

Aus dem Institut für Signalverarbeitung und Prozessrechentchnik
der Universität zu Lübeck

Direktor:
Prof. Dr. rer. nat. Walter Dosch (kommissarisch)

Algorithmen zur Bestimmung überlagerter Bewegungen und Orientierungen in Bilddaten

Inauguraldissertation
zur
Erlangung der Doktorwürde (Dr.-Ing.)
der Universität zu Lübeck
– Aus der Technisch-Naturwissenschaftlichen Fakultät –

Vorgelegt von
Diplom-Informatiker Ingo Stuke
aus Bremen

Lübeck, im März 2006

Vorwort

Im Laufe meines Studiums an der Universität zu Lübeck wurde durch meine Diplomarbeit zur Bewegungsbestimmung mein Interesse an der Bildverarbeitung und dem wissenschaftlichen Arbeiten geweckt. Die Assistenzstätigkeit am Institut für Signalverarbeitung und Prozessrechenstechnik ermöglichte mir, mich weiter intensiv mit angewandten medizinischen aber auch theoretischen Fragestellungen im Bereich der Bildverarbeitung zu beschäftigen sowie an einem von der DFG geförderten Projekt zur Bewegungsbestimmung und Zeitreihenanalyse mitzuwirken. Im Rahmen dieses Projektes bildete sich mein Forschungsschwerpunkt zur Bestimmung überlagerter Bewegungen und Orientierungen aus, den ich durch Einreichen meiner Dissertation an der hiesigen Technisch-Naturwissenschaftlichen Fakultät zu einem vorläufigen Abschluss bringe.

Bedanken möchte ich mich bei Herrn Professor T. Aach für seine vielen fachlichen Impulse, die freie Forschungsatmosphäre und das hohe Interesse an meiner Forschungsarbeit sowohl während unserer gemeinsamen Zeit an seinem Lübecker Institut für Signalverarbeitung als auch nach seinem Wechsel zum Lehrstuhl für Bildverarbeitung an der RWTH Aachen. Durch seine bereitwillige und kritische Auseinandersetzung mit dem Thema dieser Arbeit hat er wesentlich zu ihrem Gelingen beigetragen. Mein ganz besonderer Dank gilt den DFG-Projektpartnern C. Mota und E. Barth aus dem Institut für Neuro- und Bioinformatik für die intensive und erfolgreiche Zusammenarbeit. Unser freundschaftlicher Gedankenaustausch über Institutsgrenzen hinweg sowie ihre inhaltlichen Anregungen, Hinweise und Diskussionen leisteten einen hohen Beitrag am Zustandekommen dieser Arbeit. Entsprechend danke ich der DFG für die Unterstützung dieses Projektes. Bedanken möchte ich mich darüber hinaus für die vielen fachlichen Diskussionen und Anregungen bei meinen lieben Kollegen am Institut.

Ferner bedanke ich mich bei Herrn Professor Siegfried J. Pöpl für den Vorsitz im Promotionsausschuss, bei Herrn Privatdozent Erhardt Barth für die Annahme des Referats und bei Herrn Professor Til Aach für die Übernahme des Korreferates.

Ohne die emotionale Unterstützung, die Geduld und das Verständnis meiner Frau Kathrin und unseres 6 Monate alten Sohnes Lukas wäre die Umsetzung dieser Arbeit kaum möglich gewesen. Auch bei ihnen möchte ich mich herzlich bedanken.

Lübeck, im März 2006

Ingo Stuke

Datum der mündlichen Prüfung: 23. Juni 2006

Erstgutachter: PD Dr.-Ing. Erhardt Barth

Zweitgutachter: Prof. Dr.-Ing. Til Aach

Vorsitzender: Prof. Dr.-Ing. Dr. med. habil. S. J. Pöppel

Inhaltsverzeichnis

1	Einführung	1
1.1	Bewegungsbestimmung	2
1.1.1	Matchingmethoden	3
1.1.2	Differenzielle Methoden	3
1.1.3	Filterbasierte Methoden	4
1.2	Überlagerte Bewegungen	5
1.2.1	Transparente Überlagerung	6
1.2.2	Verdeckende Überlagerung	9
1.3	Leitfaden	10
2	Optischer Fluss	13
2.1	Optischer Fluss für mehrere Bewegungen	14
2.1.1	Linearisierung des optischen Flusses	17
2.1.2	Separierung der Geschwindigkeitsvektoren	19
2.1.3	Überlagerte Orientierungen	21
2.2	Konsistenz der Lösung	22
2.3	Zusammenfassung	26
3	Verallgemeinertes Strukturtensorverfahren	27
3.1	Tensorverfahren	27
3.1.1	Strukturtensor für eine Bewegung	28
3.1.2	Praktische Berechnung des Strukturensors	30
3.1.3	Konfidenzmaß des Strukturensors	30
3.1.4	Generalisierter Strukturtensor	32
3.1.5	Transparente Bewegungen in Farbsequenzen	33
3.1.6	Größe des Strukturensors	35
3.1.7	Konfidenzmaße und hierarchische Bewegungsbestimmung	35
3.1.8	Hierarchische Bewegungsbestimmung	39
3.2	Lösungsraum für mehrfache Orientierungen	43
3.3	Ergebnisse	48
3.4	Zusammenfassung	54
4	Regularisierung der Bewegungsfelder	57
4.1	Regularisierung für zwei Bewegungen	58
4.1.1	Lineare und nichtlineare Formulierung	60
4.1.2	Iterationsvorschrift	61

4.2	Erweiterung zu N Bewegungen	63
4.3	Ergebnisse	64
4.4	Zusammenfassung	70
5	Erweiterungen des optischen Flusses	71
5.1	Additive Helligkeitsänderung	73
5.2	Multiplikative Helligkeitsänderung	73
5.3	Exponentielle Helligkeitsänderung	75
5.4	Ergebnisse	76
5.5	Diskussion	80
6	Phasenbasierte Bewegungsbestimmung	83
6.1	Phasen-Modell	84
6.2	Bestimmung der Phasenverschiebungen	84
6.3	Singularität der Matrix \mathbf{A}_N	86
6.3.1	Beispiel für zwei Bewegungen	87
6.4	Berechnung der Geschwindigkeitsvektoren	88
6.5	Zusammenfassung	92
7	Trennung der bewegten Schichten	95
7.1	Trennung global bewegter Schichten im Fourierraum	98
7.2	Invertierbarkeit der Vandermonde-Matrix	98
7.2.1	Beispiele	100
7.3	Ausblick	103
8	Block Matching für überlagerte Bewegungen	105
8.1	Die Block Matching Bedingung	107
8.1.1	Kausalität und Verfeinerung	108
8.2	Hierarchischer Algorithmus	109
8.2.1	Stochastisches Bildmodell	110
8.2.2	Einfache und zweifach transparente Bewegungen	111
8.3	Verdeckte Bewegungen	112
8.3.1	Wahl des adäquaten transparenten Bewegungsmodells	113
8.3.2	Der hierarchische Algorithmus	114
8.4	Bewegungsbestimmung mittels Markov-Zufallsfeldern	115
8.4.1	Bayes'sche Formulierung für zwei Bewegungen	116
8.4.2	Das Beobachtungsmodell	116
8.4.3	Räumliche Glattheit	117
8.4.4	Optimierungsalgorithmus	118
8.4.5	Zeitliche Glattheit	120
8.5	Ergebnisse	122
8.6	Zusammenfassung	128
9	Verdeckung	129
9.1	Verdeckende Bewegungen	130
9.2	Räumliche Analyse	131
9.3	Verdeckung im Frequenzraum	134

9.3.1	Verlauf des Störterms	136
9.4	Gerade Verdeckungsgrenze	137
9.5	Experimentelle Verifikation	142
9.6	Zusammenfassung	146
10	Überlagerte Orientierungen in Bildern	149
10.1	Einfache Orientierung	151
10.2	Transparente Orientierungen	151
10.2.1	Separierung der Orientierungsvektoren	152
10.2.2	Der Winkel zwischen beiden Orientierungsvektoren	155
10.2.3	Orientierung in Farbbildern	156
10.2.4	Konfidenz und Invarianten	157
10.3	Bestimmung verdeckender Doppelorientierung	158
10.3.1	Orientierungsbestimmung in Farbbildern	160
10.4	Schichtentrennung und Regionenbestimmung	162
10.4.1	Trennung transparenter Muster im Frequenzraum	162
10.4.2	Regionenermittlung bei verdeckenden Orientierungen	163
10.5	Ergebnisse	163
10.6	Zusammenfassung	175
11	Diskussion	177
A	Analytische Bewegungsbestimmung	181
A.1	Berechnung des Eigenvektors	181
A.2	Der gemittelte Eigenvektor	183
A.3	Analytische Konfidenzmaßberechnung	184
B	Separierungsverfahren von Shizawa & Mase	185
C	Algorithmen für eine, zwei und drei Bewegungen	187
C.1	Algorithmus für eine Bewegung	187
C.2	Algorithmus für zwei Bewegungen	187
C.3	Algorithmus für drei Bewegungen	189
D	Mehrdimensionale δ-Funktionen	191
E	Spektrum der Indikatorfunktion	193
F	Symbol-Tabelle	195

Zusammenfassung

Rechnergesteuert dynamische Bewegungsvorgänge in Bildfolgen zu erkennen sowie zuverlässig und genau zu bestimmen, stellt eine wichtige Aufgabe in der digitalen Bildverarbeitung dar. Beispiele sind die Videokomprimierung, die Objektsegmentierung, das digitale Fernsehen oder die Rauschreduktion in medizinischen Bilddaten durch bewegungsabhängige Filterung. Für die Roboternavigation ist es sogar eine Voraussetzung.

Zum Erkennen einfacher Bewegungen in Bildfolgen oder Orientierungen sind in den letzten Jahren mehrere Bestimmungsverfahren entwickelt worden. Mit ihnen liegen inzwischen umfangreiche Erfahrungen vor. Ihre Grenzen werden unter anderem dann erreicht, wenn an den Bildpunkten einer Sequenz mehr als nur eine Bewegung auftritt und sich mehrere verschiedene Bewegungen transparent überlagern. Zum Beispiel treten an durchsichtigen Scheiben unabhängig voneinander bewegte reflektierte und transparente Bilder gleichzeitig auf oder medizinische Röntgenaufnahmen nehmen bei größerer Durchstrahlungstiefe ein Bild auf, das eine Überlagerung mehrerer Gewebsschichten zeigt. Neben der reinen Bewegungsüberlagerung können solche Reflexions- und Durchstrahlungsaufnahmen zu einer Überlagerung orientierter Strukturen führen und lokale Umgebungen erzeugen, die nicht mehr durch nur eine Orientierung beschreibbar sind. An solchen Stellen versagen die Standardverfahren. Doch genau bei diesen Bereichen handelt es sich oft um markante Stellen in der Bild- und mehrdimensionalen Signalverarbeitung. Ecken bzw. Punkte mit mehreren Orientierungen eignen sich besonders gut als Korrespondenzpunkte für Stereoskopie- und Registrierungsaufgaben. Für Mustererkennungs- und Klassifikationsaufgaben ist es notwendig, auch mehrfache orientierte Bereiche analysieren und aus der Orientierungsinformation Merkmale gewinnen zu können. Diese Arbeit erweitert Erkennungsverfahren auch auf Anwendungen mit mehrfachen Überlagerungen und Orientierungen.

Bekanntere Verfahren zur Bestimmung einfacher Bewegungen lassen sich grob in drei Klassen einteilen: Die differenziellen Methoden nutzen Ableitungen der Bildsequenz für die Bewegungsbestimmung, die phasenbasierten Methoden ermitteln die Phasenverschiebungen, die von den Bewegungen verursacht werden, und die Vergleichsmethoden versuchen, im Folgebild Bereiche des Vorgängerbildes wieder zu finden. Für jede dieser Klassen konnten wir hier passende Verfahren zur Bestimmung einer beliebigen Anzahl transparent überlagelter Bewegungen entwickeln.

Im Bereich der differenziellen Methoden haben wir zwei Verfahren vorgestellt, die auf der Pionierarbeit von Shizawa & Mase [147, 148] basieren. Shizawa & Mase [147, 148] stellten eine optische Flussgleichung für transparente Bewegungen auf, die Nichtlinearitäten in den gesuchten Bewegungsvektorkomponenten aufweist. Durch Einführen der gemischten Ge-

schwindigkeitsparameter können wir hier diese Nichtlinearität auf die Nullstellenbestimmung eines komplexen Polynoms reduzieren und damit eine einfache Lösungsvorschrift für eine beliebige Anzahl überlagerter Bewegungen angeben. Die Ordnung des Polynoms wächst linear mit der Anzahl überlagerter Bewegungen.

Die erste differenzielle Methode ist ein lokales Verfahren, das auf der Eigenystemanalyse von Tensoren basiert. Jeder Bewegungsanzahl N ist genau ein Tensor zugeordnet, dessen Einträge sich aus Produkten partieller Ableitungen N -ter Ordnung ergeben. Der Eigenvektor zum kleinsten Eigenwert ergibt den gesuchten gemischten Geschwindigkeitsparametervektor. Ferner lassen sich aus den Invarianten der Tensoren Konfidenzkriterien ableiten, mit denen entschieden werden kann, ob das gerade betrachtete Modell mit der Anzahl beobachteter Bewegungen übereinstimmt oder nicht. Dieses Konfidenzmaß ermöglicht die Angabe eines hierarchischen Verfahrens, um eine beliebige Anzahl überlagerter Bewegungen zu identifizieren und zu bestimmen.

Das zweite differenzielle Verfahren ist im Gegensatz dazu der Gruppe der globalen Methoden zuzuordnen, bei denen anstelle von lokalen Energieausdrücken ein globales Kriterium optimiert wird. Die eigentliche Optimierung wird später jedoch lokal durchgeführt. Statt einzelner Bewegungsvektoren betrachtet es das gesamte Bewegungsvektorfeld. Die notwendigen Glattheitsbedingungen, die zur Lösung des Energiefunktional an die Bewegungsvektorfelder zu stellen sind, werden indirekt auf den gemischten Geschwindigkeitsparameterfeldern formuliert, um ein lineares Gleichungssystem von Euler-Lagrangeschen Differenzialgleichungen zu erhalten. Nach der iterativen Lösung des Differenzialgleichungssystems mit dem Gauss-Seidel-Verfahren ergeben sich die gesuchten Bewegungsvektorfelder aus den Nullstellen komplexer Polynome. Die Koeffizienten der Polynome sind durch die gemischten Geschwindigkeitsparameter bestimmt. Dieses Verfahren ergibt im Gegensatz zu den lokalen dicht besetzte Bewegungsvektorfelder und eignet sich besonders gut in den Fällen, in denen mit den lokalen Methoden eine Bewegungsbestimmung nur an sehr wenigen Punkten der Sequenz möglich ist. Das hier vorgestellte Verfahren führt jedoch keine Unterscheidung im Sinne der Anzahl von Überlagerungen durch, sondern nimmt ein Modell an, das dann für den kompletten Bildbereich gilt. Experimente zeigen jedoch, dass selbst dann, wenn weniger Bewegungen beobachtet werden als vom Modell vorgesehen und damit eine Komponente frei wählbar ist, gute Ergebnisse erzielt werden.

Kommen Helligkeitsänderungen in der Sequenz vor, ist die Annahme, dass die Intensitätsänderungen nur auf die Bewegungen zurückzuführen sind, nicht mehr gültig. Um dennoch gute Ergebnisse zu erzielen, ist die optische Flussgleichung den jeweiligen Beleuchtungssituationen anzupassen. Das transparente Bewegungsmodell ist hier unter diesem Aspekt untersucht und um ein additives und ein multiplikatives Beleuchtungsmodell erweitert worden. Die daraus abgeleiteten Gleichungen lösten wir mit einem Tensorverfahren, das die simultane Bestimmung der Bewegungen und der Beleuchtungsmodellparameter erlaubt und direkt in das differenzielle hierarchische Verfahren integriert werden kann.

Bei den phasenbasierten Methoden konnte die Methode von Vernon [174] auf eine beliebige Anzahl von überlagerten Bewegungen erweitert werden. Ähnlich wie bei der differenziel-

len Methode führen wir zur Linearisierung des Problems einen so genannten gemischten Phasenterm ein. Er bildet die Koeffizienten eines komplexen Polynoms, dessen Nullstellen den gesuchten Phasentermen entsprechen. Dieses Verfahren eignet sich besonders zur Bestimmung globaler Bewegungen und stellt ein Werkzeug zur Verfügung, mit dem sich schnell ein Eindruck über die Größenordnungen der Geschwindigkeiten ermitteln lässt.

Die ermittelten Phaseninformationen können zur Trennung der bewegten Schichten im Frequenzbereich genutzt werden. Allerdings ist die Trennung nicht an allen Frequenzen durchführbar. Einige dieser Bereiche sind durch Geraden durch den Ursprung beschrieben, deren Ausrichtung von den Bewegungen abhängig ist, während die anderen von den Spektren der Signale selbst abhängen. Zur Rücktransformation in den Ort werden die fehlenden Informationen interpoliert und es ergeben sich für jede Gerade im Frequenzraum orientierte Fehlerstrukturen im Ortsraum.

Das Block-Matching-Verfahren ist ein sehr weit verwendetes Bewegungsbestimmungsverfahren und z.B. oft bei der Videokompression eingesetzt. Die Verallgemeinerung der phasenbasierten Methode ermöglicht es, ein Block-Matchingverfahren herzuleiten, das auch für eine beliebige Anzahl von überlagerten Bewegungen geeignet ist und sich leicht in bestehende Verfahren integrieren lässt. In diesem Zusammenhang wurden zwei Verfahren vorgestellt. Das erste Verfahren führt eine hierarchische Bewegungsbestimmung bis zu einer beliebigen Anzahl von Bewegungen durch. Anhand eines statistischen Konfidenzkriteriums wird über die Annahme oder Ablehnung eines Bewegungsmodells entschieden. Die zweite Methode nutzt ein Markov-Zufallsfeld zur Formulierung von Glattheitsbedingungen der Bewegungsvektorfelder und zur simultanen Bestimmung einer Segmentierung, die die Anzahl beobachteter Bewegungen an jedem Bildpunkt angibt und zusammenhängende Bereiche gleicher Bewegungszahl bevorzugt.

Die Bewegungsbestimmung an Verdeckungsgrenzen ist Gegenstand vieler Forschungsarbeiten. Im Rahmen dieser Arbeit wurde daher sowohl im Orts- als auch im Frequenzbereich untersucht, welche Störterme auftreten, wenn das transparente Modell zur Bestimmung der Bewegungen an der Verdeckungsgrenze genutzt wird. Im Frequenzraum besitzt der Störterm im Allgemeinen einen hyperbolischen Verlauf. Ist die Verdeckungsgrenze durch eine Gerade beschrieben, kann der exakte Ausdruck angegeben werden. Entlang bestimmter Parallelebenen besitzt der Störterm das identische aber mit einer hyperbolischen Funktion gewichtete Spektrum wie das verdeckte Signal. Als ein interessanter Effekt ergibt sich dabei, dass sich das verdeckende Modell unter bestimmten Umständen, die von der Form der Verdeckungsgrenze und den Bewegungen abhängen, wie das transparente verhält.

Das primäre Ziel dieser Arbeit war die Entwicklung verschiedener Algorithmen zur Bestimmung transparent überlagerter Bewegungen. Sie sollen sich möglichst an bestehenden Verfahren für eine Bewegung orientieren, um ihre Integrierbarkeit zu vereinfachen und ihre Akzeptanz zu erhöhen. Besonders wichtig war mir dabei die Entwicklung entsprechender differenzieller Methoden und das Block-Matchingverfahren. Insgesamt werden sieben Verfahren vorgestellt, die die Freiheit bieten, für die jeweilige Anwendung das adäquateste auszuwählen. Da die Bewegungsbestimmung als Spezialfall der Orientierungsanalyse auf-

gefasst werden kann, stellte sich die Frage, inwiefern die Konzepte der Bewegungsbestimmung auch auf die Orientierungsanalyse in Bildern übertragbar sind. Die Übertragung für zwei Orientierungen gestaltet sich als einfach. Zwei verschiedene Orientierungsmodelle wurden vorgestellt, mit denen wirkungsvolle Ergebnisse für eine Vielzahl von realen und synthetischen Beispielen erzielt wurden. Gleichzeitig demonstrieren diese Beispiele die hohe Relevanz der Mehrfachorientierungsanalyse in Bilddaten und damit ihre Bedeutung für Computer Vision Anwendungen, wie z.B. Tracking-, Klassifikations- oder Registrierungsanwendungen.

Kapitel 1

Einführung

Im Bereich der Computer Vision und der digitalen Bildverarbeitung wird normalerweise angenommen, dass Tiefe und Intensität an jedem Bildpunkt skalare Größen sind. Trotzdem finden sich in der realen Welt immer wieder Beispiele, in denen sich die Beobachtung aus einer Superposition mehrerer oft unabhängiger Beiträge ergibt. An Fensterscheiben beobachten wir solche Vorgänge täglich. Beim Blick aus einem Fenster überlagern sich ein äußeres Bild und die Reflexion eines inneren Bildes. Das Gesamtbild setzt sich aus den Intensitäten der beiden Bilder zusammen. Solche Bilderüberlagerungen werden häufig in Analogie zu Reflexionsvorgängen an transparenten Oberflächen als *transparente Schichten* bezeichnet. Ein ähnlicher Vorgang kann auch in der medizinische Röntgenbildgebung beobachtet werden, in der sich eine Projektion aus unterschiedlicher Gewebeschichten zusammensetzt.

Die Überlagerung mehrerer unabhängiger Bilder erschwert die Analyse und Interpretation, eine besondere Behandlung wird erforderlich. Beispielsweise könnte der Autofocus einer Kamera auf Grund der unterschiedlichen Distanzen beider Objekte zur Scheibe bzw. Kamera fehlgeleitet werden, oder Merkmalsextraktions- und Bewegungsbestimmungsverfahren in die Irre führen. Um solche Situationen zu verhindern, gilt es, diese Phänomene zu detektieren und sie gegebenenfalls in ihre Bestandteile zu zerlegen.

Ein einziges Bild ist für die Trennung transparenter Schichten nicht ausreichend. Vielmehr sind zwei oder mehrere Aufnahmen nötig, die zusätzlich einen gewissen Grad an Unterschieden aufweisen müssen. Einige Ansätze nutzen Bewegungsinformationen. Werden an den Bildpunkten aber mehrere Bewegungen beobachtet, lösen Standardverfahren, die für die Bestimmung einfacher Bewegungen konzipiert sind, das Problem nicht. Entsprechende Verfahren sind daher notwendig und zu entwickeln.

Zu diesem Zweck ist die Bestimmung transparent überlagerter Bewegungen das Thema dieser Arbeit. Ziel dabei ist, die klassischen Standardverfahren zur Bestimmung einfacher Bewegungen auf die Bestimmung beliebig vieler transparent überlagerter Bewegungen zu erweitern. Die Trennung überlagerter Schichten wird nicht Bestandteil dieser Arbeit sein, obwohl am Rande kurz auf sie eingegangen wird.

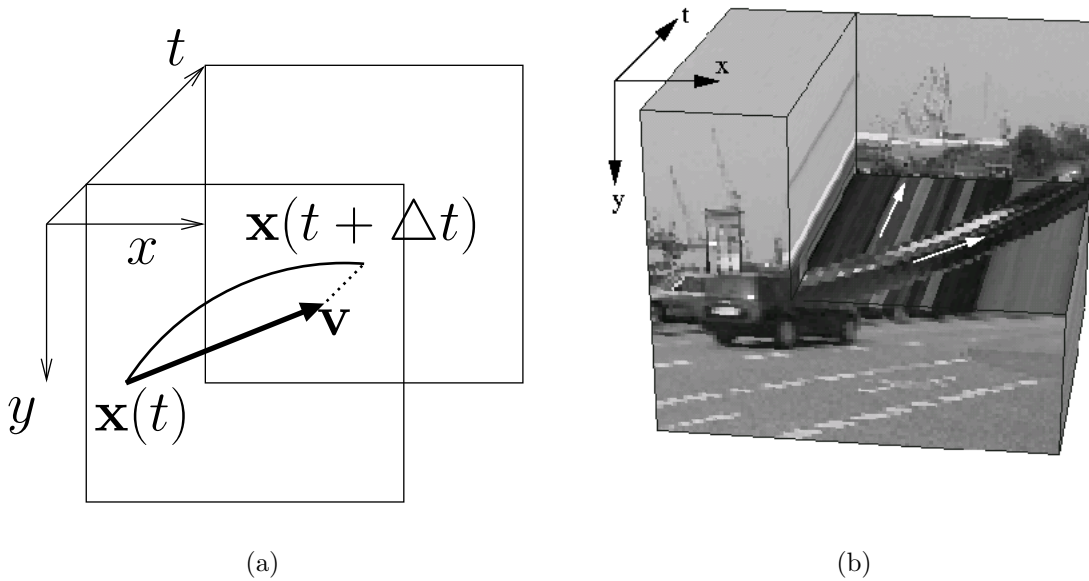


Abbildung 1.1: (a) Approximation kontinuierlicher Bewegungskurven durch Verschiebungsvektoren zwischen aufeinander folgenden Bildern. (b) Bewegte Punkte erzeugen Trajektorien gleicher Helligkeit im Ortszeitraum.

1.1 Bewegungsbestimmung

Bewegung ist eine der häufigsten Gründe für zeitlichen Änderungen in Bildsequenzen. Ihre Bestimmung und Analyse ist daher für viele Anwendungen wichtig, sie wird unter anderem bei autonomen Robotern, Objektverfolgung, Überwachung, Kamarabildstabilisierung, Videokompression, etc. eingesetzt. Während der Aufzeichnung werden die dreidimensionalen Bewegungen der Objekte im Raum auf zweidimensionale Bewegungen in der Bildebene projiziert. Diese zweidimensionalen Bewegungen, die in der Literatur auch häufig als *optischer Fluss* bezeichnet sind, gilt es aus den aufgenommenen Intensitäten beziehungsweise Farbinformationen zu ermitteln. Bewegungstrajektorien der realen Welt werden damit auf zweidimensionale Bewegungstrajektorien $(\mathbf{x}(t), t)$, $\mathbf{x} = (x, y)$ projiziert. Bildsequenzen sind durch eine schnelle Abfolge von einzelnen Bildern dargestellt und damit zeitdiskret. Die Bewegungstrajektorien werden dann ebenfalls nur zu bestimmten Zeitpunkten gemessen. Daher ergeben sich in vielen Bewegungsbestimmungsverfahren die Bewegungen entlang der Trajektorien durch Verschiebungsvektoren $\mathbf{v} = \mathbf{x}(t + \Delta t) - \mathbf{x}(t)$ aus je zwei aufeinander folgenden Bildern mit dem zeitlichen Aufnahmeversatz Δt (vergleiche Abbildung 1.1(a)).

Sind die Modelle für die zugrunde liegenden Bewegungen im Dreidimensionalen bekannt, kann ein Modell für die projizierten 2D Bewegungen abgeleitet werden, aus dem sich dann ein Lösungsverfahren ergibt [157]. Solche Situationen stellen vielfach jedoch nur eine Ausnahme dar, viele Verfahren gehen zumindest lokal von einfachen Translationen aus. Neben der Translationseigenschaft nimmt man vereinfachend konstante Bildintensitäten entlang der Trajektorie an und impliziert, dass die Intensitätsänderungen nur auf die Bewegungen

zurückzuführen sind, die Beleuchtungen aber konstant und die Objekte opak sind. Obwohl diese Voraussetzungen oft nicht erfüllt sind, stellen sie lokal meist eine ausreichende Approximation dar, mit der gewöhnlich gute Ergebnisse erzielt werden.

Translationseigenschaft und Intensitätskonservierung zusammen implizieren ein Bildmodell, bei dem ein Bild $g(\mathbf{x})$ in jedem Zeitschritt um den Vektor $\mathbf{v} = (v_x, v_y)^T$ verschoben wird. Unter diesen Voraussetzungen können wir die Bildsequenz $f(\mathbf{x}, t)$ durch

$$f(\mathbf{x}, t) = g(\mathbf{x} - \mathbf{v}t) \quad (1.1)$$

darstellen. Nach diesem Modell erzeugen bewegte Punkte, wie Abbildung 1.1(b) zeigt, orientierte Muster im Ortszeitraum und die Bewegungsbestimmung kann dann als ein Spezialfall der Orientierungsanalyse aufgefasst werden. Gleichung (1.1) bildet die Grundlage vieler Bewegungsbestimmungsverfahren, die sich in die drei wesentlichen Klassen differenzielle, Matching- und Filtermethoden einteilen.

1.1.1 Matchingmethoden

Matchingmethoden, auch Korrelationsmethoden genannt, stellen konzeptionell einfache Methoden der Bewegungsbestimmung dar und leiten sich direkt aus Gleichung (1.1) ab. Sie werden unter anderem gerne im Bereich der Videokompression eingesetzt. Prinzipiell ermitteln diese Verfahren den Verschiebungsvektor für einen Bildpunkt, in dem sie versuchen, eine lokale Region um jeden Punkt herum im nächsten Bild wieder zu finden. Der Bewegungsvektor errechnet sich dann aus alter und neuer Position der Region. Der Grad der Übereinstimmung wird vielfach durch die Summe quadrierter Differenzen (SSD) gemessen, und der Bewegungsvektor $\mathbf{v} = (v_x, v_y)^T$ am Punkt \mathbf{x} durch Minimieren der Fehlerfunktion

$$E_{\mathbf{x}}(\mathbf{v}) = \sum_{\mathbf{y} \in N_{\mathbf{x}}} (f(\mathbf{y}, t) - f(\mathbf{y} + \mathbf{v}, t - 1))^2 \quad (1.2)$$

ermittelt. Summiert wird dabei über eine kleine Nachbarschaft $N_{\mathbf{x}}$. Einige Variationen dieser Fehlerfunktion existieren (z.B. [12, 30]). Ein großer Nachteil dieser Verfahren ist die Minimierung der Fehlerfunktion: Die Optimierung wird meist mit mehr oder weniger schnellen Suchmethoden [89, 100, 101, 103, 166] durchgeführt. Der Suchraum ist diskret, und man gibt sich häufig mit ganzzahligen Geschwindigkeiten zufrieden.

