

UNIVERSITÄT MANNHEIM

FAKULTÄT FÜR MATHEMATIK UND
INFORMATIK

UNIVERSITÄT
MANNHEIM

Statistische Modellierung und Detektion perzeptueller Primitive in Bildern

Diplomarbeit

von

Ralf Hettinger

Institut für Technische Informatik

Lehrstuhl für Bildverarbeitung, Mustererkennung und Computergrafik

Mai 2004

Gutachter: Prof. Dr. Christoph Schnörr
Zweitgutachter: Prof. Dr. Wolfgang Effelsberg

Danksagung

Diese Arbeit ist meinen Eltern Wolfgang und Helga Hettinger gewidmet. Während der Zeit meines Studiums und dieser Diplomarbeit standen sie immer bedingungslos hinter mir. Ohne ihre Unterstützung wäre mir mein Studium und diese Diplomarbeit als sein Abschluss in dieser Form nicht möglich gewesen. Ich danke Euch beiden!

Besonders bedanken möchte ich mich bei Prof. Dr. Christoph Schnörr. Während des Entstehens dieser Arbeit in den letzten sechs Monaten gab er mir immer wieder neue Anhaltspunkte und stand mir mit kompetentem Rat zur Seite.

Für das Aufzeigen von Unklarheiten und wertvolle Verbesserungsvorschläge gilt mein ausdrücklicher Dank Carsten Bienek, Stefan Büttner, Kai Eckert, Ingo Neumann, Stefan Schmidt, Tobias Schumacher, Oliver Soffke und Harald Wuest.

Ich möchte hier auch den Personen am Lehrstuhl für Bildverarbeitung, Mustererkennung und Computergrafik der Universität Mannheim danken. Die Arbeit am Lehrstuhl mit ihnen war immer ein Vergnügen.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Aufgabenstellung	1
1.2	Vergleich zu anderen Ansätzen	2
1.2.1	Canny-Operator	2
1.2.2	Hough-Transformation	4
1.3	Überblick dieser Arbeit	5
2	Statistischer Ansatz	7
2.1	Gestalttheorie und Computer Vision	7
2.1.1	Idee der Gestalttheorie	7
2.1.2	Gestaltgesetze	8
2.1.3	Verwendung der Gestalttheorie in der Computer Vision	12
2.2	Maskierungsproblem	13
2.3	Vorstellung des Helmholtz-Prinzips	15
2.4	Das Helmholtz-Prinzip in der Anwendung	17
2.4.1	Modellierung des perzeptuellen Primitivs P	17
2.4.2	Bestimmung der Detektionsfunktion $k(u)$ über den Erwartungswert	19
2.4.3	Durchmusterung des Suchraums	23
2.5	Large Deviations und das Helmholtz-Prinzip	24
2.5.1	Idee und Statistischer Kontext von Large Deviations	25
2.5.2	Folgerungen der Binomialverteilung	27
2.5.3	Obere Schranke der Detektionsfunktion $k(u)$	28
2.5.4	Asymptote der Detektionsfunktion $k(u)$	30
3	Liniensegmente	35
3.1	Modellierung	35
3.1.1	Linienbegriff	35
3.1.2	Diskrete Geometrie	36
3.1.3	Linienpunkte eines Liniensegments	39
3.1.4	Richtung von Bildpunkten	41

3.1.5	Linienpunkte nach dem Ordnungsprinzip "Richtung"	43
3.2	Bestimmung der Detektionsfunktion $k(l)$	44
3.2.1	Richtung der Linienpunkte als gleichverteilte Zufallsvariablen	44
3.2.2	Verteilung gerichteter Linienpunkte	45
3.2.3	Auftrittswahrscheinlichkeit gerichteter Linienpunkte	46
3.2.4	Erwartungswert für das Primitiv "Liniensegment" und Schranke ϵ	46
3.2.5	Detektionsfunktion $k(l)$	47
3.3	Suchraum	47
3.3.1	Rand-zu-Rand Segment L_R	47
3.3.2	Detektion mit Scan-Line Prinzip	48
3.4	Demaskierung	50
3.4.1	Direkte Demaskierung	51
3.4.2	Wechselseitige Demaskierung	52
3.5	Komplexität der Laufzeit und Verbesserungsansätze	56
3.5.1	Komplexität der Laufzeit	56
3.5.2	Verbesserungsansätze	61
4	Testergebnisse	63
4.1	Visualisierung der Ergebnisse und Ausgangsbilder	63
4.2	Demonstration an einem synthetischen Bild	64
4.3	Wahl der Parameter	68
4.3.1	Präzisionsparameter p	68
4.3.2	Grenzwert ϵ des Erwartungswerts	70
4.4	Einfluss von Rauschen	72
4.5	Auswirkungen des Gradienten	74
4.6	Problem quasi-paralleler Texturen	76
4.6.1	Parallelität durch Perspektive	77
4.6.2	Parallele Linientexturen	77
4.6.3	Parallelität an Farbverläufen	77
4.7	Unproblematische Bilder	80
5	Fazit und Ausblick	83
A	Anhang	85
A.1	Operatoren	85
A.1.1	Rundungsoperator <i>round</i>	85
A.1.2	Vorzeichenoperator <i>sign</i>	85
A.2	Herleitung der Rekursionsformel für die Binomialsumme	85
	Literaturverzeichnis	87

Abbildungsverzeichnis

2.1	Figur-Grund Differenzierung: Kippfigur von Rubin	8
2.2	Gestaltgesetz der Nähe	9
2.3	Gestaltgesetz der Geschlossenheit	10
2.4	Gestaltgesetz der Innenseite	11
2.5	Maskierung durch Einbettung in Textur	14
2.6	Schema der Anwendung des Helmholtz-Prinzips	18
2.7	Detektionsfunktion $k(u)$ und Schranken für das Primitiv “Liniensegment”	23
2.8	Vergleich der Asymptote gegen $k(u)$ mit [DMM00]	34
3.1	Verschiedene Linienpunkte bei gleicher euklidischer Linie	36
3.2	Kartesisches und polares Koordinatensystem	37
3.3	Liniensegmente unter Cityblock- und Schachbrett-Metrik	38
3.4	Orientierung von Liniensegmenten	40
3.5	Abtastung durch Rand-zu-Rand Segmente mit quantisiertem Winkel	49
3.6	wechselseitige Maskierung	54
3.7	wechselseitige Maskierung bei parallelen Liniensegmenten	54
4.1	Farbzuordnung bei Testergebnissen	63
4.2	Testsequenz: Synthetisches Demonstrationsbild	65
4.3	Testsequenz: Beet mit Stiefmütterchen	69
4.4	Testsequenz: Grundriss Mannheimer Schloss	71
4.5	Testsequenz: Korbessel und Einfluss von Rauschen (1)	72
4.6	Testsequenz: Korbessel und Einfluss von Rauschen (2)	73
4.7	Filtermasken für Sobel-Operator und Binomialoperator	74
4.8	Verschiedene Gradienten: Grundriss Mannheimer Schloss	75
4.9	Verschiedene Gradienten: Korbessel	75
4.10	Testsequenz: Autos auf einem Parkplatz	76
4.11	Testsequenz: Gebäudeteil “A” der Technischen Informatik	78
4.12	Testsequenz: Rückseite Mannheimer Schloss, Westflügel	79
4.13	Testsequenz: Oberfläche des Grafikprogramms <i>The GIMP</i>	81

Verzeichnis der Algorithmen

I	Linienpunkte mit dem Bresenham Algorithmus	41
II	Detektion von Liniensegmenten mit Scan-Line Prinzip	51
III	Erweiterung des Scan-Line Prinzips: Direkte Demaskierung	53

1 Einleitung

1.1 Motivation und Aufgabenstellung

Die visuelle Wahrnehmung der Umwelt bestimmt einen wesentlichen Teil unseres Lebens. Sobald morgens die Augen geöffnet werden, beginnt mit der Reizung der Sinneszellen auf der Netzhaut durch Licht der Wahrnehmungsprozess: Die einzelnen Reizimpulse werden nicht bewusst registriert, es entstehen unmittelbar Eindrücke von einfachen Strukturen. Konturen, Ecken und Farben sind nur einige Beispiele solcher *perzeptueller Primitive*.

Diese Wahrnehmung ist in der Regel ein unbewusster Schritt, es entstehen größere Strukturen. Umrisse mit eingeschlossenen Bereichen konstanter Farbe bilden beispielsweise Flächen. Zwei parallele Konturen, die einen rechten Winkel einschließen, ergeben ein Rechteck, eine runde Kontur wird durch ihre Regelmäßigkeit zu einem Kreis oder einer Ellipse.

Dieser Prozess wiederholt sich: Es werden Fenster, Gesichter und vieles andere mehr gesehen, die Strukturen werden als Gestalten oder Objekte identifiziert.

Ein Schwerpunkt der Computer Vision ist es, den menschlichen Wahrnehmungsprozess nachzubilden. Es ist schwierig, das für den Menschen selbstverständliche Erkennen zusammengesetzter Primitive oder größerer Strukturen so zu abstrahieren, dass eine Methodik daraus abgeleitet werden kann. Daher kommen oft verschiedene Verfahren für die Detektion unterschiedlicher Primitive zum Einsatz.

Bei der Kombination mehrerer Methoden müssen für die verwendeten Verfahren oft Parameter gewählt werden, die eine unterschiedliche Bedeutung und Auswirkung haben. Für einen einzelnen Algorithmus mag bekannt sein, wie seine Parameter geschickt zu wählen sind. Welchen Einfluss aber hat die Kombination verschiedener Verfahren auf die Auswirkung der einzelnen Parameter?

Die wachsende Anzahl der Parameter für verschiedene Verfahren und die Frage nach Wechselwirkungen der Parameter ist ein Hauptproblem bei dieser Herangehensweise.

In [DMM00] wird von Desolneux et al. ein statistischer Ansatz vorgestellt, dessen Parameter eine so allgemeine Bedeutung haben, dass eine gleiche Interpretation für viele perzeptuellen Primitive gewährleistet ist: Das so genannte *Helmholtz-Prinzip*.

Es basiert auf den Erkenntnissen der Gestaltpsychologie, die seit Anfang des 20. Jahrhunderts das Entstehen eines größeren Ganzen aus einzelnen visuellen Objekten untersucht. Der Kernpunkt des Helmholtz-Prinzips ist die methodische Formulierung des Begriffs “Prägnanz” aus der Gestalttheorie.

Vereinfachend gesagt wird dem Helmholtz-Prinzip zufolge ein perzeptuelles Primitiv dann wahrgenommen, wenn sein zufälliges Zustandekommen sehr unwahrscheinlich und damit bedeutungsvoll ist.

In dieser Arbeit wird gezeigt, wie der von Desolneux et al. für Liniensegmente vorgestellte Ansatz für perzeptuelle Primitive zu abstrahieren ist und in drei Stufen gegliedert werden kann. Die von Cao in [Cao03] vorgestellte Verwendung des Helmholtz-Prinzips für das Primitiv “gut fortgesetzte Linie” wird dieser Abstraktion zugeordnet.

Die algorithmische Anwendung wird am Beispiel von Liniensegmenten demonstriert. Die Unterschiede und Probleme im Vergleich zur Arbeit von Desolneux et al. werden dabei explizit aufgezeigt.

Wie groß der Unterschied zwischen der Auftrittswahrscheinlichkeit eines wahrnehmbaren Primitivs und dem Erwartungswert eines zufälligen Primitivs sein muss, erklärt die statistische Betrachtung so genannter *Large Deviations*.

1.2 Vergleich zu anderen Ansätzen

In diesem Abschnitt werden die Zusammenhänge zwischen dem Helmholtz-Prinzip und zwei anderen populären Methoden zur Bildanalyse gezeigt. In Abschnitt 1.2.1 wird der Canny-Operator zur Kantendetektion als Beispiel parametrisierter Verfahren vorgestellt. In Abschnitt 1.2.2 wird die Hough-Transformation als Verfahren der Merkmalsextraktion zur Gruppierung von geometrischen Figuren betrachtet.

1.2.1 Canny-Operator

Mit [Can86] stellt Canny ein oft eingesetztes Verfahren zur Kantendetektion vor: Den Canny-Operator. Das Ziel des Canny-Operators ist es, eine optimale Kantendetektion bezüglich folgender Kriterien zu liefern:

1. **Gute Detektion:** Detektion von so vielen “echten” Kanten wie möglich.
2. **Gute Lokalisierung:** Detektierte Kanten liegen möglichst nahe an Kanten im Ausgangsbild.

3. **Minimale Beschreibung:** Eine "echte" Kante erzeugt nur eine einzige Detektion, der Einfluss von Rauschen ist minimal.

Der Canny-Operator ist durch Variationsrechnung über diese Kriterien hergeleitet. Daraus ergibt sich der im Folgenden schematisch dargestellte Algorithmus:

1. **Reduktion von Rauschen:** Mit einem Gaußfilter der Größe G wird das Ausgangsbild geglättet.
2. **Berechnung der Richtung:** Für jeden Bildpunkt (i, j) wird der Gradient $\mathbf{g}(i, j)$ auf den Helligkeitswerten der 2×2 -Nachbarschaft berechnet. Die gemäß der acht Nachbarpunkte quantisierte Orthogonale zum Gradienten bestimmt den potentiellen Nachfolger (i_f, j_f) einer Kante durch (i, j) .
3. **Kantenverfolgung:** Jeder Bildpunkt (i, j) mit $|\mathbf{g}(i, j)| > T_1$ ist der Beginn einer Kante, wenn er nicht bereits Teil einer Kante ist. Für die Nachfolgekante (i_f, j_f) gilt: Falls $|\mathbf{g}(i_f, j_f)| > T_2$ und (i_f, j_f) noch in keiner Kante enthalten ist, dann gehört (i_f, j_f) zur selben Kante wie (i, j) . Für den so genannten *Hystherese*-Parameter T_2 gilt: $T_1 \approx 2T_2$.

Bei geeigneter Wahl der Parameter ist der Canny-Operator unempfindlich gegen Rauschen (Stufe 1, G). Detektierte Kanten beginnen nur an Bildpunkten mit starkem lokalem Kontrast (Stufe 3, T_1) und werden nach Möglichkeit fortgesetzt (Stufe 3, T_2). T_1 sorgt dafür, dass detektierte Kanten die wesentlichen Kanten eines Bildes sind. T_2 hat zur Folge, dass eine detektierte Kante auch dann keine Lücken enthält, wenn der Helligkeitsunterschied nicht sehr groß ist.

Vergleich mit dem Helmholtz-Prinzip. Der Canny-Operator ist ein typisches Beispiel für Verfahren, deren Ergebnisse stark von der Wahl der Parameter abhängig sind. Die Parameter G , T_1 und T_2 haben nur eine Bedeutung für das Primitiv "Kante". Ihre Wahl ist abhängig von der Art des Ausgangsbildes, eine unüberwachte Anwendung auf beliebige Ausgangsbilder ist aus diesem Grund nicht zu empfehlen.

Die Parameter des Helmholtz-Prinzips haben dagegen für alle Primitive dieselbe Bedeutung und ergeben sich aus einer allgemeineren Zielvorstellung. Das ist der Grund dafür, dass das Helmholtz-Prinzip auf viele Primitive und Ausgangsbilder unüberwacht besser angewendet werden kann, als Verfahren, deren Parameter von den Bildinhalten und speziellen Primitiven abhängen.

1.2.2 Hough-Transformation

Einen dem Helmholtz-Prinzip ähnlichen Weg zur Detektion von Primitiven geht die Hough-Transformation ([Hou62]). Die Grundidee ist folgendes Abstimmungsverfahren: Jedes potentielle Bestandteil einer geometrischen Figur mit bekannter Parametrisierung gibt für die Parameter, die es am besten beschreiben, eine Stimme ab. Hat jeder Bestandteil seine Parameter “gewählt”, erfolgt die Auswertung: Geometrische Figuren werden dann detektiert, wenn ihre Parameter mehr Stimmen auf sich akkumulieren, als ein bestimmter Schwellwert h .

Vergleich mit dem Helmholtz-Prinzip. Die algorithmische Anwendung der Hough-Transformation ist dem Helmholtz-Prinzip ähnlich: Auch für das Helmholtz-Prinzip werden alle Bestandteile im Hinblick auf ihre Erklärung durch ein perzeptuelles Primitiv (für die Hough-Transformation geometrische Figur) untersucht. Während die Grundidee der Hough-Transformation “Voting” ist, beruht das Helmholtz-Prinzip auf “Polling”: Die Bestandteile eines perzeptuellen Primitivs “fragen” eine mit dem Helmholtz-Prinzip bestimmte Detektionsfunktion, ob sie zu einem perzeptuellen Primitiv gruppiert werden.

Die Detektionsfunktion wird nicht in Abhängigkeit der betrachteten Bestandteile bestimmt, sondern mit der in Abschnitt 2.4 gezeigten Herleitung vorab berechnet. Das Helmholtz-Prinzip –mit der Terminologie der Hough-Transformation gesprochen– bestimmt sozusagen schon vor der Untersuchung der Bestandteile, wie viele Stimmen eine parametrisierte Figur benötigt, um detektiert zu werden.

Ein wesentlicher Unterschied ist in der Bestimmung des Schwellwerts bzw. der Berechnung der Detektionsfunktion zu sehen.

Die in [KEB91] und [SYK96] vorgestellte Bestimmung von h für die Hough-Transformation über Large Deviations (siehe Abschnitt 2.5) ist der Herleitung zur Berechnung der Detektionsfunktion des Helmholtz-Prinzips in Abschnitt 2.4 ähnlich (siehe [DMM00]).

Ein anderer Unterschied bleibt bestehen: Während die Bestandteile mit der Hough-Transformation ihre Stimme a priori parametrisierten geometrischen Figuren zuordnen, geht das Helmholtz-Prinzip mit der Nullhypothese davon aus, dass kein perzeptuelles Primitiv in einem Bild existiert. In dieser Arbeit wird deswegen die Terminologie des perzeptuellen Primitivs gewählt und nicht von einer parametrisierten geometrischen Figur gesprochen.

1.3 Überblick dieser Arbeit

Kapitel 2 stellt zunächst die für den Ansatz ideegebende Gestalttheorie mit Abschnitt 2.1 vor. Anhand der Gestaltgesetze wird gezeigt, wie sich die Disziplinen Gestalttheorie und Computer Vision überschneiden. In Abschnitt 2.2 wird das “Maskierungsproblem” beschrieben, das bei der späteren Anwendung Einfluss hat.

Abschnitt 2.3 erläutert die Idee des Helmholtz-Prinzips, Abschnitt 2.4 zeigt auf allgemeiner Ebene seine Anwendung und stellt dafür ein dreistufiges Grundgerüst vor:

- Modellierung des perzeptuellen Primitivs
- Aufstellen der Detektionsfunktion
- Durchmusterung des Suchraums nach perzeptuellen Primitiven

Abschnitt 2.5 zeigt, dass die Berechnung der Detektionsfunktion im Hinblick auf *Large Deviations* aus der Statistik sinnvoll ist und plausibilisiert damit den Ansatz.

Kapitel 3 beschreibt nach Desolneux et al. [DMM00] die Anwendung des Helmholtz-Prinzips für Liniensegmente in Abschnitt 3.1 bis Abschnitt 3.3. Der Schwerpunkt liegt dabei als Ergänzung zur Arbeit von Desolneux et al. in der Beschreibung des Modellierungsschrittes und der algorithmischen Durchmusterung des Suchraums.

In Abschnitt 3.4 wird das Maskierungsproblem für Liniensegmente in zwei Formen identifiziert und beschrieben, wie es gelöst werden kann.

Kapitel 4 zeigt die entstandenen Testergebnisse. Abschnitt 4.1 erklärt zunächst, wie die Visualisierung der Ergebnisse erfolgt.

Abschnitt 4.2 demonstriert, dass sich die theoretisch erwarteten Ergebnisse einstellen. Abschnitt 4.3 verdeutlicht die Auswirkung der Parameter des Helmholtz-Prinzips und begründet ihre Wahl.

Abschnitt 4.4 gibt einen Eindruck davon, wie sich verrauschte Eingangsdaten auswirken. Abschnitt 4.5 zeigt, dass bessere Ergebnisse erzielt werden können, wenn die Annahme statistischer Unabhängigkeit des Ansatzes leicht verletzt wird.

Abschnitt 4.6 verdeutlicht, inwiefern parallele Texturen für die Detektion von Liniensegmenten ein Problem darstellen. Schließlich sind in Abschnitt 4.7 als Schlussfolgerung aus den Testergebnissen Kriterien dafür angegeben, wann eine Detektion von Liniensegmenten nach dem Helmholtz-Prinzip gut gelingt.

2 Statistischer Ansatz

2.1 Gestalttheorie und Computer Vision

2.1.1 Idee der Gestalttheorie

Die Gestalttheorie beschäftigt sich mit der Frage, wie die Wahrnehmung des Menschen funktioniert. Sie beschreibt unter anderem die Beziehungen zwischen den Reizmustern auf der Netzhaut und der Wahrnehmung, die der Mensch durch diese Reize erfährt. Grundidee ist, dass sich Reize gemäß verschiedener Ordnungsprinzipien zu Elementen zusammenfügen, die sich wiederum zu einem größeren Ganzen zusammensetzen, das wir Menschen als *Gestalt* erkennen können.

Die Gestalttheorie wird als wesentlicher Bestandteil der Gestaltpsychologie gesehen. Sie entwickelte sich in den 20er Jahren des 20. Jahrhunderts als Abgrenzung zur so genannten klassischen Assoziationspsychologie: Diese erklärt die Wahrnehmung des Menschen über die Wechselwirkung zwischen Sinnesreizen und den dabei erlebten elementaren Gefühlen.

Max Wertheimer widerlegte als einer der Begründer der Gestalttheorie die Thesen der Assoziationspsychologie. In seinem Grundsatzwerk [Wer23] werden einige Regeln vorgestellt, die den Zusammenhang zwischen bestimmten Reizen und der erzeugten Wahrnehmung beschreiben. Eine Übersicht dieser und weiterer *Gestaltgesetze* findet sich in [Met75].

Die Grundlagen aller Gestaltgesetze lassen sich wie folgt zusammenfassen:

1. Wahrnehmungsgegebenheiten unterliegen einer Tendenz zur Organisation in Gestalten: Kleinere visuelle Elemente gruppieren sich rekursiv zu einem größeren Ganzen (hierarchische Strukturierung, siehe Abschnitt 2.1.2).
2. Gestalten heben sich als “Figuren” von einem undifferenzierten “Grund” ab (Figur-Grund Differenzierung). Je einfacher (prägnanter) und je bedeutungsvoller eine Gestalt ist, desto größer ist ihre Chance, als Figur hervorzutreten (siehe Abbildung 2.1).

3. Die Bestandteile des Ganzen beeinflussen einander gegenseitig. Veränderungen eines Teils einer Einheit führen daher zu Veränderungen des Ganzen (Irradiation, siehe Abbildung 2.3).

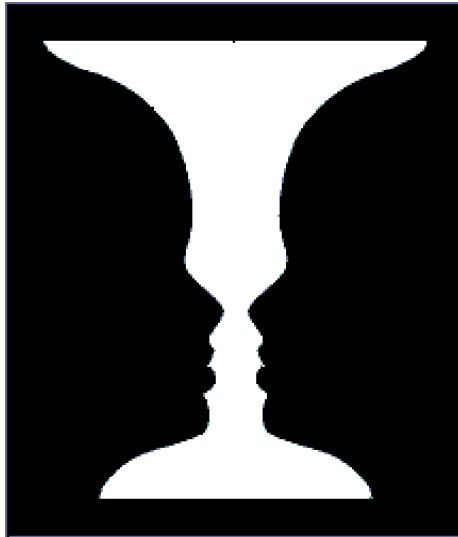


Abbildung 2.1: Kippfigur aus [Rub21]. Die Figur-Grund Differenzierung schwankt bei diesem berühmten Beispiel: Der Betrachter nimmt entweder eine Vase (Figur weiß, Hintergrund schwarz) oder zwei Gesichter im Profil (Figur schwarz, Hintergrund weiß) wahr, nie aber beides gleichzeitig.

Die programmatischen Ausführungen Wertheimers sind die konkretisierten Schlussfolgerungen aus Erkenntnissen, die bis zu Johann Wolfgang von Goethe zurückreichen. Ihre Bedeutung für die Wahrnehmung des Menschen findet sich unter anderen auch durch Hermann von Helmholtz (z.B. [vH98], [vH67]) näher untersucht.

2.1.2 Gestaltgesetze

Die Gestaltgesetze sind von der Gestaltpsychologie anerkannte Ordnungsprinzipien, die darüber entscheiden, was der Betrachter in einem Bild als Figur oder Gestalt wahrnimmt. Alles andere in diesem Bild wird für den Betrachter dann automatisch zu Hintergrund und damit nicht mehr bewusst wahrgenommen.

Um einen Eindruck der Vielfalt der Gestaltgesetze aus der Gestaltpsychologie zu vermitteln, werden in diesem Abschnitt einige Gestaltgesetze vorgestellt. Diese Beispiele vermitteln, dass sich die Gestalttheorie zu weiten Teilen mit ähnlichen Problemen wie die Computer Vision beschäftigt.

Allerdings wird auch der wesentliche Unterschied der beiden Disziplinen deutlich: Während die Gestalttheorie einen beschreibenden Ansatz der menschlichen Wahrnehmung verfolgt, zielt die Computer Vision darauf ab, die menschliche Wahrnehmung algorithmisch fassbar zu machen und –wenn möglich– nachzubilden.

Das Gesetz der Nähe. Das wohl elementarste Gesetz der Gestalttheorie ist das Gesetz der Nähe. Es besagt, dass nah beieinander gelegene Bildinhalte sich eher zu einer Figur gruppieren als weiter entfernte. Abbildung 2.2 zeigt zwei Beispiele für die Auswirkungen des Gesetzes der Nähe.

Das Gesetz der Nähe ist noch recht einfach auf die Computer Vision zu übertragen: Alle lokalen Mustererkennungsansätze zielen darauf ab, Untersuchungen auf einer definierten Nachbarschaft durchzuführen, um dann Objekte zu erkennen. Auch bei globalen Ansätzen werden Nachbarschaftsuntersuchungen durchgeführt, um diese dann im Kontext des gesamten Bildes zu bewerten. Eine Nachbarschaft aus der Computer Vision lässt sich mit dem Begriff der *Nähe* aus der Gestalttheorie identifizieren.

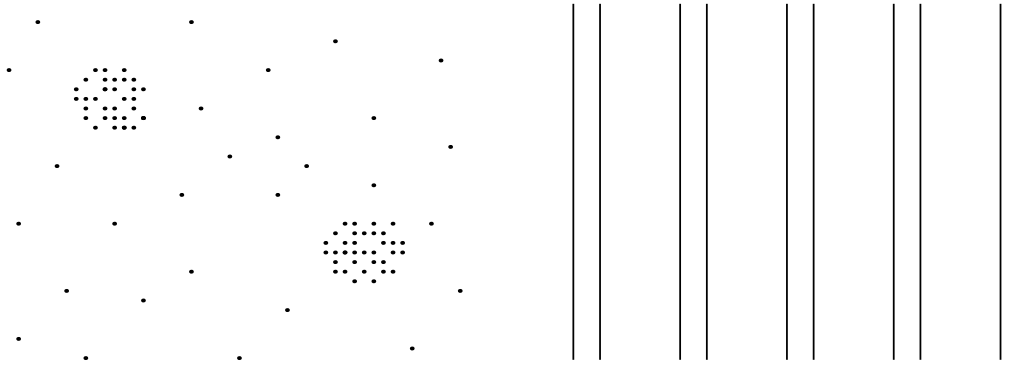


Abbildung 2.2: Das Gesetz der Nähe: Im linken Bild sind einzelne Punkte enthalten. Trotzdem werden die zwei Kernregionen bestehend aus nah beieinander liegenden Punkten wahrgenommen. Im rechten Bild sind zehn Liniensegmente enthalten. Allerdings werden fünf Streifen wahrgenommen, die aus jeweils zwei nah beieinander liegenden Liniensegmenten bestehen.

Das Gesetz der Geschlossenheit. Ein weiteres wichtiges Gesetz der Gestalttheorie ist das der Geschlossenheit: Figuren ergeben sich dann, wenn sie von einer geschlossenen Linie beschrieben werden.

Am Gesetz der Geschlossenheit werden zwei wesentliche Schwierigkeiten deutlich, die sich für die Anwendung der Gestalttheorie in der Computer Vision ergeben:

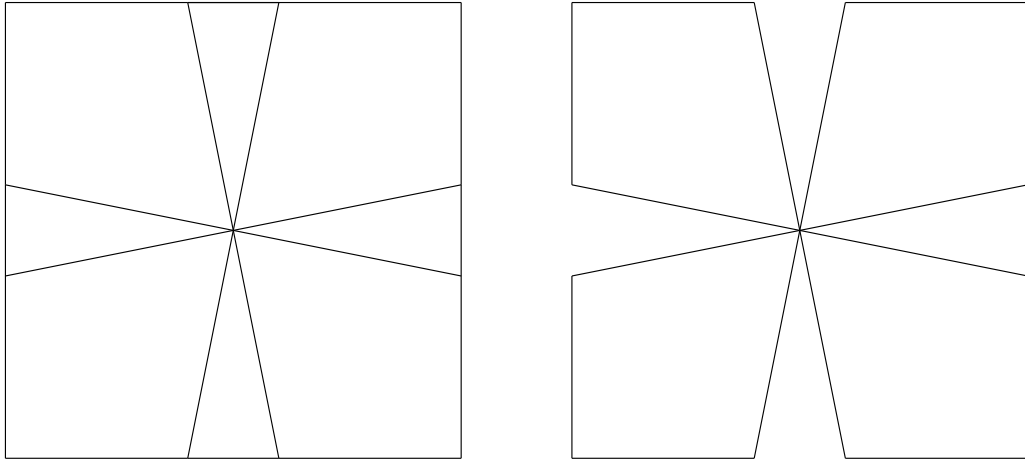


Abbildung 2.3: Das Gesetz der Geschlossenheit gegen das Gesetz der Nähe (aus [Met75]): Im linken Bild wird eher ein Kreuz wahrgenommen, da die begrenzenden Linien nahe beieinander liegen. Im rechten Bild sind die Außenkanten der Kreuz-Flächen entfernt, aufgrund des Gesetzes der Geschlossenheit werden nun vier annähernd rechteckige Flächen !!! wahrgenommen. Das Kreuz ist auf Antrieb nicht mehr zu sehen (Irradiation).

