**⁶ kennzeichnet, ob die Klasse in den Voruntersuchungen untersucht wurde. Die Spalte **W**⁷ bewertet die Eignung subjektiv und in der letzten Spalte wird eine **Begründung** der Wertung gegeben.

Aus den Einzelbetrachtungen der Basismethoden ergibt sich, dass keine Basismethode die Zielstellung dieser Arbeit allein erfüllt. Erst eine Kombination und Adaption von geeigneten Basismethoden kann die Zielstellung erfüllen. Absteigend nach der Wertung (siehe Tabelle 3.1 bei $W > 3$) geordnet sind Graph-Cut (OP_G), empirisch ermittelte Holzfarbe (F) und einige Methoden der Objektdetektion (O_E, O_{HC}) geeignete Methoden für eine Kombination. Durch die ermittelten Basismethoden ergeben sich verschiedene Konzepte, welche sich in die zwei Klassen der Segmentierung f_s^h und f_s^{st} unterteilen. Die binäre Segmentierung des Schnittholzes f_s^h ist im Vergleich zur Multi-Objekt-Segmentierung der Stammschnittflächen f_s^{st} einfacher, da der Lösungsraum von f_s^{st} größer als von f_s^h ist (siehe Abschnitt 2.2.2). Es ist sinnvoll, zuerst die einfache Klasse f_s^h zu untersuchen, und anhand der Ergebnisse die schwierigere Klasse f_s^{st} zu betrachten.

Herleitung von Konzepten

Graph-Cut wurde als die Basismethode mit der besten Eignung ermittelt und ist analog eine Möglichkeit zur Lösung der binären Segmentierung f_s^h . Graph-Cut benötigt eine Vorsegmentierung oder Wahrscheinlichkeitsbilder I^p . Mit einer farbbasierten Holzsegmentierung lässt sich eine Vorsegmentierung und durch Abstandsmaße zu Holzfarbmodellen (z.B. Cluster im Farbraum) ein Wahrscheinlichkeitsbild ermitteln. Es muss allerdings zuvor untersucht werden, ob und wie es möglich ist, Holzfarbmodelle zu modellieren

⁵S (State of the Art): Häufigkeit von gar nicht (0) über durchschnittlich (3) zu häufig (6)

⁶U (untersucht): Klasse wurde untersucht: j (ja), Klasse wurde nicht untersucht n (nein)

⁷W (eigene Wertung): ungeeignet (0) über brauchbar (3) zu geeignet (6)

3.1 Problemdiskussion und Herleitung von Konzepten

oder anhand der Holzfarbe zu segmentieren. Die zweite und zugleich letzte geeignete Basismethodenklasse ist die Objektdetektion. Die Basismethoden der Objektdetektion sind verschieden und die Eignung noch ungewiss. Erst nach weiteren Analysen und Experimenten können eine oder mehrere Methoden bestimmt werden. Unabhängig der verwendeten Methode ist eine weitere Segmentierung notwendig, um O^{st} auf Basis eines detektierten Bereiches S^{win} zu segmentieren. Aus den beiden benannten Konzepten lässt sich ein drittes Konzept, die Kombination von Objektdetektion und Graph-Cut Segmentierung, herleiten. Anhand der detektierten Bereiche können Farbinformationen extrahiert werden, wodurch die Nutzung der empirischen Holzfarbe oder heuristischen Segmentierung entfällt. Weiterhin sollten sich mit Hilfe der detektierten Bereiche und dem binären Ergebnisbild von f_s^h die einzelnen Stammschnittflächen OG^{st} ermitteln lassen. Es ergeben sich folglich die drei groben Konzepte:

1. **Konzept der Schnittholzsegmentierung:** Automatische binäre Segmentierung des Schnittholzes mit Graph-Cut und empirischer Holzfarbe
2. **Konzept der Stammschnittflächendetektion:** Automatische Multi-Objekt-Segmentierung der Stammschnittflächen auf Basis einer Objektdetektion
3. **Konzept der Kombination:** Automatische Multi-Objekt-Segmentierung der Stammschnittflächen mit Graph-Cut und Objektdetektion

Zur Ausarbeitung der Konzepte müssen folgende korrespondierende **Fragestellungen und Forschungsinhalte** genauer betrachtet werden:

1. Wie kann Graph-Cut adaptiert und die Holzfarbe genutzt werden zur Segmentierung von Schnittholz?
2. Welche Methoden der Objektdetektion sind geeignet und wie können die Methoden zur Segmentierung von Stammschnittflächen adaptiert werden?
3. Wie kann eine binäre Segmentierung mit Graph-Cut mit Methoden der Objektdetektion zur Segmentierung der Stammschnittflächen kombiniert werden?

Die einzelnen Konzepte und Fragestellungen werden im Weiteren in der genannten Reihenfolge in den Unterkapiteln 3.2, 3.3 und 3.4 geprüft, untersucht und verfeinert.

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

In diesem Unterkapitel wird eine Graph-Cut basierte Schnittholzsegmentierung f_s^h in Kombination mit der Holzfarbe konzipiert und die Ergebnisse evaluiert. In Abschnitt 3.2.1 wird zuerst das Konzept der Segmentierung erarbeitet. Im Konzept ist Graph-Cut zum einen Bestandteil der Vorsegmentierung und die Kernmethode zur Segmentierung des gesamten Bildes. Da beide Graph-Cut Segmentierungen identisch sind, wird nur der finale Graph-Cut detailliert beschreiben. Der erste Schritt des Konzeptes ist eine Vorsegmentierung auf Basis der Holzfarbe und einer Graph-Cut-Segmentierung zur Abschätzung des Vorder- und Hintergrundes. Aus der Vorsegmentierung wird im nächsten Schritt ein Vorder- und Hintergrundmodell erstellt. Im letzten Schritt wird mit den Modellen ein gewichteter Graph erzeugt und mit Graph-Cut zur Segmentierung des Schnittholzes geschnitten. Die Vorsegmentierung wird in Abschnitt 3.2.2 untersucht. Die Modelle, die Erstellung des Graphen und Graph-Cut werden in Abschnitt 3.2.3 betrachtet. Im letzten Abschnitt 3.2.4 werden die Ergebnisse evaluiert und die wesentlichen Erkenntnisse zusammengefasst.

Im Folgenden wird von Vorder- und Hintergrund gesprochen, wie es in der Graph-Cut-Segmentierung üblich ist. Vordergrund FG ist die vereinigte Menge aller Schnittholzpixel in OG^{st} . Hintergrund BG ist die Menge aller nicht Schnittholzpixel und entspricht der Menge $\overline{OG^{st}}$. Zudem finden Wahrscheinlichkeitsbilder I^p Verwendung. $p_{x,y} \in I^p$ entspricht einer Wahrscheinlichkeit und korrespondiert zu $p_{x,y}$ in I_{wood} an der Position (x, y) . Es ergeben sich folgende Definitionen:

- Menge der Vordergrundpixel ist FG und die Menge der Hintergrundpixel ist BG
- Ein Vordergrundpixel wird mit px_{fg} und ein Hintergrundpixel mit px_{bg} notiert.
- Wahrscheinlichkeitsbild I^p des Vordergrunds ist I_{fg}^p und des Hintergrunds I_{bg}^p
- $I^p = \{p | \forall px \in I_{wood} \exists p : p \in [0, 1] \wedge p \in \mathbb{R}\}$
- $I_{fg}^p = \{p | \forall p \in I^p : p \text{ ist die Wahrscheinlichkeit, dass zugehöriges } px \in OG^{st}\}$
- $I_{bg}^p = \{p | \forall p \in I^p : p \text{ ist die Wahrscheinlichkeit, dass zugehöriges } px \notin OG^{st}\}$
- $px_{bg} \in BG; px_{fg} \in FG; |I_{bg}^p| = |I_{fg}^p| = |I_{wood}|$

3.2.1 Konzeption der Schnittholzsegmentierung

Die Segmentierung mit Graph-Cut basiert auf den Schnitt eines gewichteten Graphen $G = (V, E)$ mit den Knoten $v \in V$ und Kanten $e \in E$. Die Gewichte $w \in W$ der Kanten E sind die entscheidenden Faktoren für die Segmentierung. Je genauer die Gewichte bestimmt sind, umso genauer ist die Segmentierung. Die Gewichte werden in *region weights* (w_r) und *boundary weights* (w_b) unterteilt (Details siehe Abschnitt 2.3.2).

Die Gewichte w_r korrespondieren zu den Wahrscheinlichkeiten $p_{bg} \in I_{bg}^p$ und $p_{fg} \in I_{fg}^p$. Die Wahrscheinlichkeiten (p_{bg}, p_{fg}) korrespondieren wiederum zu einem Pixel im Inputbild

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

I_{wood} . Die Vordergrundwahrscheinlichkeiten p_{fg} sind die Gewichte zum Terminal-Knoten (*source*) und die Hintergrundwahrscheinlichkeiten p_{bg} zum Terminal-Knoten (*sink*). Die Betrachtung der Gewichte als Wahrscheinlichkeitsbilder wurde in dieser Arbeit verwendet, um die folgenden Erklärungen zu vereinfachen. Die Wahrscheinlichkeiten ($p_{fg}, p_{bg} \in \mathbb{R}$) liegen im Intervall $[0, 1]$. Eine Wahrscheinlichkeit von $p_{fg} = 1$ bedeutet Vordergrund und 0 Nicht-Vordergrund. Selbiges gilt für p_{bg} in Beziehung zum Hintergrund.

Zur Ermittlung der Wahrscheinlichkeiten ist ein gutes Vorder- und Hintergrundmodell notwendig sowie eine gute Metrik (Abstandsfunktion). Auch müssen die Modelle erstellt werden, wofür Trainingsdaten notwendig sind. Am besten als Trainingsdaten sind entsprechend dem Modell einige Pixel vom Vorder- (FG) und einige vom Hintergrund (BG) geeignet. Eine gute Ermittlung der Pixel ist essentiell und muss durch eine Vorsegmentierung, heuristischer Annahmen oder Nutzer-Interaktion geschehen. Zielstellung dieser Arbeit ist eine automatische Segmentierung, weshalb eine Vorsegmentierung mit möglichst wenig heuristischen Annahmen erfolgen muss. Eine Vorsegmentierung anhand der Holzfarbe bietet sich daher an. Die geschilderte Argumentation führt zu einem groben Konzept, welches in Abbildung 3.11 zu sehen ist.

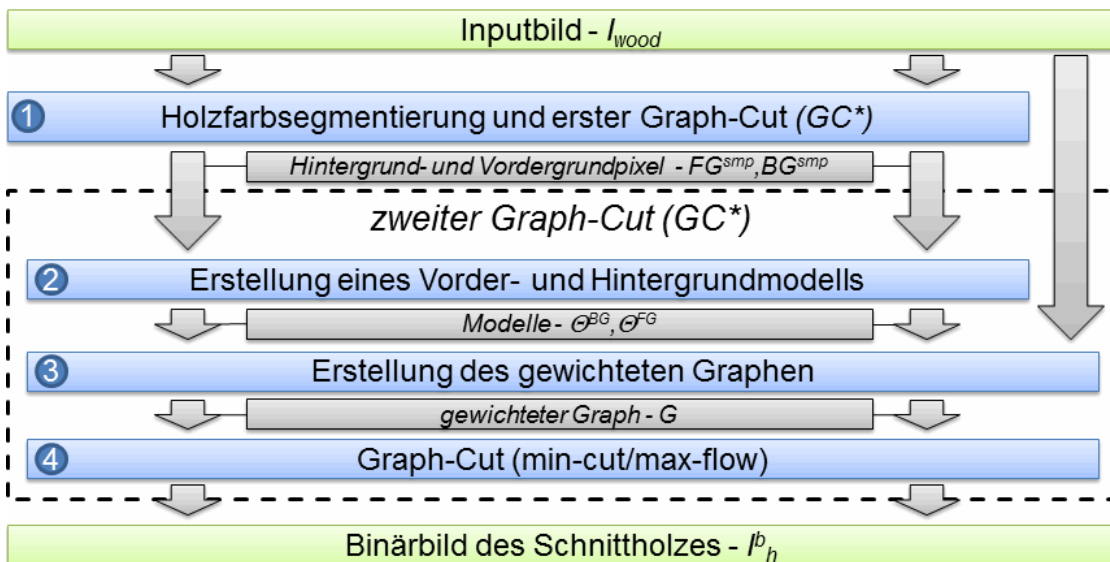


Abbildung 3.11: Grobes Konzept zur binären Segmentierung von Schnittholz mit Graph-Cut und Holzfarbe.

Das Konzept in Abbildung 3.11 wird in vier Schritte unterteilt. Im ersten Schritt wird das Inputbild I_{wood} mit Hilfe der Holzfarbe, einigen heuristischen Annahmen und einer ersten Graph-Cut-Segmentierung segmentiert. Die Graph-Cut-Segmentierung ist die Erste eine von Zweien. Beide sind identisch und enthalten die Schritte zwei, drei und vier des Konzeptes, die in Abbildung 3.11 verdeutlicht sind. Ergebnis der Vorsegmentierung sind abgeschätzte Pixel (px_{fg}, px_{bg}) vom Vorder- und Hintergrund. Die Pixel (px_{fg}, px_{bg}) können fehlerbehaftet sein. Unter der **Annahme**, dass die Abschätzung genau ist, ergeben

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

sich zwei weitere Teilmengen (FG^{smp}, BG^{smp}) .

$$FG^{smp} \subseteq FG \subset I_{wood}; BG^{smp} \subseteq BG \subset I_{wood} \quad (3.3)$$

$$FG \cap BG = \emptyset; FG \cup BG = I_{wood} \quad (3.4)$$

$$px_{fg} \in FG^{smp}; px_{bg} \in BG^{smp}; px_{fg}, px_{bg} \in I_{wood} \quad (3.5)$$

Im zweiten Schritt wird mit der Menge der Vordergrundpixel FG^{smp} ein Vordergrundmodell Θ^{FG} und mit der Menge der Hintergrundpixel BG^{smp} ein Hintergrundmodell Θ^{BG} erstellt. Im dritten Schritt werden die Modelle Θ^{FG} und Θ^{BG} durch die Wahrscheinlichkeitsbilder I_{fg}^p und I_{bg}^p repräsentiert. Mit den Gradienten von I_{wood} und den Modellen (I_{fg}^p, I_{bg}^p) wird ein gewichteter Graph G erzeugt. Im vierten und letzten Schritt wird der gewichtete Graph G mit Graph-Cut geschnitten. Die sich ergebenden Partitionen (Cliques) des Graphen G bilden Segmente im Bildraum. Das Ergebnis ist ein binär segmentiertes Bild I_h^b vom Schnittholz. Aus dem groben Konzept ergeben sich die folgenden **Fragestellungen und Forschungsinhalte**:

- Gibt es Charakteristiken der Holzfarbe, die zur Segmentierung nutzbar sind?
- Müssen Heuristiken angenommen werden, um sicher Vorder- (FG^{smp}) und Hintergrundpixel (BG^{smp}) abzuschätzen?
- Wie lässt sich Graph-Cut an die Segmentierungsaufgabe adaptieren?
 - Welche Modelle Θ sind nutzbar und geeignet?
 - Wie können die Gewichte des Graphen G bestimmt werden?

Nachfolgend wird in Abschnitt 3.2.2 die Fragestellung der Holzfarbe und der heuristischen Annahmen zur Ermittlung von Trainingsdaten (FG^{smp}, BG^{smp}) untersucht. Es wird analog Schritt eins des Konzeptes detailliert. In Abschnitt 3.2.3 wird die Fragestellung der Graph-Cut-Segmentierung betrachtet. Es werden verschiedene Methoden zum Setzen der Gewichte w des Graphen G auf Basis von FG^{smp} und BG^{smp} untersucht und beschrieben. Analog werden die Schritte zwei, drei und vier des groben Konzeptes erarbeitet und detailliert. Die Ergebnisse der Segmentierung unter Anwendung der verschiedenen Methoden werden in Abschnitt 3.2.4 gegenübergestellt und diskutiert.

3.2.2 Holzfarbsegmentierung zur Vorder- und Hintergrundabschätzung

Die Vorsegmentierung in diesem Abschnitt hat das Ermitteln einiger Pixel vom Vordergrund und Hintergrund (FG^{smp}, BG^{smp}) zum Ziel. Die Pixelmengen FG^{smp} und BG^{smp} müssen eine aussagekräftige Stichprobe von FG und BG enthalten, damit adäquate Modelle $(\Theta^{FG}, \Theta^{BG})$ erstellt werden können. Auftretende geringe Fehler, wie $px_{fg} \in FG^{smp}$, aber $px_{fg} \notin FG$, können durch die optimierungsbasierte Segmentierung mit Graph-Cut kompensiert werden. Folglich ist der Begriff Abschätzung der Stichproben (FG^{smp}, BG^{smp}) an dieser Stelle angebracht. Zur Konzeption der Vorsegmentierung und Abschätzung der Stichproben wird zuerst die Holzfarbe und ihre Ausprägung im Farbraum untersucht. Anhand der Ergebnisse wird anschließend die Vorsegmentierung und Abschätzung von FG^{smp} und BG^{smp} betrachtet.

Analyse der Holzfarbe im Farbraum

Die Basis zur Analyse der Holzfarbe bilden die Vorder- und Hintergrundpixel FG_{gt} und BG_{gt} von 71 Evaluierungsbildern (siehe Anhang C.2). Zur Erhebung der Pixelmengen FG_{gt} und BG_{gt} wurde die Grundwahrheit **GT** (*Ground Truth*) zu den 71 Bildern erstellt. Alle Schnittholzpixel px_{fg} wurden manuell markiert, womit zu jedem Bild eine korrespondierende Bildmaske für Vorder- und Hintergrund erstellt wurde. Die Mengen FG_{gt} und BG_{gt} ergeben sich aus der Vereinigung der jeweiligen Pixelmengen FG_i und BG_i zum Bild i wie folgt:

$$FG_{gt} = FG_1 \cup FG_2 \cup \dots \cup FG_i \dots \cup FG_{71} \quad (3.6)$$

$$BG_{gt} = BG_1 \cup BG_2 \cup \dots \cup BG_i \dots \cup BG_{71} \quad (3.7)$$

Zur Untersuchung verschiedener Farbräume (RGB, Luv, YCrCb, HSV, siehe [KS08]) wurden die Bilder I_{wood} erst entsprechend konvertiert. Anschließend wurde zu FG_{gt} und BG_{gt} ein 3D-Histogramm mit 256^3 *bins* erstellt. Jedes *bin* enthält die Häufigkeit einer Farbe im jeweiligen dreidimensionalen Farbraum. Zur Untersuchung der Farbräume ist eine Visualisierung der Daten notwendig. Eine Möglichkeit ist es die Visualisierung direkt in 3D durch Punktwolken im Farbwürfel oder einer direkten Volumenvisualisierung vorzunehmen, wie bspw. bei **VioNeS** (siehe [Gut08] [Ung+09]). Eine 3D-Darstellung erschwert jedoch die Analyse von Beziehungen zwischen zwei Dimensionen. Um die Beziehungen von zwei Achsen des jeweiligen Farbraumes untersuchen zu können, wird stattdessen ein Klassiker der wissenschaftlichen Visualisierung verwendet, die *2D-Scatterplots* (siehe [SM00a], [PB07]). Zur Visualisierung der Häufigkeit eines Farbwertes im *2D-Scatterplot* wird die Häufigkeit auf den Grauwert abgebildet. Je höher der Grauwert, umso häufiger tritt die Farbe auf. Problematisch ist, dass Peaks im Histogramm zu vielen, kaum mehr erkennbaren, geringen Grauwerten führen. Um das Problem zu umgehen, wurde zusätzlich die 3. Wurzel der Werte gezogen und alle Werte bzgl. $[0, 255]$ normalisiert. Außerdem ist zur besseren Interpretation der Visualisierung der jeweilige Farbwert auf die x- und y-Achse abgebildet. Alle Visualisierungen der Farbräume RGB, Luv, YCrCb und HSV sind im Anhang C.1 zu finden. In Abbildung 3.12 sind ausgewählte *2D-Scatterplots* vom RGB- und HSV-Farbraum zu sehen.

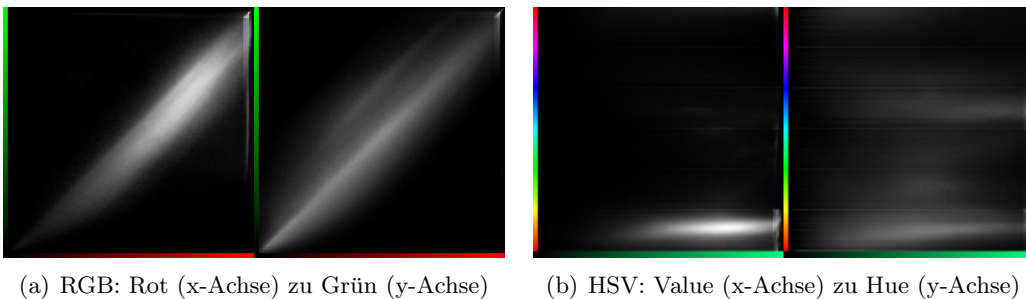


Abbildung 3.12: Visualisierung und Gegenüberstellung von FG_{gt} (linke Bildhälfte) zu BG_{gt} (rechte Bildhälfte) mit *2D-Scatterplots*.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Die *2D-Scatterplots* in Abbildung 3.12 zeigen, dass sich FG_{gt} und BG_{gt} stark überschneiden. Weiterhin sind sowohl die Pixel von FG_{gt} als auch von BG_{gt} im RGB-Farbraum verstreut. Es ist ein elliptisch geformter, großer Cluster entlang der kompletten Diagonalen zu erkennen. Im HSV-Farbraum sind die Pixel von BG_{gt} entlang des Farbtones (Hue) verstreut. Die Pixel von FG_{gt} hingegen bilden einen Cluster beim Farbton gelb bis rot. Der entsprechende Cluster bildet sich aber auch bei den Pixeln von BG_{gt} schwach aus. Durch die starken Überlappungen in den unterschiedlichsten Farbräumen (siehe auch Anhang C.1) ist es nicht möglich eine Holzfarbe, die überschneidungsfrei zum Hintergrund ist, abzuleiten.

Zur weiteren Analyse und Findung einer Trennung im Farbraum wird im Folgenden der Hintergrund unterteilt betrachtet. In einem typischen Holzpolterbild I_{wood} gibt es *i.d.R.* drei Arten von Hintergründen, den unteren Teil (z.B. Gras, Wiese), den Bereich zwischen den Baumstämmen und den oberen Teil (z.B. Wald, Himmel). Aufgrund der vielfältigen Möglichkeiten des Bildinhaltes variiert der untere und obere Hintergrund in der Farbgebung sehr stark. Demgegenüber ist der Bereich zwischen den Baumstämmen *i.d.R.* dunkler als die Stammschnittflächen OG^{st} , da wenig Licht in die Zwischenräume einfällt. Im Bildraum setzen sich die **Baumstammzwischenräume** zumeist aus Rinde, Rinde im Schatten und Bereichen ohne Licht zusammen. Die Varianz in der Farbgebung ist geringer als der restliche Hintergrund. Zur Untersuchung der Farbgebung der **Baumstammzwischenräume** soll ein Teilbild vom Bild I_{wood} mit der Höhe h und Weite w betrachtet werden:

$$I_{wood}^{sub} = \left\{ px_{x,y} \mid \forall px_{x,y} \in I_{wood} : x \in \left[\frac{w}{3}, \frac{2 * w}{3} \right] \wedge y \in \left[\frac{h}{3}, \frac{2 * h}{3} \right] \right\} \quad (3.8)$$

Das Teilbild I_{wood}^{sub} liegt im Zentrum des Bildes I_{wood} (siehe Abbildung 3.15(a)) und enthält nur Schnittholzpixel und Pixel der Baumstammzwischenräume. Die Vereinigung der Vorder- und Hintergrundpixel aller 71 **GT**-Bilder für das Teilbild I_{wood}^{sub} führt zu den

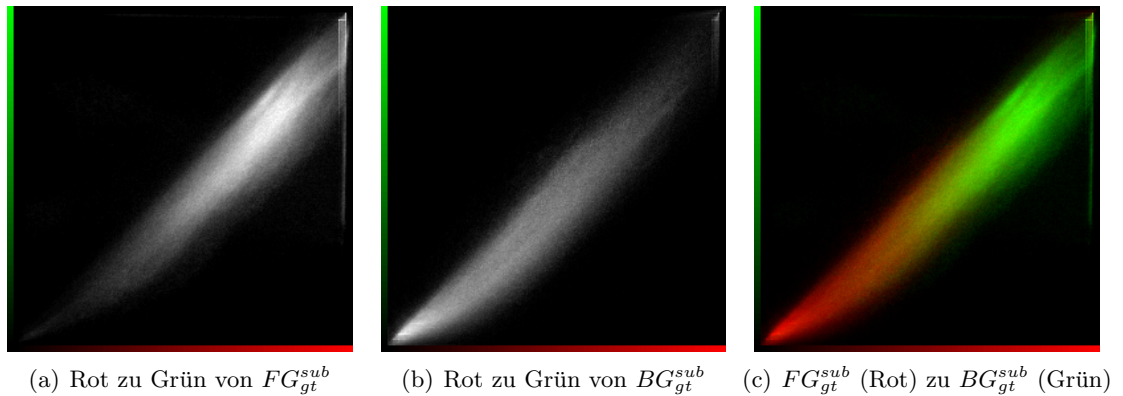


Abbildung 3.13: Visualisierung des RGB Farbraumes von Rot (x-Achse) zu Grün (y-Achse) mit *2D Scatterplots* und Differenzdarstellung von FG_{gt}^{sub} (Rot-Kanal) zu BG_{gt}^{sub} (Grün-Kanal).

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

reduzierten Mengen FG_{gt}^{sub} und BG_{gt}^{sub} . Die Visualisierung der Mengen inklusive einer Differenzdarstellung ist in Abbildung 3.13 und 3.14 zu sehen. In der Differenzdarstellung (Abbildung 3.13(c) und 3.14(c)) ist die Häufigkeit von FG_{gt}^{sub} auf dem Rot- und von BG_{gt}^{sub} auf dem Grünkanal abgebildet. Je roter oder grüner die Punkte im $2D$ -Scatterplot sind, umso weniger überschneiden sich die Farbbereiche. Die Visualisierungen zeigen ähnlich

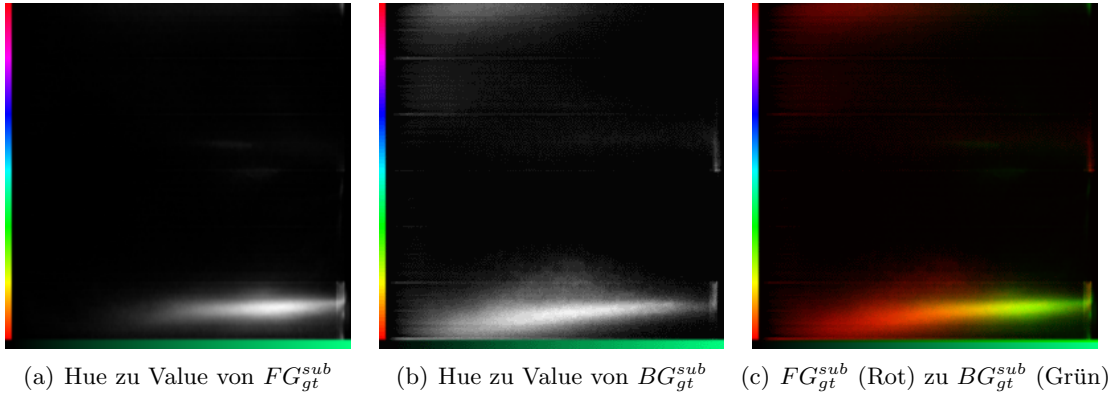


Abbildung 3.14: Visualisierung des HS Farbraumes von Hue (x-Achse) zu Value (y-Achse) mit $2D$ Scatterplots und Differenzdarstellung von FG_{gt}^{sub} (Rot-Kanal) zu BG_{gt}^{sub} (Grün-Kanal).

wie in Abbildung 3.12 eine Überschneidung der Farbbereiche. Die Überschneidungen sowie die Streuungen sind allerdings geringer als bei FG_{gt} und BG_{gt} . Die Differenzdarstellungen zeigen deutlich, dass es überschneidungsfreie Bereiche von FG_{gt}^{sub} und BG_{gt}^{sub} gibt. Im RGB Farbraum ist BG_{gt}^{sub} im dunklen Bereich deutlich stärker ausgeprägt als FG_{gt}^{sub} durch einen geringen Rot-, Grün- und Blauanteil. Auch im HSV -Farbraum ist im Verhältnis zu FG_{gt}^{sub} auf der Helligkeitsachse (Value) eine Verschiebung von BG_{gt}^{sub} ins Dunkle zu erkennen. Weiterhin ist ein Cluster bei FG_{gt}^{sub} im hellen gelben bis roten Farbton (hue) zu sehen.

Zusammenfassend ergibt sich, dass sich der Vordergrund (FG) mit dem Hintergrund (BG) im Farbraum überschneidet und eine einfache Trennung anhand der Farbe nicht möglich ist. Betrachtet man den mittleren Bildbereich, in dem nur Schnittholzpixel und Pixel der Baumstammzwischenräume zu sehen sind, so ergeben sich Möglichkeiten der farbbasierten Trennung. Zum einen liegt der Farbton (Hue) des Vordergrundes im gelben bis roten Bereich und zum anderen ist der Hintergrund dunkler.

Vorsegmentierung und Abschätzung von Vorder- und Hintergrundpixeln

Ziel der Vorsegmentierung ist es, eine aussagekräftige Stichprobe der Vorder- (FG) und Hintergrundpixel (BG) zu ermitteln. Aus der Analyse der Holzfarbe hat sich ergeben, dass der Vorder- und Hintergrund im mittleren Bildbereich anhand der Farbe trennbar ist. Dagegen ist die Trennung im ganzen Bild nicht so einfach möglich. Um eine repräsentative Stichprobe (FG^{smp} , BG^{smp}) von (FG , BG) zu erhalten, wird eine **Restriktion in der**

Bildaufnahme eingeführt. Die Einführung der Restriktion ermöglicht es, die Eignung von Graph-Cut zur Segmentierung von Schnittholz ohne potenzielle Fehlerquellen in der Vorsegmentierung zu untersuchen. Folgende Bedingungen müssen durch die Restriktion in der Bildaufnahme erfüllt sein:

- Zentraler Bildbereich (I_{wood}^{sub}) enthält Schnittholz und Baumstammzwischenräume
- Oberer Bildbereich (I_{bg1}^{sub}) enthält oberen Hintergrund, wie Himmel und Bäume
- Unterer Bildbereich (I_{bg2}^{sub}) enthält unteren Hintergrund, wie Boden und Gras

Die Bildhöhe von I_{bg1}^{sub} und I_{bg2}^{sub} ist $\frac{h}{20}$, wobei h die Bildhöhe von I_{wood} ist. Die Bildweite ist gleich der Bildweite von I_{wood} . Eine Illustration der Teilbilder ist in 3.15(a) zu sehen.

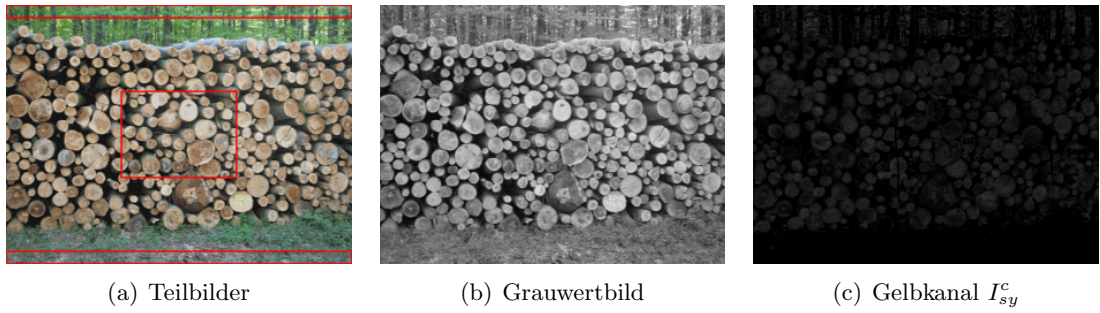


Abbildung 3.15: Von links nach rechts sind rot umrahmt die Teilbilder, das Grauwertbild und der spezielle Gelbkanal von I_{wood} zu sehen.

Mit den Restriktionen und den Teilbildern ($I_{wood}^{sub}, I_{bg1}^{sub}, I_{bg2}^{sub}$) können nahezu fehlerfrei Pixel des oberen, unteren und mittleren Hintergrundes ermittelt werden. Zur Abschätzung von Hintergrundpixeln BG^{smp} werden alle Pixel aus I_{bg1}^{sub} und I_{bg2}^{sub} mit den segmentierten Pixeln in I_{wood}^{sub} vereinigt. Eine Vorsegmentierung des mittleren Bildbereiches I_{wood}^{sub} führt zu den beiden Mengen FG_{center}^{sub} und BG_{center}^{sub} , wobei FG_{center}^{sub} Schnittholzpixel und BG_{center}^{sub} Pixel der Baumstammzwischenräume enthält. Final ergeben sich die gesuchten Pixelmengen (FG^{smp}, BG^{smp}) durch:

$$FG^{smp} = FG_{center}^{sub} \quad (3.9)$$

$$BG^{smp} = I_{bg1}^{sub} \cup I_{bg2}^{sub} \cup BG_{center}^{sub} \quad (3.10)$$

Mit dieser Methodik werden die Vordergrundpixel FG^{smp} durch Schnittholzpixel des zentralen Bildbereiches abgeschätzt. Die Farben der Stammschnittflächen OG^{st} sind in einem Bild ähnlich, wodurch eine ausreichende Stichprobe durch den zentralen Bildbereich gegeben ist.

Die entsprechende **Vorsegmentierung** basiert auf den Erkenntnissen der Holzfarbanalyse. Es hat sich gezeigt, dass im zentralen Bildbereich (I_{wood}^{sub}) Schnittholzpixel heller sind und einen gelben bis roten Farbton aufweisen. Beide Erkenntnisse werden durch zwei

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

verschiedene binäre Segmentierungen aufgegriffen. Zum einen wird das Teilbild (I_{wood}^{sub}) in ein Graustufenbild konvertiert und zum anderen ein spezieller Gelbkanal extrahiert. Beide Ergebnisbilder werden anschließend mit der Methode von Otsu [Ots79] segmentiert. Der spezielle Gelbkanal folgt der Annahme, dass sich im RGB-Raum der Gelbton durch gleiche Rot- und Grünanteile kennzeichnet. Der spezielle Gelbkanal I_{sy}^c (*special yellow*) ergibt sich im Bild I_{wood} durch die Konvertierung der Rot-, Grün- und Blauanteile jedes Pixels von $px = (R, G, B)$ zu $px_{sy} = (va_{sy})$ durch:

$$va_{sy} = \max(\min(R - B, G - B), 0) \quad (3.11)$$

Die genutzte Formel wurde empirisch und visuell auf verschiedenen Bildern I_{wood} untersucht und führt zu einer deutlichen Hervorhebung von Schnittholzpixeln. Der spezielle Gelbkanal für das gesamte Bild I_{wood} ist in Abbildung 3.15(c) und für das zentrale Teilbild in Abbildung 3.16(b) zu sehen.

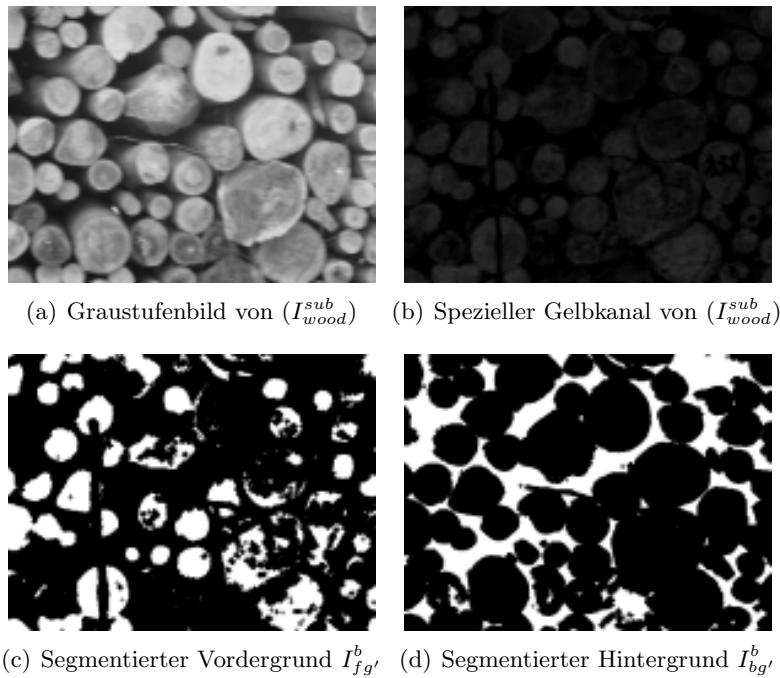


Abbildung 3.16: Oben sind die konvertierten Bilder (Graustufenbild, spezieller Gelbkanal) und unten die Ergebnisse der farbbasierten Vorsegmentierung von I_{wood}^{sub} zu sehen.

Zur weiteren Ermittlung der Vorder- (FG_{center}^{sub}) und Hintergrundpixel (BG_{center}^{sub}) werden die binären Segmentierungsergebnisse vom Graustufenbild I_{gr}^b und vom speziellen

Gelbkanal I_{sy}^b wie folgt miteinander verschnitten⁸:

$$I_{fg}^b = I_{gr}^b \overset{\frown}{\cap} I_{sy}^b; \quad I_{bg}^b = \overline{I_{gr}^b} \overset{\frown}{\cap} \overline{I_{sy}^b} \quad (3.12)$$

Zur Sicherstellung, dass nur Schnittholzpixel und Pixel der Zwischenräume segmentiert sind, wird zusätzlich eine Erosion (siehe [Jah05], [Ton05] und Anhang A.1) auf I_{fg}^b und I_{bg}^b mit einen 5×5 Kernel K durchgeführt:

$$I_{fg'}^b = I_{fg}^b \ominus K; \quad I_{bg'}^b = I_{bg}^b \ominus K; \quad (3.13)$$

Die Erosion führt zu den Bildern $I_{fg'}^b$ und $I_{bg'}^b$, welche in Abbildung 3.16 zu sehen sind. Die Pixel der Bilder mit dem Label 1 sind abgeschätzte Stichproben des mittleren Bereiches von FG und BG und ergeben die Pixelmengen FG_1^{sub} und BG_1^{sub} . Um eine genauere Stichprobe zu erhalten wird das Bild I_{wood}^{sub} anhand der Stichproben (FG_1^{sub}, BG_1^{sub}) mit Graph-Cut segmentiert. Die Graph-Cut-Segmentierung erfolgt nach dem gleichen Schema, wie die Segmentierung des gesamten Bildes I_{wood} im nachfolgenden Abschnitt 3.2.3. Das Ergebnis der Segmentierung ist das Binärbild I_{center}^b , welches sehr genau Schnittholz von Nicht-Schnittholz trennt. Alle Pixel mit dem Label 1 in I_{center}^b werden der Menge FG_{center}^{sub} und alle Pixel mit dem Label 0 der Menge BG_{center}^{sub} zugeordnet. Abbildung 3.17 zeigt das Ergebnis der ersten Graph-Cut-Segmentierung.

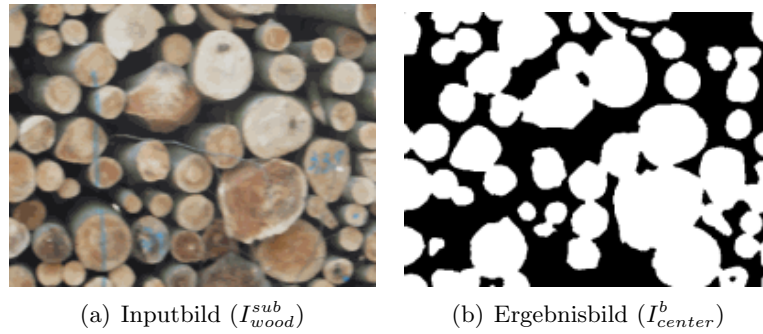


Abbildung 3.17: Das Inputbild (links) und Ergebnisbild (rechts) der ersten Graph-Cut-Segmentierung.

3.2.3 Erstellung des gewichteten Graphen und Graph-Cut

Der Schwerpunkt in der Graph-Cut-Segmentierung liegt in der Bestimmung geeigneter Gewichte. Zur Bestimmung der Gewichte (w_r, w_b) des Graphen G anhand von (FG^{smp}, BG^{smp}) werden in diesem Abschnitt mehrere Varianten beschrieben. Zur Ermittlung der Regionengewichte w_r gilt es, geeignete Methoden zur Ermittlung der

⁸ Der Schnitt mit dem Symbol ($\overset{\frown}{\cap}$) bedeutet, dass die Pixel an der gleichen Position logisch UND verknüpft werden. Normalerweise ist das Symbol (\cap) in der Binärbildverarbeitung üblich. Zur Vermeidung von Mehrdeutigkeiten wird in dieser Arbeit das Symbol speziell gekennzeichnet. Das Symbol $\overline{I^b}$ kennzeichnet das Komplement von I^b (siehe auch Anhang A.1).

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

Wahrscheinlichkeitsbilder I_{fg}^p und I_{bg}^p zu adaptieren und zu entwickeln. Die Wahrscheinlichkeitsbilder korrespondieren zu den Gewichten des Graphen G und beeinflussen die Segmentierung des Bildes durch Graph-Cut. Je genauer die Wahrscheinlichkeiten der Zugehörigkeit eines Pixels px zum Vorder- oder Hintergrund ermittelt werden kann, umso genauer ist die Graph-Cut-Segmentierung. Die Ermittlung erfolgt auf Basis der abgeschätzten Stichproben (FG^{smp}, BG^{smp}) vom Vorder- und Hintergrund. Folglich muss für jedes Pixel $px_{x,y} \in I_{wood}$ eine Möglichkeit gefunden werden, die Wahrscheinlichkeiten $(p1, p2)$ der Zugehörigkeit zum Vordergrund $p1_{x,y} \in I_{fg}^p$ und zum Hintergrund $p2_{x,y} \in I_{bg}^p$ zu ermitteln. Es ergeben sich verschiedene Methoden zur Ermittlung von $(p1, p2)$.

Abstrakt betrachtet, gibt es in allen Methoden eine Datenstruktur DS , eine Metrik (Abstandsfunktion) m , ein Wahrscheinlichkeitsbild I^p und Parameter PA zum Justieren der Metrik und/oder der Datenstruktur. Aus diesem Grund soll ein Modell Θ durch das folgende Tripel definiert werden:

$$\Theta = (m, DS, I^p, PA) \quad (3.14)$$

Die Metrik m bildet ein Pixel px unter Nutzung der gefüllten Datenstruktur DS auf eine Wahrscheinlichkeit p ab und soll wie folgt definiert werden:

$$m : I \rightarrow \mathbb{R}, px \mapsto p; \quad p \in [0, 1] \quad (3.15)$$

Das Ergebnis der Abbildung ist das Wahrscheinlichkeitsbild I^p . Die Datenstruktur DS ist ein strukturierter Datencontainer, welcher die abgeschätzten Pixel im entsprechenden Farbraum unverändert oder aufbereitet beinhaltet. Theoretisch ist in DS und deren enthaltenen Pixel jeder dreidimensionale Farbraum anwendbar. Die Parameter PA steuern und justieren die Algorithmen. Sie sollen als n -Tupel

$$PA = (x_i)_{i \in \{1, \dots, n\}}; \quad x_i \in \mathbb{R}^{>0} \quad (3.16)$$

notiert werden. Je nach Methode ist die Anzahl n und die Bedeutung der Parameter verschieden. Zur Normalisierung der Wahrscheinlichkeiten auf $p \in [0, 1]$ werden in dieser Arbeit die Metriken um die e-Funktion $f(x) = 1 - e^{-x}$ erweitert. Wie in Abbildung 3.18 dargestellt, konvergiert die Funktion gegen 1 und kompensiert durch den Verlauf Peaks der Werte in der Datenstruktur.

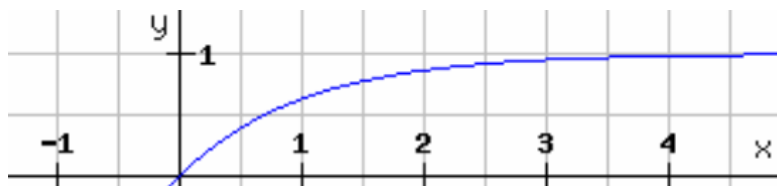


Abbildung 3.18: Darstellung von $f(x) = 1 - e^{-x}$

Im Folgenden werden die Methoden 3D-Histogramm, *Clustering* mit K-Mean und **GMM**, sowie die eigene Methode **KD-NN**, welche auf einer Dichteabschätzung im Farbraum basiert, adaptiert und beschrieben. Es wird nicht unterschieden in Vorder- und

Hintergrund, da pro Methode die Bestimmung der Metrik und Datenstruktur für Θ^{FG} und Θ^{BG} gleich ist und die Beschreibungen verständlicher werden. Die Ermittlung der Randgewichte w_b und die Graph-Cut-Segmentierung werden abschließend beschrieben.

3D-Histogramm

Die einfachste Methode zur Ermittlung von Wahrscheinlichkeiten ist die Nutzung eines 3D-Histogramms. Das Modell

$$\Theta_{his} = (m_{his}, \text{3D-Histogramm}, I_{his}^p, (h_s, \beta)) \quad (3.17)$$

besteht aus der Metrik m_{his} , einem 3D-Histogramm, und erzeugt das Wahrscheinlichkeitsbild I_{his}^p . Die Parameter sind:

- h_s - Skalierungsfaktor zur Ermittlung der Anzahl der Bins (Voxel)
- β - ein freier Skalierungsfaktor zum Justieren der Wahrscheinlichkeit

Ein Histogramm besteht aus Bins⁹. Ein Bin in einem Histogramm enthält die Anzahl der Pixel des entsprechenden Farbbereiches. Im dreidimensionalen Fall, dem 3D-Histogramm, ist ein **Bin** ein **Voxel**. Alle Trainingspixel FG^{smp} oder BG^{smp} werden in ein 3D-Histogramm einsortiert. Zur Rauschreduzierung wird das Histogramm mit $h_s \in [0, 1]$ skaliert. Das 3D-Histogramm hat pro Farbkanal $hb = \lfloor 256 * h_s \rfloor$ Bins und folglich über alle drei Kanäle hb^3 Bins. In Histogrammen können Peaks durch viele Pixel der gleichen Farbe entstehen. Durch die Peaks kommt es zu vielen kleinen Werten. Zur Vorverarbeitung der Peaks wird das 3D-Histogramm normalisiert, so dass der maximale Wert 1 ist und alle Werte der Bins b_v im Intervall $[0, 1]$ liegen. Durch Multiplikation von b_v mit der Summe der Werte aller Bins werden die Werte größer und es ergibt sich eine Abhängigkeit zu allen Trainingspixeln. Mit der e-Funktion werden die Werte erneut normalisiert und Peaks kompensiert. Die entsprechende Metrik dazu ist:

$$p_{his} = 1 - e^{\beta * sum_h * b_v} \quad (3.18)$$

Der Wert eines Bins ist mit b_v und die Summe aller b_v des 3D-Histogrammes mit sum_h gekennzeichnet. Die Metrik erzeugt das Wahrscheinlichkeitsbild I_{his}^p , welches für den Graphen G die Regionengewichte (*region weights*) enthält. Die Regionengewichte müssen für einen optimalen Graph-Cut gut mit den Randgewichten (*boundary weights*) abgestimmt sein. Aus diesem Grund wurde der freie Skalierungsfaktor β in die Metrik integriert. Zur Erzeugung von I_{his}^p wird für jedes Pixel px von I_{wood} das korrespondierende Bin berechnet und die Metrik m_{his} angewendet.

Clustering mit K-Mean

K-Mean ist ein klassischer Algorithmus (siehe [Ste56], [HW79], [Llo82]) zur Ermittlung von Clustern in einem n-dimensionalen Raum. Der Algorithmus ist bis heute aktuell

⁹Ein Bin bezeichnet in der CV ein Intervall / Bereich einer diskreten Zerlegung.

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

und es gibt viele Beschleunigungen und Verbesserungen, wie bspw. in [Kan+02a] und [Kan+02b] beschrieben. In der CV wird *K-Mean* häufig zur Ermittlung von k Clustern im Farbraum eingesetzt. Das Ergebnis von *K-Mean* sind die Mittelwerte (*mean*) von k Clustern. Durch die k Mittelwerte:

$$KM_k = (\mu_i)_{i \in \{1, \dots, k\}}; \quad \mu_i = (x, y, z); \quad x, y, z \in \mathbb{R}^{\geq 0} \quad (3.19)$$

im dreidimensionalen Farbraum und der Metrik m_{km} ergibt sich das Modell:

$$\Theta_{km} = (m_{km}, KM_k, I_{km}^p, (k, \gamma)) \quad (3.20)$$

Das Modell erzeugt das Wahrscheinlichkeitsbild I_{km}^p mit den Parametern:

- k - Anzahl der Cluster (Mittelwerte)
- γ - ein freier Skalierungsfaktor zum Justieren der Wahrscheinlichkeit

Für die Erzeugung von I_{km}^p auf Basis von k Clustern und deren Mittelwerte μ_i gibt es verschiedene Möglichkeiten. Eine Möglichkeit ist, die Distanz d_{min} vom Pixel px zum dichtesten Cluster μ_i zu ermitteln und eine Wahrscheinlichkeit p abzuleiten. Eine andere Möglichkeit ist, alle Distanzen von px zu den Mittelwerten der Cluster KM zu ermitteln, den Mittelwert der Distanzen zu berechnen und dann p zu ermitteln. Zusätzlich ist es auch möglich, die Anzahl der Pixel pro Cluster in die Berechnung einzubeziehen, wie bspw. die Gewichtung der Distanzen anhand der Pixel pro Cluster. In empirischen Untersuchungen erzielte die Nutzung der minimalen Distanz d_{min} die besten Ergebnisse mit der Metrik:

$$p_{km} = 1 - e^{\gamma * d_{min}} \quad (3.21)$$

Zur Erzeugung von I_{km}^p werden zuerst k Cluster von FG^{smp} oder BG^{smp} ermittelt. Anschließend wird für jedes Pixel px von I_{wood} der euklidische Abstand d_{min} zum dichtesten Mittelpunkt μ_i berechnet und die Metrik m_{km} angewendet.

Clustering mit Gaussian Mixture Models (GMM-EM)

Gaussian Mixture Models (GMM) sind eine Möglichkeit, die Dichte im Farbraum kompakt zu modellieren (siehe [HTF01], [Pre+07]). Ein GMM besteht aus k Gaussian-Modellen. Ein Gaussian-Modell (*Single Gaussian Model*) wird durch den Mittelwert μ und die Kovarianzmatrix Σ beschrieben. Ein Gaussian-Modell bildet im dreidimensionalen Raum einen ellipsoidförmigen Cluster und approximiert besser als der kugelförmige Cluster von *K-Mean*. Ein *Gaussian Mixture Model* ist eine Vereinigung von k Gaussian-Modellen. Sie werden vielfältig in der CV angewandt, wie bspw. zur Modellierung von Farb- und/oder Merkmalsverteilungen (siehe [Hai+04], [XP11]). Ein GMM soll durch das Tupel:

$$GMM = (\Sigma_1, \dots, \Sigma_k, \mu_1, \dots, \mu_k) \quad (3.22)$$

beschrieben werden, wobei Σ_i die Kovarianzmatrix und μ_i der Mittelwert des i 'ten Gaussian-Modells ist. Mit GMM und der Metrik m_{gm} ergibt sich das Modell:

$$\Theta_{gmm} = (m_{gmm}, GMM, I_{gmm}^p, (k)) \quad (3.23)$$

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Das Modell erzeugt das Wahrscheinlichkeitsbild I_{gmm}^p mit der Metrik m_{gmm} und dem Parameter k , der Anzahl der Gaussian-Modelle (Cluster). Die Metrik m_{gmm} summiert die gewichteten Mahalanobis-Distanzen der Pixel $px = (va_1, va_2, va_3)$ zum Cluster. Ein Pixel hat im entsprechenden Farbraum eine Position, welche durch die Farbwerte va_1 , va_2 und va_3 beschrieben wird. Durch Auffassung eines Pixels $px \in I_{wood}$ als Ortsvektor im Farbraum ergibt sich die Metrik:

$$p_{gmm} = \sum_{i=1}^k \pi_k * \exp\left(-\frac{1}{2} * (\mu_i - px)^T \Sigma^{-1} (\mu_i - px)\right) \quad (3.24)$$

Der Faktor π_i ist eine Gewichtung pro Gaussian Modell und summiert die Mixture der Modelle bis zu einem Wert von 1 auf. Das heißt, dass p_{gmm} einen Wertebereich von $[0, 1]$ besitzt.