1.1.2 Differenzielle Methoden

Die differenziellen Methoden nutzen Ableitungen der Bildsequenz für die Bewegungsbestimmung und leiten sich aus der Helligkeitskonstanzannahme ab. Bei konstanten Helligkeiten sind die Intensitätsänderungen von einem Bild zum Nächsten nur auf die Bewegungen zurückzuführen. Die Intensität eines Punktes ändert sich entlang seiner Bewegungstrajektorie nicht und bleibt damit über der Zeit konstant. Entsprechend verschwindet die totale Ableitung der Bildsequenz $f(\mathbf{x}, t)$ nach der Zeit, d.h.

$$\frac{df}{dt} = 0. \quad (1.3)$$

Drücken wir die totale Ableitung durch die partiellen aus, ergibt dies die klassische [84] *Brightness constancy constraint equation (BCCE)* (auch optische Flussgleichung genannt)

$$\frac{\partial f}{\partial x}v_x + \frac{\partial f}{\partial y}v_y + \frac{\partial f}{\partial t} = 0. \quad (1.4)$$

Der gesuchte Bewegungsvektor ist durch $\mathbf{v} = (v_x, v_y)^T$ gegeben. Die totale Ableitung ist damit als Richtungsableitung im Dreidimensionalen in Richtung $(v_x, v_y, 1)$ dargestellt und verdeutlicht, dass die Bestimmung als ein Spezialfall der Orientierungsanalyse aufgefasst werden kann, bei der die Komponente in zeitlicher Richtung fest vorgegeben ist. Eine entsprechende Bedingung zur Orientierungsanalyse im Dreidimensionalen würde daher $\frac{\partial f}{\partial x}v_x + \frac{\partial f}{\partial y}v_y + \frac{\partial f}{\partial t}v_t = 0$ mit dem Orientierungsvektor $\mathbf{v} = (v_x, v_y, v_t)^T$ lauten.

Die optische Flussgleichung, eine Gleichung mit zwei Unbekannten, ist unterbestimmt, mit ihr allein kann höchstens der Normalenfluss, d.h. der Geschwindigkeitsanteil senkrecht zum Bildgradienten ermittelt werden. Für die Bestimmung beider Bewegungsvektorkomponenten sind daher weitere Gleichungen erforderlich. Generell haben sich zur Lösung dieses Problems zwei Ansatzlinien etabliert.

Lokale Verfahren nehmen lokal konstante Bewegungen an und bestimmen die Bewegung durch Minimieren des Energiefunktional

$$E(\mathbf{v}) = \int_N \left(\frac{\partial f}{\partial x}v_x + \frac{\partial f}{\partial y}v_y + \frac{\partial f}{\partial t} \right)^2 \mathbf{d}\mathbf{x}, \quad (1.5)$$

wobei N eine offene Region um den betrachteten Punkt darstellt, für den die Bewegung ermittelt werden soll. Der Bewegungsvektor kann aus der Ableitung von $E(\mathbf{v})$ nach dem Bewegungsvektor bestimmt werden (siehe z.B. [28, 80, 81, 104]).

Die globalen Methoden optimieren anstelle lokaler globale Energiefunktionen, die sich auf den ganzen Bildbereich Ω beziehen. Zusätzliche Gleichungen werden durch Einführen der Nebenbedingung gewonnen, dass das Bewegungsfeld $v = (\mathbf{v}(\mathbf{x}))_{\mathbf{x} \in \Omega}$ glatt sein soll. Die Glattheit wird durch die partiellen Ableitungen der Bewegungsvektorkomponenten ausgedrückt und das Bewegungsfeld durch Energiefunktionale der Form

$$E(v) = \int_{\Omega} \left(\frac{\partial f}{\partial x}v_x + \frac{\partial f}{\partial y}v_y + \frac{\partial f}{\partial t} \right)^2 + \lambda (\|\nabla v_x\|^2 + \|\nabla v_y\|^2) \mathbf{d}\mathbf{x} \quad (1.6)$$

bestimmt.

1.1.3 Filterbasierte Methoden

Einige Methoden nutzen räumlich-zeitliche Filter, um daraus die Bewegungen, bzw. die lokalen Phasen zu bestimmen [8, 179, 70, 151, 62, 61] und sind zum Teil durch Anschauungen im Frequenzraum motiviert. Werden die Ableitungen als Ableitungsfiler betrachtet, so können auch die differenziellen Methoden als Filtermethoden interpretiert werden [151]. Die Bewegungen lassen sich auch durch Filteroperationen im Frequenzraum bestimmen.

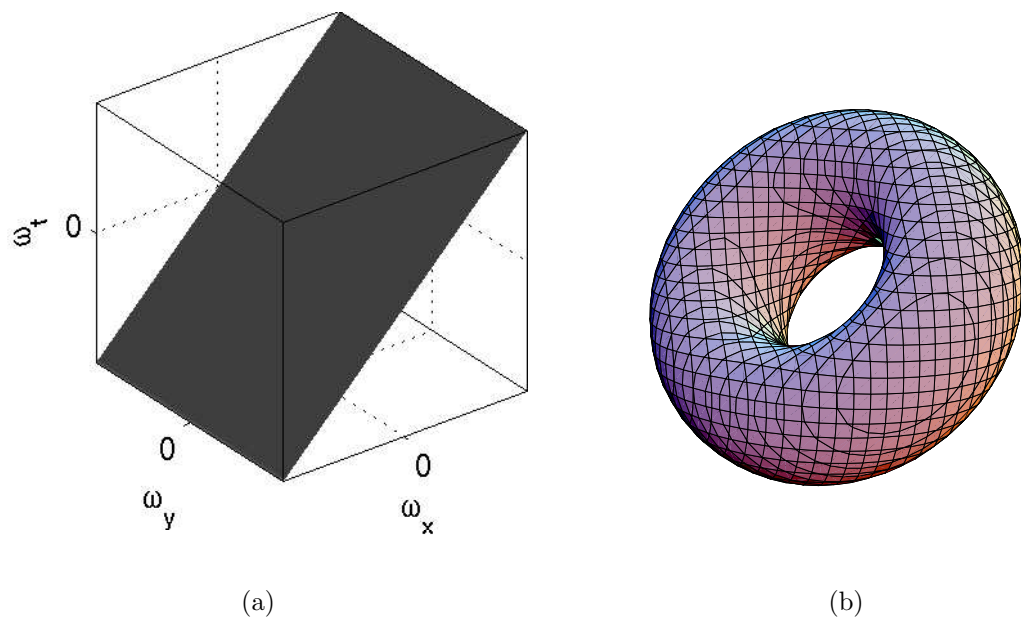


Abbildung 1.2: (a) zeigt die Ebene im Frequenzraum mit dem Normalenvektor $\mathbf{n} = (\mathbf{v}, 1)^T$ für eine Sequenz, die sich mit der Geschwindigkeit $\mathbf{v} = (-1, -1)^T$ bewegt. (b) zeigt einen von Simoncelli's [151] Donut-Filtern. Durch Rotation des Filters wird die Lage der Ebene bestimmt.

Wenden wir die dreidimensionale Fouriertransformation auf das Sequenzmodell (1.1) an, so erhalten wir [178]

$$f(\mathbf{x}, t) \circ - \bullet G(\boldsymbol{\omega}) \delta(\boldsymbol{\omega} \cdot \mathbf{v} + \omega_t), \quad (1.7)$$

wobei $G(\boldsymbol{\omega})$ die Fouriertransformierte des Bildes $g(\mathbf{x})$, $\boldsymbol{\omega} = (\omega_x, \omega_y)$ den Ortsfrequenzvektor und ω_t die Zeitfrequenz darstellen. Nach dieser Gleichung besetzt ein sich mit der Geschwindigkeit $\mathbf{v} = (v_x, v_y)^T$ bewegendes Objekt einen zweidimensionalen Unterraum des dreidimensionalen Fourierraumes. Dieser zweidimensionale Unterraum beschreibt eine Ebene durch den Ursprung mit einer Normalen proportional zu dem Vektor $\mathbf{n} = (v_x, v_y, 1)$. Daher bestimmen einige Verfahren die Bewegung, in dem sie mit geeigneten Filtern im Frequenzraum nach der Ebene suchen [151, 190, 187]. Abbildung 1.2(a) veranschaulicht die Ebenenstruktur für eine Sequenz, bei der sich alle Punkte mit der Geschwindigkeit $\mathbf{v} = (-1, -1)^T$ bewegen. Durch Rotation des von Simoncelli [151] vorgeschlagenen Donut-Filters (b) kann die Lage der Ebene und damit die Bewegung bestimmt werden. Mit solchen Filteroperationen lassen sich auch transparente Bewegungen ermitteln.

1.2 Überlagerte Bewegungen

Neben Beleuchtungsänderungen, Schatten und Reflexen wird die Annahme konstanter Bildintensitäten von transparenten Überlagerungen verletzt. Bei transparenten Bewegungen kreuzen sich die Bewegungstrajektorien, die Bildintensitäten überlagern sich und Gleichung (1.4) kann nicht mehr erfüllt werden, da eine Bewegung sich nicht zur Beschreibung mehrerer Bewegungen eignet. Abbildung 1.3 veranschaulicht die Überlagerung transpa-



Abbildung 1.3: Zeigt die Überlagerung der Bewegungstrajektorien anhand eines Schnittes durch die Monalisasequenz. Die weiße Linie in (a) markiert die Schnittebene (b) in der Sequenz. Die Überlagerung der Bewegungstrajektorien ist als Überlagerung zweier orientierter Muster zu erkennen.

renter Bewegungstrajektorien am Beispiel der Monalisasequenz.

Um transparente Bewegungen bestimmen zu können, benötigen wir zunächst ein adäquates Bildmodell. Aus diesem Bildmodell gilt es dann, ähnlich wie bei der optischen Flussgleichung (1.4) für eine Bewegung, entsprechende Bedingungsbedingungen für mehrere Bewegungen herzuleiten und Lösungsverfahren vorzustellen.

In der letzten Dekade entwickelte sich die Bewegungsbestimmung zu einem vielschichtigen Problem [177, 147, 148, 156, 157]. Die Bestimmung überlagerter Bewegungen teilt sich in zwei wesentliche Gruppen ein, die deutlich unterschieden werden müssen.

1.2.1 Transparente Überlagerung

Bei der ersten Gruppe handelt es sich um transparente Überlagerungen, die wir in der realen Welt weitaus häufiger beobachten, als man zunächst vermuten würde. Der Begriff *Transparenz* leitet sich aus dem lateinischen Wort *trans* für “durch” und *parere* für “scheinen” ab. Er bezeichnet in der Optik die Eigenschaft eines Körpers, Licht passieren zu lassen und damit ganz oder teilweise durchsichtig zu sein. Glas, Wasser und Luft sind wohl die bekanntesten Vertreter dieser Eigenschaft.

Transparente Überlagerungen sind die Folge von Reflexionen an Grenzflächen transparenter Objekte. An Fensterscheiben beobachten wir sie nahezu täglich. Ein Betrachter schaut von innen durch eine Fensterscheibe nach außen. Ist es außen dunkel, sieht er in der Scheibe nur ein inneres Bild als Spiegelung an der Scheibe. Ist es außen sehr viel heller als im Inneren, sieht er häufig nur das äußere Bild. Die Spiegelung des inne-

ren Bildes ist jedoch nicht verloren gegangen, sie wird für den menschlichen Betrachter nur von dem äußeren Bild überstrahlt. Bei geeigneten Lichtverhältnissen sieht er inneres und äußeres Bild überlagert. Beide Bilder sind demnach unabhängig voneinander und die Lichtstrahlen des inneren Bildes werden an der Scheibe reflektiert ohne dabei von den äußeren Lichtstrahlen beeinflusst zu werden oder sie zu beeinflussen. Dieser Umstand erlaubt uns die transparente Überlagerung f als additive Superposition des inneren Bildes g_1 und des äußeren g_2 gemäß

$$f(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x}) \quad (1.8)$$

darzustellen, wobei $\mathbf{x} = (x, y)^T$ den Ortsvektor in der Bildebene bezeichnet. Die Bilder $g_1(\mathbf{x})$ und $g_2(\mathbf{x})$ bezeichnen wir als Schichten.

Beim Auftreffen des Lichtes an einer transparenten Oberfläche wird meist ein Teil des Lichtes reflektiert und ein Teil transmittiert. Im Extremfall können dadurch, wenn z.B. das Licht des inneren Bildes das äußere beleuchtet, Abhängigkeiten entstehen, die wir im Folgenden jedoch nicht berücksichtigen wollen. Wir setzen ausreichend starke externe Lichtquellen voraus.

Im Bereich der Bildgebung sind Durchstrahlungsaufnahmen, wie z.B. in der medizinischen Röntgenbildgebung, bedeutend. Lichtstrahlen oder elektromagnetische Wellen, die einen Körper durchdringen, werden von ihm absorbiert. Der Grad der Absorption ist dabei in der Regel von der Dicke des Körpers, seiner Materialeigenschaften und der Wellenlänge abhängig. Nach dem Lambert-Beer'schen Gesetz ist die gemessene Intensität I nach dem Durchdringen des Körpers durch den Zusammenhang

$$I = I_0 \exp(-\mu(\lambda)d) \quad (1.9)$$

beschrieben, wobei d die Dicke des Körpers, $\mu(\lambda)$ seine Absorptionseigenschaft in Abhängigkeit von der Wellenlänge λ und I_0 die Eingangsintensität der Strahlung bezeichnet. Werden mehrere Schichten unterschiedlicher Dicken d_i und Materialeigenschaften $\mu_i(\lambda)$ durchlaufen bzw. durchstrahlt, so ergibt sich für die Absorption der multiplikative Zusammenhang

$$I = I_0 \prod_i \exp(-\mu_i(\lambda)d_i). \quad (1.10)$$

Im Gegensatz zu dem transparenten Phänomen ist dieses transluzent, d.h. lichtdurchlässig und nicht bilddurchlässig, und wir modellieren das Durchleuchten der hintereinander liegenden Schichten g_1 und g_2 verallgemeinert durch

$$f(\mathbf{x}) = g_1(\mathbf{x}) g_2(\mathbf{x}). \quad (1.11)$$

Durch einfaches Logarithmieren überführen wir das multiplikative Modell in das additive und erhalten,

$$\log f(\mathbf{x}) = \log g_1(\mathbf{x}) + \log g_2(\mathbf{x}). \quad (1.12)$$

Das additive Modell reicht damit zur Beschreibung beider Phänomene aus, eine weitere Unterscheidung ist nicht erforderlich.

Treten in den jeweiligen Schichten Bewegungen auf, so sprechen wir von transparenten

Bewegungen. Um die Angabe eines entsprechenden Bildmodells für transparente Bewegungen zu vereinfachen, machen wir hier die gleiche Annahme wie bei der optischen Flussgleichung und beschreiben die Bewegungen durch einfache Translationen. Bewegt sich die Schicht $g_i(\mathbf{x})$ mit der Geschwindigkeit \mathbf{v}_i , so modellieren wir die N transparenten Bewegungen durch das Sequenzmodell

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - \mathbf{v}_1 t) + g_2(\mathbf{x} - \mathbf{v}_2 t) + \cdots + g_N(\mathbf{x} - \mathbf{v}_N t). \quad (1.13)$$

Ähnlich wie bei den einfachen Bewegungen ist Gleichung (1.13) Ausgangspunkt für viele Ansätze zur Bestimmung transparenter Bewegungen.

Aufgrund der Linearitätseigenschaft der Fouriertransformation kommt mit jeder weiteren Schicht g_i eine weitere Ebene durch den Ursprung im Frequenzraum hinzu. Der Normalenvektor jeder Ebene entspricht dem Bewegungsvektor in homogenen Koordinaten. Einige Verfahren behandeln daher die Bestimmung transparenter Bewegungen als ein Ebenensuchproblem im Frequenzraum, die sie durch parametrisierbare Filter beschreiben [151, 21, 187, 189] (Vergleiche Abbildung 1.2(a)). Für die Berechnung der dreidimensionalen Fouriertransformation werden ausreichend große räumliche und zeitliche Abschnitte benötigt, in denen die Bewegungen lokal konstant sein müssen, um die Ebenenstruktur zu erhalten. Variationen im Bewegungsfeld werden mit zunehmender Ausschnittgröße wahrscheinlicher, und der Aufwand zur Suche der Ebenen ist hoch. Als Alternative zur Ebenensuche im dreidimensionalen Fourierraum kann, wie Milanfar [108, 109] zeigt, die Bewegungsbestimmung als ein Liniensuchproblem in den zweidimensionalen Fouriertransformierten zweier geeigneter Radonprojektionen der Bildsequenz behandelt werden. Vernon [174] zeigte, dass bei zwei transparenten Überlagerungen vier aufeinander folgende Bilder für die Bestimmung der Bewegungen ausreichen. Die Bilder werden einzeln (2D) Fouriertransformiert und dann ein Gleichungssystem für jede Frequenz aufgestellt. Aus der Lösung der Gleichungssysteme können in einem weiteren Verarbeitungsschritt die durch die Bewegungen induzierten Phasenverschiebungen ermittelt werden.

Pingault et. al [131] übertrugen Vernons Verfahren in den Ortsraum, um daraus eine Bedingungsgleichung für zwei transparente Bewegungen mit Hilfe einer Taylorreihenentwicklung herzuleiten. Zusätzlich beschreiben sie die Bewegungsfelder durch B-Spline Funktionen, die sie für Regularisierungszwecke nutzen.

Shizawa & Mase [147, 148] leiteten direkt aus obigem Bildmodell (1.13) eine optische Flussgleichung für eine beliebige Anzahl transparenter Bewegungen her. Diese Gleichung besteht aus einer Konkatenation von Richtungsableitungsoperatoren, die der einfachen optischen Flussgleichung ähnlich sind. Wie die Gleichung von Pingault et al. [131] weist sie ebenfalls Nichtlinearitäten in den Bewegungsvektorkomponenten auf, die Ordnung der Nichtlinearitäten fällt jedoch niedriger aus. Die Bewegungsbestimmung sowie das Lösen der Nichtlinearitäten führen Shizawa & Mase auf Eigenystemanalysen symmetrischer Tensoren zurück. Für den Fall von zwei Bewegungen stellten Toro et. al. [169] ein weiteres Lösungsverfahren für die erweiterte optische Flussgleichung von Shizawa & Mase [147, 148] vor. Ihr Lösungsansatz basiert auf dem EM-Verfahren und erlaubt die Bestimmung dicht besetzter Bewegungsfelder.

Bergen et. al. [22] hingegen bestimmen zwei transparente Bewegungen aus Differenzen dreier aufeinander folgender Bilder. Das iterative Verfahren ermittelt zunächst eine Bewegung, führt eine *nullende Operation* zur Entfernung der Intensitätsmuster, die dieser Bewegung zuzuordnen sind, durch, um anschließend in dem Resultat die zweite Bewegung zu bestimmen.

Im Rahmen dieser Arbeit werden wir differenzielle, Matching- und phasenbasierte Methoden für die Bestimmung einer beliebigen Anzahl transparenter Bewegungen entwickeln. Eine Limitierung auf zwei Überlagerungen, wie sie in vielen der obigen Ansätzen gemacht wurde, entfällt.

Neben der transparenten Überlagerung gibt es noch eine weitere Form der Überlagerung, die verdeckende Überlagerung.

1.2.2 Verdeckende Überlagerung

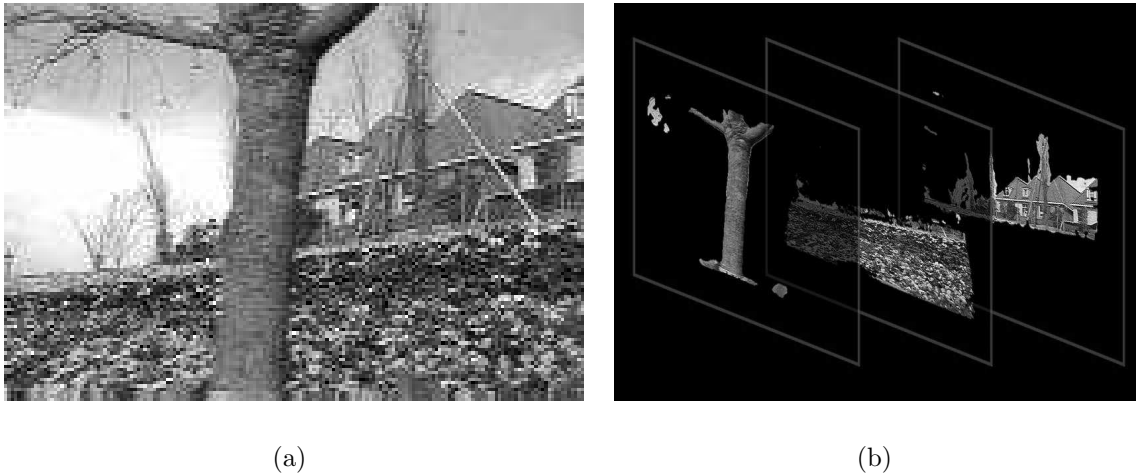
Die in natürlichen Sequenzen am häufigsten beobachtete Form der Überlagerung ist die verdeckende Überlagerung: Ein bewegtes Vordergrundobjekt verdeckt ein bewegtes Hintergrundobjekt. Da das Vordergrundobjekt opak ist, lässt es keine Lichtstrahlen des Hintergrundobjektes durch, und wir beobachten im Unterschied zu dem vorangegangenen Fällen nur eine Bewegung pro Bildpunkt. Insbesondere im Bereich der Videokomprimierung und der Bewegungsanalyse ist man an dem Auftrennen der Sequenz in Vorder- und Hintergrundobjekt interessiert. Ähnlich wie bei den transparenten Bewegungen nutzt man Schichtenmodelle zur Modellierung verdeckender Bewegungen [177, 156, 176]. Zusammenhängende Bereiche ähnlicher Bewegungen werden zu Schichten zusammengefasst und als Repräsentation bewegter Objekte genutzt. So ermitteln Wang & Adelson [177] zum Beispiel vor der eigentlichen Einteilung der Bildsequenz in opaken Schichten zunächst alle einfachen Bewegungen mit einem Kleinstequadrant-Ansatz. Anschließend teilen sie mit einem k-means-clustering-Verfahren die Sequenz in Regionen unterschiedlicher affiner Bewegungen ein, die dann die Schichten definieren. Abbildung 1.4 zeigt zur Veranschaulichung eines der Ergebnisse von Wang & Adelson.

Assoziieren wir ein Vordergrundobjekt, das sich mit der Geschwindigkeit \mathbf{u} bewegt, mit der Schicht g_1 und ein sich mit der Geschwindigkeit \mathbf{v} bewegendes Hintergrundobjekt mit der Schicht g_2 , so können wir die Verdeckung von Vorder- und Hintergrundobjekt durch

$$f(\mathbf{x}) = g_1(\mathbf{x} - \mathbf{u})\chi(\mathbf{x} - \mathbf{u}) + (1 - \chi(\mathbf{x} - \mathbf{u}))g_2(\mathbf{x} - \mathbf{v}) \quad (1.14)$$

modellieren [64]. Die charakteristische Funktion $\chi(\mathbf{x})$ wirkt dabei wie ein Schalter: An Punkten, in denen nur das Vordergrundobjekt sichtbar ist, hat sie den Wert 1 und dort, wo nur das Hintergrundobjekt sichtbar ist, den Wert null.

Dennoch gestaltete sich die Bewegungsbestimmung in Bereichen nahe der Verdeckungsgrenze als schwierig. Am Grenzübergang der sichtbaren Objektabschnitte ist die Bewegungstrajektorie des Hintergrundobjektes unterbrochen während unmittelbar daneben die des Vordergrundobjektes fortgesetzt wird. Damit gilt an der Verdeckungsgrenze, dem Grenzübergang beider Objekte, die optische Flussgleichung nicht und in der Umgebung



(a)

(b)

Abbildung 1.4: Veranschaulichung der Sequenzdarstellung durch Schichten: a) ein Bild der Flower-garden-Sequenz und b) Ergebnis der Schichteneinteilung mit dem Verfahren von Wang & Adelson [177] (Bilder kommen von <http://www-bcs.mit.edu/people/jyawang/demos/garden-layer/layer-demo.html>).

solcher Grenzübergänge werden zwei Bewegungen beobachtet. Mit einfachen Bewegungsbestimmungsverfahren sind auf Grund dieses *generalisierten Aperturproblems* die Bewegungen an der Verdeckungsgrenze nicht ermittelbar, da eine Bewegung nicht zur Beschreibung von zwei Bewegungen ausreicht.

Im Rahmen dieser Arbeit werden wir daher untersuchen, wie sich transparente Bewegungsbestimmungsverfahren bei der Anwendung auf verdeckende Bewegungen verhalten.

1.3 Leitfaden

In Kapitel 2 greifen wir zunächst die erweiterte optische Flussgleichung von Shizawa & Mase [147, 148] auf. Da diese Gleichung Nichtlinearitäten in den Bewegungsvektorkomponenten aufweist, entwickeln wir dort eine Linearisierungsmethode, die die Lösung der Nichtlinearitäten auf eine Nullstellenbestimmung eines komplexen Polynoms reduziert.

Aufbauend auf dieser linearisierten Gleichung stellen wir in Kapitel 3 ein lokales Bewegungsbestimmungsverfahren vor, das auf der Eigensystemanalyse von Tensoren basiert. Aus deren Invarianten leiten wir Konfidenzkriterien ab, mit denen wir feststellen können, ob das gerade betrachtete Bewegungsmodell der Anzahl der beobachteten Bewegungen in der Sequenz entspricht und damit die Angabe eines hierarchischen Verfahrens ermöglicht. Kapitel 2 und 3 enthalten Bestandteile meiner Diplomarbeit [158], die zu der Veröffentlichung [118] führte und von Cicero Mota und Erhardt Barth betreut wurde.

In Kapitel 4 entwickeln wir ein weiteres differenzielles Verfahren. Im Unterschied zu den vorangegangenen Verfahren bestimmen wir die Bewegungsvektoren durch Einführen der Nebenbedingung, dass die Bewegungsfelder glatt sein sollen. Die Glattheitsterme werden

indirekt als partielle Ableitungen der gemischten Geschwindigkeitsterme formuliert. Es ergibt sich dann ein lineares Differenzialgleichungssystem, das wir mit einem iterativen Verfahren lösen.

Differenzielle sowie viele andere Verfahren haben bei Beleuchtungsänderungen Schwierigkeiten, die Bewegungen zu ermitteln, da die zugrunde liegende Annahme konstanter Helligkeiten verletzt ist. Die erweiterte optische Flussgleichung von Shizawa & Mase [147, 148] basiert ebenfalls auf dieser Annahme und Beleuchtungsänderungen wirken sich auch hier negativ auf die Bewegungsbestimmung aus. In Kapitel 5 erweitern wir daher die transparente Flussgleichung um ein additives und ein multiplikatives Beleuchtungsmodell. Ferner werden wir dort zeigen, wie sich sowohl die Bewegungen als auch die Parameter der Beleuchtungsmodelle simultan mit einem Tensorverfahren berechnen lassen. Beide Verfahren sind direkt in den hierarchischen Algorithmus aus Kapitel 3 integrierbar.

Kapitel 6 verallgemeinert die phasenbasierte Methode von Vernon [174] auf eine beliebige Anzahl transparenter Bewegungen. Anschließend zeigen wir in Kapitel 7 kurz, wie die gewonnenen Phaseninformationen zur Trennung der bewegten Schichten genutzt werden können.

Die gewonnenen Erkenntnisse aus Kapitel 6 dienen in Kapitel 8 als Basis für die Herleitung einer Block-Matchingbedingung zur Bestimmung einer beliebigen Anzahl transparenter Bewegungen. Ähnlich wie bei den Tensorverfahren aus Abschnitt 3 entwerfen wir ein hierarchisches Bewegungsbestimmungsverfahren, bei dem das Konfidenzkriterium für ein bestimmtes Bewegungsmodell durch einen statistischen Signifikanztest realisiert ist. Anschließend entwerfen wir noch eine Methode, bei der die Glattheitsbedingungen, die wir zur Erhöhung der Robustheit an die Bewegungsfelder stellen möchten, durch ein Markov-Zufallsfeld realisiert wird. Glatte Bewegungsfelder sind wahrscheinlicher als nicht glatte.

In Kapitel 9 untersuchen wir, wie sich transparente Bewegungsbestimmungsverfahren bei der Ermittlung der Bewegungen an Verdeckungsgrenzen verhalten. Zu diesem Kapitel gibt es Vorarbeiten, die hier zusammengefasst [117, 18, 19] und um einen Aspekt ergänzt wurden.

Der Orientierungsanalyse in Bildern widmen wir in Kapitel 10 einen gesonderten Abschnitt, da sie etwas außerhalb der eigentlichen Thematik liegt, gleichzeitig aber für viele praktische Anwendungen hoch interessant ist. Dort stellen wir zwei Ansätze zur mehrfachen Orientierungsbestimmung vor und demonstrieren ihre Relevanz an mehreren realen und synthetischen Beispielen. Ferner zeigen wir für Doppelorientierungen, dass der Winkel zwischen den beiden Orientierungsvektoren eine Invariante darstellt, die sich besonders gut als Merkmal für Klassifikations- und Trackinganwendungen eignet.

Kapitel 2

Optischer Fluss für transparente Bewegungen und Orientierungen

Shizawa und Mase [147, 148] führten als erste eine optische Flussgleichung für zwei oder mehr transparent überlagerte Bewegungen ein. Ihre Flussgleichung basiert auf einer Hintereinanderausführung von Richtungsableitungsoperatoren, die im wesentlichen mit dem Richtungsableitungsoperator des optischen Flusses für eine Bewegung übereinstimmen. Ihr Konzept ist nicht nur auf die Bestimmung transparenter Bewegungen beschränkt, es lässt sich auch leicht auf die Bestimmung transparent überlagerter orientierter Strukturen im n -dimensionalen Raum übertragen, da die Bewegungsanalyse als ein Spezialfall der Orientierungsanalyse aufgefasst werden kann. Voraussetzung dafür ist allerdings, dass die Bewegungen in einer kleinen räumlichen und zeitlichen Umgebung des betrachteten Punktes konstant sind.

Die optische Flussgleichung weist jedoch eine Nichtlinearität auf, die, um die Bewegungs- bzw. Orientierungsvektoren zu erhalten, gelöst werden muss. Shizawa und Mase [147, 148] lösten die Nichtlinearität durch eine Eigensystembestimmung eines symmetrischen Tensors für den Fall von zwei Bewegungen/Orientierungen. Die Methode kann selbst dann zur Lösung der Nichtlinearität verwendet werden, wenn die betrachtete Struktur oder Sequenz mehr als drei Dimensionen besitzt. Allerdings ist unabhängig von der Dimension ihr Lösungsansatz auf zwei Überlagerungen beschränkt. In [148] skizzieren sie jedoch einen iterativen Algorithmus, mit dem sie die Nichtlinearität auch für mehr als drei Überlagerungen lösen können, gaben ihn jedoch auf Grund seiner komplizierten Form nicht konkret an und zeigten auch keine Beispielresultate mit ihm.