Während die Gestalttheorie anschaulich beschreibt, wie geschlossene Linien als Kriterium gelten können, ist es in der Computer Vision nicht klar, wie eine solche geschlossene Linie allgemeingültig überhaupt zu definieren ist: Beispielsweise haben geometrische Figuren geschlossene Linien als Grenze. Aber schon jeder fortgesetzte Helligkeitsunterschied kann ebenfalls eine geschlossene Linie bedeuten. Dagegen ist ein Helligkeitsunterschied innerhalb eines Graustufenverlaufs nicht als geschlossene Linie zu verstehen. Es sind offensichtlich viele weitere Beschreibungsmöglichkeiten für geschlossene Linien denkbar, die in einem anderen Kontext keine geschlossenen Linien mehr bedeuten würden. Die Allgemeingültigkeit des Begriffs *geschlossene Linie* für ein beliebiges Eingangsbild algorithmisch umzusetzen oder zumindest methodisch zu formulieren ist daher sehr schwierig.

Das zweite Problem besteht darin, dass nicht immer klar ist, welches Gestaltgesetz für die Wahrnehmung verantwortlich ist: Die in Abbildung 2.3 als Beispiel gezeigte Konkurrenz zwischen verschiedenen Gestaltgesetzen (hier dem Gesetz der Nähe und dem Gesetz der Geschlossenheit) ist nur schwer zu quantifizieren, da subjektive Eindrücke eine Rolle spielen. Sollte es dennoch gelingen, das Entgegen- oder Zusammenwirken verschiedener Gestaltgesetze zu erkennen, dann ist nicht klar, welches das maßgebliche ist: Der (subjektive) Kontext spielt eine zu große Rolle.

Das Gesetz der Innenseite. Das so genannte Gesetz der Innenseite (auch: Gesetz der Konvexität) wird in [Met75] wie folgt beschrieben: “Jede gekrümmte Linie, auch wenn sie nicht ganz in sich zurückkehrt, und ebenso jeder Winkel, [hat] eine Innenseite und eine Außenseite. Um den Raum auf ihrer Innenseite geht sie herum, auch wenn sie ihn nicht ganz einschließt, und dieser Raum wird besonders leicht zur Figur”.

Abbildung 2.4 veranschaulicht das Gesetz der Innenseite.

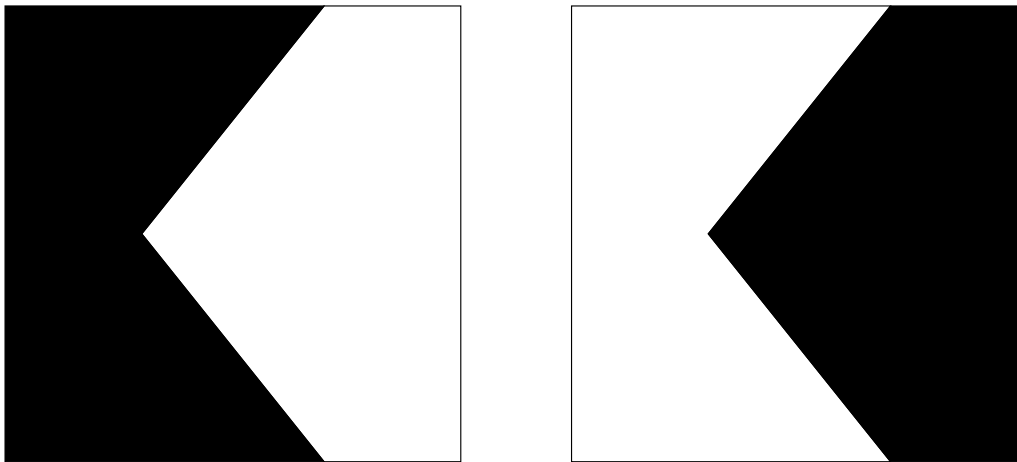


Abbildung 2.4: Das Gesetz der Innenseite (aus [Rub21]): Das linke und das rechte Bild zeigen beide den gleichen Umriss in jeweils unterschiedlicher Färbung. Nach dem Gesetz der Innenseite wird in beiden Bildern die rechte Hälfte (linkes Bild: weiß, rechtes Bild: schwarz) vorzugsweise als Figur erkannt. Die jeweils linke Hälfte der Bilder tritt in den Hintergrund. Die Fläche der rechten Hälfte hat einen spitzen Winkel (und ist damit konvex), die der linken Hälfte einen stumpfen Winkel.

Am Gesetz der Innenseite wird der Unterschied und die damit einhergehenden Schwierigkeiten zwischen der beschreibenden Gestalttheorie und der Computer Vision besonders deutlich: Die Beschreibung der Gestaltpsychologie ist oft so unspezifisch und allgemein gehalten, dass daraus keine konkrete Methodik abzuleiten ist. Am Beispiel des Gesetzes der Innenseite reicht schon eine *teilweise konvexe* Umrandung, damit dieses Gesetz zum Tragen kommt. Was aber unter “teilweise konvex” zu verstehen ist, hängt letztendlich vom Betrachter ab und ist mathematisch nicht zu formulieren.

2.1.3 Verwendung der Gestalttheorie in der Computer Vision

Der Ausgangspunkt der Mustererkennung und darauf aufbauender Bildanalysen (*Computer Vision*) ist wie die Retina des Menschen ein zweidimensionales Raster: Die Rohdaten sind beispielsweise als quantisierte Werte der Helligkeit an den Bildpunkten eines digitalen Bild gegeben.

Auch hier geht man davon aus, dass Informationen, die in Form von Helligkeitswerten an Bildpunkten vorliegen, anhand gewisser Ordnungsprinzipien gruppiert werden müssen, um dann Objekte algorithmisch erfassen zu können. Daher liegt es nahe, die Ordnungsprinzipien aus der Gestalttheorie auf algorithmische Umsetzbarkeit zu prüfen.

Wie in Abschnitt 2.1.2 gezeigt wurde, ist eine direkte Umsetzung der beschreibenden Gestaltgesetze aus der Gestaltpsychologie in der Regel nicht möglich. Zu sehr unterscheidet sich die Herangehensweise der Gestaltpsychologie von der der Computer Vision: Das wesentliche Problem ist, dass die Gestalttheorie eine beschreibende Methodik wählt, um die Zusammenhänge darzustellen.

Perzeptuelle Primitive. Beide Disziplinen basieren auf der Untersuchung visueller Objekte, die durch die Gruppierung so genannter *perzeptueller Primitive* entstehen. Einige solcher Primitive aus Sicht der Gestalttheorie wurden in Abschnitt 2.1.2 anhand der Gestaltgesetze vorgestellt. Wenn sie sich auch nicht unmittelbar auf die Computer Vision übertragen lassen, so können sie doch als Anhaltspunkte dafür dienen, was in einem Bild zu untersuchen ist. Tabelle 2.1 gibt dafür einige Beispiele.

Gestalttheorie	Computer Vision
Nähe	Nachbarschaftsuntersuchungen, Clustering
Gute Fortsetzung	Differenzierbarkeit entlang Kanten [Cao03], gerade Liniensegmente [DMM00]
Innenseite	Konvexität segmentierter Regionen
Ähnlichkeit	Textur, Registrierung von Objekten
gemeinsames Schicksal	Optischer Fluss

Tabelle 2.1: Beispiele perzeptueller Primitive aus Sicht der Gestaltpsychologie und der Umsetzung in der Computer Vision

Aufgrund der methodischen Natur der Computer Vision sind die in Abschnitt 2.1.1 dargestellten Grundlagen der Gestalttheorie als gemeinsamer Ausgangspunkt für eine Anwendung interessanter: Ausgehend von der Beobachtung, dass die Wahrnehmung auf hierarchischer Gruppierung basiert, findet sich in [Mar82] ein früher algorithmischer Ansatz. Informell lässt er sich als dreistufiger Bottom-up Algorithmus darstellen:

1. Lokale Detektion von Kanten anhand einzelner Grauwerte an Bildpunkten.
2. *Primal Sketch*: globale Gruppierung der Kanten anhand der Orientierung und Schnittpunkte.
3. Einordnung der gruppierten Kanten in einen dreidimensionalen Kontext.
 - *2.5D Sketch*: anhand Farbe und Verläufen.
 - *Frame Neutral Sketch*: anhand von bewegten und unbewegten Objekten.
 - *3D Sketch*: anhand der Kombination aller vorheriger Gruppierungen.

In der sich gegen Ende des 20. Jahrhunderts rasch entwickelnden anwendungsorientierten Forschungsrichtung der Computer Vision bildete die Untersuchung neuronaler Netze für die Erklärung der menschlichen Wahrnehmung einen Schwerpunkt. Dies mag eine Erklärung dafür sein, dass sich heute die Ansätze der Gestaltpsychologie nicht stärker in der Computer Vision niederschlagen.

Erst vor kurzem sind die grundlegenden Erkenntnisse der Gestalttheorie in der Computer Vision wieder aufgegriffen worden: Das in [DMM00] vorgestellte *Helmholtz-Prinzip* (siehe Abschnitt 2.4) beschreibt, wie sich die Grundsätze der Gestalttheorie methodisch formulieren lassen. Damit ist ein generelles Verfahren basierend auf der Gestalttheorie für die Computer Vision vorhanden. Mit [Cao03] findet sich ein weiteres Beispiel für die Anwendung des Helmholtz-Prinzips.

Eine ausführliche Darstellung der Gestalttheorie im Hinblick auf die Verwandtschaft zur Computer Vision ist in [Bra98] zu finden.

2.2 Maskierungsproblem

Wenn ein perzeptuelles Primitiv P_1 aufgrund der Präsenz eines anderen Primitivs P_2 nicht mehr wahrgenommen werden kann, so spricht man von *Maskierung*: P_2 maskiert P_1 .

Das Maskierungsproblem ist von der Gestaltpsychologie ausführlich untersucht worden. In [Kan79] findet man verschiedene Ursachen für Maskierung dargestellt. Unter anderen sind drei wesentliche Gründe, die die Gestaltpsychologie identifizieren konnte:

1. **Maskierung durch Einbettung in Textur**: Besteht der Hintergrund eines Primitivs P_1 aus einer Textur, die eine zu P_1 ähnliche Struktur aufweist, dann kann P_1 von der Textur P_2 maskiert werden (siehe Abbildung 2.5).

2. **Maskierung durch Ergänzung:** Fügt man einem Primitiv P_1 Bestandteile hinzu, deren Struktur ihm ähnlich ist, dann kann P_1 durch ein neu entstehendes Primitiv P_2 maskiert werden. Entfernt man Teile eines Primitivs P_1 , so kann dadurch ein neues Primitiv P_2 entstehen, das P_1 maskiert (Irradiation, siehe Abbildung 2.3).
3. **Maskierung durch Figur-Grund Differenzierung:** Befinden sich mehrere perzeptuelle Primitive P_1, \dots, P_n verschiedener Gestaltgesetze in einem Bild, dann kann P_1 in den Hintergrund treten und somit maskiert werden, wenn sich das Gestaltgesetz eines anderen Primitivs P_2 als stärker erweist (siehe Abbildung 2.3).



Abbildung 2.5: Maskierung durch Einbettung in Textur: Im linken Bild erkennt man eine Linie (Gesetz der Nähe, gute Fortsetzung) gebildet aus einzelnen Punkten. Im rechten Bild ist die umgebende Textur bestehend aus willkürlich verteilten Punkten stärker ausgeprägt. Die Linie ist nicht mehr zu sehen.

Wie in Abschnitt 3.4 am Primitiv “Liniensegment” gezeigt wird, sind Maskierungen ein wesentliches Problem bei der Detektion perzeptueller Primitive. Um es zu lösen (*Demaskierung*) ist es notwendig, die Ursachen für Maskierung in hohem Maß zu konkretisieren.

Am Beispiel der Liniensegmente kann man sehen, dass die oben vorgestellten Gründe für Maskierung nicht so klar zu trennen sind, wie dies in der Gestalttheorie oft den Anschein hat. Das liegt daran, dass die Gestalttheorie Zusammenhänge mit Beispielen beschreibt, die explizit dafür konstruiert wurden.

In der Computer Vision sollen dagegen –idealerweise unabhängig von den Eingangsbilddaten– Aussagen über beliebige Bilder getroffen werden.

Die “Maskierung durch Figur-Grund Differenzierung” spielt in dieser Arbeit keine Rolle, da mit Liniensegmenten nur *eine* Art von perzeptuellen Primitiven untersucht wird. Diese folgen dem gleichen Ordnungsprinzip.

Betrachtet man die Beispiele in den Abbildungen dieses Abschnitts, dann ist festzustellen, dass selbst der Betrachter nicht in der Lage ist, die gezeigten Maskierungen intuitiv aufzulösen. Es muss klargestellt werden, dass Demaskierung, zu der die menschliche Wahrnehmung nicht in der Lage ist, nicht Ziel dieser Arbeit sein kann: Es fehlen nachvollziehbare Kriterien, über die entschieden werden kann, ob eine Maskierung vorliegt oder nicht.

Die Abbildungen dieses Abschnitts dienen lediglich dazu, das Maskierungsproblem zu erklären. In Abschnitt 4.2 sind Maskierungen zu sehen, die ein menschlicher Betrachter demaskieren kann. Diese Demaskierung nachzuvollziehen ist Ziel für die Computer Vision und diese Arbeit.

2.3 Vorstellung des Helmholtz-Prinzips

In [DMM00] wird das so genannte *Helmholtz-Prinzip* vorgestellt. Die Idee basiert auf dem Begriff *Prägnanz* aus der Gestalttheorie: Prägnanz ist die Eigenschaft, welche einzelnen Bestandteilen Bedeutung zukommen lässt, so dass aus ihnen eine Struktur –bezogen auf die Computer Vision ein perzeptuelles Primitiv– entsteht. Die Gestalttheorie besagt weiterhin, dass die visuelle Wahrnehmung auf hierarchischer, rekursiver Gruppierung zu immer komplexeren Strukturen basiert, bis aus ihnen schließlich eine Gestalt entsteht. Das Helmholtz-Prinzip liefert das Kriterium dafür, was eine sinnvolle Gruppierung darstellt und was als Hintergrund zu verstehen ist.

Ein perzeptuelles Primitiv P ist aus kleineren Bestandteilen $P = \{B_i, i = 1, \dots, u\}$ zusammengesetzt, die einem gemeinsamen Ordnungsprinzip O folgen.

Beispielsweise besteht das Primitiv “Liniensegment” aus den Bestandteilen “Bildpunkte”, die es schneidet. Diese Bildpunkte unterliegen dem Ordnungsprinzip gleicher Richtung. Das Primitiv “Kante” besteht ebenfalls aus den Bestandteilen “Bildpunkte”. Diese folgen dem Ordnungsprinzip, zu den Helligkeitswerten der Bildpunkte ihrer Nachbarschaft einen Helligkeitsunterschied aufzuweisen.

Es gibt Primitive komplizierterer Natur, die ebenfalls dieser Methodik genügen: Beispielsweise besteht das Primitiv Rechteck aus vier Bestandteilen “Kanten”, die dem Ordnungsprinzip “Parallelität” gehorchen. An diesem Beispiel erkennt man leicht, dass solche Primitive oft aus Bestandteilen verschiedener Art bestehen können, die sich gegenseitig unterstützen, oder aber allein betrachtet ohne Aussagekraft sein können: Ein Rechteck besteht auch aus vier Bestandteilen “Ecken”, die an den Enden von jeweils zwei der vier Kanten lokalisiert sind und so die Kanten verbinden.

Egal, welches Primitiv P man betrachtet; man kann nicht davon ausgehen, dass sämtliche $B_i \in P$ gemäß O geordnet sind. Einige Ursachen dafür sind:

- **Rauschen.** Beispiel Liniensegment: Die Richtung einiger Bildpunkte, die von einem Liniensegment geschnitten werden, stimmt nicht mehr mit seiner Richtung überein.
- **Quantisierungsfehler, geringer Kontrast.** Beispiel Kante: An sehr hellen und sehr dunklen Bereichen in einem Bild sind die Unterschiede zwischen den Helligkeitswerten der Bildpunkte auf einer Kante und denen benachbarter Bildpunkte gering.
- **Verdeckung.** Beispiel Rechteck: Ist eine Ecke verdeckt, sind nur noch drei Ecken zu sehen, die von parallelen Kanten verbunden werden.

Daher stellt sich die Frage, wieviele Bestandteile $B_j \subseteq P, j = 1, \dots, g$ dem Ordnungsprinzip O genügen müssen, damit das Primitiv P wahrgenommen wird.

Die Antwort liefert das Helmholtz-Prinzip. Seine Idee ist folgende: Unter der Annahme, dass alle Bestandteile $B_i \in P, i = 1, \dots, u$ zufällig gleichverteilt sind, wird das Primitiv P dann wahrgenommen, wenn die Wahrscheinlichkeit dafür *gering* ist, dass g der u Bestandteile $B_j, j = 1, \dots, g$ zufällig dem Ordnungsprinzip O genügen und damit die Annahme zufälliger Gleichverteilung widerlegen. Die Art der Schlussfolgerung aus einer verworfenen Annahme ist in der Statistik als *Nullhypothese* bekannt.

Lowe stellt in [Low85] ein Verfahren vor, dessen Idee der des Helmholtz-Prinzips gleicht: Zunächst werden Merkmale zu Clustern gruppiert, die so nah beieinander liegen, dass ihr zufälliges Auftreten unwahrscheinlich ist. Diese Cluster werden anschließend nach dem gleichen Prinzip zu größeren Objekten gruppiert. Im Gegensatz zum Helmholtz-Prinzip liegt der Schwerpunkt Lowes Arbeit in der rekursiven Gruppierung verschiedener Primitive.

Nullhypothese. Eine Nullhypothese wird dann aufgestellt, wenn die eigentliche Annahme nur unspezifisch formuliert werden kann. Sie besagt dann das Gegenteil der zu prüfenden Annahme. Das ist ebenso in der Computer Vision der Fall: "Ausreichende" Regelmäßigkeit von Strukturen ist nur subjektiv zu quantifizieren. Wenn perzeptuelle Primitive aufgrund eines regelmäßigen Ordnungsprinzips erkannt werden sollen, so lautet die Nullhypothese: Es gibt keine Strukturen, d.h. der Inhalt des Bildes ist unregelmäßig.

Namensgebung nach Helmholtz. Als früher Verfechter von naturwissenschaftlichen Erklärungen für die menschliche Wahrnehmung in Abgrenzung zu eher metaphysischen Ansätzen untersuchte Helmholtz das Phänomen Wahrnehmung. Bezüglich des verfolgten Ansatzes liefern seine Erkenntnisse in zweierlei Hinsicht wesentliche Grundlagen:

Zum einen erkannte Helmholtz verschiedene Wechselwirkungen der Sinneszellen der Retina. In [vH67] wird beschrieben, wie schon auf dieser Ebene eine Gruppierung als Reaktion auf die Reizung einzelner Sinneszellen stattfindet.

Zum anderen formulierte er in seiner Erkenntnistheorie (siehe [vH98]) den Einfluss des Zufalls und eben die Abweichung davon auf das Erkennen von Objekten:

“[Es ist] gerade das Gesetzmäßige, was sich am regelmäßigsten gleichartig wiederholt, während das zufällig Wechselnde verwischt wird. Dem [...] achtsamen Beobachter erwächst auf diese Weise ein Anschauungsbild des typischen Verhaltens der Objekte.”

2.4 Das Helmholtz-Prinzip in der Anwendung

Angenommen, bei der Betrachtung von u verschiedenen Bestandteilen B_1, \dots, B_u , die ein perzeptuelles Primitiv P formen könnten (z.B. Bildpunkte entlang einer Linie) folgen g Bestandteile B_1, \dots, B_g einem gemeinsamen Ordnungsprinzip O (z.B. gleiche Farbe, Helligkeit, Richtung). Dann lässt sich mit dem Helmholtz-Prinzip die *Detektionsfunktion* $k(u)$ bestimmen, nach der gilt:

Falls $g \geq k(u)$, dann bilden die Bestandteile $B_i, i = 1, \dots, u$ ein wahrnehmbares perzeptuelles Primitiv $P = \{B_i, i = 1, \dots, u\}$ und P wird detektiert.

Abbildung 2.6 zeigt schematisch, wie das Helmholtz-Prinzip zur Detektion von perzeptuellen Primitiven umgesetzt wird. In den folgenden Abschnitten werden die 3 Schritte des Helmholtz-Prinzips *Modellierung* (Abschnitt 2.4.1), *Detektion* (Abschnitt 2.4.2) und *Durchmusterung des Suchraums* (Abschnitt 2.4.3) erläutert.

2.4.1 Modellierung des perzeptuellen Primitivs P

Für jedes zu untersuchende Primitiv ist es notwendig, die Art seiner Bestandteile B zu bestimmen und bereitzustellen. Außerdem muss das Ordnungsprinzip O definiert werden, nach dem die Bestandteile B für P angeordnet werden.

Da das Helmholtz-Prinzip zufällige Gleichverteilung der Bestandteile B annimmt, sollten diese so bestimmt werden, dass sie möglichst statistisch unabhängig voneinander sind. Ansonsten stehen weniger Bestandteile für die Detektion zur Verfügung,

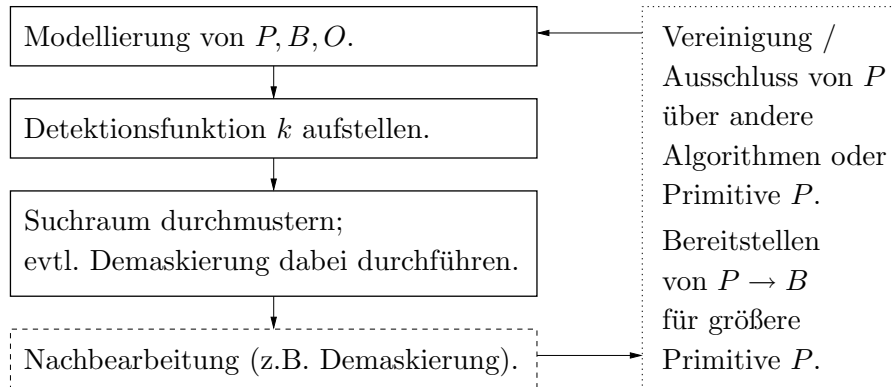


Abbildung 2.6: Anwendung des Helmholtz-Prinzips: Die geschlossenen Blöcke stellen die Umsetzung des Helmholtz-Prinzips dar. Der gestrichelte Block steht für Nachbearbeitungsschritte (eine Hauptaufgabe ist die Lösung des in Abschnitt 2.2 beschriebenen Maskierungsproblems; Abschnitt 4.2 zeigt, dass für akzeptable Ergebnisse das Maskierungsproblem berücksichtigt werden muss). Der gepunktete Block steht für weitere Aufgaben, die für rekursive Gruppierung notwendig sind. Dieser Block ist in dieser Arbeit nicht weiter untersucht worden. Er soll darstellen, wie das Helmholtz Prinzip für eine Kombination mehrerer Primitive eingesetzt werden kann (siehe Kapitel 5).

da statistisch abhängige Bestandteile nicht berücksichtigt werden sollten: Sie verletzen die Grundannahme der Nullhypothese, dass die Bestandteile als gleichverteilte Zufallsexperimente gesehen werden können.

In Abschnitt 3.1 sieht man den Modellierungsschritt für das perzeptuelle Primitiv “Liniensegment”.

Dieser Modellierungsschritt kann nicht weiter verallgemeinert werden. Sogar für das gleiche Primitiv sind verschiedene Modellierungen denkbar: Für Liniensegmente können beispielsweise verschiedene Gradienten zum Einsatz kommen (siehe Abschnitt 4.5), ein Rechteck kann über seine Ecken oder paarweise parallele Linien modelliert werden.

Die Bestandteile perzeptueller Primitive können andere “kleinere” Primitive sein, die zuvor mit dem Helmholtz-Prinzip bestimmt wurden. Damit könnte man die rekursive hierarchische Gruppierung des menschlichen Wahrnehmungsprozesses nachbilden, die in dieser Arbeit allerdings nicht weiter untersucht wird.

2.4.2 Bestimmung der Detektionsfunktion $k(u)$ über den Erwartungswert

Dieser Abschnitt beruht auf dem in [DMM00] von Desolneux et al. vorgestellten Helmholtz-Prinzip für Liniensegmente. Die dort gezeigte Herleitung der Detektionsfunktion $k(u)$ wird in dieser Arbeit abstrahiert, so dass sie für beliebige perzeptuelle Primitive angewendet werden kann. In Kapitel 3 wird die Anwendung des Helmholtz-Prinzips am perzeptuellen Primitiv “Liniensegment” demonstriert.

In diesem Zusammenhang ist die Arbeit von Cao [Cao03] zu erwähnen: Basierend auf dem Helmholtz-Prinzip werden dort Linien guter Fortsetzung und Ecken detektiert. Am Ende dieses Abschnitts wird gezeigt, dass das dort verwendete Detektionskriterium der vorgestellten Abstraktion entspricht. In Abschnitt 2.4.3 wird Caos Arbeit kurz beschrieben.

Die Grundannahme des Helmholtz-Prinzips ist, dass alle Bestandteile B_a jedes in einem Bild möglichen Primitivs P über den von O abhängigen Ergebnisraum A gleichverteilte Zufallsexperimente sind (Nullhypothese). Das setzt voraus, dass die Bestandteile B_a statistisch unabhängig voneinander sind.

Falls A kontinuierlich ist (das ist normalerweise der Fall), ist eine für die Art des Primitivs geeignete Quantisierung von A vorzunehmen. Für Liniensegmente in Abschnitt 3.1.5 wird n in Abschnitt 3.1.5 beschrieben, in Abschnitt 4.3.1 die Wirkung für $p = \frac{1}{n}$ untersucht und $n = 16$ gewählt.

Eine Zuordnung der Bestandteile $B_i, i = 1, \dots, u$ zu A entscheidet für jedes $B_i \in P$, ob es gemäß O für das betreffende Primitiv P angeordnet ist.

Die Wahrscheinlichkeit dafür, dass ein Bestandteil B gemäß O angeordnet ist, beträgt

$$p = \frac{1}{n}$$

Es ist zu bemerken, dass von

$$n \geq 2 \quad \Leftrightarrow \quad p \leq \frac{1}{2} \quad (2.1)$$

ausgegangen werden kann. Damit sind zufällig angenommene Bestandteile, die dem Ordnungsprinzip O genügen, höchstens so wahrscheinlich wie solche, die O nicht genügen. Diese Annahme ist nach der Idee des Helmholtz-Prinzips trivial: Primitive können nur dann wahrgenommen werden, wenn eine zufällige Beobachtung unwahrscheinlich ist.

Das Primitiv P bestehe aus den (statistisch unabhängigen) Bestandteilen $P = \{B_i, i = 1, \dots, u\}$. Die Wahrscheinlichkeit dafür, dass von den u Bestandteilen genau g gemäß O angeordnet sind, ist über die Binomialverteilung gegeben:

$$P\left[|B_j| = g\right] = \binom{u}{g} p^g (1-p)^{u-g}$$

Die Wahrscheinlichkeit, dass mindestens k der u Bestandteile gemäß O angeordnet sind, ergibt sich als rechtsseitiger Rand der Binomialverteilung:

$$P\left[g \geq k\right] = \sum_{i=k}^u \binom{u}{i} p^i (1-p)^{u-i} \quad (2.2)$$

$m(u)$ sei die Anzahl der möglichen perzeptuellen Primitive in einem Bild, die aus u Bestandteilen bestehen. Dann berechnet sich der Erwartungswert E des perzeptuellen Primitivs mit mindestens k_u Bestandteilen (bzw. Zufallsexperimente) $B_j, j = 1, \dots, k_u$ gemäß O zu

$$E = \sum_{u=u_{\min}}^{u_{\max}} m(u) P\left[g \geq k_u\right] \quad (2.3)$$

u_{\min} ergibt sich durch die Komposition des perzeptuellen Primitivs aus seinen Bestandteilen: Während das perzeptuelle Primitiv ‘‘Liniensegment’’ beispielsweise aus mindestens zwei Bildpunkten besteht, wird das Primitiv ‘‘Rechteck’’ aus mindestens vier Linien gebildet.

u_{\max} ist die Anzahl der Bestandteile, aus denen das größtmögliche perzeptuelle Primitiv in einem Bild besteht und damit direkt oder indirekt von der Bilddimension abhängig.

Die Anzahl aller möglichen Primitive in einem Bild ergibt sich zu

$$N_P = \sum_{u=u_{\min}}^{u_{\max}} m(u) \quad (2.4)$$

Die Herleitung der Detektionsfunktion über den Erwartungswert E jedes möglichen perzeptuellen Primitivs in einem Bild hat zur Folge, dass ein detektiertes Primitiv nicht nur signifikant verglichen zu den Primitiven mit gleicher Anzahl u an Bestandteilen ist, sondern zu *allen* Primitiven dieser Art in einem Bild.

Die Binomialverteilung fällt für $g > E$ annähernd exponentiell gegen 0 ([Fel68]). Für $g > u$ ist sie identisch 0. Das Auftreten eines perzeptuellen Primitivs mit einer Wahrscheinlichkeit, die am Rand der Binomialverteilung liegt, ist weit entfernt vom Erwartungswert (siehe Abschnitt 2.5).

Grenzwert ϵ des Erwartungswertes als Herleitung für $k(u)$. Um die Detektionsfunktion $k(u)$ zu bestimmen, wird ein Grenzwert ϵ für den Erwartungswert gefordert:

$$E < \epsilon$$

Mit dem Erwartungswert aus Gleichung (2.3) gilt:

$$\sum_{u=u_{\min}}^{u_{\max}} m(u) \mathbb{P} \left[g \geq k_u \right] < \epsilon$$

Die Frage, wie der Parameter ϵ zu wählen ist, kann nicht endgültig beantwortet werden. Interpretieren kann man ϵ als Anzahl der Primitive, die detektiert würden, falls die mit der Nullhypothese getroffene Annahme unabhängiger zufällig verteilter Bestandteile tatsächlich zutreffen würde. Insofern lässt sich $\frac{\epsilon}{N_P}$ als Falschalarmrate verstehen (siehe Abschnitt 2.5.2).

In Abschnitt 2.5.2 ist der erlaubte Wertebereich für ϵ hergeleitet. Gleichung (2.24) aus Abschnitt 2.5.4 zeigt, dass der Einfluss von ϵ auf $k(u)$ lediglich logarithmisch ist. In Abschnitt 4.3.2 wird auf den Einfluss von ϵ aus empirischer Sicht noch einmal eingegangen.