Die Berechnung und folglich das *Clustering* der **GMMs** ist komplexer als das *Clustering* mit *K-Mean*. Zur Berechnung soll der bewährte *Expectation Maximization (EM)*-Algorithmus verwendet werden (siehe [HTF01], [Pre+07]). Die Methode wird im Weiteren mit **GMM-EM** abgekürzt.

Dichteabschätzung im Farbraum (KD-NN)

Die selbst entwickelte Methode basiert auf einer Dichteabschätzung im Farbraum durch Nutzung einer Kugelumgebung. Die Methode nutzt einen KD-Baum (siehe [Ben75]) zur schnellen Suche der Nächsten Nachbarschaft und soll daher **KD-NN** (KD-Baum mit Nächster Nachbarschaft) genannt werden. Die Datenstruktur DS ist folglich ein KD-Baum und die Metrik m_{kn} errechnet aus den Ergebnissen der Suchanfrage im KD-Baum eine Wahrscheinlichkeit, womit sich das Modell Θ_{kn} ergibt:

$$\Theta_{kn} = (m_{kn}, \text{KD-Baum}, I_{kn}^p, (r, w_m)) \quad (3.25)$$

Das Modell erzeugt das Wahrscheinlichkeitsbild I_{kn}^p durch die Metrik m_{kn} , welche auf Basis einer Dichteabschätzung im KD-Baum $p \in I_{kn}^p$ ermittelt wird. Die Parameter zum Modell Θ_{kn} sind:

- r - Radius zur Bestimmung der Dichte
- w_m - korrespondierendes Gewicht zur durchschnittlichen Dichte

Die entwickelte Methode ist vom *Photon Mapping*, einem Verfahren zur fotorealistischen Bildgenerierung (siehe [Jen01]), abgeleitet. Beim *Photon Mapping* wird innerhalb einer Kugelumgebung die Dichte anhand einer *photon map* abgeschätzt. Ähnlich wird die Dichte auch bei der **KD-NN** Methode abgeschätzt. Der Unterschied besteht darin, dass keine Photonendichte, sondern die Pixeldichte abgeschätzt und eine Wahrscheinlichkeit abgeleitet wird. Die Trainingspixel FG^{smp} oder BG^{smp} werden zuerst in einen KD-Baum eingelagert. Ein Knoten des KD-Baumes enthält die Werte des Pixels $px = (va_1, va_2, va_3)$ im entsprechenden Farbraum und die für einen KD-Baum typische Spaltdimension (siehe [Ben75]). Zur Reduzierung der Daten im KD-Baum, werden in der **KD-NN**-Methode

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

keine Pixel doppelt gespeichert. Stattdessen wird ein zusätzlicher Zähler im Knoten hochgezählt. Nach dem Befüllen des KD-Baumes wird er balanciert, um eine schnelle Suche zu ermöglichen. Das Ausbalancieren des KD-Baumes ist eine $O(n * \log n)$ -Operation.

Nach der Erstellung des KD-Baumes wird zuerst eine durchschnittliche Dichte ρ_{avg} bestimmt. Zur Bestimmung wird die Anzahl der Pixel p_{all} im KD-Baum durch das Volumen des dreidimensionalen Farbraumes dividiert. Die Werte eines Pixels $px = (va_1, va_2, va_3)$ liegen i.d.R. im Intervall $va_1, va_2, va_3 \in [0, 255]$. Es ergibt sich die Formel:

$$\rho_{avg} = \frac{p_{all}}{255^3} \quad (3.26)$$

Im nächsten Schritt wird die Dichte innerhalb einer Kugelumgebung mit den Suchergebnissen im KD-Baum ermittelt. Die Suche der Nächsten Nachbarschaft in einem KD-Baum hat eine Komplexität von $O(\log n)$. Eine Suchanfrage besteht aus einem Pixel $px_s \in I_{wood}$ und einem Suchradius r . Das Ergebnis der Suchanfrage ist die Anzahl der Pixel n_{px} innerhalb der Kugel mit dem Radius r an Position px_s . Das Volumen der Kugel ist:

$$V_k = \frac{3}{4} * \pi * r^3 \quad (3.27)$$

Aus dem Kugelvolumen V_k und der gefundenen Pixelanzahl n_{px} ergibt sich die Dichte ρ_k innerhalb der Kugel:

$$\rho_k = \frac{n_{px}}{V_k} \quad (3.28)$$

In Abbildung 3.19 ist die Kugelumgebung und Abschätzung der Dichte ρ_k am Beispiel des RGB-Farbraumes verdeutlicht. Die durchschnittliche Dichte ρ_{avg} im Farbraum ist

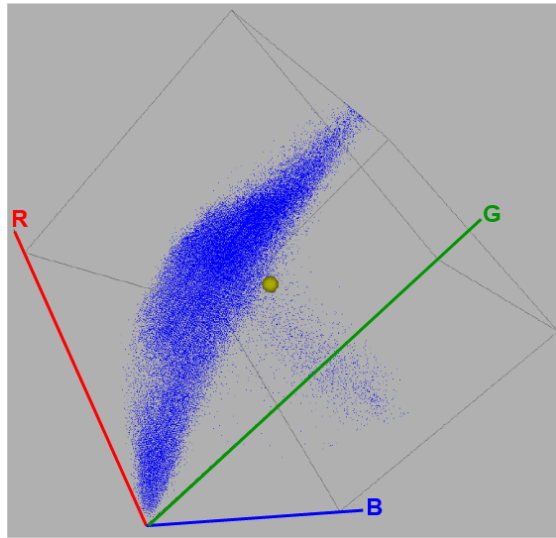


Abbildung 3.19: Die Stichprobe FG^{smp} im RGB-Farbraum als blaue Punktewolke und der in Gelb dargestellten Kugelumgebung zur Dichteabschätzung.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

von Bild zu Bild verschieden. Es muss folglich die Dichte des Farbraumes ρ_{avg} und die Dichte in der Kugelumgebung ρ_k in Beziehung zu einander gesetzt werden, um eine relative Wahrscheinlichkeit zu ermitteln. Die Dichte ρ_{avg} ist die durchschnittliche Dichte der Kugel in allen möglichen Positionen. Die Idee ist, die Dichte ρ_{avg} auf eine definierte Wahrscheinlichkeit p_m abzubilden. Das bedeutet, wenn die durchschnittliche Dichte ρ_{avg} gefunden wurde, so ist die ermittelte Wahrscheinlichkeit p_{kn} gleich p_m . Weiterhin muss die Wahrscheinlichkeit p_{kn} größer werden, wenn die Dichte ρ_k größer wird, und umgekehrt. Zur Erfüllung der Bedingungen und Berechnung des Skalierungsfaktors s dient die folgende Formel:

$$w_m = 1 - e^{-\rho_{all} * s} \quad (3.29)$$

$$s = \frac{\ln(1 - w_m)}{-\rho_{all}} \quad (3.30)$$

Die Metrik m_{kn} ergibt sich abschließend aus dem Skalierungsfaktor s und der Dichte in der Kugelumgebung ρ_k , wie folgt:

$$p_{kn} = 1 - e^{-\rho_k * s} \quad (3.31)$$

Setzen der Randgewichte und Graph-Cut-Segmentierung

Mit den beschriebenen vier Modellen ($\Theta_{his}, \Theta_{km}, \Theta_{gmm}, \Theta_{kn}$) ist es möglich, auf verschiedene Art und Weise die Wahrscheinlichkeitsbilder (I_{fg}^p, I_{bg}^p) anhand der Stichproben (FG^{smp}, BG^{smp}) zu ermitteln. Mit I_{fg}^p und I_{bg}^p werden die Regionengewichte w_r des Graphen G direkt gesetzt. Für einen optimalen Schnitt des Graphen G ist es zusätzlich notwendig, die Randgewichte (*boundary weights*) w_b adäquat zu setzen.

Die Randgewichte spiegeln die Gradienten des Bildes I_{wood} wieder und müssen in einem Bezug zu den Regionengewichten stehen. Das heißt, der Wertebereich ($w_b \in [0, 1]$) muss identisch und das Verhalten der Wahrscheinlichkeitsfunktion ähnlich sein. Daher bietet sich die Nutzung der e-Funktion an. Ein Gradient ist der Unterschied von einem Pixel px_i zum Nachbarpixel px_j im Farbraum und ergibt sich über die entsprechende Distanz. Es ist **i.d.R.** die euklidische oder Manhattan-Distanz üblich. In diesem Kontext ist die Manhattan-Distanz angebracht, da diese schneller und der Genauigkeitsverlust vernachlässigbar ist. In empirischen Experimenten wurde keine wesentliche Verbesserung durch die euklidische Distanz festgestellt. Zur Berechnung der Randgewichte ergibt sich die Formel:

$$w_b = 1 - e^{-\alpha * \|px_i - px_j\|_1} \quad (3.32)$$

Die Indizes i und j kennzeichnen benachbarte Knoten V des Graphen G und korrespondieren zu benachbarten Pixeln im Bildraum. Der freie Parameter α ist zum Justieren des Einflusses der Gradienten auf die Segmentierung. Je kleiner α , um so mehr Gewicht bekommen die Gradienten in der Segmentierung.

Mit den Randgewichten w_b und den Wahrscheinlichkeitsbildern I_{fg}^p und I_{bg}^p werden die Gewichte des Graphen $G = (V, E)$ gesetzt. Nach dem Setzen der Gewichte und folglich der Erzeugung eines gewichteten Graphen G , wird G durch den Graph-Cut-Algorithmus,

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

beschrieben in [BK04], geschnitten. Der Schnitt (*min-cut*) führt zu einer Partitionierung (Cliques) des Graphen. Die Knoten $v \in V$ einer Clique bilden im Bildraum ein Segment. Im Ergebnis entsteht ein binär segmentiertes Bild von Schnittholz I_h . In Abbildung 3.20 sind die Gewichte von G visualisiert und das Segmentierungsergebnis I_h abgebildet. In Abbildung 3.15 ist das zugehörige Inputbild I_{wood} zu finden.

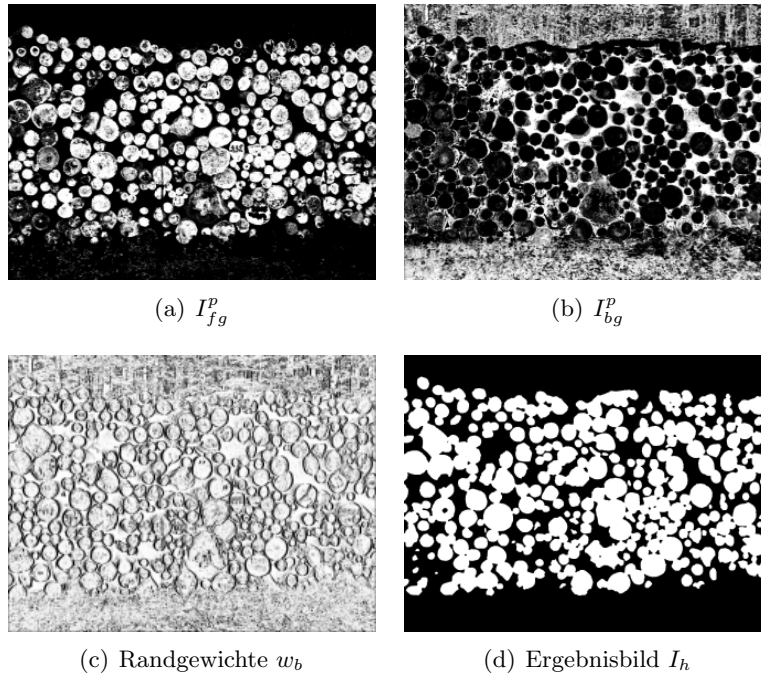


Abbildung 3.20: Von links nach rechts sind die Wahrscheinlichkeitsbilder, die Randgewichte und das Ergebnisbild des Graph-Cuts zu sehen. Verwendet wurde der RGB Farbraum, $\alpha = 0,005$ und Θ_{kn} mit den Parametern $r = 2,5$ und $w_m = 0,5$.

3.2.4 Zusammenfassung und Ergebnisse

In diesem Unterkapitel 3.2 wurde eine Methodik zur Segmentierung von Schnittholz erarbeitet. Die Methodik basiert auf Graph-Cut und einer Vorsegmentierung anhand der Holzfarbe. Die Segmentierung benötigt Eingabebilder I_{wood} , welche unter gewissen Restriktionen aufgenommen wurden. Das heißt, im oberen und unteren Bildbereich muss Hintergrund und im mittleren Bildbereich Stammschnittflächen zu sehen sein. Die besagten Restriktionen sind in der Praxis einfach einzuhalten. Die erarbeitete Methodik soll im Weiteren **WS-GC-R** (*Cut Wood Segmentation by Graph-Cut with Restrictions*) genannt werden. Die Methodik besteht aus den vier Schritten:

1. Vorsegmentierung zur Ermittlung von Trainingsdaten

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

2. Vorder- und Hintergrundmodellerstellung
3. Erstellung eines gewichteten Graphen
4. Graph-Cut-Segmentierung

Für den zweiten und dritten Schritt wurde ein allgemeines Modell Θ eingeführt und die vier speziellen Modelle Θ_{his} , Θ_{km} , Θ_{gmm} und Θ_{kn} erarbeitet. Die Modelle sind in der Arbeitsweise verschieden und haben verschiedene Parameter:

- $\Theta = (m, DS, I^p, PA)$ - **allgemeines Modell**
 - m ist eine Metrik und DS eine Datenstruktur
 - PA ist ein n -Tupel von Parametern
- $\Theta_{his} = (m_{his}, \text{3D-Histogramm}, I_{his}^p, (h_s, \beta))$ - **3D-Histogramm**
 - h_s ist ein Skalierungsfaktor der Binanzahl
 - β ist ein Faktor zum Justieren der Wahrscheinlichkeit
- $\Theta_{km} = (m_{km}, KM_k, I_{km}^p, (k, \gamma))$ - **K-Mean**
 - k ist die Anzahl der Cluster (Mittelwerte)
 - γ ist ein Faktor zum Justieren der Wahrscheinlichkeit
- $\Theta_{gmm} = (m_{gmm}, GMM, I_{gmm}^p, (k))$ - **GMM-EM**
 - k ist die Anzahl der Cluster (Gaussian Modelle)
- $\Theta_{kn} = (m_{kn}, \text{KD-Baum}, I_{kn}^p, (r, w_m))$ - **KD-NN**
 - r ist der Radius zur Bestimmung der Dichte
 - w_m ist das korrespondierende Gewicht zur durchschnittlichen Dichte

Die vier unterschiedlichen Modelle wurden anhand von 71 Evaluierungsbildern und deren Grundwahrheit evaluiert. Genaueres zu den Evaluierungs- und GT-Bildern ist im Anhang C.2 zu lesen. Zur Evaluierung wurden die im Bereich der Segmentierung üblichen Maße, *precision*, *recall* und *f-score* (siehe [Pow07] und Anhang A.4) verwendet. *Precision* ist der positive Vorhersagewert (*positive predictive value*), *recall* die Richtig-Negativ-Rate und *f-score* das harmonische Mittel von *precision* und *recall*. Alle drei Maße haben einen Wertebereich von $[0, 1]$. Ein Wert von 1 kennzeichnet eine perfekte Segmentierung. Die Ergebnisse der Evaluierung sind in Tabelle 3.2 zu sehen. Pro Modell Θ und dessen Parametern PA wurde der Mittelwert von den 71 Ergebnissen berechnet. Zusätzlich wurde die Standardabweichung ermittelt. Je geringer die Standardabweichung, umso geringer weichen die Maße voneinander ab. Das bedeutet, dass die Segmentierung robuster ist und wenig bis keine Ausreißer enthalten sind. Ausreißer entstehen zum Beispiel, wenn der Hintergrund anstatt des Schnittholzes segmentiert wird.

3.2 Schnittholzsegmentierung durch Graph-Cut und Holzfarbe

Tabelle 3.2: *precision*, *recall* und *f-score* der Schnittholzsegmentierung mit verschiedenen Modellen Θ und Parametern PA bzgl. 71 **GT**-Bildern. Es wurde RGB und $\alpha = 0,0005$ verwendet. Die besten Werte sind fettgedruckt.

Methode		Mittelwert			Standardabweichung		
Modell	PA	prec.	recall	f-score	prec.	recall	f-score
Θ_{his}	(0.0625, 1000)	0,7574	0,6386	0,574	0,2194	0,3916	0,308
Θ_{his}	(0.125, 1000)	0,8110	0,8121	0,7512	0,1715	0,2776	0,2333
Θ_{his}	(0.25, 1000)	0,8297	0,9116	0,8491	0,1406	0,1469	0,1289
Θ_{his}	(0.5, 1000)	0,8341	0,9298	0,8649	0,1393	0,1069	0,1114
Θ_{his}	(1.0, 1000)	0,8336	0,9281	0,8682	0,1388	0,0617	0,0858
Θ_{gmm}	(2)	0,8803	0,8959	0,8800	0,0955	0,0947	0,0610
Θ_{gmm}	(4)	0,8718	0,9022	0,8790	0,0921	0,0984	0,0654
Θ_{gmm}	(8)	0,8652	0,9087	0,8795	0,0969	0,0830	0,0580
Θ_{gmm}	(12)	0,8631	0,9074	0,8770	0,1007	0,0885	0,0639
Θ_{gmm}	(16)	0,8601	0,9103	0,8769	0,0996	0,0870	0,0626
Θ_{km}	(8, 0.02)	0,8865	0,9042	0,8898	0,0785	0,0945	0,0648
Θ_{kn}	(2.5, 0.5)	0,872	0,9406	0,8981	0,0886	0,0473	0,0518

In der Visualisierung von Hintergrundfarbe und Holzfarbe zeigten sich hauptsächlich im RGB-Farbraum sichtbare Unterschiede. Aus diesem Grund, und weil andere Farbräume in Experimenten keine guten Ergebnisse erzielten, wurde in der Evaluierung der RGB-Farbraum verwendet. Die Wahl der Parameter PA hat verschiedene Gründe. Im Modell Θ_{km} wurden k und γ , in Θ_{his} wurde β und in Θ_{kn} wurde r experimentell ermittelt. Bei Histogrammen hat die Anzahl der Bins **i.d.R.** einen entscheidenden Einfluss auf die Genauigkeit, weshalb in Θ_{his} die Anzahl der Bins durch h_s variiert wurde. Im Modell Θ_{gmm} ist hingegen die Anzahl der Cluster k der entscheidende Einflussfaktor. Zu wenige Cluster modellieren den Farbraum ungenau und zu viele führen zu einer Übermodellierung. Eine Anzahl von 2 bis 16 ist in der **CV** üblich. Das mittlere Gewicht (w_m) in Θ_{kn} wurde auf 0,5 gesetzt, da 0,5 der mittlere Wert des Wertebereiches $[0, 1]$ eines Gewichtes w ist.

Wie in Tabelle 3.2 anhand vom *f-score* zu sehen ist, sind die Ergebnisse mit dem **KD-NN**-Ansatz (Θ_{kn}) am besten. Der *f-score* ist das harmonische Mittel von *precision* und *recall* und folglich das aussagekräftigste Maß. Geringfügig schlechter ist *K-Mean* (Θ_{km}) und **GMM-EM** (Θ_{gmm}). Das 3D-Histogramm (Θ_{his}) ist die Methode mit den schlechtesten Ergebnissen. Bei Betrachtung der Parameter zeigt sich, dass **GMM-EM** nur einen Parameter hat, wohingegen alle anderen Methoden zwei Parameter aufweisen. Trotzdem führen gut gewählte Parameter zu einem stabilen Ergebnis, wie die geringe Standardabweichung bei **KD-NN** und auch *K-Mean* zeigen. Zusammenfassend erweist sich die **KD-NN**-Methode als geeignete Methode zur Segmentierung von Schnittholz. Die **KD-NN**-Methode, sowie das beschriebene Konzept **WS-GC-R**, wurden in [Gut+10] und [Gut+11a] veröffentlicht. Ergebnisbilder der Evaluierung der verschiedenen Modelle sind im Anhang B.3 zu finden.

3.3 Stammschnittflächensegmentierung mit Objektdetektion

In diesem Unterkapitel werden eine Segmentierung der Stammschnittflächen f_s^{st} durch Adaption einer Objektdetektion konzipiert und die Ergebnisse evaluiert. Im Abschnitt 3.3.1 wird zuerst das Konzept der Segmentierung erarbeitet. Das Konzept untergliedert sich grob in Detektion der Stammschnittflächen, Verifikation der Ergebnisse der Detektion und Segmentierung der Stammschnittflächen. Im Abschnitt 3.3.2 werden die Methoden der Objektdetektion untersucht, eine geeignete Methode ausgewählt und die Detektion spezifiziert. Im Abschnitt 3.3.3 wird die Verifikation der detektierten Bildobjekte beschrieben und im letzten Abschnitt 3.3.4 werden die Ergebnisse ausgewertet und die wesentlichen Erkenntnisse zusammengefasst.

3.3.1 Konzeption der Segmentierung mit Objektdetektion

Die Methoden der Objektdetektion detektieren Objekte mit Hilfe eines Objektmodells. Das Modell muss **i.d.R.** mit Trainingsbildern I^t angelehrt werden. Ist das Modell erstellt, wird im Bild oder in einer Transformation des Bildes das Objekt gesucht. Die meisten Methoden suchen fensterbasiert und klassifizieren ein Fenster in Objekt oder Nicht-Objekt. Ein Fenster (*window*) soll als das 4-Tupel win definiert werden mit:

$$win = (x, y, w, h); \quad x, y \in \mathbb{N}^{\geq 0}; \quad w, h \in \mathbb{N}^{> 0} \quad (3.33)$$

Hierbei sind x und y sind die Koordinaten der linken oberen Ecke und w und h die Höhe und Weite des Fensters. Die fensterbasierten Methoden [LBH08] sind gut erforscht und haben sich in der Praxis bewährt, wie bspw. Haarkaskaden [VJ01] und Eigenfaces [TP91] zur Gesichtsdetektion. Daher wird hier auf die fensterbasierte Methodenklasse eingeschränkt. Eine zusätzliche Betrachtung anderer Methodenklassen ist nicht mit einem einzelmem Konzept abzudecken, wie bspw. die Nutzung von lokalen Bildmerkmalen (z.B. SURF [BTG06] oder SIFT [Low99]). Lokale Bildmerkmale und deren Komposition zur finalen Objektdetektion bedingen eine andere Strategie der Suche und Auswertung.

Das Ergebnis der fensterbasierten Objektdetektion ist eine Menge W von Fenstern win . Ein Fenster markiert ein rechteckiges Segment S^{win} , und alle Fenster zusammen bilden eine Menge von Segmenten WS (*window segments*). Die Mengen W und WS sollen wie folgt definiert werden:

$$W = \{win | \text{detektiertes Fenster im Bild}\}$$

$$WS = \{S | \forall p_{x,y} \in S \exists (x_w, y_w, w, h) \in W : x \in [x_w, x_w + w) \wedge y \in [y_w, y_w + h)\}$$

Allerdings ist eine Stammschnittfläche (O^{st}) **i.d.R.** nicht rechteckig ($O^{st} \not\equiv WS$). Aus diesem Grund bietet es sich an, statt Fenster Kreisparameter cp (*circle parameters*) als Tripel von Parametern zu betrachten mit:

$$cp = (x, y, r); \quad x, y \in \mathbb{N}^{\geq 0}; \quad r \in \mathbb{R}^{> 0} \quad (3.34)$$

Wobei x und y sind die Koordinaten des Mittelpunktes und r der Radius des Kreises sind. Die Menge der Kreisparameter CP und korrespondierenden Segmentmenge CPS

3.3 Stammschnittflächensegmentierung mit Objektdetektion

lassen sich wie folgt durch die detektierten Fenster W ermitteln:

$$CP = \left\{ \left(x + \frac{w}{2}, y + \frac{h}{2}, \frac{w+h}{4} \right) \mid (x, y, w, h) \in W \right\} \quad (3.35)$$

$$CPS = \{ S \mid \forall p_{x,y} \in S \exists (x_c, y_c, r) \in CP : (x_c - x)^2 + (y_c - y)^2 \leq r^2 \}$$

Ein Kreis ist die ideale Form einer Stammschnittfläche und eine grobe Approximation der tatsächlichen Form. Theoretisch ist mit Ellipsen eine genauere Approximation möglich. Ellipsen sollen aber nicht verwendet werden, da sie nicht rotationsinvariant sind. Da ein Kreis eine Approximation der tatsächlichen Form einer Stammschnittfläche O^{st} ist, muss eine Segmentierung von O^{st} nach der Detektion erfolgen. In der Vorbetrachtung in Abschnitt 2.2 wurde gezeigt, dass die Objektdetektion einen größeren Lösungsraum hat als die Segmentierung, da Überlagerungen von detektierten Objekten möglich sind. Überlagerungen können im Bildraum bei Stammschnittflächen nicht vorkommen. Ein mehrfach detektiertes Pixel kann nur zu genau einer Stammschnittfläche gehören. Überlagerungen müssen folglich aufgelöst werden. Weiterhin sind falsch-positiv (*false positive*) und falsch negativ (*false negativ*) detektierte Stammschnittflächen nicht ausgeschlossen. Eine perfekte Detektion in Praxisbildern und zudem mit einer noch nicht erforschten Objektklasse, den Stammschnittflächen, ist illusorisch und kann nicht vorausgesetzt werden. Demnach ist eine Verifikation der Menge von Kreisparametern und Segmenten (CP, CPS) anhand von Vorwissen sinnvoll. In einem Holzpolter liegen Baumstämme $O^{st,r}$ aufeinander, was zu einer Gruppierung von O^{st} im Bildraum führt. Das Vorwissen, das die Stammschnittflächen eine Gruppe bilden, soll zur Verifikation genutzt werden.

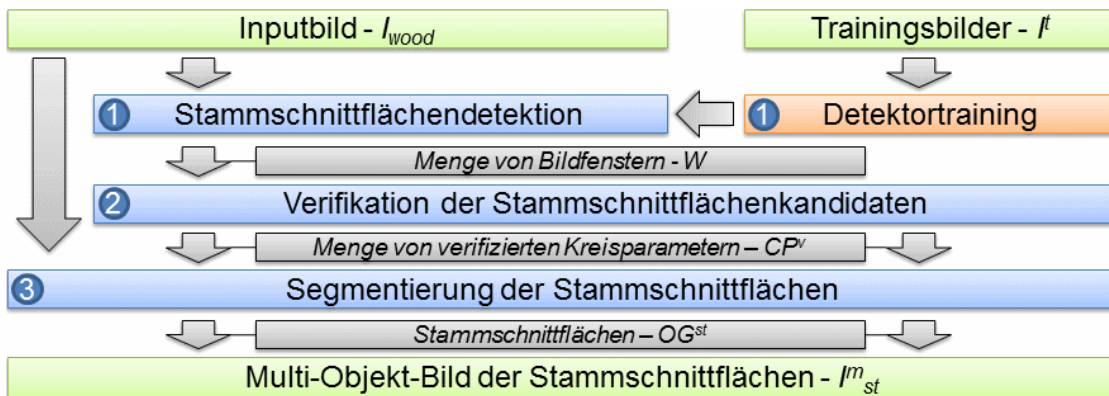


Abbildung 3.21: Grobes Konzept zur Segmentierung von Stammschnittflächen durch Objektdetektion.

Aus den Erläuterungen ergibt sich das grobe Konzept in Abbildung 3.21. Das Konzept untergliedert sich in die drei wesentlichen Schritte Objektdetektion, Verifikation der Detektionsergebnisse und Segmentierung. Im ersten Schritt sollen potenzielle Stammschnittflächenfenster W mit Methoden der Objektdetektion detektiert werden.

Die Fenster W sollen im zweiten Schritt verifiziert und zu einer Menge von verifizierten Kreisparametern CP^v aufbereitet werden. Anhand von CP^v sollen im letzten Schritt die Stammschnittflächen OG^{st} segmentiert werden.

Aus dem groben Konzept ergeben sich die folgenden **Fragestellungen und Forschungsinhalte**:

1. Welche Methode der fensterbasierten Objektdetektion ist zur Detektion der Stammschnittflächen geeignet?
 - Wie müssen die Trainingsdaten zur Erstellung eines guten Detektors ausgeprägt sein?
 - Welche Such- und Klassifikationsstrategie soll verwendet werden?
2. Wie lassen sich die Ergebnisse der Detektion nutzen, um die Stammschnittflächen zu segmentieren?
 - Lassen sich die Ergebnisse verifizieren und ggf. verbessern?
 - Welche Basismethoden der Segmentierung sind geeignet?

Nachfolgend werden die Fragestellungen in der aufgeführten Reihenfolge in den Abschnitten 3.3.2 und 3.3.3 betrachtet und das grobe Konzept verfeinert.

3.3.2 Wahl des Objektdetektors und Detektion

Etablierte Methoden der fensterbasierten Objektdetektion sind Template Matching, Eigenobjekte, HoG und Haarkaskaden. Die genannten etablierten Methoden sollen hinsichtlich der Eignung zur Detektion von Stammschnittflächen etwas genauer untersucht werden. Zur Einschränkung des Lösungsraumes werden andere, weniger etablierte Methoden nicht weiter betrachtet. Die genannten Methoden sind nicht rotationsinvariant und nicht skalierungsinvariant. Die Rotationsinvarianz ist zur Detektion der Stammschnittflächen OG^{st} nicht erforderlich, da die Charakteristiken von OG^{st} rotationsunabhängig sind. An einer idealen Stammschnittfläche ist die runde Form und die Rotationssymmetrie charakteristisch, wodurch die Merkmale um den Mittelpunkt von O^{st} rotieren. Hingegen ist O^{st} im Bild I_{wood} in der Größe verschieden, womit eine Skalierungsinvarianz zur Detektion erforderlich ist. Bei jeder Methode wird weiterhin ein Modell vom Objekt Θ^O benötigt, welches auch Detektor genannt wird. Das Modell wird i.d.R. mit Trainingsdaten I^t erstellt bzw. angelernt.

Das **Template Matching** ist i.d.R. gut geeignet zur Detektion von statischen Objekten, die sich kaum im Bildraum verändern, wie bspw. ein Firmen-Logo. Das Modell Θ^O ist ein sogenanntes Template. Zur Detektion von O^{st} ist mindestens ein Template sowie ein geeignetes Abstandsmaß vom Template zu S^{win} erforderlich. In der Problemdiskussion wurde gezeigt, dass O^{st} in Form, Farbe und Textur variiert. Ein Template wird folglich die Varianz der Objekte in OG^{st} kaum abdecken. Die Anwendung mehrerer Templates ist denkbar, aber die Laufzeit und die Anfälligkeit für falsch detektierte Objekte wird größer. In Abbildung 3.22 ist ein Ergebnis von Template Matching mit einer fixen Template-Größe ohne Skalierung zu sehen.

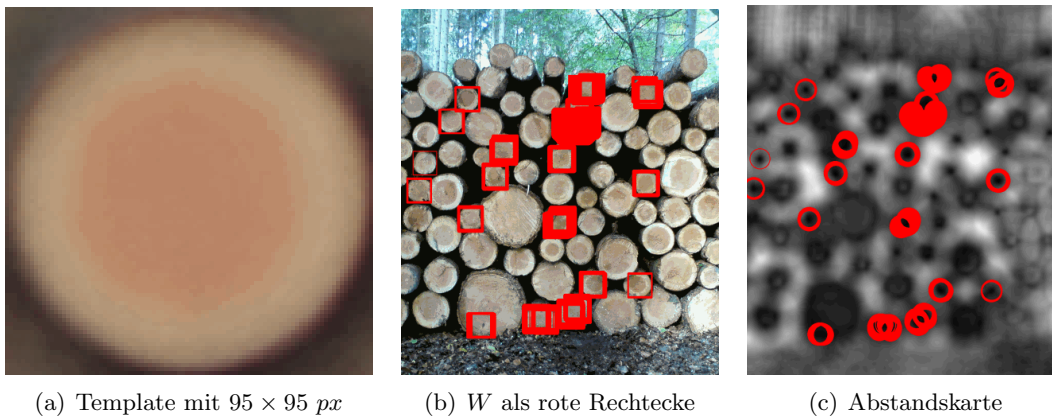


Abbildung 3.22: Ergebnisse mit Template Matching und der Summe der quadratischen Differenzen, als Abstandsmaß. Links ist das Template, in der Mitte W und rechts die Abstandskarte, in der rote Kreise den Abstand unterhalb eines Schwellwertes kennzeichnen.

Eigenobjekte, speziell *Eigenfaces*, sind bekannt aus der Gesichtserkennung und Detektion. Primär liegt die Anwendung in der Gesichtserkennung (siehe [SK87], [TP91], [PMS94]) auf Basis von bereits detektierten Objekten. Die Grundidee von Eigenobjekten ist, aus Trainingsdaten I^t ein Modell Θ^O zu berechnen, welches die prinzipiellen Komponenten von allen I^t enthält. Alle Trainingsbilder müssen in der Bildgröße $w \times h$ identisch sein. Die Verkettung der Zeilen vom i 'ten Trainingsbild I_i^t ergibt einen Bildvektor b_i . Aus dem Mittelwert μ der Bildvektoren und b_i ergibt sich die Kovarianzmatrix Σ . Aus Σ lassen sich die Eigenvektoren v_i und Eigenwerte λ_i ermitteln:

$$\Sigma = \frac{1}{n} \sum_{i=1}^n (\mu - b_i)(\mu - b_i)^T; \Sigma v_i = \lambda_i v_i \quad (3.36)$$

Die Bestimmung von v und λ anhand einer $(w * h)^2$ Matrix ist nicht trivial (siehe [TP91]). Die k Eigenvektoren v mit den größten Eigenwerten λ enthalten die meisten Informationen und bilden zusammen einen kleineren Raum mit weniger Dimensionen, den Eigenraum. Jeder von den k Eigenvektoren ist analog ein Eigenobjekt. Das Modell Θ^O besteht folglich aus k Eigenobjekten, speziell k **Eigenlogs**. Zur Detektion der Stammschnittflächen muss jedes S^{win} in den Eigenraum transformiert und der Abstand zu den Eigenlogs bestimmt werden. In Abbildung 3.23 sind einige experimentelle Ergebnisse zu sehen, wobei das Bild in diskreten Stufen skaliert wurde, um verschiedene Größen abzudecken.

Das **Histogramm von orientierten Gradienten (HoG)** findet zumeist Anwendung in der Detektion von Personen, was in [DT05] und [SAT11] zu finden ist. Ein Bild wird in Blöcke und jeder Block weiter in Zellen zerlegt. Pro Zelle wird ein Deskriptor, basierend auf der Gradientenverteilung, in der Zelle errechnet. Bei der eigentlichen Objektdetektion wird ein Teilbild S^{win} (Block) wieder in Zellen unterteilt und anhand der Deskriptoren klassifiziert. Das Modell Θ^O ist ein Klassifikator, wie **SVM** oder **NN**,

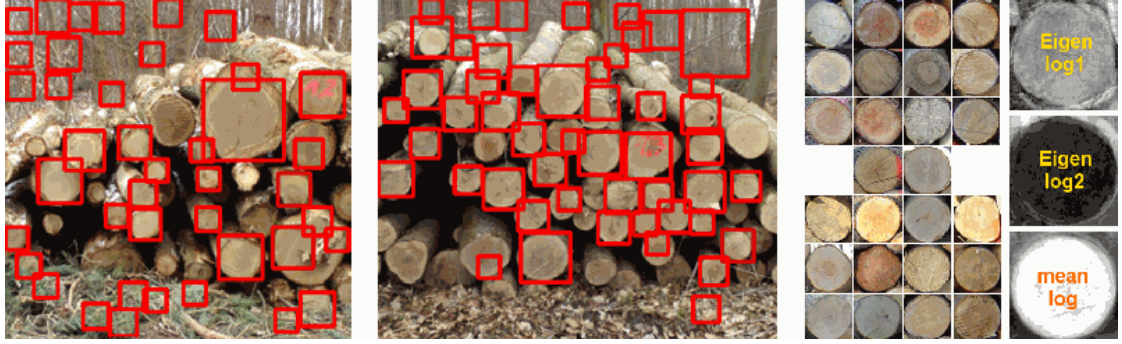


Abbildung 3.23: Links und in der Mitte sind die Fenster W der Detektion durch Eigenlogs als rote Rechtecke abgebildet. Rechts sind alle I^t , die Eigenlogs mit den größten Eigenwerten und μ markiert mit *meanlog* zu sehen.

womit **HoG's** eine Ausnahme der fensterbasierten Methoden bilden. Zur Adaption von **HoGs** an Stammschnittflächen ist die Auswahl eines geeigneten Klassifikators und dessen Training anhand von I^t notwendig. In der Problemdiskussion im Abschnitt 3.1 wurde gezeigt, dass sehr viele mehrdeutige Gradienten im Bild I_{wood} sind. Fehldetektionen sind folglich nicht ausgeschlossen. Auch wurde in Abschnitt 3.2.2 gezeigt, dass sich Schnittholz farblich gut abgrenzt. **HoGs** hingegen beziehen keine Farbmerkmale in die Detektion mit ein, sondern nur Gradienten. Aufgrund der vielen Nachteile wird in dieser Arbeit der Fokus auf andere Methoden der Objektdetektion gelegt und **HoG** ausgeschlossen werden.

Die **Haarkaskaden**, entwickelt von [VJ01] zur Detektion von Gesichtern, basieren auf einer Kaskade von schwachen Klassifikatoren, den Haar-Templates. Die Kaskade mit den Haar-Templates bilden das Modell Θ^O . Die Haar-Templates sind sehr einfach aufgebaut und zerlegen ein Teilbild S_0^{win} in n weitere disjunkte rechteckige Segmente S_i^{win} , wobei gilt

$$S_0^{win} = \bigcup_{i=1}^n S_i^{win} \quad (3.37)$$

Die Anzahl der Segmente n ist i.d.R. klein (< 5). Sei $s(S_i^{win})$ die Summe aller Werte der Pixel im Segment S_i^{win} , so ergibt sich das Merkmal eines Haar-Templates mit:

$$d = s(S_i^{win}) - s(S_j^{win}); \quad i, j \in [1, n]; \quad i \neq j \quad (3.38)$$

Die Einfachheit der Merkmale macht es möglich sehr schnell Objekte zu detektieren. Die Basis bildet ein Integralbild, in welchem ein Pixel die Summe der Vorgängerpixel und sich selbst enthält. Mit lediglich vier Operationen wird die Summe aller Pixel in einem Rechteck im Integralbild bestimmt. Durch die Differenzen d , die unterschiedlichen Zerlegungen in Rechtecke und die Kaskadierung, ergibt sich ein vielseitiger Detektor. Der Detektor verschlüsselt Farbunterschiede, Texturen und indirekt auch Gradienten, wodurch die Eignung zur Detektion von OG^{st} positiv zu bewerten ist. Zur Erstellung von Θ^O sind positive und negative Trainingsbilder I^t erforderlich. Anhand von I^t müssen die

3.3 Stammschnittflächensegmentierung mit Objektdetektion

einzelnen Haar-Templates erstellt werden. Die Erstellung von Θ^O ist bereits in openCV [Lag11] umgesetzt, ebenso eine Detektion mit verschiedenen Skalierungen. Einige erste Ergebnisse der Detektion mit einem zuvor trainierten Modell für Baumstämme sind in der Abbildung 3.24 zu sehen.

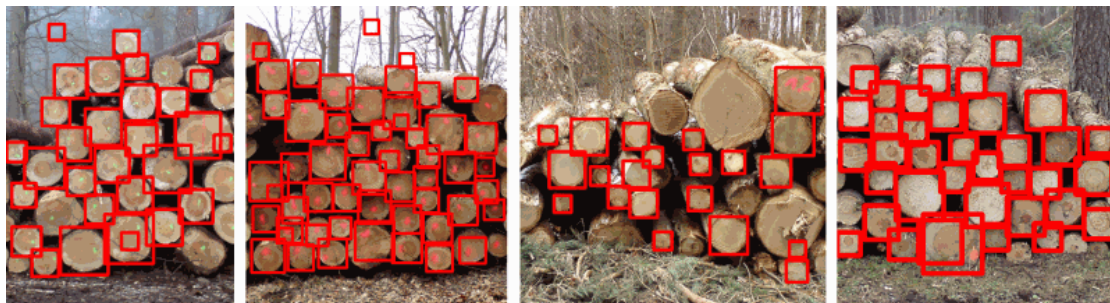


Abbildung 3.24: Ergebnisse mit Haarkaskaden mit openCV 2.4.7 umgesetzt, wobei rote Rechtecke die Fenster W markieren.

Aus den Beschreibungen und Experimenten der Methoden der fensterbasierten Objektdetektion ergeben sich verschiedene Vor- und Nachteile. Theoretisch lassen sich mit allen Methoden Stammschnittflächen detektieren. Allerdings mit unterschiedlicher Komplexität, Genauigkeit und Rechenzeit. Zur Auswahl einer Methode sollen einige Kriterien dienen. Die Kriterien und deren Bewertung sind in Tabelle 3.3 zu sehen, wobei Template Matching mit **TM**, Eigenobjekte mit **EO** und Haarkaskaden mit **HK** abgekürzt sind. Die Kriterien sind mit Zahlen von 0 bis 6 bewertet, wobei die Zahlen die Bedeutung schlecht (0) über durchschnittlich (3) bis sehr gut (6) widerspiegeln.

Tabelle 3.3: Eigene Bewertung der Methoden der fensterbasierten Objektdetektion zur Detektion von Stammschnittflächen.

Kriterium	Bemerkung	TM	EO	HoG	HK
Rechenzeit	je geringer, umso besser	0	1	3	6
Genauigkeit	basierend auf Experimenten	1	3	3	5
Umsetzung	je einfacher umsetzbar, umso besser	6	2	3	5
Parameter	je weniger Parameter, umso besser	2	2	2	6

Die Tabelle 3.3 zeigt, dass Haarkaskaden die Kriterien am besten erfüllen. Zur besseren Auswertung der Eignung wurden sieben verschiedene Haarkaskaden erstellt und mit 71 **GT**-Bildern evaluiert (siehe Anhang C.2). Bei der Evaluierung wurden die statistischen Werte *precision*, *recall* und *f-score* (siehe [Pow07] und Anhang A.4) der detektierten Segmente SW verwendet. Die Berechnung der Werte für eine Objektdetektion ist schwieriger als die Berechnung für eine binäre Segmentierung. Bei einer binären Segmentierung ist die Grundwahrheit eines Pixels eindeutig, wohingegen detektierte Objekte mehrdeutig sind. Zur Lösung des Problems wurde ein Flächenvergleich von OG^{st} zu SW anhand eines

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Schwellwertes umgesetzt. Zuerst werden die umschließenden rechteckigen Segmente $O^{st,w}$ korrespondierend zu OG^{st} berechnet. Alle rechteckigen Segmente der Grundwahrheit werden mit den rechteckigen Segmenten S^{win} in SW verglichen, wobei folgende Bedingung erfüllt sein muss:

$$1 - \frac{|O^{st,w} \cup S^{win}|}{2|O^{st,w} \cap S^{win}|} < 0,25 \quad (3.39)$$

In Worten ausgedrückt ist die Bedingung erfüllt, wenn eine relativ große gemeinsame Fläche überdeckt wird. Mit dieser Bedingung wurden die statischen Werte *precision*, *recall* und *f-score* für sieben verschiedene Modelle Θ^O ermittelt. Die Modelle wurden mit unterschiedlichen Trainingsdaten berechnet. Die Ergebnisse der Evaluierung sind in Tabelle 3.4 zu sehen, wobei die Spalte **K** die Anzahl der Kaskaden, die Spalte **pos.** die Anzahl positiver und **neg.** die Anzahl negativer Trainingsbilder kennzeichnet.

Tabelle 3.4: Der Mittelwert und die Standardabweichung von *precision*, *recall* und *f-score* der detektierten Fenster in W bzgl. der 71 **GT**-Bilder.

Modellerstellung				Mittelwert			Standardabweichung		
Nr.	K	pos.	neg.	prec.	recall	f-score	prec.	recall	f-score
Θ_1^O	9	99	106	0,9279	0,1177	0,2026	0,2158	0,0755	0,1165
Θ_2^O	9	1065	3019	0,9954	0,1773	0,2831	0,0173	0,1264	0,1710
Θ_3^O	12	825	3019	0,9851	0,1623	0,2625	0,1187	0,1205	0,1653
Θ_4^O	13	99	106	0,9070	0,0623	0,1126	0,2439	0,0537	0,0902
Θ_5^O	10	1065	3019	0,9910	0,3049	0,4513	0,0205	0,1283	0,1554
Θ_6^O	11	825	3019	0,9875	0,4152	0,5703	0,0265	0,1413	0,1467
Θ_7^O	19	5729	3019	0,8764	0,6529	0,7412	0,0868	0,1371	0,1143

Die Tabelle 3.4 zeigt, dass das Modell Θ_7^O der Haarkaskade die besten Ergebnisse erzielt. Der beste *f-score* liegt im Durchschnitt bei 0,7412, was ein akzeptables Ergebnis ist. Es ist zu erkennen, dass mit steigender Anzahl der Kaskaden und Trainingsbilder der *f-score* meist besser wird. Man kann allerdings nicht daraus schlussfolgern, dass mit mehr Trainingsbildern und Kaskaden ein noch besseres Ergebnis erzielt werden kann. Mit mehr Kaskaden und Trainingsbildern sinkt die *precision*, wie bei Θ_5^O , Θ_6^O und Θ_7^O zu sehen ist. Das Verhalten erklärt sich dadurch, dass die Varianz der Stammschnittflächen in Form, Farbe und Textur hoch ist. Defekte, nasse oder farblich markierte Baumstämme werden bei einem sehr genauen Detektor ggf. nicht mehr detektiert.

Haarkaskaden sind schnell, haben kaum Parameter, sind einfach mit openCV umsetzbar und erzielen gute Ergebnisse, was in Tabelle 3.4 und in Abbildung 3.24 zu sehen ist. Folglich wird der Fokus auf Haarkaskaden gesetzt und es soll von Haarkaskaden in den weiteren Betrachtungen ausgegangen werden. Haarkaskaden sowie die anderen Methoden der fensterbasierten Objektdetektion sind nicht skalierungs- und translationsinvariant. Zur Detektion der Segmente SW wurde deshalb ein Fenster über das Bild geschoben und das Fenster entsprechend skaliert.

3.3.3 Verifikation der Bildobjekte und Segmentierung

Im vorherigen Abschnitt hat sich ergeben, dass der beste *f-score* mit Haarkaskaden einen Wert von 0,7412 hat mit einer *precision* von 0,8764 und einem *recall* von 0,6529. Es befinden sich folglich einige falsch-positiv und einige falsch-negativ detektierte Fenster in W . Durch Anwendung von Vorwissen auf die Menge von Fenstern W ist es möglich die Anzahl der falsch detektierten (falsch-positiv) Stammschnittflächen O^{st} zu verringern. Nicht detektierte O^{st} (falsch-negativ) sind schwieriger zu ermitteln und können maximal abgeschätzt werden, was hier nicht weiter betrachtet wird. Die Stammschnittflächen OG^{st} in einem Bild bilden eine Gruppe von Bildobjekten, da die Baumstämme *i.d.R.* aufeinanderliegen. Wird die Menge WS als Cluster von S^{win} im Bildraum aufgefasst, so entsprechen theoretisch Ausreißer vom Cluster den falsch detektierten O^{st} . Weiterhin sind Überlappungen von O^{st} im Bildraum nicht möglich, da die Mengen in OG^{st} disjunkt ist. Die Disjunktheit basiert auf der Annahme, dass theoretisch ein Pixel nur zu einem O^{st} gehört.

Es ergeben sich folglich zwei Ansätze zur Beseitigung von falsch-positiv detektierten Stammschnittflächen, nämlich die Auflösung von Überlappungen und die Eliminierung von Ausreißern. In Abbildung 3.25 sind die Ansätze (Teilschritte) und deren Zusammenspiel zur Verifikation der Stammschnittflächen skizziert. Bei beiden Ansätzen ist es sinnvoll die

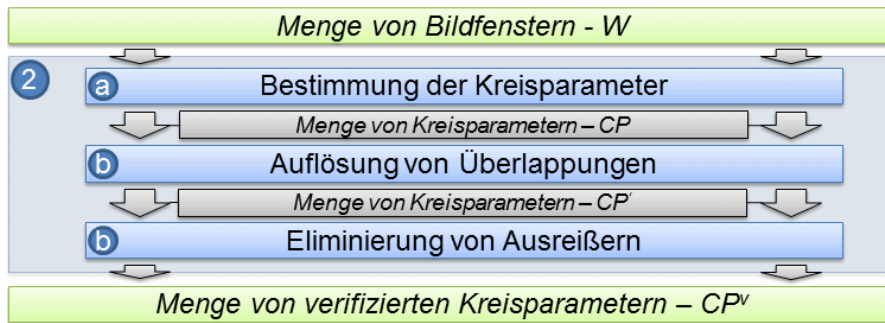


Abbildung 3.25: Teilschritte zur Verifikation der Stammschnittflächen.

Menge W in eine Menge von Kreisparametern CP zu überführen. Ein Kreis ist die ideale Form einer Stammschnittfläche und besser geeignet als ein Rechteck. In der Evaluierung hat sich gezeigt, dass im Fall von Überdeckungen ein größerer Kreis in CP *i.d.R.* exakter ist als ein kleinerer. Zur Bestimmung von Überdeckungen ist der euklidische Abstand der Kreismittelpunkte hilfreich mit:

$$d(cp_1, cp_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}; \quad cp_1, cp_2 \in C \quad (3.40)$$

Liegt ein kleinerer Kreismittelpunkt in einem größeren Kreis, so wird der kleinere Kreis entfernt. Dies soll mit folgender Bedingung o (*overlapping*) formalisiert werden:

$$o(cp_1, cp_2) = \begin{cases} \text{wahr,} & d(cp_1, cp_2) < r_1 - r_2 \wedge r_1 > r_2 \\ \text{falsch,} & \text{sonst} \end{cases} \quad (3.41)$$

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Bei geringen Überdeckungen ist die Bedingung nicht erfüllt. Geringe Überdeckungen sind möglich, da eine Stammschnittfläche selten ein idealer Kreis ist. Die Menge CP' bildet die Menge aller verbleibenden Kreise.

Zur Eliminierung von Ausreißern besteht die Möglichkeit, eine Gravitation g anzunehmen, die größer wird, umso größer ein Baumstamm O' ist. Der Idee liegt zu Grunde, dass Baumstämme mit großem Durchmesser einen größeren Abstand zu benachbarten Baumstämmen verursachen, als Baumstämme mit kleinem Durchmesser. Alle Kreise in CP , die sich gegenseitig anziehen sind valide und alle anderen sind Ausreißer. Zwei Kreise ziehen sich an, wenn sie sich mit skaliertem Radius überdecken und folgende Bedingung g (*gravity*) erfüllen:

$$g(cp_1, cp_2) = \begin{cases} \text{wahr,} & d(cp_1, cp_2) < r_1 * a + r_2 * a \\ \text{falsch,} & \text{sonst} \end{cases} \quad (3.42)$$

Ein geeigneter empirisch ermittelter Wert ist $a = 2$. Algorithmisch wird zuerst der zentral liegende Kreis cp_c in CP' bestimmt. Der zentrale Kreis cp_c ist der zu dem Schwerpunkt (siehe Anhang A.1) der Pixelmenge P am dichtesten liegende Kreis:

$$P = \bigcup_{i \in \Lambda} S_i; \quad S_i \in CPS \quad (3.43)$$

Der Kreis cp_c wird der Menge CP^v zugefügt und aus CP' entfernt. Ausgehend von den Kreisen in CP^v werden alle überdeckenden Kreise in CP ermittelt, die korrespondierenden Kreise aus CP' entfernt und CP^v zugefügt. Der Schritt wird solange wiederholt, bis keine neuen Überdeckungen mehr ermittelt werden. Das Ergebnis ist die Menge von verifizierten Kreisparametern CP^v für die gilt:

$$CP^v = \{cp_i | \exists cp_j \in CP^v : (i \neq j \wedge g(cp_i, cp_j)) \wedge \nexists cp_j \in CP^v : (i \neq j \wedge o(cp_i, cp_j))\} \quad (3.44)$$

Die Grundidee des beschriebenen eigenen Ansatzes ist in [GV12] veröffentlicht. Der Ansatz wurde und soll auch weiter mit **PWL-H** (*Post Processed Wood Log Haarcascades*) abgekürzt werden. Einige Ergebnisse sind in Abbildung 3.26 verdeutlicht.



Abbildung 3.26: Verifikation der detektierten Kreise, wobei jeweils links die Menge CP und rechts die Menge CP^v in Form von roten Kreise abgebildet ist.