Im Rahmen dieser Arbeit ist es für den Fall von Bewegungen in Bildsequenzen gelungen, die Nichtlinearität auf ein Nullstellenproblem zu reduzieren [118] und somit für eine beliebige Anzahl transparenter Bewegungen zu lösen. Bei bis zu vier transparenten Bewegungen sind die Nullstellen durch geschlossene Formeln bestimmbar, ab fünf müssen sie durch numerische Verfahren ermittelt werden.

Im Folgenden werden wir die optische Flussgleichung für transparente Bewegungen herleiten und zeigen, wie wir die auftretende Nichtlinearität lösen können. Anschließend werden wir kurz auf den Aspekt der Orientierungsanalyse im Dreidimensionalen eingehen.

2.1 Optischer Fluss für mehrere Bewegungen

Um mehrere Bewegungen in einer Bildfolge zu bestimmen, wird zunächst eine Bedingung oder Gleichung benötigt, an Hand derer, ähnlich wie bei der optischen Flussgleichung, die Bewegungsparameter bestimmt werden können. In diesem Abschnitt soll die optische Flussgleichung für den einfachen Fall einer Bewegung auf den Fall mehrerer Bewegungen erweitert werden. Zunächst wird dies an einem Beispiel für zwei Bewegungen durchgeführt, die Erweiterung auf mehrere Bewegungen erhält man in analoger Weise. Eine Bildsequenz $f(\mathbf{x}, t) : \mathbb{R}^2 \times \mathbb{R} \rightarrow \mathbb{R}$ mit zwei additiven gleichförmigen Bewegungen modellieren wir durch die Formel

$$\begin{aligned} f(\mathbf{x}, t) &= \hat{g}_1(\mathbf{x}, t) + \hat{g}_2(\mathbf{x}, t) \\ &= g_1(\mathbf{x} - \mathbf{u}t) + g_2(\mathbf{x} - \mathbf{v}t), \end{aligned} \quad (2.1)$$

wobei die jeweiligen Teilsequenzen \hat{g}_1 und \hat{g}_2 durch die Verschiebung der Bilder g_1 mit der Geschwindigkeit $\mathbf{u} = (u_x, u_y)^T$ und g_2 mit der Geschwindigkeit $\mathbf{v} = (v_x, v_y)^T$ modelliert sind. Für die weitere Betrachtung soll vorausgesetzt werden, dass die Geschwindigkeiten \mathbf{u} und \mathbf{v} nicht identisch und unabhängig von der Zeit sind. Zunächst definieren wir den Ableitungsoperator α in Abhängigkeit von der Geschwindigkeit

$$\alpha(\mathbf{v}) = v_x \frac{\partial}{\partial x} + v_y \frac{\partial}{\partial y} + \frac{\partial}{\partial t}. \quad (2.2)$$

Wenden wir diesen auf eine der Teilsequenzen (z.B. g_1) an, d.h. [84]

$$\alpha(\mathbf{u})g_1 = \left(u_x \frac{\partial}{\partial x} + u_y \frac{\partial}{\partial y} + \frac{\partial}{\partial t} \right) g_1 = 0, \quad (2.3)$$

so erhalten wir die optische Flussgleichung. Nach dieser Gleichung entfernt der Operator α alle Anteile in Richtung der Bewegung, d.h. $\mathbf{u} = (u_x, u_y)^T$ und wir bezeichnen ihn daher als einen nullenden Operator.

Kommen additiv überlagerte Bewegungen in der Bildsequenz vor, muss mehr als diese eine Bedingung in jedem Punkt definiert werden. Wir benötigen daher ein lokales Maß, das robust gegenüber mehrfachen Bewegungen in einem Punkt ist. Für Bildsequenzen, die additiv kombiniert werden, erweitern wir unsere Bedingung durch eine Verkettung des Operators für die einzelnen Geschwindigkeiten. In unserem Beispiel für zwei Geschwindigkeiten sieht dies wie folgt aus

$$\begin{aligned} \alpha(\mathbf{u})\alpha(\mathbf{v})f &= \alpha(\mathbf{u})\alpha(\mathbf{v})g_1 + \alpha(\mathbf{u})\alpha(\mathbf{v})g_2 \\ &= \alpha(\mathbf{v})\alpha(\mathbf{u})g_1 + \alpha(\mathbf{u})\alpha(\mathbf{v})g_2 \\ &\stackrel{!}{=} 0. \end{aligned} \quad (2.4)$$

Insbesondere gilt in diesem Fall, wenn die wahren Geschwindigkeiten \mathbf{u} und \mathbf{v} eingesetzt werden, dass $\alpha(\mathbf{u})g_1 = 0$ und $\alpha(\mathbf{v})g_2 = 0$ sind. Des weiteren gilt für den Operator α unter der Annahme, dass die Bewegungen lokal konstant sind, das Kommutativgesetz $\alpha(\mathbf{u})\alpha(\mathbf{v}) = \alpha(\mathbf{v})\alpha(\mathbf{u})$ und dass der Ausdruck null wird, wenn die richtigen Geschwindigkeiten gefunden wurden.

Die Operatortechnik birgt einen Vorteil in sich. Die optische Flussgleichung kann sehr einfach in den Frequenzbereich übertragen werden, ohne dass dabei Faltungen auftreten. Damit kann die Analyse sowohl im Orts-Zeit-Bereich als auch im Frequenzbereich leicht durchgeführt werden. Die Transformierte des obigen Beispiels lautet

$$\begin{aligned}\hat{\alpha}(\mathbf{u})\hat{\alpha}(\mathbf{v})F(\boldsymbol{\omega}, \omega_t) &= (\omega_x u_x + \omega_y u_y + \omega_t)(\omega_x v_x + \omega_y v_y + \omega_t)F(\boldsymbol{\omega}, \omega_t) \\ &= 0,\end{aligned}\tag{2.5}$$

mit $\hat{\alpha}(\mathbf{v}) = (\omega_x v_x + \omega_y v_y + \omega_t)$, wobei $\boldsymbol{\omega} = (\omega_x, \omega_y)^T$ der Ortsfrequenzvektor ist und ω_t der Zeitfrequenz entspricht, in denen der Faktor 2π bereits zur Vereinfachung der Schreibweise mit aufgenommen ist und im Folgenden daher nicht weiter explizit angegeben werden muss. Der große Buchstabe F bezeichnet dabei die Fouriertransformierte der Bildsequenz f . Nach dem einführenden Beispiel eines Operators für zwei Bewegungen gilt es, diesen auf eine beliebige Anzahl von additiv überlagerten Bewegungen zu erweitern. Die verallgemeinerte Formel für eine Sequenz, die sich aus N additiv überlagerten gleichförmig bewegten Teilsequenzen zusammensetzt, lautet

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - \mathbf{v}_1 t) + \dots + g_N(\mathbf{x} - \mathbf{v}_N t),\tag{2.6}$$

wobei jedes g_i N mal differenzierbar sein muss und mit $\mathbf{v}_i = (v_{ix}, v_{iy})^T$ die Geschwindigkeitsvektoren bezeichnet werden. Sollten zwei Geschwindigkeitsvektoren identisch sein, so reduziert sich das Modell auf $(N - 1)$ Bewegungen. In dem anschließenden Satz wird der Operator zur Bewegungsbestimmung auf beliebige N ausgedehnt.

Satz 1 *Gegeben sei eine Bildsequenz $f = g_1 + g_2 + \dots + g_N$, die aus N additiv überlagerten einzelnen Bildsequenzen besteht und mindestens N mal differenzierbar ist. Zur Bestimmung der N Geschwindigkeiten in dieser Sequenz müssen wir folgende Gleichung lösen*

$$\alpha(\mathbf{v}_1)\alpha(\mathbf{v}_2)\cdots\alpha(\mathbf{v}_N)f = 0\tag{2.7}$$

oder im Frequenzbereich

$$\hat{\alpha}(\mathbf{v}_1)\hat{\alpha}(\mathbf{v}_2)\cdots\hat{\alpha}(\mathbf{v}_N)F = 0.\tag{2.8}$$

Beweis

Wenden wir den Operator $\alpha(\mathbf{v}_1)\alpha(\mathbf{v}_2)\cdots\alpha(\mathbf{v}_N)$ auf die Bildsequenz $f = g_1 + g_2 + \dots + g_N$ an, so erhalten wir nach dem Ausmultiplizieren den Ausdruck

$$\alpha(\mathbf{v}_1)\alpha(\mathbf{v}_2)\cdots\alpha(\mathbf{v}_N)f = \alpha(\mathbf{v}_1)\cdots\alpha(\mathbf{v}_N)g_1 + \dots + \alpha(\mathbf{v}_1)\cdots\alpha(\mathbf{v}_N)g_N.\tag{2.9}$$

Aus der vorangegangenen Diskussion ist bekannt, dass $\alpha(\mathbf{v}_i)g_i = 0$. Nach Ausnutzung des Kommutativgesetzes für den Operator α für lokal konstante Bewegungen können die Teilausdrücke wie folgt umgestaltet werden

$$\alpha(\mathbf{v}_1)\cdots\alpha(\mathbf{v}_{i-1})\alpha(\mathbf{v}_{i+1})\cdots\alpha(\mathbf{v}_N)\underbrace{\alpha(\mathbf{v}_i)g_i}_{=0} = 0.\tag{2.10}$$

Die Herleitung der Formel für den Frequenzbereich ergibt sich durch die Transformaten des Ableitungsoperators α . Die Transformierte des Ableitungsoperators lautet

$$\hat{\alpha}(\mathbf{v}) = (\omega_x v_x + \omega_y v_y + \omega_t). \quad (2.11)$$

□

Gleichung (2.7) ist eine notwendige Bedingung für die Bestimmung von N überlagerten Bewegungen in einer Bildsequenz. Unter der Annahme, dass Gleichung (2.7) an jedem Punkt der Bildsequenz gilt, stellt sie zusätzlich eine hinreichende Bedingung dar: Mit dieser Annahme kann die Sequenz fouriertransformiert werden und jeder Ausdruck der Form $\hat{\alpha}(\mathbf{v}) = (\omega_x v_x + \omega_y v_y + \omega_t) = 0$ beschreibt eine Ebene im Frequenzraum. Da Gleichung (2.7) im Ort gilt, muss auch Gleichung (2.8) im Frequenzraum erfüllt sein und $F(\boldsymbol{\omega})$ kann nur an Frequenzen $\boldsymbol{\omega}$ Werte ungleich null haben, die in einer der Ebenen $\hat{\alpha}(\mathbf{v}_i) = 0$, $i = 1, \dots, N$ liegen. Betrachten wir jede Ebene als die Fouriertransformierte einer der bewegten Schichten, so existiert für $f(\mathbf{x}, t)$ eine Darstellung in der Form $f(\mathbf{x}, t) = g_1(\mathbf{x} - \mathbf{v}_1 t) + \dots + g_N(\mathbf{x} - \mathbf{v}_N t)$.

Da die erweiterte optische Flussgleichung (2.7) Nichtlinearitäten in den Bewegungsvektorkomponenten aufweist, werden wir sie im ersten Schritt in eine linearisierte Darstellung überführen und in dieser lösen. Die Lösung nutzen wir dann in einem zweiten Schritt, um die eigentliche Nichtlinearität aufzulösen und damit die zugrunde liegenden Bewegungsvektoren zu bestimmen. Wie wir noch zeigen werden, verlieren wir mit der Linearisierung die hinreichende Bedingung und in der linearisierten Darstellung sind Lösungen möglich, die keine Bewegungen repräsentieren. Betroffen davon sind insbesondere Fälle, in denen Aperturprobleme in einer oder mehreren Schichten auftreten oder Datensätze $f(\mathbf{x}, t)$, die nichts mit Bewegungen gemein haben, deren Spektren aber besondere Formen aufweisen.

Für zwei überlagerte Bewegungen leiten wir ein Entscheidungskriterium her, mit dem wir die gefundene Lösung auf Konsistenz mit der Bewegungsannahme überprüfen können, und damit gegebenenfalls eine Lösung ausschließen können.

Bisher sind wir davon ausgegangen, dass die Bildsequenz f N -mal differenzierbar ist. Srinivasan [155] zeigt, dass die Bewegungsparameter durch die Anwendung von linearen Filtern auch ohne diese Bedingung bestimmt werden können. Er benutzt dazu Faltungen mit glatten Funktionen h und wendet sie auf beide Seiten der optischen Flussgleichung an, um die Anzahl der Gleichungen zu erhöhen. Für den Fall einer Bewegung erhält man

$$(\alpha(\mathbf{v})f) * h = f * (\alpha(\mathbf{v})h) = 0 \quad (2.12)$$

und ausgeschrieben lautet dies

$$v_x f * h_x + v_y f * h_y + f * h_t = 0. \quad (2.13)$$

An dieser Gleichung kann nun auf die Differenzierbarkeit von f verzichtet und trotzdem noch die Geschwindigkeit \mathbf{v} bestimmt werden, Dazu betrachten wir noch einmal das folgende Faltungsintegral

$$f * h(\mathbf{x}, t) = \int f(\mathbf{y} - \mathbf{v}r)h(\mathbf{x} - \mathbf{y}, t - r)d\mathbf{y}dr. \quad (2.14)$$

Nach Substitution mit den Variablen $\mathbf{z} = \mathbf{y} - \mathbf{v}r$ und $s = t - r$ erhalten wir

$$f * h(\mathbf{x}, t) = \int f(\mathbf{z})h(\mathbf{x} - \mathbf{v}t - \mathbf{z} + s\mathbf{v}, s)dzds. \quad (2.15)$$

Wird nun unter dem Integralzeichen differenziert, so ist einfach zu erkennen, dass $h * f$ differenzierbar ist, da h differenzierbar ist und dass $g * f$ die optische Flussgleichung erfüllt, denn

$$f * \frac{\partial}{\partial t}h = -v_x f * h_x - v_y f * h_y. \quad (2.16)$$

Mit dieser Erkenntnis kann auf die Differenzierbarkeit von f verzichtet werden und wir können folgenden Satz bilden:

Satz 2 *Sei L ein ortsinvarianter linearer Filter mit einer Impulsantwort h , die N mal differenzierbar ist, dann ist $L(f)$ N mal differenzierbar und es gilt*

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N)L(f) = 0. \quad (2.17)$$

Beweis

Der Beweis kann auf ähnliche Weise durchgeführt werden wie der vorangegangene. $L(f)$ kann als $L(f) = g_1 * h + g_2 * h + \cdots + g_N * h$ geschrieben werden. Aus der vorangegangenen Diskussion für eine Bewegung wissen wir, dass $\alpha(\mathbf{v}_i)(g_i * h) = 0$ ist. Aus diesem Grund sind die $\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N)(g_i * h) = 0$ für alle $i = 1, \dots, N$.

□

Mit diesem Satz haben wir nun gezeigt, dass für jeden ortsinvarianten linearen Operator die Geschwindigkeitsparameter für f und $L(f)$ gleich sind. Dies ist insbesondere dann interessant, wenn es sich bei h um einen Tiefpassfilter handelt, der hochfrequente Anteile in der Bildsequenz, die in der Regel durch Rauschen entstehen, entfernt und daher die Ableitungen robuster gegen Rauschen macht. Hochfrequente Anteile in den bewegten Strukturen gehen dann allerdings auch verloren.

2.1.1 Linearisierung des optischen Flusses

Die Flussgleichungen (2.7) und (2.17) weisen Nichtlinearitäten in Abhängigkeit der Bewegungsvektorkomponenten auf. Daher werden wir in diesem Abschnitt die Flussgleichungen so umformulieren, dass sie als inneres Produkt zweier Vektoren darstellbar sind und somit der gesuchte Parametervektor durch Lösen eines linearen Gleichungssystems bestimmbar wird. Die Umformulierung löst jedoch nicht die Nichtlinearität, sondern verschiebt sie nur auf einen späteren Zeitpunkt.

Betrachten wir zunächst wieder ein Beispiel für zwei additive Bewegungen

$$\begin{aligned} \alpha(\mathbf{u})\alpha(\mathbf{v})f &= u_x v_x f_{xx} + u_y v_y f_{yy} \\ &+ (u_x v_y + u_y v_x) f_{xy} \\ &+ (u_x + v_x) f_{xt} + (u_y + v_y) f_{yt} + \\ &+ f_{tt} \\ &= 0. \end{aligned} \quad (2.18)$$

Ziel ist es, diese Gleichung durch ein System zu lösen. Der gesuchte Lösungsvektor besteht aus den Vorfaktoren der partiellen Ableitungen und es ist daher sinnvoll, die Summe als inneres Produkt eines Vektors, der die partiellen Ableitungen der Bildfolge f enthält, mit einem Vektor, der aus Produkten und Summen der Geschwindigkeitskomponenten besteht, umzuschreiben. Im ersten Schritt werden die folgenden Hilfsgrößen eingeführt

$$\begin{aligned}
c_{xx} &= u_x v_x \\
c_{yy} &= u_y v_y \\
c_{xy} &= (u_x v_y + u_y v_x) \\
c_{xt} &= (u_x + v_x) \\
c_{yt} &= (u_y + v_y) \\
c_{tt} &= 1.
\end{aligned} \tag{2.19}$$

Die Indizierung ist dabei so gewählt, dass die Indizes der c 's den Indizes der partiellen Ableitungen entsprechen. Damit ist z.B. c_{xx} der Vorfaktor, der zu der partiellen Ableitung f_{xx} gehört, und c_{xy} der Vorfaktor, der zu der partiellen Ableitung f_{xy} gehört. Mit Hilfe dieser Größen wird die Summe in die Form

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f = \sum_I c_I f_I = 0 \tag{2.20}$$

überführt, wobei I aus den geordneten Sequenzen

$$I = (xx, yy, xy, xt, yt, tt) \tag{2.21}$$

besteht. Die Ordnung soll verhindern, dass das System unnötig vergrößert wird, denn für die partiellen Ableitungen gilt (meist) das Kommutativgesetz und es spielt daher keine Rolle, nach welcher Variablen zuerst abgeleitet wird. Nach der Sequenz xy ist die Sequenz yx nicht mehr nötig und kann ausgeschlossen bleiben. Die Summe wird als inneres Produkt der Vektoren $\mathbf{d}f$ und \mathbf{c} dargestellt (d.h. $\mathbf{d}f \cdot \mathbf{c} = 0$), wobei

$$\begin{aligned}
\mathbf{d}f &= (f_{xx}, f_{yy}, f_{xy}, f_{xt}, f_{yt}, f_{tt})^T = (f_I)_I \\
\mathbf{c} &= (c_{xx}, c_{yy}, c_{xy}, c_{xt}, c_{yt}, c_{tt})^T = (c_I)_I.
\end{aligned} \tag{2.22}$$

Nach dem einführenden Beispiel für zwei Bewegungen soll diese Vorgehensweise auf eine beliebige Anzahl von additiven Bewegungen erweitert werden. Gleichung (2.17) für N Bewegungen wird durch die Summe

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f * h = \sum_I c_I f * h_I \equiv 0 \tag{2.23}$$

beschrieben, wobei $I = (I_1, I_2, \dots, I_m)$ aus geordneten Sequenzen der Länge N besteht, die sich aus den Elementen $i_j \in \{x, y, t\}$ zusammensetzen. Als Konvention legen wir fest, dass $I_m = t \cdots t$ immer nur aus einer Abfolge des Buchstaben t besteht und damit die letzte Komponente des Vektors \mathbf{c} den Wert $\mathbf{c}_{I_m} = c_{t \cdots t} = 1$ besitzt. Die h_I 's sind die partiellen Ableitungen der linearen Impulsantwort h . Die vermischten Bewegungsparameter c_I sind homogene und symmetrische Funktionen, die von den Geschwindigkeitsvektoren abhängen und sich leicht aus dem symbolischen Ausdruck $\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N)$ extrahieren lassen. Wird

konstante Bewegung vorausgesetzt, so ist Gleichung (2.23) für jede Funktion h gültig. Aus diesem Grund kann eine Menge von l geeignet gewählten Funktionen h_k verwendet werden, um die Anzahl der Gleichungen, wie Srinivasan [155] für eine Bewegung zeigt, zu erhöhen. Damit erhält man das Gleichungssystem

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f * h_k = \sum_I c_I f * h_{k,I} \equiv 0 \quad (2.24)$$

mit $k = 1, \dots, l$.

Genau wie in dem Beispiel für zwei Bewegungen kann diese Summe als ein inneres Produkt zwischen zwei Vektoren ausgedrückt werden

$$\mathbf{L}(\mathbf{x}) \cdot \mathbf{c} = 0 \quad (2.25)$$

mit $\mathbf{L}(\mathbf{x}) = [(f * h_{k,I})_I(\mathbf{x})]^T$ und $\mathbf{c} = (c_I)_I^T$. Der Vektor \mathbf{c} ist eine indirekte Darstellung der Bewegungen und wir bezeichnen ihn daher als den *gemischten Geschwindigkeitsparametervektor*. Im Folgenden soll nun gezeigt werden, wie wir aus einem gegebenen \mathbf{c} auf die Bewegungsvektoren schließen können.

2.1.2 Separierung der Geschwindigkeitsvektoren

Nehmen wir an, ein Algorithmus liefere uns als Ergebnis den gemischten Geschwindigkeitsvektor \mathbf{c} . Es stellt sich nun die Frage, wie die Geschwindigkeitsvektoren \mathbf{v}_i aus dem Vektor \mathbf{c} extrahiert werden. Zunächst sei die Vorgehensweise zur Lösung dieser Fragestellung anhand eines Beispiels für zwei Bewegungen durchgeführt. Der Vektor \mathbf{c} (2.23) setzt sich in diesem Fall aus den Hilfsgrößen c_I zusammen

$$\mathbf{c} = (c_{xx}, c_{yy}, c_{xy}, c_{xt}, c_{yt}, c_{tt})^T. \quad (2.26)$$

Die Hilfsgrößen für den Fall von zwei Bewegungen lauten nach Formel (2.23)

$$\begin{aligned} c_{xx} &= u_x v_x \\ c_{yy} &= u_y v_y \\ c_{xy} &= (u_x v_y + u_y v_x) \\ c_{xt} &= (u_x + v_x) \\ c_{yt} &= (u_y + v_y) \\ c_{tt} &= 1. \end{aligned} \quad (2.27)$$

Die c 's weisen jedoch Nichtlinearitäten in Abhängigkeit der Variablen \mathbf{u} und \mathbf{v} auf und können daher nicht durch ein lineares Gleichungssystem gelöst werden. Die Geschwindigkeitsvektoren \mathbf{u} und \mathbf{v} seien als Vektoren in der komplexen Zahlenebene betrachtet

$$\mathbf{u} = u_x + iu_y, \quad \mathbf{v} = v_x + iv_y. \quad (2.28)$$

Zwischen diesen komplexen Zahlen und den c 's bestehen die Beziehungen

$$\begin{aligned} \mathbf{u} \mathbf{v} &= A_0 = c_{xx} - c_{yy} + ic_{xy} \\ \mathbf{u} + \mathbf{v} &= A_1 = c_{xt} - ic_{yt}. \end{aligned} \quad (2.29)$$

A_0 und A_1 sind die elementarsymmetrischen Funktionen über den Variablen \mathbf{u} und \mathbf{v} und nach dem Vietaschen Wurzelsatz die Koeffizienten des komplexen Polynoms

$$Q(\mathbf{z}) = (\mathbf{z} - \mathbf{u})(\mathbf{z} - \mathbf{v}) = \mathbf{z}^2 - A_1\mathbf{z} + A_0 \quad (2.30)$$

mit den komplexen Nullstellen \mathbf{u} und \mathbf{v} . Die A_i 's sind in den Variablen \mathbf{u} und \mathbf{v} symmetrisch und homogen vom Homogenitätsgrad¹ kleiner gleich zwei.

In der Verallgemeinerung für N Bewegungen werden die Komponenten v_{ix} und v_{iy} in N komplexen Zahlen

$$\mathbf{v}_i = v_{ix} + i v_{iy} \quad (2.31)$$

zusammengefasst. Aus ihnen lassen sich die N elementarsymmetrischen Funktionen

$$\begin{aligned} A_{n-1}(c_I) &= \sum_i \mathbf{v}_i \\ A_{n-2}(c_I) &= \sum_i \mathbf{v}_i \sum_{i<j} \mathbf{v}_j \\ A_{n-3}(c_I) &= \sum_i \mathbf{v}_i \sum_{i<j} \mathbf{v}_j \sum_{k<j} \mathbf{v}_k \\ &\vdots \\ A_0(c_I) &= \mathbf{v}_1 \cdots \mathbf{v}_N \end{aligned} \quad (2.32)$$

vom Homogenitätsgrad $\leq N$ bilden, die wiederum durch die c_I 's ausgedrückt werden. Die N Geschwindigkeiten \mathbf{v}_i sind dann die Nullstellen des komplexen Polynoms

$$Q(z) = z^n - A_{N-1}(c_I)z^{N-1} + \cdots + (-1)^N A_0(c_I) \quad (2.33)$$

Beispielsweise lauten die A_i 's für drei Bewegungen

$$\begin{aligned} A_2 &= c_{xtt} + i c_{ytt} \\ A_1 &= c_{xxt} - c_{yyt} + i c_{xyt} \\ A_0 &= c_{xxx} - c_{xyy} + i(c_{xxy} - c_{yyy}) \end{aligned} \quad (2.34)$$

Mit Gleichung (2.33) lassen sich bis zu vier Bewegungen exakt bestimmen, da für die Nullstellen eines Polynoms vom Grad 4 noch eine geschlossene Formel angegeben werden kann. Für Polynome vom höheren Grad als vier sind keine geschlossenen Formeln für die Berechnung der Nullstellen bekannt. In diesen Fällen muss daher auf einen iterativen Algorithmus zurückgegriffen werden. Ein mögliches Verfahren wäre z.B. dasjenige von *Newton-Raphson*.

¹Eine Funktion mit n unabhängigen Variablen nennt man homogen vom Grade m , wenn für die Multiplikation aller Variablen mit dem Wert λ gilt: $f(\lambda x_1, \dots, \lambda x_n) = \lambda^m f(x_1, \dots, x_n)$.

2.1.3 Überlagerte Orientierungen

Die Bewegungsanalyse kann auch als ein Spezialfall der Orientierungsanalyse aufgefasst werden, bei dem der zeitliche Orientierungsanteil immer eins ergibt. Im Dreidimensionalen heißt eine Funktion $f(\mathbf{x})$ mit $\hat{\mathbf{x}} = (x, y, z)^T$ in einer kleinen Umgebung Ω ideal orientiert wenn

$$f(\hat{\mathbf{x}}) = f(\hat{\mathbf{x}} + \mathbf{v}) \quad (2.35)$$

für alle $\hat{\mathbf{x}}, \hat{\mathbf{x}} + \lambda \mathbf{v} \in \Omega, \lambda \in \mathbb{R}$ ist. Der Vektor $\mathbf{v} = (v_x, v_y, v_z)^T$ gibt die Richtung der Orientierung an.

Um auch Orientierungen behandeln zu können, schreiben wir den Richtungsableitungsoperator α jetzt als

$$\alpha(\mathbf{v}) = v_x \partial_x + v_y \partial_y + v_z \partial_z \quad (2.36)$$

wobei wir im Vergleich zu vorher die indirekte Darstellung des Geschwindigkeitsvektor $\mathbf{v} = (v_x, v_y, 1)^T$ durch den Orientierungsvektor $\mathbf{v} = (v_x, v_y, v_z)^T$ ersetzt haben. Schnell wird deutlich, dass die Bewegungsanalyse ein Sonderfall der Orientierungsanalyse ist, bei der die Komponente v_z eins ergibt. Daher gilt in Analogie zum optischen Fluss für eine Bewegung

$$\alpha(\mathbf{v})f = v_x f_x + v_y f_y + v_z f_z = 0. \quad (2.37)$$

Besteht $f(\hat{\mathbf{x}})$ aus einer additiven Überlagerung zweier Muster $g_1(\hat{\mathbf{x}})$ und $g_2(\hat{\mathbf{x}})$, die in Richtung $\mathbf{u} = (u_x, u_y, u_t)$ bzw. $\mathbf{v} = (v_x, v_y, v_z)$ orientiert sind, so gilt $\alpha(\mathbf{u})g_1 = 0$ und $\alpha(\mathbf{v})g_2 = 0$. Folglich ist

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f = 0. \quad (2.38)$$

Bis zu diesem Punkt scheint es keinen nennenswerten Unterschied zwischen der Bewegungs- und Orientierungsproblematik zu geben. Offensichtlich werden diese Unterschiede erst, wenn wir Gleichung (2.38) durch die Summe

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f = \sum_I c_I f_I = 0 \quad (2.39)$$

mit den *gemischten Orientierungsparametern*

$$\begin{aligned} c_{xx} &= u_x v_x \\ c_{yy} &= u_y v_y \\ c_{xy} &= (u_x v_y + u_y v_x) \\ c_{xz} &= (u_x v_z + v_x u_z) \\ c_{yz} &= (u_y v_z + v_y u_z) \\ c_{zz} &= u_z v_z. \end{aligned} \quad (2.40)$$

darstellen. Bei den gemischten Geschwindigkeitsparametern (2.23) handelt es sich um einen Spezialfall der gemischten Orientierungsparameter aus Gleichung (2.40), in der $u_z = v_z = 1$ ist. Aufgrund der zusätzlichen Komponenten u_z und v_z lässt sich das im vorherigen Abschnitt hergeleitete Verfahren zur Separierung der Bewegungsvektoren nicht für die Separierung der Orientierungsvektoren anwenden.