Die Anzahl an Möglichkeiten $m(u)$ für ein Primitiv mit u Bestandteilen ist abhängig von der Art und Definition des Primitiv. Die Bestimmung von $m(u)$ ist oft ein kombinatorisches geometrisches Problem und daher unter Umständen sehr aufwändig.

Eine Berechnung oder Abschätzung der Gesamtanzahl aller Möglichkeiten N_P für Primitive aus Gleichung (2.4) ist häufig viel leichter.

Daher ist eine Detektionsfunktion $k(u)$ gesucht, für die gilt

$$\begin{aligned} \mathbb{P} \left[g \geq k(u) \right] \underbrace{\sum_{u=u_{\min}}^{u_{\max}} m(u)}_{N_P} &< \epsilon \\ \mathbb{P} \left[g \geq k(u) \right] &< \frac{\epsilon}{N_P} \end{aligned} \quad (2.5)$$

Der Grund für die Herleitung von ϵ über den Erwartungswert und nicht über die zugrunde liegende Wahrscheinlichkeitsdichtefunktion ist, dass mögliche perzeptuelle Primitive nicht für sich allein betrachtet werden können. Dies ist eine Auswirkung des Maskierungsproblems (siehe Abschnitt 2.2):

Zum Beispiel (siehe Abschnitt 3.4) überdeckt ein sehr unwahrscheinliches Liniensegment andere Liniensegmente, deren Wahrscheinlichkeit klein genug ist, um Gleichung (2.5) zu genügen und damit detektiert zu werden.

Dies ist der Grund für den Unterschied zu der in der Statistik gebräuchlichen α -Signifikanz: Auch bei der α -Signifikanz wird eine Nullhypothese aufgestellt, die es zu verwerfen gilt, falls die Wahrscheinlichkeit für sie kleiner ist als α . Eine ausführliche Darstellung der α -Signifikanz ist z.B. in [HEK02] zu finden. Verglichen mit der α -Signifikanz entspricht $\alpha = \frac{\epsilon}{N_P}$. Es wird im Unterschied zur α -Signifikanz eine Relation zur Zahl aller Möglichkeiten N_P hergestellt, die sich nicht mit dem Grenzwert ϵ ändert, sondern allein von der Anzahl der möglichen Primitive abhängt.

Da $k(u)$ ein Mindestwert ist, ergibt sich mit $P(k, u) = \mathbb{P}\left[g \geq k(u)\right]$ formal die Detektionsfunktion $k(u)$

$$k(u) = \min \left\{ k(u) \in \mathbb{N}, P(k, u) \leq \frac{\epsilon}{N_P} \right\} \quad (2.6)$$

Die Summe $P(k, u)$ aus Gleichung (2.2) direkt zu berechnen ist ineffizient: Für jede Kombination aus (k, u) sind Summen zu bilden, deren Summanden mit der Wahrscheinlichkeit p gewichtete Binomialkoeffizienten sind.

Anhang A.2 zeigt eine rekursive Formulierung von $P(k, u)$:

$$P(k + 1, u + 1) = pP(k, u) + (1 - p)P(k + 1, u) \quad (2.7)$$

Mit dieser Rekursionsformel lässt sich eine Lookup-Tabelle erstellen, der man die Werte $k(u)$ für jedes u entnehmen kann. Abbildung 2.7 zeigt die Detektionsfunktion $k(u)$ anhand des in Abschnitt 3.1 beschriebenen perzeptuellen Primitivs “Liniensegment”.

Vereinfachte Detektion spezieller Primitive. Eine Vereinfachung ergibt sich für Primitive, bei denen *alle* Bestandteile $P = \{B_i, i = 1, \dots, u\}$ dem Ordnungsprinzip O genügen müssen, damit das perzeptuelle Primitiv P wahrnehmbar ist.

Ein Beispiel dafür ist das perzeptuelle Primitiv “Linie guter Fortsetzung”, wie es in [Cao03] beschrieben ist: Eine Linie gilt dann als gut fortgesetzt, falls die Richtung zwischen allen Bestandteile einen bestimmten Winkel κ nicht übersteigt (nach Cao $|\kappa| < \frac{\pi}{2}$, d.h. keine rechten Winkel auf einer Linie mit guter Fortsetzung).

Bei so modellierten Primitiven gilt $g = k = u$. Die Wahrscheinlichkeit für P unter Annahme, dass die Bestandteile Zufallsexperimente sind, ist dann nach Gleichung (2.2) mit der Wahrscheinlichkeit $p \in [0, 1]$ für ein Bestandteil $B \in P$

$$\begin{aligned} \mathbb{P}\left[g = k = u\right] &= \sum_{i=u}^u \binom{u}{i} p^i (1 - p)^{u-i} \\ &= p^u \end{aligned}$$

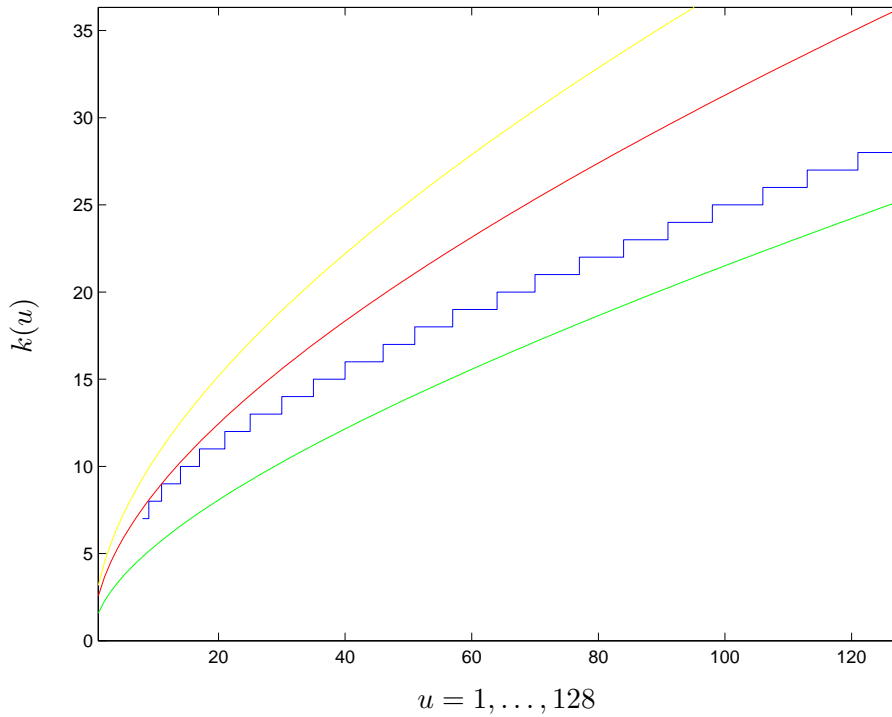


Abbildung 2.7: Detektionsfunktion $k(u)$ (nach Gleichung (2.7), blaue Treppenkurve) für das Primitiv “Liniensegment”. Abgetragen sind außerdem die oberen Schranken von $k(u)$ (nach Gleichung (2.17), rot bzw. nach Gleichung (2.18), gelb) und die Asymptote von unten (nach Gleichung (2.24), grün). $N_P = 128^2(128^2 - 1)$ ist die Anzahl aller möglichen Liniensegmente in einem Bild mit Dimension 128×128 . Nach der Modellierung der Liniensegmente in Abschnitt 3.1 besteht eine Linie aus höchstens $u_{\max} = 128$ Bestandteilen. $p = \frac{1}{16}$, $\epsilon = 1$.

In diesem Fall wird gemäß Gleichung (2.5) ein perzeptuelles Primitiv P bestehend aus u Bestandteilen detektiert, falls

$$p^u < \frac{\epsilon}{N_P}$$

2.4.3 Durchmusterung des Suchraums

Der dritte Schritt ist bei der Umsetzung des Helmholtz-Prinzips algorithmisch gesehen der aufwändigste: Die Idee des Helmholtz-Prinzips ist es, jede mögliche Konfiguration, die sich über das Ordnungsprinzip O für die Bestandteile eines perzeptuellen

Primitiv P in einem Bild ergibt, mittels der Detektionsfunktion $k(u)$ aus Abschnitt 2.4.2 zu prüfen.

Für die meisten Primitive ist die Anzahl aller möglichen Konfigurationen sehr groß: Für Liniensegmente in Abschnitt 3.3 gibt es beispielsweise mögliche Konfiguration der Größenordnung $O(N^4)$ für ein Bild mit Bilddimension $N \times N$.

Am Beispiel perzeptueller Primitive “Linien mit guter Fortsetzung”, wie sie von Cao in [Cao03] vorgestellt werden, ist schon eine Abschätzung der Größenordnung schwierig und in hohem Maß von der Modellierung des Primitivs abhängig.

Cao modelliert das Primitiv “gute Fortsetzung” auf *level lines* (Höhenlinien), die mit dem in [MG00] vorgestellten Algorithmus *fast level lines transform* erzeugt werden. Dadurch wird dort der Suchraum schon von vornherein eng eingegrenzt.

An diesem Beispiel ist zu sehen, wie das Helmholtz-Prinzip auf einem zuvor erzeugten perzeptuellen Primitiv (den Höhenlinien) aufsetzt: Die Höhenlinien bilden den eingeschränkten Suchraum für das perzeptuelle Primitiv “Linien mit guter Fortsetzung”.

Bemerkenswert an Caos Ansatz ist, dass die Höhenlinien nicht mit dem Helmholtz-Prinzip bestimmt sind. Das verdeutlicht, dass das Helmholtz-Prinzip auch auf anderen Ansätzen aufbauen kann. Außerdem macht dieses Beispiel deutlich, dass eine allgemeine Beschreibung, wie der Suchraum zu durchmustern ist, nicht sinnvoll ist: Der Charakter des Suchraums und seine Größe ist zu sehr von der Natur des betrachteten Primitivs abhängig. Verschiedene Suchräume für das gleiche Primitiv sind vorstellbar; allerdings ist dann zu erwarten, dass man unterschiedliche Resultate erhält.

2.5 Large Deviations und das Helmholtz-Prinzip

In diesem Abschnitt wird der Zusammenhang zwischen dem Helmholtz-Prinzip und *Large Deviations* (große Abweichungen) aus der Statistik aufgezeigt. In Abschnitt 2.5.1 wird die Idee von Large Deviations vorgestellt und eine Einordnung in den Kontext der Wahrscheinlichkeitsrechnung vollzogen. In den darauf folgenden Unterabschnitten wird die Detektionsfunktion $k(u)$ des Helmholtz-Prinzips aus Abschnitt 2.4.2 mit Ungleichungen aus der Statistik abgeschätzt:

In Abschnitt 2.5.2 sind die Folgerungen dargestellt, die sich aus der $k(u)$ zugrunde liegenden Binomialverteilung ergeben. Abschnitt 2.5.3 zeigt, dass $k(u)$ einer oberen Schranke nach Hoeffding ([Hoe63]) genügt. In Abschnitt 2.5.4 wird eine untere Asymptote mit Hilfe des Grenzwertsatzes von de Moivre und Laplace (siehe z.B. [Fel68]) für $k(u)$ gegeben. In Abbildung 2.7 sind die obere Schranke und die Asymptote gezeigt.

Diese Abschätzungen der Detektionsfunktion wurden in [DMM00] von Desolneux et al. vorgestellt; die Unterschiede zu dieser Arbeit werden explizit aufgezeigt. In der Arbeit von Desolneux et al. finden sich weitere Zusammenhänge zu Large Deviations, die aber eher theoretischer Natur sind. Diese werden nicht weiter betrachtet.

Die im Folgenden gezeigten Abschätzungen sind für die Anwendung des Helmholtz-Prinzips nicht notwendig, da sich die Detektionsfunktion $k(u)$ direkt berechnen lässt. Sie sollen vielmehr den statistischen Hintergrund der Detektionsfunktion darstellen und damit aufzeigen, dass nur solche Primitive mit ihr detektiert werden können, deren Auftrittswahrscheinlichkeit unter Annahme der Nullhypothese aus Abschnitt 2.3 auch aus der Perspektive statistischer Theorie gering ist.

2.5.1 Idee und Statistischer Kontext von Large Deviations

Wenn man in der Statistik von Large Deviations spricht, dann beschäftigt man sich mit Ereignissen, deren Auftrittswahrscheinlichkeiten große Abweichungen von den jeweiligen Erwartungswerten darstellen.

Der Begriff Large Deviations kann nicht als eigenständige Theorie gesehen werden (siehe [DS89]); vielmehr ist er eine Sammlung vieler statistischer Theorien und Schlussweisen, die für die Quantifizierung von Ereignissen mit großer Abweichung vom Erwartungswert nötig sind. Im Folgenden werden einige Bereiche der Statistik erwähnt, die eine wichtige Rolle spielen, wenn man Large Deviations untersucht:

- **(Sehr) seltene Ereignisse - (very) rare events:** Häufig als Synonym für Large Deviations benutzt.
- **Maßtheorie, Signifikanz - measure theory, significance:** Konzentriert sich auf die Schlussweise und Aussagekraft von Ereignissen großer Abweichung.
- **Starkes und schwaches Gesetz der großen Zahlen, Zentraler Grenzwertsatz - Strong and Weak Law of Large Numbers, Central Limit Theorem:** Konzentriert sich auf Mittelwert und Varianz einer Wahrscheinlichkeitsfunktion und kann daher als Gegensatz zu Large Deviations verstanden werden; jedoch bedingen sich beide Bereiche gegenseitig und bedienen sich der gleichen Methodik.
- **Konvergenz und Asymptotik:** Wichtige Werkzeuge bei der Untersuchung von Large Deviations, um Aussagen über die Quantifizierung zu erhalten.

Eine ausführliche Behandlung der Theorie über Large Deviations ist z.B. in [DZ98] und [dH00] zu finden. Eine praktisch orientierte Einführung mit einigen wesentlichen theoretischen Grundlagen findet sich mit [LR96]. Im Folgenden werden nur die für diese Arbeit interessanten Theoreme kurz vorgestellt.

Cramérs Theorem. Den mathematischen Grundstein von Large Deviations legte der schwedische Mathematiker Harald Cramér in [Cra99] mit dem Beweis folgender Aussage:

Sei X_1, \dots, X_n eine Sequenz endlicher, unabhängig gleichverteilter Zufallsvariablen mit Mittelwert m . M_n sei der empirische Mittelwert

$$M_n = \frac{1}{n}(X_1 + \dots + X_n)$$

Dann fallen mit $n \rightarrow \infty$ die Ränder der M_n zugrunde liegenden Wahrscheinlichkeitsfunktion exponentiell ab. Der exponentielle Abfall wird quantifiziert über die konvexe Ratenfunktion $I(x)$:

$$\exp(-nI(a)) \asymp \begin{cases} \text{P}[M_n > a] & \text{für } a > m \\ \text{P}[M_n < a] & \text{für } a < m \end{cases}$$

Mit dem Beweis der Existenz einer exponentiell fallenden, konvexen Ratenfunktion I gab Cramér den Anstoß, Wahrscheinlichkeiten am Rande der zugrunde liegenden Wahrscheinlichkeitsfunktion zu quantifizieren.

Cramér bestimmte I über die Entwicklung der Exponentialfunktion als Potenzreihe

$$\exp(x) = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

und Stirlings Formel (siehe z.B. [Fel68])

$$n! \stackrel{n \rightarrow \infty}{\sim} \sqrt{2n\pi} n^n \exp(-n)$$

Chernoff- und Hoeffding-Ungleichung. Chernoff zeigt in [Che52] eine Abschätzung für $\text{P}[S_n \geq na]$ bzw. $\text{P}[S_n \leq na]$ mit $S_n = nM_n$. Diese Vereinfachung für die Bestimmung der Ratenfunktion wird von Hoeffding in [Hoe63] formalisiert und weiterentwickelt.

Für Zufallsvariablen $X_i, i = 1, \dots, n$ mit Ereignisraum $]0, 1[$, $S = nM_n$ und Erwartungswert $E(S) = \mu$ lautet sie:

$$\begin{aligned} \text{P}\left[\frac{S}{n} - \mu \geq t\right] &\leq \left(\left(\frac{\mu}{\mu+t}\right)^{\mu+t} \left(\frac{1-\mu}{1-\mu-t}\right)^{1-\mu-t}\right)^n \\ &\leq \exp(-nt^2 h(\mu)) \\ &\leq \exp(-2nt^2) \end{aligned}$$

mit

$$h(\mu) = \begin{cases} \frac{1}{1-2\mu} \ln \frac{1-\mu}{\mu} & , 0 < \mu < \frac{1}{2} \\ \frac{1}{2\mu(1-\mu)} & , \frac{1}{2} \leq \mu < 1 \end{cases}$$

Diese Ungleichung wird in Abschnitt 2.5.3 auf die Detektionsfunktion bezogen.

2.5.2 Folgerungen der Binomialverteilung

In diesem Abschnitt wird gezeigt, in welchem Wertebereich mit $k(u)$ detektierte Primitive liegen.

Im Folgenden wird wie in Gleichung (2.6) die Summe über den Rand der Binomialverteilung für ein festes p mit $P(k, u)$ bezeichnet:

$$P(k, u) := \mathbb{P} \left[g \geq k(u) \right] = \sum_{i=k}^u \binom{u}{i} p^i (1-p)^{u-i}$$

Da $p \leq \frac{1}{2}$ (siehe Gleichung (2.1)) vor der Berechnung der Detektionsfunktion $k(u)$ fixiert wird, folgen daraus keine weiteren Nebenbedingungen.

Die folgende Eigenschaften der Summe über den Rand der Binomialverteilung $P(k, u)$ werden benötigt (siehe [Fel68], [Slu77]):

$$P(k = u, u) = p^u \tag{2.8}$$

$$P(k + 1, u) < P(k, u) \tag{2.9}$$

$$P(k, u) < P(k, u + 1) \tag{2.10}$$

$$P(k + 1, u + 1) < P(k, u)$$

Grenzwert ϵ des Erwartungswertes

Nach Gleichung (2.5) folgt direkt eine Eingrenzung des Wertebereichs für den Parameter ϵ :

$$0 < \epsilon < N_P \tag{2.11}$$

Für $\epsilon \leq 0$ könnte kein Primitiv detektiert werden, da für jedes mögliche Primitiv gilt $P(k, u) > 0$. Für $\epsilon \geq N_P$ würde jedes Primitiv detektiert werden, da $P(k, u) < 1$ für $p \leq \frac{1}{2}$, $N > 1$ gilt. Das plausibilisiert die Interpretation von $\frac{\epsilon}{N_P} \in [0, 1]$ als Falschalarmrate.

Nach dem Ordnungsprinzip geordnete Bestandteile g

Für ein detektiertes Primitiv P , welches nach dem Ordnungsprinzip O gebildet ist, gilt für die Anzahl g gemäß O geordneter Bestandteilen $B_j, j = 1, \dots, g$

$$\forall P|O : g \geq 1,$$

da sonst für $P(k = 0, u)$ nach Gleichung (2.11) zur Detektion des Primitivs mit $g = 0$ gelten müsste: $\epsilon \geq N_P$.

Mindestanzahl u der Bestandteile eines Primitivs

Für detektierte Primitive gilt nach Gleichung (2.8):

$$p^u = P(u, u) \leq P(k, u) < \frac{\epsilon}{N_P}$$

Daraus folgt für die Anzahl u an Bestandteilen, aus denen ein Primitiv mindestens besteht

$$u \geq \frac{\ln \epsilon - \ln N_P}{\ln p} \quad (2.12)$$

Es ist klarzustellen, dass diese Mindestanzahl u nichts mit u_{\min} aus Gleichung (2.3) zu tun hat: Während Gleichung (2.12) die Anzahl u an Bestandteilen angibt, aus denen ein Primitiv mindestens bestehen muss, um mit $k(u)$ detektiert werden zu können, ist u_{\min} die Anzahl an Bestandteilen, aus denen ein mögliches Primitiv mindestens besteht, um der Modellierung des Primitivs zu genügen. In Abschnitt 4.2 wird der Unterschied anhand des Primitivs "Liniensegment" veranschaulicht.

2.5.3 Obere Schranke der Detektionsfunktion $k(u)$

Hoeffding-Ungleichung

In [Hoe63] wird mit Theorem 1 die *Hoeffding-Ungleichung* für die Wahrscheinlichkeit P vorgestellt, dass die Summe S über u unabhängige Zufallsvariablen $X_i, i = 1, \dots, u$ im Ereignisraum $]0, 1[$ ihren Erwartungswert $E(S)$ übersteigt:

$$P \left[\frac{S}{u} - \frac{E(S)}{u} \geq t \right], \quad 0 < t < 1 - \frac{E(S)}{u}, \quad 0 \leq X_i \leq 1, i = 1, \dots, u$$

Ein detektiertes Primitiv P bestehe aus u Bestandteilen $P = \{B_i, i = 1, \dots, u\}$. Darunter seien g gemäß O geordnete Bestandteilen $B_j \subseteq P, j = 1, \dots, g$. Dann gilt

$$S = g, \quad X_i = \begin{cases} 0 & , B_i \text{ genügt nicht } O \\ 1 & , B_i \text{ genügt } O \end{cases}; i = 1, \dots, u$$

Nach der in Abschnitt 2.3 angenommenen Nullhypothese, dass alle $B_i \in P$ zufällig gleichverteilt sind, gilt:

$$\frac{E(S)}{u} = p$$

Nach Gleichung (2.5) gilt $0 < k < up$; ansonsten würden Primitive detektiert werden, die weniger O genügende Bestandteile besitzen, als man unter Annahme zufälliger Bestandteile erwarten könnte.

Um die Hoeffding-Ungleichung für $P(k, u)$ aus Gleichung (2.2) zu verwenden, gilt mit $r = \frac{k}{u}$:

$$t = r - p \quad \Leftrightarrow \quad P(k, u) = \mathbb{P} \left[\frac{S}{u} - \frac{E(S)}{u} \geq t \right] \quad (2.13)$$

Für $P(k, u)$ gilt dann nach Gleichung (2.13):

$$\begin{aligned} P(k, u) &\leq \underbrace{\left(\left(\frac{p}{r} \right)^r \left(\frac{1-p}{1-r} \right)^{1-r} \right)^u}_{\exp \left(ur \ln \left(\frac{p}{r} \right) + u(1-r) \ln \left(\frac{1-p}{1-r} \right) \right)} \\ &\leq \exp \left(-u(r-p)^2 h(p) \right) \end{aligned} \quad (2.14)$$

$$\leq \exp \left(-2u(r-p)^2 \right) \quad (2.15)$$

mit

$$h(p) = \begin{cases} \frac{1}{1-2p} \ln \frac{1-p}{p} & , 0 < p < \frac{1}{2} \\ \frac{1}{2p(1-p)} & , \frac{1}{2} \leq p < 1 \end{cases}, \quad \min \{h(p)\} = 2 \quad (2.16)$$

Gleichung (2.15) ist eine direkte Folgerung aus Gleichung (2.16).

Obere Schranke nach Hoeffding-Ungleichung

Mit der oberen Abschätzung von $P(k, u)$ aus Gleichung (2.14) ergibt sich

$$\begin{aligned}
 P(k, u) &\leq \exp\left(-u(r-p)^2 h(p)\right) \leq \frac{\epsilon}{N_P} \\
 u(r-p)^2 &\geq \frac{\ln \epsilon - \ln N_P}{h(p)} \\
 \left(\frac{k}{u} - p\right)^2 &\geq \frac{\ln \epsilon - \ln N_P}{uh(p)} \\
 k(u) &\geq pu + \sqrt{\frac{u}{h(p)}(\ln \epsilon - \ln N_P)} \quad (2.17)
 \end{aligned}$$

Eine noch gröbere Abschätzung von $k(u)$ nach oben ergibt sich mit Gleichung (2.15):

$$k(u) \geq pu + \sqrt{\frac{u}{2}(\ln \epsilon - \ln N_P)} \quad (2.18)$$

Wie man in Abbildung 2.7 sehen kann, genügt $k(u)$ der Abschätzung nach Hoeffding. Sie kann daher als obere Schranke für $k(u)$ gesehen werden.

2.5.4 Asymptote der Detektionsfunktion $k(u)$

Grenzwertsatz von de Moivre-Laplace

In [Fel68] wird der Grenzwertsatz von de Moivre-Laplace für die Summe S über den rechtsseitigen Rand der Binomialverteilung durch Annäherung mit der Poisson-Verteilung vorgestellt. Mit einem beliebigen, aber festem α gilt demzufolge für $u \rightarrow \infty$:

$$\begin{aligned}
 \mathbb{P}\left[S \geq pu + \alpha\sqrt{p(1-p)u}\right] &\xrightarrow{u \rightarrow \infty} 1 - \underbrace{\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha} \exp\left(-\frac{x^2}{2}\right) dx}_{\frac{1}{\sqrt{2\pi}} \int_{\alpha}^{+\infty} \exp\left(-\frac{x^2}{2}\right) dx}
 \end{aligned}$$

Wie in Abschnitt 2.5.3 ist u die Anzahl an Bestandteilen, aus denen ein Primitiv besteht. u ist durch $u \leq u_{\max}$ (siehe Gleichung (2.3)) beschränkt. u_{\max} ist abhängig

von der Modellierung des betrachteten Primitivs und dem daraus resultierenden Suchraum.

Aus diesem Grund ist $u \rightarrow \infty$ für Primitive in Bildern nicht haltbar. Deswegen wird der in [Fel68] gezeigte Spezialfall des Grenzwertsatzes betrachtet: Falls

$$\frac{\alpha(u)^3}{\sqrt{u}} \xrightarrow{\alpha(u) \rightarrow \infty} 0, \quad (2.19)$$

dann gilt

$$\mathbb{P} \left[S \geq pu + \alpha(u) \sqrt{p(1-p)u} \right] \sim \frac{1}{2\pi} \frac{1}{\alpha(u)} \exp \left(-\frac{\alpha(u)}{2} \right) \quad (2.20)$$

In [DMM00] wird gezeigt, dass für

$$\alpha_*(u) = \frac{k(u) - pu}{\sqrt{p(1-p)u}} \quad (2.21)$$

gilt:

$$\frac{\alpha_*(u)^6}{u} \xrightarrow{\frac{u}{N_P} \rightarrow \infty} 0$$

Damit gilt die Bedingung aus Gleichung (2.19) und die Formulierung des Grenzwertsatzes von de Moivre und Laplace aus Gleichung (2.20) ist anwendbar.

Asymptote nach dem Spezialfall aus Gleichung (2.20)

Nach Gleichung (2.5) gilt für detektierte Primitive mit Gleichung (2.20) und $\alpha(u) = \alpha_*(u)$ aus Gleichung (2.21):

$$\begin{aligned}
 \frac{1}{\sqrt{2\pi}} \frac{1}{\alpha(u)} \exp\left(-\frac{\alpha(u)^2}{2}\right) &\leq \frac{\epsilon}{N_P} \\
 \frac{\sqrt{p(1-p)u}}{\sqrt{2\pi}(k-pu)} \exp\left(-\frac{(k-pu)^2}{2p(1-p)u}\right) &\leq \frac{\epsilon}{N_P} \\
 \ln \sqrt{p(1-p)u} - \ln \sqrt{2\pi}(k-pu) - \frac{1}{2} \frac{(k-pu)^2}{p(1-p)u} &\leq \ln \epsilon - \ln N_P \\
 \ln N_P - \ln \epsilon + \ln \sqrt{p(1-p)u} - \ln \sqrt{2\pi}(k-pu) &\leq \frac{1}{2} \frac{(k-pu)^2}{p(1-p)u} \\
 pu + \sqrt{2p(1-p)u \left(\ln \frac{N_P}{\epsilon} + \ln \sqrt{p(1-p)u} - \ln \sqrt{2\pi}(k-pu) \right)} &\leq k \\
 pu + \sqrt{2p(1-p)u \ln \left(\frac{N_P}{\epsilon} \underbrace{\frac{\sqrt{p(1-p)u}}{\sqrt{2\pi}(k-pu)}}_{k^*} \right)} &\leq k \quad (2.22)
 \end{aligned}$$

Um k zu isolieren und dadurch eine Asymptote für $k(u)$ zu erhalten, ist eine Abschätzung der linken Seite von Gleichung (2.22) notwendig.

Das Gesetz des iterierten Logarithmus nach Khintchine (siehe [Fel68])

$$\limsup_{u \rightarrow \infty} \frac{S - pu}{\sqrt{p(1-p)u}} = \sqrt{2 \ln \ln u} \quad (2.23)$$

erlaubt eine Abschätzung von k^* der linken Seite von Gleichung (2.22).

Mit der Definition von $k(u)$ in Gleichung (2.6) folgt, dass $k \leq S$. Daraus folgt:

$$\ln \frac{\sqrt{2\pi}(k-pu)}{\sqrt{p(1-p)u}} \leq \ln \frac{\sqrt{2\pi}(S-pu)}{\sqrt{p(1-p)u}}$$

Für $u \rightarrow \infty$ wird $u = u_{\max}$ gesetzt, da u_{\max} aus Gleichung (2.3) das größtmögliche Primitiv (also das mit den meisten Bestandteilen) in einem Bild darstellt. Gleichung (2.23) kann damit wie folgt formuliert werden:

$$\frac{\sqrt{2\pi}(k-pu)}{\sqrt{p(1-p)u}} = O(\ln \ln u_{\max})$$

Damit lässt sich k^* aus Gleichung (2.22) abschätzen und k isolieren. Es folgt für $k(u)$

$$k(u) \geq pu + \sqrt{2p(1-p)u \ln \frac{N_P}{\epsilon O(\ln \ln u_{\max})}}$$

Diese Ungleichung ergibt sich mit Gleichung (2.20) als untere Asymptote gegen die Detektionsfunktion $k(u)$:

$$k(u) \asymp pu + \sqrt{2p(1-p)u \left(\ln \frac{N_P}{\epsilon} - \ln O(\ln \ln u_{\max}) \right)} \quad (2.24)$$

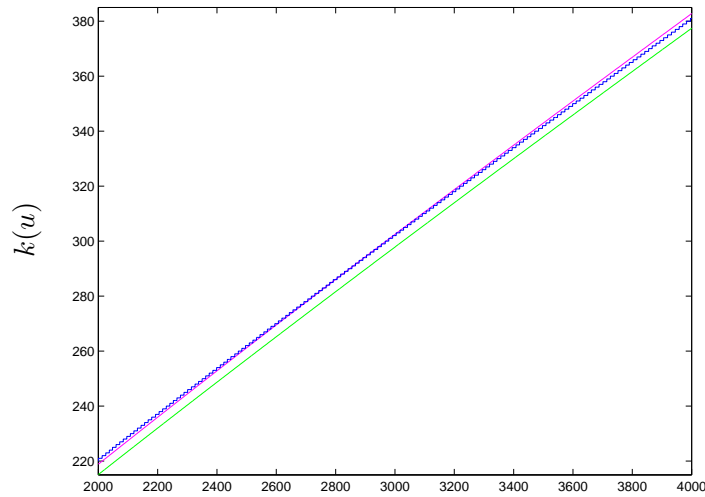
Vergleich der Asymptote mit [DMM00]. Die gezeigte Herleitung der Asymptote für $k(u)$ basiert auf dem gleichen Ansatz wie die von Desolneux et al. [DMM00] vorgestellte: Dem Grenzwertsatz von de Moivre und Laplace.