Zur Segmentierung von OG^{st} mit CP^v ergeben sich mehrere Möglichkeiten. Zum einen sind bereits Kreiskonturen und folglich Segmente durch die Kreise CP^v gegeben. Zum

3.3 Stammschnittflächensegmentierung mit Objektdetektion

anderen lassen sich die Kreiskonturen noch durch Anpassung an das Bild I_{wood} verbessern. In dieser Arbeit werden beide Möglichkeiten betrachtet, wodurch die zwei Ansätze

1. Direkte Abbildung der Kreisparameter CP^v
2. Anpassung der Kreiskonturen durch aktive Konturen (siehe [KWT88], [CV01])

verfolgt werden. Im ersten Ansatz ergeben sich die Segmente CPS^v direkt durch CP^v . Im zweiten Ansatz passt sich jeweils eine aktive Kontur, beginnend bei der Kontur C des Kreisparameters $cp \in CP^v$, iterativ an O^{st} an. Es gibt verschiedene Varianten von aktiven Konturen, wie bspw. in [KWT88], [XP98], [XP97] und [CV01] beschrieben. Alle Varianten lassen sich i.d.R. durch die Energiegleichung

$$E_{snake}^* = \int_0^1 E_{snake}(c(s)) = \int_0^1 E_{int}(c(s)) + E_{image}(c(s)) + E_{con}(c(s)) ds \quad (3.45)$$

formulieren. Details sind im Abschnitt 2.3.2.2 und im Anhang A.2 zu finden. Die interne Energie wird durch E_{int} , die externe Energie durch E_{image} und die externe Restriktion durch E_{con} definiert. In dieser Arbeit werden spezielle aktive Konturen verwendet. Die aktiven Konturen haben zwei interne Energien E_{int} , eine externe Energie E_{image} und eine externe Restriktion E_{con} . Die erste interne Energie basiert auf dem Abstand ausgewählter Konturpunkte und hält den Abstand der Punkte annähernd gleich. Die Zweite beschreibt die Konturkrümmung (zweite Ableitung von C) und verhindert das Zusammenfallen von C und starke Ausbeulungen. Die externe Energie E_{image} basiert auf den Bildgradienten $\nabla I(c(s))$ und bewegt die Kontur in Richtung hoher Gradienten. Die externe Restriktion E_{con} basiert auf einem Gradienten-Vektor-Feld (**GVF**, siehe [XP98] [XP97]). In einem **GVF** zeigen die Gradienten in Richtung der Kanten. Die Gradienten des **GVF** erzwingen eine Iteration der Kontur in Richtung der Kanten des Bildes. Die Bestimmung der finalen Konturen unterteilt sich in vier Schritte:

1. Berechnung eines Gradienten-Vektor-Feldes (**GVF**)
2. Bestimmung n äquidistanter Punkte auf jeder Kreiskontur
3. Iterative Minimierung der Energie E_{snake} anhand der n Konturpunkte
4. Approximation der Ergebnispunkte durch Splines

Nach der Berechnung des **GVF** werden n äquidistante Punkte entlang jedes Kreises in CP^v bestimmt. Die Konturpunkte werden iterativ durch Minimierung der genannten Energien neu bestimmt. Nach Konvergenz oder Überschreitung einer maximalen Anzahl von Iterationen wird die finale Kontur C' durch Splines approximiert. Die genannten Schritte sind in Abbildung 3.27 verdeutlicht.

Bei den aktiven Konturen und der Segmentierung nur mit Kreisen kommt es zu Überschneidungen und folglich Mehrdeutigkeiten. Die verifizierten Segmente CPS^v der Kreisparameter CP^v und die finalen Konturen C' können sich überschneiden. Eine Segmentierung und anschließende Abbildung eines Pixels px auf ein Label l ist folglich nicht

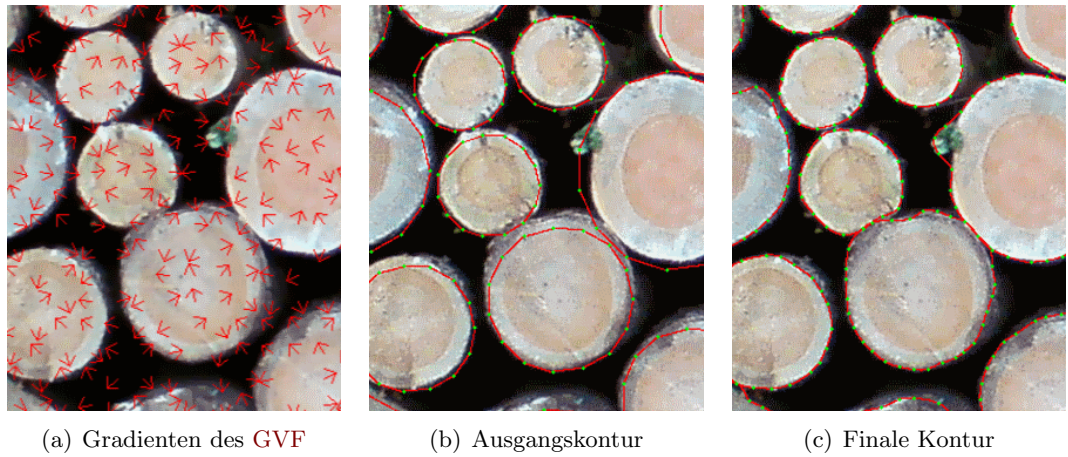


Abbildung 3.27: Links ist die Visualisierung des GVF, in der Mitte die Ausgangskontur mit n Punkten und rechts das Ergebnis nach der Iteration inklusive der Spline-Approximation zu sehen.

eindeutig. Ein Pixel $px \in I$ kann ein Element von mehreren Segmenten sein. Eine entsprechende Segmentierungsfunktion $f_{s|od}$ muss die Mehrdeutigkeiten auflösen und möglichst alle O^{st} überschneidungsfrei segmentieren. Abbildung 3.28 zeigt die Mehrdeutigkeiten beider Ansätze.

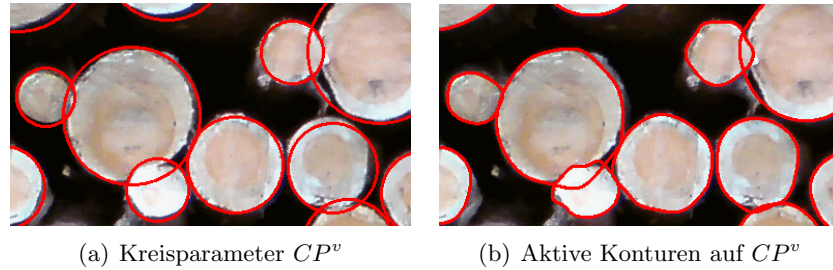


Abbildung 3.28: Links sind Überschneidungen der Kreisparameter und rechts der Konturen zu sehen. Alle Pixel in Überschneidungsbereichen sind mehrdeutig.

Im ersten Ansatz, Abbildung der Kreisparameter CP^v , reicht ein Vergleich eines mehrdeutigen Pixels px zu den Kreismittelpunkten der überlagerten Kreise aus. Die Kreisparameter CP^v lassen sich indexieren mit:

$$CP^v = \{cp_i | i \in \Lambda\}; \quad \Lambda = \mathbb{N}_{>0} \quad (3.46)$$

Die Indexmenge Λ soll analog auch das Label l kennzeichnen. Die nachfolgende Abstandsfunktion dr bestimmt den Abstand eines Pixels $px_{x,y}$ an der Position (x, y) relativ zum i -ten Kreisparameter cp_i :

$$dr(px_{x,y}, cp_i) = \sqrt{(x - x_i)^2 + (y - y_i)^2} - r_i; \quad (x_i, y_i, r_i) = cp_i \in CP^v \quad (3.47)$$

3.3 Stammschnittflächensegmentierung mit Objektdetektion

Die Auflösung der Mehrdeutigkeiten erfolgt durch die Segmentierungsfunktion:

$$f_{s|od}^{cp}(px) = \begin{cases} 0, & \nexists S \in CPS^v : px \in S \\ i, & \forall i \in \Lambda \nexists cp_j \in CP^v : (i \neq j) \wedge dr(px, cp_i) < dr(px, cp_j) \end{cases} \quad (3.48)$$

Die Funktion ordnet einem mehrdeutigen Pixel das Label desjenigen Kreises zu, dessen relativer Abstand vom Kreismittelpunkt zu px minimal ist.

Im zweiten Ansatz werden die Konturen der Kreise CP^v mit aktiven Konturen an das Bild I_{wood} angepasst. Das Ergebnis der Anpassung ist eine Menge CO aller Konturen C' , welche indiziert werden soll mit:

$$CO = \{C'_i | i \in \Lambda\}; \quad \Lambda = \mathbb{N}_{>0} \quad (3.49)$$

Die Auflösung der Mehrdeutigkeiten der resultierenden Konturen CO ist ähnlich dem ersten Ansatz. Aus CO lassen sich aktualisierte Kreisparameter CP^{co} ableiten. Je eine Kontur $C \in CO$ umschließt ein Segment S . Der Schwerpunkt (*center of mass*) und ein angepasster Radius r des Segmentes S ergeben zusammen die Menge der aktualisierten Kreisparameter:

$$CP^{co} = \{(x, y, r) | \text{angepasster Kreisparameter der Kontur } C' \in CO\} \quad (3.50)$$

Der Schwerpunkt (\bar{x}, \bar{y}) der Kontur C' ist trivial. Er ist der Schwerpunkt (siehe Anhang A.1) des umschlossenen Segmentes S . Der Radius r ergibt sich aus der Fläche A von S durch $r = \sqrt{A \div \pi}$. Mit den angepassten Kreisparametern CP^{co} und Konturen CO werden Mehrdeutigkeiten mit der Funktion

$$f_{s|od}^{co}(px) = \begin{cases} 0, & \text{außerhalb aller Konturen in } CO \\ i, & \forall i \in \Lambda \nexists cp_j \in CP^{co} : (i \neq j) \wedge dr(px, cp_i) < dr(px, cp_j) \end{cases} \quad (3.51)$$

aufgelöst. In Abbildung 3.28 wurden bereits die Mehrdeutigkeiten von CP^v und CO gezeigt. In Abbildung 3.29 ist die korrespondierende Lösung und analog das Ergebnis der Segmentierung zu sehen.

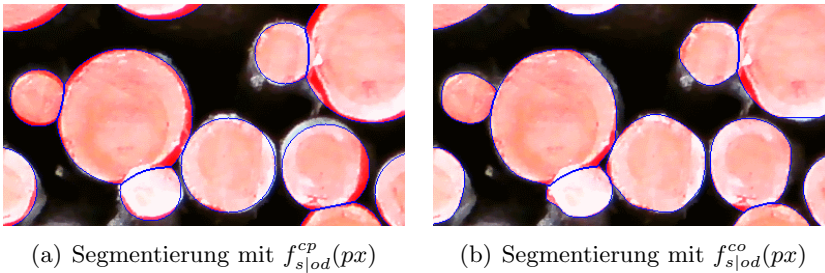


Abbildung 3.29: Rechts ist das Ergebnis der Segmentierung durch die Kreisparameter CP^v und links durch aktive Konturen zu sehen.

3.3.4 Zusammenfassung und Ergebnisse

In diesem Unterkapitel 3.3 wurde ein Konzept zur Segmentierung der Stammschnittflächen durch Objektdetektion erarbeitet. Im Gegensatz zum **WS-GC-R** Ansatz aus Abschnitt 3.2 sind weniger Restriktionen für die Bildaufnahme notwendig. Es müssen lediglich Stammschnittflächen eines Polters im Bild zu sehen sein. Das entwickelte Konzept unterteilt sich in die drei Schritte:

1. Detektion von Stammschnittflächenfenstern W
2. Verifikation der Ergebnisse der Detektion
3. Segmentierung des Bildes durch die detektierten Fenster

Im ersten Schritt werden die Stammschnittflächenfenster W anhand eines Modells Θ^O detektiert. Das Modell muss zuvor mit entsprechenden Trainingsdaten erstellt werden. Es wurden verschiedene Methoden der Objektdetektion untersucht und getestet. Von den untersuchten Methoden erwies sich die Haarkaskaden-Detektion als am besten geeignet. In Schritt zwei werden die Stammschnittflächenfenster (Rechtecke) W verifiziert, wobei falsch-positiv detektierte Fenster entfernt werden. Das Ergebnis ist die verifizierte Menge CP^v von Kreisparametern. Anhand der Kreisparameter wird das Bild I_{wood} zum einen direkt und zum anderen durch aktive Konturen segmentiert. Entstehende Mehrdeutigkeiten werden durch relative Abstandsvergleiche eines Pixels zu den entsprechenden Segmenten aufgelöst. Die Ergebnisse der Evaluierung sind in Tabelle 3.5 dargestellt. Einige Ergebnisbilder des Konzeptes sind im Anhang B.3 abgebildet.

Tabelle 3.5: Im oberen Bereich sind *precision*, *recall* und *f-score* der Stammschnittflächendetektion und im unteren Bereich der Schnittholzsegmentierung zu sehen. Im Mittleren Bereich ist das HD-Maß und der S-Fehler der Multi-Objekt-Segmentierung abgebildet. Die besten Werte sind fettgedruckt.

Method	Mittelwert			Standardabweichung		
Objektdetektion von OG^{st}	prec.	recall	f-score	prec.	recall	f-score
Detektierte Fenster W	0,876	0,653	0,74	0,087	0,137	0,114
Verifizierte Kreise CP^v	0,901	0,649	0,75	0,078	0,138	0,115
Segmentierung von OG^{st}	HD-Maß	S-Fehler	HD-Maß	S-Fehler		
Kreise (PWL-H)	0,889	0,191	0,0279	0,14		
Aktive Konturen	0,907	0,191	0,0263	0,14		
Schnittholzsegmentierung	prec.	recall	f-score	prec.	recall	f-score
Kreise (PWL-H)	0,853	0,741	0,788	0,052	0,111	0,078
Aktive Konturen	0,907	0,736	0,807	0,045	0,115	0,083

Evaluert wurden die Detektion mit Haarkaskaden, die Verifikation und die Multi-Objekt-Segmentierung von OG^{st} . Zusätzlich wurden die Ergebnisbilder der Multi-Objekt-Segmentierung I_m^{st} in Binärbilder I_h^b überführt. Alle nicht Hintergrundlabels ($l \neq 0$)

3.3 Stammschnittflächensegmentierung mit Objektdetektion

wurden auf 1 gesetzt. Die Binärbilder I_h^b repräsentieren eine Schnittholzsegmentierung und wurden ebenfalls evaluiert.

Zur Evaluierung wurden *precision*, *recall*, *f-score* und das HD-Maß (Huang-Dom-Maß) verwendet. Das HD-Maß basiert auf der Hamming Distanz eines Segmentes der Grundwahrheit zu allen Segmenten der Segmentierung und umgekehrt. Das Maß ist geeignet zur Evaluierung einer Multi-Objekt-Segmentierung. Weitere Details zum HD-Maß sind in [HD95] und im Anhang A.4 zu finden. Alle genannten Maße haben einen Wertebereich von $[0, 1]$ und das bestmögliche Ergebnis hat den Wert 1. Weiterhin wurde der S-Fehler (Stammanzahl-Fehler) verwendet. Der S-Fehler kennzeichnet den relativen Fehler der Stammanzahl. Er ergibt sich aus der realen s_r und der detektierten Stammanzahl s_d durch:

$$\text{S-Fehler} = \frac{|s_r - s_d|}{s_r} \quad (3.52)$$

Je kleiner der Fehler, umso besser das Ergebnis. Alle Werte der Evaluierung wurden anhand der 71 GT-Bilder bestimmt (siehe Anhang C.2).

In der Tabelle 3.5 hat der *f-score* der Stammschnittflächendetektion einen Wert von 0,741. Das bedeutet, dass im Durchschnitt die Stammschnittflächen zu 74 % richtig detektiert wurden. Nach der Verifikation ist der *f-score* leicht besser und liegt bei 0,75. Allerdings hat sich der *recall* von 0,876 auf 0,901 verbessert. Im Fazit führt die Verifikation zu einer Verbesserung der Ergebnisse.

Die beiden Segmentierungsansätze Kreise und aktive Konturen zeigen unterschiedliche HD-Maße. Das HD-Maß bei der Segmentierung durch aktive Konturen ist etwas besser, als die durch einfache Kreise. Der S-Fehler ist bei beiden Ansätzen gleich, da sich die Anzahl der Segmente bei beiden Ansätzen nicht verändert. Beide Ansätze basieren auf den verifizierten Kreisparametern CP^v . Der S-Fehler zeigt, dass die Anzahl der Stämme geringfügig fehlerhaft ist. Die Betrachtung der Ergebnisse als binäres Schnittholzbild I_h^b bestätigt, dass aktive Konturen zu besseren Ergebnissen führen.

Alles in allem ist das beschriebene und evaluierte Konzept der Segmentierung durch Stammschnittflächendetektion durchaus für die Lösung der Problemstellung geeignet. Es werden wenige Restriktionen in der Bildaufnahme benötigt und die Stammschnittflächen werden einzeln segmentiert. Das Ergebnis der Multi-Objekt-Segmentierung ist das Bild I_{st}^m . Das Konzept hat allerdings den Nachteil, dass von den Stammschnittflächen ein gutes Objektmodell Θ^O notwendig ist und nicht detektierte Stammschnittflächen auch nicht segmentiert werden.

3.4 Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion

Im Unterkapitel 3.2 wurde ein Konzept der Schnittholzsegmentierung mit Graph-Cut und im Unterkapitel 3.3 ein Konzept der Stammschnittflächendetektion entwickelt und evaluiert. In diesem Unterkapitel sollen beide Konzepte kombiniert werden, um die jeweiligen Vorteile zu vereinen und die Nachteile zu verringern. Ein Vorteil der Graph-Cut-Segmentierung ist die Genauigkeit. Ein Vorteil der Segmentierung mit Objektdetektion ist, dass es eine Multi-Objekt-Segmentierung ist, welche einzelne Stammschnittflächen segmentiert. Ein weiterer Vorteil ist, dass weniger Restriktionen in der Bildaufnahme notwendig sind.

In diesem Unterkapitel wird eine Segmentierung der Stammschnittflächen OG^{st} mit Graph-Cut und Objektdetektion konzipiert, beschrieben und evaluiert. In Abschnitt 3.4.1 wird zuerst das Konzept der kombinierten Segmentierung erarbeitet. Durch die detektierten Stammschnittflächenfenster W wird eine **Trimap** erstellt, welche wie folgt definiert werden soll:

$$I^{tri} = \{l^{tri} | \forall px \in I_{wood} \exists l^{tri} \in L^{tri}\}; \quad L^{tri} = \{fg, bg, unknown\} \quad (3.53)$$

Ein Label l^{tri} der **Trimap** korrespondiert zu einem Pixel im Bild I_{wood} und kennzeichnet ein Pixel als unbekannt (*unknown*), Vorder- (*fg*) oder Hintergrund (*bg*). Auf Basis der **Trimap** wird Schnittholz binär segmentiert. Im Binärbild I_h^b werden anschließend anhand der detektierten Stammschnittflächenfenster W einzelne Segmente extrahiert (Separierung). In Abschnitt 3.4.2 wird die Erstellung der **Trimap** und in Abschnitt 3.4.3 die Separierung der Stammschnittflächen OG^{st} untersucht und beschrieben. Im letzten Abschnitt 3.4.4 werden die wesentlichen Erkenntnisse zusammengefasst und die Ergebnisse evaluiert.

3.4.1 Konzeption einer kombinierten Segmentierung

Die entwickelten Konzepte der Schnittholzsegmentierung und der Stammschnittflächendetektion aus den Unterabschnitten 3.2 und 3.3 haben verschiedene Vor- und Nachteile. Die Graph-Cut-basierte binäre Segmentierung von Schnittholz hat sich als robust und weitestgehend genau herausgestellt. Sie hat allerdings zwei Nachteile. Der erste Nachteil ist die Restriktion in der Bildaufnahme von I_{wood} . Speziell muss im oberen und unteren Bildbereich von I_{wood} Hintergrund (Nicht-Schnittholzpixel) und im mittleren Bildbereich Schnittholz zu sehen sein. Der zweite Nachteil ist die binäre Segmentierung. Es wird nur Schnittholz segmentiert, wodurch die einzelnen Stammschnittflächen unbekannt sind. Zur Berechnung der Fläche des Schnittholzes nach einer entsprechenden Maßstabsbestimmung ist das Konzept geeignet, aber nicht zur Ermittlung der Stammgrößen. Das Konzept der Stammschnittflächendetektion segmentiert hingegen einzelne Stammschnittflächen und benötigt weniger Restriktionen in der Bildaufnahme. Allerdings benötigt das Konzept ein gutes Objektmodell Θ^O . Außerdem werden atypische Stammschnittflächen, wie kaputte, dreckige oder verdeckte Flächen, nicht detektiert.

3.4 Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion

Die Vorteile beider Konzepte lassen sich kombinieren. Theoretisch kann die Abschätzung von Vorder- und Hintergrund durch die detektierten Stammschnittflächen erfolgen. Damit verringern sich die Restriktionen in der Bildaufnahme und der Graph zur Graph-Cut-Segmentierung kann ohne heuristische Annahmen in der Polterposition erstellt werden. Das Ergebnis der Graph-Cut-Segmentierung ist ein binäres Bild und das Ergebnis der Stammschnittflächendetektion sind n Bildobjekte. Das binäre Bild segmentiert das Schnittholz genauer, als die Bildobjekt. Der Vergleich beruht auf der binären Betrachtung der Bildobjekte als Bildobjektpixel und Nicht-Bildobjektpixel. Die Ergebnisse der Stammschnittflächendetektion zeigen, dass nach der Verifikation der detektierten Objekte mit Haarkaskaden (Θ_7^O) nur wenige falsch-positiv detektierte Bildobjekte vorkommen. Der größte Fehler liegt in der falsch-negativ Detektion. Das heißt, dass nur ein geringer Teil des Hintergrunds als Stammschnittfläche detektiert wird, aber stattdessen einige Stammschnittflächen gar nicht detektiert werden. Es folgt daraus die Idee, die detektierten Stammschnittflächen aus dem binären Bild zu separieren und zusätzlich auch aus den verbleibenden Schnittholzsegmenten die Stammschnittflächen zu segmentieren. Aus der Diskussion der Vor- und Nachteile leitet sich das grobe **Konzept der Kombination** in Abbildung 3.30 ab.



Abbildung 3.30: Grobes Konzept zur Segmentierung der Stammschnittflächen durch Objektdetektion und Graph-Cut.

Das Konzept gliedert sich in fünf Schritte. Die ersten beiden Schritte sind identisch mit dem Konzept der Stammschnittflächendetektion und erzeugen eine Menge von verifizierten Kreisparameter CP^v . Der dritte Schritt, die Vorder- und Hintergrundabschätzung,

ist hinzugekommen und berechnet eine **Trimap** auf Basis von CP^v . Der vierte Schritt beinhaltet eine angepasste Graph-Cut-Segmentierung des **KD-NN**-Ansatzes und segmentiert binär Schnittholz im Bild I_{wood} anhand der **Trimap**. Der fünfte und letzte Schritt ist ebenfalls neu und separiert Stammschnittflächen auf Basis von CP^v aus dem binären Schnittholzbild I_h^b . Das Ergebnis der Multi-Objekt-Segmentierung ist das Bild I_{st}^m .

Aus dem groben Konzept ergeben sich die folgenden **Fragestellungen und Forschungsinhalte**:

1. Wie können mit den verifizierten Kreisparametern CP^v Vorder- und Hintergrund (FG^{smp} , BG^{smp}) abgeschätzt werden?
 - Wie können fehlerhafte oder fehlende Objekte in CP^v kompensiert werden?
 - Welche Möglichkeiten der **Trimap** gibt es und welche ist davon geeignet?
2. Wie lassen sich CP^v und das binäre Segmentierungsergebnis I_h^b zur Segmentierung von OG^{st} kombinieren.
 - Welche Möglichkeiten bestehen, I_h^b weiter in kreisähnliche Segmente zu zerlegen?
 - Sind die Ergebnisse einer Zerlegung von I_h^b genau genug?
 - Wie kann die fehlerbehaftete Menge CP^v zur Separierung beitragen?

Nachfolgend werden nur die neuen ergänzten Schritte drei und fünf des Konzeptes in Abbildung 3.30 ausführlich betrachtet. Die Erzeugung einer **Trimap** auf Basis der verifizierten Kreise CP^v und die Anpassung der Graph-Cut-Segmentierung werden in Abschnitt 3.4.2 betrachtet. In Abschnitt 3.4.3 werden verschiedene Methoden zur Separierung von Stammschnittflächen aus dem binären Schnittholzbild I_h^b diskutiert und weiterentwickelt. Die Ergebnisse der kombinierten Segmentierung der verschiedenen Methoden werden in Abschnitt 3.4.4 gegenübergestellt und diskutiert.

3.4.2 Ermittlung der **Trimap** und Graph-Cut

Im Unterkapitel 3.2 erzielte die eigene Methode **KD-NN** in der Schnittholzsegmentierung die besten Ergebnisse. Die auf Graph-Cut basierende Methode benötigt je ein Wahrscheinlichkeitsbild vom Vorder- und vom Hintergrund (I_{fg}^p , I_{bg}^p). Die Wahrscheinlichkeitsbilder werden auf Basis abgeschätzter Stichproben von Schnittholz- (FG) und nicht Schnittholzpixeln (BG) ermittelt. Die Stichproben werden durch die zwei disjunkten Mengen FG^{smp} und BG^{smp} repräsentiert. Die Menge der verifizierten Kreisparameter CP^v ist nicht frei von Fehlern. Der häufigste Fehler besteht darin, dass einige Stammschnittflächen nicht in CP^v enthalten sind.

Zur Erzeugung einer **Trimap** I^{tri} anhand von CP^v ergeben sich verschiedene Möglichkeiten. So besteht eine Möglichkeit darin, die Pixel der Kreissegmente CPS^v als Vordergrund ($L^{tri} = fg$) und alle Anderen als Hintergrund ($L^{tri} = bg$) anzunehmen. Da die Kreise **i.d.R.** nicht genau eine Stammschnittfläche umschließen, sollte zusätzlich entlang der Kreiskontur ein Ring mit einer gewissen Breite relativ zum Radius mit dem Label $L^{tri} = unknown$ versehen werden. Allerdings können durch das Vorgehen viele Hintergrundpixel als

3.4 Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion

Vordergrund angenommen werden, da zumeist nicht alle Stammschnittflächen durch CP^v repräsentiert werden. Eine andere und genauere Möglichkeit besteht ähnlich wie im **WS-GC-R**-Ansatz in einer zweistufigen Segmentierung. Im Bildraum bildet CP^v ein Cluster. Auch wenn die Kreise von CP^v fehlerbehaftet sind, streuen die Kreise theoretisch immer um den mittleren Bereich des Schnittholzes im Bild. Damit lässt sich der mittlere Schnittholzbereich automatisch bestimmen und die entsprechende Restriktion in der Bildaufnahme ist nicht mehr notwendig, wie im **WS-GC-R**-Ansatz. Der automatisch bestimmte mittlere Bereich wird segmentiert und mit dem Ergebnis eine **Trimap** I^{tri} erstellt. Der mittlere Bereich ist analog das Zentrum von CP^v . Das Zentrum wird durch die innersten Kreise gebildet. Alle Pixel der Segmente CPS^v von CP^v ergeben zusammen die Pixelmenge P :

$$P = \bigcup_{i \in \Lambda} S_i; \quad S_i \in CPS; \quad (3.54)$$

Der Schwerpunkt der Segmentmenge hat die Position (\bar{x}, \bar{y}) und ist durch P einfach berechenbar (siehe Anhang A.1). Die innersten Kreise von CP^v sind die n dichtesten Kreise zum Schwerpunkt. Da die Kreise CP^v fehlerbehaftet sind, ist es sinnvoll eine geringe relative Anzahl n von Kreisen zu nehmen, wie $n = |CP^v|/4$. Es wird damit gewährleistet, dass der Bereich immer innerhalb des Schnittholzes liegt. Die konvexe Hülle über alle n inneren Kreise ergibt den gesuchten mittleren Schnittholzbereich mit Stammschnittflächen und Stammzwischenräume. In Abbildung 3.31 ist die Methodik der Bestimmung des inneren Schnittholzbereiches verdeutlicht.



Abbildung 3.31: Die Zwischenschritte zur Bestimmung des inneren Schnittholzbereiches: Von links nach rechts sind CP^v , die inneren Kreise und der innere Bereich zu sehen.

Der innere Schnittholzbereich enthält Stammschnittflächen und Stammzwischenräume. Wie bereits im **WS-GC-R**-Ansatz wird der Bereich segmentiert und das Resultat sind Schnittholz- und Nicht-Schnittholzpixel. In der **Trimap** I^{tri} bekommen die Pixel des inneren Bereiches jeweils das Label fg oder bg . Alle Schnittholzpixel werden mit dem Label fg und alle Nicht-Schnittholzpixel mit dem Label bg versehen. Alle übrigen Pixel erhalten das Label *unknown*. Der Hintergrund ($L^{tri} = bg$) besteht bisher nur

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

aus Pixeln der Stammzwischenräume des inneren Schnittholzbereiches. Um eine gute Segmentierung mit Graph-Cut zu erzeugen, müssen auch nicht Schnittholzpixel des oberen und unteren Bildbereiches in BG^{smv} enthalten sein. Zur Bestimmung der Pixel ist es sinnvoll, wieder die verifizierten Kreissegmente CPS^v zu nutzen, um die Restriktionen in der Bildaufnahme abzuschwächen. Aus den verifizierten Kreisparametern lässt sich durch die Segmentierungsfunktion $f_s^c(px)$ das binäre Bild I_c^b erzeugen:

$$f_s^c(px) = \begin{cases} 1, & px \in S \wedge S \in CPS^v \\ 0, & \text{sonst} \end{cases} \quad (3.55)$$

Im Bild I_c^b gehören große leere Bereiche $px = 0$ mit hoher Wahrscheinlichkeit zum Hintergrund, da die Stammschnittflächen im Bildraum einen Cluster mit kleinen Zwischenräumen bilden. Durch eine Distanztransformation (siehe [RP66], [LW13]) ist es möglich, die Distanz von einem Pixel zum dichtesten Schnittholzpixel zu bestimmen. Nach einer Transformation des Bildes I_c^b in das Distanzbild I^d erhalten Pixel in leeren Bereichen des Bildes I^d einen relativ hohen Wert. Mit einer Schwellwertsegmentierung nach einer Normalisierung von I^d lassen sich leere Bereiche segmentieren. Ein guter Schwellwert T für die zwischen $[0, 1]$ normierten Distanzen ist 0,75. Alle Distanzen größer als T werden als Nicht-Schnittholzpixel segmentiert und erhalten in der **Trimap** das Label bg . Die Abbildung 3.32 verdeutlicht die einzelnen Schritte.

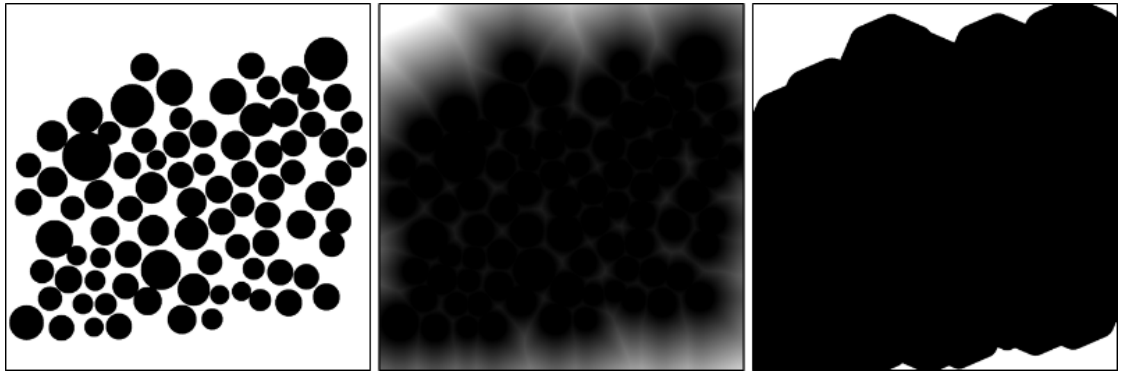


Abbildung 3.32: Die Zwischenschritte zur Bestimmung des Außenbereiches eines Polters: Von links nach rechts sind das Bild I_c^b , das Distanzbild I^d und das segmentierte Bild zu sehen

Zusammenfassend haben in der finalen **Trimap** I_v^{tri} die Labels L^{tri} den Wert:

- fg , wenn px ein Schnittholzpixel des inneren Schnittholzbereiches ist
- bg , wenn px ein Nicht-Schnittholzpixel des inneren Schnittholzbereiches ist
- bg , wenn px ein Pixel des segmentierten Außenbereiches ist
- $unknown$, wenn px weder bg noch fg ist

3.4 Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion

Für die Graph-Cut-Segmentierung werden alle Pixel der **Trimap** mit dem Label fg der Pixelmenge FG^{smp} und mit bg der Pixelmenge BG^{smp} hinzugefügt. Letztendlich wird mit den Pixelmengen FG^{smp} und BG^{smp} das Bild mit Graph-Cut (**KD-NN**) segmentiert. Die einzelnen Schritte der **Trimap**-Erzeugung und Segmentierung sind in Abbildung 3.33 verdeutlicht.

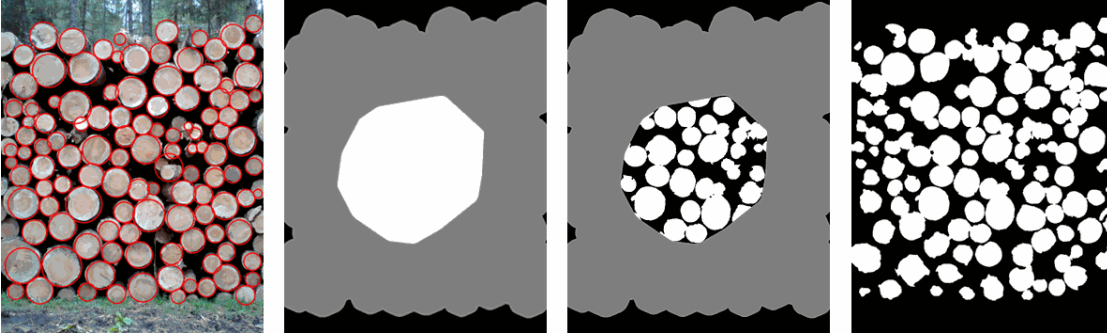


Abbildung 3.33: Die Zwischenschritte in der binären Schnittholzsegmentierung durch CP^v : Von links nach rechts sind CP^v , die **Trimap** I^{tri} , die verfeinerte **Trimap** I_v^{tri} und das Ergebnisbild I_h^b zu sehen. Weiße Pixel in I^{tri} sind fg , schwarze sind bg und graue *unknown*.

Der spezielle Ansatz der Schnittholzsegmentierung mit Hilfe der Kreisparameter CP^v ist unter dem Namen **KD-NN-A** (*KD-NN adaptive*) in [GV12] veröffentlicht.

3.4.3 Separierung der Stammschnittflächen

In diesem Abschnitt soll ein Ansatz zur Separierung von Stammschnittflächen auf dem binären Schnittholzbild I_h^b entwickelt werden. Die Kreisparameter CP^v enthalten bereits Positionen der Stammschnittflächen. Entsprechend der Evaluierung sind die Positionen nicht frei von Fehlern und decken *i.d.R.* die Stammschnittflächen nicht vollständig ab. Dennoch enthält CP^v Vorwissen über die Positionen der Stammschnittflächen und soll entsprechend zur Separierung genutzt werden.

Das binäre Bild I_h^b trennt Schnittholzpixel von Nicht-Schnittholzpixeln. Die Evaluierung anhand der 71 verschiedenen Evaluierungsbilder des **WS-GC-R** Ansatzes hat gezeigt, dass die Schnittholzsegmentierung relativ genau ist ($f\text{-score}=0,8981$, Abschnitt 3.2.4). Im Vergleich dazu sind die verifizierten Kreise CP^v als Binärbild betrachtet ungenauer ($f\text{-score}=0,788$, Abschnitt 3.3.4). Da sowohl das Binärbild I_h^b und die detektierten Kreise einen relativ hohen $f\text{-score}$ aufweisen, ist es sinnvoll, zuerst die Ergebnisse zu verbessern. Die Frage ist, wie zwei fehlerbehaftete Verfahren verbessert werden können? Eine Lösung ist, die Verbesserung der Ergebnisse eines Verfahrens durch die Ergebnisse des Anderen. Allerdings kann sowohl CP^v anhand von I_h^b als auch I_h^b anhand von CP^v verbessert werden. Es ergibt sich ein **Henne-Ei-Problem**. Da eine Entscheidung nicht möglich ist, ist ein kombinierter Ansatz aus beiden Richtungen notwendig.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Im kombinierten Ansatz wird zuerst das Binärbild und CP^v mit Hilfe der Überdeckungsbedingung

$$oc(cps, I^b, T) = \begin{cases} \text{wahr,} & \frac{|cps \check{\cap} I^b|}{|cps|} > T \\ \text{falsch,} & \text{sonst} \end{cases}; cps \in CPS^v; T \in [0, 1]; T \in \mathbb{R} \quad (3.56)$$

verbessert. Die Bedingung oc (*overlap circle*) ist erfüllt, wenn die relative Überdeckung des Segments $cps \in CPS^v$ mit dem korrespondierenden Segment in I^b einen gewissen Schwellwert T überschreitet. Ein empirisch ermittelter guter Schwellwert ist $T=0,6$. Ist die Bedingung erfüllt, so wird cps zu CPS^{oc} hinzugefügt. Für CPS^{oc} gilt:

$$CPS^{oc} = \{cps | \forall cps \in CPS^{oc} : oc(cps, I^b, T) \wedge cps \in CPS^v\} \quad (3.57)$$

Das Binärbild I_h^b wird anschließend verbessert, indem die Segmente der Menge CPS^{oc} mit I_h^b vereinigt werden. Um sicherzustellen, dass nur wenige bis keine falschen Holzpixel hinzukommen, werden die Kreise CP^{oc} der Segmente CPS^{oc} um T skaliert.

$$CP^s = \{(x, y, T * r) | (x, y, r) \in CP^{oc}\}; T \in [0, 1]; T \in \mathbb{R} \quad (3.58)$$

Nach der Skalierung ergeben sich die skalierten Kreisparameter CP^s und entsprechenden Segmente CPS^s . Sie werden mit dem Bild I_h^b zu dem verifizierten Bild I_v^b wie folgt vereinigt¹⁰:

$$I_v^b = \bigcup_{\lambda \in \Lambda} (CPS_{\lambda}^s \check{\cup} I_h^b), \quad (3.59)$$

wobei Λ wieder die Indexmenge ist und CPS^s indiziert. Durch die Verbesserung werden kleine Löcher in I_v^b geschlossen und im idealen Fall falsch-positiv detektierte Stammschnittflächen entfernt. In Abbildung 3.34 sind die beschriebenen Schritte verdeutlicht.

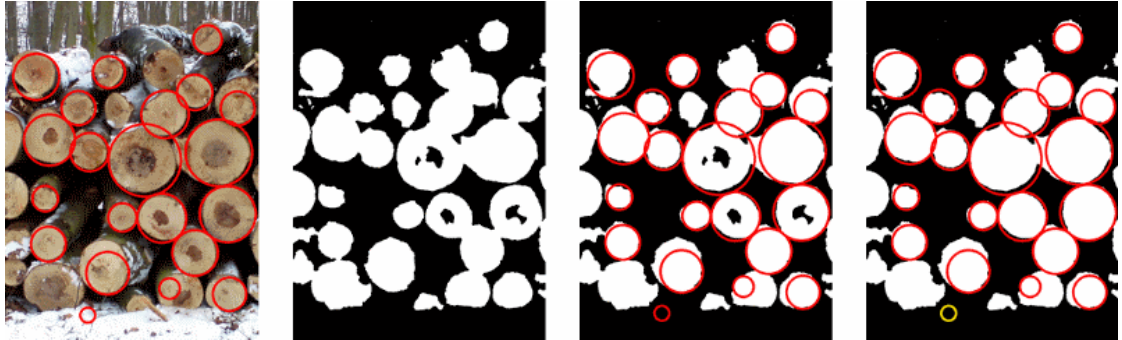


Abbildung 3.34: Die Schritte der Verbesserung von CP und I^b : Von links nach rechts sind CP^v in I_{wood} , I_h^b , I_h^b mit CP^v und die Ergebnisse I_v^b und CP^{oc} zu sehen. Gelbe Kreise sind eliminierte Kreisparameter.

¹⁰Das Symbol $\check{\cup}$ kennzeichnet die Pixelweise ODER-Verknüpfung; siehe Anhang A.1

3.4 Stammschnittflächensegmentierung mit Graph-Cut und Objektdetektion

Die Ergebnisse der genannten Schritte sind die bereinigten Kreisparameter CP^{oc} und das verbesserte Schnittholzbild I_v^b . Nach der Verbesserung von I_h^b zu I_v^b durch CP^v wird nun entgegengesetzt CP^{oc} durch I_v^b verbessert. Bisher wurde die Segmentmenge CPS der Kreisparameter CP ohne Überföhrungsfunktion verwendet, da die Erzeugung aus der Formel 3.35 klar hervorgeht. Für die weitere Betrachtung ist eine Überföhrungsfunktion einzelner Kreisparameter notwendig. Sie soll definiert werden mit:

$$\begin{aligned} f_{cps} &: (x, y, r) \mapsto S; x, y, r \in \mathbb{R}; S \in SI \\ f_{cps}(cp) &= \{S | \forall p_{x,y} \in S : (x_c - x)^2 + (y_c - y)^2 \leq r^2\} \end{aligned} \quad (3.60)$$

Die Kreisparameter in CP^{oc} lassen sich optimieren, indem die Positionen (x, y) und Radien r an das Binärbild I_v^b angepasst werden. Dazu wird für jeden Kreisparameter die Funktion ec mit der Nebenbedingung, dass der Radius r maximal wird, minimiert.

$$\min_{x,y,r} ec(cp, I^b) = 1 - \frac{|f_{cps}(cp) \cap I^b|}{|f_{cps}(cp)|}; cp = (x, y, r); x, y, r \in \mathbb{R} \quad (3.61)$$

Zur Lösung des Minimierungsproblems wurde der **GMC**-Algorithmus (*Growing Moving Circle*) entwickelt. Die Beschreibung inklusive des Pseudocodes ist im Anhang D.1 zu finden. Der **GMC**-Algorithmus startet mit den herunterskalierten Kreisparametern CP^{oc} . Für alle Positionen $(x, y, r) \in CP^s$ innerhalb einer Achter-Nachbarschaft und Zwei-Pixel-Suchweite $(x+1, x+2, y+1, y+2, r+1, r+2, \text{ usw.})$ wird geprüft, wo die Energie (ec) am geringsten ist. Entsprechend der neuen Position wachsen oder bewegen sich die Kreise, bis keine neue Position mehr existiert, wo die Energie ec kleiner als der Schwellwert T ist. Das Wachstum der Kreise hat Vorrang vor einer Bewegung, da sich die Kreise ansonsten hin und her bewegen und es im schlechtesten Fall zu keiner Konvergenz kommt. Alle neuen Kreisparameter werden im letzten Schritt nochmal hinsichtlich Überschneidungen geprüft und ggf. miteinander verschmolzen. Das Ergebnis sind die angepassten Kreisparameter CP^a . In Abbildung 3.35 ist der **GMC**-Algorithmus verdeutlicht.

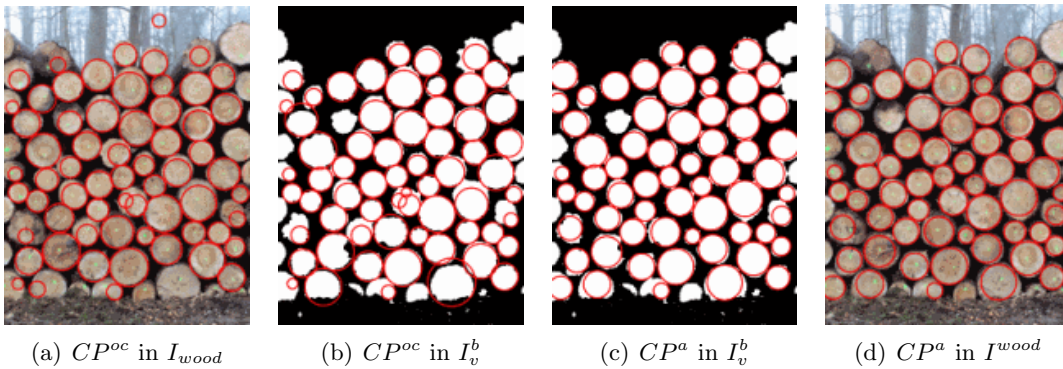


Abbildung 3.35: Die Abbildung zeigt in (a) und (b) die Ausgangskreise CP^{oc} und in (c) und (d) die Ergebniskreise CP^a des **GMC**-Algorithmus.

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

Die angepassten Kreise CP^a des GMC-Algorithmus überdecken optimal das Binärbild I_v^b . Allerdings gibt es in I_v^b noch Bereiche, in denen Schnittholz segmentiert ist, aber kein Kreis existiert. Der Grund ist, dass Stammschnittflächen nicht detektiert wurden. Stammschnittflächen kennzeichnen sich in I_v^b i.d.R. durch homogene kreisförmige Bildbereiche mit $l = 1$ (siehe Abbildung 3.34, 3.35 und Anhang B.3). Damit auch in CP^a fehlende, aber in I_v^b segmentierte Stammschnittflächen O^{st} separiert werden, werden als nächstes die Kreise CP^a vom Schnittholzbild I_v^b entfernt¹¹ durch:

$$I_{rest}^b = I_v^b \setminus \left(\bigcup_{\lambda \in \Lambda} CPS_{\lambda}^a \right) \quad (3.62)$$

Die Kreissegmentmenge CPS^a wird wieder durch die Indexmenge Λ indexiert. Das Ergebnis der Formel 3.62 ist das Binärbild I_{rest}^b , welches überwiegend nicht detektierte Holzpixel enthält. Um potenzielle Kreise in I_{rest}^b zu finden, wird eine Distanztransformation auf dem invertierten Binärbild \bar{I}_{rest}^b ausgeführt. Relativ große lokale Maxima im Distanzbild repräsentieren Kreismittelpunkte (x, y) eines Kreissegmentes. Aus diesem Grund werden die lokalen Maxima gesucht. Der Wert eines lokalen Maximums entspricht dem Radius r des Kreissegmentes. Aus der Position und den Werten der Maxima wird die neue Menge CP^{rest} von Kreisparametern generiert. Da im Distanzbild viele unbedeutende lokale Maxima auftauchen können, wird die Menge CP^{rest} zuerst bereinigt. Dazu werden alle Kreise, die sich mit einem größeren Kreis in mehr als 60% der Fläche überdecken, aus CP^{rest} entfernt. Die Kreise in CP^{rest} sind noch nicht an das Schnittholzbild I_h^b angepasst. Außerdem können einige Kreise im schlechtesten Fall in der Nähe eines bereits in CP^a enthaltenen Kreises liegen. Aus den genannten Gründen werden zuerst alle Kreismengen vereinigt $CP_{rest} \cup CP^a$. Die vereinigte Menge wird anschließend wieder durch den GMC-Algorithmus an das Schnittholzbild I_v^b angepasst. Im finalen Schritt werden die ermittelten Kreise CP_{all} durch die Funktion $f_{s|od}^{co}(px)$ (siehe Formel 3.51 in Abschnitt 3.3.3) auf das Binärbild I_v^b abgebildet und das Labelbild I_{st}^m erzeugt.

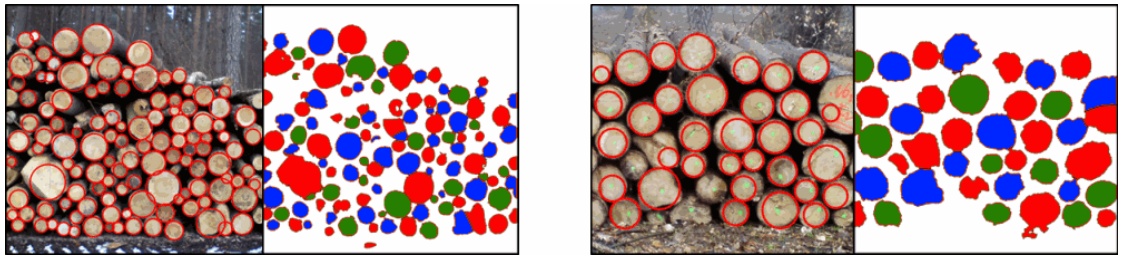


Abbildung 3.36: Die Abbildung zeigt zwei Ergebnisse des LSGMC, wobei jeweils links I_{wood} mit CP^v und rechts I_{st}^m zu sehen ist. Zur besseren visuellen Unterscheidung sind die Segmente in drei Farben dargestellt.

¹¹Das Symbol \setminus kennzeichnet die pixelweise Differenz, siehe Anhang A.1.

Das beschriebene Verfahren zur Separierung der Kreise CP_{all} und Bestimmung des Labelbildes I_{st}^m aus einem Schnittholzbild I_h^b ist unter dem Namen **LSGMC** (*Log Separation by Growing Moving Maxima*) in [GV12] veröffentlicht. Zwei Ergebnisbilder sind in Abbildung 3.36 dargestellt. Der entsprechende Algorithmus und Pseudocode befindet sich im Anhang D.2.

3.4.4 Zusammenfassung und Ergebnisse

In diesem Unterkapitel 3.4 wurde ein kombiniertes Konzept zur Segmentierung von Stammschnittflächen OG^{st} erarbeitet. Das Konzept kombiniert das Konzept der Schnittholzsegmentierung (Unterkapitel 3.2) mit dem Konzept der Stammschnittflächendetektion (Unterkapitel 3.3). Das kombinierte Konzept hat weniger Restriktionen in der Bildaufnahme als das Konzept der Schnittholzsegmentierung (**WS-GC-R**-Ansatz). Die einzige Restriktion ist, dass einige Stammschnittflächen im Inputbild I_{wood} zu sehen sein müssen. Weiterhin wird nicht nur Schnittholz binär segmentiert, sondern einzelne Stammschnittflächen. Das kombinierte Konzept setzt sich primär aus den zwei entwickelten Algorithmen **KD-NN-A** und **LSGMC** zusammen. Die einzelnen Schritte des Konzeptes der Kombination sind:

1. Stammschnittflächendetektion
2. Verifikation der Stammschnittflächenkandidaten
3. Vorder- und Hintergrundabschätzung sowie **Trimap**-Erstellung (**KD-NN-A**, Teil 1)
4. Graph-Cut-Segmentierung (**KD-NN-A**, Teil 2)
5. Separierung der Stammschnittflächen (**LSGMC**)

Das Ergebnis der Anwendung des Konzeptes ist das Labelbild I_{st}^m , welches Hintergrund und einzelne Stammschnittflächen voneinander trennt. Die Schritte 1 und 2 sind aus dem Konzept der Stammschnittflächendetektion und der Schritt 4 aus dem Konzept der Schnittholzsegmentierung. Die Schritte 3 und 5 wurden neu entwickelt. Die Schritte 3 und 4 zusammen kennzeichnen den **KD-NN-A**. Der **KD-NN-A** besteht aus den Teilschritten:

- a) **Trimap**-Abschätzung anhand verifizierter Stammschnittflächen CP^v
 - Abschätzung des mittleren Holzbereiches durch Schwerpunkt von CP^v
 - Hintergrundabschätzung durch Segmentierung des Distanzbildes zu CP^v
- b) Segmentierung des mittleren Holzbereiches anhand der Holzfarbe und Graph-Cut
- c) Erstellung eines Vorder- und Hintergrundmodells (**KD-NN**) und Graph-Cut

Der Schritt 5 ist der **LSGMC**-Algorithmus zur Separierung von Stammschnittflächen mit verifizierten Kreisparametern CP^v aus dem Schnittholzbild I_h^b . Die grundlegenden Teilschritte sind:

- a) Verifikation des Schnittholzbildes I_h^b mit CP^v und umgekehrt; Output: I_v^b, CP^{oc}

3 Automatische Segmentierung von Stammschnittflächen und Schnittholz

- b) Ausführung des GMC-Algorithmus zur Anpassung von CP^{oc} an I_v^b
- c) Entfernung der Kreise CP^{oc} aus I_v^b zu I_{rest}^b
- d) Ermittlung fehlender Kreise durch Auswertung lokaler Maxima in I_{rest}^b
- e) Ausführung des GMC-Algorithmus und Anpassung aller Kreise
- f) Abbildung der angepassten Kreise auf das Binärbild I_v^b und Bestimmung von I_{st}^m

Für das Problem der Kreisanpassung in ein Binärbild wurde der GMC-Algorithmus entwickelt. Der Algorithmus und Pseudocode des LSGMC und GMC ist im Anhang D zu finden.

Für die Evaluierung des Konzeptes wurden die Ergebnisse mit den 71 GT-Bildern in Anhang C.2 verglichen. Weiterhin werden zur quantitativen Evaluierung *precision*, *recall*, *f-score* und das HD-Maß (siehe Anhang A.4) angewandt. Alle Maße liegen im Intervall $[0, 1]$, wobei ein Wert von 1 das bestmögliche Ergebnis kennzeichnet. Weiterhin wurde der S-Fehler (siehe Anhang A.4) verwendet. Je kleiner der S-Fehler, umso besser das Ergebnis. Die Ergebnisse des Konzeptes bestehen aus den binären Schnittholzbildern I_v^b und den Labelbildern der Stammschnittflächen I_{st}^m . In Tabelle 3.6 sind die Ergebnisse evaluiert und anderen Methoden gegenübergestellt. Die Ergebnisse der Schnittholzsegmentierung werden mit dem WS-GC-R-Ansatz verglichen. Die Ergebnisse der Stammschnittflächensegmentierung sind der Methode in [CYP04] gegenübergestellt. In [CYP04] wird zur Objektseparierung im Binärbild ein saarpunktbasierter *Watershed* angewandt. Die Saarpunkte entsprechen den lokalen Maxima im Distanzbild.