Shizawa & Mase [147, 148] ermittelten die Orientierungsvektoren, in dem sie die gemischten Orientierungsparameter in der Matrix

$$\mathbf{C} = \frac{1}{2} (\mathbf{u} \otimes \mathbf{v} + \mathbf{v} \otimes \mathbf{u}) = \begin{pmatrix} c_{xx} & \frac{1}{2}c_{xy} & \frac{1}{2}c_{xz} \\ \frac{1}{2}c_{xy} & c_{yy} & \frac{1}{2}c_{yz} \\ \frac{1}{2}c_{xz} & \frac{1}{2}c_{yz} & c_{zz} \end{pmatrix} \quad (2.41)$$

anordneten. Sie zeigten, dass die Matrix \mathbf{C} nur zwei von null verschiedene Eigenwerte λ_1 und λ_2 besitzt, von denen einer kleiner null und der andere größer null ist. Die gesuchten Orientierungsvektoren bestimmten sie anhand der Formeln

$$\begin{aligned} \mathbf{u} &= \sqrt{\frac{\lambda_1}{\lambda_1 - \lambda_2}} \mathbf{e}_1 + \sqrt{\frac{-\lambda_2}{\lambda_1 - \lambda_2}} \mathbf{e}_2 \\ \mathbf{v} &= \sqrt{\frac{\lambda_1}{\lambda_1 - \lambda_2}} \mathbf{e}_1 - \sqrt{\frac{-\lambda_2}{\lambda_1 - \lambda_2}} \mathbf{e}_2, \end{aligned} \quad (2.42)$$

wobei $\lambda_2 < 0$ und $\mathbf{e}_1, \mathbf{e}_2$ die zu den Eigenwerten korrespondierenden Eigenvektoren der Matrix \mathbf{C} sind. Eine Herleitung der Gleichungen (2.42) findet sich im Anhang B. Ihre Methode eignet sich auch zur Trennung der Bewegungsvektoren aus den gemischten Bewegungsparametern, da die Bewegungsvektoren Orientierungsvektoren entsprechen, bei denen die letzte (zeitliche) Komponente eins ergibt. Entsprechend erhalten wir die Bewegungsvektoren, in dem wir die Orientierungsvektoren jeweils durch ihre letzte Komponente teilen, d.h.

$$\mathbf{u} = \left(\frac{u_x}{u_z}, \frac{u_y}{u_z}, 1 \right)^T, \quad \mathbf{v} = \left(\frac{v_x}{v_z}, \frac{v_y}{v_z}, 1 \right)^T. \quad (2.43)$$

Das Verfahren von Shizawa und Mase gilt auch für die Orientierungsanalyse in höherdimensionalen Räumen, eignet sich aber auch dort nur zur Trennung von zwei Orientierungsvektoren. Das in dem vorangegangenen Abschnitt vorgestellte Verfahren zur Trennung der Bewegungsvektoren gilt nur im Dreidimensionalen, erlaubt aber die Trennung von einer beliebigen Anzahl von überlagerten Bewegungen.

2.2 Konsistenz der Lösung mit den Bewegungen bzw. Orientierungen

Haben wir eine Lösung \mathbf{s} für den gemischten Orientierungsparametervektor gefunden, die die erweiterte optische Flussgleichung

$$\sum_I s_I f_I = 0 \quad (2.44)$$

erfüllt, so bedeutet dies nicht unbedingt, dass die Lösung die Orientierungen bzw. Bewegungen repräsentiert. Für den Fall von zwei Überlagerungen leiten wir daher aus der Separierungsmethode von Shizawa & Mase [147, 148] ein Entscheidungskriterium ab, mit dem wir die Konsistenz der gefundenen Lösung mit den Orientierungen bzw. Bewegungen überprüfen können. Die Methode von Shizawa & Mase [147, 148] eignet sich für die Orientierungsbestimmung zweier Orientierungen in n -dimensionalen Räumen mit $n \geq 2$. Entsprechendes gilt dann auch für das Konsistenzkriterium, das wir aus ihrer Methode

ableiten, und wir werden die folgende Diskussion gleich für den n -dimensionalen Raum angeben. Ein n dimensionales Signal $g_1 : \mathbb{R}^n \rightarrow \mathbb{R}$ heißt ideal in Richtung $\mathbf{v} = (v_1, \dots, v_n)$ orientiert, wenn $\partial g(\hat{\mathbf{x}})/\partial \mathbf{v} = 0$ ist. Demnach ist für die Summe $f(\hat{\mathbf{x}}) = g_1(\hat{\mathbf{x}}) + g_2(\hat{\mathbf{x}})$, $\hat{\mathbf{x}} = (x_1, \dots, x_n)^T$ zweier ideal in Richtung \mathbf{u} und \mathbf{v} orientierter Signale

$$\frac{\partial^2}{\partial \mathbf{u} \partial \mathbf{v}} f(\hat{\mathbf{x}}) = \alpha(\mathbf{u})\alpha(\mathbf{v})f(\hat{\mathbf{x}}) = \sum_{i \leq j} c_{ij} f_{ij} = \mathbf{c} \cdot (f_{ij})_{i \leq j} = 0, \quad (2.45)$$

mit dem gemischten Orientierungsparametervektor

$$\mathbf{c} = (c_{ij})_{i \leq j}^T = \begin{cases} u_i v_j & \text{falls } i = j \\ u_i v_j + u_j v_i & \text{sonst} \end{cases}. \quad (2.46)$$

Betrachten wir den gemischten Orientierungsparametervektor $\mathbf{c} = \mathbf{c}(\mathbf{u}, \mathbf{v})$ als eine Abbildung, die von den Orientierungsvektoren \mathbf{u} und \mathbf{v} abhängt, so stellt sich für die gefundene Lösung \mathbf{s} die Frage, ob sie in der Form $\mathbf{s} = \mathbf{c}(\mathbf{u}, \mathbf{v})$ darstellbar ist. Dazu überführen wir Gleichung (2.45) in den Frequenzraum:

$$\mathbf{u} \cdot \boldsymbol{\omega} \mathbf{v} \cdot \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0, \quad (2.47)$$

wobei $F(\boldsymbol{\omega})$ die Fouriertransformierte von $f(\hat{\mathbf{x}})$ über einer Region Ω und $\boldsymbol{\omega} = (\omega_1, \dots, \omega_n)^T$ der Frequenzvektor sind. Diese Gleichung ist äquivalent zu

$$\begin{aligned} & \boldsymbol{\omega}^T \frac{1}{2} (\mathbf{u} \mathbf{v}^T + \mathbf{v} \mathbf{u}^T) \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0 \\ \iff & \frac{1}{2} (\boldsymbol{\omega}^T \mathbf{u} \mathbf{v}^T \boldsymbol{\omega} + \boldsymbol{\omega}^T \mathbf{v} \mathbf{u}^T \boldsymbol{\omega}) F(\boldsymbol{\omega}) = 0 \\ \iff & \frac{1}{2} (\mathbf{u} \cdot \boldsymbol{\omega} \mathbf{v} \cdot \boldsymbol{\omega} + \mathbf{u} \cdot \boldsymbol{\omega} \mathbf{v} \cdot \boldsymbol{\omega}) F(\boldsymbol{\omega}) = 0 \end{aligned} \quad (2.48)$$

wobei der Ausdruck $\boldsymbol{\omega}^T \mathbf{C}_s \boldsymbol{\omega} = \boldsymbol{\omega}^T \frac{1}{2} (\mathbf{u} \mathbf{v}^T + \mathbf{v} \mathbf{u}^T) \boldsymbol{\omega}$ eine quadratische Form beschreibt, die an den Stellen mit $F(\boldsymbol{\omega}) \neq 0$ den Wert null besitzt, damit die optische Flussgleichung erfüllt ist. Die Einträge der Matrix

$$\mathbf{C}_s = \frac{1}{2} (\mathbf{u} \otimes \mathbf{v} + \mathbf{v} \otimes \mathbf{u}) = \frac{1}{2} (\mathbf{u} \mathbf{v}^T + \mathbf{v} \mathbf{u}^T) \quad (2.49)$$

entsprechen den Einträgen des gemischten Orientierungsparametervektors \mathbf{c} , abgesehen von dem Faktor $1/2$ der Nichtdiagonalelemente von \mathbf{C}_s . Die Matrix \mathbf{C}_s besitzt die folgenden Eigenschaften:

Satz 3 Seien \mathbf{u} und \mathbf{v} Einheitsvektoren und θ der Winkel zwischen beiden Vektoren. Dann hat $\mathbf{C}_s = \frac{1}{2} (\mathbf{u} \otimes \mathbf{v} + \mathbf{v} \otimes \mathbf{u})$ nur zwei von null verschiedene Eigenvektoren $\lambda_1 = \cos^2 \frac{\theta}{2}$ und $\lambda_2 = -\sin^2 \frac{\theta}{2}$ mit den korrespondierenden Eigenvektoren $\mathbf{u} + \mathbf{v}$ und $\mathbf{u} - \mathbf{v}$.

Beweis:

Zunächst zeigen wir, dass $\mathbf{u} + \mathbf{v}$ ein Eigenvektor von \mathbf{C}_s ist, d.h. $\mathbf{C}_s(\mathbf{u} + \mathbf{v}) = \lambda_1(\mathbf{u} + \mathbf{v})$:

$$\begin{aligned}
\mathbf{C}_s(\mathbf{u} + \mathbf{v}) &= \frac{1}{2}(\mathbf{u}\mathbf{v}^T + \mathbf{v}\mathbf{u}^T)(\mathbf{u} + \mathbf{v}) \\
&= \frac{1}{2}(\mathbf{u}(\mathbf{v}^T\mathbf{u} + \mathbf{v}^T\mathbf{v}) + \mathbf{v}(\mathbf{u}^T\mathbf{u} + \mathbf{u}^T\mathbf{v})) \\
&= \frac{1}{2}(\mathbf{u}(\mathbf{v}^T\mathbf{u} + 1) + \mathbf{v}(1 + \mathbf{v}^T\mathbf{u})) \\
&\quad \text{da } \mathbf{v}^T\mathbf{v} = \mathbf{u}^T\mathbf{u} = 1 \text{ und } \mathbf{v}^T\mathbf{u} = \mathbf{u}^T\mathbf{v} \\
&= \frac{1}{2}(\mathbf{v}^T\mathbf{u} + 1)(\mathbf{u} + \mathbf{v}) = \cos^2\frac{\theta}{2}(\mathbf{u} + \mathbf{v}) = \lambda_1(\mathbf{u} + \mathbf{v})
\end{aligned} \tag{2.50}$$

Der Beweis, dass $\mathbf{u} - \mathbf{v}$ Eigenvektor ist, wird in analoger Weise durchgeführt. Wir erhalten

$$\begin{aligned}
\mathbf{C}_s(\mathbf{u} - \mathbf{v}) &= \frac{1}{2}(\mathbf{u}(\mathbf{v}^T\mathbf{u} + 1) - \mathbf{v}(\mathbf{v}^T\mathbf{u} - 1)) \\
&= -\frac{1}{2}(1 - \mathbf{v}^T\mathbf{u})(\mathbf{u} - \mathbf{v}) = -\sin^2\frac{\theta}{2}(\mathbf{u} - \mathbf{v}) = \lambda_2(\mathbf{u} - \mathbf{v}).
\end{aligned} \tag{2.51}$$

Bis jetzt haben wir gesehen, dass es mindestens zwei Eigenwerte gibt, die von null verschieden sind. Es bleibt noch zu zeigen, dass alle anderen Eigenwerte null sind. Sei \mathbf{e} ein Eigenvektor von \mathbf{C}_s , dann gilt $\mathbf{e} \perp (\mathbf{u} + \mathbf{v})$ und $\mathbf{e} \perp (\mathbf{u} - \mathbf{v})$. Damit gilt auch $\mathbf{e} \perp \mathbf{u}$ und $\mathbf{e} \perp \mathbf{v}$. Das Tensorprodukt zwischen \mathbf{u} und \mathbf{v} lässt sich als $\mathbf{u} \otimes \mathbf{v} = \mathbf{u}\mathbf{v}^T$ schreiben und damit erhalten wir $(\mathbf{u} \otimes \mathbf{v})\mathbf{e} = \mathbf{u}\mathbf{v}^T\mathbf{e} = \mathbf{u}\mathbf{0} = \mathbf{0}$, woraus folgt, dass alle weiteren Eigenwerte gleich null sind.

□

Ist \mathbf{A} eine Matrix und $p(\lambda) = |\mathbf{A} - \lambda\mathbf{I}| = \sum_{i=0}^p S_i \lambda^{p-i}$ ihr charakteristisches Polynom, dann sind die Koeffizienten S_i die Invarianten (symmetrische Polynome der Eigenwerte) von \mathbf{A} , die sich analytisch durch die Einträge von \mathbf{A} schreiben lassen. Aus dem obigen Satz können wir nun für \mathbf{C}_s folgendes schließen:

$$S_1 = -(\lambda_1 + \dots + \lambda_p) = -\cos^2\frac{\theta}{2} + \sin^2\frac{\theta}{2} = -\cos\theta \tag{2.52}$$

$$S_2 = \sum_{i < j} \lambda_i \lambda_j = -\cos^2\frac{\theta}{2} \sin^2\frac{\theta}{2} = -\left(\cos\frac{\theta}{2} \sin\frac{\theta}{2}\right)^2 = -\frac{\sin^2\theta}{4} \tag{2.53}$$

$$S_0 = 1, \quad S_i = 0 \quad \text{für } i > 2, \tag{2.54}$$

da für $i > 2$ pro Summand immer mindestens drei Eigenwerte miteinander multipliziert werden und demnach null ergeben². Aus dieser Beobachtung folgt nun, dass eine gefundene Lösung $\mathbf{c} = (c_{ij})_{i \leq j}^T$ genau dann in der Form $\mathbf{c}(\mathbf{u}, \mathbf{v})$ darstellbar ist, wenn die Elemente von \mathbf{c} angeordnet in der Matrix

$$\mathbf{C}_s = \begin{cases} c_{jj} & \text{falls } i = j \\ \frac{c_{ij}}{2} & \text{falls } i < j \\ \frac{c_{ji}}{2} & \text{falls } i > j \end{cases} \tag{2.55}$$

²Soll Gleichung (2.52) für die Bestimmung des Winkels zwischen beiden Vektoren genutzt werden, so müssen in der Regel beide Vektoren durch Teilen der Summen durch den Ausdruck $\lambda_1 - \lambda_2$ auf Einheitslänge normiert werden.

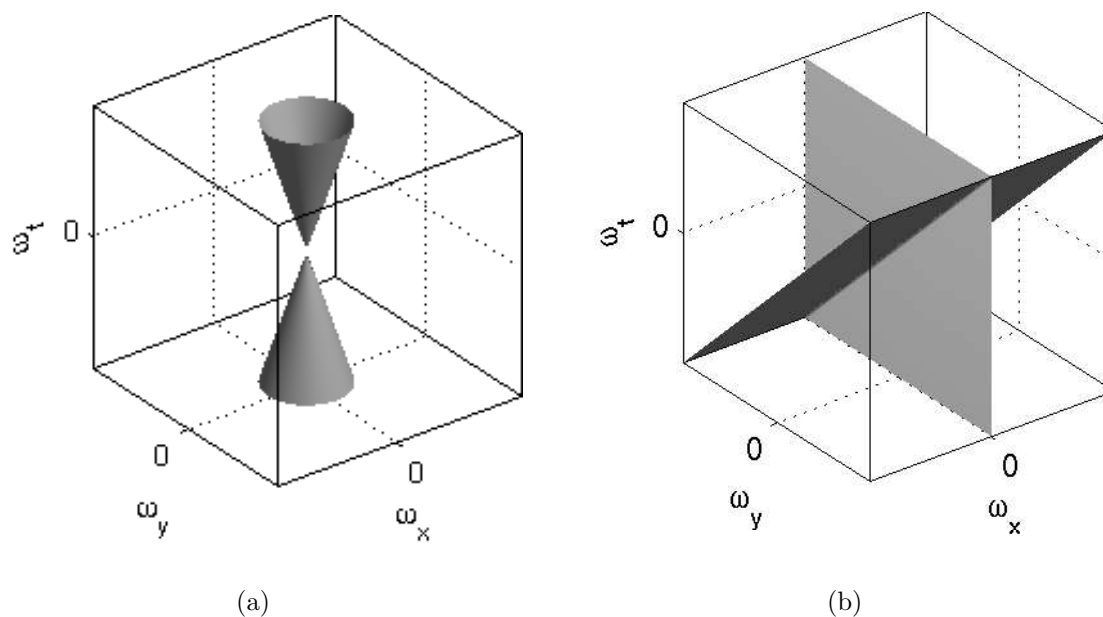


Abbildung 2.1: (a) Beispiel für ein Spektrum, bei dem es sich nicht um Orientierungen oder Bewegungen handelt und \mathbf{C}_s die Invariantenbedingungen nicht erfüllt. Nur Spektren die eine Ebenenstruktur wie in (b) aufweisen sind Lösungen im Sinne der Orientierung.

die Invariantenbedingungen

$$S_2 < 0, \quad S_i = 0 \quad \text{für } i > 2 \quad (2.56)$$

erfüllen.

Nach der Bestimmung eines gemischten Orientierungsparametervektors, sollte daher stets die Matrix \mathbf{C}_s aufgestellt und die Erfüllung der Invariantenbedingung überprüft werden, um gegebenenfalls Lösungen, die nicht mit dem Bewegungs- bzw. Orientierungsmodell konform sind zu verwerfen.

Abbildung 2.1(a) zeigt ein Beispielspektrum für eine Funktion $f(x, y, t)$, die die oben hergeleitete Invariantenbedingungen nicht erfüllt. Die Spektralkoeffizienten sind dabei nur an den Stellen des dargestellten Doppelkegels ungleich null. Solch ein Doppelkegel ist mittels einer quadratischen Form darstellbar und erfüllt die Bedingung

$$\boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0 \quad (2.57)$$

mit $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_t)^T$. Die quadratische Form

$$\boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} = \boldsymbol{\omega}^T \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \boldsymbol{\omega} = 0 \quad (2.58)$$

oder äquivalent $10\omega_x^2 + 10\omega_y^2 - \omega_t^2 = 0$ entspricht dem abgebildeten Doppelkegel. Auffallend an der Matrix \mathbf{A} ist ihre Diagonalstruktur. Im Vergleich zu der Matrix \mathbf{C}_s aus

Satz 3 hat \mathbf{A} drei von null verschiedene Eigenwerte, damit sind die Bedingungen aus Gleichung (2.56) für kein Element aus \mathcal{S}_2 erfüllt. Folglich handelt es sich bei der gefundenen Lösung weder um eine Orientierung noch um eine Bewegung.

Die hier vorgestellte Überprüfungsmethode ist leider nur für zwei Orientierungen bzw. Bewegungen anwendbar, da sich die Fouriertransformierte der erweiterten optische Flussgleichung ab drei Bewegungen nicht mehr durch eine quadratische Form beschreiben lässt. Die Herleitung eines Entscheidungskriteriums für drei oder mehr Bewegungen bleibt offen und erfordert aller Voraussicht nach eine neue Herangehensweise. In diesen Fällen können wir daher nur hoffen, dass die gefundene Lösung auch wirklich die Bewegungen repräsentiert. In der Regel sollte dies aber kein Problem darstellen, da die Bewegungsbestimmung nur auf Sequenzen angewendet werden sollte, in denen Bewegungen bzw. Orientierungen vorkommen und damit der Ebenencharakter im Frequenzraum sichgestellt ist und somit die Bewegungen gefunden werden sollten.

Das Überprüfungsverfahren ist auf die Orientierungsbestimmung in höherdimensionalen Räumen übertragbar. Konzeptionell sind keine weiteren Änderungen erforderlich, da mit der Dimension von $f(\hat{\mathbf{x}})$ auch die Größe der Orientierungsvektoren \mathbf{u} und \mathbf{v} wächst und entsprechend auch die Matrix \mathbf{C}_s und die damit verbundene Anzahl der Elemente in dem gemischten Orientierungsparametervektor. Gleichung (2.48) und die damit verbundene Orientierungsbedingung im Ort gelten ebenfalls in höheren Dimensionen.

2.3 Zusammenfassung

In diesem Abschnitt erweiterten wir die optische Flussgleichung von Shizawa & Mase [147, 148] auf die Bestimmung einer beliebigen Anzahl transparent überlagerter Bewegungen in Bildsequenzen. Die resultierende Gleichung weist Nichtlinearitäten in den Bewegungsvektorkomponenten auf und muss, damit sie später durch lineare Gleichungssysteme gelöst werden kann, linearisiert werden. Die Linearisierung erfolgte durch Einführung der sogenannten gemischten Geschwindigkeitsparameter als eine implizite und komplette Darstellung der Bewegungsvektoren. Nach der Bestimmung der gemischten Geschwindigkeitsparameter erhalten wir die gesuchten Geschwindigkeitsvektoren als Nullstellen eines komplexen Polynoms, dessen Koeffizienten sich aus den gemischten Geschwindigkeitsparametern ergeben. In der linearisierten Darstellung sind Lösungen, die zwar die linearisierten Bedingungen erfüllen aber keine Lösung im Sinne des Bewegungsmodells repräsentieren, möglich. Für zwei transparente Bewegungen bzw zwei transparente Orientierungen konnten wir ein Kriterium herleiten, um Lösungen, die nicht mit den Bewegungsmodell konsistent sind auszuschließen.

In den nun folgenden Kapitel werden wir zwei Verfahren vorstellen mit denen wir die gesuchten gemischten Bewegungsparameter berechnen können.

Kapitel 3

Verallgemeinertes Strukturtensorverfahren

Verfahren, die auf Ableitungen der Bildsequenz basieren, werden als die differenziellen Methoden der Bewegungsbestimmung bezeichnet. Die erweiterte optische Flussgleichung für transparent überlagerte Bewegungen basiert auf partiellen Ableitungen höherer Ordnung der Bildsequenz und ist damit den differenziellen Methoden der Bewegungsbestimmung zuzuordnen. Im Folgenden werden wir zwei unterschiedliche differenzielle Ansätze zur Bestimmung überlagerter Bewegungen vorstellen. Beide basieren auf der im vorherigen Kapitel hergeleiteten erweiterten optischen Flussgleichung. Das erste Verfahren leiten wir wie aus dem *Strukturtensor* [26, 74, 92, 97, 66, 94, 81, 88, 80, 17, 27, 28] ab, eine etablierte Methode zur Orientierungsanalyse in Bildern und mehrdimensionalen Daten, die sich auch zur Bewegungsbestimmung eignet.

3.1 Tensorverfahren

Das sogenannte *Strukturtensorverfahren* [26, 81, 88, 80, 17, 27, 28] für eine Bewegung bzw. Orientierung wird gerne in der Bildverarbeitung eingesetzt. Neben der Orientierungs- bzw. Bewegungsbestimmung erlauben seine Eigenwerte einen Rückschluss auf die zugrunde liegende Signalstruktur. Im Falle einer einzigen Bewegung bzw. Orientierung besteht der Tensor aus einer 3×3 Matrix. Sind alle Eigenwerte gleich null, liegt eine homogene Grauwertstruktur vor. Sind zwei Eigenwerte gleich null, existieren in der betrachteten Umgebung zwei Richtungen, in denen sich die Grauwerte nicht ändern, wie es zum Beispiel bei dem Aperturproblem der Fall ist. Ist nur ein Eigenwert gleich null, so gibt es nur eine Richtung, in der die Grauwerte konstant sind. Der Eigenvektor zu diesem Eigenwert gibt die Richtung an und entspricht dem Orientierungs- bzw. Bewegungsvektor. Sind alle Eigenwerte ungleich null, ändern sich die Grauwerte in jeder Richtung und es liegt keine Bewegung bzw. Orientierung vor, hervorgerufen zum Beispiel durch Störungen oder überlagerte Bewegungen. In diesem Abschnitt möchten wir zunächst das Tensorverfahren auf die Bestimmung überlagerter Bewegungen bzw. Orientierungen erweitern. Anschließend werden wir ein Konfidenzkriterium vorstellen, mit dessen Hilfe wir durch einen relativen Vergleich der Tensorinvarianten feststellen können, ob das gerade betrachtete Bewegungs/Orientierungsmodell zutreffend ist oder nicht. Dieses Konfidenzmaß erlaubt die Angabe eines hierarchischen Algorithmus, der zunächst versucht, eine Bewegung zu

bestimmen. Sollte das Konfidenzkriterium für eine Bewegung nicht erfüllt sein, fährt es mit der Bestimmung von zwei überlagerten Bewegungen fort. Dieses Vorgehen wird bis zur einer beliebigen Anzahl von Überlagerungen fortgeführt.

3.1.1 Strukturtensor für eine Bewegung

In diesem Abschnitt führen wir zunächst den Strukturtensor für eine Bewegung ein, um ihn dann in dem folgenden Abschnitt auf mehrere Bewegungen erweitern zu können. Wie in dem einführenden Abschnitt bereits gezeigt wurde, ergibt ein bewegter Punkt eine Gerade in dem Orts-Zeit-Raum. Um nun die Geschwindigkeit, mit der dieser Punkt bewegt wird, bestimmen zu können, brauchen wir nur die Richtung, in der die Linie verläuft, zu suchen. Diese Richtung entspricht dann unserem Geschwindigkeitsvektor.

In Bereichen konstanter Grauwerte oder solcher mit isotroper Grauwertverteilung (z.B unkorreliertes Rauschen) ist eine Bestimmung der Orientierung nicht möglich. In diesen Fällen ergibt sich ein Orientierungsvektor mit dem Betrag null.

Ausgangspunkt dieser Betrachtung ist wieder die optische Flussgleichung

$$f_x v_x + f_y v_y + f_t = 0. \quad (3.1)$$

Die optische Flussgleichung soll nun als inneres Produkt zweier Vektoren ausgedrückt werden, d.h das innere Produkt zwischen dem Gradientenvektor (∇f) und dem Geschwindigkeitsvektor \mathbf{v} . Bisher besteht der Geschwindigkeitsvektor \mathbf{v} nur aus den zwei Komponenten v_x und v_y . Damit obige Gleichung als inneres Produkt zweier Vektoren ausgedrückt werden kann, muss der Geschwindigkeitsvektor \mathbf{v} um eine weitere Komponente erweitert werden. Bei genauer Betrachtung der optischen Flussgleichung fällt auf, dass die partielle Ableitung von f nach der Zeit mit dem Faktor eins gewichtet wird. Aus diesem Grund wird die Geschwindigkeitskomponente v_t eingeführt, die (später) eins sein soll. Damit wird die optische Flussgleichung in die Form

$$f_x v_x + f_y v_y + f_t v_t = (\nabla f)^T \mathbf{v} = 0 \quad (3.2)$$

überführt mit dem Geschwindigkeitsvektor $\mathbf{v} = (v_x, v_y, v_t)^T$. Ziel ist es nun, diese Gleichung auf ein Minimierungsproblem zu reduzieren. Die obige Gleichung stellt ein Skalarprodukt zweier Vektoren dar. Dieses Skalarprodukt ist dann null, wenn die beiden Vektoren orthogonal zueinander stehen.

Zunächst ist es sinnvoll, zwischen den Begriffen *Orientierung* und *Richtung* zu unterscheiden. Die Richtung ist über dem gesamten Winkelbereich (360°) definiert. In diesem Fall gelten zwei Vektoren, die in entgegengesetzte Richtungen zeigen als unterschiedlich.

Betrachten wir nun die Richtungen lokal, d.h. in einer kleinen Nachbarschaft um einen Punkt herum, so können wir ein um 180° gedrehtes Muster nicht von einem nicht gedrehten Muster unterscheiden, da das Muster immer noch die gleiche Richtung hat. In diesem Fall brauchen wir also nur den Winkelbereich zwischen 0° und 180° zu betrachten und sprechen hier nicht mehr von Richtung sondern von Orientierung.

Das Skalarprodukt zwischen zwei Vektoren ist positiv, wenn die beiden Vektoren in dieselbe Richtung zeigen, und es ist negativ, wenn die beiden Vektoren in die entgegengesetzte Richtung zeigen. Die Entkoppelung von der Richtung wird dadurch erreicht, dass das Skalarprodukt quadriert wird. Angewandt auf die optische Flussgleichung heißt dies, dass die Gleichung quadriert werden muss, um die Entkoppelung von der Richtung zu erhalten. Ziel ist nun, den Ausdruck

$$|(\nabla f(\mathbf{x}))^T \mathbf{v}|^2 \rightarrow \min \quad (3.3)$$

über den Vektor \mathbf{v} zu minimieren.

Nun ist die triviale Lösung Geschwindigkeit $\mathbf{v} \equiv 0$ immer ein Minimum dieser Gleichung und muss ausgeschlossen werden. Der Gradient $\nabla f(\mathbf{x})$ und der Geschwindigkeitsvektor \mathbf{v} sind beide dreidimensionale Vektoren. Alle Geschwindigkeitsvektoren, die in der Ebene senkrecht zu dem Gradientenvektor $\nabla f(\mathbf{x})$ liegen, erfüllen die Minimierungsbedingung. Diese Lösungsmenge für \mathbf{v} muss also weiter eingeschränkt werden, um die gesuchte Orientierung zu finden. Zur Konstruktion weiterer Gleichungen können wir sie im Sinne der Kleinsten-Quadrate lösen, in dem wir sie über eine kleine Nachbarschaft Ω integrieren. Damit muss der Ausdruck

$$\begin{aligned} Q(\mathbf{x}, \mathbf{v}) &= \int_{\Omega} |(\nabla f(\mathbf{x}'))^T \mathbf{v}|^2 h(\mathbf{x} - \mathbf{x}') d\mathbf{x}' \\ &= \int_{\Omega} \mathbf{v}^T (\nabla f(\mathbf{x}')) (\nabla f(\mathbf{x}'))^T \mathbf{v} h(\mathbf{x} - \mathbf{x}') d\mathbf{x}' \rightarrow \min \end{aligned} \quad (3.4)$$

minimiert werden, wobei mit $h(\mathbf{x})$ eine kleine Nachbarschaft bezeichnet wird. Gesucht ist nun ein Minimierungsverfahren, das die Lösung $\mathbf{v} \equiv 0$ ausschließt.