Der Unterschied besteht darin, dass die hier gezeigte Herleitung k^* aus Gleichung (2.22) nach dem Gesetz des iterierten Logarithmus abschätzt. Desolneux et al. isolieren k , indem $\ln \alpha(u) \approx O(\ln \ln N)$ abgeschätzt wird und erhalten damit

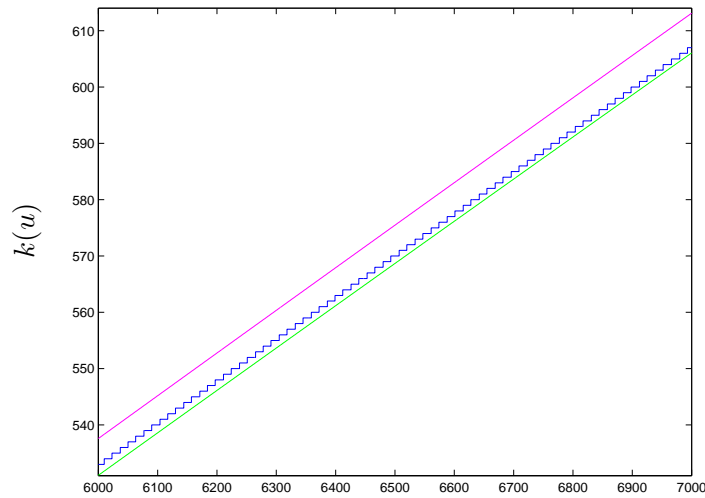
$$k(u) \asymp pu + \sqrt{2p(1-p)u \left(\ln \frac{N_P}{\epsilon} + O(\ln \ln u_{\max}) \right)} \quad (2.25)$$

Das Problem der von Desolneux et al. als Asymptote vorgestellten Abschätzung ist, dass sie für große N nicht standhält: Sie schneidet $k(u)$ und steigt danach stets stärker als $k(u)$. Abbildung 2.8 zeigt den Vergleich zwischen Gleichung (2.24) und Gleichung (2.25) für $N_P = 7000^2(7000 - 1)^2$. Aufgrund numerischer Ungenauigkeiten bei der Berechnung der Detektionsfunktion mit Gleichung (2.7) ist jedoch zu befürchten, dass auch die Asymptote aus Gleichung (2.24) die Detektionsfunktion für noch größere Werte von N_P schneidet.

Diese Erkenntnisse sind allerdings eher theoretischer Natur: Eine praktische Anwendung des Helmholtz-Prinzips auf Primitive, für die es eine so große Anzahl an Möglichkeiten in einem Bild gibt, ist in näherer Zukunft nicht zu erwarten.



(a) $u = 2000, \dots, 4000$



(b) $u = 6000, \dots, 7000$

Abbildung 2.8: Vergleich der Asymptote aus Gleichung (2.24) (grün) mit der in [DMM00] vorgestellten nach Gleichung (2.25) (magenta). Zur Orientierung ist die Detektionsfunktion $k(u)$ (nach Gleichung (2.7), blaue Treppenkurve) abgetragen. Die magentafarbene Kurve von Desolneux et al. schneidet $k(u)$ bei $u \approx 3000$ (2.8(a)) und steigt auch danach stärker als $k(u)$ (2.8(b)). Die hier gezeigte Asymptote in grün bleibt auch für $N = 7000$ stets kleiner als $k(u)$ und nähert sich $k(u)$ an. Die abgetragenen Kurven gelten für Liniensegmente mit $N_P = 7000^2(7000^2 - 1)$, $u_{\max} = 7000$ (vgl. Abbildung 2.7). $p = \frac{1}{16}$, $\epsilon = 1$.

3 Liniensegmente

In diesem Kapitel wird die Anwendung des Helmholtz-Prinzips aus Abschnitt 2.4 für das perzeptuelle Primitiv “Liniensegment” beschrieben. Ausgangspunkt ist die Arbeit von Desolneux et al. [DMM00]. Dieses Kapitel zeigt als Erweiterung zu dieser Arbeit ausführlich den Modellierungsschritt in Abschnitt 3.1 und beschreibt noch einmal kurz das Aufstellen der Detektionsfunktion gemäß Abschnitt 2.4.2 in Abschnitt 3.2. In Abschnitt 3.3 findet sich der Algorithmus, mit dem der Suchraum durchmustert wird.

Neben der Anwendung des Helmholtz-Prinzips wird in Abschnitt 3.4 auf das Maskierungsproblem nach Abschnitt 2.2 eingegangen und aufgezeigt, wie es gelöst wird.

3.1 Modellierung

Bestandteile “Linienpunkte”. Abschnitt 3.1.1 zeigt, dass ein Liniensegment aus den Bildpunkten besteht, die es schneidet (*Linienpunkte*). Abschnitt 3.1.2 geht auf die zugrunde liegende Geometrie ein, in der Bilder in einem Computer vorliegen und zeigt die Folgen für Liniensegmente auf. Abschnitt 3.1.3 stellt den Algorithmus vor, mit dem die Linienpunkte eines Liniensegments bestimmt werden.

Ordnungsprinzip “Richtung”. Als Ordnungsprinzip für Liniensegmente wird ihre Richtung identifiziert. Die Idee ist folgende: Die Bestandteile “Linienpunkte” eines Liniensegments folgen dann dem Ordnungsprinzip “Richtung”, wenn ihre Richtung der des Liniensegments entspricht. Abschnitt 3.1.4 beschreibt, wie die Richtung von Bildpunkten berechnet wird und welchen Einfluss die geforderte statistische Unabhängigkeit hat. In Abschnitt 3.1.5 ist festgelegt, wann die Richtung eines Linienpunkts der seines Liniensegments entspricht.

3.1.1 Linienbegriff

Bei der Betrachtung von Liniensegmenten in digitalen Bildern ist die zugrunde liegende Geometrie zu beachten. In der euklidischen Geometrie wird eine Linie eindeutig durch ihren Start- und Endpunkt oder ihren Startpunkt und Angabe von Länge und Richtung beschrieben.

In digitalen Bildern ist die euklidische Geometrie eingeschränkt durch ein diskretes Raster, auf dem die einzelnen Bildpunkte $\mathbf{b} = (i, j)$ angeordnet sind. Es liegt eine diskrete euklidische Geometrie vor, die hier vereinfacht auch als diskrete Geometrie bezeichnet wird.

In der diskreten Geometrie wird ein *Liniensegment* (diskrete Linie) L eindeutig beschrieben durch seine Linienpunkte beschrieben: Diese stellen eine geordnete Menge der Bildpunkte $\mathcal{B} = \{\mathbf{b}_1 = \mathbf{b}_s, \mathbf{b}_2, \dots, \mathbf{b}_l = \mathbf{b}_e\}$ dar. Eine Beschreibung über Startpunkt \mathbf{b}_s und Endpunkt \mathbf{b}_e oder über Startpunkt, Länge l und Richtung $r(L)$ ist möglich, führt aber nicht immer zum selben Liniensegment (Abbildung 3.1). Deswegen genügt die Betrachtung von \mathbf{b}_s , \mathbf{b}_e , l und $r(L)$ nicht. Allerdings lässt sich aus ihnen \mathcal{B} bestimmen (siehe Abschnitt 3.1.3).

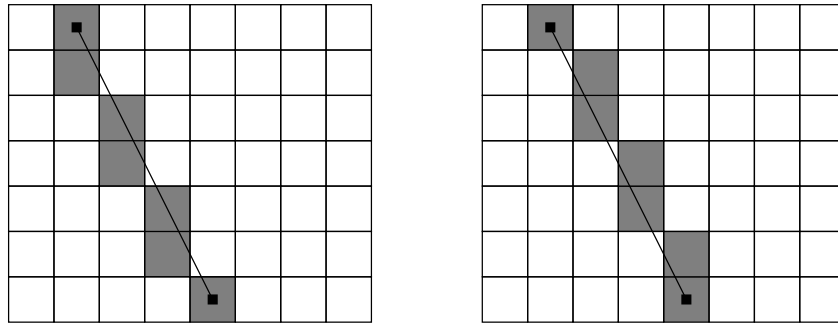


Abbildung 3.1: Mit demselben Start- und Endpunkt kann ein Liniensegment aus unterschiedlichen Linienpunkten zusammengesetzt sein.

3.1.2 Diskrete Geometrie

Koordinatensysteme. Das kartesische Koordinatensystem kann direkt auf das Bildraster und damit die diskrete Geometrie übertragen werden. Dazu wird die Vertikalachse mit $i \in \mathbb{N}$ und die Horizontalachse mit $j \in \mathbb{N}$ indiziert. Die linke obere Bildecke wird als Ursprung festgelegt.

Für Richtungsangaben ist der Übergang zu Polarkoordinaten praktischer: Die Richtung eines zweidimensionalen Richtungsvektors lässt sich eindeutig als Skalar r darstellen. $r(L)$ ist dabei der vom Liniensegment L und der Polarachse eingeschlossene Winkel.

Die Polarachse wird als Parallele zur Horizontalachse des kartesischen Koordinatensystems durch den betrachteten Linienpunkt eines Liniensegments definiert. r liegt demzufolge im Wertebereich des zyklischen Intervalls $r \in [-\pi, \dots, +\pi[$.

Abbildung 3.2 zeigt eine schematische Darstellung der beiden benutzten Koordinatensysteme.

Die Umrechnung von einem Richtungsvektor (i, j) in kartesischen Koordinaten zur Richtung $-\pi < r \leq \pi$ in Polarkoordinaten erfolgt nach [BSMM99]:

$$r(i, j) = \begin{cases} \arctan\left(\frac{i}{j} + \pi\right) & , j < 0 \\ \arctan\left(\frac{i}{j}\right) & , j > 0 \\ \frac{\pi}{2} & , j = 0, i > 0 \\ -\frac{\pi}{2} & , j = 0, i < 0 \\ \text{undefiniert} & , j = 0, i = 0 \end{cases} \quad (3.1)$$

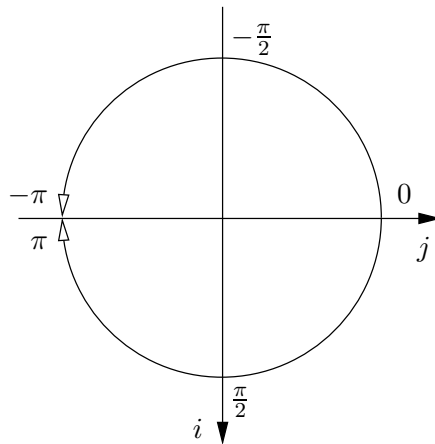


Abbildung 3.2: Kartesisches und polares Koordinatensystem. In dieser Abbildung entspricht die kartesische Horizontalachse j der Polarachse.

Bilddimension. Die Größe des Bildrasters eines digitalen Bild ist durch i_{\max}, j_{\max} begrenzt. Dafür wird folgende Schreibweise verwendet: Das Bild hat die Größe $i_{\max} \times j_{\max}$.

Die Bilddimension N wird mit $1 \leq i \leq i_{\max}, 1 \leq j \leq j_{\max}$ als geometrisches Mittel definiert:

$$N = \sqrt{i_{\max} j_{\max}} \quad (3.2)$$

Nachbarschaft. Die Festlegung der Nachbarschaft entscheidet über die Liniensegmente, aus denen ein Liniensegment besteht (Abbildung 3.3). Im Folgenden wird die *Schachbrett-Metrik* genannte 8er-Nachbarschaft verwendet.

Die Schachbrett-Metrik ist für die Anwendung des Helmholtz-Prinzips geeignet: Liniensegmente mit statistisch unabhängiger Richtung haben im Gegensatz zur Cityblock-Metrik einen äquidistanten Abstand voneinander. Das vereinfacht die Betrachtung von statistisch unabhängigen Bestandteilen "Linienpunkte" bezüglich dem Ordnungsprinzip "Richtung", da nicht für jeden Liniensegment eine unabhängige Richtung berechnet werden kann (siehe Abschnitt 3.1.4).

Anti-Aliasing von Liniensegmenten. Bei der Darstellung von Liniensegmenten über die Zuordnung zu einer (diskreten) Metrik, kommt es zu so genannten *Treppen*: Die Liniensegmente haben durch die Zuordnung der Liniensegmente auf das diskrete Raster keine "glatten Ränder" mehr (Abbildung 3.3 verdeutlicht dieses Phänomen).

Für die Darstellung von Liniensegmenten kommen daher oft Verfahren zum Einsatz, die unter dem Begriff *Anti-Aliasing* bekannt sind. Ohne näher auf den theoretischen Hintergrund von Aliasing einzugehen (Abtasttheorem, siehe [Sha48]), gilt für alle Anti-Aliasing Verfahren:

Um glatte Ränder zu erzeugen, wird zwischen dem Helligkeitswert der einzelnen Liniensegmente und den Helligkeitswerten der benachbarten Bildpunkte interpoliert. Moderne Anti-Aliasing Algorithmen leisten dies mit einer Subpixel-Filtermaske; ein Überblick des Themas findet sich z.B. in [Jäh97].

Da eine Interpolation zwischen den Helligkeitswerten der Liniensegmente keine statistisch unabhängige Richtung gewährleistet (siehe Abschnitt 3.1.4), wird auf Anti-Aliasing verzichtet: Liniensegmente, die mit einem Anti-Aliasing Verfahren erzeugt wurden, widersprechen der Annahme der Nullhypothese statistisch unabhängiger Ausgangsdaten.

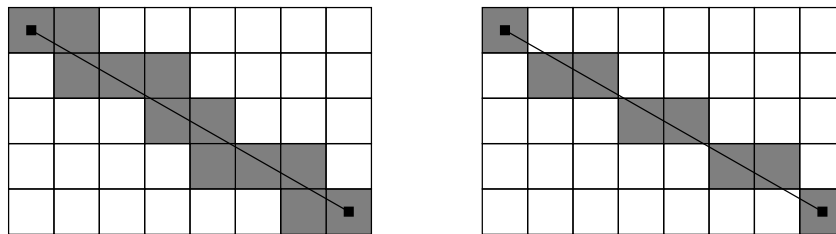


Abbildung 3.3: Links ein Liniensegment in der 4er-Nachbarschaft (*Cityblock-Metrik*). Rechts das gleiche Liniensegment in der 8er-Nachbarschaft (*Schachbrett-Metrik*). Beide Liniensegmente haben keine glatten Ränder mehr, sondern erscheinen als Treppen.

Distanz und Länge. Die Distanz $d(\mathbf{b}_s, \mathbf{b}_e)$ zwischen zwei Bildpunkten $\mathbf{b}_s = (s_i, s_j)$, $\mathbf{b}_e = (e_i, e_j)$ ist die Anzahl der Liniensegmente, aus denen ein Liniensegment L be-

steht, die an \mathbf{b}_s beginnt und an \mathbf{b}_e endet. Diese Distanz ist gleich der Länge eines Liniensegments l . Unter der Schachbrett-Metrik berechnet sich die Distanz mit

$$\begin{aligned} d(\mathbf{b}_s, \mathbf{b}_e) &\equiv l, \\ d(\mathbf{b}_s, \mathbf{b}_e) &= \max \left\{ |e_i - s_i|, |e_j - s_j| \right\} \end{aligned} \quad (3.3)$$

Zum gleichen Ergebnis gelangt man über die Betrachtung des Bildes als Graph (Bildraaster sind Kanten, Bildpunkte Knoten): Die Distanz ist dann kürzester Weg zwischen den Knoten \mathbf{b}_s und \mathbf{b}_e . In [Soi98] findet man eine ausführliche Beschreibung dieser Betrachtungsweise und eine Gegenüberstellung von euklidischer und diskreter Geometrie.

3.1.3 Linienpunkte eines Liniensegments

Dieser Abschnitt zeigt, wie die Linienpunkte eines Liniensegments algorithmisch bestimmt werden.

Im Folgenden sei der Startpunkt $\mathbf{b}_s = (s_i, s_j)$ und Endpunkt $\mathbf{b}_e = (e_i, e_j)$ eines Liniensegments gegeben. Die Linienpunkte werden mit der geordneten Menge \mathcal{B} bezeichnet.

Das bedeutet keine Einschränkung, da sich \mathbf{b}_s und \mathbf{b}_e aus allen anderen Beschreibungen eines Liniensegments leicht errechnen lassen: Bei Angabe von \mathcal{B} ist \mathbf{b}_s das erste, \mathbf{b}_e das letzte Element von \mathcal{B} .

Bei Beschreibung des Liniensegments über \mathbf{b}_s , l und $r(L)$ kann durch Umkehrung von Gleichung (3.1) die vektorielle Richtung bestimmt werden; eine Anpassung von Algorithmus I führt dann zu \mathcal{B} .

Richtung $r(L)$. Der Richtungsvektor (m_i, m_j) eines Liniensegments lässt sich mit

$$(m_i, m_j) = (e_i - s_i, e_j - s_j)$$

bestimmen. Die skalarwertige Richtung $r(L)$ des Liniensegments L ergibt sich nach Gleichung (3.1):

$$r(L) = r(m_i, m_j)$$

Länge l . Die Länge eines Liniensegments ergibt sich nach Gleichung (3.3):

$$l = d(\mathbf{b}_s, \mathbf{b}_e)$$

Liniensegmente \mathcal{B} . Um mit dem Startpunkt $\mathbf{b}_s = (s_i, s_j)$ und dem Endpunkt $\mathbf{b}_e = (e_i, e_j)$ die Liniensegmente $\mathcal{B} = \{\mathbf{b}_1 = \mathbf{b}_s, \dots, \mathbf{b}_l = \mathbf{b}_e\}$ zu bestimmen, wird eine Modifikation des Bresenham-Algorithmus aus [Bre65] verwendet.

Sein Ansatz ist die unterschiedliche Bestimmung der Liniensegmente von vertikal und horizontal orientierten Liniensegmenten. Diese Differenzierung ermöglicht einen effizienten Ganzzahl-Algorithmus.

Einmalig wird die *Hauptsteigungskomponente* \mathbf{m}_h des Liniensegments bestimmt. \mathbf{m}_h kann man interpretieren als gerichtete Orientierung des Liniensegments bezüglich der kartesischen Koordinatenachsen (Abbildung 3.4).

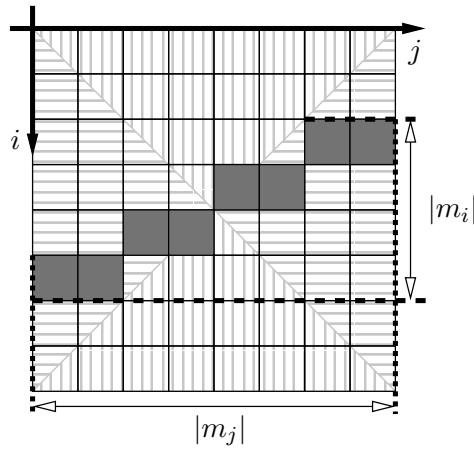


Abbildung 3.4: Orientierung von Liniensegmenten: Die horizontal bzw. vertikal schraffierte Fläche kennzeichnet den Bereich für horizontal bzw. vertikal orientierte Liniensegmente. Als Beispiel ist ein horizontal orientiertes Liniensegment mit $|m_j| > |m_i|$ auf dem Pixelraster abgetragen.

$$\forall \mathbf{b}_e \neq \mathbf{b}_s : \mathbf{m}_h = \begin{cases} (1, 0) & , |m_j| \geq |m_i|, m_j > 0 \\ (-1, 0) & , |m_j| \geq |m_i|, m_j < 0 \\ (0, 1) & , |m_j| < |m_i|, m_i > 0 \\ (0, -1) & , |m_j| < |m_i|, m_i < 0 \end{cases}$$

Die *Nebensteigungskomponente* \mathbf{m}_n wird für jeden Liniensegment mit dem Rundungsoperator *round* aus Anhang A.1.1 berechnet:

$$\forall \mathbf{b}_i, i = 2, \dots, l - 1 : \mathbf{m}_n = \begin{cases} (0, \text{round}(\frac{m_i}{l}i)) & , |m_j| \geq |m_i| \\ (\text{round}(\frac{m_i}{l}i), 0) & , |m_j| < |m_i| \end{cases} \quad (3.4)$$

Die Linienpunkte $\mathbf{b}_i \in \mathcal{B}$ ergeben sich als

$$\forall i = 2, \dots, l-1 : \mathbf{b}_i = \mathbf{m}_h + \mathbf{m}_n$$

Die Berechnung der Nebensteigungskomponente in Gleichung (3.4) ist algorithmisch ineffizient, da für jeden Linienpunkt die rechenaufwändigen Operationen Division, Rundung und Multiplikation durchgeführt werden müssen. Der in [Bre65] vorgestellte Algorithmus vermeidet diese Operationen, indem er nach einmaliger Vorberechnung rekursiv die Nebensteigungskomponente für jeden Linienpunkt \mathbf{b}_{n+1} über \mathbf{b}_n ausschließlich mit Addition bestimmt.

In Algorithmus I wird die Bestimmung der Linienpunkte für den Fall $|m_j| \geq |m_i|$ gezeigt. Der Fall $|m_j| < |m_i|$ funktioniert analog und ist aus Gründen der Übersichtlichkeit nicht ausgeführt.

Algorithmus I Bresenham Linienpunkte $\mathbf{b}_1, \dots, \mathbf{b}_l$ für $|m_j| \geq |m_i|$

Voraussetzung: $|m_j| \geq |m_i|$.

gegeben: $\mathbf{b}_s, \mathbf{b}_e$

Vorzeichenoperator *sign* aus Anhang A.1.2.

- 1: $\mathbf{b}_1 = \mathbf{b}_s, l = 1$
 - 2: $step_j \leftarrow sign(m_j)1, step_i \leftarrow sign(m_i)1$
 - 3: $m_j \leftarrow |2m_j|, m_i \leftarrow |2m_i|$
 - 4: $fr \leftarrow 2m_i - m_j$
 - 5: **while** $s_j < e_j$ **do**
 - 6: **if** $fr \geq 0$ **then**
 - 7: $s_i \leftarrow s_i + step_i$
 - 8: $fr \leftarrow fr - m_j$
 - 9: $s_j \leftarrow s_j + step_j$
 - 10: $fr \leftarrow fr + m_i$
 - 11: $l \leftarrow l + 1$
 - 12: $\mathbf{b}_l = (s_i, s_j)$
-

3.1.4 Richtung von Bildpunkten

Mit dem Ordnungsprinzip gleicher Richtung von Liniensegmenten und seinen Linienpunkten ist es notwendig, die Richtung $r(\mathbf{b})$ eines Bildpunkts \mathbf{b} zu bestimmen.

Ansatz hierfür ist die Beobachtung, dass Liniensegmente an Grenzen zwischen Regionen konstanter Helligkeit wahrgenommen werden. Der Gradient $\mathbf{g}(\mathbf{b})$ zeigt in die Richtung des größten Helligkeitsanstiegs an \mathbf{b} . Der Richtungsvektor $\mathbf{g}_\perp(\mathbf{b})$ verläuft senkrecht zum Helligkeitsanstieg und berechnet sich als Orthogonale zu $\mathbf{g}(\mathbf{b})$. Das

wird durch Rotation von $\mathbf{g}(\mathbf{b})$ um $\frac{\pi}{2}$ erreicht. Die skalare Richtung $r(\mathbf{b})$ ergibt sich als Winkel von $\mathbf{g}_\perp(\mathbf{b})$ im Polarkoordinatensystem.

Problem statistischer Unabhängigkeit. Der Gradient $\mathbf{g}(\mathbf{b})$ an einem Bildpunkt \mathbf{b} kann nur über die Nachbarschaft von \mathbf{b} berechnet werden. Die nach Abschnitt 2.4.2 geforderte statistische Unabhängigkeit bezüglich der Richtung steht im Konflikt dazu:

Bezieht man für die Berechnung von $\mathbf{g}(\mathbf{b})$ die Umgebung eines Bildpunkts \mathbf{b}_u mit ein, vergrößert sich der Abstand zum nächsten Bildpunkt \mathbf{b}_{u+1} mit statistisch unabhängiger Richtung: Die bereits berücksichtigte Umgebung von \mathbf{b}_u darf nicht noch einmal zur Berechnung der Richtung an \mathbf{b}_{u+1} verwendet werden.

Zugunsten der geforderten statistischen Unabhängigkeit der Richtungswerte voneinander wird zur Gradientenberechnung ein Binomialfilter 1. Ordnung gewählt, da er den Gradienten auf einer kleinen Nachbarschaft von 2×2 Bildpunkten berechnet (siehe Gleichung (3.5)).

Unter Annahme statistisch unabhängiger Helligkeitswerte $y(\mathbf{b})$ gilt dann: Jeder Gradient $\mathbf{g}(\mathbf{b}_{i+1})$ mit einer Distanz nach Gleichung (3.3) $d(\mathbf{b}_i, \mathbf{b}_{i+1}) \geq 2$ ist statistisch unabhängig von $\mathbf{g}(\mathbf{b}_i)$.

Methoden, die einen robusteren Gradienten erzeugen (z.B. Sobel-Operator; siehe [Jäh97] und Abschnitt 4.5), beziehen zur Berechnung eine größere Nachbarschaft mit ein und liefern daher weniger statistisch unabhängige Richtungswerte.

Die wesentlichen Nachteile, die aus der Gradientenberechnung mit dem Binomialfilter 1. Ordnung verglichen mit robusteren Methoden folgen, sind:

1. $\mathbf{g}(\mathbf{b})$ ist Gradient zum Mittelpunkt von $(i, j), (i + 1, j), (i, j + 1), (i + 1, j + 1)$, wird jedoch als Gradient für den Bildpunkt $\mathbf{b} = (i, j)$ festgelegt. Daher verschiebt sich der Bezug zum Bildraster: $(i, j) \rightarrow (i - \frac{1}{2}, j - \frac{1}{2})$.
2. Die Richtungen der Bildpunkte am rechten und unteren Bildrand lassen sich nicht berechnen, da die notwendige Umgebung fehlt. Daher ist die Größe eines Bildes bestehend aus Richtungswerten der Bildpunkte kleiner als Größe des Ausgangsbildes aus Helligkeitswerten der Bildpunkte: $i_{\max} \times j_{\max} \rightarrow (i_{\max} - 1) \times (j_{\max} - 1)$.
3. Minimale Breite im Ortsbereich und daher schlechte Eingrenzung im Frequenzbereich.

Gradientenberechnung. $\mathbf{g}(\mathbf{b})$ wird auf der 2×2 -Nachbarschaft mit einem Binomialfilter 1. Ordnung als Hochpass berechnet. Ein praktischer Nebeneffekt ist die effiziente algorithmische Umsetzbarkeit, da nur wenige Additionen benötigt werden:

$$\mathbf{g}(\mathbf{b}) = \frac{1}{2} \begin{pmatrix} -y(i, j) - y(i, j + 1) + y(i + 1, j) + y(i + 1, j + 1) \\ -y(i, j) - y(i + 1, j) + y(i, j + 1) + y(i + 1, j + 1) \end{pmatrix} \quad (3.5)$$

Richtungsberechnung. Rotation von $\mathbf{g}(\mathbf{b})$ um $\frac{\pi}{2}$ ergibt den Richtungsvektor $\mathbf{g}_\perp(\mathbf{b})$:

$$\begin{aligned} \mathbf{g}_\perp(\mathbf{b}) &= \begin{pmatrix} \cos\left(\frac{\pi}{2}\right) & -\sin\left(\frac{\pi}{2}\right) \\ \sin\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) \end{pmatrix} \mathbf{g}(\mathbf{b}) \\ &= \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \mathbf{g}(\mathbf{b}) \\ &= \frac{1}{2} \begin{pmatrix} -y(i, j + 1) - y(i + 1, j + 1) + y(i, j) + y(i + 1, j) \\ -y(i, j) - y(i, j + 1) + y(i + 1, j) + y(i + 1, j + 1) \end{pmatrix} \end{aligned}$$

Mit $\mathbf{g}_\perp(\mathbf{b}) = (g_{1\perp}, g_{2\perp})$ und Gleichung (3.1) ergibt sich der skalare Richtungswert $r(\mathbf{b}) \in [-\pi, \dots, +\pi[$ im Polarkoordinatensystem:

$$r(\mathbf{b}) = r(g_{1\perp}, g_{2\perp}) \quad (3.6)$$

Der entfallene Freiheitsgrad (der Abstand zum Ursprung im Polarkoordinatensystem) bei der Reduzierung des Richtungsvektors $\mathbf{g}_\perp(\mathbf{b})$ auf die skalarwertige Richtung $r(\mathbf{b})$ ist die Größenordnung des Helligkeitsunterschieds.

Diese ist für die spätere Detektion nicht ausschlaggebend: Liniensegmente finden sich nicht notwendigerweise an starken Kontrasten; sie werden charakterisiert durch ihre gerade Fortsetzung entlang einer Richtung. Das ist der Grund für die Kontrastinvarianz des hier vorgestellten Detektionsverfahrens.

3.1.5 Linienpunkte nach dem Ordnungsprinzip "Richtung"

Beim Vergleich zwischen der Richtung eines Linienpunkts $r(\mathbf{b})$, $\mathbf{b} \in \mathcal{B}$ und der Richtung $r(L)$ des dazugehörigen Liniensegments ist zu berücksichtigen, dass die Berechnung von $r(\mathbf{b})$ nach Gleichung (3.5) niemals exakt sein kann. Nur einige Gründe dafür sind:

1. Die Helligkeitswerte, die der Richtungsberechnung zugrunde liegen, sind notwendigerweise quantisiert und damit ungenau.

2. Die Auflösung eines digitalen Bildes ist begrenzt. Bei realen Bildern bestimmen Größe und Form des Sensors die räumlichen Abtastung der Umwelt.
3. Es gelten die in Abschnitt 3.1.4 erläuterten Einschränkungen für die Richtungsberechnung mit dem Gradienten aus Gleichung (3.5).