Tabelle 3.6: In der oberen Tabelle sind *precision*, *recall* und *f-score* der Schnittholzsegmentierung und in der unteren das HD-Maß und der S-Fehler der Stammschnittflächensegmentierung abgebildet. Die besten Werte sind fettgedruckt.

Segmentierungsmethode	Mittelwert			Standardabweichung		
Schnittholz f_s^h	prec.	recall	f-score	prec.	recall	f-score
WS-GC-R	0,872	0,94	0,9	0,089	0,0473	0,0518
KD-NN-A	0,878	0,95	0,91	0,08	0,0334	0,043
Stammschnittflächen f_s^{st}	HD-Maß	S-Fehler	HD-Maß	S-Fehler		
Water-Dis auf KD-NN-A I_h^b	0,918	0,8046	0,03	0,7296		
LSGMC auf KD-NN-A I_h^b	0,926	0,1508	0,03	0,153		

Tabelle 3.6 zeigt, dass die binäre Schnittholzsegmentierung mit dem KD-NN-A besser ist als mit WS-GC-R. Auch die Ergebnisse der Separierung sind mit LSGMC besser als mit der Methode in [CYP04]. Eine deutliche Verbesserung ist in der Stammanzahl zu erkennen, da durch die *Watershed*-Segmentierung auf die lokalen Maxima des Distanzbildes in [CYP04] viele kleine falsche Regionen entstehen. Zusammenfassend benötigt das Konzept weniger Restriktionen in der Bildaufnahme, segmentiert Stammschnittflächen und ist genauer als andere Methoden. Das beschriebene Konzept mit dem LSGMC und KD-NN-A wurde in [GV12] veröffentlicht.

3.5 Evaluierung und Diskussion der Konzepte

In diesem Kapitel 3 wurde die Segmentierung von Schnittholz und Stammschnittflächen betrachtet. Zur Lösung der Segmentierungsaufgabe wurden drei verschiedene Konzepte entwickelt, welche in den Unterkapiteln 3.2, 3.3 und 3.4 beschrieben wurden. Sie sind für Bilder einer Holzpolter-Frontseite (I_{wood}) konzipiert.

Das **Konzept der Schnittholzsegmentierung (WS-GC-R)** im Unterkapitel 3.2 ermöglicht die Segmentierung von Schnittholz. Auf Basis der Holzfarbe und einer Restriktion in der Bildaufnahme wird ein Vorder- und ein Hintergrundmodell (Schnittholz und Nicht-Schnittholz) erstellt. Mit den Modellen wird das Bild I_{wood} binär durch Graph-Cut segmentiert. Das Ergebnis ist das Schnittholzbild I_h^b . Das **Konzept der Stammschnittflächendetektion** im Unterkapitel 3.3 ermöglicht die Segmentierung einzelner Stammschnittflächen. Im Konzept werden Stammschnittflächen durch ein Objektmodell detektiert, verifiziert und Kreisparameter generiert. Mit den Kreisparametern wird ein Labelbild I_{st}^m berechnet, welches Hintergrund und einzelne Stammschnittflächen separiert. Dazu wurden zwei Ansätze entwickelt. Zum einen die direkte Abbildung von Kreisen (**PWL-H**) und zum anderen die Anpassung der Kreiskonturen an das Bild I_{wood} mit aktiven Konturen [KWT88]. Das **Konzept der Kombination (KD-NN-A, LSGMC)** im Unterkapitel 3.4 kombiniert die Vorteile der ersten zwei Konzepte und ermöglicht die Segmentierung einzelner Stammschnittflächen. Auf Basis einer Detektion, Verifikation und Kreisbestimmung wird eine **Trimap** erstellt. Das Bild I_{wood} wird anschließend mit Graph-Cut unter Anwendung der **Trimap** binär segmentiert. Anhand der verifizierten Kreise werden abschließend einzelne Stammschnittflächen im Binärbild separiert.

Tabelle 3.7: Die Tabelle enthält die Evaluierung aller Konzepte mit *precision*, *recall* und *f-score* der Detektion (oben) und der Schnittholzsegmentierung (Mitte) sowie HD-Maß und S-Fehler der Stammschnittflächensegmentierung (unten).

Evaluierung	Mittelwert			Standardabweichung		
Objektdetektion f_{od}	prec.	recall	f-score	prec.	recall	f-score
Detektierte Fenster W	0,876	0,653	0,7412	0,087	0,137	0,114
Verifizierte Kreise CP^v	0,901	0,649	0,748	0,078	0,138	0,115
Schnittholz f_s^h	prec.	recall	f-score	prec.	recall	f-score
WS-GC-R	0,872	0,94	0,9	0,089	0,0473	0,0518
Kreise (PWL-H) von CP^v	0,8534	0,7409	0,7881	0,0520	0,1112	0,0783
Aktive Konturen auf CP^v	0,907	0,7357	0,8070	0,045	0,115	0,083
KD-NN-A	0,878	0,95	0,91	0,08	0,0334	0,043
Stammschnittflächen f_s^{st}	HD-Maß	S-Fehler	HD-Maß	S-Fehler		
Kreise (PWL-H) von CP^v	0,8887	0,1908	0,0279	0,14		
Aktive Konturen auf CP^v	0,907	0,1908	0,0263	0,14		
Water-Dis auf KD-NN-A I_h^b	0,918	0,8046	0,03	0,7296		
LSGMC auf KD-NN-A I_h^b	0,926	0,1508	0,03	0,153		

Alle bisherigen Evaluierungen der Konzepte (siehe Abschnitt 3.2.4, 3.3.4, 3.4.4) sind in Tabelle 3.7 gegenübergestellt. Alle Evaluierungen wurden mit den 71 Evaluierungsbildern aus Anhang C.2 vorgenommen. Die Genauigkeiten der Ergebnisse sind mit den Maßen *precision*, *recall*, *f-score*, HD-Maß und S-Fehler (siehe Anhang A.4) hinterlegt. Die Tabelle unterteilt sich in drei weitere Tabellen. Die oberste Tabelle zeigt die Ergebnisse der Stammschnittflächendetektion und Verifikation. Die mittlere Tabelle enthält die Ergebnisse der binären Schnittholz- und die Untere die Ergebnisse der Stammschnittflächensegmentierung.

Aus der oberen Tabelle geht hervor, dass die verifizierten Kreise CP^v im *f-score* geringfügig besser sind als die detektierten Fenster W . Hingegen ist eine deutliche Verbesserung in der *precision* zu erkennen. Die Verifikation ist Bestandteil der Konzepte der Stammschnittflächendetektion und der Kombination. Das Ziel ist die Verringerung von falsch-positiv detektierten O^{st} . Falsch-positive O^{st} führen im ungünstigen Fall zur Segmentierung von Hintergrundbereichen als Stammschnittfläche. Auch gelangen im ungünstigen Fall Hintergrundpixel BG in das Vordergrundmodell Θ^{FG} . Die Verbesserung der *precision* ist ein gutes Zeichen und bestätigt die Notwendigkeit der Verifikation.

In der mittleren Tabelle ist die Genauigkeit der Schnittholzsegmentierung aller drei Konzepte gegenübergestellt. Das sind für das Konzept der Schnittholzsegmentierung die Ergebnisse des **WS-GC-R**, für das Konzept der Objektdetektion aktive Konturen und Kreise (**PWL-H**) und für das Konzept der Kombination die Ergebnisse der **KD-NN-A**-Segmentierung. Nur das Konzept der Schnittholzsegmentierung ist primär zur Segmentierung von Schnittholz konzipiert. Die Anderen sind zur Segmentierung einzelner Stammschnittflächen. Um eine Vergleichbarkeit zu ermöglichen, wurden alle Labelbilder I_{st}^m in den binären Fall I_h^b umgewandelt. Das heißt, alle Nicht-Hintergrundpixel ($l \neq 0$) werden als Schnittholzpixel betrachtet. Die Ergebnisse zeigen, dass mit dem Konzept der Stammschnittflächendetektion zwar Schnittholz segmentiert werden kann, aber die Segmentierung ungenauer ist als mit den anderen Konzepten. Das Konzept der Schnittholzsegmentierung ist hingegen relativ genau und das Konzept der Kombination noch genauer. Obwohl im Konzept der Schnittholzsegmentierung Restriktionen in der Bildaufnahme notwendig sind, ist das kombinierte Konzept mit weniger Restriktionen genauer. Folglich war die Kombination der Konzepte sinnvoll, und eine Verbesserung wurde in zweierlei Hinsicht erreicht.

Die untere Tabelle enthält die Ergebnisse der Stammschnittflächensegmentierung. Das bedeutet, die Ergebnisse sind Multi-Objekt-Segmentierungen. Die Konzepte der Stammschnittflächendetektion (**PWL-H**, aktive Konturen) und der Kombination (**LSGMC**) ermöglichen eine Multi-Objekt-Segmentierung. Die Ergebnisse zeigen, dass alle Konzepte relativ gut Stammschnittflächen segmentieren und das Konzept der Kombination mit dem **LSGMC**-Algorithmus dabei die besten Ergebnisse erzeugt.

Im Fazit bestätigen die Daten in der Tabelle 3.7, dass das Konzept der Kombination am besten zur Segmentierung von Stammschnittflächen geeignet ist. Das Konzept der Kombination ist im **AFoRS**-System umgesetzt und in der forstwirtschaftlichen Anwendung (weiteres siehe Kapitel 5). Die beschriebenen Konzepte wurden in [Gut+10], [Gut+11a] und [GV12] veröffentlicht. Ausgehend vom Konzept der Kombination wird im nachfolgendem Kapitel eine Generalisierung für beliebige **organo**-Gruppen erarbeitet.

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

In diesem Kapitel wird aus den Konzepten der Stammschnittflächensegmentierung in Kapitel 3 ein generalisiertes Konzept zur Segmentierung von verschiedenen **organo**-Gruppen erarbeitet. Zuerst werden in Abschnitt 4.1 die zu lösenden Probleme diskutiert und Anforderungen an die Bildaufnahme aufgestellt. Ausgehend von den Anforderungen, Problemen und den Konzepten der Stammschnittflächensegmentierung wird in Abschnitt 4.2 ein generelles Konzept abgeleitet. Aus dem Konzept ergeben sich einige Schritte, die generalisiert werden müssen. Die Generalisierungen der einzelnen Schritte werden in den Abschnitten 4.3, 4.4 und 4.5 beschrieben. In Abschnitt 4.3 wird die Objektdetektion und Verifikation, in Abschnitt 4.4 die binäre Segmentierung und in Abschnitt 4.5 die Objektseparierung generalisiert. Das erarbeitete Konzept wird im letzten Abschnitt 4.6 kurz zusammengefasst und anhand vier verschiedener **organo**-Gruppen evaluiert. Weiterhin werden die Grenzen des entwickelten Konzeptes aufgezeigt und diskutiert.

4.1 Problemdiskussion

Die Zielstellung dieses Kapitels ist die Generalisierung der bisherigen Konzepte der Schnittholz- und Stammschnittflächensegmentierung. Generalisierung heißt, dass auch andere **organo**-Gruppen automatisch segmentiert werden sollen. Es stellt sich zuerst die Frage, was kann und soll alles unter einer **organo**-Gruppe zu verstehen sein? Eine **organo**-Gruppe ist eine Gruppe von gleichartigen organischen Objekten. Zu beachten sei das Adjektiv gleichartig, welches eine gleiche Art kennzeichnet. In der biologischen Taxonomie ist eine Art (*species*) die kleinste Einheit der Klassifikation von Lebewesen. Wann man von einer Art spricht, ist allerdings keine triviale Sache und hängt von vielen Faktoren ab. Das Artenproblem ist nicht Bestandteil dieser Arbeit und kann in [Heu08] nachgelesen werden. Viel wichtiger ist im Kontext dieser Arbeit, dass die Objekte (Lebewesen) Ähnlichkeiten in Form, Farbe und Textur, aber dennoch, wie bei den Spezies einer biologischen Art, auch eine Varianz in diesen Charakteristiken aufweisen. Lebewesen oder einzelne Bestandteile einer Art wie Blütenblätter sollten in den meisten Fällen die Einschränkung erfüllen.

Der dargelegte aktuelle Stand der Forschung zur Segmentierung zeigt, dass die existierenden Ansätze sehr verschieden sind. Bisherige spezielle Methoden adaptieren und kombinieren die verschiedensten Basismethoden. Darüber hinaus sind die speziellen Methoden zumeist für spezielle **organo**-Gruppen und/oder für spezielle Rahmenbedingungen entwickelt. Bisher existiert noch keine generelle Lösung für das Problem.

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

In Abbildung 4.1 sind einzelne Objekte von drei weiteren **organo**-Gruppen abgebildet. Die drei **organo**-Gruppen repräsentieren verschiedene Arten und sind eine kleine Stichprobe der enorm großen Menge der **organo**-Gruppen. Sie werden im weiteren Verlauf näher betrachtet (siehe auch Evaluierungsbilder in Anhang C.2).



Abbildung 4.1: Die einzelnen organischen Objekte von drei weiteren **organo**-Gruppen (Plattfische, Kartoffeln und Äpfel).

Bei der Segmentierung von Stammschnittflächen ergaben sich die folgenden Probleme:

- Objekte variierender Form, Farbe und Textur
- Hintergrund (Nicht-Objektpixel) unbekannter Farbe und Textur
- Varianzen in Farbe, Form und Textur innerhalb der Objektklasse
- Varianzen in Farbe und Textur im Hintergrund
- Überschneidung der Farbbereiche vom Hintergrund mit den Objekten

Die Probleme bleiben auch bei der Segmentierung von anderen **organo**-Gruppen bestehen. Allerdings können aufgrund der Unterschiedlichkeit der **organo**-Gruppen weniger Annahmen über die Bildobjekte getroffen werden. Bei der Stammschnittflächensegmentierung wurde die Annahme getroffen, dass die Stammschnittflächen O^{st} im Bildraum ein Cluster bilden. Bei **organo**-Gruppen ist dies nicht möglich, da zum Beispiel Kartoffeln über einem Feld verstreut liegen können. Weiterhin wurde bei der Schnittholzsegmentierung in der Vorsegmentierung eine Farbheuristik angewandt. Eine generelle Farbheuristik für alle **organo**-Gruppen ist nicht bestimmbar, da die Menge der **organo**-Gruppen in der Farbe zu verschieden ist. Die Bestimmung einer Farbheuristik für eine einzelne **organo**-Gruppe ist möglich, aber sehr aufwändig. Weiterhin sind im idealen Fall nur einige Bildobjekte O^o kreisförmig oder elliptisch. Im Fazit ergeben sich kaum Annahmen für OG .

Um dennoch eine Lösung der Segmentierungsaufgabe durch Generalisierung der bisherigen Konzepte zu finden, sind Restriktionen in der Bildaufnahme notwendig. Im

4.2 Ableitung eines allgemeingültigen Konzeptes

Fall der Stammschnittflächen lagen alle OG^{st} auf einer Ebene. Auch gab es kaum Überschneidungen und Stammschnittflächen sind im idealen Fall rundlich. Im Weiteren soll deshalb von folgenden Restriktionen ausgegangen werden:

- Mehrere Bildobjekte OG einer **organo**-Gruppe in einem Bild
- OG überschneiden sich wenig bis gar nicht
- OG liegen ungefähr auf einer Ebene, die orthogonal zur Kamera ausgerichtet ist
- Mehrere ähnliche organische Objekte in einem Bild
- Bildobjekte haben ähnliche Eigenschaften in Größe, Form, Farbe und Textur
- Bildobjekte sind quasi-rund, wobei die Überdeckung des besten angepassten Kreises mit identischem Flächeninhalt $>60\%$ ist

4.2 Ableitung eines allgemeingültigen Konzeptes

In der Evaluierung führte das Konzept der Kombination zu den besten Ergebnissen. Das Konzept ermöglicht eine Multi-Objekt-Segmentierung und soll der Generalisierung zu Grunde gelegt werden. Eine entsprechende Generalisierung ist in Abbildung 4.2 zu sehen.

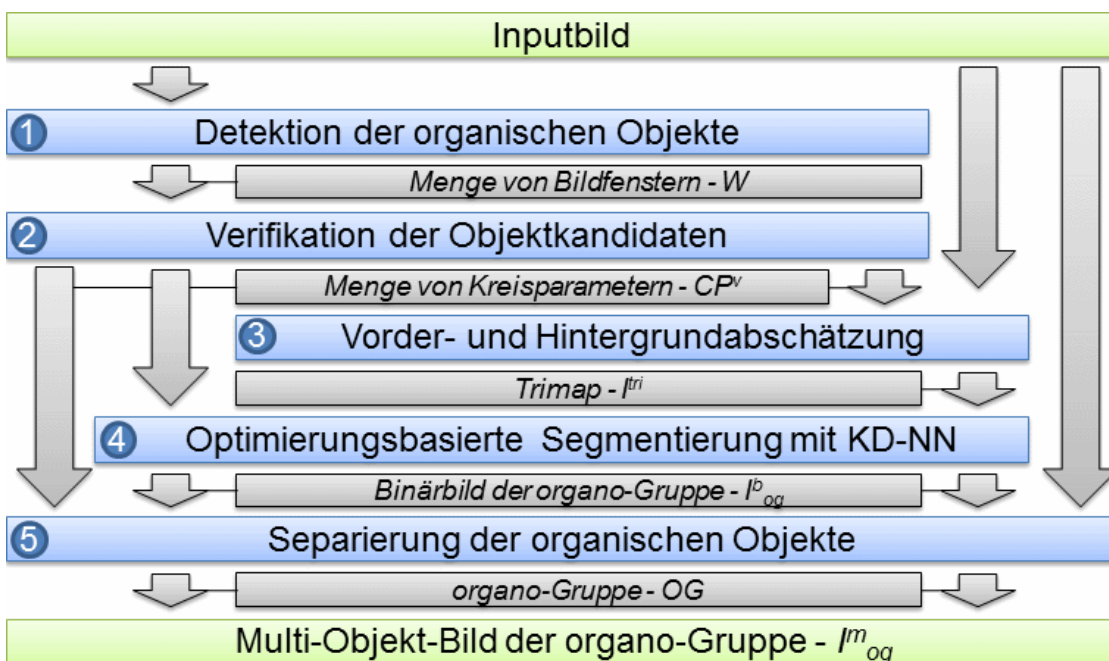


Abbildung 4.2: Die Generalisierung des Konzeptes der Kombination zur Segmentierung von **organo**-Gruppen.

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

Das Konzept in Abbildung 4.2 beinhaltet fünf Schritte. Der erste Schritt ist identisch mit dem Konzept der Kombination und detektiert organische Objekte OG in Form von Objektfenstern W . Im Konzept der Stammschnittflächendetektion haben sich Haarkaskaden als angebracht erwiesen. Die Eignung zur Detektion von beliebigen **organo**-Gruppen muss erneut untersucht und geprüft werden. Der zweite Schritt hat zum Ziel, die Objektfenster zu verifizieren und falsch-positiv detektierte Fenster W zu entfernen. Im Unterschied zu den Stammschnittflächen ist eine Beseitigung von Ausreißern durch Clustering der Bildobjekte im Bildraum nicht sinnvoll, da die Bildobjekte O^o auch verstreut im Bild liegen können. An dieser Stelle muss eine allgemeinere Methode gefunden werden. Das Ergebnis des Schrittes sind wieder die Kreisparameter CP^v . Durch CP^v ergibt sich als weitere Annahme die Einschränkung der **organo**-Gruppen auf **quasi-runde** Bildobjekte (Überdeckung zum besten angepassten Kreis mit identischem Flächeninhalt $>60\%$).

Im dritten Schritt wird eine **Trimap** I^{tri} erstellt, welche grob Vorder- und Hintergrund abschätzt. Das Problem im dritten Schritt ist, dass zur Abschätzung von I^{tri} keine Farb- oder Positionsheuristiken verwendet werden können. Im vierten Schritt werden Bildobjekt- (O^o) und Nicht-Bildobjektpixel (\bar{O}^o) binär segmentiert. Die binäre Schnittholzsegmentierung mit Graph-Cut auf Basis von I^{tri} hat sich als relativ genau herausgestellt. Graph-Cut ist eine optimierungsbasierte Segmentierungsmethode. Außer Graph-Cut gibt es noch weitere optimierungsbasierte Methoden, die theoretisch ebenso geeignet sind und entsprechend betrachtet werden sollen. Der **KD-NN**-Ansatz zur Bestimmung der Wahrscheinlichkeitskarten I^p hat sich im Fall der Schnittholzsegmentierung als beste Methode herausgestellt. Aus diesem Grund soll der **KD-NN**-Ansatz in der Generalisierung weiter verwendet werden.

Im fünften und letzten Schritt werden die einzelnen Objekte voneinander getrennt. Zur Stammschnittflächensegmentierung wurde speziell der **LSGMC**-Algorithmus für das Problem entwickelt. Der Algorithmus soll entsprechend generalisiert werden. Im Gegensatz zu den Stammschnittflächen O^{st} sind beliebige O^o in der Form unbekannt und weniger rundlich. Aus diesem Grund wird zur Separierung auch das Inputbild I_{og} einbezogen.

Aus dem generellen Konzept ergeben sich die folgenden **Fragestellungen**:

- Sind Haarkaskaden auch zur Detektion beliebiger **organo**-Gruppen geeignet?
- Wie lässt sich die Objektverifikation generalisieren?
- Wie kann die Vorder- und Hintergrundabschätzung mit CP^v generalisiert werden?
- Welche Methode der optimierungsbasierten Segmentierung ist geeignet?
- Wie können beliebige organische Objekte O^o einzeln segmentiert werden?

In den folgenden Abschnitten werden die einzelnen Fragestellungen aufgefasst und entsprechende Lösungen erarbeitet. In Abschnitt 4.3 wird die Objektdetektion und Verifikation untersucht. In Abschnitt 4.4 wird die optimierungsbasierte binäre Segmentierung mit **KD-NN** generalisiert und in Abschnitt 4.5 die Objektseparierung. Zum Abschluss wird in Abschnitt 4.6 das generalisierte Konzept zusammengefasst und evaluiert und die Grenzen werden aufgezeigt und diskutiert.

Das generelle Konzept mit Problembetrachtung, Diskussion der Lösungsmöglichkeiten und exemplarischen Beispielen (Äpfel am Boden, Äpfel am Baum, Hühnern und Fischen unter Wasser) ist in [GL13] veröffentlicht.

4.3 Generalisierung der Objektdetektion und Verifikation

Im Abschnitt 3.3.2 wurden verschiedene Methoden der Objektdetektion untersucht. Zur Detektion von Stammschnittflächen haben sich Haarkaskaden [VJ01] als angebracht herausgestellt. Sie sind schnell, relativ genau, haben kaum Parameter und ein erstes Objektmodell Θ^o zur Detektion ist einfach mit openCV [Lag11] generierbar. Aus den benannten Gründen soll zuerst die Eignung von Haarkaskaden zur Detektion von O^o untersucht werden.

Zur Prüfung der Eignung wurde je ein Detektor für Äpfel, Kartoffeln und Plattfische antrainiert. In Tabelle 4.1 sind die Daten der Modellerstellung (Θ^o) und die Genauigkeiten der Detektion aufgelistet. Die verschiedenen Objektmodelle Θ^o und detektierten Fenster W werden im Weiteren spezialisiert durch:

- Äpfel (*apples*): Objektmodell Θ_a^o , Objektfenster W_a
- Kartoffeln (*potatoes*): Objektmodell Θ_p^o , Objektfenster W_p
- Plattfische (*flatfishes*): Objektmodell Θ_f^o , Objektfenster W_f
- Stammschnittflächen (*wood*): Objektmodell Θ_w^o , Objektfenster W_w

Ein Modell der Stammschnittflächen¹ (Θ_w^o) wurde bereits betrachtet und ist der Vollständigkeit halber mitaufgelistet. Die Spalte **K** kennzeichnet die Anzahl der Kaskaden, die Spalte **pos.** die Anzahl positiver und **neg.** die Anzahl negativer Trainingsbilder. Es wurde zum einen der Mittelwert und zum anderen die Standardabweichung von *precision*, *recall* und *f-score* der Objektdetektion ermittelt. Der Mittelwert und die Standardabweichung der Maße wurden über eine Menge von Evaluierungsbildern pro Gruppe vorgenommen (siehe Anhang C.2). Die Anzahl der Evaluierungsbilder ist jeweils in der Spalte **GT** aufgelistet.

Tabelle 4.1: Der Mittelwert und die Standardabweichung von *precision*, *recall* und *f-score* der detektierten Fenster W auf Basis der jeweiligen **GT**-Bilder.

	Modellerstellung			Mittelwert			Standardabweichung			Eval.
	K	pos.	neg.	prec.	recall	f-score	prec.	recall	f-score	GT
Θ_w^o	19	5729	3019	0,818	0,691	0,745	0,104	0,113	0,099	71
Θ_a^o	20	171	301	0,975	0,955	0,964	0,043	0,058	0,046	25
Θ_p^o	17	159	472	0,701	0,597	0,601	0,182	0,275	0,206	25
Θ_f^o	20	197	482	0,943	0,659	0,760	0,111	0,209	0,166	25

¹Die Ergebnisse der Detektion von Θ_w^o weichen etwas von den Ergebnissen in Abschnitt 3.3.2 ab. Der Grund liegt in der Verwendung eines sensibleren und allgemeineren Skalierungsfaktors der Fenster.

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

Die Ergebnisse in der Tabelle 4.1 zeigen, dass die Detektion mit Haarkaskaden bereits mit wenigen positiven und negativen Trainingsbildern zu relativ guten Ergebnissen führt. Besonders Äpfel werden sehr genau detektiert ($f\text{-score}=0,964$). Verschiedene Äpfel sind sich *i.d.R.* sehr ähnlich. Sie sind rund und haben zumeist eine rote, grüne und/oder gelbe Farbe. Im Gegensatz dazu ist die Detektion von Kartoffeln und Plattfischen ungenauer. Kartoffeln sind nur im idealen Fall rund und Plattfische sind sehr verschieden in der Form. Auch haben Plattfische auf der Ober- und Unterseite unterschiedliche Farb- und Textureigenschaften. Aufsteigend geordnet nach dem Schwierigkeitsgrad ergibt sich die Reihenfolge: Äpfel, Kartoffeln und Plattfische.

Zusammenfassend sind die Ergebnisse als Grundlage der Generalisierung vollkommen ausreichend. Im Vergleich zu den Stammschnittflächen Θ_w^O sind die Ergebnisse ähnlich und in vielen Fällen sogar besser. Einige Ergebnisbilder inklusive den detektierten Fenstern W der verschiedenen *organo*-Gruppen sind in Abbildung 4.3 zu sehen.



Abbildung 4.3: Von links nach rechts sind die detektierten Fenster W mit den Objektmodellen Θ_w^O , Θ_f^O , Θ_p^O und Θ_a^O dargestellt.

Die detektierten Fenster W sind zwar relativ genau, aber dennoch sind Verbesserungen durch eine Verifikation möglich und angebracht. Besonders falsch-positiv detektierte Fenster sind eliminierbar. Eine Verringerung ist notwendig, da in den nachfolgenden Schritten anhand der Ergebnisse ein Vordergrundmodell Θ^{FG} erstellt wird. Das Vordergrundmodell Θ^{FG} wird ungenauer, je mehr falsch-positiv detektierte Fenster zu Grunde gelegt werden. Im Abschnitt 3.3.3 wurden dazu zwei Methoden der Verifikation entwickelt, die Beseitigung von Überlappungen und Ausreißern im Bildraum. Nur die Beseitigung von Überlappungen ist auch auf andere Bildobjekte O^o übertragbar, da nicht von einer Clusterbildung von O^o im Bildraum ausgegangen werden kann. Die Konvertierung der Fenster W zu Kreisparametern CP ist auch in der Generalisierung sinnvoll. Eine entsprechende Konvertierung soll vor der Verifikation erfolgen. Ein Kreis beschreibt ein Objekt genauer als ein Rechteck und ist besser algorithmisch anzuwenden, da ausgehend vom Detektionszentrum der Abstand zum Segmentrand (Radius) identisch ist.

Eine weitere Möglichkeit zur Verringerung der falsch-positiv detektierten Fenster W ist die Nutzung von Bildobjektmerkmalen und ein anschließendes *Clustering* ähnlicher

4.3 Generalisierung der Objektdetektion und Verifikation

Bildobjekte. Es gibt eine Menge von Bildmerkmalen (siehe [Jah05] und [Ton05]), wie die Merkmale der **GLCM** oder diverse Farbverteilungen. Theoretisch sind viele Farb- oder Texturmerkmale adaptierbar. An dieser Stelle soll nur ein Beispiel und eine Methodik der Beseitigung aufgezeigt und untersucht werden. Speziell wird von der Annahme ausgegangen, dass die durchschnittliche Farbe innerhalb eines Kreises CP ähnlich ist. Zur Verifikation anhand der Annahme sind ein Bildmerkmal und eine Metrik notwendig. Die Bildobjekte werden durch die Kreisparameter CP beschrieben. Für jeden Kreisparameter $cp \in CP$ wird zuerst die durchschnittliche Farbe (*mean*) $px^m = (r^m, g^m, b^m)^T$ eines Pixels $px = (r, g, b)^T$ im **RGB**-Farbraum bestimmt. px^m errechnet sich aus allen Pixeln innerhalb des herunterskalierten Kreises von cp . Eine Skalierung des Radius $s * r$ von cp um einen Faktor $s \in [0.5, 0.75]$ ist notwendig, da Objektgrenzen innerhalb des Kreises den Mittelwert verfälschen. Anschließend wird der Mittelwert \bar{px} und die Varianz σ_{px} über alle Mittelwerte px^m mit dem Euklidischen Abstand bestimmt durch:

$$\bar{px} = \frac{1}{n} \sum_{i=1}^n px_i^m; \sigma_{px} = \frac{1}{n} \sum_{i=1}^n \|\bar{px} - px_i^m\|; n = |CP| \quad (4.1)$$

Zu jedem $cp \in CP$ korrespondiert ein Mittelwert px^m . Der Index i indexiert die entsprechenden Mittelwerte. Ist die Varianz σ_{px} größer als ein gewisser Schwellwert T , so sind nicht alle Kreise ähnlich in der Farbe und enthalten Ausreißer. Für T hat sich ein Wert von 20 als sinnvoll erwiesen. Ein Ausreißer px^m hat die größte Euklidische Distanz zu \bar{px} und wird entfernt. Der Prozess der Mittelwert- und Varianzbestimmung wird über alle verbleibenden Kreise wiederholt, bis alle Kreise in der Farbe ähnlich sind ($\sigma_{px} < T$). Das Ergebnis der farbbasierten Ausreißerentfernung und der Auflösung von Überdeckungen wird durch die Menge der verifizierten Kreise CP^v repräsentiert. In Abbildung 4.4 sind die Ergebnisse dargestellt.



Abbildung 4.4: Von links nach rechts sind die verifizierten Kreisparameter CP^v der detektierten Fenster W_w , W_f , W_p und W_a dargestellt.

Eine quantitative Evaluierung und Gegenüberstellung der detektierten Fenster W und verifizierten Kreise CP^v der beschriebenen Methodik ist in Tabelle 4.2 abgebildet. Die Genauigkeit der verifizierten Kreise wurde mit der gleichen Methodik wie in Tabelle 4.1 bestimmt und den detektierten Fenstern W gegenübergestellt.

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

Tabelle 4.2: Der Mittelwert und die Standardabweichung von *precision*, *recall* und *f-score* der detektierten Fenster W und verifizierten Kreisparameter CP^v auf Basis der jeweiligen **GT**-Bildern.

Evaluierung	Mittelwert			Standardabweichung		
	prec.	recall	f-score	prec.	recall	f-score
detektierte Fenster W_w	0,818	0,691	0,745	0,104	0,113	0,099
verifizierte Kreise CP_w^v	0,867	0,680	0,758	0,071	0,114	0,093
detektierte Fenster W_a	0,975	0,955	0,964	0,043	0,058	0,046
verifizierte Kreise CP_a^v	0,976	0,942	0,958	0,044	0,061	0,048
detektierte Fenster W_p	0,701	0,597	0,601	0,182	0,275	0,206
verifizierte Kreise CP_p^v	0,884	0,57	0,652	0,2	0,271	0,24
detektierte Fenster W_f	0,943	0,659	0,760	0,111	0,209	0,166
verifizierte Kreise CP_f^v	0,967	0,654	0,765	0,093	0,213	0,173

Die Tabelle 4.2 zeigt, dass die *precision* nach der Verifikation bei allen **organo**-Gruppen besser wird. Eine Verbesserung der *precision* bedeutet, dass weniger falsch-positive Elemente enthalten sind. Im Fall der Stammschnittflächen CP_w^v , der Kartoffeln CP_p^v und der Plattfische CP_f^v wird auch der *f-score* besser. Nur bei den Äpfeln ist eine geringfügige Verschlechterung vom *f-score* zu sehen. Der *recall* hingegen verschlechtert sich bei allen **organo**-Gruppen geringfügig. Der Grund ist, dass in einigen Fällen auch richtig-positive Elemente entfernt werden und folglich die Anzahl der falsch-negativen Elemente steigt. Im Fazit verschlechtert sich zwar der *recall*, aber die *precision* wird umso besser. Es gelangen folglich weniger falsche Bildobjekte in das Vordergrundmodell Θ^{FG} , wodurch die Verifikation mit der beschriebenen Methodik ein sinnvoller Schritt ist.

4.4 Generalisierung der binären Segmentierung

Die binäre Segmentierung zerlegt das Bild I_{og} in Objekt- und Nicht-Objekt-Pixel (O^o , $\overline{O^o}$). Sie besteht aus zwei prinzipiellen Schritten, der **Trimap**-Erstellung und der optimierungsbasierten Segmentierung. Im ersten Schritt wird eine **Trimap** I^{tri} erstellt. Auf Basis von I^{tri} werden im zweiten Schritt Wahrscheinlichkeitsbilder (I_{fg}^p , I_{bg}^p) vom Vorder- und Hintergrund erstellt und das Bild wird mit einer optimierungsbasierten Basismethode segmentiert.

Erstellung einer generellen **Trimap** I^{tri}

Im Konzept der Kombination wurde eine **Trimap** I^{tri} eingeführt. I^{tri} kennzeichnet ein Pixel als unbekannt ($l = unknown$ oder kurz $l = u$), Vorder- ($l = fg$) oder Hintergrund ($l = bg$). Die **Trimap** markiert die abgeschätzten Vorder- (FG^{smp}) und Hintergrundpixel (BG^{smp}) für die nachfolgende optimierungsbasierte Segmentierung. Im Konzept der Kombination wurde zur Abschätzung von I^{tri} auf Basis von CP^v zuerst der mittlere Schnittholzbereich im Bild ermittelt und anschließend segmentiert. Beides ist in einem

generellen Ansatz nicht möglich, da die Objekte O^o kein Cluster im Bildraum bilden und keine Farbheuristiken zur Segmentierung bekannt sind. Es sind nur die verifizierten Kreisparameter CP^v bekannt und dass es eine **organo**-Gruppe ist.

Eine Lösung ist, die Abschätzung der Pixelmengen (FG^{smp} , BG^{smp}) unter der Annahme vorzunehmen, dass ein Kreis $c \in CP^v$ ein Objekt O^o zum größten Teil überlagert. Ausgehend von der Annahme wird zuerst eine temporäre **Trimap** I_{tmp}^{tri} mit der Funktion f^{tri} erstellt durch:

$$f^{tri}(x, y, CP^v, a) = \begin{cases} fg, & \exists(x_c, y_c, r) \in CP^v : \sqrt{(x_c - x)^2 + (y_c - y)^2} < ar \\ u, & \exists(x_c, y_c, r) \in CP^v : \sqrt{(x_c - x)^2 + (y_c - y)^2} \in [ar, r] \\ bg, & \text{sonst} \end{cases} \quad (4.2)$$

Die Argumente von f^{tri} sind die Pixelposition $(x, y)^T$, die Menge der verifizierten Kreise CP^v und ein Skalierungsfaktor a . Ein Wert von 0,7 ist ein guter Skalierungsfaktor a . Die Funktion bildet alle Pixel außerhalb der Kreise CP^v auf bg und die Pixel innerhalb des, um a skalierten, Radius r von CP^v auf fg ab. Übrig bleiben die äußeren Ringe von CP^v , welche auf u abgebildet werden. Die ringförmigen Bereiche ($l = u$) werden auf unbekannt gesetzt, weil die Konturen von O^o noch nicht bekannt sind. Die resultierende **Trimap** I_{tmp}^{tri} markiert unterschiedlich viele Vorder- (FG^{tmp}) und Hintergrundpixel (BG^{tmp}). Die markierten Pixelmengen (FG^{tmp} , BG^{tmp}) sind nicht gleich groß und können falsch markierte Pixel enthalten. Es gilt **i.d.R.** $|FG^{tmp}| \ll |BG^{tmp}|$. In den meisten Fällen werden nicht detektierte Objekte als Hintergrund markiert. Um den Einfluss von falsch klassifizierten Pixeln abzuschwächen, werden zufällige Stichproben (*random sampling*) über die Mengen erhoben. Das Ergebnis sind zwei Teilmengen gleicher Mächtigkeit (FG^{smp} , BG^{smp}). Die Mächtigkeit ist relativ zur Weite w und Höhe h von I_{og} und nicht größer als $|FG^{tmp}|$ und $|BG^{tmp}|$. Sie ergibt sich durch:

$$\min(w * h * b, \min(|FG^{tmp}|, |BG^{tmp}|)) = |FG^{smp}| = |BG^{smp}| \\ FG^{smp} \subseteq FG^{tmp}; BG^{smp} \subseteq BG^{tmp}; b \in [0, 1]$$

Ein geeigneter Wert für den Skalierungsfaktor b ist 0,05.

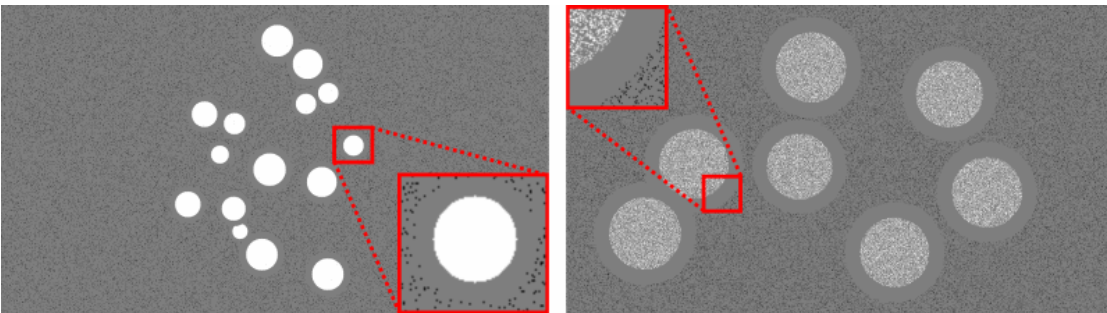


Abbildung 4.5: Beispiele von I^{tri} mit $l = fg$ (weiß), $l = u$ (grau) und $l = bg$ (schwarz).

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

Die korrespondierende **Trimap** zu FG^{smp} und BG^{smp} ist die Basis für die optimierungsbasierte Segmentierung. Zwei Beispiele von I^{tri} sind in Abbildung 4.5 zu sehen.

Adaption optimierungsbasierter Segmentierungen

Optimierungsbasierte Basismethoden segmentieren ein Bild anhand von Wahrscheinlichkeitsbildern I^p oder Vorsegmentierungen. Im Kontext der Schnittholzsegmentierung wurde bereits Graph-Cut untersucht und adaptiert. Ausgehend von den abgeschätzten Pixeln FG^{smp} und BG^{smp} wurden mit verschiedenen Modellen Wahrscheinlichkeitsbilder I^p erstellt. Die besten Ergebnisse wurden mit dem Modell Θ_{kn} erzielt, dem **KD-NN**-Ansatz. Aufgrund der guten Ergebnisse soll **KD-NN** die Basis bilden und generell zur Erstellung von I_{fg}^p und I_{bg}^p verwendet werden. Die im Vorschritt ermittelte **Trimap** I^{tri} markiert bereits, die für **KD-NN** benötigten Pixelmengen, FG^{smp} und BG^{smp} . In Abbildung 4.6 sind zwei Ergebnisse von I_{fg}^p und I_{bg}^p mit **KD-NN** zu sehen.

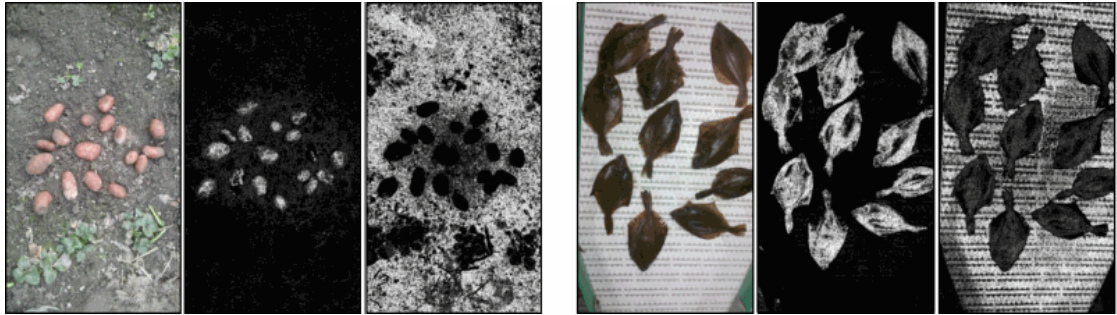


Abbildung 4.6: In der Abbildung sind I_{og} (links), I_{fg}^p (Mitte) und I_{bg}^p (rechts) von Kartoffeln und Plattfischen zu sehen. In der Visualisierung ist $p \in I^p$ auf den Grauwert abgebildet (1 auf weiß, 0 auf schwarz).

Theoretisch lassen sich alle optimierungsbasierten Methoden zur Segmentierung adaptieren. Allerdings sind ausgehend von den berechneten Wahrscheinlichkeitsbildern I^p entsprechende approximierete Schlussfolgerungen (*probabilistic inference*) auf **MRF** sinnvoll. Für die Problemstellung haben sich in der Forschung Graph-Cut (**GC**) und Belief-Propagation (**BP**) etabliert. **GC** wurde zur Segmentierung von OG^{st} bereits adaptiert und soll auch zur Segmentierung von OG in gleicher Weise verwendet werden (siehe Abschnitt 3.2.3). **BP** wurde hingegen noch nicht adaptiert, da **GC** bereits gute Ergebnisse erzielte und eines der besten Verfahren ist. Dennoch soll der Vollständigkeit halber zusätzlich zu **GC** auch **BP** untersucht, adaptiert und evaluiert werden.

Zur Untersuchung der Eignung von **BP** soll der (*loopy max-product*) **BP**-Algorithmus von [FH06] adaptiert werden. Der Algorithmus tauscht Nachrichten μ innerhalb eines über das Bild I aufgespannten Graphen aus (Bild-Graph). Im Fall von I_{og} wird von einer Vierer-Nachbarschaft ausgegangen. Das heißt, der Graph verbindet vertikal und horizontal benachbarte Pixel miteinander. Eine Nachricht von einem Knoten i zu einem verbundenen Knoten j wird mit $\mu_{i \rightarrow j}^k$ bezeichnet. Der Index k kennzeichnet die Iteration.

Im Fall einer binären Segmentierung gibt es nur zwei mögliche Labels $L = \{0, 1\}$. Folglich ist die Nachricht $\mu = (p^{fg}, p^{bg})^T$ zweidimensional. p^{fg} ist die Wahrscheinlichkeit, dass das Pixel Vordergrund ist ($l = 1$). p^{bg} ist die Wahrscheinlichkeit, dass das Pixel Hintergrund ist ($l = 0$). Eine Nachricht errechnet sich nach jeder Iteration (k) durch die Nachrichten der Iteration zuvor ($k - 1$) durch:

$$\mu_{j \rightarrow i}^k(l_i) = \min_{l_i} \left(c * \psi(l_j, l_i) + c * \phi(l_i) + \sum_{s \in N(j) \setminus i} \mu_{s \rightarrow j}^{k-1}(l_j) \right) \quad (4.3)$$

Die Formel 4.3 ist eine erweiterte Form aus [FH06]. Die Formel wurde um einen Skalierungsfaktor ($c = 0,9$) erweitert, um eine Konvergenz zu erzwingen und die Startzustände (Labels) höher zu gewichten. $l_i, l_j \in L$ sind die Labels der Knoten i und j . ψ kennzeichnet die Diskontinuitätskosten (*discontinuity cost*) und ϕ die Anpassungskosten (*matching cost*). Als Kostenfunktionen ψ und ϕ werden

$$\psi(l_i, l_j) = \begin{cases} 0, & l_i = l_j \\ 1, & l_i \neq l_j \end{cases}; \quad \phi(l_i) = \begin{cases} 1 - p_i^{fg}, & l_i = 1 \\ 1 - p_i^{bg}, & l_i = 0 \end{cases}; \quad p_i^{fg} \in I_{fg}^p; \quad p_i^{bg} \in I_{bg}^p, \quad (4.4)$$

verwendet. Die Indizes i, j indexieren die Knoten des Bild-Graphen und die korrespondierenden Wahrscheinlichkeiten in I^p . Die Funktion ψ gibt 1 zurück, wenn zwei benachbarte Labels ungleich sind, und 0, wenn die Labels gleich sind. Die Kosten sind folglich bei gleichen Labels geringer. Die Funktion ϕ gibt die korrespondierende inverse Wahrscheinlichkeit $1 - p$ des Labels zurück. p wird invertiert, weil die Formel 4.3 minimiert und somit 0 der wahrscheinlichste Wert für ein Label l ist. Der finale *belief* und das entsprechende Ergebnisbild I_{og}^b wird wie in [FH06] berechnet. In Abbildung 4.7 sind einige Ergebnisse mit **GC** und **BP** zu sehen. Die besten Ergebnisse werden zumeist mit **GC** erzeugt.

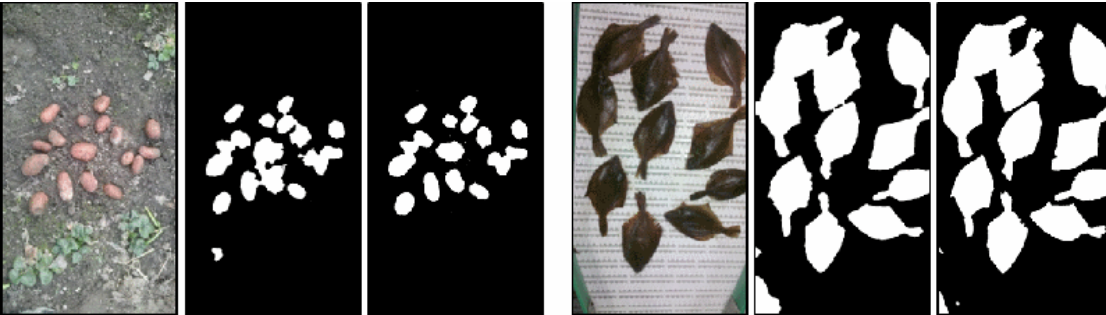


Abbildung 4.7: In der Abbildung sind I_{og} (links), I_{og}^b mit **BP** (Mitte) und I_{og}^b mit **GC** (rechts) von Kartoffeln und Plattfischen zu sehen.

Zusammenfassend wurde in diesem Abschnitt eine generelle Methodik zur binären Segmentierung von I_{og} anhand von verifizierten Kreisen CP^v mit optimierungsbasierten Methoden erarbeitet. Der beschriebene Ansatz der Trimap-Erstellung inklusive einer optimierungsbasierten Segmentierung generalisiert den **KD-NN-A** und soll mit **KD-NN-G** (*KD-NN-generalized*) bezeichnet werden.

4.5 Generalisierung der Objektseparierung

Die Objektseparierung segmentiert einzelne Bildobjekte O^o anhand der binären Segmentierung I_{og}^b , der verifizierten Kreise CP^v und dem Inputbild I_{og} . Im Unterschied zum Konzept der Kombination der Stammschnittflächensegmentierung sollen nicht nur die Objekte in I_{og}^b durch CP^v separiert werden, sondern beide Zwischenergebnisse zur Multi-Objekt-Segmentierung von I_{og} angewandt werden. Eine geeignete Basismethode zur Multi-Objekt-Segmentierung ist die *Watershed*-Segmentierung (siehe [BL79], [Beu91], [RM00]). Durch Ansteuerung einer *Watershed*-Segmentierung mit CP^v und I_{og}^b sollten sich theoretisch auch Objekte verschiedenster Form im Bild I_{og}^b entsprechend gut segmentieren lassen.

Die Basis der generellen Objektseparierung bildet der im Abschnitt 3.4.3 entwickelte **LSGMC**-Algorithmus zur Separierung von Stammschnittflächen. Das Hauptproblem des Algorithmus ist, dass nicht rundliche Objekte im schlechtesten Fall in mehrere Bestandteile zerlegt werden. Es kommt folglich zur Übersegmentierung. Zur Vermeidung einer Übersegmentierung werden zusätzlich Statistiken erster Ordnung mit CP^v über die Objekte O^o abgeschätzt. Es muss an dieser Stelle das Wort Abschätzung gebraucht werden, weil nicht von fehlerfreien Kreisparametern CP^v ausgegangen werden kann. Die Abschätzung besteht aus der Berechnung des Mittelwerts \bar{r} und der Standardabweichung σ_r der Radien in CP^v .

Die Separierung der Objekte auf Basis der Statistiken erfolgt in mehreren aufeinanderfolgenden Schritten. Zuerst werden, wie auch im **LSGMC**-Algorithmus, die Kreisparameter CP^v durch den **GMC**-Algorithmus in das Binärbild I_{og}^b eingepasst. Das Ergebnis ist die Menge von angepassten Kreisparametern CP^a . Danach werden alle Pixel in I_{og}^b innerhalb von CP^a auf $l = 0$ gesetzt. Anschließend wird eine Distanztransformation auf das Ergebnisbild I_{rest}^b angewandt und das Distanzbild segmentiert mit dem Schwellwert:

$$T = \max\left(\bar{r} - 3\sigma_r, \frac{\bar{r}}{2}\right) \quad (4.5)$$

Der Schwellwert T basiert auf den abgeschätzten Statistiken und führt zu einem Binärbild I_{rest}^b mit mehreren Segmenten S_{rest} . Ein Segment S_{rest} repräsentiert, wenn die binäre Segmentierung zuvor entsprechend genau ist, **i.d.R.** ein nicht detektiertes Objekt. Für alle Segmente S_{rest} wird der Radius r_s und der Schwerpunkt $mc = (x_m, y_m)^T$ ermittelt (siehe Schwerpunktberechnung im Anhang A.1). Der Radius eines Segmentes ergibt sich aus der Pixelanzahl S_{rest} , welche analog die Segmentfläche widerspiegelt, durch:

$$r_s = \sqrt{\frac{|S_{rest}|}{\pi}} \quad (4.6)$$

Der Schwerpunkt mc zusammen mit dem Radius r_s ergibt die Kreisparameter $c_{rest} = (x_m, y_m, r_s)$. Alle Kreisparameter c_{rest} werden mit CP^a vereinigt. Das Ergebnis ist die erweiterte Menge von Kreisparametern CP^e . Die erweiterte Menge CP^e wird erneut mit dem **GMC** in das Binärbild I_{og}^b eingepasst. Es ergeben sich die angepassten und erweiterten Kreisparameter CP^{ae} , welche im idealen Fall die Bildobjekte O^o entsprechend gut markieren.

Im letzten Schritt wird mit den erweiterten Kreisparametern CP^{ae} eine saatkpunkt-basierte *Watershed*-Segmentierung des Bildes I_{og} durchgeführt. Eine saatkpunkt-basierte *Watershed*-Segmentierung² zerlegt ausgehend von Saatkpunkten ein Bild in Segmente. Ausgehend von den Saatkpunkten und der Betrachtung des Bildes als Grauwert-Gebirge wird das Bild mit Wasser geflutet. Stoßen während des Flutens zwei Seen aneinander, entstehen Segmentgrenzen (Wasserscheiden). Da die Segmentgrenzen **i.d.R.** an Bildkanten entstehen, ist die Segmentierung mit *Watershed* eine adäquate Möglichkeit, die Konturen der Bildobjekte O^o zu finden **bzw.** in Bezug zu I_{og}^b zu verbessern. Es muss allerdings beachtet werden, dass sich Wasserscheiden auch innerhalb texturierter Bereiche bilden können. Aus diesem Grund ist es angebracht, den Flutbereich so gering wie möglich zu halten. Der Flutbereich wird durch alle Pixel von I_{og} , die keine Saatkpunkte sind, gebildet. Eine Begrenzung des Flutbereiches ist durch Bestimmung adäquater Saatkpunkte (Labels) mit I_{og}^b und CP^{ae} möglich.