Eine Nebenbedingung, die die Lösung $\mathbf{v} \equiv 0$ ausschließt, kann mit Hilfe der Lagrange'schen Multiplikatormethode eingeführt werden. Als Nebenbedingung wird die Funktion

$$g(\mathbf{v}) = 1 - \|\mathbf{v}\|^2 \stackrel{!}{=} 0 \quad (3.5)$$

gewählt. Diese Nebenbedingung legt die Norm von \mathbf{v} auf eins, es wird damit auf der Einheitssphäre minimiert. Nach dem Einsetzen dieser Nebenbedingung mit dem Lagrange'schen Multiplikator λ ergibt sich der Ausdruck

$$\frac{\partial}{\partial \mathbf{v}} \left(\int_{\Omega} \mathbf{v}^T (\nabla f(\mathbf{x}')) (\nabla f(\mathbf{x}'))^T \mathbf{v} h(\mathbf{x} - \mathbf{x}') d\mathbf{x}' - \lambda g(\mathbf{v}) \right) = 0. \quad (3.6)$$

Diese Vorgehensweise führt zu dem Eigenvektorproblem

$$\mathbf{J}(\mathbf{x}) \mathbf{v} = \lambda \mathbf{v} \quad (3.7)$$

mit dem *Strukturtensor*

$$\mathbf{J}(\mathbf{x}) = \int_{\Omega} h(\mathbf{x} - \mathbf{x}') (\nabla f(\mathbf{x}')) (\nabla f(\mathbf{x}'))^T d\mathbf{x}'. \quad (3.8)$$

Der Eigenvektor \mathbf{v} zum Eigenwert null minimiert die Gleichung (3.4) und gibt die Richtung der Bewegung an. Es stellt sich nun die Frage, wie man aus dem Richtungsvektor \mathbf{v} die richtigen Geschwindigkeitskomponenten erhält. Die letzte Komponente des Geschwindigkeitsvektors muss eins ergeben, d.h. $v_t = 1$. Damit muss der Eigenvektor zum Eigenwert null durch seine letzte Komponente, die v_t -Komponente, geteilt werden, um die richtigen Geschwindigkeitskomponenten v_x und v_y zu erhalten

$$\mathbf{v}_{\text{gesucht}} = \frac{1}{v_t} \mathbf{v} = \begin{pmatrix} \frac{v_x}{v_t} \\ \frac{v_y}{v_t} \\ 1 \end{pmatrix}. \quad (3.9)$$

3.1.2 Praktische Berechnung des Strukturtenors

Nach der Einführung des Strukturtenors soll kurz gezeigt werden, wie der Strukturtenor in der Praxis aufgestellt wird. Zunächst müssen alle partiellen Ableitungen der Bildfolge, d.h. f_x , f_y und f_t , bestimmt werden. Die Nachbarschaftsfunktion $h(\mathbf{x})$ wird durch einen kleinen diskreten Faltungskern, z.B. eine Gaußmaske, realisiert. Der Strukturtenor \mathbf{J} für den Punkt \mathbf{x} lautet nun

$$\mathbf{J}(\mathbf{x}) = \begin{pmatrix} (f_x(\mathbf{x})f_x(\mathbf{x})) * h(\mathbf{x}) & (f_x(\mathbf{x})f_y(\mathbf{x})) * h(\mathbf{x}) & (f_x(\mathbf{x})f_t(\mathbf{x})) * h(\mathbf{x}) \\ (f_y(\mathbf{x})f_x(\mathbf{x})) * h(\mathbf{x}) & (f_y(\mathbf{x})f_y(\mathbf{x})) * h(\mathbf{x}) & (f_y(\mathbf{x})f_t(\mathbf{x})) * h(\mathbf{x}) \\ (f_t(\mathbf{x})f_x(\mathbf{x})) * h(\mathbf{x}) & (f_t(\mathbf{x})f_y(\mathbf{x})) * h(\mathbf{x}) & (f_t(\mathbf{x})f_t(\mathbf{x})) * h(\mathbf{x}) \end{pmatrix} \quad (3.10)$$

Nach der Berechnung des Eigenvektors \mathbf{v} zum Eigenwert null kann der Geschwindigkeitsvektor nach der Formel (3.9) berechnet werden.

3.1.3 Konfidenzmaß des Strukturtenors

Nicht alle Punkte in der Bildfolge eignen sich zur Bestimmung der Geschwindigkeiten. Daher ist es notwendig, ein Entscheidungskriterium zu entwerfen, mit dem festgestellt werden kann, ob ein Punkt in der Bildfolge für die Bestimmung der Geschwindigkeit geeignet ist. Solche Aussagen können anhand der Eigenwerte des Strukturtenors \mathbf{J} getroffen werden. Bevor dieses Maß entworfen werden soll, ist es jedoch hilfreich, einige Eigenschaften des Strukturtenors zu betrachten.

Satz 4 Für den Strukturtenor \mathbf{J} gelten folgende Aussagen:

1. \mathbf{J} ist symmetrisch
2. \mathbf{J} ist positiv semidefinit (d.h. $\mathbf{x}^T \mathbf{J} \mathbf{x} \geq 0, \forall \mathbf{x}$)
3. \mathbf{J} hat nur reelle Eigenwerte
4. Alle Eigenwerte von \mathbf{J} sind größer gleich null
5. Eigenvektoren zu verschiedenen Eigenwerten sind orthogonal

Beweis

1. es gilt: $f_i f_j = f_j f_i$ mit $i, j \in \{x, y, t\}$

$$2. \mathbf{x}^T (\nabla f) (\nabla f)^T \mathbf{x} \geq 0, ((\nabla f)^T \mathbf{x}) \cdot ((\nabla f)^T \mathbf{x}) \geq 0$$

Das Skalarprodukt eines Vektors mit sich selbst ist immer größer gleich null.

3. folgt aus 1.

4. folgt aus 2.

5. folgt aus 1.

□

Da der Tensor \mathbf{J} symmetrisch und positiv semidefinit ist, folgt aus dem Satz von Courant-Fischer [73], dass Gleichung (3.4) für den Eigenvektor \mathbf{v}_1 korrespondierend zu dem kleinsten Eigenwert λ_1 minimal ist und es gilt

$$Q(\mathbf{x}, \mathbf{v}_1) = \min_{\|\mathbf{v}\|=1} \mathbf{v}^T \mathbf{J} \mathbf{v} = \mathbf{v}_1^T \mathbf{J} \mathbf{v}_1 = \lambda_1. \quad (3.11)$$

Damit ist sichergestellt, dass der Eigenvektor v_1 dem Minimum von $Q(\mathbf{x}, \mathbf{v}_1)$ entspricht. Im Idealfall ist $\lambda_1 = 0$ und folglich ist $Q(\mathbf{x}, \mathbf{v}_1) = 0$. Ist zusätzlich noch ein weiterer Eigenwert λ_2 mit dem zugehörigen Eigenvektor \mathbf{v}_2 gleich null, so ist ebenfalls $Q(\mathbf{x}, \mathbf{v}_2) = 0$ und es existiert eine Ebene, aufgespannt durch \mathbf{v}_1 und \mathbf{v}_2 , für die Q ihr Minimum annimmt. An dieser Stelle soll nun eine Aussage über die Bewegungsbestimmung an Hand der Eigenwerte des Strukturtenors getroffen werden. *Tabelle (3.1)* listet die Bedeutung der

Bedingung	Rang	Bedeutung
$\lambda_1 = \lambda_2 = \lambda_3 = 0$	0	Die Grauwerte ändern sich nicht, es liegt eine konstante Umgebung vor.
$\lambda_1 > 0, \lambda_2 = \lambda_3 = 0$	1	Die Grauwerte ändern sich nur in einer Richtung, die durch den Eigenvektor des Eigenwertes, der ungleich null ist, bestimmt wird.
$\lambda_1 > 0, \lambda_2 > 0, \lambda_3 = 0$	2	Die Grauwerte ändern sich in zwei Richtungen und sind konstant in einer dritten. Der Eigenvektor des Eigenwertes mit dem Wert null, gibt die Richtung der konstanten Grauwerte an. Dieser Eigenvektor gibt bei der Bewegung von Bildern die Geschwindigkeit der Bewegung an.
$\lambda_1 > 0, \lambda_2 > 0, \lambda_3 > 0$	3	Die Grauwerte ändern sich in allen drei Richtungen

Tabelle 3.1: *Bedeutung der Eigenwerte des Strukturtenors.*

Eigenwerte des Strukturtenors für die Bewegungsbestimmung auf und gibt gleichzeitig den Rang des Strukturtenors für die jeweiligen Eigenwerte an. Sind alle Eigenwerte null, so bedeutet dies, dass Gleichung (3.4) in jeglicher Richtung minimal ist. Dies wiederum bedeutet, dass der Gradient null sein muss und damit eine isotrope Grauwertstruktur vorliegt. In diesem Fall kann also keine Geschwindigkeit bestimmt werden. Sind zwei Eigenwerte null und ist einer von null verschieden, so bedeutet dies, dass der Gradient über die Nachbarschaft hinweg konstant ist und damit eine Lösungsebene beschrieben wird. In diesem Fall liegt ein Blendenproblem vor. Ist ein Eigenwert null und alle anderen Eigenwerte sind von null verschieden, so stellt die Lösungsmenge eine Linie dar, deren

Orientierung der gesuchten Geschwindigkeit entspricht. Sind alle Eigenwerte ungleich null, so ist $Q(\mathbf{x}, \mathbf{v})$ nur für die triviale Lösung $\mathbf{v} = \mathbf{0}$ gleich null, und die Grauwerte ändern sich in allen Richtungen. Für die Bewegungsbestimmung ist also nur der Fall “ein Eigenwert null, alle anderen ungleich null” relevant.

3.1.4 Generalisierter Strukturtensor

Die Erweiterungen des Strukturtensorverfahrens zu mehreren Bewegungen gestaltet sich mit den Erkenntnissen des vorangegangenen Kapitels denkbar einfach. Die erweiterte optische Flussgleichung

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f = \sum_I c_I f_I = 0 \quad (3.12)$$

stellt eine Gleichung mit vielen Unbekannten dar und lässt sich – ähnlich wie die optische Flussgleichung für eine Bewegung – als inneres Produkt zweier Vektoren, d.h.

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f = \mathbf{d}f \cdot \mathbf{c} = 0 \quad (3.13)$$

schreiben. Der Datenvektor $\mathbf{d}f = (f_I)_I$ ist ortsabhängig und besteht aus den partiellen Ableitungen N-ter Ordnung der Bildsequenz. Der Parametervektor $\mathbf{c} = (c_I)_I$ ist zumindestens lokal vom Ort unabhängig und enthält die gemischten Geschwindigkeits- bzw. Orientierungsparameter.

Für die Bestimmung des Vektors \mathbf{c} kann wie bei dem Strukturtensor für eine Bewegung die Lagrange’sche Multiplikatorenmethode verwendet werden, um die triviale Lösung $\mathbf{c} \equiv 0$ auszuschließen und um das minimale \mathbf{c} zu finden. Nach dem Multiplizieren von Gleichung (3.13) mit $\mathbf{c} \cdot \mathbf{d}f(\mathbf{x})$ erhält man eine quadratische Gleichung. Unter der Annahme, dass die \mathbf{v}_i in einer kleinen Umgebung konstant sind, kann für jeden Punkt $\mathbf{x} = (x, y, t)$ über eine kleine gewichtete Umgebung integriert werden. Damit erhält man eine Minimierungsaufgabe

$$Q(\mathbf{x}, \mathbf{c}) = \int_{\Omega} \mathbf{c}^T \mathbf{d}f(\mathbf{x}') \mathbf{d}f(\mathbf{x}')^T \mathbf{c} h(\mathbf{x} - \mathbf{x}') d\mathbf{x}' \rightarrow \min, \quad (3.14)$$

die in Analogie zu dem Strukturtensor für eine Bewegung auf die Eigenvektoraufgabe zum Eigenwert null reduziert wird

$$\mathbf{J}_N(\mathbf{x}) \cdot \mathbf{c} = \lambda \mathbf{c}, \quad (3.15)$$

mit

$$\mathbf{J}_N(\mathbf{x}) = \int_{\Omega} \mathbf{d}f(\mathbf{x}') \mathbf{d}f(\mathbf{x}')^T h(\mathbf{x} - \mathbf{x}') d\mathbf{x}'. \quad (3.16)$$

Die quadratische Matrix \mathbf{J}_N wird in dem Folgenden als der *generalisierte Strukturtensor* für N Bewegungen/Orientierungen bezeichnet. Für ihn gelten dieselben Eigenschaften aus Satz 4, wie beim Strukturtensor für eine Bewegung. Der gesuchte gemischte Bewegungs- bzw. Orientierungsparametervektor \mathbf{c} entspricht wieder dem Eigenvektor zum Eigenwert null der Matrix \mathbf{J}_N . Nachdem der Vektor \mathbf{c} bestimmt wurde, ergeben sich die gesuchten Geschwindigkeitsvektoren $\mathbf{v}_i, i = 1, \dots, N$ aus den Nullstellen des komplexen Polynoms (2.33) oder im Falle von zwei Orientierungen durch Berechnen von (2.42). Für die

Bestimmung der Bewegungsvektoren muss die letzte Komponente von \mathbf{c} , die per Konvention dem Vorfaktor der rein zeitlichen partiellen Ableitung N -ter Ordnung der Bildsequenz entspricht, eins ergeben. Nach der Berechnung von \mathbf{c} als Eigenvektor zum Eigenwert null von \mathbf{J}_N ist dies im Allgemeinen nicht der Fall. Der Eigenvektor \mathbf{c} und der gesuchte gemischte Geschwindigkeitsparametervektor $\mathbf{c}_{\text{gesucht}}$ zeigen in die gleiche Richtung. Daher erhalten wir $\mathbf{c}_{\text{gesucht}}$ durch skalieren des Vektors \mathbf{c} , indem wir \mathbf{c} durch seine letzte Komponente teilen:

$$\mathbf{c}_{\text{gesucht}} = \frac{\mathbf{c}}{c_m}, \quad (3.17)$$

wobei m der Länge des Vektors \mathbf{c} entspricht.

3.1.5 Transparente Bewegungen in Farbsequenzen

Bisher sind wir davon ausgegangen, dass unsere Bildsequenz nur aus Grauwerten besteht. Viele Bildsequenzen und Videofilme werden jedoch in Farbe aufgenommen. Für die Bewegungs- bzw. Orientierungsbestimmung ist die Farbinformation nicht zwingend erforderlich, da sich die Farbbilder in Grauwertbilder umrechnen lassen, andererseits kann in einigen Fällen die Berücksichtigung der Farben helfen, die Ergebnisse zu verbessern, da sich durch die Farbinformation weitere Gleichungen gewinnen lassen. Handelt es sich bei der Bildsequenz oder dem dreidimensionalen Datensatz um Multispektral- bzw. Farbdaten, so können wir sie durch die Abbildung

$$\mathbf{f} : \mathbb{R}^3 \rightarrow \mathbb{R}^q, \quad (3.18)$$

beschreiben. In den meisten Fällen besitzt q den Wert drei, z.B. bei der RGB-Farbdarstellung. Ähnlich wie vorher auch bezeichnen wir \mathbf{f} in einer offenen Region Ω als ideal orientiert, wenn ein Unterraum $E \subset \mathbb{R}^3$ existiert, so dass

$$\mathbf{f}(\mathbf{x} + \mathbf{v}) = \mathbf{f}(\mathbf{x}) \quad (3.19)$$

für alle $\mathbf{x}, \mathbf{v}; \mathbf{x}, \mathbf{x} + \mathbf{v} \in \Omega, \mathbf{v} \in E$ ist, unabhängig von q . An dieser Stelle möchten wir jedoch nur die Fälle betrachten in denen $\dim(E) = 1$ ist, da dieser Unterraum die Orientierung bzw. Bewegung repräsentiert und es daher nur eine Richtung gibt, in der sich nichts ändert. Leiten wir Gleichung (3.19) in dieser Richtung ab, so erhalten wir als Resultat den Nullvektor:

$$\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{v}} = \alpha(\mathbf{v})\mathbf{f} = v_x \mathbf{f}_x + v_y \mathbf{f}_y + v_t \mathbf{f}_t = \mathbf{0}. \quad (3.20)$$

Damit erfüllt jeder Farbkanal $\mathbf{f} = (f_1, \dots, f_q)^T$ die Bedingung

$$\alpha(\mathbf{v})f_k = 0, \quad k = 1, \dots, q. \quad (3.21)$$

In Analogie zu dem Tensorverfahren für eine Bewegung/Orientierung minimieren wir das Fehlerfunktional

$$\int_{\Omega} \left| \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{v}} \right|^2 \mathbf{d}\mathbf{x} = 0, \quad (3.22)$$

das im Idealfall null ergibt. Mit $\partial \mathbf{f}(\mathbf{x})/\partial \mathbf{v} = (\nabla f_1(\mathbf{x}) \cdot \mathbf{v}, \dots, \nabla f_q(\mathbf{x}) \cdot \mathbf{v})^T$ ist

$$\left| \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{v}} \right|^2 = \mathbf{v}^T \left(\sum_{i=1}^q \nabla f_i \nabla f_i^T \right) \mathbf{v} \quad (3.23)$$

und wir können Gleichung (3.22) als

$$\mathbf{v}^T \mathbf{J}(\mathbf{f}) \mathbf{v} = 0 \quad (3.24)$$

mit dem Tensor

$$\mathbf{J}(\mathbf{f}) = \int_{\Omega} \left(\sum_{i=1}^q \nabla f_i \nabla f_i^T \right) \mathbf{d}\mathbf{x}. \quad (3.25)$$

schreiben. Äquivalent können wir diesen farbwertigen Tensor durch die Summe der Tensoren

$$\mathbf{J}(\mathbf{f}) = \sum_{i=1}^q \mathbf{J}(f_i) \quad (3.26)$$

darstellen, wobei $\mathbf{J}(f) = \int_{\Omega} \nabla f \nabla f^T \mathbf{d}\mathbf{x}$ dem grauwertigen Tensor für eine Bewegung bzw. Orientierung entspricht. Der farbwertige Tensor ergibt sich demnach aus einer Summe von grauwertigen Tensoren angewandt auf die jeweiligen Farbkanäle. Ein vergleichbares Ergebnis wurde von Di Zeno [191] zur Orientierungsanalyse in Farbbildern erzielt. $\mathbf{J}(\mathbf{f})$ besitzt die gleichen Eigenschaften wie $\mathbf{J}(f)$ und der gesuchte Bewegungs bzw. Orientierungsvektor ergibt sich wieder aus dem Eigenvektor zum Eigenwert null von $\mathbf{J}(\mathbf{f})$. Dieses Konzept ist direkt auf die Bestimmung von überlagerten Orientierungen bzw. Bewegungen übertragbar. Für eine Überlagerung von N farbwertigen orientierten Strukturen

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^N \mathbf{g}_i(\mathbf{v}_i \cdot \mathbf{x}) \quad (3.27)$$

mit $\partial \mathbf{g}_i/\partial \mathbf{v}_i = \mathbf{0}$, $i = 1, \dots, N$ ist die entsprechende Orientierungsbedingung durch die Linearkombination

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) \mathbf{f} = \frac{\partial^N \mathbf{f}}{\partial \mathbf{v}_1 \cdots \partial \mathbf{v}_N} = \sum_I c_I \mathbf{f}_I = \mathbf{0} \quad (3.28)$$

beschrieben, wobei wieder die c 's die gemischten Orientierungsparameter darstellen. Mit Hilfe der Vektoren $\mathbf{d}f = (f_I)_I$ und $\mathbf{c} = (c_I)_I$ können wir obige Gleichung als

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) \mathbf{f} = (\mathbf{c} \cdot \mathbf{d}f_1, \dots, \mathbf{c} \cdot \mathbf{d}f_q)^T = \mathbf{0} \quad (3.29)$$

schreiben und es gilt

$$|\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) \mathbf{f}|^2 = \mathbf{c}^T \left(\sum_{i=1}^q \mathbf{d}f_i \mathbf{d}f_i^T \right) \mathbf{c}. \quad (3.30)$$

Dementsprechend ist der farbwertige Tensor für N Orientierungen durch

$$\mathbf{J}_N(\mathbf{f}) = \int_{\Omega} \sum_{i=1}^q \mathbf{d}f_i \mathbf{d}f_i^T \mathbf{d}\mathbf{x} \quad (3.31)$$

bestimmt und ebenfalls durch eine Summe von grauwertigen Tensoren angewandt auf die Farbkanäle darstellbar:

$$\mathbf{J}_N(\mathbf{f}) = \sum_{i=1}^q \mathbf{J}_N(f_i), \quad \text{mit } \mathbf{J}_N(f) = \int_{\Omega} \mathbf{d}f \mathbf{d}f^T \mathbf{d}\mathbf{x}. \quad (3.32)$$

Für den farbwertigen Tensor gelten ebenfalls die Eigenschaften aus Satz 4, denn eine Summe von positiv semidefiniten Matrizen ergibt eine positiv semidefinite Matrix. Das die Summe positiv semidefinit ist, ist aus Gleichung (3.29) ersichtlich, denn das Skalarprodukt $(\mathbf{c} \cdot \mathbf{d}f_1, \dots, \mathbf{c} \cdot \mathbf{d}f_q)(\mathbf{c} \cdot \mathbf{d}f_1, \dots, \mathbf{c} \cdot \mathbf{d}f_q)^T \geq 0$ eines Vektors mit sich selbst ist immer größer oder gleich null.

Wie wir gesehen haben, erfordert die Berücksichtigung der Farbinformation nur eine geringfügige Modifikation der ursprünglich für grauwertige Datensätze entworfenen Verfahren. Der benötigte Rechenaufwand steigt dabei proportional zur Anzahl der einbezogenen Farbkanäle. Es bleibt daher zwischen dem erhofften Zugewinn und dem Rechenaufwand abzuwägen. Eines steht jedoch fest: Unterscheiden sich alle Gradienten der jeweiligen Farbkanäle nur durch einen Proportionalitätsfaktor, so ist durch die Berücksichtigung der Farben kein Gewinn zu verzeichnen.

3.1.6 Größe des Strukturensors

Für jede Anzahl von Bewegungen wird ein eigener Strukturtensor benötigt. Es soll nun gezeigt werden, wie sich die Größe der Tensoren mit der Anzahl der Bewegung erhöht. Betrachtet man die optische Flussgleichung für N Bewegungen

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f = 0 \quad (3.33)$$

so erhält man unter der Vernachlässigung aller Symmetrien 3^N Summanden, da jedes $\alpha(\mathbf{v}_i)$ drei Summanden enthält. Werden die Symmetrien ausgenutzt, so reduziert sich die Anzahl der Summanden deutlich. Die Anzahl der Summanden berechnet sich nach der Formel

$$\sum_{i_1=1}^3 \sum_{i_2=i_1}^3 \cdots \sum_{i_N=i_{N-1}}^3 1 = 1 + 2 + \cdots + (N + 1). \quad (3.34)$$

Für die Summe lässt sich eine geschlossene Formel

$$m = \frac{1}{2}(N + 1)(N + 2) \quad (3.35)$$

angeben, m ist die Anzahl der Zeilen und Spalten des Strukturensors. Für den Fall von zwei Bewegungen hat der Strukturtensor die Größe 6×6 , für drei die Größe 10×10 und für vier Bewegungen die Größe 15×15 .

3.1.7 Konfidenzmaße und hierarchische Bewegungsbestimmung

Nicht in allen Bereichen einer Bildsequenz können Bewegungen bestimmt werden, z.B. in Bereichen, in denen keine Bewegung vorkommt oder in denen ein Aperturproblem vorliegt.

Es ist ebenfalls nicht sinnvoll, an Stellen mit nur einer Bewegung mehrere Bewegungsparameter bestimmen zu wollen. Aus diesem Grunde ist ein Maß notwendig, mit dessen Hilfe entschieden werden kann, ob sich ein Bereich zur Bewegungsbestimmung eignet oder nicht. In den nun anschließenden Abschnitten soll solch ein Kriterium entwickelt werden, welches wir im folgenden als *Konfidenzmaß* [118] bezeichnen.

Eigenwerte als Konfidenzmaß

Im Falle einer Bewegung ist die Konfidenz hoch, wenn ein Eigenwert null und die anderen beiden Eigenwerte von null verschieden sind. In diesem Fall hat der Strukturtensor \mathbf{J} den Rang zwei. In Bereichen mit einem Aperturproblem hat \mathbf{J} zwei Eigenwerte null und in Bereichen mit Bewegungsdiskontinuitäten hat \mathbf{J} drei von null verschiedene Eigenwerte. Kommen N Bewegungen in einem Bereich vor, so ist die Konfidenz hoch, wenn der Strukturtensor \mathbf{J}_N den Rang $m - 1$ hat.

Das lineare Eigenwertssystem (3.15) hat nur dann eine Lösung, wenn die Determinante $\det(\mathbf{J}_N - \lambda \mathbf{I})$ verschwindet. Ausgerechnet ergibt sich das charakteristische Polynom in der Form

$$\lambda^m - S_1 \lambda^{m-1} + S_2 \lambda^{m-2} - \dots \pm S_{m-1} \lambda \pm S_m = 0, \quad (3.36)$$

wobei die S_i Funktionen der Elemente des Strukturensors \mathbf{J}_N sind. Gleichung (3.36) ist eine skalare Gleichung, λ und seine Produkte sind ebenfalls Skalare, dann müssen auch die S_i Skalare sein. Als die Grundinvarianten des Tensors \mathbf{J}_N sind sie von der speziellen Wahl eines Koordinatensystems unabhängig. Für das weitere Vorgehen werden die drei Invarianten S_m , S_1 und S_{m-1} benötigt, die noch geeignet normiert werden

$$K = \det(\mathbf{J}_N) = \lambda_1 \lambda_2 \cdots \lambda_m \quad (3.37)$$

$$H = \frac{1}{m} \text{trace}(\mathbf{J}_N) = \frac{1}{m} (\lambda_1 + \cdots + \lambda_m) \quad (3.38)$$

$$S = \frac{1}{m} \text{trace}(\mathbf{M}) = \frac{1}{m} \sum_i \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m, \quad (3.39)$$

wobei die λ_i die Eigenwerte des Systems sind und $\lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m = \lambda_1 \cdots \lambda_{i-1} \lambda_{i+1} \cdots \lambda_m$ ist. Die Matrix \mathbf{M} entspricht der Unterdeterminantenmatrix von \mathbf{J}_N , deren Einträge \mathbf{M}_{ij} durch Streichen der $(m - i + 1)$ -ten Zeile und $(m - j + 1)$ -ten Spalte von \mathbf{J}_N entstanden sind. Ist mindestens ein Eigenwert null, so verschwindet die Determinante von \mathbf{J}_N und bei der Summe für die Berechnung von S bleibt nur noch der Summand übrig, bei dem der Eigenwert null gestrichen wurde. Damit ist K gleich null und S größer null. Sind nun mindestens zwei Eigenwerte null, so ist die Determinante ebenfalls null und der Wert S verschwindet. In diesem Fall darf keine Bewegung bestimmt werden. Sollten alle Eigenwerte von null verschieden sein, so ist die Determinante ungleich null und der Wert S größer als null. Eine Bewegungsbestimmung ist auch in diesem Fall nicht möglich. Mit diesen Werten können wir nun das Konfidenzmaß $K = 0$ und $S \neq 0$ verwenden. Bei der numerischen Berechnung der Eigenwerte kann es aufgrund von Rundungsfehlern sein, dass der kleinste Eigenwert zwar sehr klein aber nicht ideal null ist und damit die Determinante nicht verschwindet. Aus diesem Grund ist ein relativer Vergleich zwischen K und S sinnvoll.

Satz 5 *Es gilt $K^{\frac{1}{m}} \leq S^{\frac{1}{m-1}} \leq H$. Die Gleichheit gilt genau dann, wenn es sich um einen Punkt mit einer konstanten Umgebung handelt.*

Beweis

Die erste Ungleichung ist die Konsequenz der bekannten Ungleichung zwischen dem arithmetischen und dem geometrischen Mittelwert

$$(x_1 x_2 \cdots x_m)^{\frac{1}{m}} \leq \frac{1}{m}(x_1 + x_2 + \cdots + x_m), \quad (3.40)$$

wobei $x_i = \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m$. Die Ungleichung für das arithmetische und das geometrische Mittel gilt nur, wenn alle x_i größer oder gleich null sind. Das ist hier der Fall, da alle Eigenwerte größer oder gleich null sind. x_i eingesetzt erhalten wir

$$(\lambda_1^{m-1} \lambda_2^{m-1} \cdots \lambda_m^{m-1})^{\frac{1}{m}} \leq \frac{1}{m} \left(\sum_{i=1}^m \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m \right) \quad (3.41)$$

und damit

$$(\lambda_1 \lambda_2 \cdots \lambda_m)^{\frac{1}{m}} \leq \left(\frac{1}{m} \sum_{i=1}^m \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m \right)^{\frac{1}{m-1}}. \quad (3.42)$$

Für den Beweis der zweiten Ungleichung wird die Differenz $D = S - \alpha H$ gebildet und nach dem Extremwert gesucht. Wir leiten D nach den λ_i ab und erhalten die Gleichungen

$$\sum_{i \neq k} \lambda_1 \cdots \hat{\lambda}_i \cdots \hat{\lambda}_k \cdots \lambda_m = \alpha \quad \text{für } k = 1, \dots, m \quad (3.43)$$

was $\lambda_1 = \lambda_2 = \cdots = \lambda_m$ impliziert: Um z.B. $\lambda_1 = \lambda_2$ zu zeigen, subtrahieren wir die erste Gleichung von der zweiten und erhalten

$$(\lambda_2 - \lambda_1) \sum_{i \geq 3} \lambda_3 \cdots \hat{\lambda}_i \cdots \lambda_m = 0. \quad (3.44)$$

Dies impliziert $\lambda_2 = \lambda_1$ für die inneren Punkte in dem Definitionsbereich von S . Um $\lambda_3 = \lambda_2$ zu zeigen, benutzen wir die zweite und dritte Gleichung, usw. Setzen wir $H = r$, so ist $\lambda_1 = \lambda_2 = \cdots = \lambda_m = r$ und wir erhalten $S = r^{m-1}$. Wir haben nun die Gleichheit der Aussage für die Extrema gezeigt.

Nun werden wir S entlang der Kante $\lambda_j = 0$ ermitteln. Zur Vereinfachung nehmen wir $j = m$ an. In diesem Fall ergibt sich

$$S = \frac{\lambda_1 \lambda_2 \cdots \lambda_{m-1}}{m} \quad \text{und} \quad H = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_{m-1}}{m}. \quad (3.45)$$

Mit Ungleichung (3.40) erhalten wir die Gleichung

$$(mS)^{\frac{1}{m-1}} = (\lambda_1 \lambda_2 \cdots \lambda_{m-1})^{\frac{1}{m-1}} \leq \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_{m-1}}{m-1} = \frac{m}{m-1} H, \quad (3.46)$$

welche für $H = r$ äquivalent zu

$$S \leq \frac{1}{m} \left(\frac{m}{m-1} \right)^{m-1} r^{m-1} \quad (3.47)$$

ist. Wenn die Funktion $s(m) = \frac{1}{m} \left(\frac{m}{m-1} \right)^{m-1} \leq 1$ ist, ist die zweite Ungleichung der zu zeigenden Aussage erfüllt. Um dies zu erreichen, betrachten wir die Funktionen

$$\begin{aligned} s(x) &= \frac{1}{x} \left(\frac{x}{x-1} \right)^{x-1} \\ t(x) &= \log(s(x)). \end{aligned} \quad (3.48)$$

Nach dem Differenzieren dieser Funktionen erhalten wir

$$\begin{aligned} s''(x) &= \frac{s^2(x)t''(x) + (s')^2(x)}{s(x)} \\ t''(x) &= \frac{x-2}{(x-1)x^2}, \end{aligned} \quad (3.49)$$

was impliziert $s''(x) \geq 0$ für $x \geq 2$. Aus diesem Grund ist s eine konvexe Funktion. Nun brauchen wir nur noch zu beachten, dass $s(2) = 1$ und $\lim_{x \rightarrow +\infty} s(x) = 0$. Damit haben wir $S \leq H^{m-1}$ gezeigt.