In Übereinstimmung mit [DMM00] wird deswegen hier angenommen, dass die Richtung $r(\mathbf{b})$ eines Linienpunkts der Richtung $r(L)$ eines Liniensegments entspricht, falls gilt:

$$\forall \mathbf{b} \in \mathcal{B} : |r(\mathbf{b}) - r(L)| \leq \frac{2\pi}{n} \quad (3.7)$$

Weiterhin gilt $n > 2$ nach Gleichung (2.1). In Abschnitt 4.3.1 wird gezeigt, welche Auswirkung die Wahl von $p = \frac{1}{n}$ hat und dass $p \simeq \frac{1}{16}$ für Liniensegmente eine gute Wahl ist.

Gerichteter Linienpunkt. Ein *gerichteter Linienpunkt* ist ein Linienpunkt, dessen Richtung nach Gleichung (3.7) der Richtung des Liniensegments entspricht, zu dem er gehört.

3.2 Bestimmung der Detektionsfunktion $k(l)$

In diesem Abschnitt wird die Detektionsfunktion $k(l)$ des Helmholtz-Prinzips für das Primitiv ‘‘Liniensegment’’ vorgestellt. Die ausführliche Herleitung der Detektionsfunktion ist in Abschnitt 2.4.2 gegeben, in diesem Abschnitt ist sie noch einmal kurz skizziert. Insbesondere wird dabei auf die Behandlung der statistisch abhängigen Linienpunkte eingegangen.

3.2.1 Richtung der Linienpunkte als gleichverteilte Zufallsvariablen

Die Wahrscheinlichkeit p dafür, dass die Richtung an einem Linienpunkt $r(\mathbf{b})$ der Richtung $r(L)$ seines Liniensegments nach Gleichung (3.7) entspricht, berechnet sich zu:

$$p := \mathrm{P} \left[r(\mathbf{b}) = r(L) \right] = \frac{1}{n}$$

Da $n > 2$, gilt $p < \frac{1}{2}$. Für $n = 16$ bedeutet das beispielsweise, dass sich die Richtungen von Linienpunkt und Liniensegment entsprechen, falls sie nicht mehr als $\pm 22,5^\circ$ voneinander abweichen. Man kann p daher interpretieren als die Präzision, mit der die Richtung eines Linienpunkts der Richtung seines Liniensegments entspricht.

Behandlung statistisch abhängiger Linienpunkte. Die Nullhypothese besagt für die Bestandteile “Linienpunkte” nach dem Ordnungsprinzip “Richtung”, dass die nach Gleichung (3.6) berechnete Richtung $r(\mathbf{b})$ auf zufällig gleichverteilten Helligkeitswerten basiert.

Da die Richtungsberechnung nach Gleichung (3.5) auf einer 2×2 Nachbarschaft erfolgt, kann die Richtung jedes zweiten Linienpunkts als statistisch unabhängige Zufallsvariable gesehen werden. Deshalb wird im Folgenden die Untermenge der Linienpunkte $\mathcal{B}_u = \{\mathbf{b}_i, i = 1, \dots, l_u\} \subset \mathcal{B}$ betrachtet, für die nach Gleichung (3.3) gilt: $d(\mathbf{b}_i, \mathbf{b}_{i+1}) = 2$.

Ein Liniensegment der Länge l besteht also aus $l_u = \lfloor \frac{l}{2} \rfloor$ Linienpunkten mit statistisch unabhängiger Richtung.

Es stellt sich die Frage, wie die Linienpunkte zu bewerten sind, die wegen statistischer Abhängigkeit ihrer Richtung unberücksichtigt bleiben.

Eine Möglichkeit ist die Annahme, dass die Linienpunkte $\mathbf{b} \in \mathcal{B}_u$ eines Liniensegments L für alle Linienpunkte $\mathbf{b} \in \mathcal{B}$ auf L repräsentativ sind.

Die vorsichtigeren und im Folgenden getroffene Annahme ist, dass die Richtung aller unberücksichtigten Linienpunkte nicht der Richtung ihres Liniensegments L nach Gleichung (3.7) entsprechen:

$$\forall \mathbf{b} \in \mathcal{B} \setminus \mathcal{B}_u : |r(\mathbf{b}) - r(L)| > \frac{2\pi}{n}$$

Plausibilisiert wird diese Annahme dadurch, dass nach Gleichung (2.1) gilt: $p < \frac{1}{2}$. Für die hier vorgestellten Liniensegmente wird $p = \frac{1}{16}$ gewählt; dies ist in Abschnitt 4.3.1 veranschaulicht. Damit ist die Wahrscheinlichkeit dafür, dass die Richtung eines Linienpunkts mit zufälliger Richtung der seines Liniensegments entspricht, viel kleiner als die Wahrscheinlichkeit, dass sich Richtung von Linienpunkt und Liniensegment nicht entsprechen.

3.2.2 Verteilung gerichteter Linienpunkte

Die gerichteten Linienpunkte mit statistisch unabhängiger Richtung bilden die Menge $\mathcal{B}_r \subseteq \mathcal{B}_u \subset \mathcal{B}$.

Die Wahrscheinlichkeit für $g = |\mathcal{B}_r|$ lässt sich mit der Binomialverteilung $B(g; l, p)$ formulieren:

$$B(g; l, p) := \mathrm{P} \left[|\mathcal{B}_r| = g, |\mathcal{B}| = l \right] = \binom{l}{g} p^g (1-p)^{l-g}$$

3.2.3 Auftrittswahrscheinlichkeit gerichteter Linienpunkte

Kriterium für die Detektion eines Liniensegments $L(g \geq k, l)$ mit l Linienpunkten und $g \geq k$ gerichteten Linienpunkten ist ihre Auftrittswahrscheinlichkeit.

Korrespondierend zu Abschnitt 2.3 ist ab einer gewissen Wahrscheinlichkeit ein zufälliges Auftreten eines Liniensegments so unwahrscheinlich, dass das Liniensegment wahrnehmbar ist.

Die Wahrscheinlichkeit für ein Liniensegment $L(g \geq k, l)$ berechnet sich als die Summe $P(k; l, p)$ über den rechtsseitigen Rand der Binomialverteilung:

$$P(k; l, p) := \mathbb{P} \left[|\mathcal{B}_r| \geq k, |\mathcal{B}| = l \right] = \sum_{i=k}^l B(i; l, p)$$

3.2.4 Erwartungswert für das Primitiv "Liniensegment" und Schranke ϵ

Anforderung an die zu bestimmende Detektionsfunktion $k(l)$ ist es, für alle möglichen Liniensegmente $L(g \geq k(l), l)$ eines Bildes mit Länge $2 < l < \max\{i_{\max}, j_{\max}\} = l_N$ zu gelten. Betrachtet man die Kombinationen der möglichen Start- und Endpunkte in einem Bild mit Bilddimension N nach Gleichung (3.2), erhält man als Gesamtanzahl aller möglichen Liniensegmente mit $m(l)$ Liniensegmenten mit Länge l

$$\sum_{l=2}^{l_N} m(l) = N^2(N^2 - 1) \tag{3.8}$$

Der Erwartungswert E für Liniensegmente aus l Linienpunkten mit mindestens $k(l)$ gerichteten Linienpunkten ist mit Gleichung (3.8):

$$E = \sum_{l=2}^{l_N} m(l) P(k; l, p)$$

ϵ wird nun so gewählt (siehe Gleichung (2.5) aus Abschnitt 2.4.2), dass gilt

$$E \leq \epsilon \quad \Leftrightarrow \quad P(k; l, p) \leq \frac{\epsilon}{N^2(N^2 - 1)}$$

Hier wird eine Relation zur Gesamtanzahl $N^2(N^2 - 1)$ hergestellt, die sich nicht mit dem Grenzwert ϵ ändert, sondern allein von der Bilddimension abhängt. Das erklärt den geringen Einfluss des Parameters ϵ für Liniensegmente (ϵ hat nur logarithmischen Einfluss auf die Detektionsfunktion, siehe Gleichung (2.24)).

3.2.5 Detektionsfunktion $k(l)$

Die formale Definition für die Detektionsfunktion $k(l)$ ist gemäß Gleichung (2.6) aus Abschnitt 2.4.2

$$k(l) = \min \left\{ k \in \mathbb{N}, P(k; l, p) \leq \frac{\epsilon}{N^2(N^2 - 1)} \right\}$$

Die mit der Rekursionsformel aus Gleichung (2.7) berechnete Detektionsfunktion $k(l)$ ist für $p = \frac{1}{16}$, $\epsilon = 1$, $N = 128$ in Abbildung 2.7, Abschnitt 2.4.2 zu sehen.

3.3 Suchraum

In diesem Abschnitt wird der Algorithmus beschrieben, mit dem ein Bild gemäß der Detektionsfunktion $k(l)$ nach Liniensegmenten durchsucht wird. Nach Gleichung (3.8) ist die Anzahl aller möglichen Liniensegmente in einem Bild $N^2(N^2 - 1)$. Zur besseren algorithmischen Erfassung wird der Suchraum gegliedert: Abschnitt 3.3.1 stellt dazu die so genannten “Rand-zu-Rand Segmente” als ersten Gliederungsschritt vor. Auf diesen wird dann mit dem in Abschnitt 3.3.2 beschriebenen Scan-Line Prinzip die eigentliche Detektion mit $k(l)$ vollzogen. Eine Untersuchung der Komplexität ist in Abschnitt 3.5.1 zu finden.

3.3.1 Rand-zu-Rand Segment L_R

Plausibilisierung und Definition. Die Gliederung des Suchraums (bestehend aus allen möglichen Liniensegmenten in einem Bild) erklärt sich durch folgende Beobachtung:

Die Linienpunkte \mathcal{B} jedes Liniensegments sind auch Linienpunkte eines Liniensegments, dessen Startpunkt \mathbf{b}_s auf einem Bildrand

$$\begin{aligned} R_s &\in \{(1, o := \text{oben}), (i_{\max}, u := \text{unten}), (1, l := \text{links}), (j_{\max}, r := \text{rechts})\}, \\ o, u &= 1, \dots, j_{\max} \quad l, r = 1, \dots, i_{\max} \end{aligned}$$

und Endpunkt \mathbf{b}_e auf einem anderen Bildrand

$$\begin{aligned} R_e &\in \{(1, o), (i_{\max}, u), (1, l), (j_{\max}, r)\}, \\ o, u &= 1, \dots, j_{\max} \quad l, r = 1, \dots, i_{\max} \quad R_e \neq R_s \end{aligned}$$

liegt. Ein solches Liniensegment wird im Folgenden *Rand-zu-Rand Segment* L_R genannt.

Um den Suchraum zu durchmustern, werden mit Algorithmus I alle Rand-zu-Rand Segmente von allen Bildpunkten auf den Bildrändern R_s zu allen Bildpunkten auf den Bildrändern R_e bestimmt. Mit Bilddimension N sind das $O(4N \cdot 3N) = O(N^2)$ Rand-zu-Rand Segmente L_R .

Unterschied zu den Rand-zu-Rand Segmenten in [DMM00]. Eine andere Möglichkeit zur Durchmusterung mit Rand-zu-Rand Segmenten ist es, von jedem der $4N$ Startpunkte alle L_R mit Richtungen $\frac{\pi}{x}, x = 1, \dots, q$ zu untersuchen. Diese Methode wurde von Desolneux et al. in [DMM00] vorgeschlagen.

Sie hat jedoch zwei Nachteile:

1. Die Granularität der Durchmusterung hängt von N ab, eine vollständige Durchmusterung ist nicht mehr gewährleistet. Abbildung 3.5 veranschaulicht diesen Nachteil.
2. Ein (unnötiger) zusätzlicher Parameter q muss bestimmt werden.

Verfeinerung von Rand-zu-Rand Segmenten. Bei der Bestimmung von Rand-zu-Rand Segmenten mit Algorithmus I ist unberücksichtigt, dass bei der Detektion nach Abschnitt 3.2 nur die Richtung jedes unabhängigen (d.h. zweiten) Linienpunkts ausgewertet wird.

Um eine anschließende effiziente Detektion von Liniensegmenten auf Rand-zu-Rand Segmenten mit dem Scan-Line Prinzip zu ermöglichen, wird Algorithmus I modifiziert:

1. Nur noch jeder zweite Linienpunkt (das sind die Linienpunkte mit unabhängiger Richtung $\mathcal{B}_u = \mathbf{b}_s, \dots, \mathbf{b}_e$) wird gespeichert.
2. Bei der inkrementellen Bestimmung der Linienpunkte wird die Folge a_r nach Gleichung (3.7) aufgestellt:

$$a_r = \begin{cases} 0 & , r(\mathbf{b}) \neq r(L_R) \\ 1 & , r(\mathbf{b}) = r(L_R) \end{cases} \quad (3.9)$$

3.3.2 Detektion mit Scan-Line Prinzip

Notation. Auf jedem Rand-zu-Rand Segment L_R werden alle Liniensegmente L_i der Länge $l = l_{\min}, \dots, l_{\max}$ untersucht.

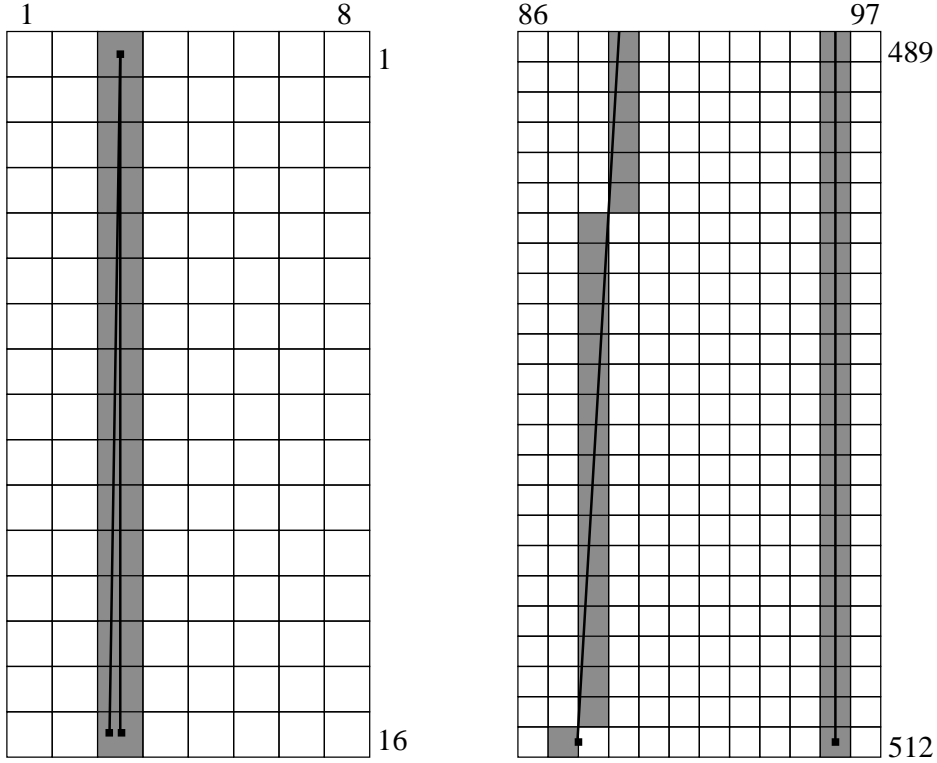


Abbildung 3.5: Zwei aufeinanderfolgende Rand-zu-Rand Segmente, die einen mit q quantisiertem Winkel $\varphi = \frac{\pi}{q}$, $q = 200$ einschließen. Während im linken Bild (16×8 , $\mathbf{b}_s = (1, 3)$) die aufeinanderfolgenden Rand-zu-Rand Segmente genau dieselben Punkte abtasten, zeigt sich im rechten Bild (Ausschnitt von 512×512 , $\mathbf{b}_s = (1, 96)$) eine Abtastlücke. Diese wird zwar zum Teil durch Rand-zu-Rand Segmente mit anderen Start- und Endpunkten abgetastet, alle Liniensegmente können so jedoch nicht erfasst werden.

l_{\min} ergibt sich aus Gleichung (2.12). l_{\max} ist das Liniensegment, die mit $\sum a_r$ als Anzahl der gerichteten Linienpunkte unter Berücksichtigung der Detektionsfunktion detektiert werden kann. Die Obergrenze von l_{\max} ist durch $l_R = l(L_R)$ gegeben, die Anzahl der Linienpunkte mit unabhängiger Richtung $|\mathcal{B}_u|$. Mit Gleichung (3.9) ergibt sich für l_{\max}

$$l_{\max} = \min \left\{ k \left(\sum_{r=1} l_R a_r \right), l_R \right\} \quad (3.10)$$

Um Verwechslungen mit dem Start- und Endpunkt von Rand-zu-Rand Segmenten

zu vermeiden, wird der Startpunkt von L_i mit s , der Endpunkt mit e bezeichnet mit $s, e \in \{1, \dots, l_R\}$. Eine Zuordnung zu den Bildpunkten \mathbf{b}_i ergibt sich über die Indizierung der Menge der Linienpunkte mit unabhängiger Richtung \mathcal{B}_u durch s, e .

Verringerung der Komplexität durch Scan-Line Prinzip. Sei L_i das Liniensegment auf L_R mit Länge $l = l_R$. Betrachtet man die Anzahl der Liniensegmente mit Länge $l = l_R - 1$, so ist anschaulich klar, dass es zwei davon gibt. Weiterhin ist anschaulich klar, dass es l_R Liniensegmente mit Länge $l = 1$ auf L_R gibt.

Die Anzahl $c(l_i)$ an Liniensegmenten L_i mit Länge l_i auf L_R mit Länge l_R gilt:

$$c(l_i) = \sum_{i=l_i}^{l_R} i \quad (3.11)$$

Die Anzahl g an gerichteten Linienpunkte auf L_i , beginnend an s und an e endend, wird mit der Detektionsfunktion $k(l)$ aus Gleichung (2.6) verglichen. Falls

$$g \geq k(l), \quad (3.12)$$

wird das Liniensegment L_i detektiert. Um g zu bestimmen, ist die Summe über den betreffenden Teil der Folge a_r aus Gleichung (3.9) zu bilden:

$$g = \sum_{i=s}^e a_i \quad (3.13)$$

Die Summe aus Gleichung (3.13) muss mit Verwendung des Scan-Line Prinzips (siehe z.B. [OW96]) nicht für jedes Liniensegment L_i auf dem Rand-zu-Rand Segment L_R komplett berechnet werden. Für die $c(l_i)$ Liniensegmente (Gleichung (3.11)) auf L_R mit gleicher Länge l_i lässt sich g effizient inkrementell berechnen. Das Scan-Line Prinzip wird in Algorithmus II konkretisiert.

3.4 Demaskierung

In diesem Abschnitt wird das in Abschnitt 2.2 informell beschriebene Maskierungsproblem behandelt. Dabei werden für Liniensegmente zwei unterschiedliche Arten von Maskierung identifiziert: In Abschnitt 3.4.1 wird gezeigt, dass sich Liniensegmente auf dem gleichen Rand-zu-Rand Segment maskieren können. Diese Art von Maskierung lässt sich als *Maskierung durch Ergänzung* aus Abschnitt 2.2 interpretieren. Abschnitt 3.4.2 behandelt Maskierungen von Liniensegmenten, die nicht auf dem gleichen Rand-zu-Rand Segment liegen und eher mit *Maskierung durch Einbettung in Textur* korrespondieren.

Algorithmus II Scan-Line Detektion

Voraussetzung: $reg(L, s, e)$ legt aktuelles Liniensegment L in Detektionsliste ab.

```

1: for  $l = l_{\min}, \dots, l_{\max}$  do
2:    $g \leftarrow \sum_{r=1}^{l-1} a_r$ 
3:    $s \leftarrow 1$ 
4:   for  $e = l, \dots, l_R$  do
5:      $g \leftarrow g + a_e$ 
6:     if  $g \geq k(l)$  then
7:        $reg(L, s, e)$ 
8:      $g \leftarrow g - a_s$ 
9:      $s \leftarrow s + 1$ 

```

3.4.1 Direkte Demaskierung

Zustandekommen direkter Maskierung. Wie in Abschnitt 3.3.1 dargestellt, können die Linienspunkte jedes Liniensegments $L_u \neq L_R$ aufgefasst werden als Teilmenge der Linienspunkte eines Liniensegments L_o . Anschaulich bedeutet das, dass das Liniensegment L_u vollständig von L_o verdeckt wird. Falls sowohl L_u als auch L_o mit Algorithmus II als Liniensegmente detektiert werden, so *maskiert* L_o L_u .

Demaskierte Liniensegmente L_d . Um diese Maskierung zu lösen, ist es erforderlich zu entscheiden, ob L_u oder L_o zu vernachlässigen ist. Nach der Idee des Helmholtz-Prinzip ist das Liniensegment zu vernachlässigen, dessen Auftrittswahrscheinlichkeit $P(k; l, p)$ nach Gleichung (3.8) größer ist:

Ein detektiertes Liniensegment ist dann ein demaskiertes Liniensegment L_d , wenn gilt:

1. Es maskiert kein Liniensegment, dessen Auftrittswahrscheinlichkeit kleiner ist:

$$\forall L_u \subset L_d : P(k; l(L_u), p) \geq P(k; l(L_d), p) \quad (3.14)$$

2. Es wird von keinem Liniensegment maskiert, dessen Auftrittswahrscheinlichkeit kleiner ist:

$$\forall L_o \supset L_d : P(k; l(L_o), p) \geq P(k; l(L_d), p) \quad (3.15)$$

Eigenschaften demaskierter Liniensegmente. Zwei wesentliche Eigenschaften demaskierter Liniensegmente folgen unmittelbar aus den Eigenschaften der Binomialverteilung aus Abschnitt 2.5.2:

1. Wegen Gleichung (2.9) gilt: Startpunkt s und Endpunkt e eines demaskierten Liniensegments sind gerichtete Linienpunkte.
2. Wegen Gleichung (2.10) gilt: Falls $s-1$ bzw. $e+1$ gültige Bildpunkte sind (d.h. $s > 1$ bzw. $e < l_R$), dann sind $s-1$ bzw. $e+1$ keine gerichteten Linienpunkte.

Diese beiden Eigenschaften lassen sich direkt in Algorithmus II anwenden. Das Beispiel in Abschnitt 4.2 zeigt, dass ihre Berücksichtigung bereits einen wesentlichen Teil zur Demaskierung auf Rand-zu-Rand Segmenten beitragen.

Eine dritte Eigenschaft kann aus Gleichung (3.14) und Gleichung (3.15) abgeleitet werden:

3. Zwei voneinander verschiedene demaskierte Liniensegmente $L_d, L_{d'}$ auf einem Rand-zu-Rand Segment L_R haben keine gemeinsamen Linienpunkte.

Algorithmische Konkretisierung. Um Liniensegmente L_i von anderen Liniensegmenten $L_{i'}$ auf demselben Rand-zu-Rand Segment L_R vollständig zu demaskieren, wird hier folgender Vorgehensweise benutzt: Die Auftrittswahrscheinlichkeit jedes detektierten Liniensegments L_i , welches den Eigenschaften 1 und 2 genügt, wird zunächst in einer Liste demaskierter Liniensegmente gespeichert.

Maskiert ein anderes Liniensegment $L_{i'}$ auf L_R , welches ebenfalls Eigenschaften 1 und 2 genügt, bei nachfolgender Detektion das Liniensegment L_i oder wird von ihm maskiert, so wird das Liniensegment aus $\{L_i, L_{i'}\}$ verworfen, welches die größere Auftrittswahrscheinlichkeit besitzt.

Um diese Maskierung festzustellen, wird für jedes Rand-zu-Rand Segment eine zu 0 initialisierte *Demaskierungsliste* $D(l = 1, \dots, l(L_R)) = [p, id]$ geführt, die für jeden Linienpunkt einen Eintrag mit der Wahrscheinlichkeit $p(L)$ und einen Identifizierungseintrag $id(L) > 0$ für das korrespondierende Liniensegment enthält. Für jedes detektierte Liniensegment $L_i(s, e)$ wird mittels der Demaskierungsliste überprüft, ob innerhalb von $[s, e]$ bereits ein Liniensegment $L_{i'}$ detektiert ist; falls ja, wird das Liniensegment mit der größeren Wahrscheinlichkeit verworfen.

Eigenschaft 3 erlaubt unter der Voraussetzung, dass Algorithmus II Liniensegmente beginnend mit Maximallänge l_R bis hin zur Minimallänge l_{\min} untersucht, eine effiziente Realisierung direkter Demaskierung. Diese ist in Algorithmus III dargestellt.

3.4.2 Wechselseitige Demaskierung

Zustandekommen wechselseitiger Maskierung. Bei digitalen Bildern treten Maskierungen auf, die auf die Natur der zugrunde liegenden diskreten Geometrie zurück-

Algorithmus III Scan-Line Detektion mit direkter Demaskierung**Voraussetzung:** $a_{r < 1} = 0, a_{r > l(R)} = 0$;Demaskierungsliste $D(l = 1, \dots, l(L_R)) = [0, 0]$; $reg(L, s, e, id)$ legt aktuelles Liniensegment L in Detektionsliste ab. $newid()$ erzeugt eindeutige Nummer für Detektionsliste.

```

1: for  $l = l_{\max}, \dots, l_{\min}$  do
2:    $g \leftarrow \sum_{r=1}^{l-1} a_r$ 
3:    $s \leftarrow 1$ 
4:   for  $e = l, \dots, l_R$  do
5:      $g \leftarrow g + a_e$ 
6:     if  $g \geq k(l)$  and  $a_e = 1, a_{e+1} = 0, a_s = 1, a_{s-1} = 0$  then
7:       if  $D(s)[p] > k(l)$  then
8:          $id = D(s)[id]$ 
9:          $D(i = s(L_{id}), \dots, e(L_{id})) = [0, 0]$ 
10:         $reg(L, s, e, id)$ 
11:         $D(i = s, \dots, e) = [p(L_{id}), id]$ 
12:       else if  $D(s)[p] = 0$  then
13:          $id = newid()$ 
14:          $reg(L, s, e, id)$ 
15:          $D(i = s, \dots, e) = [p(L_{id}), id]$ 
16:      $g \leftarrow g - a_s$ 
17:      $s \leftarrow s + 1$ 

```

zuführen sind: Liniensegmente maskieren andere, obwohl sie nicht die gleiche Richtung haben.

Während im euklidischen Raum zwei Linien unterschiedlicher Richtung höchstens einen Schnittpunkt haben, bilden die entsprechenden zwei Liniensegmente in der diskreten Geometrie eine Schnittmenge aus gemeinsamen Bildpunkten, deren Kardinalität nicht auf 1 beschränkt ist.

Das bedeutet, dass zwei Liniensegmente unterschiedlicher Richtung mehr als zwei Schnittpunkte haben können. Abbildung 3.6 und Abbildung 3.7 veranschaulichen diese Art von *wechselseitiger Maskierung*; um die Darstellung zu vereinfachen, wird hier die Richtung jedes Liniensegmentes als unabhängig angenommen.

Abbildung 3.6 ist ein Beispiel für typische reale Eingangsdaten: Die Richtung der Liniensegmente stimmt nicht überall mit der Richtung der Liniensegmente überein (Rauschen), die Richtung an benachbarten Bildpunkten stimmen dagegen teilweise mit der Richtung der Liniensegmente überein (begrenzte räumliche Abtastung).

Abbildung 3.7 ist ein Beispiel dafür, dass auch bei idealisierten Eingangsdaten (alle Liniensegmente auf L_1, L_2 sind gerichtet, daneben gibt es keine gerichteten Liniensegmente)

punkte) wechselseitige Maskierungen auftreten, wenn es Liniensegmente gibt, die näherungsweise parallel sind.

Während die Fehldetektion im zweiten Beispiel bei ausreichend hoher Präzision für Gleichung (3.7) vermieden werden könnte, hätte das im ersten Beispiel eine Detektion der vorhandenen Linienstrukturen verhindert (siehe Abschnitt 3.1.5).

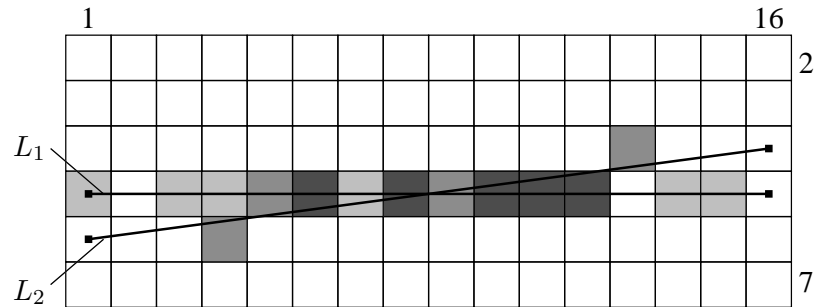


Abbildung 3.6: Ausschnitt aus 16×16 Bildpunkten mit unabhängiger Richtung ($p = \frac{1}{16}, \epsilon = 1$). Gerichtete Linienpunkte für L_1 sind hell, für L_2 etwas dunkler und für $L_1 \cap L_2$ dunkel gekennzeichnet. Auf L_1 wird ein direkt demaskiertes Liniensegment mit $\mathbf{b}_s = (3, 1), \mathbf{b}_e = (4, 15)$ und auf L_2 eines mit $\mathbf{b}_s = (5, 4), \mathbf{b}_e = (3, 13)$ erkannt. Offensichtlich handelt es sich aber um die gleiche Linienstruktur.

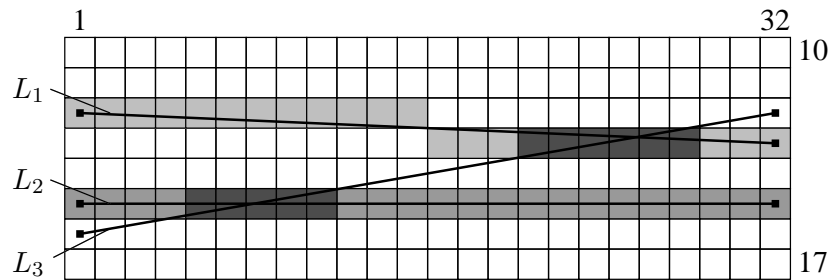


Abbildung 3.7: Ausschnitt aus 32×32 Bildpunkten mit unabhängiger Richtung ($p = \frac{1}{16}, \epsilon = 1$). Gerichtete Linienpunkte für L_1 sind hell, für L_2 etwas dunkler und für L_3 dunkel gekennzeichnet. Wechselseitig demaskierte Liniensegmente bestehen offensichtlich aus allen Linienpunkten von L_1 und L_2 . Zusätzlich wird ein direkt demaskiertes Liniensegment auf L_3 mit $\mathbf{b}_s = (16, 5), \mathbf{b}_e = (13, 29)$ durch wechselseitige Maskierung erkannt. Dieses direkt demaskierte Liniensegment ist eine Fehldetektion.