Saatkpunkte entsprechen vordefinierten Labels im Ergebnisbild I_{og}^m . Im Kontext der generellen Objektseparierung werden genau $|C^{ae}| + 2$ Labels vordefiniert. Zuerst wird für jedes Objektpixel ($l = 1$) mit der Position x und y in I_{og}^b ein Label in I_{og}^m mit der Funktion

$$f_{s|od}(x, y, a) = \begin{cases} i, & \exists (x_c, y_c, r)_i \in CP^{ae} : i \in \Lambda \wedge \sqrt{(x_c - x)^2 + (y_c - y)^2} < a * r \\ -1, & \text{sonst} \end{cases} \quad (4.7)$$

zugeordnet. Dabei entspricht Λ der Indexmenge und $a \in [0, 1]$ einem Skalierungsfaktor. Ein geeigneter Skalierungsfaktor ist 0,8. Die Funktion $f_{s|od}$ setzt ein Label innerhalb des herunterskalierten Kreises CP^{ae} auf den Index i des jeweiligen Kreises. Alle anderen Labels werden auf unbekannt (-1) gesetzt. Weiterhin werden alle Hintergrundpixel ($l = 0$) in I_{og}^b , die mehr als σ_r Pixel vom dichtesten Objektpixel ($l = 1$) entfernt sind, in I_{og}^m auf Hintergrund ($l = 0$) gesetzt. Letztendlich wird die *Watershed*-Segmentierung auf I_{og}^m durchgeführt. In Abbildung 4.8 ist das beschriebene Vorgehen illustriert.

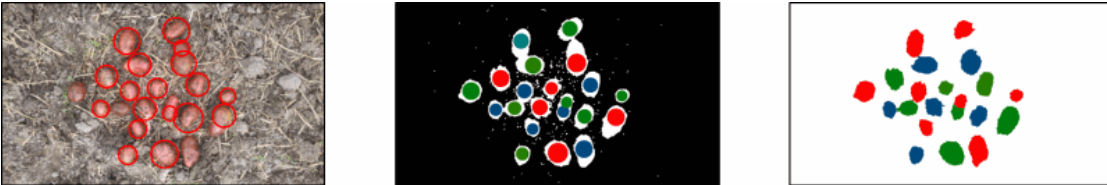


Abbildung 4.8: Die Abbildung zeigt CP^v in I_{og} (links), die vordefinierten Labels in I_{og}^m und das Ergebnis I_{og}^m . Die Objekt-Labels sind mit drei Farben markiert. Im mittleren Bild kodiert Weiß das Label unbekannt (-1).

Das beschriebene Verfahren generalisiert den **LSGMC**-Algorithmus und wird im Weiteren **GMC-G** (*GMC-generalized*) genannt. Der Pseudo-Code des **GMC-G** ist im Anhang **D.3** zu finden.

²Der adaptierte Watershed ist Bestandteil von OpenCV, Version 2.4.7, docs.opencv.org

4.6 Zusammenfassung, Evaluierung und Grenzen

In diesem Kapitel wurde ein generelles Konzept zur Segmentierung verschiedenster **organo**-Gruppen entwickelt. Das generelle Konzept gliedert sich in die Schritte:

1. Detektion der organischen Objekte
2. Verifikation der Objektkandidaten
3. Vorder- und Hintergrundabschätzung (**KD-NN-G**, Teil 1)
4. Optimierungsbasierte Segmentierung (**KD-NN-G**, Teil 2)
5. Separierung der organischen Objekte mit **GMC-G**

Im ersten Schritt werden mit Haarkaskaden antrainierte Objekte im Bild I_{og} detektiert. Im zweiten Schritt werden die detektierten Fenster verifiziert und in verifizierte Kreisparameter CP^v transformiert. Genauer betrachtet, werden in der Verifikation potenzielle falsch-positiv detektierte Objekte durch Nutzung von Statistiken erster Ordnung und Farbähnlichkeiten entfernt. Im dritten Schritt wird eine **Trimap** anhand der verifizierten Kreisparameter CP^v und zufälligen Stichproben (*random sampling*) erstellt. Im vierten Schritt werden mit der **Trimap** und I_{og} durch den **KD-NN**-Algorithmus Wahrscheinlichkeitsbilder vom Vorder- und Hintergrund berechnet. Mit den Wahrscheinlichkeitsbildern wird anschließend eine optimierungsbasierte Segmentierung durchgeführt. Speziell wurden Graph-Cut und Beliefpropagation adaptiert. Der dritte und vierte Schritt ergeben zusammen den **KD-NN-G**-Algorithmus. Im fünften und letzten Schritt wird der **GMC-G**-Algorithmus zur Segmentierung einzelner Objekte im Bild I_{og} ausgeführt. Der **GMC-G** erweitert den **LSGMC** um Statistiken erster Ordnung und eine *Watershed*-Segmentierung.

Das generelle Konzept verallgemeinert die Schritte (1-5) des Konzeptes der Kombination (siehe Abschnitt 3.4.1) durch folgende Veränderungen:

1. Spezieller Haarkaskaden-Detektor je **organo**-Gruppe
2. Farb- und Größenverifikation statt Entfernung von Ausreißern im Bildraum
3. **Trimap**-Erstellung mit CP^v und *random sampling*
4. Segmentierung mit **BP** oder **GC** (Segmentierung wählbar, **GC** allerdings besser)
5. Erweiterung um Statistiken erster Ordnung über CP^v und *Watershed*

Zur Evaluierung des generalisierten Konzeptes sind manuell **GT**-Bilder weiterer **organo**-Gruppen erstellt worden. Insgesamt ergibt sich ein erweiterter Evaluierungsdatensatz (siehe Anhang C.2):

- 25 Bilder von Kartoffeln auf dem Boden (Wiese, Stein, Acker, Beton)
- 25 Bilder von Äpfeln auf dem Boden (Wiese, Stein, Sand)
- 25 Bilder von Plattfischen auf einem Förderband und dem Boden (Beton, Platten)

- 71 Bilder von Stammschnittflächen (**GT**-Bilder der Stammschnittflächen-Konzepte)

Die Genauigkeiten der Segmentierungen der einzelnen **organo**-Gruppen wurden mit den Maßen *precision*, *recall*, *f-score* und HD-Maß (siehe Anhang A.4) ermittelt. Evaluiert wurde die Genauigkeit der binären Segmentierung I_{og}^b mit Graph-Cut (**GC**) und Belief-propagation (**BP**) sowie die Genauigkeit der Multi-Objekt-Segmentierung I_{og}^m . In Tabelle 4.3 sind die Ergebnisse der Evaluierung dargestellt.

Die Tabelle 4.3 untergliedert sich in drei Untertabellen. Die obere Tabelle zeigt die Ergebnisse der binären Segmentierung mit **BP** und die mittlere Tabelle die Ergebnisse mit **GC**. Die untere Tabelle zeigt die Ergebnisse der Multi-Objekt-Segmentierung nach der **GC**- und nach der **BP**-Segmentierung. Weiterhin unterteilt sich die Tabelle in drei grobe Spalten. Die mittlere Spalte kennzeichnet den Mittelwert und die linke die Standardabweichung über die entsprechenden **GT**-Bilder.

Tabelle 4.3: Evaluierung der generalisierten Algorithmen zur Segmentierung verschiedener **organo**-Gruppen. Zu sehen ist *precision*, *recall* und *f-score* der binären Segmentierung mit **BP** (oben) und **GC** (Mitte) sowie die HD-Maße der nachfolgenden Multi-Objekt-Segmentierung (unten).

Evaluierung	Mittelwert			Standardabweichung		
<i>Binäre Segmentierung mit KD-NN-G (Belief Propagation - BP)</i>						
organo-Gruppe	precision	recall	f-score	precision	recall	f-score
Holzpolter	0,786	0,939	0,852	0,098	0,045	0,066
Äpfel	0,877	0,71	0,744	0,189	0,193	0,135
Kartoffeln	0,905	0,763	0,822	0,073	0,087	0,05
Plattfische	0,957	0,693	0,801	0,068	0,101	0,081
<i>Binäre Segmentierung mit KD-NN-G (Graph-Cut - GC)</i>						
organo-Gruppe	precision	recall	f-score	precision	recall	f-score
Holzpolter	0,917	0,915	0,915	0,045	0,047	0,032
Äpfel	0,987	0,893	0,935	0,011	0,085	0,049
Kartoffeln	0,933	0,871	0,899	0,047	0,065	0,041
Plattfische	0,971	0,816	0,884	0,078	0,099	0,08
<i>Multi-Objekt-Segmentierung mit GMC-G nach GC oder BP</i>						
organo-Gruppe	HD-M GC	HD-M BP	HD-M GC	HD-M BP		
Holzpolter	0,9353	0,8782	0,0232	0,04266		
Äpfel	0,9887	0,9521	0,007	0,0506		
Kartoffeln	0,9742	0,9628	0,01437	0,0159		
Plattfische	0,9443	0,9104	0,0277	0,0252		

Die Tabelle zeigt, dass der *f-score* der binären Segmentierung mit **GC** (mittlere Tabelle) bei allen **organo**-Gruppen besser ist als mit **BP** (obere Tabelle). Es ist auch zu erkennen, dass sowohl der Mittelwert vom *f-score* beim **GC** immer höher (mittlere Spalte) und die Standardabweichung (rechte Spalte) immer geringer ist als beim **BP**. Damit stellt Graph-Cut (**GC**) in allen Fällen die bessere Wahl der binären Segmentierung dar. Die adaptierte

4 Generalisierung für organische Objekte einer gleichartigen Gruppe

GC-Segmentierung basiert auf Wahrscheinlichkeitsbildern (I^p) und einem Gradientenbild. Die adaptierte **BP**-Segmentierung hingegen legt kein Gradientenbild zugrunde. Da die Konturen der **organo**-Gruppen sich **i.d.R.** durch hohe Gradienten auszeichnen, ist der benannte Unterschied eine mögliche Erklärung für das deutlich bessere Abschneiden der **GC**-Segmentierung. Ausgehend vom f -score und der **GC**-Segmentierung wurden mit **KD-NN-G** Äpfel (0,935) am besten segmentiert, gefolgt von Holzpoltern (0,915), Kartoffeln (0,899) und Plattfischen (0,884).

Die Multi-Objekt-Segmentierung mit **GMC-G** basiert auf den **KD-NN-G**-Ergebnissen. Schlechte Ergebnisse mit **KD-NN-G** führen zu entsprechenden Ergebnissen des **GMC-G**. Folglich sind (untere Tabelle) die HD-Maße vom **GC** besser als vom **BP**. Das HD-Maß gibt die Genauigkeit der Segmentierung eines Segmentes relativ zu allen anderen Segmenten wider. Ausgehend vom HD-Maß und der besten binären Segmentierung, der **GC**-Segmentierung, wurden Äpfel (0,9887) am besten segmentiert, gefolgt von Kartoffeln (0,9443), Plattfischen (0,9443) und Holzpoltern (0,9353). Die Reihenfolge der Genauigkeiten ist bis auf die Holzpolter identisch mit der Reihenfolge der **GC**-Segmentierung. Das HD-Maß (0,9353) ist bei den Holzpoltern zwar am geringsten, aber dennoch sehr gut und höher als das HD-Maß (0,926) mit dem **LSGMC** auf die **KD-NN-A**-Segmentierung.

Im Fazit segmentiert das generelle Konzept die exemplarisch gewählten **organo**-Gruppen sehr gut und segmentiert ähnlich genau (in einigen Fällen sogar besser) Stammschnittflächen wie die speziellen Konzepte. Weitere Ergebnisse des generellen Konzeptes sind im Anhang **B.3** zu sehen. Die Grundidee des Konzeptes ist in [GL13] veröffentlicht. Untersuchungen der **organo**-Gruppe Zooplankton und Hände sind in [Rus+13], [Gut+14] und [Gut+11b] veröffentlicht.

Das Konzept hat aufgrund der verwendeten Annahmen und der gesetzten Rahmenbedingungen in den Evaluierungsbildern (siehe Anhang **C.2**) einige Grenzen. Die Bildobjekte O^o in den Evaluierungsbildern:

1. sind gleichartige organische Objekte (*same species*),
2. sind ähnlich in Textur und Farbe,
3. überdecken sich gegenseitig nicht mehr als 20 %,
4. liegen ungefähr auf einer Ebene orthogonal zur Kamera-Blickrichtung und
5. sind **quasi-rund** (Überdeckung des besten angepassten Kreises zu O^o ist >60 %).

Die Rahmenbedingungen (1-4) wurden von Anfang an in dieser Arbeit gesetzt. Allerdings hat sich die fünfte Rahmenbedingung (quasi-runde Bildobjekte O^o) erst innerhalb der Arbeit als notwendig erwiesen. Bei der Verwendung des generellen Konzeptes auf nicht quasi-runde O^o sind ggf. entsprechende Anpassungen notwendig. Dies soll an dieser Stelle nicht weiter ausgeführt werden und wird stattdessen im Ausblick **6.2** aufgegriffen.

5 Anwendung und Ergebnisse der Holzpoltervermessung

In diesem Kapitel wird auf die Anwendung der Stammschnittflächensegmentierung im Vermessungssystem **AFoRS** eingegangen. Das **AFoRS**-System besteht aus mehreren Modulen und enthält eine Umsetzung des Konzeptes der Kombination mit den Algorithmen **KD-NN-A** und **LSGMC**. Das **AFoRS**-System ermittelt automatisch Volumen und Stammgrößen eines Holzpolters und wurde hinsichtlich der Vermessungsgenauigkeit mehrfach evaluiert. In diesem Kapitel werden zuerst in Abschnitt 5.1 einige zum Verständnis notwendige forstwirtschaftliche Begriffe geklärt. Anschließend wird in Abschnitt 5.2 **AFoRS** vorgestellt und die Architektur aufgezeigt. Im Abschluss wird im Abschnitt 5.3 das **AFoRS**-System evaluiert und ein Rückschluss auf die Anwendbarkeit des Konzeptes der Kombination zur Vermessung von Holzpoltern gezogen.

5.1 Forstwirtschaftliche Begriffe

In der forstwirtschaftlichen Praxis werden einige spezielle Begriffe verwendet. Die wichtigsten und zum weiteren Verständnis notwendigen Begriffe werden nachfolgend geklärt. Weiteres zur Forstwirtschaft kann unter [BB11] nachgelesen werden.

Polter, Rundholzabschnitte und Polterfrontfläche

Als **Polter** werden in der Forstwirtschaft gestapelte Baumstämme bezeichnet, welche zumeist im Wald oder an der Straße zu finden sind. Die einzelnen Baumstämme liegen aufeinander, haben eine einheitliche Länge und bilden an der Frontfläche ein typisches Trapez. Etwas abstrakter ist der Begriff **Rundholzabschnitt** zu betrachten, welcher lediglich aufeinander gestapelte Stämme beschreibt. Der Begriff inkludiert **bspw.** auch eine Ladung Baumstämme auf einem LKW. Die **Polterfrontfläche** bezeichnet die Seite eines Polters, welche die Stammschnittflächen zeigt. Nach dieser Definition gibt es folglich zwei Polterfrontflächen pro Polter.

Festmeter und Raummeter

Ein **Festmeter** (Fm) kennzeichnet das Volumen von Holz und entspricht einem Kubikmeter (m^3) festem Holz. Ein **Raummeter** (Rm) hingegen bezeichnet das Volumen inklusive Lufteinschlüsse von gestapeltem Holz wie Rundholzabschnitte. Folglich ist ein Raummeter ein Kubikmeter aufeinandergestapeltes Holz. Ein typischer Umrechnungsfaktor von Raum- zu Festmeter bei 3 m langen Holzstämmen ist 0,6 ($Fm = 0,6 * Rm$).

Rüttelmaß, Brutto- und Nettoraummeter

In der Praxis und im Holzhandel wird bei den Raummetern in Brutto- und Nettoraummeter unterschieden. Die **Bruttoraummeter** kennzeichnen das Volumen von Holz direkt nach dem Aufstapeln. Liegt das aufgestapelte Holz längere Zeit, so verdichtet sich das Volumen auf die sogenannten **Nettoraummeter**. Zur Umrechnung von netto zu brutto wird das **Rüttelmaß** angewandt, das zumeist einem Wert von 1,04 entspricht. Der Nettoraummeter ergibt sich aus dem Bruttoraummeter dividiert durch das Rüttelmaß.

Mittendurchmesser

Ein Baumstamm ist kein geometrisch perfekter Zylinder und entlang der Baumstammachse *i.d.R.* unterschiedlich im Durchmesser. Zur Beschreibung des Durchmessers eines Baumstammes ist in der Praxis daher der **Mittendurchmesser** üblich. Der Mittendurchmesser ist der Durchschnitt aller Durchmesser entlang der Baumachse.

Stammgrößenklassen

In der Forstwirtschaft werden die Stammgrößen abhängig vom Durchmesser d_{st} (Mittendurchmesser ohne Rinde) eines Baumstammes in Klassen eingeteilt. In diesem Kapitel wird von den gesetzlichen Bestimmungen in [For83] ausgegangen. Die **Stammgrößenklassen (L0 - L6)** teilen sich ein in:

- **L0:** $d_{st} < 10 \text{ cm}$; **L1a:** $d_{st} \in [10 \text{ cm}, 15 \text{ cm})$; **L1b:** $d_{st} \in [15 \text{ cm}, 20 \text{ cm})$
- **L2a:** $d_{st} \in [20 \text{ cm}, 25 \text{ cm})$; **L2b:** $d_{st} \in [25 \text{ cm}, 30 \text{ cm})$
- **L3a:** $d_{st} \in [30 \text{ cm}, 35 \text{ cm})$; **L3b:** $d_{st} \in [35 \text{ cm}, 40 \text{ cm})$
- **L4:** $d_{st} \in [40 \text{ cm}, 50 \text{ cm})$; **L5:** $d_{st} \in [50 \text{ cm}, 60 \text{ cm}]$; **L6:** $d_{st} > 60 \text{ cm}$

5.2 Das Holzpolter-Vermessungssystem AFoRS

AFoRS steht für Automatischer Fotogrammetrischer Rohholzvermessungsservice und verkörpert nach [Foe+12] „*einen Online-Service inklusive mobiler Endgeräte für den optimierten Datenaustausch entlang der Holzbereitstellungskette*“. Der Grundstein für das **AFoRS**-System¹ wurde in den gleichnamigen Forschungsprojekten AFoRS (05.2008-06.2010) und AFoRS II (09.2010 - 08.2011) gelegt. Die Scheller Systemtechnik GmbH (SST) und das Fraunhofer IGD entwickelten kooperativ das **AFoRS**-System, das aktuell vermarktet und zum Teil weiterentwickelt wird. Das Gesamtverfahren zur fotometrischen Mengenermittlung von Holzpoltern ist in [Sch+14] patentiert.

Das **AFoRS**-System wurde zur automatischen Vermessung von Holzpoltern sowie der entsprechenden Metadatenerfassung und Verwaltung entwickelt. Zur Vermessung müssen zuerst vom Endanwender (zumeist einer Person aus der Forstwirtschaft) einige Marker

¹Informationen zu AFoRS sich auch unter www.afors.eu zu finden.

an die Polterfrontfläche gesteckt werden. Anschließend werden einzelne Fotos von der Frontfläche aufgenommen, wobei mindestens zwei Marker in einem Bild sein müssen und alle Bilder zusammen die Frontfläche abdecken müssen. Die Aufnahme mehrerer Bilder ist in der Praxis oft unvermeidbar, da ein Polter mehrere Meter lang sein kann. Nach der Aufnahme der Bilder und der Angabe einiger Metadaten werden die Daten an eine Serviceplattform übertragen und verarbeitet. Die Gesamtarchitektur gliedert sich in eine Serviceplattform, eine mobile Applikation und der AFoRS-BV-Pipeline (Weiteres siehe [Foe+12]). Die Serviceplattform steuert die AFoRS-BV-Pipeline an und enthält ein Webfrontend für den Endanwender. Die Serviceplattform und mobile Applikation wurde von SST und die AFoRS-BV-Pipeline vom Fraunhofer IGD entwickelt. Die AFoRS-BV-Pipeline berechnet automatisch aus einer Serie von Bildern und einigen Metadaten die Volumen (Fest- und Raummeter), die Stammanzahl und die Stammgrößenverteilung eines Polters. Die Module der AFoRS-BV-Pipeline sind in Abbildung 5.1 skizziert.

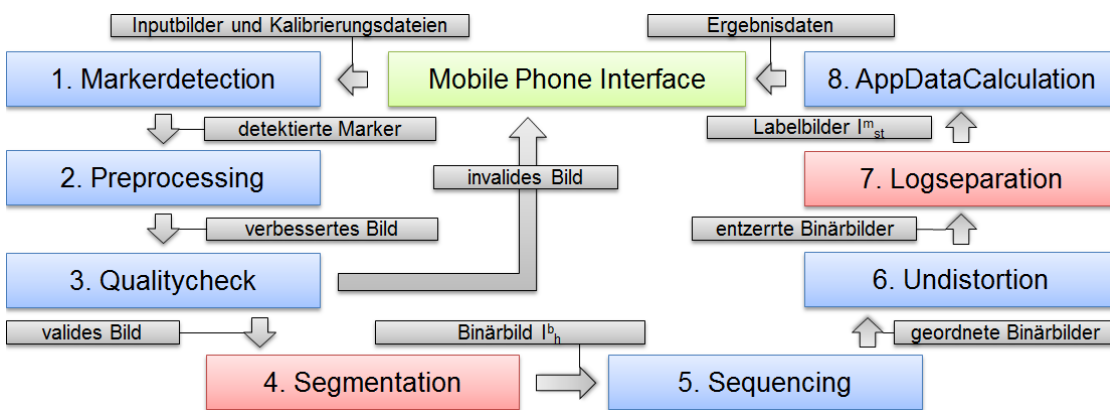


Abbildung 5.1: Die einzelnen Module der AFoRS-BV-Pipeline (Stand 02.05.2013). Das grüne Modul markiert das Interface und analog Start- und Endpunkt der Pipeline. Die roten Module beinhalten Ergebnisse dieser Arbeit.

Die AFoRS-BV-Pipeline besteht insgesamt aus acht Modulen und einem Interface (*Mobile Phone Interface*). Die Pipeline arbeitet in den ersten vier Modulen Einzelbilder sequenziell ab. Wenn alle Einzelbilder abgearbeitet wurden, werden ab dem fünften Modul alle Bilder zusammen sequenziell weiterverarbeitet. Die einzelnen Module haben folgende wesentliche Aufgaben und Inhalte:

1. **Markerdetektion**: detektiert die Marker und deren lokale Koordinatensysteme
2. **Preprocessing**: verbessert die Farbe, beseitigt Rauschen und die Randabschattung
3. **Qualitycheck**: prüft die Bildqualität hinsichtlich Bildschärfe, Über- und Unterbelichtung und bricht ggf. die Bearbeitung ab
4. **Segmentation**: segmentiert mit dem **KD-NN-A**-Algorithmus das Bild zu I_h^b

5 Anwendung und Ergebnisse der Holzpoltervermessung

5. **Sequencing**: bringt die Bilder in eine Ordnung, wodurch diese im Ergebnis von links nach rechts sortiert sind
6. **Undistortion**: entzerzt die Bilder optisch (Kissen oder Tonnenverzerrung) und perspektivisch, so dass die Bilder zur Polterfrontflächenebene korrespondieren und jeder Pixel gleich viele cm^2 überdeckt
7. **Logseparation**: separiert die Bildobjekte O^{st} mit dem **LSGMC**-Algorithmus
8. **AppData calculation**: setzt die Bilder zusammen und berechnet die Volumen, Stammanzahl sowie pro Baumstamm den Stammdurchmesser und die Stammgrößenklasse

Die AFoRS-BV-Pipeline setzt die Stämme virtuell (nicht sichtbar) zusammen und implementiert kein bildbasiertes *stitching*. Für den Nutzer werden die visualisierten Bilder lediglich an den Markern zusammengesetzt. Der Grund dafür ist die Annahme dieser Arbeit, dass Bildobjekte einer **organo**-Gruppe in einem Bild ähnlich, von Bild zu Bild aber verschieden sind. So ändern sich **bspw.** die Beleuchtungsbedingungen von einem Bild zum nächsten, wodurch die Stammschnittflächen heller oder dunkler werden. Eine Segmentierung pro Bild ist daher besser und robuster. Ein weiteres Modul ist das **Mobile Phone Interface**. Das Modul verwaltet die Eingangs- und Ausgangsdaten und ist die Schnittstelle der AFoRS-BV-Pipeline. Die Eingangsdaten bestehen aus den Bildern I_{wood} , der Stammlänge (Poltertiefe), dem Rüttelmaß, intrinsischen Kameraparametern und Parametern der Randabschattung, welche zuvor durch eine fotometrische Kalibrierung anhand einer Graukarte bestimmt werden. Die Ergebnisdaten setzen sich zusammen aus:

- Volumen in Fest- und Raummetern, Stammanzahl je Stammgrößenklasse
- Zusammengesetzten Bilder der Eingangs- und Ergebnisbilder
- Metadaten (Maßstab, Kreisparameter der Stämme im Bild, ect.)

Ein visualisiertes zusammengesetztes Ergebnisbild ist in [Abbildung 5.2](#) zu sehen.

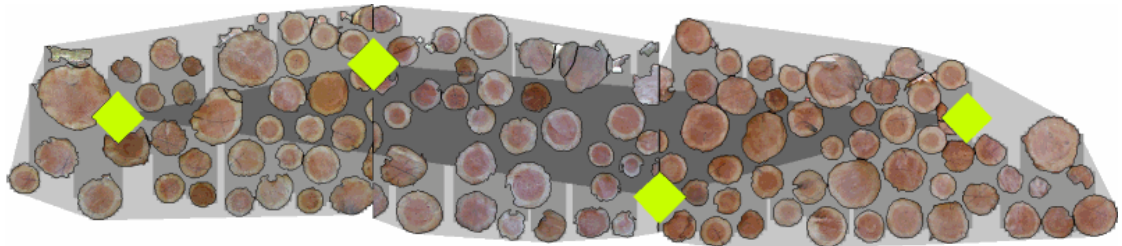


Abbildung 5.2: Die Abbildung zeigt ein an den Markern (gelb) zusammengesetztes **AFoRS**-Ergebnisbild. Jedes Einzelbild zeigt die konvexe Hülle (hellgrau), alternative Hülle (grau) sowie Marker-Hülle (dunkelgrau) und visualisiert die Überschneidung von I_{wood} und I_{st}^m .

Im Ergebnisbild in 5.2 überdeckt jedes Pixel die gleiche reale Fläche A_{px} . Die Festmeter (Rm) eines Polters berechnen sich aus der Anzahl a_{st} aller Stammschnittflächenpixel multipliziert mit der Stammlänge l_{st} und A_{px} . Die Raummeter hingegen werden mit zwei verschiedenen Verfahren ermittelt. Zum einen über die **alternative** und zum anderen über die **konvexe Hülle** der Stammschnittflächenpixel (siehe Abbildung 5.2). Zur Ermittlung der alternativen Hülle werden im Bild jeweils von oben und unten schmale Balken (ca. 15 Pixel breit) in das Bild geschoben, bis ein Stammschnittflächenpixel getroffen wird. Alle Pixel, die nicht von den Balken überdeckt werden, ergeben die alternative Hülle. Aus der Anzahl der Pixel a_h der jeweiligen Hüllen ergeben sich die Raummeter $Rm = a_h * l_{st} * A_{px}$. Die alternative Hülle wurde eingeführt, weil in der Praxis oftmals oben auf dem Polter vereinzelte Baumstämme liegen und folglich die Raummeter durch die konvexe Hülle zu groß werden.

5.3 Evaluierung von AFoRS und Rückschluss

Das AFoRS-System, speziell die AFoRS-BV-Pipeline, wurde mehrfach auf Basis von Praxisbildern evaluiert. Die Evaluierung hatte zum Ziel, die Genauigkeit der Vermessung zu ermitteln. Im Vordergrund steht im AFoRS-System die genaue Ermittlung der Raum- und Festmeter sowie die Anzahl der Stämme pro Stammgrößenklasse. Die Segmentierung hat in der AFoRS-BV-Pipeline den größten Einfluss auf die Ergebnisse. Je ungenauer die Segmentierung, desto fehlerhafter die Vermessung. Ein quantitativer Rückschluss auf die Genauigkeit der Segmentierung ist allerdings nicht möglich, da viele Faktoren die Genauigkeit der Vermessung beeinflussen. In [Fin04] stellte sich hinsichtlich des Rückschlusses heraus, dass „Abweichungen bis 5 % auch bei perfekter Segmentierung“ vorkommen können. Dennoch kann und soll in diesem Abschnitt anhand der Evaluierung ein Rückschluss über den Einfluss der Qualität der Stammschnittflächen Q_{st} und der Bildaufnahme Q_I auf die Segmentierung gezogen werden.

Die Genauigkeit der Vermessung hängt von vielen Einflussfaktoren ab. Faktoren, die die Vermessung beeinflussen, sind:

- die Genauigkeit der Segmentierung und der Maßstabsbestimmung
- die Genauigkeit des virtuellen Zusammensetzens der Stämme von Bild-zu-Bild
- die Abweichung vom Stammschnittflächendurchmesser zum Mittendurchmesser
- die Dicke der Rinde und die Qualität der Stämme

Ein weiterer Einflussfaktor in der Evaluierung ist die Genauigkeit der erhobenen Grundwahrheit. In der Evaluierung der AFoRS-BV-Pipeline wurden insgesamt 22 Bildserien von zwei **Referenzpoltern** und 90 Bildserien von 52 **Praxispoltern** verwendet. Eine Bildserie überdeckt jeweils die Frontfläche eines Polters. Die Einzeldaten der Evaluierungen sind im Anhang C.3 zu finden. Die Grundwahrheit (Raum- und Festmeter) für die Praxispolter wurde im Wald durch Forstarbeiter durch das Sektionsmessverfahren²

²Im Sektionsmessverfahren wird ein Polter in Sektionen unterteilt. Für jede Sektion wird die Höhe gemessen und das Volumen abgeleitet. Das Verfahren ist in der forstwirtschaftlichen Praxis üblich.

erhoben. Für die zwei Referenzpolter hingegen wurde die Grundwahrheit für jeden einzelnen Baumstamm im Werk (Laservermessung) erhoben. Die Daten sind präziser und im Gegensatz zu den Praxispoltern ist die Anzahl der Stämme pro Stammgrößenklasse bekannt. Die Qualität der Stammschnittflächen Q_{st} und der Bildaufnahme Q_I ist bei den evaluierten Poltern sehr verschieden. Die zwei Qualitätstypen $Q_{st} = \{A, B, C\}$ und $Q_I = \{A, B, C\}$ klassifizieren jeweils die entsprechende Qualität als A-gut, B-mittelmäßig oder C-schlecht (Details siehe Abschnitt 3.1.1). Die Referenz- und Praxispolter beinhalten Bildserien folgender Qualitäten:

- Referenzpolter (22 Bildserien) mit Q_{st} : 22 B und Q_I : 9 A, 9 B, 4 C
- Praxispolter (90 Bildserien) mit Q_{st} : 3 A, 57 B, 30 C und Q_I : 18 A, 41 B, 31 C

Die Praxispolter wurden von Personen aus der Forstwirtschaft und bewusst unter suboptimalen Bedingungen aufgenommen, um die Grenzen des AFoRS-Systems auszuloten.

In Tabelle 5.1 sind Statistiken über die Einzelevaluierungen (siehe Anhang C.3) der Referenz- und Praxispolter zu sehen. Die Statistiken zeigen den Mittelwert und die Standardabweichung der relativen Fehler von Festmeter (Fm.-F.), Raummeter (Rm.-F.) und Stammanzahl (S-F.) zur Grundwahrheit. Die evaluierten Raummeter (Rm.-F.) basieren auf der alternativen Hülle. Die ermittelten Raummeter durch die konvexe Hülle sind i.d.R. ungenauer. Weiterhin sind die Raummeter in netto angegeben und wurden mit einem Rüttelmaß von 1,04 bestimmt. Statistiken über die Stammgrößenklassen (L1-L6) der Referenzpolter werden an dieser Stelle nicht erhoben. Die Einzelevaluierungen der Stammgrößenklassen sind im Anhang C.3 in Tabelle C.2 abgebildet.

Tabelle 5.1: Statistiken über die relativen Messfehler des AFoRS-Systems von Poltern verschiedener Qualität ($Q_I; Q_{st}$); BS kennzeichnet die Anzahl der Bilder, Fm-F. und Rm-F. den relativen Fehler in % zu den Fm und Rm in m^3 und S-F. den relativen Fehler der Anzahl der Stämme.

Polter		Statistik zum relativen Fehler in %					
Referenzpolter		Mittelwert			Standardabweichung		
Qualität	BS	Fm-F.	Rm-F.	S-F.	Fm-F.	Rm-F.	S-F.
$Q_I = A; Q_{st} = B$	9	4,871	3,709	1,707	4,421	4,182	1,038
$Q_I = B; Q_{st} = B$	9	8,271	4,851	3,356	3,857	5,232	3,181
$Q_I = C; Q_{st} = B$	4	9,690	7,538	7,743	4,929	6,014	7,959
alle	22	7,138	4,872	3,479	4,536	4,918	4,258
Praxispolter		Mittelwert			Standardabweichung		
Qualität	BS	Fm-F.	Rm-F.	S-F.	Fm-F.	Rm-F.	S-F.
mittelmäßig	38	3,978	4,798	7,009	7,130	6,770	7,800
schlecht	52	14,992	12,964	10,595	22,505	18,860	13,507
alle	90	12,150	10,754	9,636	16,013	13,755	11,097

In der Tabelle 5.1 sind die relativen Fehler der Berechnungen bei verschiedenen Qualitäten (Q_{st}, Q_I) aufgeschlüsselt. Die obere Untertabelle zeigt die Ergebnisse der Re-

ferenzpolter bei einer guten, mittelmäßigen und schlechten Bildaufnahmequalität Q_I . Die Qualität der Stammschnittflächen Q_{st} ist bei allen Referenzpoltern mittelmäßig (B). Die untere Untertabelle zeigt die Ergebnisse der Praxispolter. Die Qualität der Praxispolter ist bei allen Qualitätstypen sehr verschieden. Um die Evaluierung einfach und überschaubar zu halten, sind die Kategorien mittelmäßig und schlecht verwendet worden. Unter **mittelmäßig** werden alle Polter mit guter oder mittelmäßiger Bildaufnahme und Stammschnittflächenqualität zusammengefasst ($Q_{st} \in \{A, B\}$ und $Q_I \in \{A, B\}$). Unter **schlecht** werden alle Polter mit schlechter Bildaufnahme oder Stammschnittflächenqualität zusammengefasst ($Q_I = C$ oder $Q_{st} = C$).

Die besten Ergebnisse wurden bei den Referenzpoltern mit $Q_I = A$ erzielt. Alle relativen Fehler sind in dem Fall unter 5 %. Bei den Praxispoltern hingegen wurden die besten Ergebnisse mit guter oder mittelmäßiger Qualität (Q_I oder Q_{st}) erzielt, wobei alle relativen Fehler kleiner-gleich 7 % sind. Allerdings liegen bei den Praxispoltern bei schlechter Bildaufnahme ($Q_I = C$) oder Stammschnittflächenqualität ($Q_{st} = C$) recht hohe relative Fehler vor. Bei den Referenzpoltern sind die relativen Fehler geringer, wenn ($Q_I = C$) ist.

Ein Rückschluss auf die Genauigkeit der Segmentierung ist aufgrund der vielen Einflussfaktoren kaum möglich. Dennoch zeigt sich, dass mit besserer Bild- und Stammschnittflächenqualität die Genauigkeit zunimmt. Diese Aussage bestätigt auch die visuelle Begutachtung der Ergebnisbilder. Es kann auch die Aussage getroffen werden, dass bei guter bis mittelmäßiger Bild- und Stammschnittflächenqualität die Berechnung der Stammanzahl einen durchschnittlichen relativen Fehler kleiner-gleich 7 % besitzt. Weiterhin ist der entsprechende relative Fehler bei guter bis mittelmäßiger Qualität in der Raummeterberechnung sogar unter 5 %.

6 Zusammenfassung und Ausblick

6.1 Zusammenfassung

Die Segmentierung ist ein breites Forschungsfeld und hat zum Ziel, die Pixel eines oder mehrerer gewünschter Bildobjekte wie eine Tasse aus einem Bild oder Video zu trennen. Eine allgemeine Lösung für beliebige Segmentierungsaufgaben gibt es nicht und ist auch kaum denkbar, da jedes Bild und deren Bildobjekte anders ist. Auch kann ein Algorithmus nicht entscheiden, was der Anwender segmentiert haben möchte. Diese Arbeit adressiert eine spezielle Klasse von Objekten, die **organo**-Gruppe, und leistet einen Beitrag zur Schaffung einer generellen Lösung für verschiedene Segmentierungsaufgaben. Eine **organo**-Gruppe ist eine Gruppe von gleichartigen organischen Objekten im Bild, wie Tomaten, Äpfel oder Stammschnittflächen. Die Segmentierung ermöglicht eine automatische bildbasierte Vermessung, Inspektion, Sortierung und Zählung von Objekten spezifischer und beliebiger **organo**-Gruppen. Als Basis zur Lösungsfindung wurde das Problem der Stammschnittflächensegmentierung zugrunde gelegt, da ähnlich wenig Annahmen über den Bildinhalt möglich sind wie bei der Menge der **organo**-Gruppen.

In dieser Arbeit wurde zuerst der aktuelle Stand der Forschung zur Segmentierung von **organo**-Gruppen betrachtet, untersucht und neu strukturiert. Es wurde gezeigt, dass noch keine robuste Lösung für das Problem der Stammschnittflächensegmentierung existiert. Es wurde weiterhin herausgearbeitet, dass auch für das allgemeine Problem der Segmentierung von **organo**-Gruppen noch keine Lösung existiert und bestehende spezifische Lösungen nur schwer übertragbar sind. Ausgehend vom aktuellen Stand der Forschung wurden drei neue Konzepte für das Problem der Stammschnittflächensegmentierung entwickelt und ein allgemeineres Konzept abgeleitet. Alle Konzepte bauen aufeinander auf und wurden quantitativ evaluiert. Die Evaluierung hat gezeigt, dass mit den Konzepten gute bis sehr gute Segmentierungsergebnisse erreicht werden.

Zuerst wurde ein **Konzept zur Schnittholzsegmentierung** durch Holzfarbe und Graph-Cut (**GC**) entwickelt. Es wurde der Farbraum von Holz untersucht und visualisiert. Es hat sich gezeigt, dass Holz einen Cluster im Farbraum bildet, welcher aber stark streut und sich mit dem Hintergrund überschneidet. Ausgehend von den Erkenntnissen wurde eine Vorsegmentierung von Holzfarbe entwickelt und mit einer **GC**-Segmentierung kombiniert. Das entwickelte Konzept zur binären Segmentierung von Schnittholz erwies sich in der Evaluierung als sehr genau.

Ausgehend von dem Gedanken der Segmentierung durch Objektdetektion wurde das **Konzept der Stammschnittflächendetektion** entwickelt. Es wurde aufgezeigt, dass die Methoden der Objektdetektion unterschiedlich gut Stammschnittflächen detektieren und Haarkaskaden eine sinnvolle Lösung bietet. Die Haarkaskaden wurden im Konzept verankert und um eine Verifikation sowie Segmentierung ergänzt. Die Evaluierung zeigte,

dass Stammschnittflächen mit dem Konzept relativ gut segmentiert werden können.

Auf Basis der neuen Erkenntnisse wurden die Vorteile der beiden Konzepte im **Konzept der Kombination** miteinander verbunden. Das resultierende Konzept kommt mit weniger Restriktionen in der Bildaufnahme aus und benötigt keine Farbheuristik. Die Evaluierung ergab, dass die Schnittholz- und die Stammschnittflächensegmentierung sehr genau und sogar besser als bei den anderen Konzepten ist. Nichtsdestotrotz ist das Konzept komplexer und benötigt entsprechende Rechenleistung. Die Ergebnisse des Konzeptes sind in ein praxistaugliches System zur automatischen Vermessung von Holzpoltern eingeflossen. Das System **AFoRS** berechnet automatisch das Volumen und die Stammgrößen eines Holzpolters und wurde hinsichtlich der Genauigkeit mehrfach evaluiert. Die Evaluierung ergab, dass bei guter bis mittelmäßiger Bildaufnahme- und Stammschnittflächenqualität ein relativer Fehler kleiner 7 % zu verzeichnen ist.

Ein weiterer Beitrag dieser Arbeit wurde durch die Entwicklung einer allgemeineren Segmentierung von **organo**-Gruppen geleistet. Das sogenannte **generelle Konzept** wurde vom Konzept der Kombination abgeleitet und ermöglicht die Segmentierung einer größeren Menge von **organo**-Gruppen. Das Konzept wurde mit Kartoffeln, Äpfeln, Plattfischen und auch Stammschnittflächen evaluiert. Für alle evaluierten **organo**-Gruppen wurden gute bis sehr gute Ergebnisse erzielt.

Die Ergebnisse dieser Arbeit sind zum einen die Konzepte als auch die dazu entwickelten Algorithmen. Innerhalb der Konzepte wurden die folgenden **neuen Methoden und Algorithmen** entwickelt:

- **KD-NN**: Eine allgemeine Methode zum Ermitteln von Wahrscheinlichkeiten auf Basis einer Vorsegmentierung oder **Trimap** – Die Methode ist genauer als bestehende Methoden und basiert auf einer Dichteabschätzung im Farbraum. Sie ermöglicht die Anbindung optimierungsbasierter Segmentierungsverfahren wie **GC** oder **BP**.
- **KD-NN-A**: Eine spezielle Methode zur Schnittholzsegmentierung basierend auf **KD-NN** und **GC** – Die Methode segmentiert automatisch Schnittholz anhand detektierter Stammschnittflächenfenster.
- **KD-NN-G**: Eine verallgemeinerte Form des **KD-NN-A** – Der Algorithmus ermöglicht eine binäre Segmentierung von **organo**-Gruppen.
- **PWL-H**: Eine spezielle Methode zur Verifikation von detektierten Stammschnittflächenfenstern – Die Methode ermöglicht die Reduktion von falsch-positiv detektierten Fenstern.
- **GMC**: Ein Algorithmus zur Kreiseinpassung in Binärbildern auf Basis von bekannten ungefähren Kreispositionen.
- **LSGMC**: Ein spezieller Algorithmus zur Separierung von Stammschnittflächen in einem binären Schnittholzbild
- **GMC-G**: Eine verallgemeinerte Form des **LSGMC**-Algorithmus – Der Algorithmus ermöglicht die Separierung von quasi-runden Objekten in einem Binärbild anhand detektierter Objektfenster.

6.2 Ausblick

In dieser Arbeit wurden Segmentierungen für **organo**-Gruppen entwickelt. Die Konzepte wurden anhand von Bildern mit **organo**-Gruppen, die

- sich gegenseitig nicht mehr als 20 % überdecken,
- ungefähr auf einer Ebene orthogonal zur Kamera-Blickrichtung liegen und
- quasi-rund sind (Überdeckung des besten angepassten Kreis zu O^o ist $>60\%$),

hergeleitet und evaluiert. Zur Segmentierung von **organo**-Gruppen unter abweichenden Rahmenbedingungen sind ggf. ergänzende Arbeiten und Anpassungen des generellen Konzeptes notwendig. Die Möglichkeiten der Anpassungen sind vielfältig. Auch die Erweiterung und/oder Adaption der Konzepte der Stammschnittflächensegmentierung ist in einigen Fällen sinnvoll.

Bei **organo**-Gruppen mit mehr als 20%igen gegenseitigen **Überdeckungen** von Objekten wird die Detektion ungenauer, da ein Objektmodell zumeist komplette Objekte besser bewertet als unvollständige Objekte. Eine mögliche Lösung ist die Detektion eines sichtbaren Objektbestandteiles, wie **bspw.** das Fischauge im Fischschwarm oder den Hühnerkopf in Hühnerherden. Eine andere Möglichkeit ist ähnlich wie im Konzept der Schnittholzsegmentierung die Anwendung eines speziellen, für die **organo**-Gruppe typischen Farbbereiches und/oder Farbraumes. Weiterhin wird bei starken Überdeckungen auch die Objektseparierung mit dem **GMC** keine genauen Ergebnisse mehr erzeugen. Anstelle der Kreise können **bspw.** aktive Konturen ausgehend von den detektierten Teilobjekten synchron auf dem Eingangsbild I_{og} und dem Binärbild I^b wachsen. Stoßen zwei Konturen aneinander, wird eine Objektgrenze angenommen.

Bei einer **organo**-Gruppe mit Objekten, die **nicht auf einer Ebene orthogonal zur Kamera** liegen, ist eine Anpassung erforderlich, wenn die Varianz in der Objektgröße zu groß wird. In Abbildung 6.1 (untere Bilderreihe) sind Hühner zu sehen, die aus einer schrägen Perspektive aufgenommen wurden. Die Hühner sind im Bildraum ähnlich groß und werden relativ gut segmentiert. In anderen Fällen, wie **bspw.** bei einem Bienenschwarm mit sehr dichten und weit entfernten Bienen, wird es problematischer. Die Objektseparierung mit Statistiken über die Objektgrößen ist nicht mehr sinnvoll, da zu hohe Größenvarianzen bestehen. Eine dankbare Erweiterung bieten Multi-Skalen-Ansätze. Auf den verschiedenen Ebenen (Skalen) können Objekte der entsprechenden Größenklasse detektiert und segmentiert werden. Das Ergebnisbild ergibt sich dann aus der Vereinigung der Ergebnisse der Ebenensegmentierung.

Anders verhält es sich mit **nicht-quasi-runden** organischen Objekten. In diesem Fall bedarf das generelle Konzept einer speziellen Anpassung auf die jeweilige Form der organischen Objekte. So sind **z.B.** Fische wie Heringe oder Aale sehr schmal und nicht quasi-rund. Die Erstellung der **Trimap** durch Kreise ist in diesem Fall nicht sinnvoll. Eine geeignetere Abschätzung ist durch die Segmentierung pro detektiertem Objektfenster und anschließender **Trimap**-Erstellung auf Basis der Segmentierungsergebnisse möglich. Geeignete Basismethoden sind **bspw.** *Region-Growing*, aktive Konturen oder auch Otsu's

Methode. In Abbildung 6.1 sind einige erste Ergebnisse der Graph-Cut Segmentierung basierend auf einer **Trimap** mit Otsu's Methode auf Fischen unter Wasser zu sehen. Auch die Objektseparierung mit dem **GMC-G**-Algorithmus ist nicht sinnvoll, da voraussichtlich die wachsenden und bewegenden Kreise konvergieren, bevor das gesamte Objekt überdeckt wird. Nichtsdestotrotz wird das generelle Konzept Ergebnisse erzeugen, aber voraussichtlich zu einer Übersegmentierung führen. Eine mögliche Lösung für das Problem der Objektseparierung besteht in der Nutzung eines spezifischen Formmodells, das die jeweilige **organo**-Gruppe beschreibt.

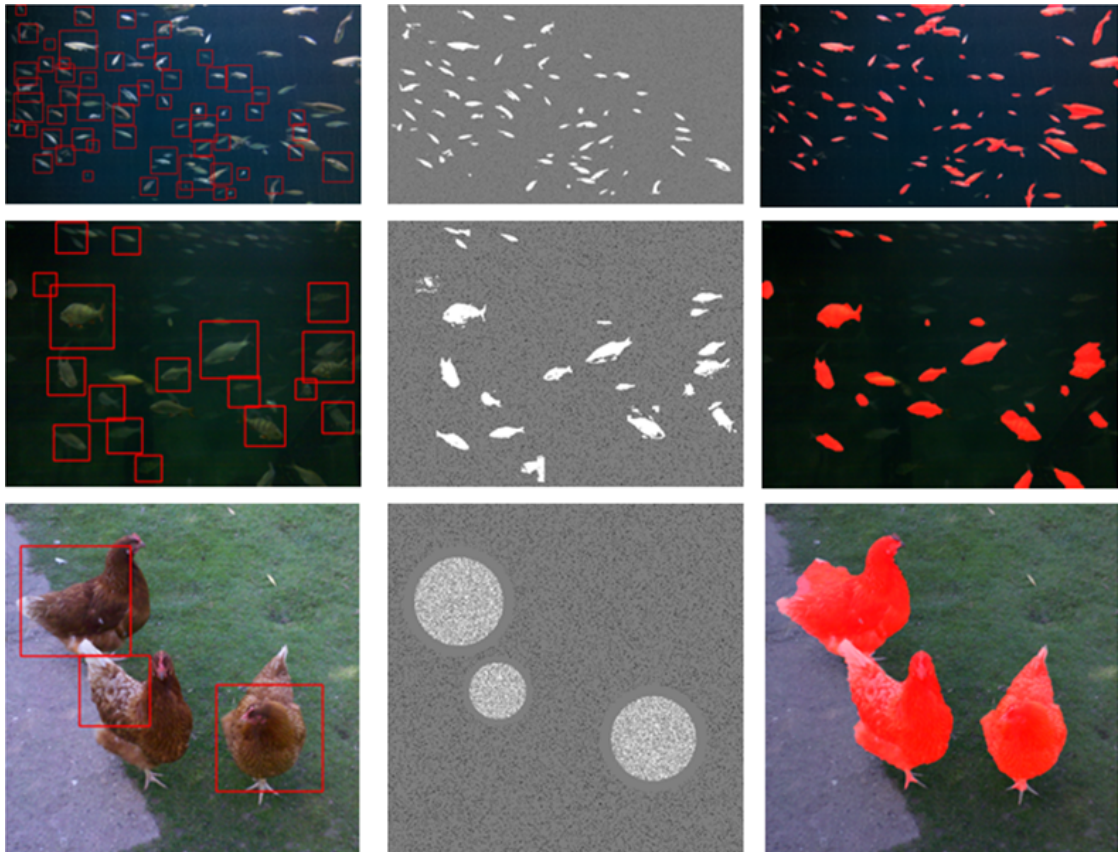


Abbildung 6.1: Segmentierung von Fischen unter Wasser und Hühnern im Hof mit dem generellen Konzept. Die linke Spalte zeigt die detektierten Objekte, die mittlere die **Trimap** und die rechte die binäre Segmentierung in rot überlagert mit I_{og} . Bei den Fischen wurde die **Trimap** auf Basis einer Einzelsegmentierung (Otsu's Methode [Ots79]) erzeugt.

Im Konzept der Objektdetektion wurde sich auf Basis von Experimenten für die Detektion von Stammschnittflächen mit Haarkaskaden entschieden und ein Detektor mit entsprechenden Trainingsdaten angelernt. Es ist sehr wahrscheinlich, dass mit einer größeren Trainingsmenge und entsprechenden aufwändigen Experimenten ein besserer De-

tektor für eine spezifische *organo*-Gruppe erstellt werden kann. Auch ist die Kombination verschiedener Methoden der Objektdetektion denkbar. In einem erst kürzlich erschienen technischen Bericht [Her14] wurde aufbauend auf dem Konzept der Kombination (siehe [GV12]) eine Segmentierung der Stammschnittflächen durch Kombination von HoG und LBP vorgenommen. Nach einer ersten Detektion wird ein Wahrscheinlichkeitsbild der Holzfarbe mit einem GMM-Modell erstellt und diese mit *Watershed* segmentiert. Die verbleibenden und nicht detektierten Segmente werden in Stammschnittfläche oder Nicht-Stammschnittfläche klassifiziert. Werden neue Stammschnittflächen erkannt, so wird das GMM-Modell neu angepasst und der Prozess wiederholt. Das beschriebene Vorgehen stellt eine Abwandlung des Konzeptes der Kombination dar und ist eine nennenswerte Alternative.

Weiterhin sind alle Konzepte nur für Einzelbilder mit einer aktuell üblichen Bildauflösung (ca. 1-4,9 Megapixel) geeignet. Größere Bilder führen zu einer polynomial wachsenden Berechnungszeit bei der optimierungsbasierten Segmentierung. Eine mögliche Lösung ist die Kachelung (*Tiling*) des Bildes, Segmentierung jeder Kachel einzeln und das Zusammensetzen der Konturen. Ein Algorithmus zur Segmentierung von Zooplankton auf sehr großen Bilddaten (80.000×80.000 Pixel) durch konturbasiertes *Split & Merge* ist in [Gut+14] veröffentlicht. Der Algorithmus in [Gut+14] zerlegt das Bild in Kacheln (*Splitting*) und segmentiert für jede Kachel Zooplankton mit Otsu's Methode. Anschließend werden die Konturen der Segmente pro Kachel bestimmt und die Konturen an den Kachelgrenzen zusammengesetzt (*Merging*). Durch die Auswechslung von Otsu's Methode mit dem generellen Konzept wird theoretisch auch die Verarbeitung von großen Bilddaten ermöglicht.