Die Gleichheit zwischen den Ungleichungen gilt aber nur, wenn der betrachtete Punkt eine isotrope Umgebung besitzt.

□

Satz 6 Wenn $K = 0$ ist, dann gilt $S \leq \frac{1}{m} \left(\frac{m}{m-1} \right)^{m-1} H^{m-1}$.

Beweis

Dies lässt sich direkt aus Gleichung (3.47) ableiten.

□

Aus numerischer Sicht ist ein Test auf null nicht sinnvoll und mit den Erkenntnissen aus Satz (5) bietet sich ein Vergleich zwischen den Werte K und S als Konfidenztest an. Anhand dieser Tatsachen lässt sich nun die Konfidenzbedingung ($K = 0$ und $S \neq 0$) in das Konfidenzmaß $K^{\frac{1}{m}} \ll S^{\frac{1}{m-1}}$ bzw. $K^{\frac{1}{m}} < \epsilon S^{\frac{1}{m-1}}$ überführen. Der Vergleichsparameter ϵ sollte innerhalb des Intervalls $(0, 1)$ liegen. Wählen wir $\epsilon = 1$, so ist der Vergleich zwischen K und S immer erfüllt und wir erzwingen die Bewegungsbestimmung.

Der Wert $H = \frac{1}{m} \text{trace}(\mathbf{J}_N)$ entspricht der Summe aller quadrierten partiellen Ableitungen N ter Ordnung von f . Ist dieser Wert null, so kann es sein, dass sich die Grauwerte in der Umgebung des betrachteten Punktes nicht ändern. In diesem Fall ist eine Bestimmung der Geschwindigkeiten nicht möglich. Das Konfidenzmaß $H > \epsilon$ soll nun ausschließen, dass versucht wird, Bewegungen an einem Punkt zu bestimmen, an dem keine Bewegung vorkommt.

In Anhang A werden wir zeigen, wie die hier vorgestellten Konfidenzkriterien sowie der Eigenvektor zum Eigenwert null für bis zu vier Bewegungen analytisch berechnet werden können und somit keine numerische Methoden zur Bestimmung der Eigenwerte und

-vektoren benötigt werden.

Ähnlich wie bei dem Strukturtensor für eine Bewegung bzw. Orientierung geben, wie Mota et. al in weiterführenden Arbeiten [114, 113, 112] zeigen, die Eigenwerte und der damit verbundene Rang der generalisierten Strukturtenoren Aufschluss über die Art der Überlagerungen.

3.1.8 Hierarchische Bewegungsbestimmung

In der Praxis variiert die Anzahl der Bewegungen von Bildpunkt zu Bildpunkt und das zutreffende Bewegungsmodell ist für die Bestimmung der Bewegungsvektoren auszuwählen. Das Konfidenzmaß ermöglicht uns, diese Auswahl zu treffen. Da wir in der Regel einfache Bewegungen häufiger als transparente beobachten, bietet sich für die Bewegungsbestimmung ein hierarchisches Vorgehen an, das wir im Folgenden für bis zu drei Bewegungen beschreiben wollen und das im Prinzip beliebig weit fortgesetzt werden kann.

In *Abbildung (3.1)* ist die Vorgehensweise und der Aufbau des hierarchischen Algorithmuses in Form einer Prozedur dargestellt. Die Prozedur bekommt als Parameter alle Ableitungen, die schon über eine Nachbarschaft integriert wurden, sowie den Punkt, an dem die Geschwindigkeitsbestimmung ausgeführt werden soll. Ebenfalls werden die Parameter c_1 bis c_3 und ϵ übergeben. Der Ablauf geht jetzt wie folgt: Zunächst wird festgestellt, ob das H des Strukturtenors für eine Bewegung größer als ein bestimmter Schwellenwert ist. Ist dies der Fall, so wird versucht, eine Bewegung zu bestimmen. Anderenfalls wird die Bewegungsbestimmung abgebrochen, da eine homogene Grauwertumgebung vorliegt. Ist das Konfidenzmaß für eine Bewegung nicht erfüllt, so wird versucht, zwei Bewegungen zu bestimmen. Ist dies ebenfalls nicht möglich, so wird zum Schluss versucht, drei Bewegungen zu bestimmen. Man erkennt deutlich, dass diese Vorgehensweise nun beliebig lange weiter fortgesetzt werden kann.

Die Beschreibung dieses Verfahrens soll nun nicht mit der Angabe eines Flussdiagramms abgeschlossen werden, sondern noch einmal an einem Pseudoalgorithmus 1 erläutert werden. In den Zeilen 2 bis 6 werden die Ableitungen zweiter und dritter Ordnung für alle Punkte der Sequenz ausgewertet und über die Nachbarschaft integriert. Effizienter ist es jedoch, die Ableitungen zweiter und dritter Ordnung mit ihren Nachbarschaftsintegrationen erst dann auszuwerten, wenn diese benötigt werden, da in der Praxis häufiger einfache als überlagerte Bewegungen zu finden sind. Die Pseudocodedarstellung soll nur den prinzipiellen Ablauf des Verfahrens demonstrieren und ist nicht im Hinblick auf eine optimale Implementierung vorgesehen. In der Zeile 7 wird H über alle Punkte in der Sequenz bestimmt. An dieser Stelle ist es nicht notwendig, die H 's für zwei und drei Bewegungen zu bestimmen, da wir diese lediglich für die Aussage benötigen, ob hier eine Bewegungsdetektion überhaupt sinnvoll ist. Nach einem erfolgreichen Durchlauf durch die Bedingung in Zeile 10 wird zuerst versucht, in den Zeilen 11 bis 15 eine Bewegung zu bestimmen. Schlägt der Test in Zeile 13 fehl, so ist es durchaus möglich, dass in diesem Punkt der Sequenz zwei Bewegungen vorkommen. In diesem Fall wird mit den Zeilen 16 bis 20 fortgefahren und versucht zwei Bewegungen zu bestimmen. Schlägt der Test in Zeile 19 ebenfalls fehl, so könnten drei Bewegungen in der Sequenz vorkommen. Sollte

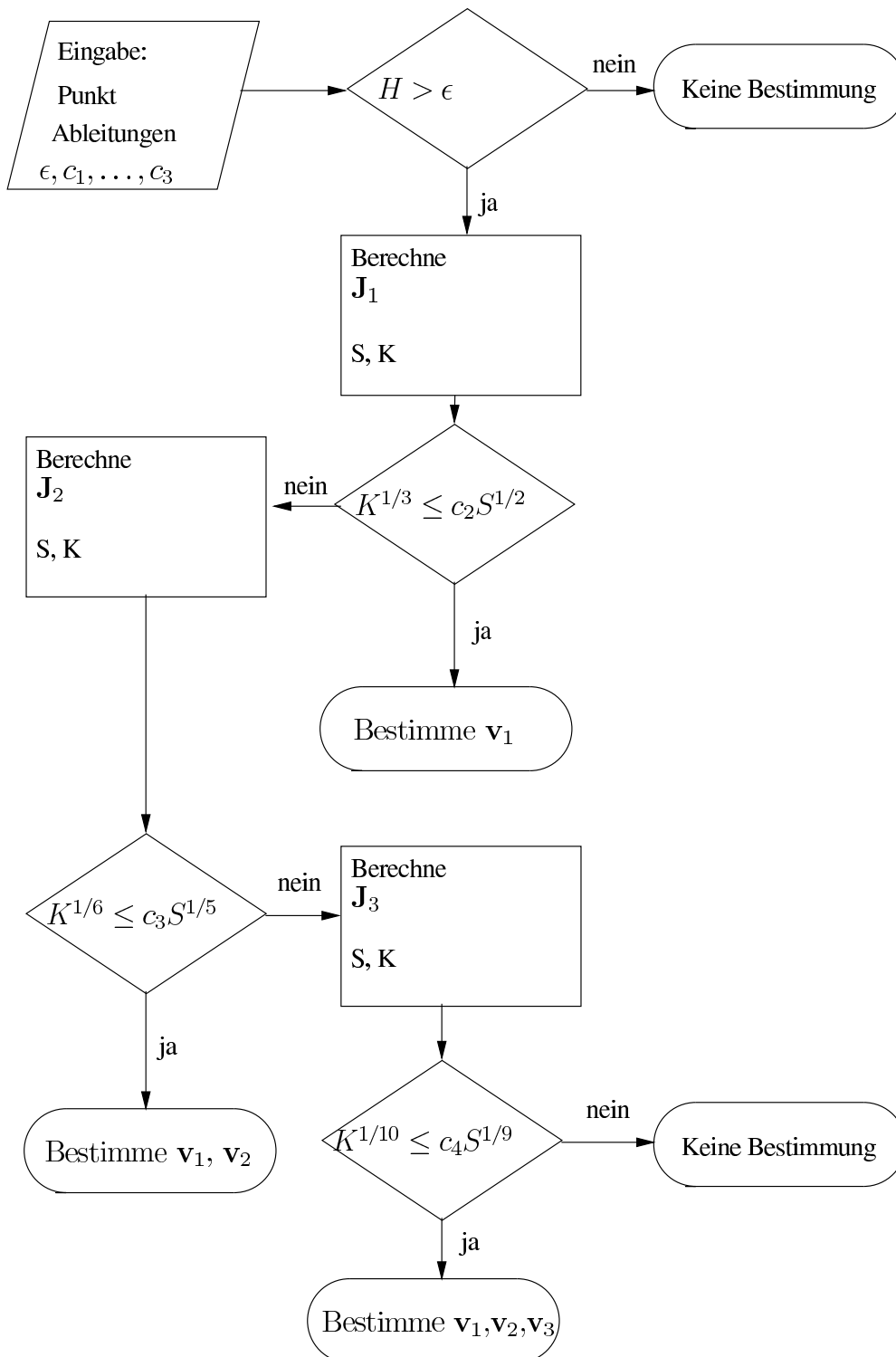


Abbildung 3.1: Flussdiagramm des hierarchischen Algorithmuses für die Berechnung bis zu drei Geschwindigkeiten in einem Punkt.

auch dies nicht der Fall sein, so brechen wir an dieser Stelle die Bewegungsbestimmung ab.

Der Vollständigkeit halber sind in Anhang C die Algorithmen für eine, zwei und drei Bewegungen noch einmal genauer beschrieben.

Algorithm 1 Hierarchischer Algorithmus

-
- 1: Bestimme die ersten partiellen Ableitungen f_x, f_y und f_t
 - 2: Bestimme die zweiten partiellen Ableitungen $f_{xx}, f_{xy}, f_{xt}, f_{yy}, f_{yt}$ und f_{tt}
 - 3: Bestimme die dritten partiellen Ableitungen $f_{xxx}, f_{xxt}, f_{xtt}, f_{yyy}, f_{yyt}, f_{ytt}, f_{xyt}, f_{xxy}, f_{xyy}$ und f_{ttt}
 - 4: Berechne die Produkte $f_x f_x, f_x f_y, \dots, f_t f_t$ für J_1 und middle sie über eine Nachbarschaft
 - 5: Berechne die Produkte $f_{xx} f_{xx}, f_{xx} f_{xy}, \dots, f_{tt} f_{tt}$ für J_2 und middle sie über eine Nachbarschaft
 - 6: Berechne die Produkte $f_{xxx} f_{xxx}, f_{xxx} f_{xxy}, \dots, f_{ttt} f_{ttt}$ für J_3 und middle sie über eine Nachbarschaft
 - 7: Berechne $H_1(x, y, t) = \text{trace}(\mathbf{J}_1(x, y, t))$ für alle Punkte (x, y, t)
 - 8: **for all** Punkte (x, y, t) **do**
 - 9: Bestimme die Eigenwerte $\lambda_1 \leq \lambda_2 \leq \lambda_3$ sowie den zu λ_1 korrespondierenden Eigenvektor \mathbf{e} von $J_1(x, y, t)$
 - 10: **if** $H_1 > \epsilon$ **then**
 - 11: $K_1 = \det(\mathbf{J}_1(x, y, t))$
 - 12: $S = \lambda_1 \lambda_2 + \lambda_1 \lambda_3 + \lambda_2 \lambda_3$
 - 13: **if** $K_1^{\frac{1}{3}} < \epsilon_2 S_1^{\frac{1}{2}}$ **then**
 - 14: Berechne $\mathbf{v}_1(x, y, t)$
 - 15: **else**
 - 16: Bestimme die Eigenwerte $\lambda_1 \leq \dots \leq \lambda_6$ sowie den zu λ_1 korrespondierenden Eigenvektor \mathbf{c} von $\mathbf{J}_2(x, y, t)$
 - 17: $K_2 = \det(\mathbf{J}_2(x, y, t))$
 - 18: $S_2 = \sum_{i=1}^6 \lambda_1 \dots \hat{\lambda}_i \dots \lambda_6$, mit $\hat{\lambda}_i = 1$
 - 19: **if** $K_2^{\frac{1}{6}} < \epsilon_3 S_2^{\frac{1}{5}}$ und $H_2 > \epsilon$ **then**
 - 20: Berechne $\mathbf{v}_1(x, y, t)$ und $\mathbf{v}_2(x, y, t)$
 - 21: **else**
 - 22: Bestimme die Eigenwerte $\lambda_1 \leq \dots \leq \lambda_{10}$ sowie den zu λ_1 korrespondierenden Eigenvektor \mathbf{c} von $\mathbf{J}_3(x, y, t)$
 - 23: $K_3 = \det(\mathbf{J}_3(x, y, t))$
 - 24: $S_3 = \sum_{i=1}^{10} \lambda_1 \dots \hat{\lambda}_i \dots \lambda_{10}$, mit $\hat{\lambda}_i = 1$
 - 25: **if** $K_3^{\frac{1}{10}} < \epsilon_4 S_3^{\frac{1}{9}}$ und $H_3 > \epsilon$ **then**
 - 26: Berechne $\mathbf{v}_1(x, y, t)$, $\mathbf{v}_2(x, y, t)$ und $\mathbf{J}_3(x, y, t)$.
 - 27: **else**
 - 28: $\mathbf{v}_1(x, y, t) = \mathbf{0}$, $\mathbf{v}_2(x, y, t) = \mathbf{0}$, $\mathbf{v}_3(x, y, t) = \mathbf{0}$
 - 29: **end if**
 - 30: **end if**
 - 31: **end if**
 - 32: **else**
 - 33: $\mathbf{v}_1(x, y, t) = \mathbf{0}$, $\mathbf{v}_2(x, y, t) = \mathbf{0}$, $\mathbf{v}_3(x, y, t) = \mathbf{0}$
 - 34: **end if**
 - 35: **end for**
-

3.2 Lösungsraum für mehrfache Orientierungen und Bewegungen

In dem vorangegangenen Kapitel hatten wir bereits gesehen, dass nicht jede Lösung der linearisierten erweiterten optischen Flussgleichung eine Lösung im Sinne der Orientierungs- bzw. Bewegungsanalyse ist. Für zwei Orientierungen konnten wir ein Entscheidungskriterium zur Überprüfung der Vereinbarkeit der Lösung mit dem Bewegungs- bzw. Orientierungsmodell angeben. Der Nullraum der Strukturtenoren stellt einen Vektorraum von gemischten Geschwindigkeitsparametervektoren dar, dessen Elemente die linearisierte Flussgleichung erfüllen und aus denen eine mit dem Modell konsistente Lösung auszuwählen ist. Im Folgenden möchten wir einige Beziehungen zwischen dem Nullraum der Tensoren und den konsistenten Lösungen genauer analysieren.

In den vorangegangenen Abschnitten haben wir gesehen, dass sich der Orientierungsvektor $\mathbf{u} = (u_x, u_y, u_t)^T$ eines Bildsignals $f(\mathbf{x})$ mittels der Bedingung

$$\mathbf{u} \cdot \nabla f(\mathbf{x}) = 0, \quad (3.50)$$

mit $\nabla f(\mathbf{x}) = (f_x, f_y, f_t)$ bestimmen lässt. Offensichtlich ist diese Darstellung nicht eindeutig, denn jedes \mathbf{u} senkrecht zu dem Gradient von f ist nach der Gleichung ein zulässiger Orientierungsvektor. Wir bezeichnen die Menge

$$E = \{\mathbf{u} : \mathbf{u} \cdot \nabla f(\mathbf{x}) = 0\} \quad (3.51)$$

als Lösungsraum von f . Indirekt haben wir schon in den vorherigen Abschnitten mit diesen Lösungsmengen gearbeitet. Wir haben um den Nullvektor auszuschließen die Nebenbedingung eingeführt, die besagt, dass die Norm des gesuchten Lösungsvektors immer eins sein soll.

Ein Signal $f(\mathbf{x})$ bestehend aus einer transparenten Überlagerung mehrerer Teilsignale $g_i(\mathbf{x})$ modellierten wir durch die Superposition

$$f(\mathbf{x}) = \sum_{i=1}^N g_i(\mathbf{x}). \quad (3.52)$$

Auch diese Darstellung ist nicht eindeutig: So kann beispielsweise der Lösungsraum von

$$f(\mathbf{x}) = g(\mathbf{u} \cdot \mathbf{x}, \mathbf{v} \cdot \mathbf{x}) = g_1(\mathbf{u} \cdot \mathbf{x}) + g_2(\mathbf{v} \cdot \mathbf{x}) \quad (3.53)$$

eindimensional bzw. durch eine Gerade darstellbar sein, während die Lösungsräume der beiden Signale g_1 und g_2 gleichzeitig zweidimensional, bzw. durch Ebenen darstellbar sind. Bei der Bewegungsbestimmung kann diese Situation beispielsweise so beschrieben werden: In den Sequenzen g_1 und g_2 wird jeweils eine Bewegung mit einem Aperturproblem beobachtet. Entsprechend sind die beiden Spektren im Frequenzraum jeweils auf eine Gerade beschränkt. Jeder Vektor, der senkrecht auf dieser Gerade steht, kommt als ein möglicher Geschwindigkeitsvektor in Frage. Folglich beschreibt der Lösungsraum aller möglichen Bewegungsvektoren eine Ebene. Addieren wir nun die beiden Sequenzen, so

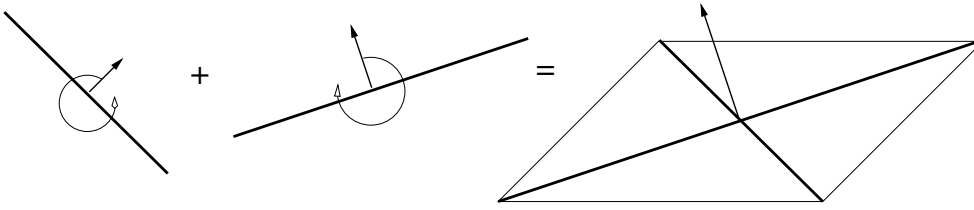


Abbildung 3.2: Die ersten beiden Grafiken zeigen zwei Spektren, die auf eine Gerade beschränkt sind. Jeder Vektor senkrecht zu den Geraden kommt als Lösungsvektor in Frage. Die Summe beider Geraden beschreibt eine Ebene. Alle Vektoren senkrecht zu dieser Ebene sind eine Lösung.

erhalten wir im Frequenzraum wieder zwei Geraden, die eine gemeinsame Ebene bilden. Der Lösungsraum dieser Ebene wird durch die Menge von Vektoren bestimmt, die senkrecht zu der Ebene sind. Demnach kann der Lösungsraum durch eine Gerade, die Normale zur Ebene, beschrieben werden¹. Die Überlagerung der beiden Bewegungen führt zu einer neuen Bildsequenz, in der nur eine, von den beiden Ausgangssequenzen unterschiedliche, Bewegung auftaucht und bei der kein Aperturproblem mehr zu beobachten ist. Die resultierende Gesamtbewegung wird häufig als kohärente Bewegung bezeichnet. Abbildung 3.2 veranschaulicht das beschriebene Beispiel noch einmal.

Sei E_n der Orientierungsraum von $g_n(\mathbf{x})$. Das Ziel der Mehrfaktororientierungsanalyse ist, die Lösungsräume E_1, \dots, E_N für gegebenes $f(\mathbf{x})$ zu bestimmen. Im Allgemeinen sind jedoch weder N noch $g_1(\mathbf{x}), \dots, g_N(\mathbf{x})$ im Voraus bekannt und können daher nicht für die Bestimmung der Lösungsräume herangezogen werden. Im Folgenden möchten wir uns auf den Fall von zwei Orientierungen, d.h. $f(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x})$ mit den Orientierungsvektoren \mathbf{u} und \mathbf{v} , beschränken. Die Analyse kann leicht auf eine beliebige Anzahl von Orientierungen erweitert werden und ist auch für die Orientierungsanalyse auf höhere Dimensionen übertragbar.

Zur Bestimmung des gemischten Orientierungsparametervektors $\mathbf{c} = \mathbf{c}(\mathbf{u}, \mathbf{v})$ hatten wir die Orientierungsbedingung durch Linearisierung in die Form

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}) = \sum_I c_I f_I = 0 \quad (3.54)$$

überführt. Die Orientierung ist eine lokale Eigenschaft. Um zu verhindern, dass sich die Vorzeichen aus Gleichung (3.54) bei der Integration über die Nachbarschaft Ω gegenseitig aufheben, wird Gleichung (3.54) vor der Integration für jeden Punkt in Ω quadriert und wir erhalten

$$\mathbf{c}^T \mathbf{J}_2 \mathbf{c} = 0, \quad (3.55)$$

wobei die Matrix $\mathbf{J}_2 = \mathbf{J}_2(f)$, der *erweiterte Strukturtenor*, durch

$$\mathbf{J}_2 = \int_{\Omega} [(f_I(\mathbf{x}))_I (f_I(\mathbf{x}))_I^T] \mathbf{d}\mathbf{x}, \quad (3.56)$$

¹An dieser Stelle muss zwischen dem Lösungsraum und der eigentlichen Bewegung unterschieden werden. Für den gesuchten Bewegungsvektor muss gelten, dass die letzte Vektorkomponente immer eins ergibt, d.h. $\mathbf{v} = (v_x, v_y, 1)^T$. Dies wird dadurch erreicht, dass wir bei einem Vektor aus dem Lösungsraum die ersten beiden Komponenten durch die letzte teilen.

gegeben ist. Der gemischte Orientierungsparametervektor $\mathbf{c} = \mathbf{c}(\mathbf{u}, \mathbf{v})$ ist die komplette implizite Darstellung der gesuchten Orientierungen. Weiter hatten wir Gleichung (3.55) im Sinne der kleinsten Quadrate gelöst:

$$\min_{|\mathbf{s}|^2=1} \mathbf{s}^T \mathbf{J}_2 \mathbf{s} \quad (3.57)$$

mit der Nebenbedingung $|\mathbf{s}|^2 = 1$, um die triviale Lösung $\mathbf{s} = \mathbf{0}$ auszuschließen. Die gesuchte Lösung ergab sich dann aus der Eigensystemanalyse von \mathbf{J}_2 . Allerdings stellt nicht jeder Lösungsvektor \mathbf{s} von Gleichung (3.57) auch eine Lösung von (3.55) und (3.54) dar. Der Lösungsraum \mathcal{S}_2 aller möglichen Lösungen von (3.57), wird durch die Eigenvektoren von \mathbf{J}_2 mit Eigenwert null aufgespannt. Dieser Lösungsraum enthält als Teilmenge

$$C = \{\mathbf{c}(\mathbf{u}, \mathbf{v}); \mathbf{u} \in E_1, \mathbf{v} \in E_2\} \quad (3.58)$$

die Lösungen von (3.55). Ein Kriterium zur Feststellung, welches Element aus \mathcal{S}_2 zu C gehört, hatten im vorangegangenen Kapitel kennen gelernt. Die Orientierungen sind nur dann eindeutig bestimmbar, wenn \mathbf{J}_2 nur einen Eigenwert gleich null besitzt und damit $\dim \mathcal{S}_2 = 1$ ist.

Für diesen Fall möchten wir den Zusammenhang zwischen C und \mathcal{S}_2 untersuchen und beginnen die Analyse mit einem Beispiel: Nehmen wir an, wir hätten eine Bildsequenz vorliegen, in der zwei Bewegungen ohne Aperturproblem auftreten, d.h.

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - \mathbf{u}t) + g_2(\mathbf{x} - \mathbf{v}t), \quad (3.59)$$

mit den Geschwindigkeiten $\mathbf{u} = (1, 0)^T$ und $\mathbf{v} = (0, 1)^T$. Demnach sind die Lösungsräume für g_1 und g_2 gegeben durch

$$\begin{aligned} E_1 &= \{(a, 0, a)^T, a \in \mathbb{R}\} \\ E_2 &= \{(0, b, b)^T, b \in \mathbb{R}\}. \end{aligned} \quad (3.60)$$

In diesem Fall lautet die Menge C

$$\begin{aligned} C &= \{(u_x v_x, u_y v_y, u_t v_t, u_x v_y + u_y v_x, u_x v_t + u_t v_x, u_y v_t + u_t v_y)^T, \mathbf{u} \in E_1, \mathbf{v} \in E_2\} \\ &= \{(0, 0, ab, ab, ab, ab)^T, a \in \mathbb{R}, b \in \mathbb{R}\}. \end{aligned} \quad (3.61)$$

Die Menge C ist ein eindimensionaler Unterraum des \mathbb{R}^6 . Da in der Bildsequenz kein Aperturproblem auftritt, wissen wir, dass $\dim \mathcal{S}_2 = 1$ ist und damit ist $\mathcal{S}_2 = C$. Diese Aussage gilt auch im allgemeinen Fall beliebiger Orientierungsrichtungen:

Satz 7 Sind $g_1(\mathbf{x})$ und $g_2(\mathbf{x})$ orientierte Signale mit den Orientierungsrichtungen \mathbf{u} und \mathbf{v} und den Lösungsräumen E_1 und E_2 für die gilt $\dim E_1 = 1$ und $\dim E_2 = 1$, dann ist $\mathcal{S}_2 = C$.

Beweis:

Nach Voraussetzung gilt für die Lösungsräume von g_1 und g_2

$$\begin{aligned} E_1 &= \{\lambda_1 \mathbf{u}, \lambda_1 \in \mathbb{R}\} \text{ und} \\ E_2 &= \{\lambda_2 \mathbf{v}, \lambda_2 \in \mathbb{R}\}, \end{aligned} \quad (3.62)$$

die von den Orientierungs- bzw. Basisvektoren \mathbf{u} und \mathbf{v} erzeugt werden. Aus $\dim E_1 = 1$ und $\dim E_2 = 1$ folgt, dass jeder Lösungsraum eine Ebene im Frequenzraum repräsentiert, innerhalb derer die Spektralanteile der Sequenz ungleich null sein dürfen. Da $\mathbf{u} \neq \mathbf{v}$ muss eine, bis auf einen Skalierungsfaktor eindeutig bestimmte, quadratische Form existieren, die beide Ebenen simultan beschreibt. Die quadratische Form ist durch $\mathbf{C}_s = 1/2(\mathbf{u}\mathbf{v}^T + \mathbf{v}\mathbf{u}^T)$ gegeben. Jede Veränderung in einer ihrer Matrixeinträge verletzt die notwendige Ebeneneigenschaft und damit wäre $\boldsymbol{\omega}^T \mathbf{C}_s \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0$ nicht mehr für alle Frequenzen erfüllt. Multiplikationen von \mathbf{C}_s mit einem Wert $\lambda \in \mathbb{R}, \lambda \neq 0$ sind jedoch davon ausgenommen, da sie lediglich zu $\lambda \boldsymbol{\omega}^T \mathbf{C}_s \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0$ führen. Entsprechend besitzt \mathbf{J}_2 nur einen Eigenwert gleich null und damit ist $\dim \mathcal{S}_2 = 1$. Der eindimensionale Lösungsraum \mathcal{S}_2 enthält nach Voraussetzung den Vektor $\mathbf{c}(\mathbf{u}, \mathbf{v})$ und wir können ihn daher als

$$\mathcal{S}_2 = \{\lambda \mathbf{c}(\mathbf{u}, \mathbf{v}), \lambda \in \mathbb{R}\}, \quad (3.63)$$

schreiben, in dem wir den Vektor $\mathbf{c}(\mathbf{u}, \mathbf{v})$ als den Basisvektor des Lösungsraumes verwenden. Die Menge C wird durch die Lösungen aus E_1 und E_2 konstruiert. Die Komponenten der Elemente aus C bestehen immer aus einer der beiden prototypischen Formen

$$u_i v_i, \quad \text{oder} \quad u_i v_j + u_j v_i \quad \text{mit } i \neq j \quad (3.64)$$

Jede Multiplikation von \mathbf{u} mit einem Skalar bewirkt dieselbe Multiplikation in C mit dem gleichen Skalar. Das Gleiche gilt auch für \mathbf{v} . Damit können wir die Menge C als

$$C = \{\lambda \mathbf{c}(\mathbf{u}, \mathbf{v}), \lambda \in \mathbb{R}\} \quad (3.65)$$

schreiben, wobei $\mathbf{c}(\mathbf{u}, \mathbf{v})$ der Basisvektor von C ist, der sich aus den Vektoren \mathbf{u} und \mathbf{v} ergibt. Der Vergleich von C und \mathcal{S}_2 zeigt, dass $C = \mathcal{S}_2$ ist.

□

Diese Aussage ist für uns besonders wertvoll. Haben wir den Eigenvektor zu dem einfachen Eigenwert null bestimmt, brauchen wir nur ihn auf die Übereinstimmung der Lösung mit dem Orientierungs- bzw. Bewegungsmodell zu überprüfen. Automatisch werden damit entweder alle Lösungen von \mathcal{S}_2 akzeptiert oder verworfen und eine Suche nach konsistenten Lösungen innerhalb von \mathcal{S}_2 ist nicht erforderlich. Falsche Lösungen treten nur auf, wenn das Spektrum keine Ebenenstruktur aufweist und damit kein orientiertes Signal vorliegt. Betrachten wir dazu noch einmal das Beispiel des vorangegangenen Kapitels in Abbildung 2.1. Hat das Spektrum $F(\boldsymbol{\omega})$ nur an dem abgebildeten Doppelkegel von null verschiedene Werte so gilt $\boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} F(\boldsymbol{\omega}) = 0$. Der Ausdruck $\boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} = 0$ oder äquivalent $10\omega_x^2 + 10\omega_y^2 - \omega_t^2 = 0$ entspricht dem abgebildeten Doppelkegel. Eine Multiplikation dieser Quadrik mit einer beliebigen Konstante ändert nichts an ihrer Form. Daher gilt $\dim \mathcal{S}_2 = 1$. Größer kann die Dimension von \mathcal{S}_2 nicht werden, denn sonst ändern sich in der Gleichung die Verhältnisse der quadrierten ω_i zueinander und damit auch die Form und Lage der Quadrik.