Ansatz für wechselseitige Demaskierung. Da die wechselseitige Maskierung bei Liniensegmenten auftritt, die verschiedene Richtungen aufweisen, kann sie anders als die direkte Maskierung nicht schon bei der Detektion von Liniensegmenten verhindert werden. Um wechselseitige Maskierung nachträglich zu demaskieren, wird folgende Annahme getroffen:

Jeder Bildpunkt ist höchstens gerichteter Liniensegmentpunkt *eines* Liniensegments.

Diese Annahme wird plausibilisiert durch die Beobachtung, dass in der (nicht diskreten) euklidischen Geometrie zwei Liniensegmente L_1, L_2 verschiedener Richtung höchstens einen gemeinsamen Schnittpunkt besitzen. Die Richtung dieses Schnittpunkts entspricht weder der Richtung von L_1 noch der von L_2 , da der Gradient an der Stelle des Schnittpunkts weder orthogonal zur Richtung von L_1 noch zur Richtung von L_2 ist.

Daraus folgt, dass in der euklidischen Geometrie jeder Punkt einer Linie, der dieselbe Richtung wie sie besitzt, auf keiner anderen Linie liegt.

Sei \mathbf{m} mit Richtung $r(\mathbf{m})$ nach Gleichung (3.6) ein gerichteter Liniensegmentpunkt mehrerer Liniensegmente L_1, \dots, L_m mit Richtungen $r(L_i, i = 1, \dots, m)$.

Für die exklusive Zuordnung von \mathbf{m} zu einem Liniensegment $L_z, z \in \{1, \dots, m\}$ werden zwei Kriterien vorgestellt: Zuordnung durch nachträgliche Anpassung der Präzision und Zuordnung durch minimale Wahrscheinlichkeit.

Nach dieser Zuordnung wird die Anzahl gerichteter Liniensegmentpunkte g aller nach Algorithmus III direkt demaskierten Liniensegmente aktualisiert, so dass alle gerichteten Liniensegmentpunkte für ein Liniensegment exklusiv sind:

$$g(L_z) = \sum_{\mathbf{m} \in L_z} 1 \quad (3.16)$$

Falls Gleichung (3.12) nicht mehr erfüllt ist, also

$$g(L_z) < k(l(L_z)), \quad (3.17)$$

dann wird L_z aus der Liste detektierter Liniensegmente entfernt.

Zuordnung durch nachträgliche Anpassung der Präzision. Eine Erklärung für die mehrfache Zuordnung von \mathbf{m} ist die mangelnde Präzision, mit der nach Gleichung (3.7) die Richtung $r(\mathbf{m})$ der Richtung eines Liniensegments $r(L_i), i = 1, \dots, m$ entspricht. Daher kann \mathbf{m} dem Liniensegment L_z zugeordnet werden, dessen Richtung am ähnlichsten zu ihm ist:

$$\mathbf{m} \in L_z \quad \Leftrightarrow \quad \min \left\{ |r(\mathbf{m}) - r(L_z)| \right\}, z \in \{1, \dots, m\}$$

Zuordnung nach minimaler Wahrscheinlichkeit. Eine andere Erklärung für die mehrfache Zuordnung ist, dass der Ansatz für die Detektionsfunktion $k(l)$ nicht die lokale Häufung von Liniensegmenten berücksichtigt: Das Helmholtz-Prinzip geht von rein zufällig verteilten Bestandteilen aus. Tatsächlich zeigen Abbildung 3.6 und Abbildung 3.7, dass in der diskreten Geometrie eine Linienstruktur zu mehreren Detektionen führt.

Um dies nachträglich für die wechselseitige Demaskierung zu nutzen, wird \mathbf{m} dem Liniensegment L_z zugeordnet, dessen Auftretswahrscheinlichkeit am geringsten ist. Dieses Verfahren wird von Desolneux et al. in [DMM00] angewendet.

$$\mathbf{m} \in L_z \quad \Leftrightarrow \quad \min \left\{ B(g(L_z); l(L_z), p) \right\}, z \in \{1, \dots, m\}$$

Vergleich der Zuordnungskriterien. Beim Test des Algorithmus (siehe Abschnitt 4.2) ergibt sich, dass die Zuordnung über nachträgliche Präzisionsanpassung in den Fällen bessere Ergebnisse liefert, in denen die Bildqualität gut genug für eine annähernd exakte Bestimmung der Richtung der Bildpunkte nach Gleichung (3.5) ist.

Ihr wesentlicher Nachteil ist allerdings, dass unter Umständen, die in Abschnitt 4.2 demonstriert werden, keine Detektion mehr gelingt, da die Zuordnung der Linienpunkte einer Linienstruktur auf zu viele verschiedenen Liniensegmente, die in ähnlicher Richtung verlaufen, verteilt wird. Die einzelnen Liniensegmente haben dann jeweils zu wenige ihnen zugeordnete Linienpunkte, um noch detektiert zu werden.

Die Zuordnung nach minimaler Wahrscheinlichkeit liefert selbst dann noch gute Ergebnisse, wenn die Bildqualität schlecht ist. Aus diesem Grund ist sie der nachträglichen Präzisionsanpassung vorzuziehen. Ansonsten ist das hier vorgestellte Detektionverfahren mit wechselseitiger Demaskierung sehr abhängig von der Qualität der Eingangsdaten.

3.5 Komplexität der Laufzeit und Verbesserungsansätze

3.5.1 Komplexität der Laufzeit

In diesem Abschnitt wird die Komplexität des Algorithmus abgeschätzt. Die hier getroffenen Aussagen sind deswegen unabhängig von der Art der Implementierung. Die Testergebnisse in Kapitel 4 sind durch ein mit [Matlab] implementiertes Programm erzeugt.

Notation und Wertung der Operationen. Die benutzten Algorithmen (Algorithmus I, Algorithmus II und Algorithmus III) sind so konzipiert, dass sie rechenintensive Operationen –wie Divisionen– vermeiden. Aus diesem Grund wird für jede Operation eine Zeiteinheit von $O(1)$ angenommen. Die Komplexität in Form von Abschätzungen mit der O-Notation wird im Folgenden mit K gekennzeichnet.

N steht dabei vereinfachend für eine Bilddimension mit $i_{\max} = j_{\max}$ gemäß Gleichung (3.2) aus Abschnitt 3.1.

Modellierungsschritt

Der Modellierungsschritt besteht aus der Richtungsberechnung an den Bildpunkten mit Gleichung (3.6) in Abschnitt 3.1.4 (wird nicht für den rechten und unteren Rand berechnet):

$$\begin{aligned} K_{Modell} &= \underbrace{2 \cdot 4}_{\text{Ops.}} \underbrace{(N-1)^2}_{\text{Bildpunkte}} \\ &= O(N^2) \end{aligned}$$

Bestimmung der Detektionsfunktion

Die Bestimmung der Detektionsfunktion erfolgt mit der rekursiven Berechnung nach Gleichung (2.7) aus Abschnitt 2.4.2: $P(k+1, l+1) = pP(k, l) + (1-p)P(k+1, l)$.

Da das längste Liniensegment N Linienpunkte enthält, die alle gerichtete Linienpunkte sein können, werden alle Kombinationen $P(k, l); k = 1, \dots, N; l = 1, \dots, N$ berechnet und es folgt dafür die Komplexität

$$K_{Kombinationen} = O(N^2)$$

Die Bestimmung des Minimums nach Gleichung (2.6) für jede Länge $l = 1, \dots, N$ erfolgt bei der Berechnung aller Kombinationen und ergibt so nur einen konstanten Vorfaktor $M = \text{const.}$. Damit ergibt sich die Komplexität $K_{Detektionsfunktion}$ für das Aufstellen der Detektionsfunktion

$$K_{Detektionsfunktion} = M K_{Kombinationen} = M O(N^2)$$

Durchmusterung des Suchraums

Die Durchmusterung des Suchraums ist der aufwändigste Schritt der vorgestellten Detektion von Liniensegmenten. Die Gesamtkomplexität $K_{Suchraum}$ ergibt sich für

jedes der $O(N^2)$ Rand-zu-Rand Segmente L_R (siehe Abschnitt 3.3.1) aus dem Bestimmen der Linienpunkte mit Algorithmus I mit Komplexität $K_{Bresenham}$ und der Detektion über das Scan-Line Prinzip mit Algorithmus II mit Komplexität $K_{Scanline}$:

$$K_{Suchraum} = O(N^2)(K_{Bresenham} + K_{Scanline}) \quad (3.18)$$

Bresenham Algorithmus. Für die Berechnung eines Rand-zu-Rand Segments mit Länge l_R ergibt sich mit Algorithmus I

$$\begin{aligned} K_{Bresenham} &= 4 + 8 \underbrace{(l_R - 1)}_{< N} \\ &= B O(N), \quad B = const. \end{aligned}$$

Detektion auf einem Rand-zu-Rand Segment. Die Detektion auf einem Rand-zu-Rand Segment erfolgt mit Algorithmus II. Da dieser die Komplexität dominiert, wird er ausführlich abgeschätzt.

Für *ein* Liniensegment mit Länge l (ein Durchlauf der inneren Schleife aus Algorithmus II) ergibt sich die Komplexität

$$K_{Liniensegment} = C, \quad C = const.$$

Damit kann Algorithmus II mit Komplexität $K_{Scanline}$ wie folgt abgeschätzt werden:

$$\begin{aligned} K_{Scanline} &= \sum_{l=l_{\min}}^{l_{\max}} \left(\sum_{s=1}^{l-1} O(1) + O(1) + \sum_{e=l}^{l_R} K_{Liniensegment} \right) \\ &< \sum_{l=1}^N \left(O(l) + \sum_{e=1}^{N-l} C \right) \quad (3.19) \\ &= \sum_{l=1}^N \underbrace{(O(l) + C O(N) - C O(l))}_{C^* O(N)} \\ &= C^* O(N^2) \end{aligned}$$

Gesamtaufwand für die Suchraumdurchmusterung. Nach Gleichung (3.18) ergibt sich für die Komplexität bei der Durchmusterung des Suchraums

$$\begin{aligned} K_{Suchraum} &= O(N^2)(B O(N) + C^* O(N^2)) \\ &= C^* O(N^4) \end{aligned}$$

Dabei ist anzumerken, dass dies eine *worst case* Abschätzung darstellt. Bei der praktischen Anwendung ergeben sich große Unterschiede für die Laufzeit. Diese resultieren aus der Abschätzung aus Gleichung (3.19) für l_{\max} (siehe Gleichung (3.10)): Falls nur wenige Richtungen an Linienpunkten mit der Richtung ihres Rand-zu-Rand Segments übereinstimmen, dann ist $l_{\max} < N$ oder sogar $l_{\max} < l_{\min}$. Für den zweiten Fall ergibt sich der *best case* $K_{Scanline} = \text{const.} \lll O(N^2)$. Dann bestimmt $K_{Bresenham} = B O(N^3)$ die Komplexität $K_{Suchraum}$.

Ein *average case* ist nicht zu berechnen, da nicht abschätzbar ist, ob und wieviele gerichtete Linienpunkte und damit detektierbare Liniensegmente ein Rand-zu-Rand Segment enthält.

Demaskierung

Die Demaskierung der hier vorgestellten Detektion von Liniensegmenten gliedert sich in die direkte Demaskierung (Abschnitt 3.4.1) und die wechselseitige Demaskierung (Abschnitt 3.4.2).

Direkte Demaskierung als Ergänzung zur Durchmusterung. Wie in Algorithmus III zu sehen ist, besteht die direkte Demaskierung aus einer Erweiterung von Algorithmus II. Für die Komplexität der Laufzeit gilt daher die Größenordnung aus dem vorangegangenen Abschnitt 3.5.1 mit $C^* \rightarrow C_2^* > C^*$.

Daran ist auch zu sehen, dass in dem hier vorgestellten Verfahren die direkte Demaskierung schon während der Durchmusterung des Suchraums durchgeführt wird. Das korrespondiert mit der Darstellung des Helmholtz Prinzips in Abbildung 2.6.

Direkte Demaskierung zur Vermeidung von Speicherproblemen. Der direkten Demaskierung kommt eine weitere wichtige Aufgabe zu: Wie an der Testsequenz in Abschnitt 4.2 zu sehen ist, wird schon für kleine Bilder ohne Demaskierung eine große Anzahl an Liniensegmenten detektiert. Da die direkte Demaskierung während der Detektion stattfindet und detektierte maskierte Liniensegmente verworfen werden, wird möglicher Speicherknappheit für das Zwischenspeichern vieler Liniensegmente vorgebeugt.

Wechselseitige Demaskierung. Die Komplexität wechselseitiger Demaskierung $K_{Wechselseitig}$ wird durch zwei Stufen bestimmt: Zunächst wird jeder Bildpunkt bezüglich der Liniensegmente untersucht, die ihn beinhalten: Je nach Zuordnungskriterium (siehe Abschnitt 3.4.2) wird jeder Punkt exklusiv einem Liniensegment

zugeordnet. Danach werden die Liniensegmente erneut mit der Detektionsfunktion überprüft. Es ergibt sich

$$K_{\text{Wechselseitig}} = K_{\text{Punktexklusion}} + K_{\text{Zweitdetektion}} \quad (3.20)$$

Wechselseitige Demaskierung: Punktexklusion. Im Folgenden wird die Anzahl $s(N)$ der Liniensegmente abgeschätzt, die denselben Linienpunkt enthalten können. Diese Abschätzung gilt für den Fall, dass die wechselseitige Demaskierung *nach* direkter Demaskierung durchgeführt wird. Da nach der direkten Demaskierung kein Linienpunkt auf einem Rand-zu-Rand Segment Bestandteil von mehr als einem Liniensegment sein kann, ist die Betrachtung der Rand-zu-Rand Segmente entscheidend, die an den $4N$ verschiedenen Randpunkten beginnen. Dabei ergeben sich folgende Konstanten für $ds(N)$:

- Mit $p < \frac{1}{2}$ nach Gleichung (2.1) folgt, dass ein Linienpunkt \mathbf{b} von zwei Rand-zu-Rand Segmenten, die von gegenüberliegenden Bildrändern starten, nicht gerichteter Linienpunkt von Liniensegmenten beider Rand-zu-Rand Segmente sein kann. Damit ergibt sich der Vorfaktor $\frac{1}{2}$ für die Abschätzung von $s(N)$.
- Ein Bildpunkt \mathbf{b} ist Linienpunkt von $N \ggg G(N) \geq 1$ Rand-zu-Rand Segmenten, die am *gleichen* Randpunkt beginnen. $G(N)$ zu bestimmen, kommt aber der Lösung eines geometrischen Kombinationsproblems gleich und wird hier nicht weiter untersucht.

Damit ergibt sich

$$\begin{aligned} s(N) &= 2G(N)N \\ &= G(N) O(N) \end{aligned}$$

und für alle N^2 Bildpunkte die *worst case* Abschätzung

$$\begin{aligned} K_{\text{Punktexklusion}} &= s(N)N^2 \\ &= G(N) O(N^3) \end{aligned}$$

Wechselseitige Demaskierung: Zweitdetektion. Die Komplexität der zweiten Stufe kann nur grob geschätzt werden: Jedes der $d(N)$ direkt demaskierten Liniensegmente wird nach exklusiver Punktzuordnung noch einmal mit der Detektionsfunktion $k(l)$ geprüft. Der Vergleich mit $k(l)$ selbst hat einen Aufwand von $O(1)$, allerdings ist die Größenordnung von $d(N)$ sehr variabel: Noch stärker als $s(N)$ ist $d(N)$ vom Inhalt des Bildes abhängig.

Eine *worst case* Abschätzung für $d(N)$ ergibt sich aus der Betrachtung der Anzahl $O(N^2)$ aller Rand-zu-Rand Segmente aus Abschnitt 3.3.1 und der möglichen Anzahl $n(N)$ der darauf detektierten Liniensegmente. $n(N)$ ergibt sich aus der Länge $l_R < N$ eines Rand-zu-Rand Segments und der minimalen Länge eines detektierten Liniensegments aus Gleichung (2.12) mit $N_P = N^2(N^2 - 1) \approx N^4$:

$$\begin{aligned} n(N) &= N \frac{\log p}{\underbrace{-\log N^4 + \log \epsilon}_{O(1/\log N^4), \text{ da } p < 0.5}} \\ &= O\left(\frac{N}{\log N^4}\right) \end{aligned}$$

Daraus folgt

$$\begin{aligned} K_{\text{Zweitdetektion}} &= \underbrace{n(N) O(N^2)}_{d(N)} O(1) \\ &= O\left(\frac{N^3}{\log N^4}\right) \end{aligned}$$

Die *worst case* Abschätzung für die wechselseitige Demaskierung ergibt sich nach Gleichung (3.20) zu

$$\begin{aligned} K_{\text{Wechselseitig}} &= G(N) O(N^3) + O\left(\frac{N^3}{\log N^4}\right) \\ &= G(N) O(N^3) \end{aligned}$$

Bei allen durchgeführten Tests war die Laufzeit der wechselseitigen Demaskierung jedoch nicht relevant gemessen an der Gesamtlaufzeit: Da $p < \frac{1}{8}$ nicht überschritten wurde folgte für $K_{\text{Punkteexklusion}}: s(N) \lll G O(N)$. Außerdem muss die Abschätzung der detektierbaren Liniensegmente auf einem Rand-zu-Rand Segment mit $n(N)$ als grobe obere Schranke angesehen werden.

3.5.2 Verbesserungsansätze

Identifizierung der Ansatzpunkte. Wie im vorherigen Abschnitt 3.5.1 gezeigt wurde, ist die Durchmusterung des Suchraums der aufwändigste Teil bei der Detektion von Liniensegmenten. Dies deckt sich mit den Erfahrungen der durchgeführten Versuche aus Kapitel 4. Dieser Abschnitt zeigt zwei Ansatzpunkte auf, wie der Aufwand besser bewältigt werden kann.

Einschränkung des Suchraums. Die in Abschnitt 3.3.1 vorgestellten Rand-zu-Rand Segmente haben den Nachteil, den Suchraum teilweise mehrfach zu durchmusteren: Dieselben Bildpunkte sind häufig Linienpunkte verschiedener Rand-zu-Rand Segmente. Das ist mit der Gliederung durch Rand-zu-Rand Segmente wegen der zugrunde liegenden diskreten Geometrie nie ganz vermeidbar. Trotzdem kann eine Untersuchung, wie häufig die Abtastung gleicher Bildpunkte in Abhängigkeit der Rand-zu-Rand Segmente vorkommt, die Durchmusterung mit weniger Rand-zu-Rand Segmenten plausibilisieren.

Dabei ist eine Quantisierung des Winkels zwischen zwei Rand-zu-Rand Segmenten, wie sie von Desolneux et al. in [DMM00] vorgeschlagen wird, nicht ratsam: Neben einem zusätzlichen Parameter ist diese Herangehensweise von der Bilddimension abhängig (siehe Abbildung 3.5).

Da in dieser Arbeit die Zuverlässigkeit der Detektion beurteilt werden soll, wird im weiteren von der Betrachtung einer solchen Einschränkung des Suchraums abgesehen. Damit ist gewährleistet, dass der Suchraum mit der beschriebenen Durchmusterung durch Rand-zu-Rand Segmente bestmöglich abgedeckt wird.

Parallelisierung. Der gezeigte Algorithmus bietet für die Durchmusterung des Suchraums ein großes Potential an Parallelisierung: Die Berechnung auf den verschiedenen $O(N^2)$ Rand-zu-Rand Segmenten mit Algorithmus III ist voneinander nicht abhängig. Die Bestimmung der Linienpunkte durch Algorithmus I, die Detektion und direkte Demaskierung durch Algorithmus III könnte daher ohne großen Aufwand auf mehrere Rechner verteilt werden.

Nach einmaligem Bereitstellen des Ausgangsbildes für jeden Rechner besteht der Kommunikationsaufwand für den Algorithmus am Beginn paralleler Ausführung lediglich in der Übermittlung von Start- und Endpunkten der Rand-zu-Rand Segmente. Am Ende der parallelen Ausführung müssen die Start- und Endpunkte der direkt demaskierten Liniensegmente wieder zusammengeführt werden.

Der Schwerpunkt dieser Arbeit ist jedoch nicht Performance-Maximierung, sondern die Beurteilung des Helmholtz-Prinzips am Beispiel des perzeptuellen Primitivs “Liniensegment”. Aus diesem Grund wird die Parallelisierbarkeit hier nicht weiter untersucht.

4 Testergebnisse

4.1 Visualisierung der Ergebnisse und Ausgangsbilder

In den folgenden Abschnitten werden die Testergebnisse präsentiert und interpretiert, die mit dem in Abschnitt 2 beschriebenen Algorithmus nach dem Helmholtz-Prinzip für Liniensegmente entstanden sind. Dieser Abschnitt erläutert, wie die Ergebnisse dargestellt werden und wie die Ausgangsbilder zustande kommen.

Farbige Darstellung des Maskierungsproblems. Um die Notwendigkeit direkter und wechselseitiger Demaskierung aufzuzeigen und Bildpunkte zu kennzeichnen, die besonders häufig Liniensegmente vieler verschiedener Liniensegmente sind, werden die Ergebnisse farbig dargestellt. Abbildung 4.1 veranschaulicht die Farbskala.



Abbildung 4.1: Farbzurordnung eines Liniensegmentes in Abhängigkeit der Anzahl Liniensegmente, die ihn beinhalten: Ein Bildpunkt wird pro Liniensegment, das ihn beinhaltet, erst schwarz (Liniensegment eines Liniensegments), dann rot (Liniensegment von 256 Liniensegmenten), dann violett (Liniensegment von 512 Liniensegmenten) und schließlich gelb (Liniensegment von mehr als 512 Liniensegmenten) gefärbt.

Eingangsdaten. Als Eingangsdaten kommen nicht immer Grauwertbilder zum Einsatz. Wenn die Originalbilder farbig sind, werden sie in einem Vorverarbeitungsschritt zu Grauwertbildern konvertiert. Durchgeführt wird diese Konvertierung zunächst durch eine Farbraumtransformation $RGB \rightarrow YUV$ für jeden Bildpunkt mit

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ -0,169 & -0,331 & 0,500 \\ 0,500 & -0,419 & -0,081 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Anschließend wird nur noch der Helligkeitswert Y als Grauwert betrachtet. Dass die Helligkeitsunterschiede in einem Bild wesentlich für die menschliche Wahrnehmung ist, wird beispielsweise in [LR95] erläutert.

Aus diesem Grund ist zu jedem Testergebnis das Grauwertbild gezeigt. Dieses ist das Ausgangsbild für den Detektionsalgorithmus.

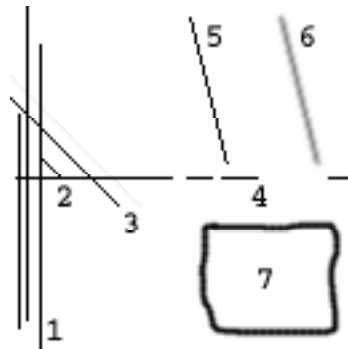
4.2 Demonstration an einem synthetischen Bild

In diesem Abschnitt wird das Detektionsergebnis auf einem mit *The GIMP* [GIMP] erzeugten Testbild (Abbildung 4.2(a)) präsentiert. Das Testbild enthält dabei einige nummerierte Linienstrukturen. Diese wurden erzeugt, um die Funktionsweise des Algorithmus zu verifizieren und zu prüfen, ob sich die theoretisch erwarteten Ergebnisse einstellen. Die Parameter für den Algorithmus sind $p = \frac{1}{16}$ und $\epsilon = 1$.

Detektion ohne Demaskierung. Abbildung 4.2(b) ist das Ergebnis einer Detektion ohne Demaskierung. Man erkennt an der großen Anzahl erkannter Liniensegmente, dass die Detektion ohne Demaskierung nur einen groben Eindruck vermitteln kann. Das spiegelt sich an der gelben Färbung wider: Alle Bildpunkte mit gelber Farbe sind Bestandteile von mehr als 512 Liniensegmenten. Aus diesem Grund wird bei den nachfolgenden Testsequenzen auf die Abbildung der detektierten Liniensegmente ohne Demaskierung verzichtet.

Detektion und direkte Demaskierung. Abbildung 4.2(c) zeigt das Ergebnis der gefundenen Liniensegmente, welche die Eigenschaften 1 und 2 aus Abschnitt 3.4.1 erfüllen: Ihre Start- und Endpunkt entsprechen der Richtung des Liniensegments, die Punkte davor und danach entsprechen nicht der Richtung des Liniensegments (im Folgenden *Endpunkteigenschaft* genannt). Abbildung 4.2(d) zeigt die Liniensegmente nach vollständiger direkter Demaskierung (nur noch Liniensegmente mit minimaler Auftrittswahrscheinlichkeit auf einem Rand-zu-Rand Segment, siehe Abschnitt 3.4.1).

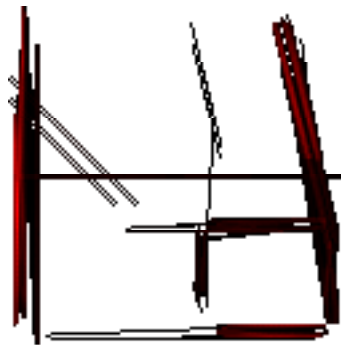
An der Anzahl der Liniensegmente im Vergleich zur Anzahl aus Abbildung 4.2(b) erkennt man, dass der erste Demaskierungsschritt über Endpunkteigenschaft in Abbildung 4.2(c) einen wesentlichen Beitrag zur direkten Demaskierung leistet. Allerdings ist auch der Unterschied zur vollständigen direkten Demaskierung in Abbildung 4.2(d) zu sehen (siehe z.B. Erläuterung zu Linienstruktur 4).



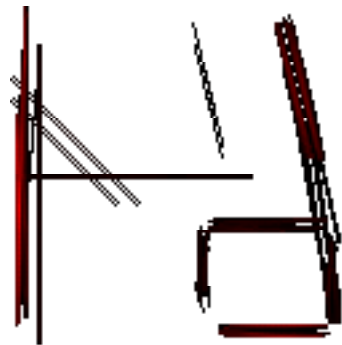
(a) Ausgangsbild



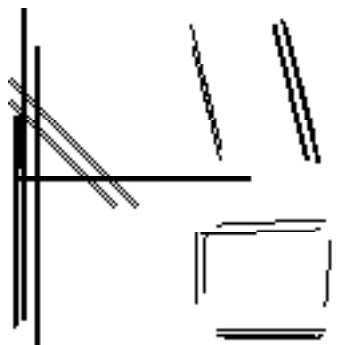
(b) Detektion, 443001 Liniensegmente.



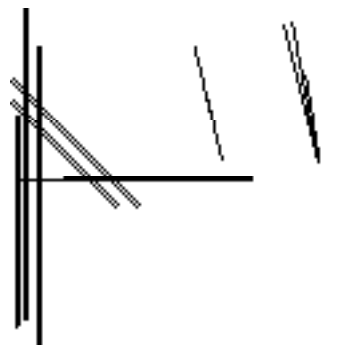
(c) Detektion mit Endpunkteigenschaft, 3416 Liniensegmente.



(d) Direkte Demaskierung, 3069 Liniensegmente.



(e) Wechselseitige Demaskierung über Wahrscheinlichkeit, 41 Liniensegmente.



(f) Wechselseitige Demaskierung über Präzisionsanpassung, 27 Liniensegmente.

Abbildung 4.2: Demonstrationsbild (128×128). $p = \frac{1}{16}$, $\epsilon = 1$.

Detektion nach zusätzlicher wechselseitiger Maskierung. Abbildung 4.2(e) und Abbildung 4.2(f) zeigen detektierte Liniensegmente, die zuerst direkt und danach wechselseitig demaskiert wurden (siehe Abschnitt 3.4.2). Es ist offensichtlich, dass das Ergebnis nach wechselseitiger Demaskierung verglichen mit dem Ergebnis nach direkter Demaskierung die originalen Linienstrukturen besser widerspiegelt.

Während in Abbildung 4.2(e) die wechselseitige Demaskierung über die Zuordnung nach minimaler Wahrscheinlichkeit durchgeführt wurde, ist Abbildung 4.2(f) über nachträgliche Präzisionsanpassung demaskiert worden (siehe Abschnitt 3.4.2). Die Unterschiede sind an den Linienstrukturen 1 und 7 besonders deutlich.

In den folgenden Testsequenzen wird nur noch das Ergebnis der wechselseitigen Demaskierung nach Zuordnung über minimale Wahrscheinlichkeit gezeigt, da dieses Zuordnungskriterium in unabhängiger von der Bildqualität des Ausgangsbildes ist (siehe Abschnitt 3.4.2).

Erläuterungen zu den nummerierten Linienstrukturen. Es folgen korrespondierend zu den Nummerierungen im Ausgangsbild (Abbildung 4.2(a)) die Interpretationen der Testergebnisse.

1. 3 vertikal verlaufende Parallelen unterschiedlicher Länge. Sie zeigen das Problem wechselseitiger Maskierung: Erst nachdem eine wechselseitige Demaskierung in Abbildung 4.2(e) und Abbildung 4.2(f) durchgeführt wurde, kann man sie voneinander unterscheiden. Man erkennt, dass in diesem Fall die wechselseitige Demaskierung über nachträgliche Präzisionsanpassung bessere Ergebnisse liefern kann: Während bei der Demaskierung über minimale Wahrscheinlichkeit die beiden linken Liniensegmente nicht eindeutig voneinander separiert werden, gelingt das mit nachträglicher Präzisionsanpassung. Voraussetzung ist, dass wie hier die Richtung der Linienpunkte annähernd mit der der Linienstruktur übereinstimmt (z.B. ist eine Bedingung dafür ein rauscharmes Ausgangsbild).
2. In Relation zur Bilddimension ist diese Linienstruktur zu kurz, um erkannt zu werden. Das ist mit Gleichung (2.12) aus Abschnitt 2.5.2 zu erklären.
3. Zwei diagonale Linienstrukturen mit unterschiedlichem Kontrast: Während die untere der beiden starke Kontraste an ihren Kanten aufweist, ist die obere im Originalbild (Abbildung 4.2(a)) kaum wahrzunehmen. Trotzdem werden beide Liniensegmente detektiert, da die Berechnung der Richtung an Linienpunkten kontrastinvariant ist (siehe Abschnitt 3.1.4). Man sieht, dass ein Liniensegment an beiden Seiten der Linienstruktur erkannt wird, da auf beiden Seiten ein Kontrastunterschied und damit ein Gradient mit Betrag größer 0 vorliegt.