Das generelle Konzept ist außerdem für Einzelbilder und nicht für Videos konzipiert. In der Videosegmentierung ist durch die Bild-zu-Bild-Kohärenz eine zusätzliche Annahme möglich. Ein Objekt wird sich voraussichtlich von einem Bild zum nächsten nur geringfügig ändern. Durch diese Annahme und der häufigen Forderung nach Echtzeit sind andere Konzepte besser geeignet. Eine mögliche Lösung speziell zur Segmentierung von Händen ist in [Gut+11b] veröffentlicht. Die Hautfarbe wird über die Zeit angelernt und verifiziert. Durch Rückprojektion der Hautfarbcluster in den Bildraum werden Hände segmentiert. Das Verfahren ist *bspw.* an *organo*-Gruppen mit spezieller Farbgebung wie die Stammschnittflächen adaptierbar. Innerhalb eines Video der Stammschnittflächen lernt sich ein Farbmodell an. Anschließend wird das Farbmodell zur Segmentierung einzelner automatisch ausgewählter Bilder angewandt.

Die Segmentierung von Einzelbildern stößt aufgrund der begrenzten Information durch 2D-Pixel an Grenzen. Verfahren der 3D-Erfassung und/oder die Fusion der Bilddaten mit zusätzlichen Sensoren wie Bewegungssensoren eröffnen weitere Möglichkeiten. Einige Untersuchungen und Diskussionen zur 3D-Erfassung und Sensorfusion sind in [Bie+09], [GVW10] und [AGU14] veröffentlicht. Fische unter Wasser sollten sich theoretisch mit Stereokameras und den entsprechenden Verfahren genauer segmentieren lassen. Gleiches gilt für die Sensorfusion bei Personen oder Tieren im Video, die mit Bewegungssensoren ausgestattet sind.

A Grundlagen und Ergänzendes zum Maschinellen Sehen

A.1 Morphologische Operationen und Schwerpunkt

Morphologische Operationen

Einige morphologische Operationen der **BV** auf Binärbilder I^b werden **i.d.R.** mit Mengenoperationen (\cap, \cup, \setminus) notiert. Um Mehrdeutigkeiten gegenüber der Mengenlehre vorzubeugen, werden die entsprechenden Operationen speziell mit “ $\tilde{}$ ” notiert. Die Operationen haben folgende Bedeutung:

- $I_1^b \tilde{\cap} I_2^b$: die pixelweise logische UND (\wedge) Verknüpfung von I_1^b und I_2^b
- $I_1^b \tilde{\cup} I_2^b$: die pixelweise logische ODER (\vee) Verknüpfung von I_1^b und I_2^b
- $I_1^b \tilde{\setminus} I_2^b$: die pixelweise Differenz von I_1^b minus I_2^b (Ausnahme: $0 - 1 = 0$)
- $\overline{I^b}$: die pixelweise logische Negation (\neg) von I^b

Weiterhin wird in dieser Arbeit die morphologische Erosion mit \ominus und Dilatation mit \oplus notiert. Bei beiden morphologischen Operationen wird ein Kernel K (auch strukturiertes Element) über das Bild geschoben und der Zustand eines Pixels (0 oder 1) abhängig von K geändert. Der Kernel K ist zumeist eine quadratische Matrix und beschreibt im Fall der Erosion und Dilatation die zu betrachtenden Nachbarschaften durch Einsen. Die Erosion $I^b \ominus K$ setzt ein Pixel von 1 zu 0 und die Dilatation $I^b \oplus K$ von 0 zu 1, wenn mindestens ein 1 Element in K auch den Wert 1 in I^b hat. Die Erosion wirkt im Allgemeinen schrumpfend und die Dilatation erweiternd. Ein Beispiel 5×5 Kernel K , der auch innerhalb der Arbeit verwendet wurde, ist:

$$K = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (\text{A.1})$$

Schwerpunkt eines binären Bildes und einer Pixelmenge

Der Schwerpunkt (siehe [Jah05], [Ton05]) ist ein spezielles Bildmoment eines Binär- oder Grauwertbildes. Bildmomente sind Merkmale und charakterisieren Segmente oder Bilder.

Im diskreten Fall errechnet sich ein Bildmoment M_{ij} durch die Werte der Pixel $px_{x,y}$ des entsprechenden Bildes und den Positionen x und y durch:

$$M_{ij} = \sum_x \sum_y x^i y^j px_{x,y} \quad (\text{A.2})$$

Das Bildmoment M_{00} ist die Summe der Werte aller Pixel. Bei einem Binärbild entspricht das der Anzahl der Pixel mit dem Label 1. Der Schwerpunkt (\bar{x}, \bar{y}) ergibt sich durch:

$$(\bar{x}, \bar{y}) = \left(\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right) \quad (\text{A.3})$$

Bezogen auf ein Binärbild I^b und die, in dieser Arbeit verwendeter Formalismen, ergibt sich der Schwerpunkt (\bar{x}, \bar{y}) durch:

$$N = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} px_{x,y}; \quad \bar{x} = \frac{1}{N} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} px_{x,y} * x; \quad \bar{y} = \frac{1}{N} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} px_{x,y} * y; \quad (\text{A.4})$$

Bezogen auf eine Pixelmenge P , welche auch einem Segment S entsprechen kann, ergibt sich der Schwerpunkt (\bar{x}, \bar{y}) durch:

$$\bar{x} = \frac{1}{|P|} \sum_{i \in \Lambda} px_{x,y}^i * x; \quad \bar{y} = \frac{1}{|P|} \sum_{i \in \Lambda} px_{x,y}^i * y; \quad px_{x,y} \in P \quad (\text{A.5})$$

wobei Λ die Indexmenge ist und P indexiert.

A.2 Aktive Konturen und die Energierme

Eine aktive Kontur nach [KWT88] der Form

$$E_{snake}^* = \int_0^1 E_{snake}(c(s)) = \int_0^1 E_{int}(c(s)) + E_{image}(c(s)) + E_{con}(c(s)) ds, \quad (\text{A.6})$$

setzt sich aus der internen Energie $E_{int}(c(s))$ und den externen Energien $E_{image}(c(s))$ und $E_{con}(c(s))$ zusammen. Die externe Energie E_{con} für externe Constrains wird **i.d.R.** vernachlässigt, womit in den meisten Fällen nur die Bildenergie und die interne Energie zur Anwendung kommt. Die interne Energie $E_{int}(c(s))$ bestimmt die Krümmung der Kurve mittels der ersten und zweiten Ableitung, wie folgt:

$$E_{int}(c(s)) = \alpha \int_0^1 |c'(s)|^2 ds + \beta \int_0^1 |c''(s)|^2 ds \quad (\text{A.7})$$

Die Gewichtung wird über α und β vorgenommen. Eine Variante, die externe Bildenergie zu Bestimmen, ist die Nutzung von Bildgradienten $\nabla I(c(s))$. Nach [OF03] wird das

Integral über die von der Kontur C überlagerten Gradienten maximiert, somit ist:

$$E_{int}(c(s)) = -\gamma \int_0^1 |\Delta I(c(s))|^2 ds \quad (\text{A.8})$$

Die Gewichtung erfolgt über γ . Zusammengefasst ergibt sich die zu minimierende Formel (siehe auch [OF03], [CV01]) mit den Gewichten α , β und γ :

$$\min_s E(c(s)) = \min_s \alpha \int_0^1 |c'(s)|^2 ds + \beta \int_0^1 |c''(s)|^2 ds - \gamma \int_0^1 |\nabla I(c(s))|^2 ds \quad (\text{A.9})$$

A.3 Klassifikationen von Segmentierungsmethoden

Die Methoden der Segmentierung werden verschieden klassifiziert. Nachfolgende Klassifikationen sind in den jeweiligen Büchern zu finden, welche absteigend chronologisch jünger werden:

- „*A guided tour of computer vision*“, [Nal93]:
 - (1) pixel classification, (2) splitting and merging
 - (3) relaxation
- „*Digital image processing: a practical introduction using Java*“, [Eff00]:
 - non contextuell
 - * (1) thresholding grey level, (2) thresholding color
 - contextuell
 - * (1) pixel connectivity, (2) region similarity
 - * (3) region growing, (4) split and merge
- „*Digital Image Processing*“, [Jah05]:
 - (1) Pixelorientierte, (2) Kantenorientierte
 - (3) Regionenorientierte, (4) Modellbasierte
- „*Grundlagen der Bildverarbeitung*“, [Ton05]:
 - Segmentierung
 - * (1) Histogrammbasiert, (2) Multiscalenstrategien
 - * (3) Segmentierung nach Homogenitätskriterien
 - * (4) Diskontinuitätskriterien, (5) Relaxation Labelling
 - modellbasierte Segmentierung
 - * (1) Interaktive Suche, (2) Vollständige Suche, (3) Veränderbare Templates

- „*Computer Imaging: Digital Image Analysis and Processing*“, [Umb05]:
 - (1) region growing Shrinking, (2) clustering, (3) boundary detection
- „*Bildverarbeitung in der Praxis*“, [Ste06]:
 - (1) Schwellwertsegmentierung, (2) Templates, (3) Kantendetektion
 - (4) Gebietswachstum, (5) Split and Merge
- „*Computer Vision: Algorithms and Applications*“, [Sze10]:
 - (1) active contours, (2) split and merge, (3) meanshift and mode finding
 - (4) normalized cut, (5) graph cut and energybased methods
- „*Computer Vision: A Modern Approach*“, [FP11]:
 - (1) Segmentation by Clustering, (2) Segmentation by Fitting a Model
 - (3) Segmentation and Fitting using Probabilistic Methods

A.4 Evaluierungsmaße der Segmentierung

precision, recall und f-score

Zur Evaluierung der binären Segmentierung sind die Maße *precision*, *recall* und *f-score* gängig und sinnvoll (siehe [Pow07]). Die Maße basieren auf den vier möglichen Fällen von Grundwahrheit zum Segmentierungsergebnis. Die vier Fälle sind *tp* (*true positive*), *tn* (*true negativ*), *fp* (*false positive*) und *fn* (*false negative*). Im Bezug zur Grundwahrheit (**GT**) am Beispiel der Vordergrundsegmentierung bedeutet dies:

tp: Pixel ist Vordergrund und Pixel ist segmentiert

tn: Pixel ist Hintergrund und Pixel ist nicht segmentiert

fp: Pixel ist Hintergrund und Pixel ist segmentiert

fn: Pixel ist Vordergrund und Pixel ist nicht segmentiert

Der Grundgedanke ist an die **Objektdetektion** adaptierbar, wobei dann gilt:

tp: Objekt ist vorhanden und Objekt wurde detektiert

tn: Objekt ist nicht vorhanden und Objekt wurde nicht detektiert

fp: Objekt ist nicht vorhanden und Objekt wurde detektiert

fn: Objekt ist vorhanden und Objekt wurde nicht detektiert

Aus der Anzahl der tp , fp und fn Pixel ermittelt sich *precision* und *recall*, wie folgt:

$$precision = \frac{tp}{tp + fp}; \quad recall = \frac{tp}{tp + fn} \quad (\text{A.10})$$

Der *f-score* ist das harmonische Mittel von *precision* und *recall* und ergibt sich durch:

$$fcore = \frac{precision * recall}{precision + recall} \quad (\text{A.11})$$

HD-Maß

Das HD-Maß ist geeignet zur Bestimmung der Genauigkeit einer Multi-Objekt-Segmentierung. Der Name des Maßes (HD-Maß) wurde übernommen aus [McG+07] und ist eine Komposition der Namen der Autoren Q. Huang und B. Dom in [HD95]. Der Wertebereich vom HD-Maß ist $[0, 1]$, wobei ein Wert von 1 eine perfekte Segmentierung kennzeichnet. Zur Bestimmung des HD-Maßes werden zuerst die Korrespondenzen der Segmente SI_{sg} der Segmentierung zu den Segmenten des GT-Bildes SI_{gt} ermittelt. Die Segmentmengen $SI_{sg} \subset SI$ und $SI_{gt} \subset SI$ enthalten die Segmente der segmentierten Bilder:

$$\begin{aligned} SI_{sg} &= \{S_{sg}^1, S_{sg}^2, \dots, S_{sg}^n\}; \quad n \in \mathbb{N}^{>0} \\ SI_{gt} &= \{S_{gt}^1, S_{gt}^2, \dots, S_{gt}^m\}; \quad m \in \mathbb{N}^{>0} \end{aligned} \quad (\text{A.12})$$

Ein Segment in $S_{sg}^i \in SI_{sg}$ korrespondiert mit einem Segment $S_{gt}^j \in SI_{gt}$, wenn $|S_{gt}^i \cap S_{sg}^j|$ maximal ist. Die Korrespondenz wird in [HD95] als direkte Hamming Distanz bezeichnet und wird mit $D_H(SI_{sg} \implies SI_{gt})$ notiert. Die direkte Hamming Distanz wird berechnet mit:

$$D_H(SI_{sg} \implies SI_{gt}) = \sum_{S_{gt}^i \in SI_{gt}} \sum_{S_{sg}^k \neq S_{sg}^j, S_{sg}^k \cap S_{gt}^i \neq \emptyset} |S_{gt}^i \cap S_{sg}^k| \quad (\text{A.13})$$

Die Indizes i , j und k indizieren die Segmente der jeweiligen Mengen. Die direkte Hamming Distanz $D_H(SI_{sg} \implies SI_{gt})$ ist die Anzahl der Pixel der Schnittfläche zwischen allen $S_{gt}^i \in SI_{gt}$ mit den nicht-maximalen Segmenten S_{sg}^k in SI_{sg} . Die umgekehrte Richtung wird in [HD95] als inverse Hamming Distanz bezeichnet und mit $D_H(S_{gt} \implies S_{sg})$ notiert. Die Berechnung ist ähnlich der direkten Hamming Distanz. Aus der direkten und inversen Hamming Distanz ergibt sich das in dieser Arbeit verwendete HD-Maß:

$$\text{HD-Maß} = 1 - \frac{D_H(S_{sg} \implies S_{gt}) + D_H(S_{gt} \implies S_{sg})}{2 * |I|} \quad (\text{A.14})$$

I bezeichnet das Bild und die Mächtigkeit von $|I|$ entspricht der Pixelanzahl.

S-Fehler

Der S-Fehler (Stammanzahl-Fehler) ist eine Ergänzung zum HD-Maß in den Evaluierungen. Der S-Fehler kennzeichnet den relativen Fehler der Stammanzahl. Er ergibt sich aus der realen s_r und der detektierten Stammanzahl s_d durch:

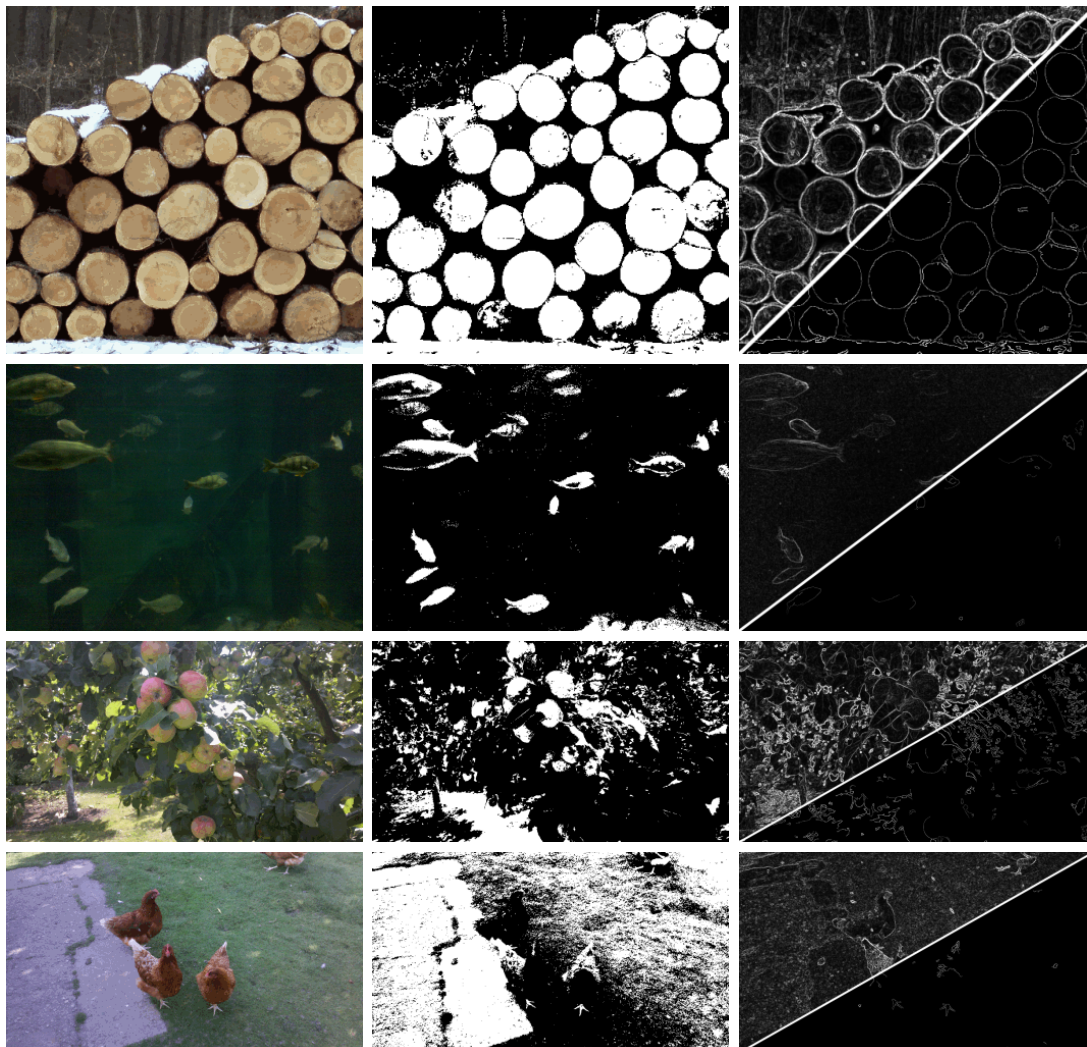
$$\text{S-Fehler} = \frac{|s_r - s_d|}{s_r} \quad (\text{A.15})$$

Je kleiner der Fehler, um so besser das Ergebnis.

B Segmentierungsergebnisbilder

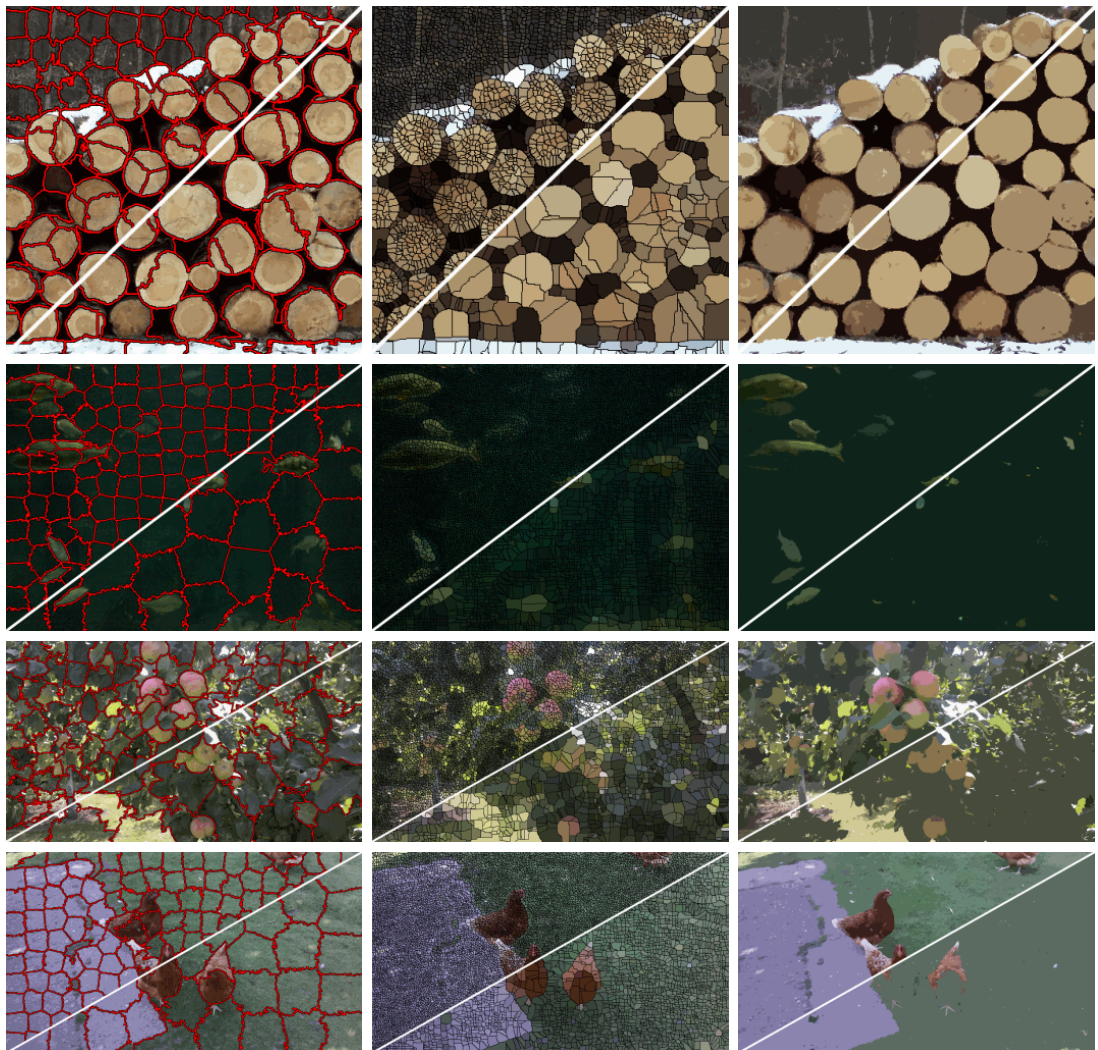
B.1 Ergebnisbilder mit Otsu's Methode, Sobel und Canny

Nachfolgend sind die Ergebnisse von Otsu's Methode [Ots79], dem Sobel Operator und Canny Kanten Detektor [Can86] zu sehen. Die Ergebnisbilder von verschiedenen **organo**-Gruppen wurden unter Nutzung von OpenCV 2.4.7 erzeugt. Rechts ist jeweils das Eingabebild, in der Mitte das Ergebnis von Otsu's Methode, links oben das Sobel-Gradientenbild und links unten das Canny-Kantenbild.



B.2 Ergebnisbilder regionenbasierter Basismethoden

Nachfolgend sind die Ergebnisse der regionenbasierten Basismethoden **SLIC**, *Watershed* und *Mean-Shift* zu sehen. Die Ergebnisse korrespondieren zu den Bildern in Anhang B.1. Die linke Spalte zeigt die Ergebnisse erzeugt mit **SLIC** [Ach+12] bei *Compactness* $C = 20$ mit 200 *Clustern* (oben links) und 50 (unten rechts). In der mittleren Spalte sind die *Watershed*-Ergebnisse der VIGRA [Kot00] 1.9.0 mit einer Skalierung der Gradienten von 2 (oben links) und 8 (unten rechts) zu sehen. In der rechten Spalte sind die *Mean-Shift*-Ergebnisse von OpenCV 2.4.7 mit einem Suchabstand im Farb- und Bildraum von 20 (oben links) und von 50 (unten rechts) zu sehen. Bei **SLIC** (linke Spalte) begrenzen rote Linien die errechneten Segmente und bei *Watershed* (mittlere Spalte) sind es schwarze Linien. Weiterhin bilden bei *Watershed* (mittlere Spalte) und *Mean-Shift* (rechte Spalte) benachbarte Pixel gleicher Farbe ein Segment.



B.3 Ergebnisbilder der verschiedenen Konzepte

Schnittholzsegmentierung mit Graph-Cut und Bildaufnahmerestriktionen

In Abbildung B.1 sind die Ergebnisse des Konzeptes der Schnittholzsegmentierung mit Restriktionen in der Bildaufnahme (**WS-GC-R**) zu sehen. Abgebildet sind die Differenzen I_{dif}^b von den Ergebnissen der Schnittholzsegmentierung I_h^b zu den korrespondierenden **GT**-Bildern.

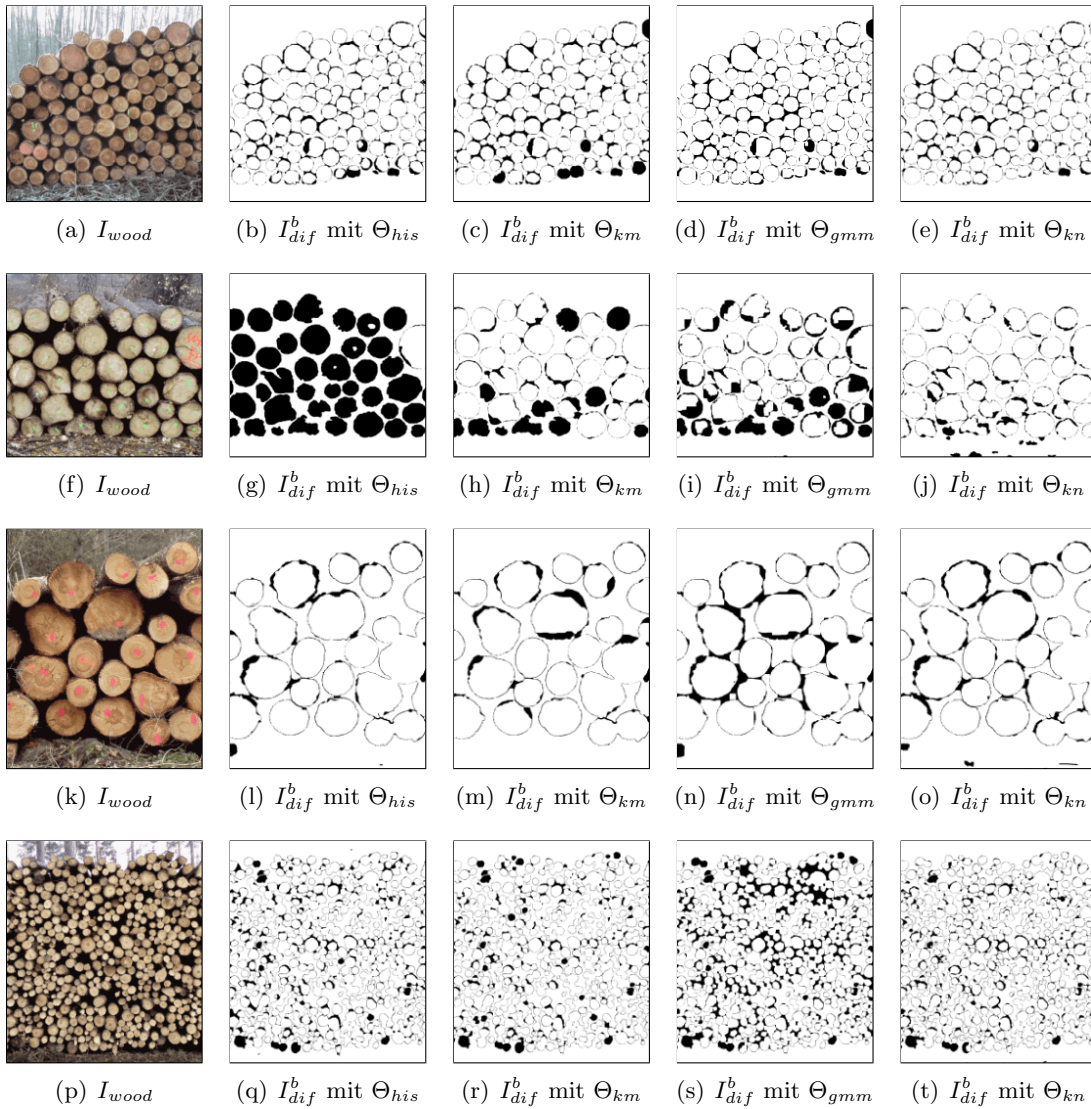


Abbildung B.1: Die Abbildung zeigt von links nach rechts I_{wood} und die Differenzen I_{dif}^b von I_h^b zur Grundwahrheit von vier verschiedenen Modellen Θ .

B Segmentierungsergebnisbilder

Schwarze Pixel kennzeichnen einen Segmentierungsfehler. Es wurden die Parameter $PA_{his} = (0.125, 1000)$, $PA_{km} = (8, 0.02)$, $PA_{gmm} = (8)$ und $PA_{kn} = (2.5, 0.5)$ für die verschiedenen Modelle Θ , der RGB-Farbraum und $\alpha = 0.0005$ verwendet. Die besten Ergebnisse wurden mit dem Modell Θ_{kn} erzielt. Einige Ergebnisse sind in Abbildung B.2 zu sehen.



Abbildung B.2: Abgebildet sind einige zufällig ausgewählte Segmentierungsergebnisse mit **KD-NN** von den 71 Evaluierungsbildern (I_{wood}). Weiße Pixel kennzeichnen Schnittholz und schwarze Nicht-Schnittholz.

Stammschnittflächensegmentierung mit Objektdetektion

In Abbildung B.3 sind einige Ergebnisse des Konzeptes der Stammschnittflächendetektion (Unterkapitel 3.3) zu sehen. Es sind sowohl die Ergebnisse der direkten Abbildung der verifizierten Kreise (PWL-H) als auch die Ergebnisse der Kreisaneinanderpassung durch aktive Konturen abgebildet.

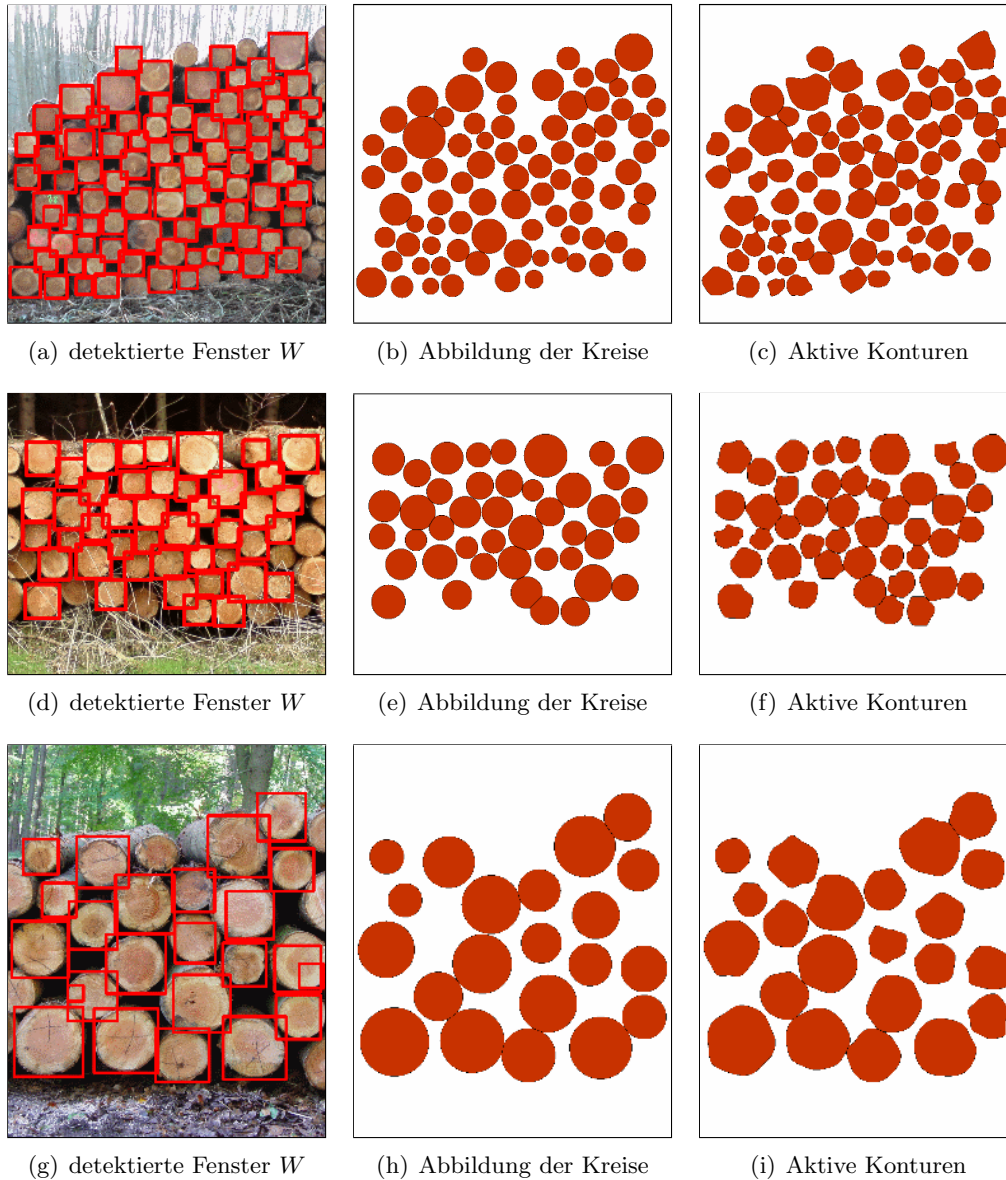


Abbildung B.3: Von links nach rechts sind die detektierten Fenster W , die Segmentierung durch Kreise und die aktiven Konturen dargestellt. Schwarze Konturen kennzeichnen in den Ergebnisbildern Segmentgrenzen.

Stammschnittflächensegmentierung mit Objektdetektion und Graph-Cut

In Abbildung B.4 sind einige Ergebnisse vom Konzept der Kombination (Unterkapitel 3.4) zu sehen. Pro Ausgangsbild I_{wood} sind jeweils die verifizierten Kreise CP^v , die Trimap I^{tri} und das Labelbild I_{st}^m abgebildet.

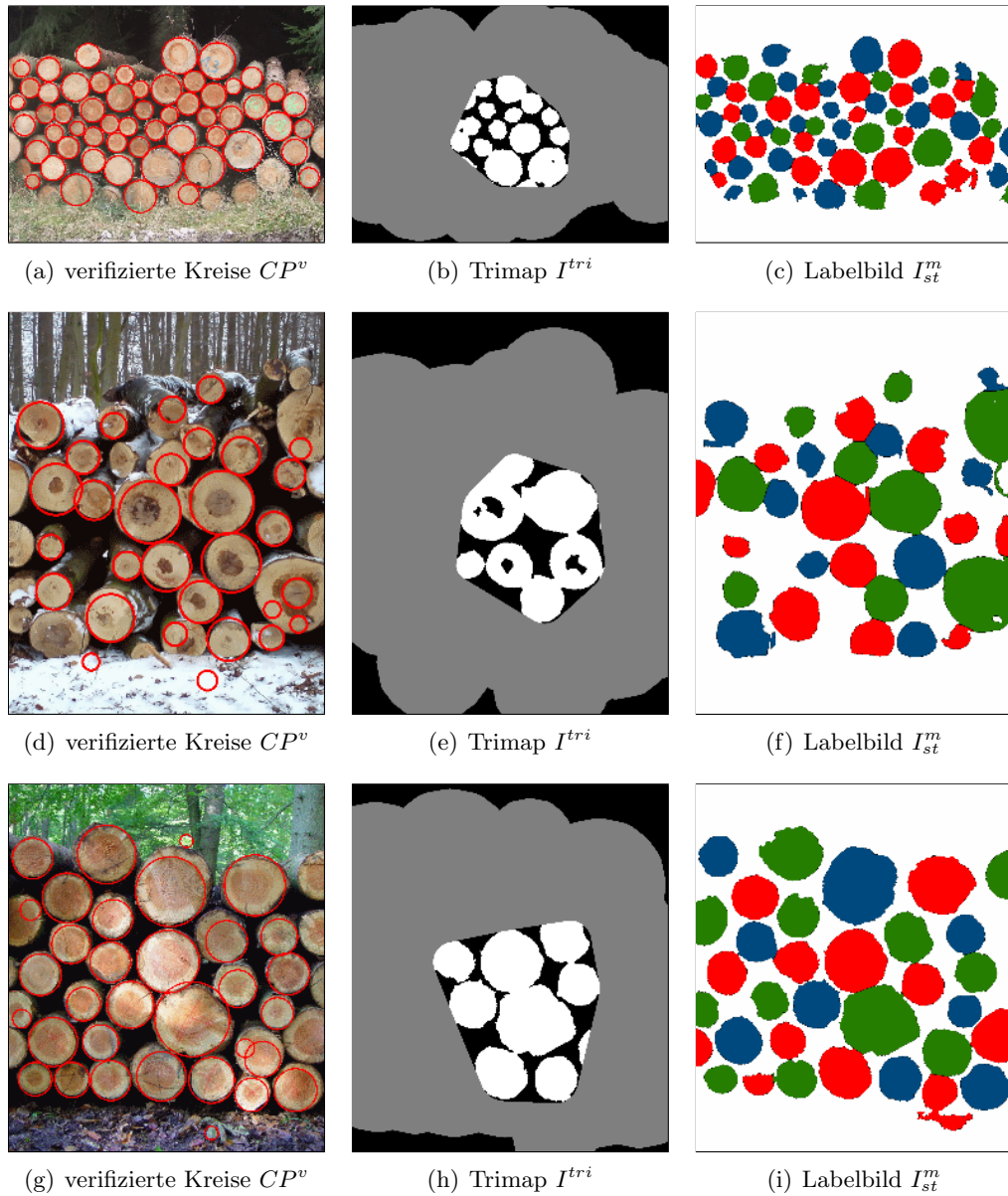


Abbildung B.4: Von links nach rechts sind die verifizierten Kreise CP^v , die Trimap I^{tri} und das Labelbild I_{st}^m dargestellt. Zur visuellen Unterscheidung sind die Segmente in I_{st}^m in drei Farben abgebildet.

Generalisierte Segmentierung von **organo**-Gruppen

In Abbildung B.4 sind einige Ergebnisse vom generalisierten Konzept (Kapitel 4) zu sehen. Pro Ausgangsbild I_{og} sind jeweils die verifizierten Kreise CP^v , die Trimap I^{tri} und das Labelbild I_{st}^m abgebildet.

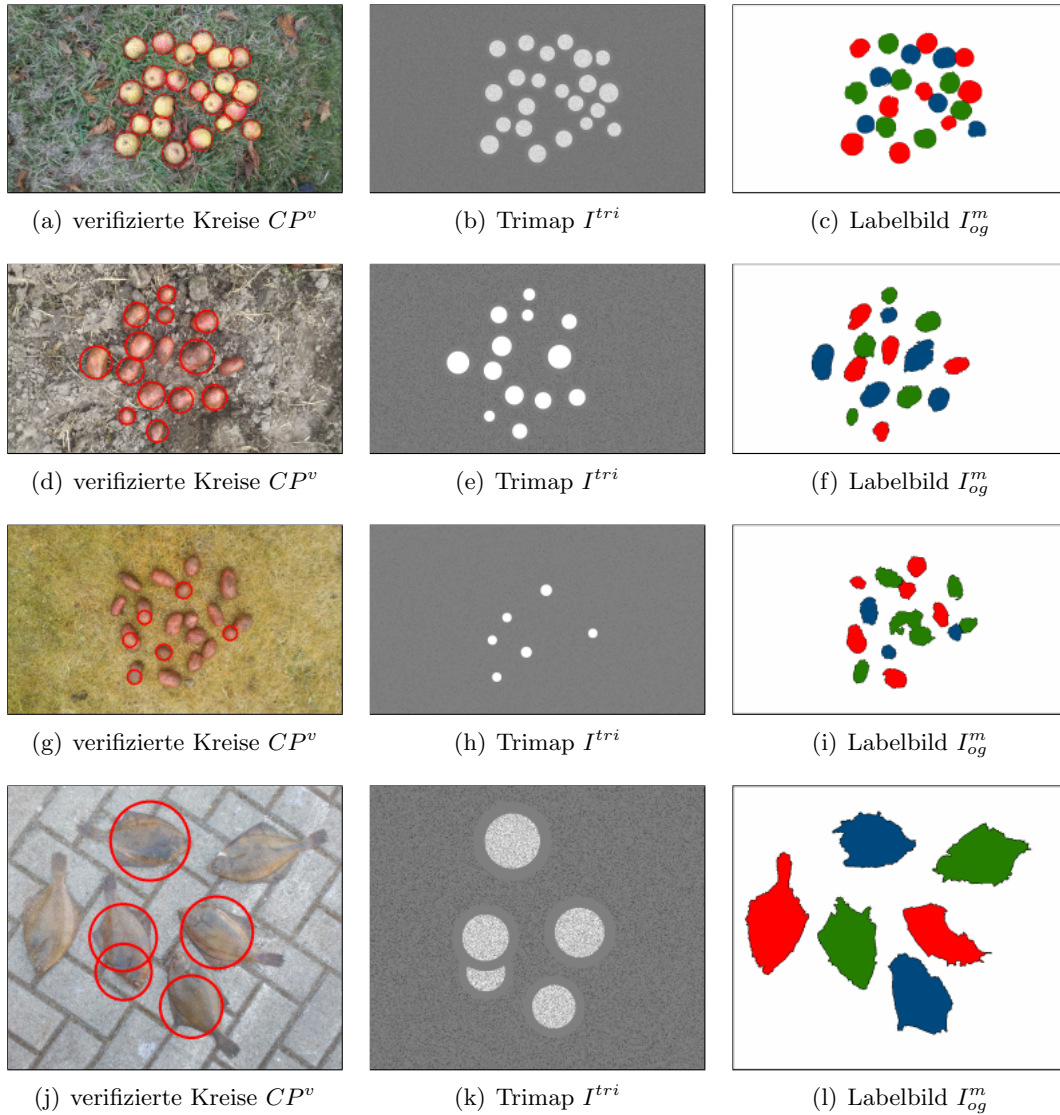


Abbildung B.5: Von links nach rechts sind die verifizierten Kreise CP^v , die Trimap I^{tri} und das Labelbild I_{st}^m dargestellt. Zur visuellen Unterscheidung sind die Segmente in I_{st}^m in drei Farben abgebildet.

C Visualisierung und Evaluierungsdaten

C.1 Farbraumvisualisierung von Schnittholz

Abbildung C.1 zeigt die Visualisierung von Schnittholzpixeln FG_{gt} und Nicht-Schnittholzpixeln BG_{gt} . FG_{gt} und BG_{gt} wurde anhand von 71 GT-Bildern erhoben (siehe Anhang C.2). Es sind normalisierte Histogramme der Farbräume RGB, HSV, YCrCb und Luv durch $2D$ Scatterplots visualisiert. Jeder Wert des Histogramms wurde zusätzlich durch die 3. Wurzel herunterskaliert.

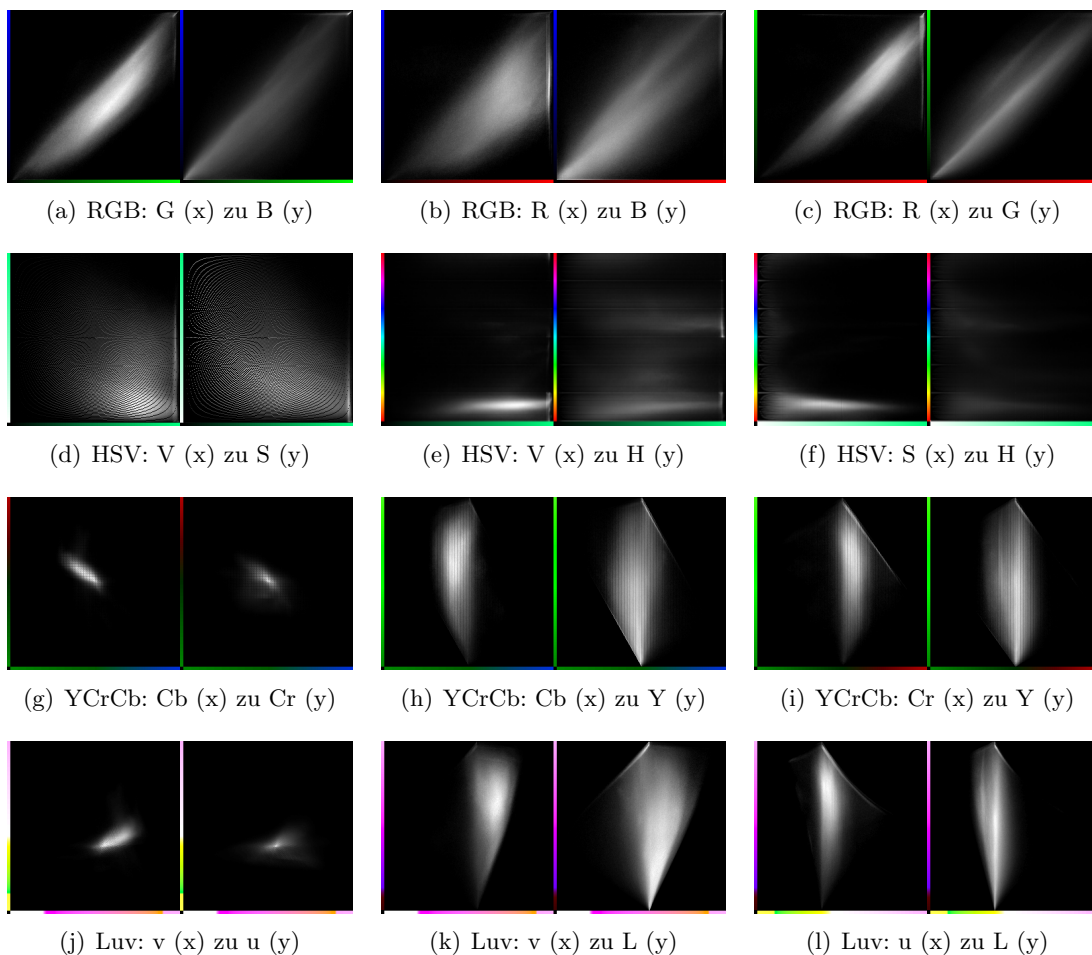


Abbildung C.1: Visualisierung von FG_{gt} (links) und BG_{gt} (rechts).

C Visualisierung und Evaluierungsdaten

Abbildung C.2 zeigt die Visualisierung von Schnittholzpixeln FG_{gt}^{sub} und Nicht-Schnittholzpixeln BG_{gt}^{sub} vom Bildzentrum (innerer Holzpolterbereich). Die Visualisierungstechnik ist die gleiche wie in Tabelle C.1. Auch wurden wieder 71 GT-Bilder verwendet.

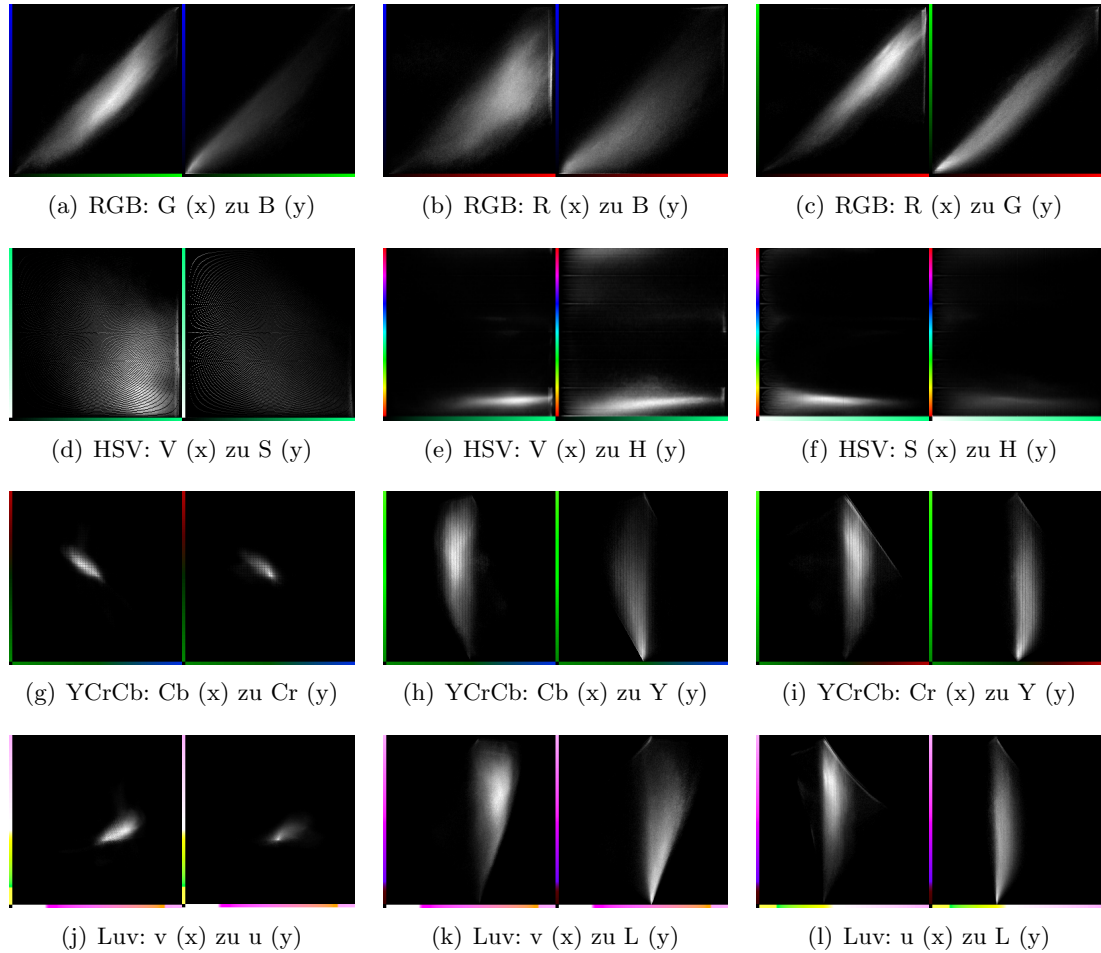


Abbildung C.2: Visualisierung von FG_{gt}^{sub} (links) und BG_{gt}^{sub} (rechts).

C.2 Evaluierungsbilder und deren Grundwahrheit

In Abbildung C.3 sind die 71 Evaluierungsbilder (I_{wood}) zu sehen.



Abbildung C.3: Abgebildet sind die 71 Evaluierungsbilder (I_{wood}). Auf Basis der Bilder wurden die **GT**-Bilder erstellt. Die Bilder sind von verschiedener Qualität Q_{st} und Q_I und haben verschiedene Bildgrößen.

Zu allen Evaluierungsbildern wurde die Grundwahrheit (*Grund Truth*) erstellt, welche in dieser Arbeit als **GT**-Bilder bezeichnet werden. In den **GT**-Bildern wurden alle Schrittholzflächen manuell markiert. Die **GT**-Bilder dienen zusammen mit den Evaluierungsbildern in Abbildung C.3 zur Evaluierung der Segmentierung und zur Visualisierung der Holzfarbverteilung.

C Visualisierung und Evaluierungsdaten

Die 71 Evaluierungsbilder sind von verschiedener Qualität (Q_{st}, Q_I) und haben verschiedene Bildgrößen. Die Varianz in der Qualität und Bildgröße ist beabsichtigt, damit eine aussagekräftige Evaluierung hinsichtlich der Robustheit der Algorithmen ermöglicht wird. Die Bilderanzahl pro Qualitätsstufe der Stammschnittflächenqualität Q_{st} und der Qualität der Bildaufnahme und/oder des Bildes Q_I ist:

Q_I : A (gut): 6; B (mittelmäßig): 53; C (schlecht): 12

Q_{st} : A (gut): 33; B (mittelmäßig): 38; C (schlecht): 0

In Abbildung C.4 sind einige Evaluierungsbilder und die korrespondierenden **GT**-Bilder verschiedener Qualität zu sehen.



(a) I_{wood} mit $Q_I = B$



(b) I_{wood} mit $Q_{st} = B$



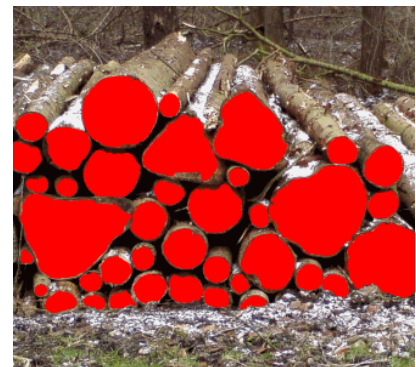
(c) I_{wood} mit $Q_{st} = C$



(d) GT-Bild zu Abb. C.4(a)



(e) GT-Bild zu Abb. C.4(b)



(f) GT-Bild zu Abb. C.4(c)

Abbildung C.4: Evaluierungsbilder (I_{wood}) verschiedener Qualität ((Q_{st}, Q_I)) und die zugehörigen **GT**-Bilder.

Die Weite und Höhe der Bilder I_{wood} variiert. Das größte Bild I_{wood} hat 2560×1920 und das kleinste 586×664 Pixel. Die Höhe h und Weite w von I_{wood} liegen im Intervall:

$$w \in [586, 2560]; \quad h \in [664, 1920]; \quad w, h \in \mathbb{N} \quad (\text{C.1})$$

C.2 Evaluierungsbilder und deren Grundwahrheit

In Abbildung C.5 sind insgesamt 75 Evaluierungsbilder (I_{og}) weiterer **organo**-Gruppen zu sehen. Die Evaluierungsbilder bestehen aus je 25 Bildern von Plattfischen, Kartoffeln und Äpfeln auf verschiedenen Hintergründen.