Obiger Satz ist auch für mehr als nur zwei Überlagerungen gültig. Aus der Fouriertransformierten $(\mathbf{v}_1 \cdot \boldsymbol{\omega}) \cdots (\mathbf{v}_N \cdot \boldsymbol{\omega}) F(\boldsymbol{\omega})$ entnehmen wir, dass sich die gemischten Orientierungsparameter immer aus Produkten der Form $(\mathbf{v}_1)_i (\mathbf{v}_2)_j \cdots (\mathbf{v}_N)_k, i, j, \dots, k \in x, y, t$ bzw. aus Summen von diesen zusammensetzen. Die Lösungsräume der Schichten sind nach

Voraussetzung eindimensional und werden von den Orientierungsvektoren aufgespannt, d.h.

$$E_i = \{\lambda \mathbf{v}_i, \lambda \in \mathbb{R}\}, \quad i = 1, \dots, N. \quad (3.66)$$

Jede Multiplikation des Vektors \mathbf{v}_i mit einem Faktor bewirkt die Multiplikation des Basisvektors $\mathbf{c}(\mathbf{v}_1, \dots, \mathbf{v}_N)$ von C mit dem gleichen Faktor und verdeutlicht, dass C ein eindimensionaler Unterraum ist. Da die Dimensionen der Lösungsräume E_i eins sind, ist die Lösung bis auf einen Skalierungsfaktor eindeutig bestimmt und damit $\dim \text{kern}(\mathbf{J}_N) = 1$. Der Vektor $\mathbf{c}(\mathbf{v}_1, \dots, \mathbf{v}_N)$ ist aufgrund der eindeutigen Lösbarkeit ein Element des Kerns von \mathbf{J}_N und $\text{kern}(\mathbf{J}_N) = C$.

Im zweiten Beispiel weisen wir nach, dass die Gleichheit zwischen \mathcal{S}_2 und C nicht immer gilt. In der Bildsequenz treten wieder zwei Bewegungen auf, jedoch seien die Geschwindigkeiten hier beide null und beide Schichten weisen ein Aperturproblem auf, so dass sich die Bildsequenz als

$$f(\mathbf{x}, t) = g_1(x) + g_2(y) \quad (3.67)$$

schreiben lässt. Für die Schicht g_1 ist auf Grund des Aperturproblems der Geschwindigkeitsanteil in y -Richtung frei wählbar und für g_2 der x -Anteil. Die entsprechenden Lösungsräume sowie die Menge C lauten

$$\begin{aligned} E_1 &= \{(0, k, t), k, t \in \mathbb{R}, \} \\ E_2 &= \{(l, 0, s), l, s \in \mathbb{R}\} \\ C &= \{(0, 0, st, lk, lt, ks), l, k, s, t \in \mathbb{R}\}. \end{aligned} \quad (3.68)$$

Äquivalent können wir die Menge C als

$$C = \{st \mathbf{e}_3 + lk \mathbf{e}_4 + lt \mathbf{e}_5 + ks \mathbf{e}_6, l, k, s, t \in \mathbb{R}\} \quad (3.69)$$

schreiben, wobei \mathbf{e}_i den i -ten Einheitsvektor darstellt. Durch Einsetzen des Bildmodells (3.67) in \mathbf{J}_2 errechnet man leicht

$$\mathcal{S}_2 = \{a \mathbf{e}_3 + b \mathbf{e}_4 + c \mathbf{e}_5 + d \mathbf{e}_6, a, b, c, d \in \mathbb{R}\}. \quad (3.70)$$

\mathcal{S}_2 und C werden von den gleichen Vektoren aufgespannt. Der Vektor $\mathbf{s} = \mathbf{e}_3 + \mathbf{e}_4 + \mathbf{e}_5$ ist ein Element von \mathcal{S}_2 aber kein Element von C , denn sonst müsste $st \neq 0 \wedge lk \neq 0 \wedge lt \neq 0 \wedge ks = 0$ gelten. Damit ks gleich null wird, muss dann entweder $k = 0$ oder $s = 0$ sein und damit wären $st = 0$ oder $lk = 0$. In diesem Fall gilt also $C \subset \mathcal{S}_2$.

Die Menge C ist zwar eine Teilmenge von \mathcal{S}_2 aber kein Unterraum. Um dies nachzuweisen, wählen wir die beiden Vektoren $\mathbf{s}_1 = \mathbf{e}_3$ und $\mathbf{s}_2 = \mathbf{e}_4 + \mathbf{e}_5$ aus C und es gilt $\mathbf{s} = \mathbf{s}_1 + \mathbf{s}_2$. Der Vektor \mathbf{s}_1 liegt in C , denn wir können ihn durch die Wahl von $s = t = 1$ und $l = k = 0$ erhalten. Der Vektor \mathbf{s}_2 liegt ebenfalls in C und wir bekommen ihn durch die Wahl von $l = k = t = 1$ und $s = 0$. Da \mathbf{s} jedoch nicht in C liegt ist damit gezeigt, dass C kein Unterraum sein kann.

Wie dieses Beispiel zeigt, ist die Bestimmung der Orientierungsräume E_i kein einfaches Problem. Für überlagerte Bewegungen im Dreidimensionalen oder zwei Orientierungen im n -dimensionalen Raum konnten Methoden zur Bestimmung der Orientierungsvektoren und den damit verbundenen Orientierungsräumen angegeben werden, vorausgesetzt die verwendeten Tensoren haben nur einen Eigenwert gleich null.

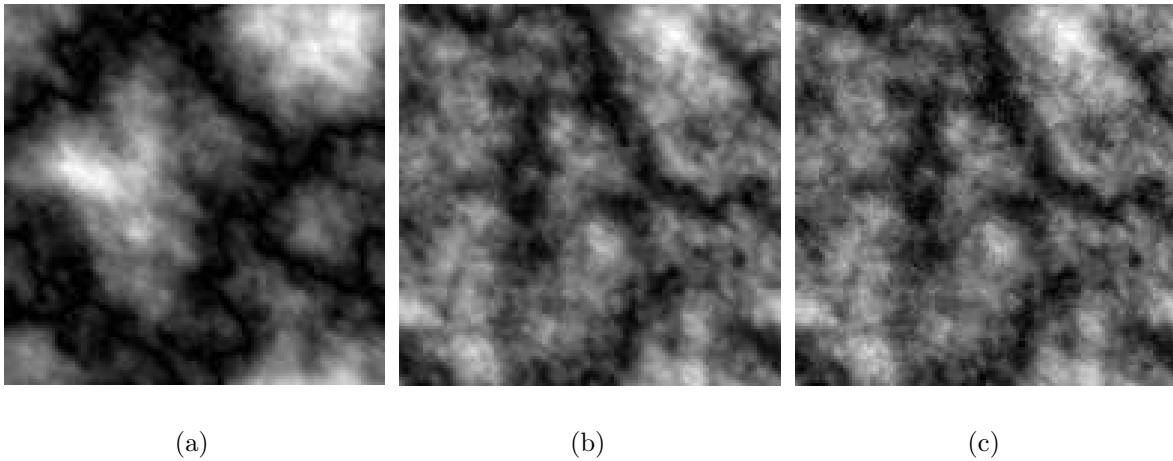


Abbildung 3.3: Testsequenzen für die Fehleranalyse: (a) Bild einer Schicht, (b) Bild einer Testsequenz ohne Rauschen und (c) mit 15dB Signalrauschverhältnis

3.3 Ergebnisse

Diesen Abschnitt möchten wir mit einer Fehleranalyse beginnen. Die Bestimmung transparenter Bewegungen ist unter anderem auch für die medizinische Röntgenbildgebung interessant. Medizinische Bilddaten zeichnen sich vielfach durch hohes Rauschen aus. Da sich Rauschen oft negativ auf die Ableitungsberechnung auswirkt, möchten wir im Folgenden eine Fehleranalyse für zwei transparente Bewegungen durchführen und dabei den Aspekt unterschiedlicher Rauschstärken berücksichtigen. Für jede Testsequenz ermitteln wir die Fehler für unterschiedliche Integrationsfenstergrößen und Ableitungsoperatoren. Die hier gewonnenen Erkenntnisse können bei der Wahl der Parameter für ein gegebenes Problem hilfreich sein.

In Abbildung 3.3 sind einige Testsequenzen mit unterschiedlichen Signalrauschverhältnissen für die anschließende Fehleranalyse dargestellt. Natürliche Bilder besitzen eine $1/f$ -Verteilung der Spektralanteile [60], wobei f die Frequenzvariable entlang einer beliebigen Geraden durch den Ursprung darstellt. Zur Simulation von realen Bilddaten erzeugten wir daher Zufallsbilder, deren Spektralkoeffizienten einen solchen $1/f$ -Verlauf besitzen. Bild (a) zeigt eines dieser Bilder. Aus je zwei dieser Bilder generierten wir Bildsequenzen, die anschließend überlagert und mit gaußverteilterm weißen Rauschen verrauscht wurden. Bild (b) zeigt eine dieser Testsequenzen ohne Rauschen und (c), zum Vergleich, eine mit einem Signalrauschverhältnis von 15dB.

Die in Abbildungen 3.4, 3.5 und 3.6 angegebenen Fehlerergebnisse sind über mehrere Tests mit sechs unterschiedlichen Bewegungsvektorkombinationen, jeweils zwei verschiedene aus der Menge $\{(u_x, u_y)^T : u_x, u_y \in \{-1, 0, 1\}\}$, gemittelt worden. Bei der Fehleranalyse haben wir bewusst auf Subpixelgeschwindigkeiten verzichtet, um Interpolationsartefakte auszuschließen. Ebenfalls reicht die Beschränkung der Geschwindigkeiten auf eine Größenordnung von einem Bildpunkt pro Bild aus, da sich die differentielle Bewegungsbestimmung bekanntlich am Besten für langsame Geschwindigkeiten im Subpixelbereich

eignet. Dennoch lassen sich mit ihnen in der Praxis auch schnelle Bewegungen durch Multi-Resolution-Ansätze behandeln. Durch Unterabtastungen werden die Bewegungen in den Subpixelbereich verschoben und dort bestimmt.

Die ermittelten Fehlerergebnisse werden im Folgenden separat für jeden Ableitungsoperator angegeben. Aufgetragen sind dort die jeweiligen absoluten Winkelfehler, deren Standardabweichung sowie die absoluten relativen Längenfehler mit ihren Standardabweichungen für unterschiedliche Integrationsfenstergrößen und Signalrauschverhältnisse. In allen Experimenten ist der Konfidenzparameter $c_2 = 0.85$ gewählt worden. Die partiellen Ableitungsfiler sind durch separierbare Funktionen dargestellt und setzen sich jeweils aus eindimensionalen Ableitungsmasken in Ableitungsrichtung und Querglättungsfilttern in den verbleibenden nicht abzuleitenden Richtungen zusammen.

Bei den in Abbildung 3.4 angegebenen Ergebnissen berechneten wir die partiellen Ableitungen erster Ordnung mit dem Ableitungsoperator $[1/2, 0, -1/2]$ und dem Bionomialfilter $[1, 2, 1]/4$ zur Querglättung. Die partiellen Ableitungen zweiter Ordnung wurden durch erneutes Anwenden der partiellen Ableitungsfiler erster Ordnung bestimmt.

Wie die Ergebnisse zeigen, nimmt bei steigendem Rauschanteil die Qualität der Ergebnisse mit zunehmender Integrationsfenstergröße zu. Die in den Grafiken für jede Kurve angegebene Zahl gibt die Größe des Integrationsfensters in x - und y -Richtung in der Einheit Bildpunkte an. In zeitlicher Richtung wurde nicht integriert. Bei den Längenfehlern beobachten wir in den Kurven ab einem gewissen Signalrauschverhältnis einen abrupten Anstieg in den Fehlern. In diesen Fällen hat das Rauschen in der lokalen Nachbarschaft die Sequenz so weit beeinflusst, dass der gemischte Geschwindigkeitsvektor keine Lösung mehr im Sinne der Bewegung ist, die Konfidenzmaße jedoch erfüllt waren. Werden an solchen Punkten dennoch die Bewegungsvektoren berechnet, so ergeben sich große Fehler. Mit zunehmender Integrationsfenstergröße verschieben sich diese Effekte zu niedrigeren Signalrauschverhältnissen. Wie die Ergebnisse zeigen, werden mit diesen Filtern gute Ergebnisse bis zu einem Signalrauschverhältnis von etwa 35dB erzielt. Mit fallendem Signalrauschverhältnis nimmt auch die Qualität der Ergebnisse ab.

Wie Abbildung 3.5 zeigt, lassen sich die Ergebnisse durch Verwendung anderer Ableitungsfiler verbessern. In dem dortigen Fall wurden die Ableitungsfiler durch eine Kombination abgeleiteter und nicht abgeleiteter eindimensionaler Gaußfunktionen mit einer Länge von 7 Bildpunkten und einer Varianz von 1.5 Bildpunkten gebildet. Gute Ergebnisse werden nun über den gesamten gemessenen Bereich von 15dB bis 50dB bei einer Fenstergröße von 11×11 Bildpunkten erzielt. Im Vergleich zu dem vorangegangenen Beispiel fallen die Ergebnisse bei schwachem Rauschen etwas schlechter, hingegen bei starkem Rauschen deutlich besser aus. Auch hier nimmt die Qualität mit steigender Integrationsfenstergröße zu.

Bei niedrigem Rauschen erwiesen sich hier die fünf Bildpunkte langen separierbaren Filter von Scharr [134, 135] als besonders günstig – siehe Abbildung 3.6. Mit fallendem Signalrauschverhältnis nehmen die Fehler jedoch schneller zu als bei den anderen beiden Verfahren.

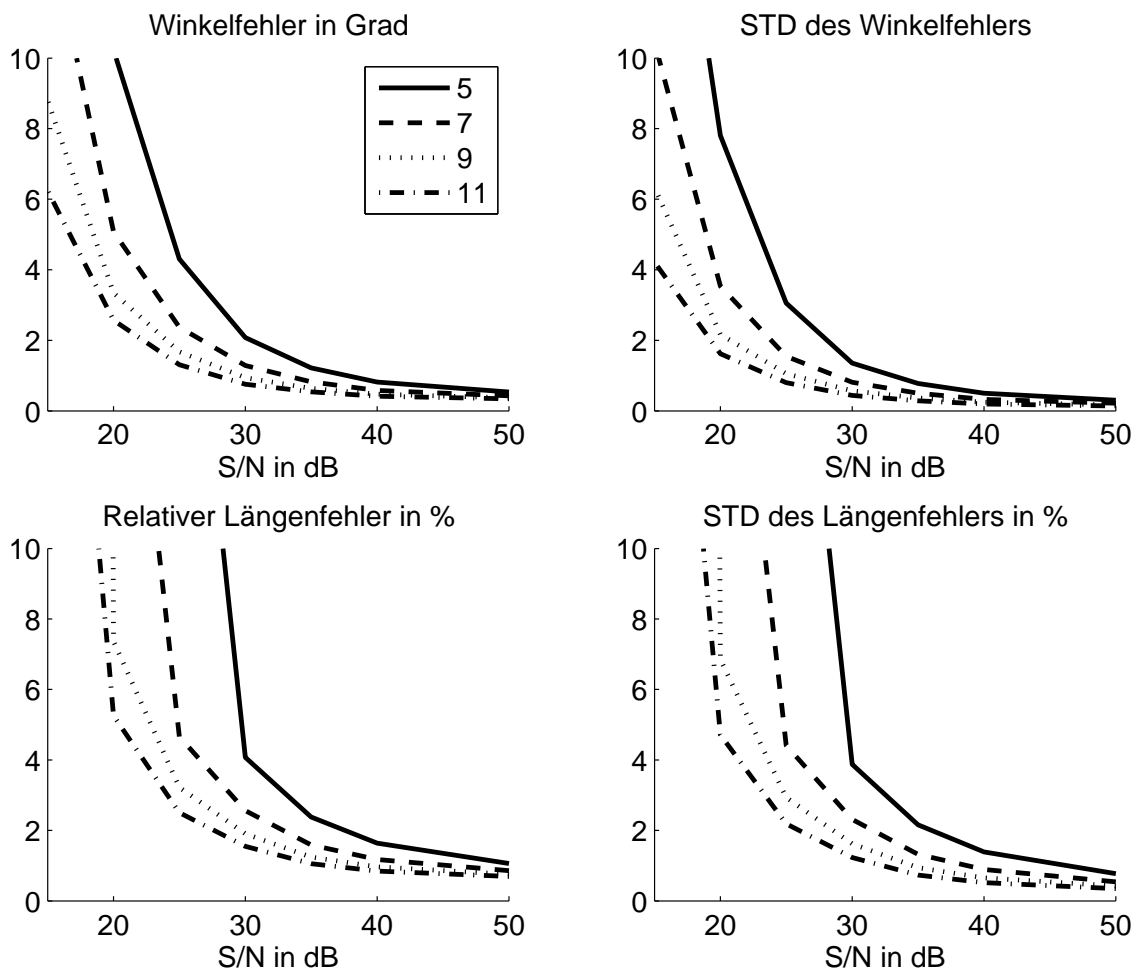


Abbildung 3.4: Ergebnisse der Fehleranalyse gerechnet mit den Binomialfiltern. (Siehe Text)

Die Wahl der Ableitungfilter und die Größe des Integrationsbereiches beeinflussen die Qualität zum Teil erheblich. Daher sollte für jede Anwendung zunächst geprüft werden, welcher Filter am Besten für die Bewegungsbestimmung geeignet ist. Der Gaußfilter stellt dabei einen guten Kompromiss zwischen Rauschrobustheit und Qualität dar. Bei starkem Rauschen liefert er bessere, bei schwachem Rauschen geringfügig schlechtere Ergebnisse als die anderen beiden Filter. Bei der Auswahl der Filter ist ebenfalls die Filtergröße zu berücksichtigen, denn mit großen Filtern dauert die Berechnung in der Regel länger als mit kleineren.

Nach der Fehleranalyse möchten wir nun anhand eines synthetischen und eines realen Beispiels zeigen, dass wir mit dem hierarchischen Verfahren für jeden Bildpunkt der Sequenz die Anzahl der überlagerten Bewegungen automatisch feststellen, das entsprechende Bewegungsmodell auswählen und damit die Bewegungsbestimmung durchführen können.

Abbildung 3.7 zeigt das Ergebnis des hierarchischen Algorithmusses für eine Sequenz

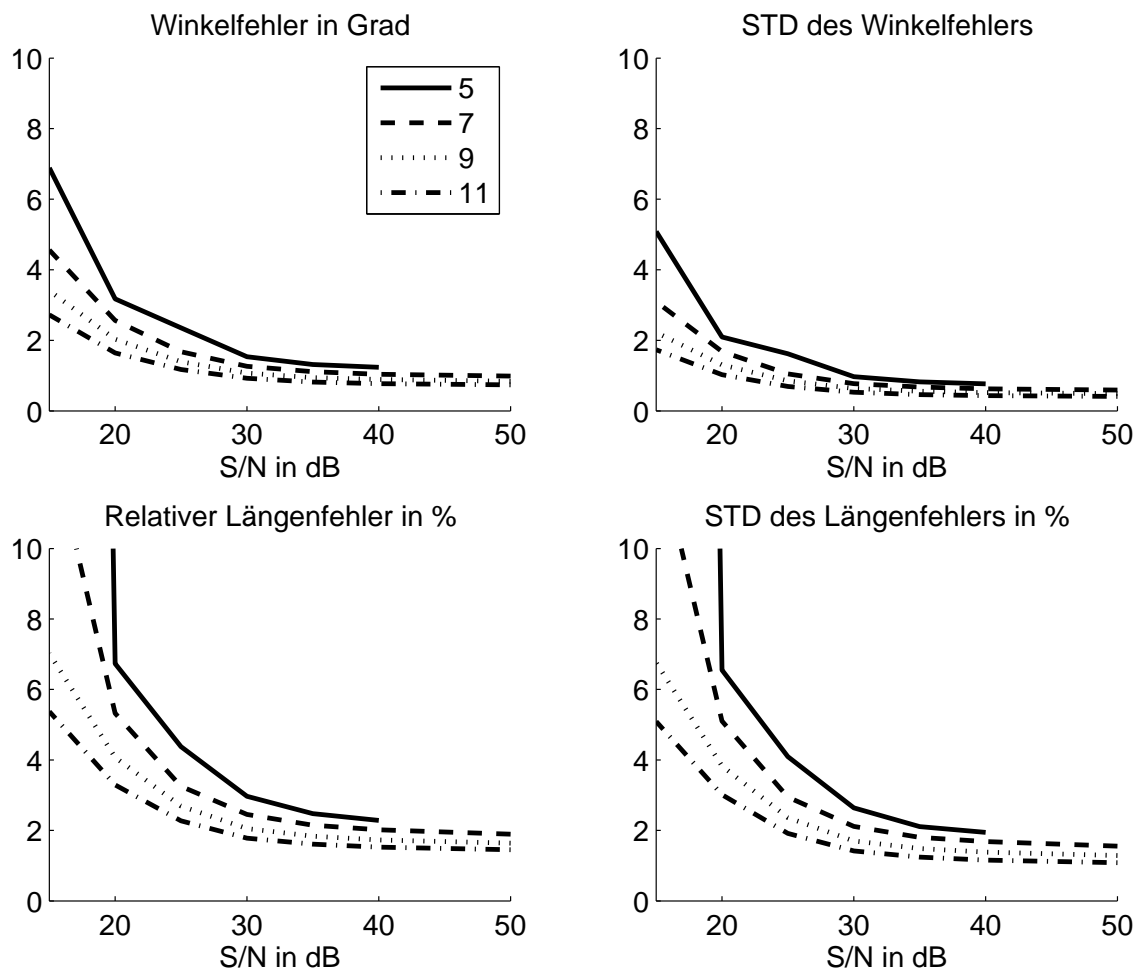


Abbildung 3.5: Ergebnisse der Fehleranalyse gerechnet mit abgeleiteten Gaußfiltern. (Siehe Text)

in der bis zu drei Bewegungen pro Bildpunkt bei einem Signalrauschverhältnis von 45dB beobachtet werden. Mit dem schwachen Rauschen simulieren wir das Kamerarauschen. Die synthetische Sequenz besteht aus drei additiv überlagerten Schichten, die sich mit den Geschwindigkeiten $\mathbf{u} = (-1, -1)^T$, $\mathbf{v} = (1, 0)^T$ und $\mathbf{w} = (0, 1)^T$ bewegen. Die Schichten sind dabei so angeordnet, dass sich das mittlere Bild der Sequenz in vier Quadranten mit unterschiedlichen Bewegungsanzahlen einteilt – siehe Abbildung 3.7(a). Die von dem Algorithmus ermittelte Anzahl der Bewegungen an jedem Bildpunkt ist in (b) in Form einer Farbkodierung dargestellt: schwarz steht für keine, dunkel grau für eine, grau für zwei und hellgrau für drei Bewegungen. Das weiße Kreuz gibt die Grenzen der Quadranten an. Die von dem Algorithmus gefundenen Bewegungsanzahlen stimmen gut mit der jeweiligen Anzahl der Bewegungen der Quadranten überein. Aufgrund der verwendeten Integrationsfensterbreite von 7×7 Bildpunkten ragen die Bereiche mit höherer Bewegungsanzahl immer etwas in die benachbarten Bereiche mit geringerer Bewegungsanzahl hinein. Die ermittelten Bewegungsvektoren für jeden fünften Bildpunkt sind in (c) dargestellt und stimmen gut mit den wahren Bewegungen überein. Auf Grund des sehr geringen Rauschens nutzten wir hier die gleichen Ableitungsfiler wie in dem Beispiel aus Abbil-

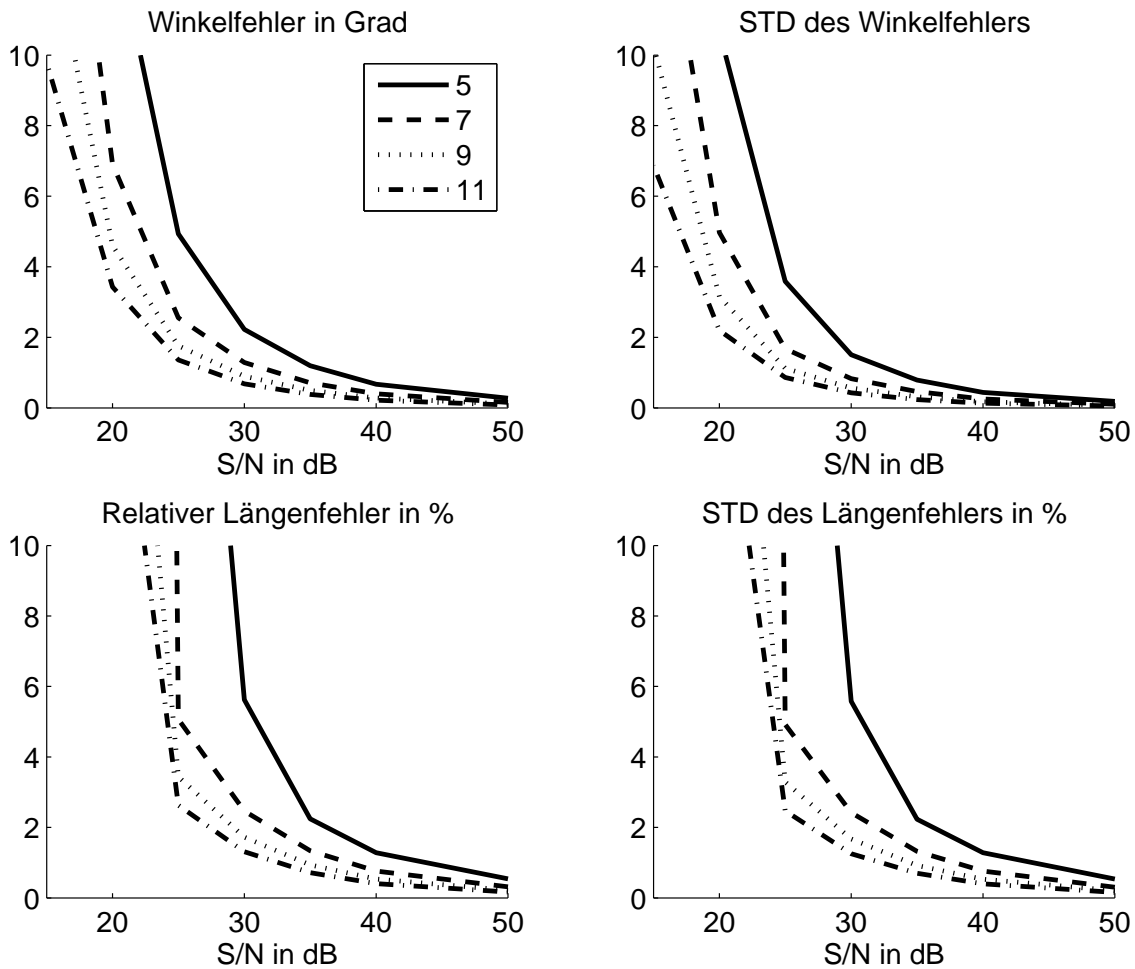


Abbildung 3.6: Ergebnisse der Fehleranalyse mit den Filtern von Scharr. (Siehe Text)

dung 3.4.

Wie Abbildung 3.8 und 3.9 verdeutlichen, liefert das hierarchische Verfahren auch unter realistischen Bedingungen gute Ergebnisse. Abbildung 3.8(a) zeigt ein Bild der Monalisa-sequenz, für das wir Bewegungsvektoren bestimmen möchten. In der Sequenz spiegelt sich eine Müslipackung in der Glasscheibe eines Bilderrahmens. Während der Aufnahme wurde nicht nur die Müslibox, sondern auch die Kamera bewegt, sodass in der Bildsequenz zwei Bewegungen beobachtet werden: eine für die Posterschicht und eine für die Spiegelungsschicht. Der Aufnahmearrangement ist in (b) veranschaulicht². In der resultierenden Sequenz bewegen sich die Müslipackung nach links und die Postertextur nach rechts. In (c) ist die Anzahl der gefundenen Bewegungen für jeden Bildpunkt der Sequenz in Form einer Farbkodierung dargestellt. Die Farbe weiß repräsentiert Punkte mit zwei, grau Punkte mit einer Bewegung und schwarz die Punkte, an denen keine Bewegung bestimmt wurde oder werden konnte. Die Bewegungsbestimmung ist nur in einem Bereich mit einem Mindestabstand von 15 Pixeln zum Rand durchgeführt worden, sodass sich ein schwar-

²Die Monalisasequenz, weitere Informationen über die Sequenzaufnahmen sowie dieses Bild finden sich unter: <http://www.ks.informatik.uni-kiel.de/~wy/motiondemo.html>

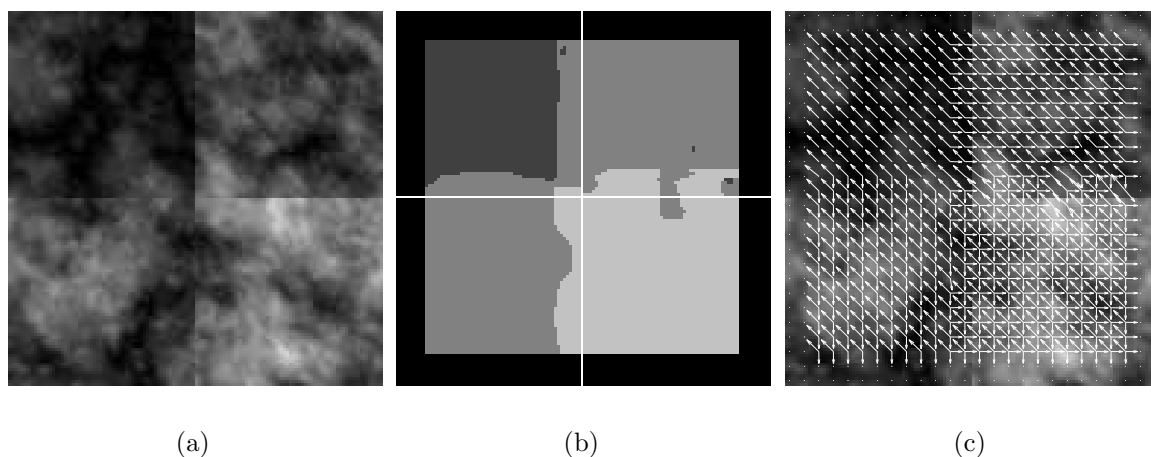


Abbildung 3.7: Synthetisches Beispiel mit Bereichen einer, zwei und drei Bewegungen.

zer Rand ergibt. Dieser Rand ist eine Folge der verwendeten Ableitungfilter sowie der großen Integrationsbereiche für die Tensoreinträge. Die Ableitungen sind nahe des Bildrandes fehlerbehaftet und können damit zu fehlerhaften Ergebnissen führen. In diesem Beispiel haben wir die gleichen Gaußableitungfilter verwendet wie bei der Fehleranalyse. Die Integration der Tensoreinträge ist durch einen $21 \times 21 \times 3$ Bildpunkte umfassenden Mittelwertfilter realisiert worden. Dieser große Integrationsbereich erlaubt trotz der vielen homogenen Bereiche und Abschnitte ein dicht besetztes Bewegungsfeld zu erhalten.