4. Eine unterbrochene horizontale Linie. Man erkennt zwischen Abbildung 4.2(c) und Abbildung 4.2(d) den Unterschied zwischen der direkten Demaskierung über die Endpunkteigenschaft und der vollständigen direkten Demaskierung auf einem Rand-zu-Rand Segment: Während in Abbildung 4.2(c) ein horizontales Liniensegment über das gesamte Bild hinweg erkannt wird, zeigt Abbildung 4.2(d), dass ein Auftreten des Liniensegments ohne das weiter entfernte Teilstück rechts unwahrscheinlicher ist, als ein Auftreten des durchgezogenen Liniensegments.
5. Eine diagonale Linie als Stufenkante gezeichnet. Man erkennt an der schwachen Detektion die Nachteile des verwendeten Gradienten (siehe Abschnitt 3.1.4): Ihm zufolge ist die Richtung der meisten Linienpunkte dieses Liniensegments $\pm \frac{\pi}{2}$ (also genau senkrecht), während die Linienstruktur in eine andere Richtung verläuft.
6. Die gleiche Linie aus Punkt 5 nach Anwendung eines Weichzeichners. Zwar sind die Stufenkanten noch immer zu erkennen, aber die Richtung der Linienpunkte verläuft gemäß dem Gradienten auf der 2×2 -Nachbarschaft schon mehr in Richtung des Liniensegments. In diesem Zusammenhang ist der Nachteil bei wechselseitiger Demaskierung nach Präzisionsanpassung zu sehen: Da die Richtung der Linienpunkte nicht genau genug entlang des Liniensegments verläuft, kommt es zu Detektionen von Liniensegmenten, die nicht der Richtung der Linienstruktur entsprechen. Außerdem erkennt man an der “Breite” der gefundenen Liniensegmente, dass auf jeder Seite mehrere Liniensegmente detektiert wurden. Das ist auf den Verlauf der Graustufen zurückzuführen, die zueinander Helligkeitsübergänge bilden (siehe Abschnitt 4.6).
7. Der Versuch, ein Rechteck freihändig mit einem Pinselwerkzeug zu zeichnen. Das Pinselwerkzeug setzt einzelne unscharfe Punkte nebeneinander und erzeugt so “gezackte” Kanten. Der Unterschied zwischen den beiden wechselseitigen Demaskierungsmethoden wird in Abbildung 4.2(e) und Abbildung 4.2(f) am deutlichsten: Während das Kriterium minimaler Wahrscheinlichkeit die wesentlichen Liniensegmente übriglässt, wird nach der Präzisionsanpassung kein Liniensegment des Rechtecks mehr erkannt: Die Linienpunkte entlang der Rechteckseiten werden bei Präzisionsanpassung verschiedenen Liniensegmenten entlang der Seiten zugeordnet. Keinem einzelnen Liniensegment werden aber so viele Linienpunkte exklusiv zugeordnet, dass es noch detektiert wird.

4.3 Wahl der Parameter

In diesem Abschnitt wird der Einfluss der beiden Parameter p (Abschnitt 4.3.1) und ϵ (Abschnitt 4.3.2) erläutert. Zu diesem Zweck sind die Testsequenzen in Abbildung 4.3 für den Einfluss von p und in Abbildung 4.4 für die Wirkungsweise von ϵ gezeigt.

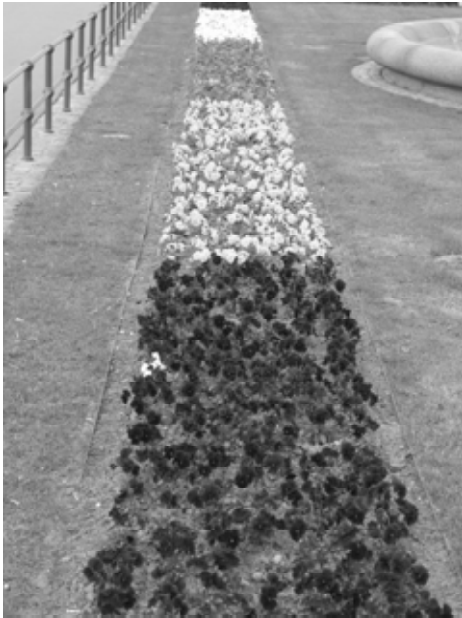
4.3.1 Präzisionsparameter p

Wie in Abschnitt 3.1.5 erläutert, lässt sich p als Präzision interpretieren. An den Testergebnissen des Ausgangsbildes in Abbildung 4.3(a) lässt sich diese Interpretation verifizieren. Nach wechselseitiger Demaskierung ergibt sich für $p = \frac{1}{8}$ Abbildung 4.3(b), für $p = \frac{1}{16}$ Abbildung 4.3(c) und für $p = \frac{1}{32}$ Abbildung 4.3(d). Der Einfluss des Parameters p ist in Abbildung 4.3 an drei Stellen zu erkennen:

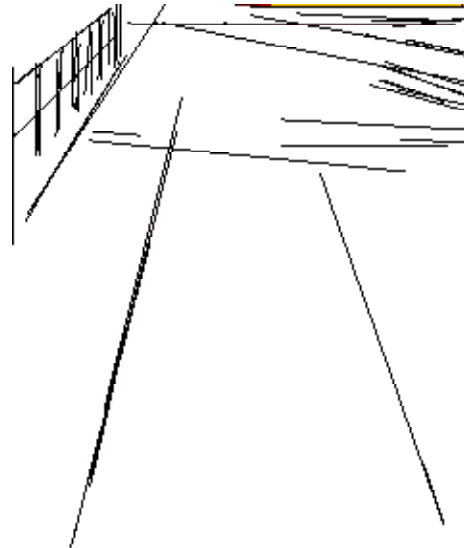
1. **Geländer links oben im Bild:** Das Geländer ergibt wegen seiner klaren Struktur sehr stabile Liniensegmente. Seine Bestandteile werden mit jedem gewählten Wert für p gut erkannt.
2. **Brunnenausschnitt und Rasenfläche rechts oben im Bild:** Während der Brunnenausschnitt rechts oben signifikante Liniensegmente aufweist (diese werden für jeden Wert von p detektiert), werden in der umgebenden Rasenfläche vor allem für $p = \frac{1}{8}$ Liniensegmente erkannt, weil sie durch die perspektivische Verzerrung als horizontale Linienstrukturen erscheinen. Mit $p = \frac{1}{32}$ kann dieser Effekt weitgehend eingeschränkt werden. Für $p = \frac{1}{16}$ sind die meisten gefundenen Liniensegmenten in diesem Gebiet ebenfalls die Linienstrukturen des Brunnens.
3. **Begrenzungssteine des Beetes unten im Bild:** Nicht alle Linienpunkte dieser Linienstruktur haben eine zu ihr annähernd gleiche Richtung, da die sie im Ausgangsbild teilweise von Rasen überdeckt wird. Mit $p = \frac{1}{8}$ werden sie trotzdem gut erkannt, mit $p = \frac{1}{32}$ gar nicht mehr. Für $p = \frac{1}{16}$ werden die korrespondierenden Liniensegmente noch ansatzweise detektiert.

Die Testsequenz aus Abbildung 4.3 zeigt den Konflikt bei der Wahl des Parameters p : Ist p zu klein (d.h. die Präzision zu hoch), dann werden vorhandene aber undeutliche Linienstrukturen (siehe Punkt 3) nicht mehr erkannt. Ist p zu groß (und damit die Präzision zu niedrig), dann kommt es vermehrt (hier z.B. durch perspektivische Verzerrung in Punkt 2) zu Fehldetektionen.

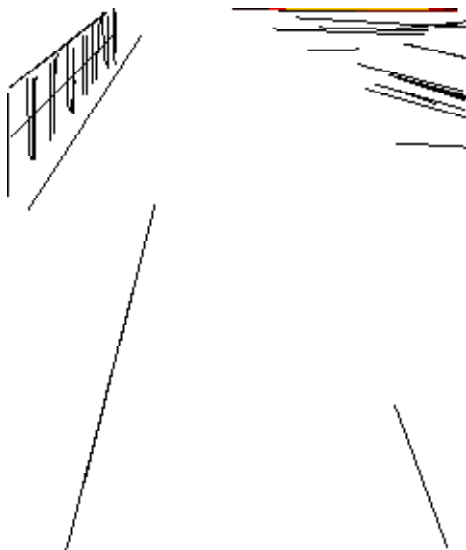
Für Liniensegmente, wie sie in dieser Arbeit vorgestellt werden, ist $p = \frac{1}{16}$ eine gute Wahl. Dieser Wert wird auch von Desolneux et al. [DMM00] vorgeschlagen und wird in den folgenden Sequenzen verwendet.



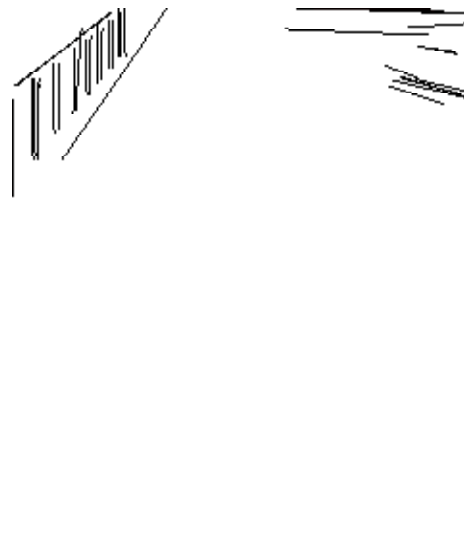
(a) Ausgangsbild



(b) Wechselseitige Demaskierung über Wahrscheinlichkeit mit $p = \frac{1}{8}$, 686 Liniensegmente.



(c) Wechselseitige Demaskierung über Wahrscheinlichkeit mit $p = \frac{1}{16}$, 455 Liniensegmente.



(d) Wechselseitige Demaskierung über Wahrscheinlichkeit mit $p = \frac{1}{32}$, 42 Liniensegmente.

Abbildung 4.3: Beet mit Stiefmütterchen (320×240) für verschiedene Werte von p , $\epsilon = 1$.

4.3.2 Grenzwert ϵ des Erwartungswerts

Der Parameter ϵ kann als die Anzahl der detektierten Liniensegmente verstanden werden, falls die Richtung an allen Bildpunkten tatsächlich zufällig gleichverteilt ist (siehe Abschnitt 2.4.2).

Damit lässt sich $\frac{\epsilon}{N^2(N^2-1)}$ als Falschalarmrate für Liniensegmente verstehen: Wie aus Gleichung (2.11) in Abschnitt 2.5.2 hervorgeht, wird für eine Falschalarmrate von 0 (d.h. $\epsilon = 0$) kein Liniensegment und für Falschalarmrate 1 (d.h. $\epsilon = N^2(N^2 - 1)$) jedes mögliche Liniensegment detektiert.

An der Testsequenz in Abbildung 4.4 ist durch die Größenordnung der Veränderung von ϵ zu sehen, dass ϵ lediglich logarithmischen Einfluss auf die Detektionsfunktion hat (siehe Gleichung (2.24) aus Abschnitt 2.5.4).

Das Ergebnis auf Abbildung 4.4(a) nach wechselseitiger Demaskierung ist für $\epsilon = 10^{-8}$ in Abbildung 4.4(b), $\epsilon = 1$ in Abbildung 4.4(c) und $\epsilon = 10^6$ in Abbildung 4.4(d) gezeigt.

Veränderungen durch die Wahl von ϵ

- **Minimale Länge der detektierten Liniensegmente:** Während für $\epsilon = 10^{-8}$ nur längere Linienstrukturen gefunden werden, ergeben sich für $\epsilon = 1$ auch kürzere Liniensegmente. Für $\epsilon = 10^6$ werden die meisten der kurzen Winkelgänge des Schlosses und sogar die längeren Bestandteile der Ziffern 1 und 7 detektiert. Dieser Effekt ist nach Gleichung (2.12) keine Überraschung.
- **Detektion runder Strukturen:** Für $\epsilon = 10^6$ werden die runden Strukturen bei Ziffer 17 als einzelne kurze Liniensegmente detektiert. Das ist auf die minimal nötige Länge als Voraussetzung für ein Liniensegment zurückzuführen: Die Rundungen können als zusammengesetzte Linien sehr geringer Länge gesehen werden.
- **Nicht durchgezogene Linienstrukturen:** Die Umrisse des Westflügels (Ziffern 12, 13, 14; in Abbildung 4.4 ist Norden unten) werden erst für $\epsilon = 10^6$ gefunden. Die Linienstruktur besteht dort großteils aus einzelnen Punkten. Wie in Abschnitt 4.5 zu sehen ist, lässt sich die Schwierigkeit bei der Detektion dieser gepunkteten Linienstruktur auf die Qualität des Gradienten zurückführen. ϵ ist hierfür nicht ausschlaggebend.

Robustheit. Trotz der beobachteten Veränderungen durch die Wahl von ϵ ist festzustellen, dass die Detektion wesentlicher Linienstrukturen gegenüber einer Verände-

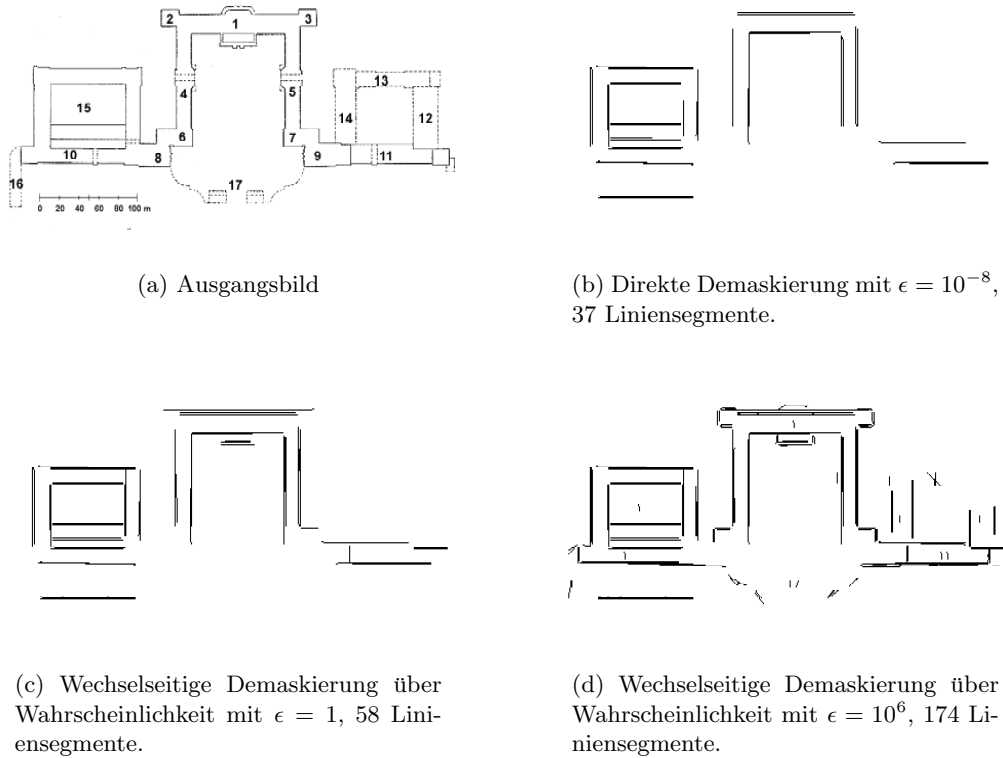


Abbildung 4.4: Grundriss Mannheimer Schloss (242×450) für verschiedene Werte von ϵ , $p = \frac{1}{16}$.

Die Wahl von ϵ robust ist: Lange und eindeutige (in diesem Fall: durchgezogene) Liniensegmente werden für jeden Wert von ϵ detektiert.

Wahl von ϵ . Als Folgerung der Beobachtungen ist $\epsilon \simeq 1$ für die meisten Bilder ein guter Wert. Wird ϵ zu niedrig gewählt, werden einige vorhandene Linienstrukturen nicht mehr gefunden. Ist ϵ zu groß (vgl. Abbildung 4.4(d)), dann werden zum einen Liniensegmente erkannt, die möglicherweise keine Linienstrukturen sind (Rundungen bei Ziffer 17). Zum anderen wird die Minimallänge eines Liniensegments klein und die Gefahr besteht, linienartige Texturen zu detektieren (die Ziffern 1 und 7 werden als Liniensegmente erkannt).

In den folgenden Testsequenzen wird daher $\epsilon = 1$ gesetzt.

4.4 Einfluss von Rauschen

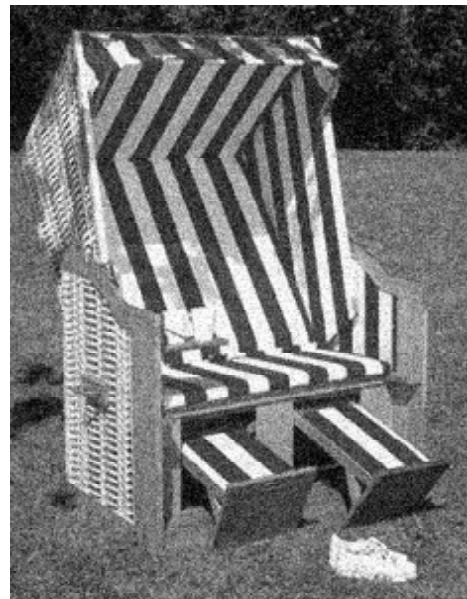
Um das Verhalten der Detektion auf verrauschten Eingangsbildern zu prüfen, wurde auf das Ausgangsbild in Abbildung 4.5(a) mit dem Grafikprogramm *The GIMP* [GIMP] ein Filter angewendet, der weißes Rauschen erzeugt. Abbildung 4.5(b) zeigt das Resultat. Anschließend wurden beide Bilder getestet.

Bewertung: Wie der Vergleich des Ergebnisses nach direkter Demaskierung in Abbildung 4.6(a) zeigt, werden weniger Liniensegmente gefunden: Einige der in Abschnitt 3.4.1 beschriebenen wechselseitigen Maskierungen bleiben aus. Die Ergebnisse nach wechselseitiger Demaskierung in Abbildung 4.6(b) zeigen durch die Anzahl detektierter Liniensegmente, dass die Detektion der wesentlichen Linienstrukturen stabil bleibt.

Probleme: An der Detektion der Fußstützen erkennt man, dass kürzere Liniensegmente problematisch sind (das Verhältnis zur Anzahl der verrauschten Linienpunkte ist kleiner). Dasselbe gilt für unterbrochene Liniensegmente (in der Testsequenz die hintere Kante des Sessels). Abschnitt 4.5 zeigt, dass der verwendete Gradient Einfluss auf die Detektion bei verrauschten Ausgangsbildern hat.

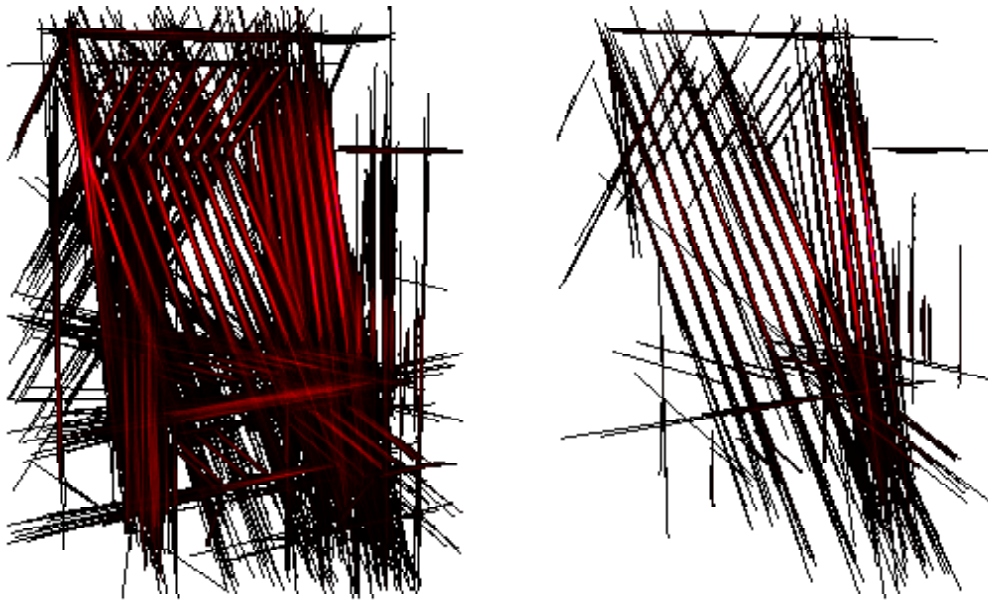


(a) Ausgangsbild

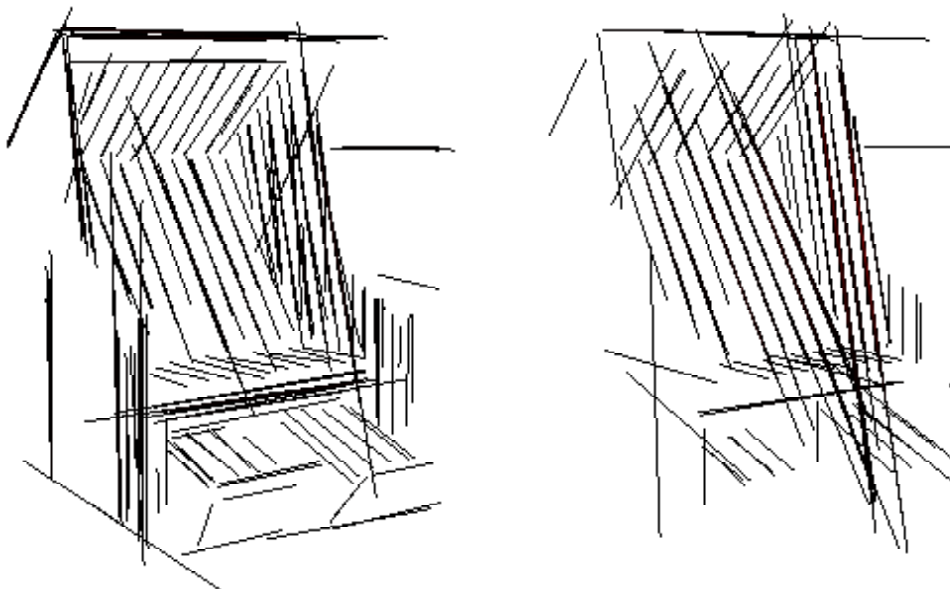


(b) verrauschtes Ausgangsbild

Abbildung 4.5: Ein Korbsessel (300×233). Links unverrauscht, rechts mit Rauschen.



(a) Direkte Demaskierung. Links: Ohne Rauschen, 23235 Liniensegmente. Rechts: Mit Rauschen, 5128 Liniensegmente.



(b) Wechselseitige Demaskierung über Wahrscheinlichkeit. Links: Ohne Rauschen, 628 Liniensegmente. Rechts: Mit Rauschen, 503 Liniensegmente.

Abbildung 4.6: Ergebnisse eines Korbsessels (300×233). Links ohne, rechts mit Rauschen nach direkter und wechselseitiger Demaskierung. $p = \frac{1}{16}$, $\epsilon = \frac{1}{73}$

4.5 Auswirkungen des Gradienten

In diesem Abschnitt werden die in Abschnitt 3.1.4 erwähnten Nachteile des verwendeten Gradienten (Binomialfilter erster Ordnung als Hochpass) demonstriert. Als Vergleich kommt der nach Scharr [SKJ97] optimierte Sobel-Operator zum Einsatz. Abbildung 4.7 zeigt vergleichend die Filtermasken für die Gradienten.

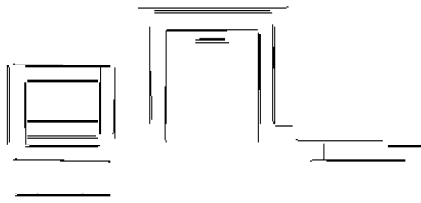
<table border="1" style="border-collapse: collapse; width: 40px; height: 40px; margin: auto;"> <tr><td style="padding: 2px;">-1</td><td style="padding: 2px;">-1</td></tr> <tr><td style="padding: 2px;">1</td><td style="padding: 2px;">1</td></tr> </table>	-1	-1	1	1	<table border="1" style="border-collapse: collapse; width: 40px; height: 40px; margin: auto;"> <tr><td style="padding: 2px;">-1</td><td style="padding: 2px;">1</td></tr> <tr><td style="padding: 2px;">-1</td><td style="padding: 2px;">1</td></tr> </table>	-1	1	-1	1	<table border="1" style="border-collapse: collapse; width: 60px; height: 60px; margin: auto;"> <tr><td style="padding: 2px;">-3</td><td style="padding: 2px;">-10</td><td style="padding: 2px;">-3</td></tr> <tr><td style="padding: 2px;">0</td><td style="padding: 2px;">0</td><td style="padding: 2px;">0</td></tr> <tr><td style="padding: 2px;">3</td><td style="padding: 2px;">10</td><td style="padding: 2px;">3</td></tr> </table>	-3	-10	-3	0	0	0	3	10	3	<table border="1" style="border-collapse: collapse; width: 60px; height: 60px; margin: auto;"> <tr><td style="padding: 2px;">-3</td><td style="padding: 2px;">0</td><td style="padding: 2px;">3</td></tr> <tr><td style="padding: 2px;">-10</td><td style="padding: 2px;">0</td><td style="padding: 2px;">10</td></tr> <tr><td style="padding: 2px;">-3</td><td style="padding: 2px;">0</td><td style="padding: 2px;">3</td></tr> </table>	-3	0	3	-10	0	10	-3	0	3
-1	-1																												
1	1																												
-1	1																												
-1	1																												
-3	-10	-3																											
0	0	0																											
3	10	3																											
-3	0	3																											
-10	0	10																											
-3	0	3																											
(a) Binomialoperator		(b) Sobel-Operator nach Scharr																											

Abbildung 4.7: Filtermasken für die der Richtungsberechnung zugrunde liegenden Gradienten. Jeweils links: Horizontale Filtermaske. Jeweils Rechts: Vertikale Filtermaske.

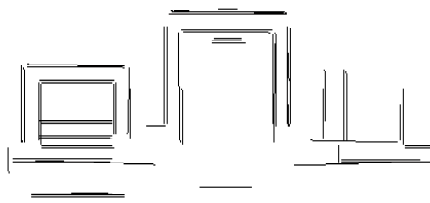
Vergleichbare Anwendung. Um die Ergebnisse vergleichen zu können, ist es notwendig, die Grundannahme statistischer Unabhängigkeit der Linienpunkte zu verletzen: Für beide Gradienten wurde die Richtung jedes zweiten Linienpunktes betrachtet, obwohl das für die Richtungsberechnung nach dem Sobel-Operator eine schwache statistische Abhängigkeit bedeutet (bis zu 3 von den 9 zur Richtungsberechnung verwendeten Helligkeitswerte nachfolgender Linienpunkte sind dieselben).

Für die beiden betrachteten Beispiele in Abbildung 4.4(a) und Abbildung 4.5(a) scheint das nicht von Nachteil zu sein; jedoch fehlt eine fundierte theoretische Untersuchung für den Fall statistischer Abhängigkeit (siehe Abschnitt 5). Beide Testsequenzen wurden mit $p = \frac{1}{16}$, $\epsilon = 1$ durchgeführt.

Auswertung. In beiden Fällen kann durch den Einsatz des Sobel-Operators eine Verbesserung der Detektion erreicht werden. Das ist vor allem auf die bessere Nutzung der lokalen Kontrastinformation zurückzuführen (es werden 9 anstatt nur 4 Helligkeitswerte berücksichtigt). Mit dem zum Mittelpunkt symmetrischen Sobel-Operator kann außerdem die in Abschnitt 3.1.4 beschriebene Verschiebung des Bildrasters um ein halbes Subpixel vermieden werden.

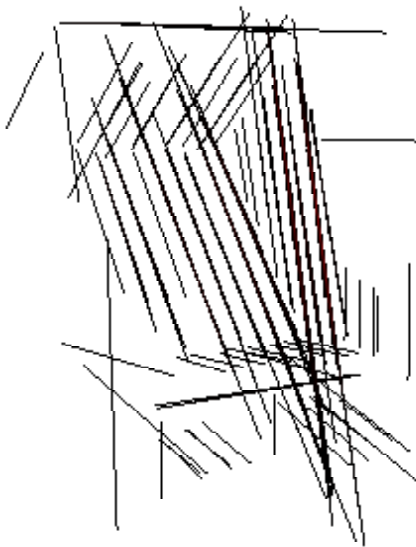


(a) Ergebnis mit 2×2 Binomialoperator, 58 Liniensegmente.

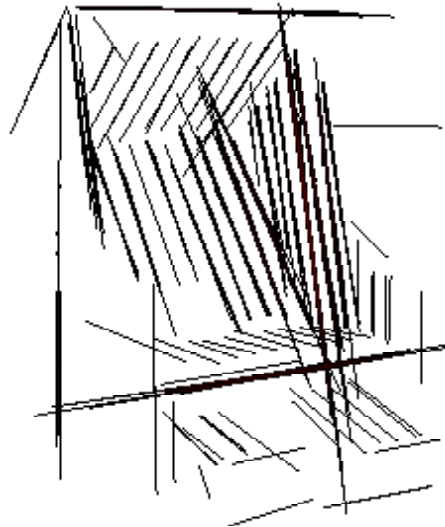


(b) Ergebnis mit 3×3 Sobel-Operator, 75 Liniensegmente.

Abbildung 4.8: Mit dem Sobel-Operator werden nicht durchgezogene Linienstrukturen erkannt (rechte Seite). Mit dem Binomialoperator ist das nur mit höherer Falschalarmrate möglich (Abbildung 4.4(d)). $p = \frac{1}{16}$, $\epsilon = 1$.



(a) Ergebnis mit 2×2 Binomialoperator, 503 Liniensegmente.



(b) Ergebnis mit 3×3 Sobel-Operator, 626 Liniensegmente.

Abbildung 4.9: Mit dem Sobel-Operator wird die Detektion weniger rauschempfindlich (Abschnitt 4.4). Das ist an den kürzeren und nicht durchgezogenen Linienstrukturen zu erkennen (siehe Abbildung 4.6). $p = \frac{1}{16}$, $\epsilon = 1$.