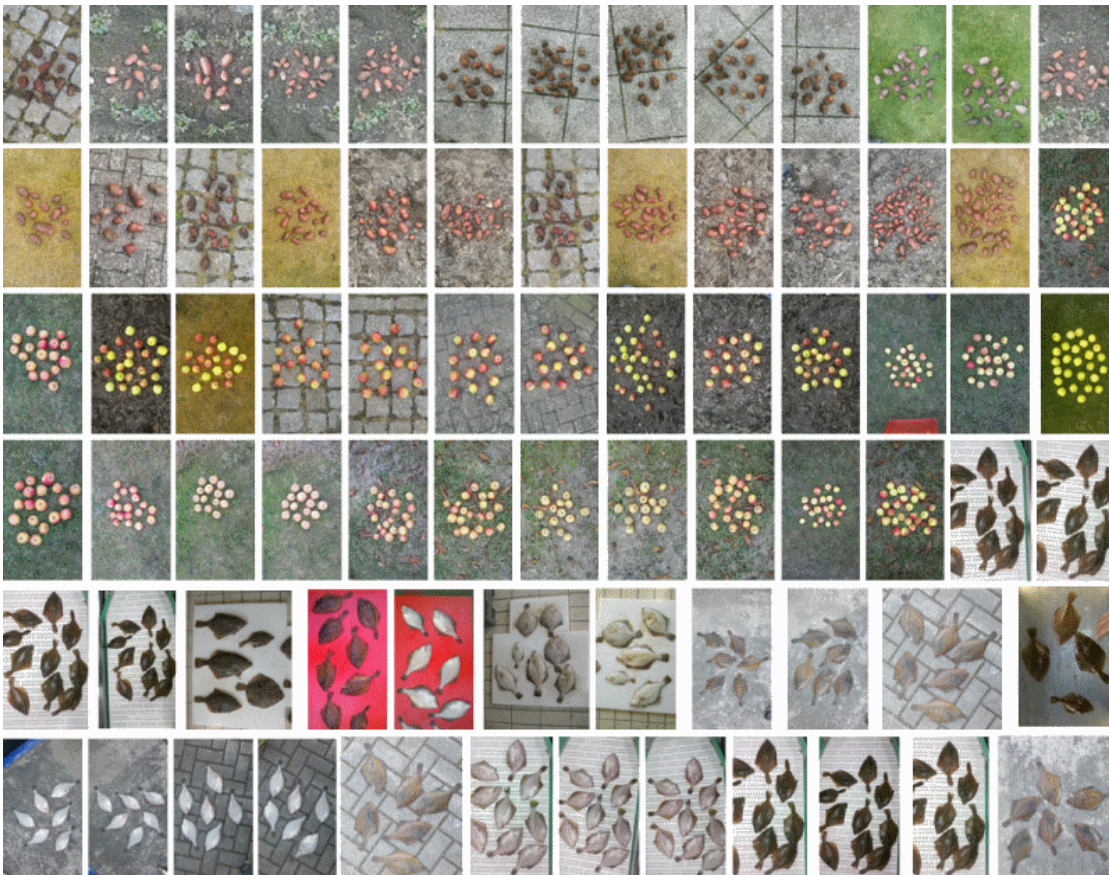


Abbildung C.5: Abgebildet sind 75 Evaluierungsbilder (I_{og}) der **organo**-Gruppen Kartoffeln, Äpfel und Plattfische. Auf Basis der Bilder wurden entsprechende **GT**-Bilder erstellt.

C.3 Praxisdaten AFoRS

Die Praxisevaluierung umfasst eine Evaluierung der AFoRS-Ergebnisse von zwei Referenzpoltern mit insgesamt 22 Aufnahmen (siehe Tabellen C.1, C.2) und von 52 Praxispoltern mit insgesamt 90 Aufnahmen (siehe Tabelle C.3).

Referenzpolter

Die Datentabelle C.1 zeigt die Ergebnisse des Volumens, der Stammanzahl und den relativen Fehler zu den Ground-Truth-Daten. Die Spalte **Bez.**¹ enthält eine verschlüsselte Bezeichnung des Polters. **S** steht für die Seite der Polterstirnfläche (V-Vorderseite, R-Rückseite), **FM** für Festmeter, **S.An.** für Anzahl der Stämme, **RM1** für Raummeter der alternativen Hülle und **RM2** der konvexen Hülle.

Tabelle C.1: Berechnungen des Volumens, Stammanzahl und rel. Fehler (14.10.2012)

Polter		Berechnungen in m^3				relativer Fehler in %			
Bez.	S	FM	RM1	RM2	S.An.	FM	RM1	RM2	S.An.
<i>1. Referenzpolter mit 56,42 Fm (Werk), 88,7 Rm (Netto) und 330 Stämmen</i>									
P75.B.1	V	55,98	89,11	99,77	333	-0,78	0,46	12,48	0,91
P75.B.2	V	56,92	89,15	101,33	335	0,88	0,51	14,23	1,52
P75.C.1	R	58,14	93,62	103,79	326	3,04	5,55	17,01	-1,21
P75.C.2	R	55,32	91,66	103,27	320	-1,94	3,34	16,42	-3,03
P75.C.1	V	48,84	82,91	92,57	318	-13,43	-6,53	4,36	-3,64
P75.C.2	V	53,45	86,22	96,85	332	-5,26	-2,80	9,19	0,61
P75.C.3	V	52,68	86,88	96,81	326	-6,63	-2,06	9,14	-1,21
P75.D.1	R	63,04	88,10	99,59	333	11,74	-0,67	12,28	0,91
P75.D.2	R	59,28	87,32	98,95	330	5,07	-1,56	11,55	0,00
P75.D.3	R	62,34	101,09	88,16	338	10,49	13,97	-0,61	2,42
P75.D.4	R	63,79	89,68	100,83	333	13,07	1,11	13,68	0,91
<i>2. Referenzpolter mit 9,504 Fm (Werk), 15,23 Rm (Netto) und 113 Stämmen</i>									
P76.B.1	R	10,82	17,88	15,79	124	9,77	17,43	3,68	9,73
P76.B.1	V	10,76	15,01	17,88	117	9,15	-1,47	17,39	3,54
P76.C.1	R	9,28	14,51	17,62	103	-5,87	-4,72	15,69	-8,85
P76.C.1	V	9,56	14,17	16,41	108	-2,99	-6,99	7,72	-4,42
P76.C.2	V	8,62	12,78	15,48	92	-12,62	-16,07	1,63	-18,58
P76.D.1	R	10,10	15,46	18,32	120	2,47	1,51	20,29	6,19
P76.D.2	R	9,69	14,68	17,55	109	-1,75	-3,58	15,21	-3,54
P76.D.1	V	10,37	14,15	16,70	111	5,17	-7,07	9,68	-1,77
P76.D.3	V	11,35	15,58	18,24	115	15,10	2,29	19,75	1,77

¹Verschlüsselung: POLTERBEZEICHNUNG.KAMERA.SERIE, wobei Kamera A dem Nokia N95 (Handy), B dem Sony Ericsson C705 (Handy), C dem Sony Ericsson C905 (Handy) und D der Sony Cybershot DSC-F707 (Digital-Kamera) entspricht

Bez.	S	FM	RM1	RM2	S.An.	FM	RM1	RM2	S.An.
P76.D.4	V	10,86	14,68	16,84	113	10,19	-3,63	10,54	0,00
P76.D.5	V	10,81	14,64	17,05	115	9,63	-3,87	11,95	1,77

Die Datentabelle C.2 zeigt die Ergebnisse zu den Stammklassen der zwei Referenzpolter sowie den relativen Fehler zu den Ground-Truth-Daten. Die Berechnungsergebnisse stellen die Anzahl der Stämme pro Stammklassen²(L1b, L2a, L2b, L3a, L3b) und die Anzahl der falsch zugeordneten Stämme **F** in den Stammklassen L0, L1a, L4 oder L5 dar.

Tabelle C.2: Berechnungen der Anzahl pro Stammklasse und rel. Fehler (14.10.2012)

Polter	Berechnung d. Stammklassen						relativer Fehler in %					
	Bez.	L1b	L2a	L2b	L3a	L3b	F	L1b	L2a	L2b	L3a	L3b
<i>1. Referenzpolter mit 169 L1b, 109 L2a, 40 L2b, 11 L3a und 1 L3b Stämmen</i>												
P75.B.1	157	82	27	16	2	49	-7,1	-24,8	-32,5	45,5	100	
P75.B.2	157	87	30	10	3	48	-7,1	-20,2	-25,0	-9,09	200	
P75.C.1	134	79	31	11	5	66	-20,7	-27,5	-22,5	0,0	400	
P75.C.2	142	72	28	12	4	62	-16	-33,9	-30,0	9,1	300	
P75.C.1	155	73	27	6	1	56	-8,3	-33,0	-32,5	-45,5	0	
P75.C.2	145	89	30	8	1	59	-14,2	-18,4	-25,0	-27,3	0	
P75.C.3	148	84	25	10	1	58	-12,4	-22,9	-37,5	-9,09	0	
P75.D.1	122	103	42	20	3	43	-27,81	-5,5	5,0	81,8	200	
P75.D.2	130	88	45	10	4	53	-23,1	-19,3	12,5	-9,1	300	
P75.D.3	125	99	44	16	4	50	-26,0	-9,2	10,0	45,5	300	
P75.D.4	134	96	44	14	4	41	-20,7	-11,9	10,0	27,3	300	
P75.D.5	143	107	36	17	1	38	-15,4	-1,8	-10,0	54,6	0	
<i>2. Referenzpolter mit 72 L1b, 31 L2a, 9 L2b und 1 L3a Stämmen</i>												
P76.B.1	44	28	5	3	-	44	-38,9	-9,7	-44,4	200	-	
P76.B.1	59	25	10	2	-	21	-18,1	-19,4	11,1	100	-	
P76.C.1	39	26	5	1	-	32	-45,8	-16,1	-44,4	0	-	
P76.C.1	50	23	5	1	-	29	-30,6	-25,8	-44,4	0	-	
P76.C.2	45	21	4	2	-	20	-37,5	-32,3	-55,6	100	-	
P76.D.1	49	23	8	1	-	28	-31,9	-25,8	-11,1	0	-	
P76.D.2	52	25	12	2	-	20	-27,8	-19,4	33,3	100	-	
P76.D.1	52	25	12	2	-	20	-27,8	-19,4	33,3	100	-	
P76.D.3	61	27	9	2	-	16	-15,3	-12,9	0,0	100	-	
P76.D.4	49	30	11	2	-	21	-31,9	-3,2	22,2	100	-	
P76.D.5	50	27	11	2	-	25	-30,6	-12,9	22,2	100	-	

²Stammklassen und zugehöriger Mittendurchmesser (ohne Rinde) nach der gesetzlichen Handelsklassensortierung für Rohholz (Stand vom 1.10.1983): L0 (unter 10 cm), L1a (10 bis 14cm), L1b (15 bis 19cm), L2a (20 bis 24cm), L2b (25 bis 29cm), L3a (30 bis 34cm), L3b (35 bis 39cm), L4 (40 bis 49cm), L5 (50 bis 59cm) und L6 (60 cm und mehr).

Praxispolter

Die Datentabelle C.3 zeigt die Ergebnisse von 52 Praxispoltern mit insgesamt 90 Aufnahmen und den relativen Fehler zu den Ground-Truth-Daten. Die Spalte **Bez.** enthält eine verschlüsselte Bezeichnung des Polters (siehe Abschnitt C.3). **B** steht für Anzahl der Bilder, **FM** für Festmeter, **S.An.** für Anzahl der Stämme, **RM1** für Raummeter der alternativen Hülle und **RM2** der konvexen Hülle.

Tabelle C.3: AFoRS-Berechnungen und relativer Fehler vom 5.10.2012.

Polter		Berechnungen in m^3				relativer Fehler in %			
Bez.	B	FM	RM1	RM2	S.An.	FM	RM1	RM2	S.An.
A1.A.1	7	108,27	187,98	207,22	756	-0,35	3,8	14,43	-3,94
A2.A.1	6	68,28	105,16	121,34	919	-13,15	-13,06	0,31	-11,63
A3.A.1	4	45,48	74,55	83,03	499	-3,8	-5,4	5,36	-8,61
A4.A.1	2	17,06	24,5	28,34	115	6,19	-8,52	5,82	-4,17
A5.A.1	3	24,5	34,06	37,34	203	10,16	-0,43	9,14	6,28
A6.A.1	4	52	85,11	95,18	444	-1,66	-3,43	8	1,37
A7.A.1	2	17,8	27,53	35,23	92	-4,59	-11,49	13,28	5,75
A8.A.1	3	85,05	123,37	138,54	226	22,28	3,31	16,01	0,89
A9.A.1	1	9,03	12,29	17,41	33	23	4,51	47,98	0
A10.A.1	2	26,35	40,9	50,91	89	60,45	55,44	93,49	12,66
A11.A.1	4	59,68	87,01	95,86	507	13,1	2,89	13,35	-4,16
A13.A.1	3	75,58	109,92	128,93	221	26,66	14,95	34,83	2,79
P1.B.1	2	5,69	9,5	13,05	54	-25,14	-25,04	2,99	-31,65
P1.C.1	2	8,03	13,12	18,04	97	5,65	3,55	42,42	22,78
P2.B.1	4	29,91	46,1	41,3	179	10,91	2,56	-8,12	55,65
P2.B.2	3	15,22	25,85	21,53	110	-43,57	-42,49	-52,1	-4,35
P2.C.1	4	29,42	40,39	45,56	190	9,07	-10,15	1,35	65,22
P2.C.2	3	15,54	21,83	25,7	128	-42,37	-51,44	-42,84	11,3
P3.B.1	5	47,07	82,22	98,09	490	-17,24	-13,27	3,47	-22,47
P4.B.1	1	5,14	8,3	10,73	55	0,83	-2,35	26,2	-14,06
P4.C.1	1	6,99	12,08	15,9	80	37,01	42,15	87,05	25
P5.B.1	2	6,96	11,14	15,07	73	-4,2	-7,96	24,54	-8,75
P5.B.2	2	7,26	11,78	15,98	76	0,06	-2,64	32,04	-5
P5.C.1	2	7,15	11,25	15,4	78	-1,48	-7,03	27,24	-2,5
P6.B.1	3	38,17	58,45	67,98	219	0,77	-7,41	57,63	-3,52
P6.C.1	3	40,24	59,05	67,43	240	6,25	-6,47	56,35	5,73
P7.B.1	4	28,96	49,23	56,21	655	-21,09	-19,52	-8,12	-4,38
P7.C.1	4	32,68	50,73	57,08	735	-10,97	-17,07	-6,68	7,3
P8.B.1	2	10,5	16,43	20,5	130	12,53	5,67	31,85	3,17
P8.C.1	2	11,27	16,63	20,41	144	20,77	6,96	31,23	14,29
P9.B.1	2	10,5	16,43	20,5	130	-2,63	-8,56	14,1	-9,09

Bez.	B	FM	RM1	RM2	S.An.	FM	RM1	RM2	S.An.
P9.C.1	2	13,34	18,04	20,45	147	23,76	0,4	13,83	2,8
P10.B.1	5	40,61	65,14	72,63	699	16,59	12,19	25,09	-3,45
P10.C.1	5	38,68	61,48	67,97	726	11,03	5,89	17,06	0,28
P11.B.1	2	12,38	16,4	19,97	73	-11,48	-26,84	-10,93	-1,35
P11.C.1	2	12,79	16,5	20,27	72	-8,59	-26,4	-9,6	-2,7
P12.B.1	1	8,62	12,01	14,83	119	10,95	-7,26	14,49	6,25
P12.C.1	1	9,35	13,22	15,68	135	20,3	2,06	21,07	20,54
P14.B.1	3	20,32	30,88	38,73	239	8,91	-0,71	24,53	-2,05
P14.C.1	3	21,01	32,24	39,75	256	12,62	3,67	27,8	4,92
P15.B.1	4	37,38	69,32	80,29	450	-22,62	-13,9	-0,28	-8,91
P15.C.1	4	37,68	68,84	82,4	470	-22,01	-14,5	2,33	-4,86
P16.B.1	2	24,78	39,12	51,87	145	-4,6	-6	24,64	5,84
P16.C.1	2	26,62	43,84	51,8	170	2,51	5,33	24,47	24,09
P18.B.1	2	9,62	14,36	19,14	84	6,06	-5,06	26,57	-5,62
P18.C.1	2	10,36	15,51	23,03	95	14,21	2,6	52,3	6,74
P19.B.1	2	5,84	8,37	13,58	37	-24,9	-35,38	4,79	-47,14
P19.C.1	2	8,03	12,85	17,68	85	3,26	-0,82	36,45	21,43
P20.B.1	6	64,68	111,1	125,59	763	-11,51	-8,8	3,09	-1,42
P20.C.1	6	75,35	120,07	133,21	864	3,09	-1,44	9,35	11,63
P21.B.1	2	6,92	10,21	13,12	80	14,42	1,29	30,12	1,27
P21.C.1	2	8,4	12,5	17,41	101	38,81	24,01	72,73	27,85
P22.B.1	2	11,69	14,43	20,67	42	80,98	39,39	99,66	13,51
P22.C.1	2	14,95	20,13	27,01	58	131,41	94,42	160,91	56,76
P23.B.1	4	50,4	69,02	75,65	229	12,07	-4,23	4,98	-0,43
P23.C.1	4	48,8	67,78	75,81	233	8,52	-5,94	5,19	1,3
P25.B.1	6	27,9	45,54	55,69	394	-24,12	-25,7	-9,14	-5,29
P25.C.1	6	30,7	49,26	60,67	459	-16,52	-19,63	-1	10,34
P26.B.1	9	108,04	167,64	184,26	1355	8,55	1,05	11,08	-3,35
P26.C.1	9	105,84	162,25	180,31	1458	6,33	-2,2	8,69	3,99
P27.B.1	4	31,11	43,01	49,91	151	-12	-24,07	-11,9	-0,66
P27.C.1	4	31,99	44,17	50,95	167	-9,52	-22,03	-10,06	9,87
P28.B.1	5	51,18	68,5	77,26	297	5,78	-11,65	-0,35	-3,26
P28.C.1	5	50,38	67,35	76,31	306	4,14	-13,13	-1,58	-0,33
P29.B.1	3	16,27	28,75	37,12	218	-0,35	5,65	36,42	-6,03
P29.C.1	3	19,83	34,53	45,3	260	21,46	26,92	66,49	12,07
P30.B.1	12	119,54	202,51	226,26	1723	-5,88	-4,33	6,89	-7,42
P30.C.1	12	129,65	210,45	230,49	2046	2,08	-0,58	8,88	9,94
P31.B.1	5	41,73	68,55	76,93	406	3,21	1,72	14,16	4,91
P31.B.2	5	43,13	72,42	83,46	397	6,66	7,47	23,84	2,58
P31.C.1	5	45,33	74,32	89,24	441	12,1	10,28	32,42	13,95
P32.B.1	4	35,83	59,57	67,19	308	14,18	13,9	28,47	0

C Visualisierung und Evaluierungsdaten

Bez.	B	FM	RM1	RM2	S.An.	FM	RM1	RM2	S.An.
P32.C.1	4	35,05	57,06	65,15	328	11,68	9,11	24,58	6,49
P34.B.1	6	71,46	117,28	137,87	526	18,83	17,02	37,57	-0,19
P36.B.1	2	12,72	21,62	25,07	124	-1,87	0,09	16,08	-6,06
P36.C.1	2	12,01	19,27	25,78	137	-7,33	-10,77	19,36	3,79
P37.B.1	5	36,74	56,69	68,92	274	-4,24	-11,35	7,79	4,98
P37.C.1	5	37,06	56,6	67,3	317	-3,39	-11,49	5,25	21,46
F1.B.1	4	28,41	44,41	51,64	708	3,22	-3,21	12,56	-7,33
F1.B.2	4	29,76	45,45	51,82	736	8,1	-0,93	12,95	-3,66
F1.B.3	4	31,34	47,71	54,13	732	13,86	3,98	17,98	-4,19
F2.B.1	2	4,94	6,32	8,43	46	5,82	-18,75	8,36	4,55
F2.B.2	2	4,87	6,24	8,39	44	4,42	-19,74	7,85	0
F3.B.1	4	27,09	64,47	84,1	1103	-41,41	-16,35	9,13	-34,35
F3.B.2	4	22,14	55,05	69,21	1016	-52,12	-28,57	-10,2	-39,52
F4.B.1	6	26,58	59,12	85,67	497	-57,48	-43,26	-17,79	-43,07
K1.B.1	6	71,9	133,86	145,71	2731	-10,4	0,08	8,94	-0,11
K2.B.1	5	44,98	73,45	81,87	899	17,91	15,53	28,77	-4,26
K3.B.1	8	75,24	165,22	196,99	1353	-40,32	-21,37	-6,25	-29,64
K4.B.1	8	12,6	23,92	32,66	74	-2,75	10,73	51,19	-29,52

D Entwickelte Algorithmen

D.1 GMC-Algorithmus

Der **GMC**-Algorithmus (*Growing Moving Circle*) sucht nach der besten Position von Kreisen in einem Binärbild I^b . Jeder Kreis wächst und bewegt sich zu einer Position mit der höchsten Energie. Mit Energie $e \in [0, 1]$ ist das Verhältnis zwischen Anzahl der segmentierten Holzpixel innerhalb des Kreises zur Kreisfläche gemeint. Es werden verschiedene Richtungen und Größen der Kreise getestet. Weiterhin wird jeweils eine Achter-Nachbarschaft mit einem zwei Pixel breiten Testbereich genutzt. Der Kreis wächst und bewegt sich, bis die Energie kleiner ist als ein gewisser Schwellwert. Nach dem Wachsen und Bewegen (Konvergenz) aller Kreise findet eine Verifikation der Kreise statt. Alle Kreise innerhalb eines anderen werden zu einem Kreis verschmolzen (maximale Überdeckung 50 %). Der nachfolgende Pseudocode verdeutlicht das prinzipielle Vorgehen.

```
1: list<circles> GMC(list<circles> lLogs, BinaryImage bImg, double e){
2:   double ec, el, er, eu, ed; //Energien (links, rechts, ...)
3:   double er2,el2,eu2,ed2; //Energien 2 Pixel weiter
4:   for(int i=0; i<lLogs.size(); i++){
5:     ec = 1.0;
6:     c = lLogs[i];
7:     while(e_c>e_min){
8:       eg = energy(c.x, c.y, c.r+1, bImg); //wächst
9:       el = energy(c.x-1, c.y, c.r, bImg); //1 Pixel links
10:      er = energy(c.x+1, c.y, c.r, bImg); //1 Pixel rechts
11:      eu = energy(c.x, c.y-1, c.r, bImg); //1 Pixel hoch
12:      ed = energy(c.x, c.y+1, c.r, bImg); //1 Pixel runter
13:      el2 = energy(c.x-2, c.y, c.r, bImg); //2 Pixel links
14:      er2 = energy(c.x+2, c.y, c.r, bImg); //2 Pixel rechts
15:      eu2 = energy(c.x, c.y-2, c.r, bImg); //2 Pixel hoch
16:      ed2 = energy(c.x, c.y+2, c.r, bImg); //2 Pixel runter
17:      ec = max(eg,er,el,eu,ed,er2,el2,eu2,ed2);
18:      c = growMoveCircle(c,eg,er,el,eu,ed,er2,el2,eu2,ed2);
19:    }
20:    lLogs[i] = c;
21:  }
22:  lLogs = improveMaxima(lLogs);
23:  return lLogs;
24: }
```

D.2 LSGMC-Algorithmus

Der nachfolgende Pseudocode beschreibt in Kurzform den **LSGMC**-Algorithmus (*Log Separation by Growing Moving Circles*). Die einzelnen Zeilen des Pseudocodes werden zur weiteren Beschreibung zu Grunde gelegt.

(11) Zuerst wird jeder Kreis mit der korrespondierenden Fläche in I^b verglichen. Wenn ein gewisses Verhältnis von Holz zu Nicht-Holzpixeln unterschritten ist, wird der Kreis entfernt. Das Ergebnis ist eine reduzierte Menge von Kreisen. (12) Zur Reduzierung von Löchern in dem Binärbild (Stammschnittflächenpixel) werden alle Kreise mit einem herunterskalierten Radius in das Binärbild I^b eingezeichnet. Das Ergebnis ist ein verbessertes Binärbild $I^{b'}$. (13) Um die besten angepassten Kreise zu erhalten, wird der erste **GMC** auf das verbesserte Binärbild $I^{b'}$ mit den reduzierten Kreisen aufgerufen. (14) Anschließend werden die angepassten Kreise aus dem Binärbild $I^{b'}$ entfernt ($l = 0$). Das resultierende Binärbild I_{rest}^b enthält die übrigbleibenden Holzpixel ($l = 1$). (15) Eine Extraktion der lokalen Maxima nach einer Distanz-Transformation auf I_{rest}^b führt zu einer neuen Kreismenge (Radius der Kreise ist identisch mit dem Distanzwert). (16) Die Vereinigung beider Kreismengen (angepasste und neue Menge) wird zum zweiten Mal (17) an das Binärbild $I^{b'}$ durch einen **GMC**-Aufruf angepasst.

```

1: list<circles> LSGMC(list<circles> lCand, BinaryImage bImg){
2:   list<circles> lLogs;
3:   list<circles> lLocalMax;
4:   BinaryImage bImgRest;
5:
6:   double a=0.45; //Verhältnis Holz zu Nicht-Holz
7:   double b=0.75; //Skalierung der Kreise für's Einzeichnen
8:   double c=0.6; //maximale Energie für den 1. GMC
9:   double d=0.85; //maximale Energie für den 2. GMC
10:
11:   lLogs = verify(lCand, bImg, a);
12:   bImg = drawFilledCircles(lLogs, bImg, b);
13:   lLogs = GMC(lLogs, bImg, c);
14:   bImgRest = removeCirclesFromImage(bImg, lLogs);
15:   lLocalMax = findMaxima(distanceTransform(bImgRest));
16:   lLogs = merge(lLogs, lLocalMax);
17:   lLogs = GMC(lLogs, bImg, d);
18:
19:   return lLogs;
10: }
```

Zusammenfassend werden durch den **LSGMC** alle ermittelten Kreise auf das Binärbild projiziert. Die Kreise wachsen und bewegen sich simultan, bis jedes Pixel einem Kreis zugeordnet werden kann.

D.3 GMC-G-Algorithmus

Der **GMC-G-Algorithmus** (*Growing Moving Circles Generalized*) ist eine Generalisierung des **LSGMC-Algorithmus** (siehe Abschnitt D.2) auf beliebige **organo-Gruppen**. Der Algorithmus erweitert den **LSGMC** um die Statistik erster Ordnung der Kreisradien und eine *Watershed*-Segmentierung auf dem Ausgangsbild. Außerdem werden beliebige Objekte und nicht nur Stammschnittflächen adressiert. Die ergänzten und geänderten Bestandteile werden anhand des nachfolgenden Pseudocodes und der Zeilennummern beschrieben. Hinzugekommen sind die Zeilen 17-21 und 26-27.

```

1: LImage GMCG(list<circles> lCand, BinaryImage bImg, RGBImg imgSrc){
2:   list<circles> lObj, lObjRest;           //Liste von Kreisen
3:   list<segments> lSegments;             //Liste von Segmenten
4:   BinaryImage bImgRest, bImgRestInv;    //Binärbilder
5:   pair<double, double> pObj_mv;        //Mittelwert und Varianz
6:   LImage imgMarker, imgResult;         //Labelbilder
7:
8:   double a=0.45; //Mindestverhältnis von Objekt zu Nicht-Objekt
9:   double b=0.75; //Skalierung der Kreise für's Einzeichnen
10:  double c=0.81; //maximale Energie für alle GMC's
11:  double d=0.95; //Skalierung des unsicheren Bereiches
12:
13:   lObj = verify(lCand, bImg, a);
14:   bImg = drawFilledCircles(lObj, bImg, b);
15:   lObj = GMC(lObj, bImg, c);
16:
17:   pObj_mv = getFirstOrderStatistic(lObj);
18:   bImgRest = removeCirclesFromImage(bImg, lObj);
19:   lSegments = segment(distanceTransform(bImgRest), pObj_mv);
20:   lObjRest = determineCirclesByMassCenterAndRadii(lSegments);
21:
22:   lObj = merge(lObj, lObjRest);
23:   lObj = GMC(lObj, bImg, c);
24:
25:   imgMarker = determineSeedPoints(bImg, lObj, d);
26:   imgResult = watershed(imgMarker, imgSrc);
27:
28:   return imgResult;
29: }
```

(13-15) Wie im **LSGMC-Algorithmus** werden die Kreise zuerst bereinigt, in das Binärbild eingezeichnet und mit dem **GMC-Algorithmus** in das Binärbild eingepasst. (17) Aus den angepassten Kreisen werden Statistiken erster Ordnung abgeleitet. (18) Die angepassten Kreise werden anschließend aus dem Binärbild entfernt. (19) Das reduzierte Binärbild wird invertiert und eine Distanztransformation durchgeführt. Anhand

der Statistiken erster Ordnung wird ein Schwellwert (minimaler Radius) ermittelt und das Distanzbild segmentiert. Nach der Segmentierung ergeben sich Segmente, welche in die Statistik der bisherigen Kreise passen und nicht detektiert wurden. (20) Aus den Segmenten werden anschließend Kreisparameter anhand des Schwerpunktes und der Segmentfläche bestimmt. (22-23) Wie bereits im **LSGMC**-Algorithmus wird ein zweiter **GMC** mit allen ermittelten Kreisen durchgeführt. Das Ergebnis ist eine Menge von optimierten Kreisen bzgl. dem Binärbild. (25) Aus dem Binärbild und den optimierten Kreisen wird ein Markerbild (Labelbild mit Saatpunkten) berechnet. Dazu wird für jeden optimierten Kreis ein einmaliges Label vergeben. Die Kreise werden herunterskaliert und mit dem entsprechenden Label ins Markerbild abgebildet. Weiterhin wird ein Hintergrundlabel abgebildet. Dazu wird eine Distanztransformation auf das Binärbild durchgeführt und das Ergebnis segmentiert. Alle segmentierten Pixel erhalten das Hintergrundlabel. (26) Im letzten und finalen Schritt wird eine *Watershed*-Segmentierung durchgeführt.

Zusammenfassend führt der **GMC-G** eine Multi-Objekt-Segmentierung auf einem Farbbild (RGBImg) durch, wobei Kreise und ein Binärbild der Objekte notwendig sind.

Abkürzungs- und Symbolverzeichnis

Abkürzungsverzeichnis

AAA	Begriff - Bedeutung (englische Bezeichnung)
AFoRS	Automatisierter Fotogrammetrischer Rohholz-Vermessungs-Service
bspw.	beispielsweise
bzgl.	bezüglich
bzw.	beziehungsweise
BK	Bayes Klassifikator
BV	Bildverarbeitung (Image Processing)
BP	Belief Propagation
CV	Maschinelles Sehen (Computer Vision)
d.h.	das heißt
DT	Entscheidungsbäume (Decision Trees)
EM	Erwartungswert Maximierung (Expectation Maximization)
FCM	Fuzzy C-Mean
FoE	Fields of Experts
GC	Graph-Cut
ggf.	gegebenenfalls
GK	Gauß Klassifikator
GLCM	Grauwertematrix (Gray Level Co-occurrence Matrix)
GMC	simultan wachsende Kreise (growing moving Circles); <i>eigener Ansatz</i>
GMC-G	GMC -generalisiert für organo -Gruppen; <i>eigener Ansatz</i>
GT	Grundwahrheit (Ground Truth)
GVF	Gradienten Vektor Feld (gradient vector field)
HoG	Histogramm von orientierten Gradienten (Histogram of oriented Gradients)
GMM	Gaussian Mixture Model
GMM-EM	Gaussian Mixture Models by Expectation Maximization
i.d.R.	in der Regel
KD-NN	Dichteabschätzung im KD-Baum mit Nächste Nachbarschaft; <i>eigener Ansatz</i>
KD-NN-A	KD-NN -adaptiert an Stammschnittflächendetektion; <i>eigener Ansatz</i>
KD-NN-G	KD-NN -generalisiert für organo -Gruppen; <i>eigener Ansatz</i>
LBP	Local Binary Pattern
LSGMC	GMC adaptiert an Baumstämme (Log Separation GMC); <i>eigener Ansatz</i>
MRF	Markov Random Fields
ML	Maschinelles Lernen (Machine Learning)
N-Cut	Normalized Cut
NN	Neuronale Netze

Abkürzungs- und Symbolverzeichnis

PCA	Prinzipielle Komponenten Analyse
PWL-H	Post Processed Wood Log Haarcascades; <i>eigener Ansatz</i>
RoI	Bereich von Interesse - Rechteckige Bildmarkierung (Region of Interest)
SGM	Single Gaussian Model
SLIC	Simple Linear Iterative Clustering
SVM	Support Vector Maschine
Trimap	Karte mit den drei Labels Unbekannt, Vorder- und Hintergrund
VioNeS	Visualisation of the Next Subvolume Method; <i>eigener Ansatz</i>
organo	gleichartige organische Objekte
WS-GC-R	Cut Wood Segmentation by Graph-Cut with Restrictions; <i>eigener Ansatz</i>
z.B.	zum Beispiel

Symbolverzeichnis

BG	Menge der Hintergrundpixel (background)
C	Kontur
cp	Kreisparameter und Tripel (x, y, r) ; Kreismittelpunkt: x, y ; Radius: r
CP	Menge von Kreisparametern
CPS	Menge von kreisförmigen Segmenten
D, d	Domain und ein Ort d der Domain
DS	Abstrakte Notation einer Datenstruktur
el	Eine Entität
FG	Menge der Vordergrundpixel (foreground)
f_k	Klassifikationsfunktion
f_l	allgemeine Labelfunktion
f_{od}	Objektdetektionsfunktion
f_{od}^{win}	fensterbasierte Objektdetektionsfunktion
f_s	Segmentierungsfunktion
f_s^{st}	Segmentierungsfunktion der Stammschnittflächensegmentierung
f_s^h	Segmentierungsfunktion der Schnittholzsegmentierung
$f_{s od}$	Segmentierungsfunktion, die zur Abbildung die Relation R_{OD} verwendet
I	Bild (Image) - Matrix und Menge von Pixeln
I^b	Binärbild; Ergebnis einer Binären Segmentierung
I_h^b	Binärbild mit Schnittholz- und nicht Schnittholzpixeln
I_{og}^b	Binärbild mit organischen Objekt- und nicht Objektpixeln
I^c	Bildkanal - Matrix und Menge von Pixelwerten
I^m	Bild mit n Labels; Ergebnis einer Multi-Objekt-Segmentierung
I_{st}^m	Bild mit Labels einzelner Stammschnittflächen OG^{st}
I_{og}^m	Bild mit Labels einzelner Organischer Objekte OG
I^{roi}	Bereich von Interesse zum Bild I
I^{smp}	Teilmenge von Pixeln (samples) des Bildes I
I^{sub}	Teilbild vom Bild I
I^{tri}	Trimap; Bild mit drei Labels $L = \{foreground, background, unknown\}$
I^t	Trainingsbild
I^p	generelles Wahrscheinlichkeitsbild
I_{fg}^p	Vordergrund (foreground) Wahrscheinlichkeitsbild
I_{bg}^p	Hintergrund (background) Wahrscheinlichkeitsbild
I_{wood}	Farbbild einer Holzpolterfrontseite
I_{og}	Farbbild mit einer organo -Gruppe
h, w	Höhe und Weite eines Bildes I
k	Eine Klasse
L, l	diskrete Labelmenge und ein Label
L^{st}, L^h	Labelmenge der Stammschnittflächen und Schnittholzsegmentierung
LD	Diskrete oder kontinuierliche Labelmenge
\vec{m}, bm	Merkmalsvektor und ein Bildmerkmal
O^r	reales 3D-Objekt

Abkürzungs- und Symbolverzeichnis

O	Bildobjekt - Menge von Pixeln
$O^{r,o}$	reales organisches 3D-Objekt
O^o	Organisches Bildobjekt - Menge von Pixeln
$O^{r,st}, O^{st}$	reale 3D-Stammschnittfläche und das 2D-Bildobjekt
\overline{O}	Menge aller nicht Bildobjekt Pixel O eines Bildes
\overline{O}^{st}	Menge aller nicht Stammschnittflächenpixel O^{st} eines Bildes
OG^r, OG	reale 3D- und abgebildete 2D-Gruppe gleichartiger organischer Objekte
OG^{st}	Menge aller Stammschnittflächen O^{st} im Bild
px	Pixel eines Bildes und n-Tupel von Werten va
S	ein Segment und Menge von Pixeln px
Q_{st}	Qualität der Stammschnittflächen; $Q_{st} \in \{A, B, C\}$
Q_I	Qualität der Bildaufnahme und/oder des Bildes; $Q_I \in \{A, B, C\}$
R_{od}	Ergebnisrelation der Objektdetektion
R_s	Ergebnisrelation der Segmentierung
S^{win}	ein rechteckiges Segment (<i>window</i>) und Menge von Pixeln px
SI	Segmentmenge - Menge aller möglichen Segmente S im Bild I
SI^{win}	Segmentmenge rechteckiger Segmente - Teilmenge von SI
va	Wert eines Pixels
win	Fenster und 4-Tupel (x, y, w, h) ; Koordinaten: x, y ; Weite: w ; Höhe: h
W	Menge von Fenstern win
WS	Menge von Teilbildern und rechteckigen Segmenten S^{win}
Θ	abstrakte Kennzeichnung eines Modells
Θ^{FG}	Vordergrundmodell
Θ^{BG}	Hintergrundmodell
Θ_{his}	Model zum Setzen der Gewichte mit einem 3D-Histogramms
Θ_{km}	Model zum Setzen der Gewichte mit <i>k-mean</i>
Θ_{gmm}	Model zum Setzen der Gewichte mit GMM-EM
Θ_{kn}	Model zum Setzen der Gewichte mit KD-NN
Λ	Indexmenge

Literaturverzeichnis

- [AB94] Rolf Adams und Leanne Bischof. “Seeded region growing”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16.6 (1994), S. 641–647. DOI: [10.1109/34.295913](https://doi.org/10.1109/34.295913).
- [Abm94] Wolfgang Abmayr. *Einführung in die digitale Bildverarbeitung*. B.G. Teubner Verlag Stuttgart, 1994. ISBN: 3519061384. URL: <http://www.worldcat.org/oclc/75438857>.
- [Ach+12] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua und Sabine Süsstrunk. “SLIC superpixels compared to state-of-the-art superpixel methods”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (2012), S. 2274–2282. DOI: [10.1109/tpami.2012.120](https://doi.org/10.1109/tpami.2012.120).
- [AF07] Asem M. Ali und Aly A. Farag. “Graph cut based segmentation of multi-modal images”. In: *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*. 2007, S. 1036–1041. DOI: [10.1109/ISSPIT.2007.4458212](https://doi.org/10.1109/ISSPIT.2007.4458212).
- [AF92] Hazem M. Abbas und Moustafa M. Fahmy. “A neural model for adaptive Karhunen Loeve transformation (KLT)”. In: *Proceedings of the International Joint Conference on Neural Networks*. Bd. 2. 1992, S. 975–980. DOI: [10.1109/IJCNN.1992.226861](https://doi.org/10.1109/IJCNN.1992.226861).
- [AHN90] Jaako Astola, Petri Haavisto und Yrjö Neuvo. “Vector median filters”. In: *Proceedings of the IEEE* 78.4 (1990), S. 678–689. DOI: [10.1109/5.54807](https://doi.org/10.1109/5.54807).
- [AJE10] K. Aiswarya, V. Jayaraj und D. Ebenezer. “A new and efficient algorithm for the removal of high density salt and pepper noise in images and videos”. In: *Computer Modeling and Simulation, 2010. ICCMS '10. Second International Conference on*. Bd. 4. 2010, S. 409–413. DOI: [10.1109/ICCMS.2010.310](https://doi.org/10.1109/ICCMS.2010.310).
- [Aki+12] Cihan Akin, Mervet Kirci, Ece Olcay Gunes und Yuksel Cakir. “Detection of the pomegranate fruits on tree using image processing”. In: *Proceedings of the International Conference on Agro-Geoinformatics*. 2012, S. 1–4. DOI: [10.1109/Agro-Geoinformatics.2012.6311724](https://doi.org/10.1109/Agro-Geoinformatics.2012.6311724).
- [Alo+12] Christopher Aloisio, Ranjan Kumar Mishra, Chu-Yin Chang und James English. “Next generation image guided citrus fruit picker”. In: *IEEE International Conference on Technologies for Practical Robot Applications*. 2012, S. 37–41. DOI: [10.1109/TePRA.2012.6215651](https://doi.org/10.1109/TePRA.2012.6215651).

- [Alp10] Ethem Alpaydin. *Introduction to Machine Learning*. 2nd. The MIT Press, 2010. ISBN: 026201243X, 9780262012430. URL: <http://www.worldcat.org/oclc/586143149>.
- [Ana+08] Christos-Nikolaos E. Anagnostopoulos, Ioannis E. Anagnostopoulos, Ioannis D. Psoroulas, Vassili Loumos und Eleftherios Kayafas. “License plate recognition from still images and video sequences: A survey”. In: *IEEE Transactions on Intelligent Transportation Systems* 9.3 (2008), S. 377–391. DOI: [10.1109/TITS.2008.922938](https://doi.org/10.1109/TITS.2008.922938).
- [AW92] Pertti Alapuranen und Tapani Westman. “Automatic visual inspection of wood surfaces”. In: *Proceedings of the International Conference on Pattern Recognition Vol.I. Conference A: Computer Vision and Applications*. 1992, S. 371–374. DOI: [10.1109/ICPR.1992.201578](https://doi.org/10.1109/ICPR.1992.201578).
- [BB11] Werner Baumeister und Heide von Berlepsch. *Der Forstwirt: Kuratorium für Waldarbeit und Forstwirtschaft*. Eugen Ulmer KG, 2011. ISBN: 3800155702. URL: <http://www.worldcat.org/oclc/730025610>.
- [Ben+08] Yannick Benezeth, Pierre-Marc Jodoin, Bruno Emile, Helene Laurent und Christophe Rosenberger. “Review and evaluation of commonly-implemented background subtraction algorithms”. In: *Proceedings of the International Conference on Pattern Recognition*. 2008, S. 1–4. DOI: [10.1109/ICPR.2008.4760998](https://doi.org/10.1109/ICPR.2008.4760998).
- [Ben75] Jon Louis Bentley. “Multidimensional binary search trees used for associative searching”. In: *Communications of the ACM* 18.9 (1975), S. 509–517. DOI: [10.1145/361002.361007](https://doi.org/10.1145/361002.361007).
- [Beu91] Serge Beucher. “The watershed transformation applied to image segmentation”. In: *Proceedings of the Scanning Microscopy International Conference*. 1991, S. 299–314. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.5229>.
- [Bez81] James C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Kluwer Academic Publishers, 1981. ISBN: 0306406713. URL: <http://www.worldcat.org/oclc/7459809>.
- [BJ01] Yuri .Y. Boykov und Marie-Pierre Jolly. “Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images”. In: *Proceedings of the IEEE International Conference on Computer Vision*. Bd. 1. 2001, S. 105–112. DOI: [10.1109/ICCV.2001.937505](https://doi.org/10.1109/ICCV.2001.937505).
- [BK04] Yuri Y. Boykov und Vladimir Kolmogorov. “An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.9 (2004), S. 1124–1137. DOI: [10.1109/TPAMI.2004.60](https://doi.org/10.1109/TPAMI.2004.60).

- [BL79] Serge Beucher und Christian Lantuejoul. “Use of watersheds in contour detection”. In: *Proceedings of the International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation*. 1979. URL: <http://cmm.ensmp.fr/~beucher/publi/watershed.pdf>.
- [Bla+05] *Automatic in situ identification of plankton*. Bd. 1. 2005, S. 79–86. DOI: [10.1109/ACVMOT.2005.29](https://doi.org/10.1109/ACVMOT.2005.29).
- [Bou95] Charles A. Bouman. *Markov random fields and stochastic image models*. Tutorial presented at the IEEE International Conference on Image Processing. 1995. URL: <http://www.robots.ox.ac.uk/~cvrg/hilary2002/mrf-tutorial.pdf>.
- [BP08] Julian Betancur und Flavio Prieto. “Active contour-based segmentation of coffee cherries images”. In: *Proceedings of the International Conference on Signal Processing*. 2008, S. 2738–2741. DOI: [10.1109/ICOSP.2008.4697714](https://doi.org/10.1109/ICOSP.2008.4697714).
- [Bru09] Roberto Brunelli. *Template matching techniques in computer vision : Theory and practice*. Wiley Publishing, 2009. ISBN: 0470517069, 9780470517062. URL: <http://www.worldcat.org/oclc/427566233>.
- [BTG06] Herbert Bay, Tinne Tuytelaars und Luc Van Gool. “Surf: Speeded up robust features”. In: *Proceedings of the European Conference on Computer Vision*. 2006, S. 404–417. DOI: [10.1007/11744023_32](https://doi.org/10.1007/11744023_32).
- [Bur83] Peter J. Burt. *The pyramid as a structure for efficient computation*. Techn. Ber. Image Processing Laboratory 38. Rensselaer Polytechnic Institute, 1983. URL: <http://www.worldcat.org/oclc/246491521>.
- [Can86] John Canny. “A computational approach to edge detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8.6 (1986), S. 679–698. DOI: [10.1109/TPAMI.1986.4767851](https://doi.org/10.1109/TPAMI.1986.4767851).
- [CM02] Dorin Comaniciu und Peter Meer. “Mean shift: A robust approach toward feature space analysis”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.5 (2002), S. 603–619. DOI: [10.1109/34.1000236](https://doi.org/10.1109/34.1000236).
- [CV01] Tony F. Chan und Luminita A. Vese. “Active contours without edges”. In: *IEEE Transactions on Image Processing* 10.2 (2001), S. 266–277. DOI: [10.1109/83.902291](https://doi.org/10.1109/83.902291).
- [CYP04] Qing Chen, Xiaoli Yang und Emil M. Petriu. “Watershed segmentation for binary images with different distance transforms”. In: *Proceedings of the IEEE International Workshop on Haptic, Audio and Visual Environments*. 2004. DOI: [10.1109/HAVE.2004.1391891](https://doi.org/10.1109/HAVE.2004.1391891).
- [DGM06] Anders Bjorholm Dahl, Meng Guo und Kristoffer Hougaard Madsen. “Scale-space and watershed segmentation for detection of wood logs”. In: *Poster Compendium at the Vision Day: Informatics and Mathematical Modelling*. 2006. URL: <http://www2.imm.dtu.dk/visiondag/VD06/posters/pdf/I13.pdf>.

- [DH72] Richard O. Duda und Peter E. Hart. “Use of the Hough transformation to detect lines and curves in pictures”. In: *Communications of the ACM* 15.1 (1972), S. 11–15. DOI: [10.1145/361237.361242](https://doi.org/10.1145/361237.361242).
- [DLR77] Arthur P. Dempster, Nan M. Laird und Donald B. Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of the Royal Statistical Society, Series B* 39.1 (1977), S. 1–38. URL: <http://www.jstor.org/stable/2984875>.
- [DLZ11] Jizhong Deng, Jiao Li und Xiangjun Zou. “Extraction of litchi stem based on computer vision under natural scene”. In: *Proceedings of the International Conference on Computer Distributed Control and Intelligent Environmental Monitoring*. 2011, S. 832–835. DOI: [10.1109/CDCIEM.2011.380](https://doi.org/10.1109/CDCIEM.2011.380).
- [DP85] Rina Dechter und Judea Pearl. “Generalized best-first search strategies and the optimality of A*”. In: *Journal of the ACM* 32.3 (1985), S. 505–536. DOI: [10.1145/3828.3830](https://doi.org/10.1145/3828.3830).
- [DT05] Navneet Dalal und Bill Triggs. “Histograms of oriented gradients for human detection”. In: *Proceedings of the International Conference on Computer Vision & Pattern Recognition*. Bd. 2. 2005, S. 886–893. DOI: [10.1109/CVPR.2005.177](https://doi.org/10.1109/CVPR.2005.177).
- [Dun73] Joseph C. Dunn. “A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters”. In: *Journal of Cybernetics* 3.3 (1973), S. 32–57. DOI: [10.1080/01969727308546046](https://doi.org/10.1080/01969727308546046).
- [DW92] Christopher E. Dunn und P. S. P. Wang. “Character segmentation techniques for handwritten text-A survey”. In: *Proceedings of the International Conference on Pattern Recognition Vol.II Conference B: Pattern Recognition Methodology and Systems*. 1992, S. 577–580. DOI: [10.1109/ICPR.1992.201844](https://doi.org/10.1109/ICPR.1992.201844).
- [Eff00] Nick Efford. *Digital image processing: A practical introduction using Java*. Includes CD-ROM. Addison-Wesley, 2000. ISBN: 0-201-59623-7. URL: <http://www.worldcat.org/oclc/43083285>.
- [EFS56] P. Elias, A. Feinstein und C. Shannon. “A note on the maximum flow through a network”. In: *IRE Transactions on Information Theory* 2.4 (1956), S. 117–119. DOI: [10.1109/tit.1956.1056816](https://doi.org/10.1109/tit.1956.1056816).
- [FH04] Pedro F. Felzenszwalb und Daniel P. Huttenlocher. “Efficient graph-based image segmentation”. In: *International Journal of Computer Vision* 59.2 (2004), S. 167–181. DOI: [10.1023/B:VISI.0000022288.19776.77](https://doi.org/10.1023/B:VISI.0000022288.19776.77).
- [FH06] Pedro F. Felzenszwalb und Daniel P. Huttenlocher. “Efficient belief propagation for early vision”. In: *International Journal of Computer Vision* 70.1 (2006), S. 41–54. DOI: [10.1007/s11263-006-7899-4](https://doi.org/10.1007/s11263-006-7899-4).
- [FH75] Keinosuke Fukunaga und Larry D. Hostetler. “The estimation of the gradient of a density function, with applications in pattern recognition”. In: *IEEE Transactions on Information Theory* 21.1 (1975), S. 32–40. DOI: [10.1109/TIT.1975.1055330](https://doi.org/10.1109/TIT.1975.1055330).

- [Fig98] Mario A. T. Figueiredo. *Bayesian methods and Markov random fields*. Tutorial presented at the Premier Annual Computer Vision Conference. 1998. URL: <http://www.lx.it.pt/~mtf/FigueiredoCVPR.pdf>.
- [Fin04] Florian Fink. “Foto-optische Erfassung der Dimension von Nadelrundholzabschnitten unter Einsatz digitaler bildverarbeitender Methoden”. Dissertation. Fakultät für Forst- und Umweltwissenschaften der Albert-Ludwigs-Universität Freiburg im Breisgau, 2004. URL: <http://www.freidok.uni-freiburg.de/volltexte/1371/>.
- [FLB93] Christine Fernandez-Maloigne, Daniele Laugier und Corinne Boscolo. “Detection of apples with texture analyse for an apple picker robot”. In: *Proceedings of the Symposium on Intelligent Vehicles*. 1993, S. 323–328. DOI: [10.1109/IVS.1993.697345](https://doi.org/10.1109/IVS.1993.697345).
- [For83] Forst-HKS. *Gesetzliche Handelsklassensortierung für Rohholz*. 1983. URL: http://forstbw.de/fileadmin/forstbw_pdf/gesetze_verordnungen/hks.pdf.
- [FP11] David Forsyth und Jean Ponce. *Computer vision: A modern approach*. Always learning. Pearson Education, Limited, 2011. ISBN: 9780136085928. URL: <http://www.worldcat.org/oclc/751787157>.
- [Fri13] Andreas Fricke. “Automatische Segmentierung und Vermessung von Muscheln auf Einzelbildern”. Betreuer: Enrico Gutzeit, Andreas Schmidt. Diplomarbeit. Fraunhofer IGD, Universität Rostock, 2013. URL: <http://publica.fraunhofer.de/dokumente/N-290905.html>.
- [Gor+10] Gaby Gorsky u. a. “Digital zooplankton image analysis using the ZooScan integrated system”. In: *Journal of Plankton Research* 32.3 (2010), S. 285–303. DOI: [10.1093/plankt/fbp124](https://doi.org/10.1093/plankt/fbp124).
- [GR12] Qi Gao und Stefan Roth. “How well do filter-based MRFs model natural images?” In: *Proceedings of the Joint Symposium of the German Association for Pattern Recognition and the Austrian Association for Pattern Recognition*. 2012, S. 62–72. DOI: [10.1007/978-3-642-32717-9_7](https://doi.org/10.1007/978-3-642-32717-9_7).
- [Gro+04] Philippe Grosjean, Marc Picheral, Caroline Warembourg und Gabriel Gorsky. “Enumeration, measurement, and identification of net zooplankton samples using the ZOOSCAN digital imaging system”. In: *ICES Journal of Marine Science* 61.4 (2004), S. 518–525. DOI: [10.1016/j.icesjms.2004.03.012](https://doi.org/10.1016/j.icesjms.2004.03.012).
- [GZZ10] Zhenhua Guo, Lei Zhang und David Zhang. “A completed modeling of local binary pattern operator for texture classification”. In: *Transactions on Image Processing* 19.6 (2010), S. 1657–1663. DOI: [10.1109/TIP.2010.2044957](https://doi.org/10.1109/TIP.2010.2044957).
- [Hai+04] Michal Haindl, Jiri Grim, Petr Somol, Pavel Pudil und Mineichi Kudo. “A Gaussian mixture-based colour texture model”. In: *Proceedings of the International Conference on Pattern Recognition*. Bd. 3. 2004, S. 177–180. DOI: [10.1109/ICPR.2004.1334497](https://doi.org/10.1109/ICPR.2004.1334497).