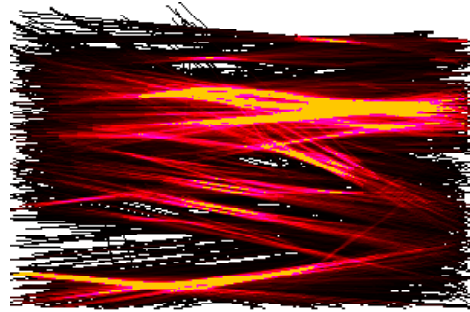
4.6 Problem quasi-paralleler Texturen

In diesem Abschnitt wird die in Abschnitt 3.4.2 schon als problematisch eingestufte Situation annähernd paralleler Linienstrukturen demonstriert. Dieses Problem kann mit Demaskierung nicht gelöst werden. Es gibt viele Beispiele, in denen tatsächlich so viele nah beieinander liegende parallele Linienstrukturen existieren, so dass eine Demaskierung nicht funktionieren kann. Das Paradebeispiel dafür sind Farb- oder Helligkeitsverläufe.

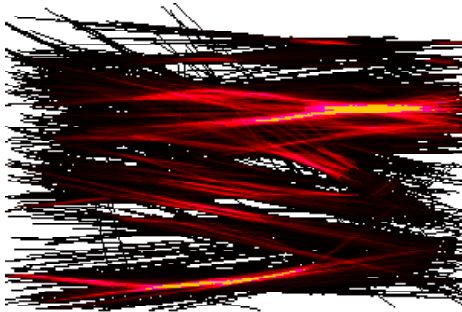
Zur Demonstration sind mit Abbildung 4.10, Abbildung 4.11 und Abbildung 4.12 drei Testsequenzen gezeigt.



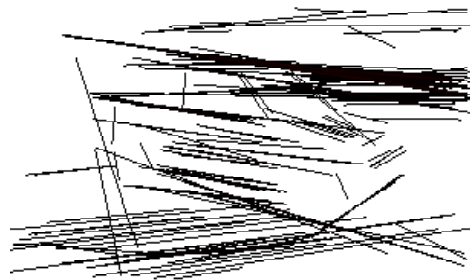
(a) Ausgangsbild



(b) Detektion mit Endpunkteigenschaft, 60585 Liniensegmente.



(c) Direkte Demaskierung, 26070 Liniensegmente.



(d) Wechselseitige Demaskierung über Wahrscheinlichkeit, 343 Liniensegmente.

Abbildung 4.10: Autos auf einem Parkplatz (202×302). $p = \frac{1}{16}$, $\epsilon = 1$.

4.6.1 Parallelität durch Perspektive

Bei der Projektion der dreidimensionalen Umwelt auf ein zweidimensionales Bild kommt es zu einer Verschiebung der Größenverhältnisse visueller Objekte. Das trifft natürlich auch für Linienstrukturen zu.

Die Testsequenz in Abbildung 4.10 zeigt das entstehende Problem: Das Kopfsteinpflaster, auf dem das Auto steht, reicht bis zum Fotografen. Da die Steine regelmäßig verlegt sind, erscheinen die Lücken zwischen ihnen als unterbrochene Linienstrukturen. Durch ihre räumliche Nähe zur Bildebene sind diese Linienstrukturen in der zweidimensionalen Aufnahme verhältnismäßig lang und stark ausgeprägt. Sie werden als parallele Liniensegmente erkannt.

Ein weiteres Beispiel ist die perspektivische Verzerrung des Schattens eines außerhalb des Aufnahmebereichs stehenden Baumes rechts im Bild: Dieser Schatten erzeugt gerade Liniensegmente. Außerdem bildet der Schatten einen Helligkeitsverlauf, was zu zusätzlichen parallelen Strukturen führt (siehe Abschnitt 4.6.3).

Anhand der Anzahl der Liniensegmente und der anfangs gelben Färbung ist an den einzelnen Detektions- und Demaskierungsstufen zu erkennen, dass viele der Liniensegmente durch die wechselseitige Demaskierung herausgefiltert werden können. Am Beispiel des Kopfsteinpflasters wird aber deutlich, dass durch Demaskierung alleine diese Liniensegmente nie ungültig werden können: Sie sind tatsächlich vorhanden.

4.6.2 Parallele Linientexturen

Viele Objekte, die dem Menschen als einheitliche Fläche erscheinen, bestehen aus nah beieinander liegenden parallelen Linienstrukturen. Das Sonnengitter in Abbildung 4.11 ist ein Beispiel für diesen Sachverhalt: Für den menschlichen Betrachter wirkt das Sonnengitter in Abbildung 4.11(a) wie eine einheitliche Fläche. Das Ergebnis nach wechselseitiger Demaskierung in Abbildung 4.11(d) zeigt aber, dass es in Wirklichkeit aus vielen einzelnen Lamellen besteht, die als Liniensegmente detektiert werden.

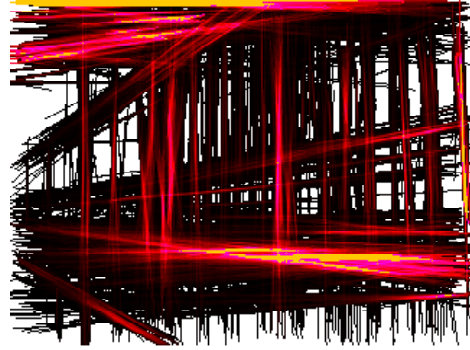
Solche Liniensegmente zu gruppieren, muss Aufgabe eines komplexeren Algorithmus sein. Das in Abschnitt 2.3 beschriebene Helmholtz-Prinzip kann zu diesem Zweck “wiederverwendet” werden (siehe Abschnitt 5).

4.6.3 Parallelität an Farbverläufen

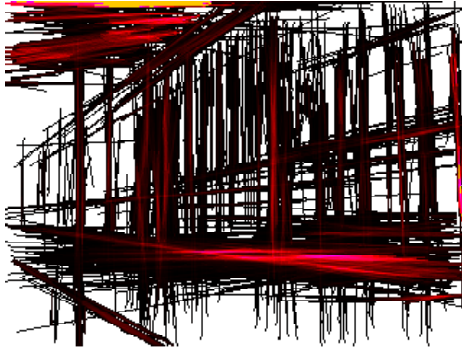
Besonders deutlich wird die Auswirkung von Helligkeitsverläufen in Abbildung 4.12: Der Himmel bildet einen gleichmäßigen Verlauf. Aufgrund der Kontrastinvarianz



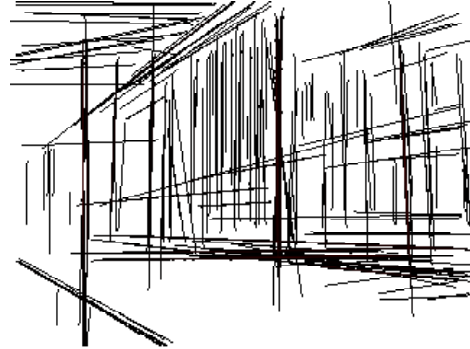
(a) Ausgangsbild



(b) Detektion mit Endpunkteigenschaft, 62363 Liniensegmente.



(c) Direkte Demaskierung, 28501 Liniensegmente.



(d) Wechselseitige Demaskierung über Wahrscheinlichkeit, 658 Liniensegmente.

Abbildung 4.11: Gebäude der Technischen Informatik (240×320). $p = \frac{1}{16}$, $\epsilon = 1$.

des Verfahrens (siehe Abschnitt 3.1.4) werden daher nebeneinander liegende (näherungsweise parallele) Liniensegmente detektiert. Die große Anzahl gefundener Liniensegmente vor wechselseitiger Demaskierung und die gelbe Färbung im Bereich des Himmels der Testsequenz in Abbildung 4.12 ist darauf zurückzuführen. Das gleiche Phänomen ist in dieser Testsequenz an dem Weg im Vordergrund zu sehen: Er ist nicht konstant grau, sondern erscheint durch den Lichteinfall als Graustufenverlauf. Das korrespondiert mit dem Verlauf im Himmel: Dieser entsteht durch die gleichen Lichtverhältnisse.

Auch in Abbildung 4.10 erkennt man die Auswirkung von Helligkeitsverläufen: Der

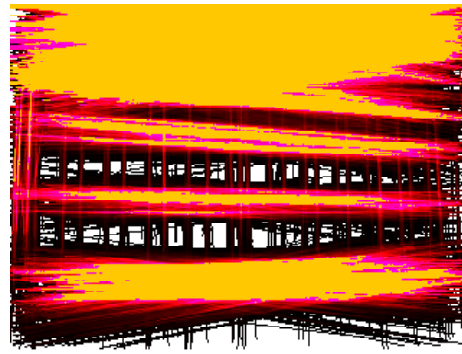
Schatten auf der Straße rechts im Bild erzeugt in der zweidimensionalen Abbildung einen Graustufenverlauf.

In Abbildung 4.11 ist dieser Effekt am Schatten zu sehen, den das Gebäude auf die Straße wirft. Ansatzweise sieht man hier die Wirkung von Verläufen wieder am Himmel; dieser macht jedoch nur einen kleinen Teil des Bildes aus und ergibt daher weniger Detektionen.

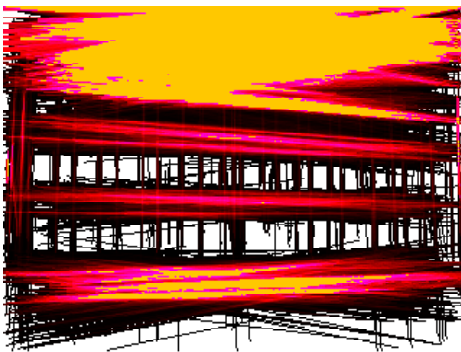
Es ist nicht möglich, dem Problem von Helligkeitsverläufen über Demaskierung beizukommen: Wie in Abschnitt 4.6.1 sind die detektierten Liniensegmente in Form von kontrastarmen Parallelen in den Bildern vorhanden und müssen detektiert werden.



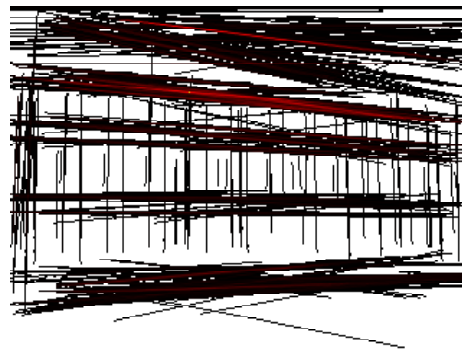
(a) Ausgangsbild



(b) Detektion mit Endpunkteigenschaft, 475334 Liniensegmente.



(c) Direkte Demaskierung, 223173 Liniensegmente.



(d) Wechselseitige Demaskierung, 3303 Liniensegmente.

Abbildung 4.12: Mannheimer Schloss, Westflügel (240×320). $p = \frac{1}{16}$, $\epsilon = 1$.

4.7 Unproblematische Bilder

In der Computer Vision ist es nicht immer einfach, im Vornherein zu beurteilen, ob ein Algorithmus ein Problem gut oder schlecht lösen wird: Viele Problemfälle eines Algorithmus ergeben sich durch Bildinhalte, die zuvor nicht berücksichtigt wurden. Ein Grund dafür ist, dass die menschliche Wahrnehmung selbstverständlich optischen Reizen des Sehnervs eine Bedeutung zuordnet, indem “Wichtiges” von “Unwichtigem” unterschieden und das Unwichtige nicht mehr bewusst wahrgenommen wird (die ideegebende Gestaltpsychologie beschreibt diesen Zusammenhang, siehe Abschnitt 2.1.1).

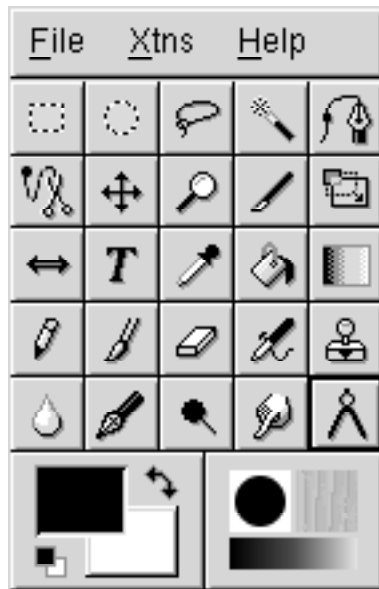
Dieses Anschauungsproblem ist auch hier vorhanden. Dennoch gibt es einige Anhaltspunkte, die eine Beurteilung erlauben, wann die Detektion mit dem hier vorgestellten Algorithmus unproblematisch ist. Diese Anhaltspunkte sind Schlussfolgerungen der vorangegangenen Abschnitte dieses Kapitels:

- Je eindeutiger Linienstrukturen voneinander differenziert sind, desto besser gelingt die Detektion (vgl. parallelen Linien in Abschnitt 4.2).
- Helligkeitsverläufe in einem Bild sollten entweder in einem Vorverarbeitungsschritt identifiziert und entfernt werden, oder nachträglich über die Eigenschaften solcher Verläufe identifiziert werden (vgl. Abschnitt 4.6.3).
- Die Größenverhältnisse der abgebildeten Objekte sollten sich bei perspektivischer Verzerrung nicht zu sehr verschieben (vgl. Abschnitt 4.6.1).
- Der lokale Gradient erfordert eine ausreichende Bildqualität, damit die Richtungen an Bildpunkten “richtig” berechnet werden (vgl. Abschnitt 4.5).

Als Beispiel für solche Voraussetzungen ist Abbildung 4.13 gegeben. Es gibt keine perspektivischen Verzerrungen, aufgrund seiner synthetischen Natur hat das Bild eine hervorragende Bildqualität. Es gibt nur einen kleinen Farbverlauf, der aber grob quantisiert ist und daher nicht problematisch wird, obwohl sein Einfluss zu sehen ist. Die vorhandenen Linienstrukturen sind klar voneinander differenziert.

Rückblick. Die vorangegangenen Abschnitte dieses Kapitels haben einen Eindruck davon vermittelt, dass die wesentlichen Linienstrukturen auch bei Bildern robust detektiert werden, die nicht alle Eigenschaften unproblematischer Bilder erfüllen.

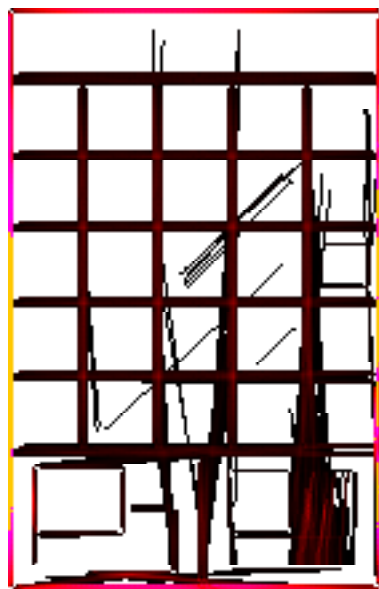
Probleme ergeben sich immer, wenn parallele Linientexturen vorhanden sind (siehe Abschnitt 4.6). Die gefundenen Liniensegmente sind in diesem Fall jedoch tatsächlich vorhanden, auch wenn sie nicht der menschlichen Wahrnehmung von Linienstrukturen entsprechen.



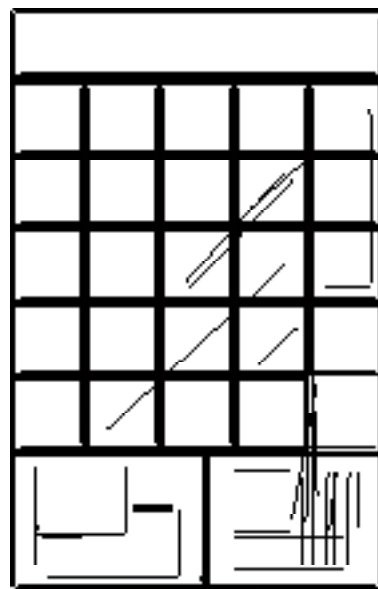
(a) Ausgangsbild



(b) Detektion (Endpunkteigenschaft), 27488 Liniensegmente.



(c) Direkte Demaskierung, 24220 Liniensegmente.



(d) Wechselseitige Demaskierung über Wahrscheinlichkeit, 136 Liniensegmente.

Abbildung 4.13: Oberfläche der “Toolbox” von *The GIMP* [GIMP] (220×142).
 $p = \frac{1}{16}, \epsilon = 1$.

5 Fazit und Ausblick

Diese Arbeit hat aufgezeigt, wie eine Detektion perzeptueller Primitive über die statistische Untersuchung von *Large Deviations* durchgeführt werden kann. Die Grundlage dafür war die Konkretisierung und Gliederung des Helmholtz-Prinzips in Kapitel 2 nach den Erkenntnissen der Gestalttheorie. Dort wurde gezeigt, dass das Helmholtz-Prinzip mit statistischen Abschätzungen plausibilisiert wird und wie der Parameter ϵ als Grenzwert für den Erwartungswert oder Falschalarmrate zu verstehen ist.

In Kapitel 3 war am Beispiel des perzeptuellen Primitivs “Liniensegment” zu verfolgen, wie die Schritte des Helmholtz-Prinzips in der Anwendung aussehen. Zunächst sind dabei die Einschränkung durch die zugrunde liegende diskrete Geometrie erklärt und gezeigt worden, wie mit dieser Einschränkung umzugehen ist. Dort wurde die Auswirkung des Maskierungsproblems veranschaulicht und für Liniensegmente gezeigt, wie eine Demaskierung erfolgt. Schließlich wurde auf die algorithmische Umsetzung und ihre Komplexität eingegangen.

Kapitel 4 zeigte an verschiedenen Testsequenzen, dass die Detektion von Liniensegmenten die erwarteten Ergebnisse liefert. Dabei wurde das Maskierungsproblem und die Demaskierung deutlich gemacht. Die Auswirkungen verschiedener Werte für den Präzisionsparameters p und die Falschalarmrate ϵ begründeten die Wahl der Parameter. Die Analyse der Detektion im Hinblick auf Rauschempfindlichkeit demonstrierte die Robustheit des Verfahrens. Die Verwendung verschiedener Gradienten zeigte einen Verbesserungsansatz und das Problem der geforderten statistischen Unabhängigkeit auf. Abschließend wurde die Problematik paralleler Strukturen erklärt und einige informelle Regeln für Bilder mit gut zu detektierenden Liniensegmenten abgeleitet.

Es ist festzustellen, dass eine algorithmische Umsetzung des Helmholtz-Prinzips je nachdem, wie gut die Durchmusterung des Suchraums realisiert wird, sehr rechenaufwändig werden kann. Ansätze zur Vereinfachung und Parallelisierung wurden für die Detektion von Liniensegmenten aufgezeigt und sollen als Ideen für eine Verbesserung der Algorithmen dienen.

Ein Schwerpunkt der Arbeit war die theoretische und praktische Verifikation der Arbeit von Desolneux et al. [DMM00], die das Helmholtz-Prinzip erstmals vorgestellt

haben. Zusammenfassend ist zu sagen, dass die Theorie leicht auf allgemeine perzeptuelle Primitive übertragen werden konnte. Eine Detektion von Liniensegmenten mit dem Helmholtz-Prinzip lieferte immer nachvollziehbare Ergebnisse, die in den meisten Situationen die wesentlichen Linien widerspiegelten.

Allerdings ist das Helmholtz-Prinzip entgegen der Behauptung von Desolneux et al. nicht als parameterfreier Ansatz zu bezeichnen, denn es müssen mit p und ϵ zwei Parameter bestimmt werden. Die Veränderungen ihrer Werte sind sichtbar, obwohl der Einfluss auf Liniensegmente als gering zu bewerten ist. Der von Desolneux et al. benötigte Quantisierungsparameter für die Durchmusterung des Suchraums konnte dagegen vermieden werden.

Weiterhin veränderte sich die Detektion der Liniensegmente durch verschiedene Wahl der Parameter auf die Art, wie es die allgemeine Interpretation erwarten ließ: Die Detektion wesentlicher Liniensegmente blieb stabil. p war direkt als Präzision zu interpretieren, die Auswirkungen von ϵ zeigten sich hauptsächlich an der minimalen Länge der Liniensegmente, was theoretisch zu erwarten war. Mit steigender Falschalarmrate durch Änderung von ϵ wurden auch Liniensegmente detektiert, die mit keinen echten Linienstrukturen korrespondierten.

Ein mehrfaches Problem war die Berücksichtigung der vom Ansatz vorausgesetzten statistischen Unabhängigkeit der Bestandteile eines perzeptuellen Primitivs. Bei der weiteren theoretischen Untersuchung des Helmholtz-Prinzips sollte darauf eingegangen werden, wie stark der Ansatz von dieser Voraussetzung abhängt. Mit den Standardwerken [DZ98], [dH00] über Large Deviations existieren fundierte Grundlagen für statistisch leicht abhängige Zufallsexperimente. [SSS93] liefert für das Helmholtz-Prinzip weitere Hinweise. Ein praktischer Test in dieser Arbeit anhand von Liniensegmenten konnte mit einem Gradienten höherer Lokalität zeigen, dass trotz leichter statistischer Abhängigkeit das Detektionsergebnis verbessert werden kann.

Eine noch ungeklärte Frage ist, wie die von der Gestalttheorie beschriebene rekursive hierarchische Gruppierung größerer Strukturen aus perzeptuellen Primitiven vollzogen werden kann. In dieser Arbeit ist bei der Konkretisierung des Ansatzes kurz aufgezeigt worden, wie das Helmholtz-Prinzip diesbezüglich einzuordnen ist.

Abschließend ist nach den erarbeiteten Ergebnissen zu erwarten, dass das Helmholtz-Prinzip als Grundlage für die verschiedensten perzeptuellen Primitive dienen kann. Die verwendeten Parameter haben aufgrund der statistischen Verifikation für jedes Primitiv die gleiche Bedeutung. Daher ist anzunehmen, dass sie für verschiedene Primitive ähnlich wenn nicht gleich zu wählen sind. Das kann eine Kombination verschiedener Detektionsergebnisse und -algorithmen ohne zahlreiche Parameterwahl ermöglichen.

A Anhang

A.1 Operatoren

A.1.1 Rundungsoperator *round*

$$\text{round}(a) = \begin{cases} \lfloor a \rfloor & , a - \lfloor a \rfloor < 0.5 \\ \lceil a \rceil & , a - \lfloor a \rfloor \geq 0.5 \end{cases}$$

A.1.2 Vorzeichenoperator *sign*

$$\text{sign}(a) = \begin{cases} +1 & , a \geq 0 \\ -1 & , a < 0 \end{cases}$$

A.2 Herleitung der Rekursionsformel für die Binomialsomme

Sei $P(k, u)$ die Summe über den linken Rand $0, \dots, k$ der Binomialverteilung mit Breite u und festem $p \in [0, 1]$:

$$P(k, u) = \sum_{i=0}^k \binom{u}{i} p^i (1-p)^{u-i}$$

Dann gilt $P(k+1, u+1) = pP(k, u) + (1-p)P(k+1, u)$.

Beweis. Es gilt folgende Relation R zwischen den Binomialkoeffizienten (siehe z.B. [Rol01], zur Veranschaulichung betrachte man das *Pascal'sche Dreieck*):

$$R: \quad \binom{u+1}{i+1} = \binom{u}{i} + \binom{u}{i+1}$$

$$\begin{aligned}
 P(k+1, u+1) &= \sum_{i=0}^k \binom{u+1}{i+1} p^{i+1} (1-p)^{u-i} + \underbrace{\binom{u+1}{0} p^0 (1-p)^u}_{(1-p)^u} \\
 &\stackrel{R}{=} \sum_{i=0}^k \left(\binom{u}{i} + \binom{u}{i+1} \right) p^{i+1} (1-p)^{u-i} + (1-p)^u \\
 &= \sum_{i=0}^k \binom{u}{i} p^{i+1} (1-p)^{u-i} + \sum_{i=0}^k \binom{u}{i+1} p^{i+1} (1-p)^{u-i} + (1-p)^u \\
 &= p \underbrace{\sum_{i=0}^k \binom{u}{i} p^i (1-p)^{u-i}}_{P(k,u)} + \\
 &\quad (1-p) \underbrace{\sum_{i=0}^k \binom{u}{i+1} p^{i+1} (1-p)^{u-(i+1)}}_{P(k+1,u)} + (1-p)^u \\
 \Leftrightarrow P(k+1, u+1) &= pP(k, u) + (1-p)P(k+1, u)
 \end{aligned}$$

Diese hier für den linken Rand der Binomialverteilung gezeigte Rekursion gilt ebenso für den rechten, da für die Binomialkoeffizienten gilt:

$$\forall i \geq u : \binom{u}{i} = 0$$

□

Literaturverzeichnis

- [Bra98] Louis Brassard. *The Perception of the Image World*. Dissertation, School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada, V5A 1S6, Jul. 1998.
- [Bre65] Jack E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, 1965.
- [BSMM99] I. N. Bronstein, K. A. Semendjajew, G. Musiol und H. Mühlig. *Taschenbuch der Mathematik*. Harri Deutsch, 4. Auflage, 1999.
- [Can86] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [Cao03] Frédéric Cao. Good continuations in digital image level lines. In *IEEE International Conference on Computer Vision, ICCV'03*, Seiten 440–447, Nice, France, October 2003.
- [Che52] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sums of observations. *Annals of Mathematical Statistics*, 23(4):493–507, 1952.
- [Cra99] Harald Cramér. *Mathematical Methods of Statistics*. Princeton Landmarks in Mathematics and Science. Princeton University Press, 1946, reprinted 1999.
- [dH00] Frank den Hollander. *Large Deviations*, Jgg. 14 of *Fields Institute Monographs*. American Mathematical Society, 2000.
- [DMM00] Agnès Desolneux, Lionel Moisan und Jean-Michel Morel. Maximal meaningful events and applications to image analysis, 2000. Preprint CMLA (<http://www.cmla.ens-cachan.fr>).
- [DS89] Jean-Dominique Deuschel und Daniel W. Stroock. *Large deviations*. Boston: Academic Press, 1989.

- [DZ98] Amir Dembo und Ofer Zeitouni. *Large Deviations Techniques and Applications*. Springer Verlag, 2. Auflage, 1998.
- [Fel68] William Feller. *An Introduction to Probability Theory and Its Applications*. John Wiley & Sons Inc., 3. Auflage, 1968.
- [GIMP] Spencer Kimball und Peter Mattis. The GIMP 1.2.5. GNU General Public License, 2004. GNU Image Manipulation Program (<http://www.gimp.org>).
- [HEK02] Joachim Hartung, Bärbel Elpelt und Karl-Heinz Klösener. *Statistik*. R. Oldenbourg Verlag, 13. Auflage, 2002.
- [Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, März 1963.
- [Hou62] Paul V. C. Hough. A Method and Means for Recognizing Complex Patterns. US Patent: 3,069,654, Dec. 1962.
- [Jäh97] Bernd Jähne. *Digitale Bildverarbeitung*. Springer Verlag, 4. Auflage, 1997.
- [Kan79] Gaetano Kanizsa. *Organization in Vision*. Praeger Publishers, 1979.
- [KEB91] Nahum Kiryati, Y. Eldar und Alfred M. Bruckstein. A Probabilistic Hough Transform. *Pattern Recognition*, 24:303–316, 1991.
- [Low85] David G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers, 1985.
- [LR95] John M. Lervik und Tor A. Ramstad. Subband Coding of Color Images with Dynamic Arithmetic Coder Allocation. In *Norwegian Signal Processing Symposium*, Seiten 211–216, 1995.
- [LR96] John Lewis und Raymond Russell. An Introduction to Large Deviations for Teletraffic Engineers. <ftp://www.stp.dias.ie/DAPG/LDtut96.ps>, 1996.
- [Mar82] David Marr. *Vision*. W. H. Freeman and Company, San Francisco, CA, USA, 1982.
- [Matlab] The MathWorks Inc. Matlab - The Language of Technical Computing, 6.5.0 Release 13, 2002. <http://www.mathworks.com>.

-
- [Met75] Wolfgang Metzger. *Gesetze des Sehens*. Waldemar Kramer Verlag, 3. Auflage, 1975.
- [MG00] Pascal Monasse und Frédéric Guichard. Fast computation of a contrast-invariant image representation. *IEEE Transactions on Image Processing*, 9(5):860–872, May 2000.
- [OW96] Thomas Ottmann und Peter Widmayer. *Algorithmen und Datenstrukturen*. Spektrum Akademischer Verlag, 3. Auflage, 1996.
- [Rol01] Timothy Rolfe. Binomial coefficient recursion: the good, and the bad and ugly. *SIGCSE Bull.*, 33(2):35–36, 2001.
- [Rub21] Edgar Rubin. *Visuell wahrgenommene Figuren*. Gyldendals, Kopenhagen, 1921.
- [Sha48] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, Jul and Oct 1948.
- [SKJ97] Hanno Scharr, Stefan Körkel und Bernd Jähne. Numerische Isotropieoptimierung von FIR-Filtern mittels Querglättung. In Erwin Paulus und Friedrich M. Wahl, Hrsg., *19. DAGM-Symposium Braunschweig*, Seiten 367–374. Springer Verlag, 1997.
- [Slu77] Eric V. Slud. Distribution inequalities for the binomial law. *Annals of Probability*, 5(3):404–412, 1977.
- [Soi98] Pierre Soille. *Morphologische Bildverarbeitung*. Springer Verlag, 1998.
- [SSS93] Jeanette P. Schmidt, Alan Siegel und Aravind Srinivasan. Chernoff-Hoeffding Bounds for Applications with Limited Independence. In *Symposium on Discrete Algorithms (A Conference on Theoretical and Experimental Analysis of Discrete Algorithms)*, 1993.
- [SYK96] Doron Shaked, O. Yaron und Nahum Kiryati. Deriving Stopping Rules For The Probabilistic Hough Transform By Sequential Analysis. *Computer Vision and Image Understanding*, 63:512–526, 1996.
- [vH67] Hermann Ludwig Ferdinand von Helmholtz. Handbuch der physiologischen Optik I und II. *Allgemeine Encyclopädie der Physik*, 9, 1867.
- [vH98] Hermann Ludwig Ferdinand von Helmholtz. *Schriften zur Erkenntnistheorie: Die Tatsachen der Wahrnehmung; kommentiert von M. Schlick und P. Hertz*. Springer Verlag, 1998.

- [Wer23] Max Wertheimer. Untersuchungen zur Lehre von der Gestalt II. *Psychologische Forschung*, 4:301–350, 1923.

Ehrenwörtliche Erklärung

Ich versichere, dass ich die vorliegende Diplomarbeit selbstständig verfasst und nur mit den angegebenen Quellen und Hilfsmitteln gearbeitet habe. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mannheim, 14. Mai 2004,