- [Han09] Heinz Handels. *Medizinische Bildverarbeitung*. Vieweg+Teubner Verlag / GWV Fachverlage GmbH, Wiesbaden, 2009. URL: <http://www.worldcat.org/oclc/426142851>.
- [Hat+96] Koichi Hata, Jun Ohya, Fumio Kishino und Ryohei Nakatsu. “Automatic extraction and tracking of contours”. In: *Proceedings of the International Conference on Pattern Recognition*. Bd. 1. 1996, S. 441–445. DOI: [10.1109/ICPR.1996.546065](https://doi.org/10.1109/ICPR.1996.546065).
- [HC71] John M. Hammersley und Peter E. Clifford. *Markov random fields on finite graphs and lattices*. Unpublished manuscript. 1971. URL: <http://www.statslab.cam.ac.uk/~grg/books/hammfest/hamm-cliff.pdf>.
- [HD95] Qian Huang und Byron E. Dom. “Quantitative methods of evaluating image segmentation”. In: *Proceedings of the International Conference on Image Processing*. Bd. 3. 1995, S. 53–56. DOI: [10.1109/ICIP.1995.537578](https://doi.org/10.1109/ICIP.1995.537578).
- [Her14] Christopher Herbon. *Wood log detection through classification and statistical modelling*. Techn. Ber. HAWK Fakultät Naturwissenschaften und Technik, 2014. URL: http://www.hawk-hhg.de/naturwissenschaften/media/WoodLogDetection_Herbon2014.pdf.
- [Heu08] Peter Heuer. *Art, Gattung, System: Eine logisch-systematische Analyse biologischer Grundbegriffe*. Symposion (Freiburg im Breisgau). Verlag Karl Alber, 2008. ISBN: 9783495483336. URL: <http://www.worldcat.org/oclc/261124704>.
- [Hoi10] Derek Hoiem. *Graph-based Segmentation*. Lecture Notes in Computer Vision, University of Illinois. 2010. URL: http://www.cs.illinois.edu/~dhoiem/courses/vision_spring10/lectures/Lecture12%20-%20Graph-based%20Segmentation.pdf.
- [Hou62] Paul V. C. Hough. *Method and means for recognizing complex patterns*. U.S. Patent 3.069.654. 1962. URL: <http://www.google.com/patents/US3069654>.
- [HP74] Steven L. Horowitz und Theo Pavlidis. “Picture segmentation by a directed split-and-merge procedure”. In: *Proceedings of the International Joint Conference on Pattern Recognition*. 1974, S. 424–433. URL: <http://www.citeulike.org/user/foleyfresh/article/9674913>.
- [HS79] Gregory M. Hunter und Kenneth Steiglitz. “Operations on images using quad trees”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1.2 (1979), S. 145–153. DOI: [10.1109/TPAMI.1979.4766900](https://doi.org/10.1109/TPAMI.1979.4766900).
- [HSD73] Robert M. Haralick, Karthikeyan Shanmugam und Its’Hak Dinstein. “Textural features for image classification”. In: *IEEE Transactions on Systems, Man and Cybernetics* SMC-3.6 (1973), S. 610–621. DOI: [10.1109/TSMC.1973.4309314](https://doi.org/10.1109/TSMC.1973.4309314).

- [Hsu+10] Roy Chaoming Hsu, Ping-Wen Kao, Wei-Jie Lai und Cheng-Ting Liu. “An initial edge point selection and segmental contour following for object contour extraction”. In: *Proceedings of the International Conference on Control Automation Robotics Vision*. 2010, S. 1632–1637. DOI: [10.1109/ICARCV.2010.5707892](https://doi.org/10.1109/ICARCV.2010.5707892).
- [HTF01] Trevor Hastie, Robert Tibshirani und Jerome Friedman. *The elements of statistical learning, data mining, inference, and prediction*. Springer-Verlag, 2001. URL: <http://www.worldcat.org/oclc/46809224>.
- [HW79] John A. Hartigan und Manchek A. Wong. “Algorithm AS 136: A k-means clustering algorithm”. In: *Applied Statistics* 28.1 (1979), S. 100–108. DOI: [10.2307/2346830](https://doi.org/10.2307/2346830).
- [HWJ05] Zhu Hao, Liu Wenyao und Wang Jintao. “Implementation of a novel watershed algorithm”. In: *Proceedings of the IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*. Bd. 2. 2005, S. 1150–1153. DOI: [10.1109/MAPE.2005.1618125](https://doi.org/10.1109/MAPE.2005.1618125).
- [HWZ11] Li He, Hui Wang und Hong Zhang. “Object detection by parts using appearance, structural and shape features”. In: *Proceedings of the International Conference on Mechatronics and Automation*. 2011, S. 489–494. DOI: [10.1109/ICMA.2011.5985611](https://doi.org/10.1109/ICMA.2011.5985611).
- [HZ83] Robert A. Hummel und Steven W. Zucker. “On the foundations of relaxation labeling processes”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-5.3 (1983), S. 267–287. DOI: [10.1109/TPAMI.1983.4767390](https://doi.org/10.1109/TPAMI.1983.4767390).
- [Jah05] Bernd Jähne. *Digital Image Processing*. 6th reviewed and extended edition. Springer Verlag, 2005. URL: <http://www.worldcat.org/oclc/262680307>.
- [Jen01] Henrik Wann Jensen. *Realistic Image Synthesis Using Photon Mapping*. The Morgan Kaufmann Series in Computer Graphics, 2001. URL: <http://www.worldcat.org/oclc/46975116>.
- [JKS95] Ramesh Jain, Rangachar Kasturi und Brian G. Schunck. *Machine vision*. McGraw-Hill, Inc., 1995. ISBN: 0-07-032018-7. URL: <http://www.worldcat.org/oclc/31933939>.
- [Kan+02a] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman und Angela Y. Wu. “A Local Search Approximation Algorithm for K-means Clustering”. In: *Proceedings of the Annual Symposium on Computational Geometry*. 2002, S. 10–18. DOI: [10.1145/513400.513402](https://doi.org/10.1145/513400.513402).
- [Kan+02b] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman und Angela Y. Wu. “An Efficient k-Means Clustering Algorithm: Analysis and Implementation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.7 (2002), S. 881–892. DOI: [10.1109/TPAMI.2002.1017616](https://doi.org/10.1109/TPAMI.2002.1017616).

- [KG08] Deepali Kelkar und Surendra Gupta. “Improved quadtree method for split merge image segmentation”. In: *Proceedings of the International Conference on Emerging Trends in Engineering and Technology*. 2008, S. 44–47. DOI: [10.1109/ICETET.2008.145](https://doi.org/10.1109/ICETET.2008.145).
- [Kim+11] Taesup Kim, Sebastian Nowozin, Pushmeet Kohli und Chang D. Yoo. “Variable grouping for energy minimization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2011, S. 1913–1920. DOI: [10.1109/CVPR.2011.5995645](https://doi.org/10.1109/CVPR.2011.5995645).
- [Kon+10] De-yuan Kong, De-an Zhao, Ying Zhang, Jin-jing Wang und Hai-xia Zhang. “Research of apple harvesting robot based on least square support vector machine”. In: *Proceedings of the International Conference on Electrical and Control Engineering*. 2010, S. 1590–1593. DOI: [10.1109/iCECE.2010.391](https://doi.org/10.1109/iCECE.2010.391).
- [Kot00] Ulrich Köthe. “Generische Programmierung für die Bildverarbeitung”. Dissertation. Universität Hamburg, 2000. URL: <http://www.worldcat.org/oclc/174936456>.
- [KPS03] Jure Kovac, Peter Peer und Franc Solina. “Human skin color clustering for face detection”. In: *Proceedings of the International Conference on Computer as a Tool*. Bd. 2. 2003, S. 144–148. DOI: [10.1109/EURCON.2003.1248169](https://doi.org/10.1109/EURCON.2003.1248169).
- [KS08] Rolf G. Kuehni und Andreas Schwarz. *Color ordered: A survey of color order systems from antiquity to the present*. Oxford University Press, 2008. ISBN: 9780195189681. URL: <http://www.worldcat.org/oclc/86113128>.
- [KWT88] Michael Kass, Andrew Witkin und Demetri Terzopoulos. “Snakes: Active contour models”. In: *International Journal of Computer Vision* 1.4 (1988), S. 321–331. DOI: [10.1007/BF00133570](https://doi.org/10.1007/BF00133570).
- [Lag11] Robert Laganiere. *OpenCV 2 computer vision application programming cookbook*. Packt Publishing, 2011. URL: <http://www.worldcat.org/oclc/748270051>.
- [Lan+06] Xiangyang Lan, Stefan Roth, Daniel Huttenlocher und Michael J. Black. “Efficient belief propagation with learned higher-order Markov random fields”. In: *Proceedings of the European Conference on Computer Vision Part II*. 2006, S. 269–282. DOI: [10.1007/11744047_21](https://doi.org/10.1007/11744047_21).
- [LBH08] Christoph H. Lampert, Matthew B. Blaschko und Thomas Hofmann. “Beyond sliding windows: Object localization by efficient subwindow search”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2008, S. 1–8. DOI: [10.1109/CVPR.2008.4587586](https://doi.org/10.1109/CVPR.2008.4587586).
- [LCC01] Ping-sung Liao, Tse-sheng Chen und Pau-choo Chung. “A fast algorithm for multilevel thresholding”. In: *Journal of Information Science and Engineering* 17.5 (2001), S. 713–727. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.85.3669>.

- [LCN12] Raphael Linker, Oded Cohen und Amos Naor. “Determination of the number of green apples in RGB images recorded in orchards”. In: *Computers and Electronics in Agriculture* 81 (2012), S. 45–57. DOI: [10.1016/j.compag.2011.11.007](https://doi.org/10.1016/j.compag.2011.11.007).
- [Li95] Stan Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag, 1995. ISBN: 4-431-70145-1. URL: <http://www.worldcat.org/oclc/32746694>.
- [Lia+11] Miao Liang, Xue Yue-ju, Kong De-yun, Huang Ke, Lu Qi-Fu und Wang kai. “A hybrid H component histogram threshold and sparse field level set algorithm for litchi image automatic segmentation”. In: *Proceedings of the International Conference on Electric Information and Control Engineering*. 2011, S. 1001–1004. DOI: [10.1109/ICEICE.2011.5777393](https://doi.org/10.1109/ICEICE.2011.5777393).
- [Lin96] Tony Lindeberg. “Scale-space: A framework for handling image structures at multiple scales”. In: *Proceedings of CERN School of Computing*, 1996. URL: <http://www.nada.kth.se/cvap/abstracts/lin96-csc.html>.
- [Llo82] Stuart P. Lloyd. “Least squares quantization in pcm”. In: *IEEE Transactions on Information Theory* 28.2 (1982), S. 129–137. DOI: [10.1109/TIT.1982.1056489](https://doi.org/10.1109/TIT.1982.1056489).
- [LM98] Thomas Leung und Jitendra Malik. “Contour continuity in region based image segmentation”. In: *Proceedings of the European Conference on Computer Vision*. 1998, S. 544–559. DOI: [10.1007/BFb0055689](https://doi.org/10.1007/BFb0055689).
- [Low99] David G. Lowe. “Object recognition from local scale-invariant features”. In: *Proceedings of the International Conference on Computer Vision*. Bd. 2. 1999, S. 1150–1157. DOI: [10.1109/ICCV.1999.790410](https://doi.org/10.1109/ICCV.1999.790410).
- [LT11] Juan Li und Yonglong Tan. “An effective segmentation method - Internal radius template method for quasi-round fruit”. In: *Proceedings of the International Conference on Electronic and Mechanical Engineering and Information Technology*. Bd. 9. 2011, S. 4546–4549. DOI: [10.1109/EMEIT.2011.6024041](https://doi.org/10.1109/EMEIT.2011.6024041).
- [Lv+11] Jidong Lv, De-an Zhao, Wei Ji, Yu Chen und Huiliang Shen. “Design and research on vision system of apple harvesting robot”. In: *Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics*. Bd. 1. 2011, S. 177–180. DOI: [10.1109/IHMSC.2011.49](https://doi.org/10.1109/IHMSC.2011.49).
- [LW13] Junli Li und Xiuying Wang. “A fast 3D euclidean distance transformation”. In: *Proceedings of the International Congress on Image and Signal Processing*. Bd. 2. 2013, S. 875–879. DOI: [10.1109/CISP.2013.6745288](https://doi.org/10.1109/CISP.2013.6745288).
- [LWG10] Jia-Hong Lee, Mei-Yi Wu und Zhi-Cheng Guo. “A tank fish recognition and tracking system using computer vision techniques”. In: *Proceedings of the IEEE International Conference on Computer Science and Information Technology*. Bd. 4. 2010, S. 528–532. DOI: [10.1109/ICCSIT.2010.5563625](https://doi.org/10.1109/ICCSIT.2010.5563625).

- [LZW08] Yinqing Li, Xiaodong Zheng und Xiaojie Wang. “Fruit discrimination on region feature”. In: *Proceedings of the International Conference on Fuzzy Systems and Knowledge Discovery*. Bd. 1. 2008, S. 590–594. DOI: [10.1109/FSKD.2008.5](https://doi.org/10.1109/FSKD.2008.5).
- [Mao+09] Wenhua Mao, Baoping Ji, Jicheng Zhan, Xiaochao Zhang und Xiaoan Hu. “Apple location method for the apple harvesting robot”. In: *Proceedings of the International Congress on Image and Signal Processing*. 2009, S. 1–5. DOI: [10.1109/CISP.2009.5305224](https://doi.org/10.1109/CISP.2009.5305224).
- [Mat+10] Rainer Matyssek, Jörg Fromm, Heinz Rennenberg und Andreas Roloff. *Biologie der Bäume: von der Zelle zur globalen Ebene*. Ulmer, 2010. ISBN: 9783800128402. URL: <http://www.worldcat.org/oclc/693876003>.
- [McG+07] Kevin McGuinness, Gordon Keenan, Tomasz Adamek und Noel E. O’Connor. “Image segmentation evaluation using an integrated framework”. In: *Proceedings of the International Conference on Visual Information Engineering*. 2007. URL: <http://doras.dcu.ie/270/>.
- [MD92] P. Moon und G. De Jager. “An heuristic graph searching algorithm to find the boundary of apple images”. In: *Proceedings of the South African Symposium on Communications and Signal Processing*. 1992, S. 233–238. DOI: [10.1109/COMSIG.1992.274278](https://doi.org/10.1109/COMSIG.1992.274278).
- [Meh+10] Madhur Mehta, Chandni Goyal, M. C. Srivastava und R. C. Jain. “Real time object detection and tracking: Histogram matching and Kalman filter approach”. In: *Proceedings of the International Conference on Computer and Automation Engineering*. Bd. 5. 2010, S. 796–801. DOI: [10.1109/ICCAE.2010.5451872](https://doi.org/10.1109/ICCAE.2010.5451872).
- [MFD92] P. Medina-Rodriguez, E. Fernandez-Garcia und Aranzazu Diaz-Urrestarazu. “Adaptive method for image segmentation based in local feature”. In: *Cybernetics and Systems* 23.3-4 (1992), S. 299–312. DOI: [10.1080/01969729208927463](https://doi.org/10.1080/01969729208927463).
- [MK82] Hiroyasu Murakami und B. V. K. Vijaya Kumar. “Efficient calculation of primary images from a set of images”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 4.5 (1982), S. 511–515. DOI: [10.1109/TPAMI.1982.4767295](https://doi.org/10.1109/TPAMI.1982.4767295).
- [MN78] D. Marr und H. K. Nishihara. “Representation and recognition of the spatial organization of three-dimensional shapes”. In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 200.1140 (1978), S. 269–294. DOI: [10.1098/rspb.1978.0020](https://doi.org/10.1098/rspb.1978.0020).
- [Mor+05] Erikson F. Morais, Mario F. M. Campos, Flavio L. C. Padua und Rodrigo L. Carceroni. “Particle filter-based predictive tracking for robust fish counting”. In: *Proceedings of the Brazilian Symposium on Computer Graphics and Image Processing*. 2005, S. 367–374. DOI: [10.1109/SIBGRAPI.2005.36](https://doi.org/10.1109/SIBGRAPI.2005.36).

- [Mov08] Javier R. Movellan. *Tutorial on gabor filters*. Tutorial of the Machine Perception Laboratory. 2008. URL: <http://mplab.ucsd.edu/tutorials/gabor.pdf>.
- [MS08] Mohammed Nordin Mokti und Rosalina Abdul Salam. “Hybrid of Mean-shift and median-cut algorithm for fish segmentation”. In: *Proceedings of the International Conference on Electronic Design*. 2008, S. 1–5. DOI: [10.1109/ICED.2008.4786645](https://doi.org/10.1109/ICED.2008.4786645).
- [MTK96] K.F. Man, K.S. Tang und S. Kwong. “Genetic algorithms: Concepts and applications [in engineering design]”. In: *IEEE Transactions on Industrial Electronics* 43.5 (1996), S. 519–534. DOI: [10.1109/41.538609](https://doi.org/10.1109/41.538609).
- [Nal+07] Krishnan Nallaperumal, Subban Ravi, C. Nelson Kennedy Babu, R. K. Selvakumar, A. Lenin Fred, Christopher Seldev und S. S. Vinsley. “Skin detection using color pixel classification with application to face detection: A comparative study”. In: *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications*. Bd. 3. 2007, S. 436–441. DOI: [10.1109/ICCIMA.2007.208](https://doi.org/10.1109/ICCIMA.2007.208).
- [Nal93] Vishvjit S. Nalwa. *A guided tour of computer vision*. Addison-Wesley, 1993. ISBN: 978-0-201-54853-2. URL: <http://www.worldcat.org/oclc/25712970>.
- [Nil71] Nils J. Nilsson. *Problem-solving methods in artificial intelligence*. McGraw-Hill Pub. Co., 1971. ISBN: 0070465738. URL: <http://www.worldcat.org/oclc/133169>.
- [Nil98] Nils J. Nilsson. *Introduction to machine learning*. Notes of a Stanford course. 1998. URL: <http://robotics.stanford.edu/~nilsson/mlbook.html>.
- [Nis+11] Alfred Nischwitz, Max Fischer, Peter Haberäcker und Gudrun Socher. *Computergrafik und Bildverarbeitung: Band II: Bildverarbeitung*. Vieweg+Teubner Verlag, 2011. ISBN: 9783834817129. URL: <http://www.worldcat.org/oclc/798967554>.
- [OF03] Stanley J. Osher und Ronald P. Fedkiw. *Level set methods and dynamic implicit surfaces*. Springer-Verlag, 2003. URL: <http://www.worldcat.org/oclc/53224633>.
- [OP97] Timo Ojala und Matti Pietikäinen. “Unsupervised texture segmentation using feature distributions”. In: *Proceedings of the International Conference on Image Analysis and Processing*. Bd. 1. 1997, S. 311–318. DOI: [10.1007/3-540-63507-6_216](https://doi.org/10.1007/3-540-63507-6_216).
- [Ots79] Nobuyuki Otsu. “A threshold selection method from gray-level histograms”. In: *IEEE Transactions on Systems, Man and Cybernetics*. 1979, S. 62–66. DOI: [10.1109/TSMC.1979.4310076](https://doi.org/10.1109/TSMC.1979.4310076).
- [PB07] Bernhard Preim und Dirk Bartz. *Visualization in medicine: Theory, algorithms, and applications*. Morgan Kaufmann Publishers Inc., 2007. ISBN: 0123705967, 9780080549057. URL: <http://www.worldcat.org/oclc/173483828>.

- [PCB01] Son Lam Phung, Douglas Chai und Abdesselam Bouzerdoum. “A universal and robust human skin color model using neural networks”. In: *Proceedings of the International Joint Conference on Neural Networks*. Bd. 4. 2001, S. 2844–2849. DOI: [10.1109/IJCNN.2001.938827](https://doi.org/10.1109/IJCNN.2001.938827).
- [PCR10] Tania Pouli, Douglas W. Cunningham und Erik Reinhard. “Image statistics and their applications in computer graphics”. In: *Proceedings of the Eurographics State of the Art Report*. 2010, S. 83–112. URL: http://www.cs.bris.ac.uk/home/reinhard/papers/eg2010_tania.pdf.
- [PGC10] Eanes Torres Pereira, Herman Martins Gomes und Joao Marques de Carvalho. “Integral local binary patterns: A novel approach suitable for texture-based object detection tasks”. In: *Proceedings of the Conference on Graphics, Patterns and Images*. 2010, S. 201–208. DOI: [10.1109/SIBGRAPI.2010.35](https://doi.org/10.1109/SIBGRAPI.2010.35).
- [Pic04] Massimo Piccardi. “Background subtraction techniques: A review”. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. Bd. 4. 2004, S. 3099–3104. DOI: [10.1109/ICSMC.2004.1400815](https://doi.org/10.1109/ICSMC.2004.1400815).
- [PMS94] Alex Pentland, Baback Moghaddam und Thad Starner. “View-based and modular eigenspaces for face recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1994, S. 84–91. DOI: [10.1109/CVPR.1994.323814](https://doi.org/10.1109/CVPR.1994.323814).
- [Pow07] David Martin W. Powers. *Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation*. Techn. Ber. SIE-07-001. School of Informatics und Engineering, Flinders University, 2007. URL: <https://csem.flinders.edu.au/research/techreps/SIE07001.pdf>.
- [PP93] Nikhil R. Pal und Sankar K. Pal. “A review on image segmentation techniques”. In: *Pattern Recognition* 26.9 (1993), S. 1277–1294. DOI: [10.1016/0031-3203\(93\)90135-J](https://doi.org/10.1016/0031-3203(93)90135-J).
- [PRD96] Yin Ping, Wang Runsheng und Liang Diannong. “A new image segmentation approach based on linked pyramid”. In: *Proceedings of the International Conference on Signal Processing*. Bd. 2. 1996, S. 1118–1121. DOI: [10.1109/ICSIGP.1996.566290](https://doi.org/10.1109/ICSIGP.1996.566290).
- [Pre+07] William H. Press, Saul A. Teukolsky, William T. Vetterling und Brian P. Flannery. *Numerical recipes: The art of scientific computing*. 3rd Edition. Cambridge University Press, 2007. ISBN: 0521880688, 9780521880688. URL: <http://www.worldcat.org/oclc/4761224743>.
- [PWT12] Richard Palmer, Geoff West und Tele Tan. “Scale Proportionate Histograms of Oriented Gradients for Object Detection in Co-Registered Visual and Range Data”. In: *Proceedings of the International Conference on Digital Image Computing Techniques and Applications*. 2012, S. 1–8. DOI: [10.1109/DICTA.2012.6411699](https://doi.org/10.1109/DICTA.2012.6411699).

- [Rau+09] Shital Adarsh Raut, M. Raghuwanshi, R. Dharaskar und Adarsh Raut. “Image segmentation - A state-of-art survey for prediction”. In: *Proceedings of the International Conference on Advanced Computer Control*. 2009, S. 420–424. DOI: [10.1109/ICACC.2009.78](https://doi.org/10.1109/ICACC.2009.78).
- [RB05] Stefan Roth und Michael J. Black. “Fields of experts: A framework for learning image priors”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Bd. 2. 2005, S. 860–867. DOI: [10.1109/CVPR.2005.160](https://doi.org/10.1109/CVPR.2005.160).
- [REP05] Gonzalo A. Ruz, Pablo A. Estevez und Claudio A. Perez. “A neurofuzzy color image segmentation method for wood surface defect detection”. In: *Forest Products Journal* 55.4 (2005), S. 52–58. URL: http://www.gonzaloruz.com/GR/Gonzalo_Ruz_Homepage_files/fpj.pdf.
- [RGY10] Gao Rui, Liu Gang und Si Yongsheng. “A recognition method of apples based on texture features and EM algorithm”. In: *Proceedings of the World Automation Congress*. 2010, S. 225–229. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5665567.
- [RKB04] Carsten Rother, Vladimir Kolmogorov und Andrew Blake. “GrabCut - Interactive foreground extraction using iterated graph cuts”. In: *ACM Transactions on Graphics* 23.3 (2004), S. 309–314. DOI: [10.1145/1015706.1015720](https://doi.org/10.1145/1015706.1015720).
- [RM00] Jos B. T. M. Roerdink und Arnold Meijster. “The watershed transform: Definitions, algorithms and parallelization strategies”. In: *Fundamenta Informaticae* 41.1-2 (2000), S. 187–228. URL: <http://dl.acm.org/citation.cfm?id=2372488.2372495>.
- [RP66] Azriel Rosenfeld und John L. Pfaltz. “Sequential operations in digital picture processing”. In: *Journal of the ACM* 13.4 (1966), S. 471–494. DOI: [10.1145/321356.321357](https://doi.org/10.1145/321356.321357).
- [RTG00] Yossi Rubner, Carlo Tomasi und Leonidas J. Guibas. “The earth mover’s distance as a metric for image retrieval”. In: *International Journal of Computer Vision* 40.2 (2000), S. 99–121. DOI: [10.1023/A:1026543900054](https://doi.org/10.1023/A:1026543900054).
- [SAT11] Yahia Said, Mohamed Atri und Rached Tourki. “Human detection based on integral histograms of oriented gradients and SVM”. In: *Proceedings of the International Conference on Communications, Computing and Control Applications*. 2011, S. 1–5. DOI: [10.1109/CCCA.2011.6031422](https://doi.org/10.1109/CCCA.2011.6031422).
- [Sch83] Fritz Hans Schweingruber. *Der Jahrring: Standort, Methodik, Zeit und Klima in der Dendrochronologie*. P. Haupt, 1983. ISBN: 9783258031200. URL: <http://www.worldcat.org/oclc/10985435>.
- [SFM02] Mohsen Sharifi, Mahmoud Fathy und Maryam Tayefeh Mahmoudi. “A classified and comparative study of edge detection algorithms”. In: *Proceedings of the International Conference on Information Technology: Coding and Computing*. 2002, S. 117–120. DOI: [10.1109/ITCC.2002.1000371](https://doi.org/10.1109/ITCC.2002.1000371).

- [Sil+05] Marlon Crislei da Silvia, Vicente Paulo Soares, Francisco de Assis Carvalho Pinto, Carlos Pedro Boechat Soares und Carlos Antonio Alvares Soares Ribeiro. “Determination of the volume wooden stacked through processing of digital images”. In: *Scientia Forest Alis* 69 (2005), S. 104–114. URL: <http://www.ipef.br/publicacoes/scientia/nr69/cap09.pdf>.
- [SK87] L. Sirovich und M. Kirby. “Low-dimensional procedure for the characterization of human faces”. In: *Journal of the Optical Society of America A* 4.3 (1987), S. 519–524. DOI: [10.1364/JOSAA.4.000519](https://doi.org/10.1364/JOSAA.4.000519).
- [SK94] Wladyslaw Skarbek und Andreas Koschan. *Colour image segmentation - A survey*. Techn. Ber. 94 32. Fachbereich Informatik, Technische Universität Berlin, 1994. URL: <http://iristown.engr.utk.edu/~koschan/paper/coseg.pdf>.
- [SKS01] Christian Schellewald, Jens Keuchel und Christoph Schnörr. “Image labeling and grouping by minimizing linear functionals over cones”. In: *Proceedings of the International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*. 2001, S. 235–250. DOI: [10.1007/3-540-44745-8_18](https://doi.org/10.1007/3-540-44745-8_18).
- [SM00a] Heidrun Schumann und Wolfgang Müller. *Visualisierung: Grundlagen und Allgemeine Methoden*. Springer-Verlag, 2000. ISBN: 9783540649441. URL: <http://www.worldcat.org/oclc/76007695>.
- [SM00b] Jianbo Shi und Jitendra Malik. “Normalized cuts and image segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.8 (2000), S. 888–905. DOI: [10.1109/34.868688](https://doi.org/10.1109/34.868688).
- [Sri11] Umamahesh Srinivas. *Markov random fields*. Notes of the iPAL Group Meeting. 2011. URL: <http://signal.ee.psu.edu/mrf.pdf>.
- [SS04] Mehmet Sezgin und Bülent Sankur. “Survey over image thresholding techniques and quantitative performance evaluation”. In: *Journal of Electronic Imaging* 13.1 (2004), S. 146–168. DOI: [10.1117/1.1631315](https://doi.org/10.1117/1.1631315).
- [Ste06] Reiner Steinbrecher. *Bildverarbeitung in der Praxis*. 2nd Edition. R. Oldenbourg Verlag, 2004/2006. URL: http://www.rst-software.de/dbv/dbv_buch.html.
- [Ste56] Hugo Steinhaus. “Sur la division des corps matériels en parties”. In: *Bulletin L’Académie Polonaise des Science Classes III*. 4 (1956), S. 801–804. URL: <http://www.citeulike.org/user/thefillm/article/3738255>.
- [Sul+99] Josephine Sullivan, Andrew Blake, Michael Isard und John MacCormick. “Object localization by Bayesian correlation”. In: *Proceedings of the IEEE International Conference on Computer Vision*. Bd. 2. 1999, S. 1068–1075. DOI: [10.1109/ICCV.1999.790391](https://doi.org/10.1109/ICCV.1999.790391).
- [Sys08] Dralle A/S Cognitive Systems. *sScale - v2.00*. Product-Flyer. 2008. URL: <http://www.dralle.dk>.

- [Sze10] Richard Szeliski. *Computer vision: Algorithms and applications*. Springer-Verlag, 2010. ISBN: 1848829345, 9781848829343. URL: <http://www.worldcat.org/oclc/758141862>.
- [Ton05] Klaus D. Tönnies. *Grundlagen der Bildverarbeitung*. Pearson Studium, 2005. ISBN: 9783827371553. URL: <http://www.worldcat.org/oclc/76743710>.
- [TP91] Matthew Turk und Alex Pentland. “Eigenfaces for recognition”. In: *Journal of Cognitive Neuroscience* 3.1 (1991), S. 71–86. DOI: [10.1162/jocn.1991.3.1.71](https://doi.org/10.1162/jocn.1991.3.1.71).
- [Tsu+05] Hiroyuki Tsuji, Suguru Saito, Hiroki Takahashi und Masayuki Nakajima. “Estimating object contours from binary edge images”. In: *Proceeding of the IEEE International Conference on Image Processing*. Bd. 3. 2005, S. 453–456. DOI: [10.1109/ICIP.2005.1530426](https://doi.org/10.1109/ICIP.2005.1530426).
- [Umb05] Scott E. Umbaugh. *Computer imaging: Digital image analysis and processing*. Taylor & Francis, 2005. ISBN: 9780849329197. URL: <http://www.worldcat.org/oclc/56412981>.
- [VJ01] Paul A. Viola und Michael J. Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Bd. 1. 2001, S. 511–518. DOI: [10.1109/CVPR.2001.990517](https://doi.org/10.1109/CVPR.2001.990517).
- [VS08] Andrea Vedaldi und Stefano Soatto. “Quick shift and kernel methods for mode seeking”. In: *Proceedings of the European Conference on Computer Vision Part IV*. 2008, S. 705–718. DOI: [10.1007/978-3-540-88693-8_52](https://doi.org/10.1007/978-3-540-88693-8_52).
- [VSA03] Vladimir Vezhnevets, Vassili Sazonov und Alla Andreeva. “A survey on pixel-based skin color detection techniques”. In: *Proceedings of the International Conference on Computer Graphics and Vision*. 2003, S. 85–92. URL: <http://graphicon.ru/oldgr/en/publications/text/gc2003vsa.pdf>.
- [Wan+09] Jin-jing Wang, De-an Zhao, Wei Ji, Jun-jun Tu und Ying Zhang. “Application of support vector machine to apple recognition using in apple harvesting robot”. In: *Proceedings of the International Conference on Information and Automation*. 2009, S. 1110–1115. DOI: [10.1109/ICINFA.2009.5205083](https://doi.org/10.1109/ICINFA.2009.5205083).
- [Wij+08] Pavithra Wijethunga, Sandhya Samarasinghe, Don Kulasiri und Ian M. Woodhead. “Digital image analysis based automated kiwifruit counting technique”. In: *Proceedings of the International Conference on Image and Vision Computing New Zealand*. 2008, S. 1–6. DOI: [10.1109/IVCNZ.2008.4762149](https://doi.org/10.1109/IVCNZ.2008.4762149).
- [Wij+09] Pavithra Wijethunga, Sandhya Samarasinghe, Don Kulasiri und Ian M. Woodhead. “Towards a generalized colour image segmentation for kiwifruit detection”. In: *Proceedings of the International Conference on Image and Vision Computing New Zealand*. 2009, S. 62–66. DOI: [10.1109/IVCNZ.2009.5378361](https://doi.org/10.1109/IVCNZ.2009.5378361).

- [WSS06] Darren J. White, C. Svellingen und Norval J.C. Strachan. “Automated measurement of species and length of fish by computer vision”. In: *Fisheries Research* 80.2-3 (2006), S. 203–210. DOI: [10.1016/j.fishres.2006.04.009](https://doi.org/10.1016/j.fishres.2006.04.009).
- [Xie09] Zhenping Xie. “Some remarks on FCMLS and its application to natural fruit image segmentation”. In: *Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence*. Bd. 3. 2009, S. 173–177. DOI: [10.1109/AICI.2009.110](https://doi.org/10.1109/AICI.2009.110).
- [XP11] Shipeng Xie und Jing Pan. “Hand detection using robust color correction and Gaussian mixture model”. In: *Proceedings of the International Conference on Image and Graphics*. 2011, S. 553–557. DOI: [10.1109/ICIG.2011.166](https://doi.org/10.1109/ICIG.2011.166).
- [XP97] Chenyang Xu und Jerry L. Prince. “Gradient vector flow: A new external force for snakes”. In: *Proceedings of the Conference on Computer Vision and Pattern Recognition*. 1997, S. 66–71. DOI: [10.1109/CVPR.1997.609299](https://doi.org/10.1109/CVPR.1997.609299).
- [XP98] Chenyang Xu und Jerry L. Prince. “Snakes, shapes, and gradient vector flow”. In: *IEEE Transactions on Image Processing* 7.3 (1998), S. 359–369. DOI: [10.1109/83.661186](https://doi.org/10.1109/83.661186).
- [XXB11] Lv Xiao-lian, Lv Xiao-rong und Lu Bing-fu. “Identification and location of picking tomatoes based on machine vision”. In: *Proceedings of the International Conference on Intelligent Computation Technology and Automation*. Bd. 2. 2011, S. 101–107. DOI: [10.1109/ICICTA.2011.316](https://doi.org/10.1109/ICICTA.2011.316).
- [Yan+07] Linghe Yang, John K. Dickinson, Q. M. Jonathan Wu und Sherman Lang. “A fruit recognition method for automatic harvesting”. In: *Proceedings of the International Conference on Mechatronics and Machine Vision in Practice*. 2007, S. 152–157. DOI: [10.1109/MMVIP.2007.4430734](https://doi.org/10.1109/MMVIP.2007.4430734).
- [YFR04] Xia Yong, Dagan Feng und Zhao Rongchun. “Optimal selection of image segmentation algorithms based on performance prediction”. In: *Proceedings of the Pan-Sydney Area Workshop on Visual Information Processing*. 2004, S. 105–108. URL: <http://dl.acm.org/citation.cfm?id=1082140>.
- [YFW03] Jonathan S. Yedidia, William T. Freeman und Yair Weiss. “Exploring Artificial Intelligence in the New Millennium”. In: Hrsg. von Gerhard Lakemeyer und Bernhard Nebel. Morgan Kaufmann Publishers Inc., 2003. Kap. Understanding Belief Propagation and Its Generalizations, S. 239–269. URL: <http://www.worldcat.org/oclc/876320471>.
- [YKA02] Ming-hsuan Yang, David J. Kriegman und Narendra Ahuja. “Detecting faces in images: A survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.1 (2002). DOI: [10.1109/34.982883](https://doi.org/10.1109/34.982883).
- [YMX08] Jianjun Yin, Hanping Mao und Yongliang Xie. “Segmentation methods of fruit image and comparative experiments”. In: *Proceedings of the International Conference on Computer Science and Software Engineering*. Bd. 1. 2008, S. 1098–1102. DOI: [10.1109/CSSE.2008.1256](https://doi.org/10.1109/CSSE.2008.1256).

- [ZB02] Thomas Zöller und Joachim M. Buhmann. “Self-organized clustering of mixture models for combined color and texture segmentation”. In: *Proceedings of the International Workshop on Texture Analysis and Segmentation*. 2002, S. 163–167. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.3068>.
- [Zhu+06] Qiang Zhu u. a. “Fast human detection using a cascade of histograms of oriented gradients”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2006, S. 1491–1498. DOI: [10.1109/CVPR.2006.119](https://doi.org/10.1109/CVPR.2006.119).
- [Zhu+12] Lu-Min Zhu, Yong-Liang Zhang, Wen Zhang, Zhu-Cheng Tao und Chao-Fan Liu. “Fish motion tracking based on RGB color space and interframe global nearest neighbour”. In: *Proceedings of the International Conference on Automatic Control and Artificial Intelligence*. 2012, S. 1061–1064. DOI: [10.1049/cp.2012.1160](https://doi.org/10.1049/cp.2012.1160).
- [ZK07] C. Lawrence Zitnick und Sing Bing Kang. “Stereo for image-based rendering using image over-segmentation”. In: *International Journal of Computer Vision* 75.1 (2007), S. 49–65. DOI: [10.1007/s11263-006-0018-8](https://doi.org/10.1007/s11263-006-0018-8).
- [ZTK05] Jun Zhao, Joel Tow und Jayantha Katupitiya. “On-tree fruit recognition using texture properties and color data”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2005, S. 263–268. DOI: [10.1109/IROS.2005.1545592](https://doi.org/10.1109/IROS.2005.1545592).
- [ZZK10] Ying Zhang, De-an Zhao und Deyuan Kong. “Application of image segmentation algorithm based on entropy clustering in apple harvesting robot”. In: *Proceedings of the IEEE International Conference on Computer Science and Information Technology*. Bd. 6. 2010, S. 360–363. DOI: [10.1109/ICCSIT.2010.5563923](https://doi.org/10.1109/ICCSIT.2010.5563923).

Eigene Veröffentlichungen

- [AGU14] Mario Ähnelt, Enrico Gutzeit und Bodo Urban. “Using activity recognition for the tracking of assembly processes: Challenges and requirements”. In: *Proceedings of the Workshop on Activity Recognition*. 2014, S. 12–21.
- [Bie+09] Gerald Bieber, Andre Hofmeyer, Enrico Gutzeit, Christian Peter und Bodo Urban. “Activity monitoring by fusion of optical and mechanical tracking technologies for user behavior analysis”. In: *Proceedings of the ACM International Conference on Pervasive Technologies Related to Assistive Environments*. 2009, S. 1–6. DOI: [10.1145/1579114.1579159](https://doi.org/10.1145/1579114.1579159).
- [Ewa+06] Roland Ewald, Enrico Gutzeit, Sebastian Schwanke, Adelinde Uhrmacher, Christian Lange, Susanne Biermann und Carsten Maus. “Multi-level modeling with DEVS - A critical inspection and steps towards a feasible approach”. In: *Poster Compendium of the Winter Simulation Conference*. 2006. URL: https://dl.dropboxusercontent.com/u/9062065/uni/documents/misc_posters/poster_wsc06_ewaldetal.pdf.
- [Foe+12] Britt Foese, Enrico Gutzeit, Martin Müller und Jörg Voskamp. “AFoRS - Automatisierter Fotogrammetrischer Rohholz-Vermessungs-Service”. In: *Lecture Notes in Informatics: Informationstechnologie für eine nachhaltige Landbewirtschaftung*. 2012, S. 91–94. URL: <http://subs.emis.de/LNI/Proceedings/Proceedings194/article6571.html>.
- [GKS14] Enrico Gutzeit, Enrico Kloß und Andreas Schröder. “3D-Registrierung zum Soll-Ist-Vergleich auf Basis unterschiedlicher Geometriedateiformate”. In: *3D-NordOst 2014, Tagungsband 17. Anwendungsbezogener Workshop zur Erfassung Verarbeitung und Auswertung von 3D-Daten*. 2014, S. 43–52.
- [GL13] Enrico Gutzeit und Uwe von Lukas. “Methodik zur Segmentierung organischer Objekte einer gleichartigen Gruppe in Farbbildern”. In: *Proceedings of the Workshop Farbbildverarbeitung*. 2013, S. 135–145. URL: <http://publica.fraunhofer.de/dokumente/N-283706.html>.
- [Gut+10] Enrico Gutzeit, Stephan Ohl, Arjan Kuijper, Jörg Voskamp und Bodo Urban. “Setting graph cut weights for automatic foreground extraction in wood log images”. In: *Proceedings of the International Conference on Computer Vision Theory and Applications*. 2010, S. 60–67. URL: <http://publica.fraunhofer.de/documents/N-134554.html>.

Eigene Veröffentlichungen

- [Gut+11a] Enrico Gutzeit, Stephan Ohl, Jörg Voskamp, Arjan Kuijper und Bodo Urban. “Automatic wood log segmentation using graph cuts”. In: *Proceedings of the International Joint Conference on Computer Vision, Imaging and Computer Graphics*. 2011, S. 96–109. DOI: [10.1007/978-3-642-25382-9_7](https://doi.org/10.1007/978-3-642-25382-9_7).
- [Gut+11b] Enrico Gutzeit, Matthias Vahl, Zhiliang Zhou und Uwe von Lukas. “Skin cluster tracking and verification for hand gesture recognition”. In: *Proceedings of the International Symposium on Image and Signal Processing and Analysis*. 2011, S. 241–246. URL: <http://publica.fraunhofer.de/dokumente/N-186483.html>.
- [Gut+14] Enrico Gutzeit, Christian Scheel, Tim Dolereit und Matthias Rust. “Contour based split and merge segmentation and pre-classification of zooplankton in very large images”. In: *Proceedings of the International Conference on Computer Vision Theory and Applications*. 2014, S. 417–424. DOI: [10.5220/0004648604170424](https://doi.org/10.5220/0004648604170424).
- [Gut+15] Enrico Gutzeit, Martin Radolko, Arjan Kuijper und Uwe von Lukas. “Optimization-based Automatic Segmentation of Organic Objects of Similar Types”. In: *Proceedings of the International Conference on Computer Vision Theory and Applications*. to appear. 2015.
- [Gut08] Enrico Gutzeit. “Effektive Visualisierung der Next Subvolume Methode”. Diplomarbeit. Universität Rostock, 2008.
- [GV12] Enrico Gutzeit und Jörg Voskamp. “Automatic segmentation of wood logs by combining detection and segmentation”. In: *Proceedings of the International Symposium on Visual Computing*. Bd. LNCS 7431. 2012, S. 252–261. DOI: [10.1007/978-3-642-33179-4_25](https://doi.org/10.1007/978-3-642-33179-4_25).
- [GVW10] E. Gutzeit, Matthias Vahl und Egmont Woitzel. “Erkennung von Handgesten und Kopforientierung in einem generellen 3D-Tracking-Framework”. In: *Proceedings of the Conference Go-3D: Go for Innovations*. 2010, S. 85–95. URL: <http://publica.fraunhofer.de/documents/N-141671.html>.
- [HGM14] Anna Hänler, Enrico Gutzeit und Alfred Mertins. “Evaluation of methods for automatic fish segmentation”. In: *Proceedings of the Student Conference Medical Engineering Science*. 2014, S. 175–179. URL: <http://www.grin.com/en/e-book/268650/student-conference-medical-engineering-science-2014>.
- [Rei+12] Stefan Reinke, Enrico Gutzeit, Benjamin Mesing und Matthias Vahl. “Tracking technical objects in outdoor environment based on CAD models”. In: *Proceedings of the International Symposium on Visual Computing*. Bd. LNCS 7431. 2012, S. 437–446. DOI: [10.1007/978-3-642-33179-4_42](https://doi.org/10.1007/978-3-642-33179-4_42).
- [RG15] Martin Radolko und Enrico Gutzeit. “Video Segmentation via a Gaussian Switch Background-Model and Higher Order Markov Random Fields”. In: *Proceedings of the International Conference on Computer Vision Theory and Applications*. to appear. 2015.

- [Rus+13] Matthias Rust, Gernot Winkler, Christian Götze, Jörg Voskamp, Enrico Gutzeit und Martin Feike. “ZooCount - Internet based counting and classification in very large images”. In: *Poster Compendium of the Focus on Microscopy Conference*. 2013. URL: http://www.focusonmicroscopy.org/2013/PDF/526_Rust.pdf.
- [Sch+14] Michael Scheller, Jörg Voskamp, Enrico Gutzeit und Steffen Mader. *Verfahren zur automatisierten Mengenermittlung von beliebig großen Holzpoltern mittels eines Fotohandys*. DE Patent 102,012,017,878. 2014. URL: <http://www.google.com/patents/DE102012017878B3?cl=de>.
- [Ung+09] Andrea Unger, Enrico Gutzeit, Matthias Jeschke und Heidrun Schumann. “VioNeS - Visual support for the analysis of the next sub-volume method”. In: *Proceedings of the International Conference on Information Visualisation*. 2009, S. 10–17. DOI: [10.1109/IV.2009.18](https://doi.org/10.1109/IV.2009.18).

Danksagung

Ich möchte mich zum Schluss bei allen Menschen bedanken, die das Entstehen dieser Arbeit erst ermöglicht und zur Fertigstellung beigetragen haben.

Es ist viel Zeit in meiner Freizeit notwendig gewesen, um die Arbeit zu verfassen und einige notwendige Forschungen durchzuführen. Ohne die gegebene Zeit und Rücksichtnahme von meiner Familie wäre die Arbeit nie entstanden. Daher möchte ich mich in erster Linie bei meiner langjährigen Freundin Yvonne für die Entlastung im Alltag bedanken. Meiner Tochter Melina danke ich für die permanente klein-kindliche Lebensfreude, die meine Motivation immer wieder aufrechterhalten hat. Mein weiterer Dank gilt meinen Eltern und meinem Bruder für die moralische Unterstützung.

Meinem Doktorvater Prof. Dr.-Ing. Bodo Urban danke ich für die motivierende Begleitung, konstruktiven Gesprächen und Anmerkungen. Weiterhin danke ich Dr. Jörg Voskamp für die Zusammenarbeit und fachlichen Gespräche in den entscheidenden Projekten. Die Arbeit wäre nicht möglich gewesen ohne die Erkenntnisse und Ergebnisse aus den Projekten. Außerdem danke ich PD Dr. habil. Ir. Arjan Kuijper für die wissenschaftliche Begleitung beim Publizieren. Auch den Projektpartnern möchte ich danken sowie den Studenten für die Implementierungen und aufwendige Erstellung der Grundwahrheit. Mein besonderer Dank gilt Martin für die Umsetzung der aktiven Konturen und Beliefpropagation.

Ich möchte mich auch für das Korrekturlesen bei Kathleen bedanken. Nicht zuletzt auch ein herzliches Dankeschön an alle Freunde und Kollegen, die mich unterstützt und an mich geglaubt haben.

Selbstständigkeitserklärung

Ich erkläre, dass ich die eingereichte Dissertation selbstständig und ohne fremde Hilfe verfasst, andere als die von mir angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Ich versichere weiterhin, dass ich bisher weder die vorliegende Dissertation noch Teile von ihr als Prüfungsarbeit oder zum Zweck der Promotion eingereicht bzw. verwendet habe.

Rostock, 3. Februar 2015

Enrico Gutzeit

Thesen

1. Vorwissen (A-Priori) umfasst Wissen über den Bildinhalt, über die zu segmentierenden Objekte und/oder über das Ziel der Segmentierung. Ohne Vorwissen sind die Ergebnisse eines Segmentierungsalgorithmus unzureichend. Ein Segmentierungsalgorithmus muss mit Vorwissen direkt gesteuert und/oder indirekt anhand von Vorwissen konzipiert werden.
2. Vorwissen leitet sich ganz oder teilweise aus der Segmentierungsaufgabe ab. Je präziser das Vorwissen und je spezieller die Segmentierungsaufgabe, um so genauer und spezieller wird ein entsprechend angepasster Segmentierungsalgorithmus. Abhängig vom Vorwissen erfolgt die Auswahl, Kombination und Adaption von Basismethoden zur Segmentierung. Die Methoden der Objektdetektion beinhalten Vorwissen in Form eines Objektmodells und sind ein Mittel zur Segmentierung.
3. Schnittholz besitzt eine charakteristische Farbgebung. Die Farbgebung ist Vorwissen und geeignet zur binären Segmentierung von Schnittholz. Eine Restriktion in der Bildaufnahme und spezielle Vorsegmentierung ermöglicht die Extraktion von Schnittholz- (Vordergrund) und Nicht-Schnittholzpixeln (Hintergrund). Mit den extrahierten Pixeln ist die Erstellung eines repräsentativen Vorder- und Hintergrundfarbmodells möglich.
4. Mit einem Farbmodell von Schnittholz und Nicht-Schnittholz ist eine genaue binäre Segmentierung von Schnittholz möglich. Eine Dichteabschätzung im Farbmodell mit einer Kugelumgebung ist geeignet zur Erstellung eines Wahrscheinlichkeitsbildes der Farbe. Wahrscheinlichkeitsbilder von Schnittholz und Nicht-Schnittholz ermöglichen das Setzen der Gewichte eines Graphen, der für die Segmentierung mit Graph-Cut geeignet ist.
5. Die Methoden der Objektdetektion sind zur Detektion von Stammschnittflächen nutzbar. Stammschnittflächen in einem Bild der Frontfläche eines Holzpolters überdecken sich gegenseitig wenig und bilden einen Cluster im Bildraum. Falsch-positiv detektierte Stammschnittflächen lassen sich durch Überdeckungsprüfung und Ausreißeranalyse beseitigen.
6. Eine richtig-positiv detektierte Stammschnittfläche durch eine fensterbasierte Objektdetektion ist indirektes Vorwissen zur Segmentierung. Die Repräsentation des detektierten Fensters als Kreis ist eine grobe Approximation der Stammschnittfläche. Der Kreis an sich ist bereits eine grobe Segmentierung einer Stammschnittfläche. Durch aktive Konturen ausgehend vom Kreis kann eine Stammschnittfläche noch genauer segmentiert werden.

7. Durch detektierte Fenster von Stammschnittflächen ist es möglich, Schnittholz und Nicht-Schnittholzpixel abzuschätzen. Eine **Trimap** bestehend aus den Labels “Unbekannt”, “Vordergrund” und “Hintergrund” ist eine geeignete Repräsentation der Ergebnisse der Abschätzung. Mit der **Trimap** lassen sich auf Basis geeigneter Farbmodelle Wahrscheinlichkeitsbilder errechnen und das Schnittholz mit Graph-Cut binär segmentieren.
8. Stammschnittflächen sind in einem binären Schnittholzbild separierbar. Zur Separierung der Stammschnittflächen eignet sich eine Kreis Anpassung im Binärbild, da die Wahrscheinlichkeit, dass große Segmente im Binärbild zu einer Stammschnittfläche gehören, hoch ist. Durch eine Kreis Anpassung an die großen Segmente und synchron an die von den detektierten Fenstern überdeckten Segmente wird eine geeignete Segmentierung einzelner Stammschnittflächen erreicht.
9. Die Methoden der Objektdetektion sind ebenso nutzbar zur Detektion von Objekten einer beliebigen **organo**-Gruppe im Bild. Ein Klassifikator zur Detektion ist mit entsprechenden Objektbildern und Nicht-Objektbildern einer **organo**-Gruppe erstellbar. Bei großer Varianz der Objekte muss von Fehlern in der Detektion ausgegangen werden. Falsch-positiv detektierte Fenster einer **organo**-Gruppe lassen sich durch Überdeckungsprüfung und Ausreißeranalyse basierend auf Ähnlichkeitsmaßen beseitigen.
10. Die Segmentierung der Stammschnittflächen durch Objektdetektion und Farbmodelle ist übertragbar auf andere **organo**-Gruppen. Wie bei den Stammschnittflächen ist auch die Farbe der Objekte einer beliebigen **organo**-Gruppe ähnlich. Ebenso ist auch eine optimierungsbasierte Segmentierung auf Basis eines Farbmodells geeignet zur binären Segmentierung einer **organo**-Gruppe. Die Objektseparierung ist auch bei vielen anderen **organo**-Gruppen durch die detektierten Objektfenster und Annahme einer Kreisform möglich. Allerdings sind spezielle Annahmen der Objektform notwendig zur Segmentierung spezieller nicht-quasi-runder **organo**-Gruppen.
11. Die Bestimmung des Volumens eines Holzpolters anhand von segmentierten Stammschnittflächen in Bildern ist möglich, wenn ein Maßstab zum Bild gegeben oder bestimmt werden kann. Die Abschätzung des Volumens eines Baumstammes durch nur eine Schnittfläche ist fehlerbehaftet, da ein Baumstamm kein perfekter Zylinder ist. Im Praxiseinsatz ist die Abschätzung dennoch eine geeignete Hilfestellung und Alternative zur manuellen Volumenabschätzung.