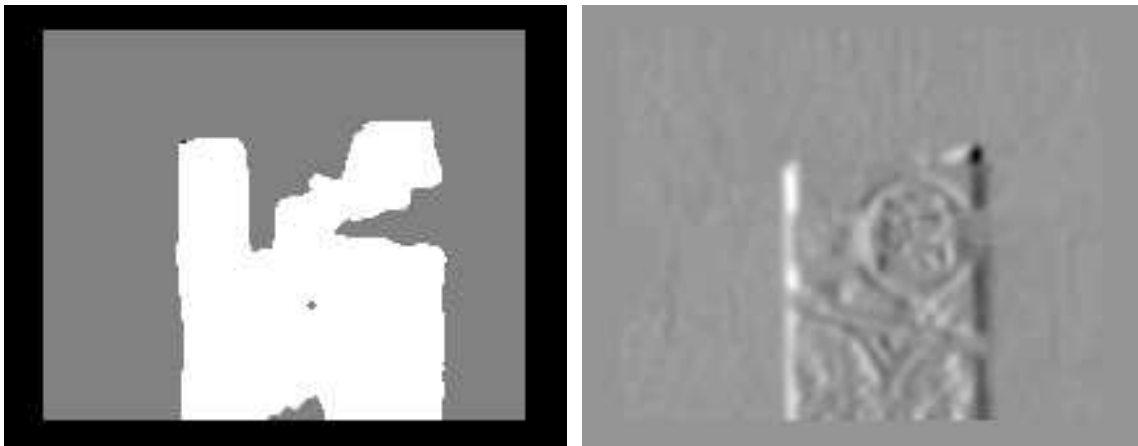
Abbildung (d) veranschaulicht die Qualität der ermittelten Bewegungsvektoren für die Monalischicht auf eine ungewöhnliche Art und Weise: Die ermittelten Bewegungsvektoren der Monalischicht wurden in die einfache optische Flussgleichung eingesetzt und dargestellt. Alle Anteile, die sich mit der eingesetzten Bewegung bewegen, werden aus der Bildsequenz entfernt. Übrig bleibt die optische Flussgleichung angewendet auf die zweite Schicht bzw. die Anteile, die sich nicht mit der Bewegung bewegen, und ihr ein reliefartiges Aussehen verleihen. Da in (d) keine Anteile der Monalischicht mehr erkennbar sind, stimmen die ermittelten Geschwindigkeiten gut mit den wahren überein.

Abbildung 3.9(a) und (b) zeigen die ermittelten Bewegungsvektoren für jeden 10ten Bildpunkt aufgeteilt nach den jeweiligen Schichten. Beide Bewegungsfelder zeigen eine gute Übereinstimmung mit den Bewegungen in der Bildsequenz. Lediglich an dem Grenzübergang zwischen einer und zwei Bewegungen können Abweichungen beobachtet werden (siehe auch (c) und (d)). Auffallend an den abweichenden Bewegungsvektoren ist ihre Orientierung entlang der Kante der Müslibox. Da die Konfidenzkriterien diese Bereiche nicht zurückwiesen, scheint an Stelle der Bewegung eine überlagerte Orientierung gefunden worden zu sein. Die Konfidenzkriterien unterscheiden nicht zwischen Bewegung und Orientierung. Abbildung (c) gibt die Richtung der gefundenen Bewegungen der Müslipackung als Absolutwert des Phasenwinkels an: schwarz bedeutet null Grad, grau ca. 90 Grad und weiß 180 Grad. Der Betrag der Bewegungsvektoren ist in (d) dargestellt. Abgesehen von den Ausreißern am Rand sind die Variationen im Bewegungsfeld gering. Für die Konfidenzkriterien verwendeten wir die Werte $c_1 = 0.69$, $c_2 = 0.71$ und $\epsilon = 0.001$.



(a)

(b)



(c)

(d)

Abbildung 3.8: Ergebniss für ein reales Beispiel. (Siehe Text)

Damit haben wir gezeigt, dass mit diesem Verfahren auch gute Ergebnisse unter realen Bedingungen erzielbar sind.

3.4 Zusammenfassung

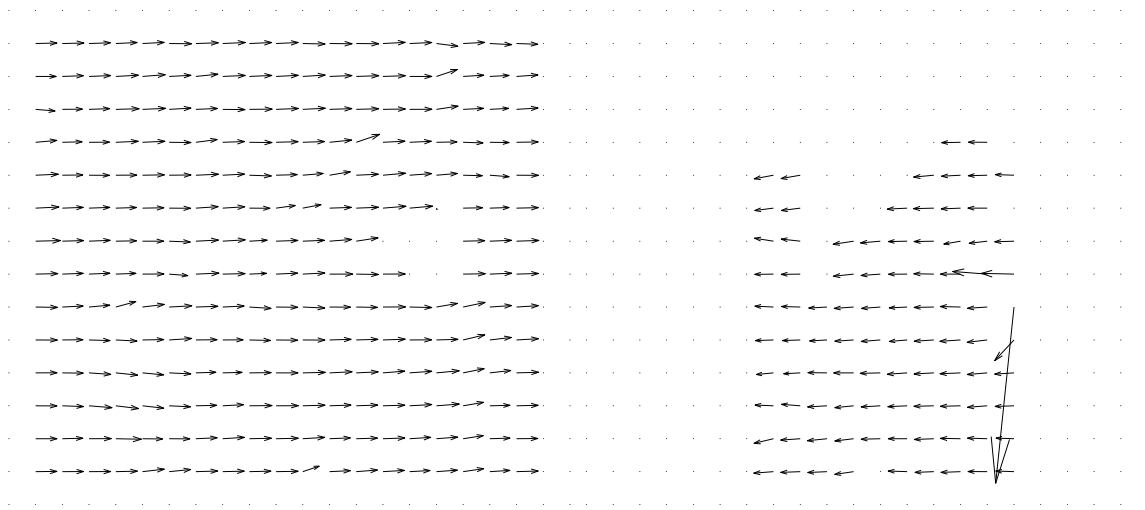
Aufbauend auf den Erkenntnissen des vorangegangenen Kapitels haben wir in diesem Kapitel ein Verfahren zur Bestimmung von transparenten Bewegungen entworfen. Das Verfahren basiert auf einer Eigensystemanalyse eines dyadischen Tensors. Die Größe des Tensors ist dabei von der jeweiligen Anzahl überlagerter Bewegungen abhängig. Der Eigenvektor zum einzigen Eigenwert null entspricht dem gesuchten gemischten Geschwindigkeitsparametervektor, aus dem sich in einem weiteren Berechnungsschritt die gesuchten Bewegungsvektoren errechnen. Aus den Eigenwerten bzw. den Invarianten der Tensoren konnten wir ein Konfidenzkriterium ableiten, mit dem wir feststellen können, ob das ge-

rade betrachtete Modell für die Beschreibung der Bewegungen in einem Punkt geeignet ist oder nicht. Dieses Konfidenzmaß ermöglicht uns die Angabe eines hierarchischen Verfahrens, das zunächst eine Bewegungsbestimmung mit dem einfachen Bewegungsmodell durchführt. Anhand des Konfidenzmaßes wird nun entschieden, ob dieses Modell anzunehmen oder zu verwerfen ist. Im ersten Fall brauchen wir nur noch den Bewegungsvektor zu bestimmen und beenden anschließend die Prozedur. Im zweiten Fall überprüfen wir die Konfidenz für zwei transparente Bewegungen und bestimmen gegebenenfalls die Bewegungsvektoren oder fahren mit der Überprüfung des nächst höheren Modells fort. Diese Kette ist nahe zu beliebig fortsetzbar. Das Verfahren selektiert demnach für jeden Bildpunkt das am besten zu treffende Modell und wir erhalten neben den Bewegungen auch Auskunft über die Anzahl der gefundenen Bewegungen in jedem Bildpunkt.

Reale und synthetische Beispiele demonstrierten die Fähigkeiten dieses Verfahrens. Ferner konnten wir zeigen, dass die Wahl der Ableitungsfiler die Ergebnisse beeinflusst. Als besonders robust zeigte sich ein auf einer Gaußfunktion basierender Filter, mit dem wir gute Ergebnisse bis zu einem Signalrauschverhältnis von 15dB erzielten.

Ferner haben wir gesehen, dass dieses Verfahren sich nicht nur für die Bewegungsbestimmung eignet, sondern auch für die Bestimmung von additiv überlagerteren orientierten Strukturen im Dreidimensionalen. Ab drei überlagerten orientierten Strukturen bleibt jedoch die Separierung der Orientierungsvektoren aus dem gemischten Orientierungsparametervektor offen. Bei Bewegungen besteht eine solche Einschränkung, wie wir gezeigt haben, nicht mehr.

Weiter zeigten wir anhand eines Beispiels, dass die Behandlung von Aperturproblemen mit dem Tensoransatz kein triviales Problem darstellt. Sollen die Normalenflüsse der Schichten ermittelt werden, so stehen wir zunächst vor der schwierigen Aufgabe eine Teilmenge die nicht notwendigerweise einen Unterraum darstellen muss, im Kern des Tensors zu identifizieren.



(a)

(b)



(c)



(d)

Abbildung 3.9: Ergebnis für ein reales Beispiel. (Siehe Text)

Kapitel 4

Regularisierung der Bewegungsfelder

Die differenziellen Methoden zur Bewegungsbestimmung lassen sich in lokale und globale Methoden einteilen. Zu den lokalen Methoden zählen zum Beispiel die Methoden von Lucas & Kanade [104, 105] und der Strukturtensoransatz von Bigün et. al. [28, 27]. Der im vorangegangenen Kapitel vorgestellte erweiterte Strukturtensor zur Bestimmung von transparent überlagerten Bewegungen ist ebenfalls dieser Klasse zuzuordnen. Bei den globalen Methoden beabsichtigt man, an Stelle von lokalen Energieausdrücken globale Energieausdrücke zu minimieren und damit ein Bewegungsvektorfeld zu bestimmen, das vom Ort $\mathbf{v}(\mathbf{x})$ und gegebenenfalls auch von der Zeit $\mathbf{v}(\mathbf{x}, t)$ abhängig ist. Als der klassische Vertreter der globalen Methoden ist hier das Verfahren von Horn & Schunck [84] zu nennen. Das Bewegungsvektorfeld $\mathbf{v}(\mathbf{x})$ bestimmen sie mit Hilfe der Variationsrechnung, wobei sie neben der Verwendung der optischen Flussgleichung noch einen weiteren Regularisierungsterm in die Minimierungsaufgabe eingehen lassen, der besagt, dass das Bewegungsvektorfeld glatt sein soll. Die Glattheit des Bewegungsvektorfeldes ist durch die partiellen Ableitungen der Bewegungsvektorkomponentenfelder ausgedrückt. Die Formulierung der Nebenbedingung ist notwendig, da die optische Flussgleichung, eine Gleichung mit zwei Unbekannten, alleine nicht ausreicht, das Bewegungsvektorfeld zu bestimmen. In der Praxis tendieren bewegte Objekte dazu, eine geschlossene Form mit glatten Bewegungsfeldern zu besitzen. Einzelne Punkte mit nicht glatten Bewegungsfeldern werden daher nur selten beobachtet, und die Forderung nach glatten dicht besetzten Bewegungsfeldern erscheint sinnvoll zu sein. Das Verfahren von Horn & Schunck [84] ist von vielen Autoren weiter entwickelt worden – [11, 122, 52, 181, 180, 143, 37, 47, 142] um nur einige zu nennen. Der Schwerpunkt der Weiterentwicklungen liegt dabei meist in der Beschreibung und Behandlung von Diskontinuitäten im Flussfeld, die durch Verdeckungen entstanden sind, oder Multigridverfahren zur effizienten Bewegungsbestimmung [72, 38, 165]. Bei transparenten Bewegungen treten glücklicherweise keine Diskontinuitäten im Flussfeld auf, da durch die Transparenzeigenschaft alle Schichten sichtbar sind.

Die lokalen Methoden sind hinsichtlich ihrer Rauschrobustheit intensiv untersucht worden [14, 59, 93, 153, 127]. Zwei vergleichende experimentelle Studien [67, 16], die auch globale Methoden berücksichtigen, zeigen: die lokalen Methoden liefern bessere Ergebnisse als die globalen Methoden. Die Ergebnisse zeigen allerdings auch, dass das resultierende Bewegungsfeld der lokalen Verfahren dünn besetzt ist, während die globalen Verfahren ein dicht besetztes Bewegungsvektorfeld ergeben.

Nicht zuletzt aus diesem Grund kombinieren Bruhn et. al [40, 39] die Vorzüge beider Verfahren miteinander. Sie ersetzen dazu den ursprünglich durch die optische Flussgleichung dargestellten Datenterm durch die Lösungsterme der Verfahren von Lucas & Kanade [104, 105] bzw. Bigün et. al. [28, 27].

Die Tatsache, dass mit den lokalen Methoden eine Bewegungsbestimmung nur an wenigen Punkten innerhalb der Bildsequenz möglich ist, ist auf homogene Grauwertbereiche und Regionen mit Aperturproblemen und gegebenenfalls Rauschen zurückzuführen. Da diese in realen Bildsequenzen häufig auftreten, müssen wir in vielen Szenarien mit transparent überlagerten Bewegungen ebenfalls von dünn besetzten Bewegungsfeldern ausgehen. In vielen Fällen wird die Anzahl der Punkte, an denen die überlagerten Bewegungen mit lokalen Methoden eindeutig bestimmbar sind, deutlich geringer ausfallen als bei Sequenzen ohne additive Überlagerungen. Ein Aperturproblem in einer der überlagerten Schichten erlaubt keine eindeutige Bewegungsbestimmung mehr.

Aus diesem Grund entwickeln wir im Folgenden eine den globalen Verfahren zuzuordnende Methode zur Bestimmung von transparent überlagerten Bewegungen, die ein dicht besetztes Bewegungsfeld ergibt [163]. Ähnlich wie bei den Verfahren von Horn & Schunck [84] wird der Datenterm des Energiefunktionals durch die erweiterte optische Flussgleichung beschrieben. Zusätzlich lassen wir einen Regularisierungsterm mit in die Minimierungsaufgabe eingehen, der besagt, dass die Bewegungsfelder glatt sein sollen. Die Glattheit wird dabei nicht direkt auf den Bewegungsvektorfeldern formuliert, sondern indirekt auf den gemischten Geschwindigkeitsparametern. Die indirekte Formulierung der Glattheitsbedingung führt zu einem linearen System von Euler-Lagrangeschen Differenzialgleichungen, das wir dann mittels des Gauß-Seidel-Verfahrens lösen.

4.1 Regularisierung für zwei Bewegungen

In diesem Abschnitt möchten wir zunächst die Regularisierung für zwei transparent überlagerte Bewegungen in Analogie zu dem für eine Bewegung von Horn & Schunck [84] vorgeschlagenen Verfahren entwickeln. Anschließend werden wir dann das hier gewonnene Ergebnis auf eine beliebige Anzahl von überlagerten Bewegungen erweitern. Im Unterschied zu dem Verfahren von Horn und Schunck wird die Glattheitsbedingung nicht direkt auf den Bewegungsvektoren sondern indirekt auf den gemischten Geschwindigkeitsparametern formuliert. Mit Hilfe dieser indirekten Formulierung erhalten wir ein in den gemischten Geschwindigkeitsparametern lineares Differenzialgleichungssystem. Am Ende der Rechnung erhalten wir die gesuchten Bewegungsvektoren wieder durch die Nullstellenbestimmung unseres komplexen Polynoms.

Die Bildsequenz für zwei transparent überlagerte Bewegungen beschreiben wir durch die Superposition

$$f(x, y, t) = g_1(x - u_x t, y - u_y t) + g_2(x - v_x t, y - v_y t) \quad (4.1)$$

der Schichten g_1 und g_2 , die sich mit den gesuchten Geschwindigkeiten \mathbf{u} und \mathbf{v} bewegen. Bisher haben wir immer konstante und damit vom Ort unabhängige Bewegungsvektoren

angenommen. Da wir im Folgenden die Bewegungen durch ein einziges globales Kriterium bestimmen möchten, betrachten wir jetzt die Bewegungsvektoren als Bewegungsfelder, die von den Variablen x und y abhängen, d.h.

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}(x, y) = \mathbf{u} \quad \text{und} \quad \mathbf{v}(\mathbf{x}) = \mathbf{v}(x, y) = \mathbf{v}. \quad (4.2)$$

Damit die erweiterte optische Flussgleichung weiterhin ihre Gültigkeit besitzt – gleiches gilt im Übrigen auch für die einfache optische Flussgleichung¹ – müssen wir jedoch sicherstellen, dass die Bewegungsvektorfelder zumindestens lokal konstant sind. Diese Aufgabe übernimmt nachher ein Glattheitsterm.

Die erweiterte optische Flussgleichung

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f = f_{xx}c_{xx} + f_{yy}c_{yy} + f_{xy}c_{xy} + f_{xt}c_{xt} + f_{yt}c_{yt} + f_{tt}c_{tt} = 0 \quad (4.3)$$

für zwei Bewegungen stellt eine Gleichung mit den fünf Unbekannten

$$\begin{aligned} c_{xx} &= u_x v_x & , & & c_{yy} &= u_y v_y \\ c_{xy} &= u_x v_y + u_y v_x & , & & c_{xt} &= u_x + v_x \\ c_{yt} &= u_y + v_x & , & & c_{tt} &= 1 \end{aligned} \quad (4.4)$$

dar. Um die Bewegungsvektoren bestimmen zu können, benötigen wir vier weitere Gleichungen. Zur Konstruktion weiterer Gleichungen bedienen wir uns der Variationsrechnung und führen in Analogie zu Horn & Schunck [84] die Nebenbedingung, die beide Bewegungsfelder berücksichtigt und mit der Glattheitsforderung konform ist,

$$\begin{aligned} N &= (\partial_x u_x)^2 + (\partial_y u_x)^2 + (\partial_x u_y)^2 + (\partial_y u_y)^2 \\ &+ (\partial_x v_x)^2 + (\partial_y v_x)^2 + (\partial_x v_y)^2 + (\partial_y v_y)^2 \end{aligned} \quad (4.5)$$

ein. Sind die Bewegungsfelder \mathbf{u} und \mathbf{v} glatt, so sind es auch die c 's. Um dies zu veranschaulichen, betrachten wir exemplarisch die Glattheit der gemischten Geschwindigkeitsparameter c_{xx} , c_{xy} und c_{xt} , der Glattheitsnachweis für die übrigen gemischten Geschwindigkeitsparameter ist in vergleichbarer Form durchführbar, da sie identische Strukturen aufweisen. Das Feld \mathbf{u} ist glatt, falls an jedem Punkt $\partial_x u_x = \partial_y u_x = \partial_x u_y = \partial_y u_y = 0$ gilt. Ein c_* ist ideal glatt, falls $\partial_x c_* = \partial_y c_* = 0$ ist. Mit den Beziehungen (4.4) folgt aus

$$\begin{aligned} \partial_x c_{xx} &= \partial_x u_x v_x + u_x \partial_x v_x = 0, & \partial_y c_{xx} &= \partial_y u_x v_x + u_x \partial_y v_x = 0, \\ \partial_x c_{xt} &= \partial_x u_x + \partial_x v_x = 0, & \partial_y c_{xt} &= \partial_y u_x + \partial_y v_x = 0, \\ \partial_x c_{xy} &= \partial_x u_x v_y + u_x \partial_x v_y + \partial_x u_y v_x + u_y \partial_x v_x = 0, \\ \partial_y c_{xy} &= \partial_y u_x v_y + u_x \partial_y v_y + \partial_y u_y v_x + u_y \partial_y v_x = 0, \end{aligned} \quad (4.6)$$

¹Eine Ersetzung des Bewegungsvektors $\mathbf{v} = (u, v)^T$ durch ein Bewegungsfeld $\mathcal{V} = (u(x, y), v(x, y))$ im einfachen Bewegungsmodell $f(x, y, t) = g(x - u(x, y)t, y - v(x, y)t)$ ergibt bei Anwendung der optischen Flussgleichung $f_x u(x, y) + f_y v(x, y) + f_t = -t(v(x, y)(g_y v_y(x, t) + g_x u_y(x, y)) + u(x, y)(g_x u_x(x, y) + g_y v_x(x, y)))$. Sind die partiellen Ableitungen $u_x(x, y) = u_y(x, y) = v_x(x, y) = v_y(x, y) = 0$, ist die rechte Seite der Gleichung null und die optische Flussgleichung erfüllt. Sollen auch lokal größere Änderungen im Bewegungsfeld zugelassen werden, so ist die rechte Seite entsprechend zu berücksichtigen.

dass auch die c 's glatt sind. Diese notwendige Beziehung erlaubt es uns die Nebenbedingung als

$$N = (\partial_x c_{xx})^2 + (\partial_y c_{xx})^2 + (\partial_x c_{yy})^2 + (\partial_y c_{yy})^2 + (\partial_x c_{xy})^2 + (\partial_y c_{xy})^2 \\ + (\partial_x c_{xt})^2 + (\partial_y c_{xt})^2 + (\partial_x c_{yt})^2 + (\partial_y c_{yt})^2 \quad (4.7)$$

zu schreiben und damit indirekt durch die gemischten Geschwindigkeitsparameter auszudrücken. Mit Hilfe dieser Nebenbedingung und der quadrierten erweiterten optischen Flussgleichung (4.3) erhalten wir die gesuchten Geschwindigkeitsparameterfelder $\mathbf{c} = (c_{xx}, \dots, c_{yt})$ durch Minimieren des Energiefunktional

$$E(\mathbf{c}) = \int_{\Omega} (\alpha(\mathbf{u})\alpha(\mathbf{v})f)^2 + \lambda^2 N \, \mathbf{d}\mathbf{x}, \quad (4.8)$$

wobei hier über den gesamten Bildbereich Ω integriert wird. Der Regularisierungsparameter λ beschreibt dabei den Einfluss der Nebenbedingung. Minimiert wird hier über die gemischten Geschwindigkeitsparameter c und nicht über die Bewegungsvektorkomponenten von \mathbf{u} und \mathbf{v} . Diese Vorgehensweise hat den Vorteil, dass das resultierende Euler-Lagrangesche Differentialgleichungssystem linear in den gemischten Geschwindigkeitsparametern ist. Würden wir die Minimierung direkt auf den Bewegungsvektorkomponenten ausführen, so wäre, wie wir noch sehen werden, das resultierende Differentialgleichungssystem nichtlinear. Aus Gleichung (4.8) ergeben sich die fünf Euler-Lagrangeschen Differentialgleichungen

$$\begin{aligned} f_{xx}^2 c_{xx} + f_{xx} f_{yy} c_{yy} + f_{xx} f_{xy} c_{xy} + f_{xx} f_{xt} c_{xt} + f_{xx} f_{yt} c_{yt} + f_{xx} f_{tt} &= \lambda^2 \Delta c_{xx} \\ f_{yy} f_{xx} c_{xx} + f_{yy}^2 c_{yy} + f_{yy} f_{xy} c_{xy} + f_{yy} f_{xt} c_{xt} + f_{yy} f_{yt} c_{yt} + f_{yy} f_{tt} &= \lambda^2 \Delta c_{yy} \\ f_{xy} f_{xx} c_{xx} + f_{xy} f_{yy} c_{yy} + f_{xy}^2 c_{xy} + f_{xy} f_{xt} c_{xt} + f_{xy} f_{yt} c_{yt} + f_{xy} f_{tt} &= \lambda^2 \Delta c_{xy} \\ f_{xt} f_{xx} c_{xx} + f_{xt} f_{yy} c_{yy} + f_{xt} f_{xy} c_{xy} + f_{xt}^2 c_{xt} + f_{xt} f_{yt} c_{yt} + f_{xt} f_{tt} &= \lambda^2 \Delta c_{xt} \\ f_{yt} f_{xx} c_{xx} + f_{yt} f_{yy} c_{yy} + f_{yt} f_{xy} c_{xy} + f_{yt} f_{xt} c_{xt} + f_{yt}^2 c_{yt} + f_{yt} f_{tt} &= \lambda^2 \Delta c_{yt} \end{aligned} \quad (4.9)$$

als notwendige Bedingung für die Variationsaufgabe.

4.1.1 Lineare und nichtlineare Formulierung

Wie bereits erwähnt ist das Gleichungssystem (4.9) linear in den c 's. Um zu zeigen, dass das bei der direkten Verwendung der Bewegungsvektorkomponenten nicht der Fall ist, möchten wir folgendes Beispiel betrachten: Zum Aufstellen des Euler-Lagrangeschen Differentialgleichungssystems muss das zu minimierende Funktional nach den unbekannt Variablen abgeleitet werden. Die Differenzierung der quadrierten optischen Flussgleichung nach der Variablen v_x ergibt

$$\frac{\partial}{\partial v_x} (\alpha(\mathbf{u})\alpha(\mathbf{v})f)^2 = \\ 2(f_{xx} u_x v_x + f_{yy} u_y v_y + f_{xy} (u_x v_y + u_y v_x) + f_{xt} (u_x + v_x) + f_{yt} (u_y + v_y) + f_{tt}) (f_{xx} u_x + f_{xy} u_y + f_{xt}) \quad (4.10)$$

und wir erhalten eine Gleichung, die nichtlinear in v_x , v_y , u_x und u_y ist. Damit wird deutlich, das resultierende Gleichungssystem ist in diesem Fall nichtlinear und damit deutlich schwieriger zu lösen.

4.1.2 Iterationsvorschrift

Zur Herleitung einer Iterationsvorschrift aus Gleichung (4.9) müssen wir das Differenzialgleichungssystem diskretisieren. Für die Diskretisierung nehmen wir an, dass das Gitternetz den Einheitsabstand habe. Im Folgenden werden wir die Gitterpunkte weiter mit den Indizes x und y bezeichnen. Damit adressieren wir beispielsweise den rechten Nachbarpunkt von (x, y) durch $(x + 1, y)$. Zur Diskretisierung von (4.9) benötigen wir eine Approximation der Laplace-Operatoren. Eine geeignete Approximation ist durch

$$\Delta c_*(x, y) \approx k(\hat{c}_*(x, y) - c_*(x, y)) \quad (4.11)$$

gegeben und in zwei Anteile aufgeteilt: in den zentralen Anteil $c_*(x, y)$ und in den lokalen Mittelwertanteil $\hat{c}_*(x, y)$. Für den Mittelwertanteil verwendeten Horn & Schunck [84] den Ausdruck

$$\begin{aligned} \hat{c}_*(x, y) = & \frac{1}{6}(c_*(x-1, y) + c_*(x, y+1) + c_*(x+1, y) + c_*(x, y-1)) + \\ & \frac{1}{12}(c_*(x-1, y-1) + c_*(x-1, y+1) + c_*(x+1, y+1) + c_*(x+1, y-1)), \end{aligned} \quad (4.12)$$

prinzipiell sind aber auch andere Realisationen denkbar. Nutzen wir diese Formel zur Berechnung des Mittelwertanteiles, so hat der Proportionalitätsfaktor k den Wert drei. Dies gilt jedoch nur unter der Annahme, dass das Diskretisierungsgitter den Einheitsabstand besitzt.

Setzen wir die Approximation des Laplace-Operators (4.11) in das Gleichungssystem (4.9) ein, so erhalten wir

$$\begin{aligned} (f_{xx}^2 + \lambda^2)c_{xx} + f_{xx}f_{yy}c_{yy} + f_{xx}f_{xy}c_{xy} + f_{xx}f_{xt}c_{xt} + f_{xx}f_{yt}c_{yt} &= \lambda^2\hat{c}_{xx} - f_{xx}f_{tt} \\ f_{yy}f_{xx}c_{xx} + (f_{yy}^2 + \lambda^2)c_{yy} + f_{yy}f_{xy}c_{xy} + f_{yy}f_{xt}c_{xt} + f_{yy}f_{yt}c_{yt} &= \lambda^2\hat{c}_{yy} - f_{yy}f_{tt} \\ f_{xy}f_{xx}c_{xx} + f_{xy}f_{yy}c_{yy} + (f_{xy}^2 + \lambda^2)c_{xy} + f_{xy}f_{xt}c_{xt} + f_{xy}f_{yt}c_{yt} &= \lambda^2\hat{c}_{xy} - f_{xy}f_{tt} \\ f_{xt}f_{xx}c_{xx} + f_{xt}f_{yy}c_{yy} + f_{xt}f_{xy}c_{xy} + (f_{xt}^2 + \lambda^2)c_{xt} + f_{xt}f_{yt}c_{yt} &= \lambda^2\hat{c}_{xt} - f_{xt}f_{tt} \\ f_{yt}f_{xx}c_{xx} + f_{yt}f_{yy}c_{yy} + f_{yt}f_{xy}c_{xy} + f_{yt}f_{xt}c_{xt} + (f_{yt}^2 + \lambda^2)c_{yt} &= \lambda^2\hat{c}_{yt} - f_{yt}f_{tt}. \end{aligned} \quad (4.13)$$

Den Faktor k haben wir an dieser Stelle nicht weiter berücksichtigt, da er mit in den Regularisierungsparameter λ^2 aufgenommen werden kann. Jede Gleichung des obigen Differenzialgleichungssystems hängt von allen gemischten Geschwindigkeitsparametern ab. Durch Lösen des Systems nach den gemischten Geschwindigkeitsparametern überführen wir es in eine äquivalente und vereinfachte Darstellung. Als Lösung für dieses Gleichungssystem

erhalten wir dann

$$\begin{aligned}
c_{xx} &= \hat{c}_{xx} - f_{xx} \frac{P}{D} \\
c_{yy} &= \hat{c}_{yy} - f_{yy} \frac{P}{D} \\
c_{xy} &= \hat{c}_{xy} - f_{xy} \frac{P}{D} \\
c_{xt} &= \hat{c}_{xt} - f_{xt} \frac{P}{D} \\
c_{yt} &= \hat{c}_{yt} - f_{yt} \frac{P}{D}
\end{aligned} \tag{4.14}$$

mit

$$\begin{aligned}
P &= f_{xx}\hat{c}_{xx} + f_{yy}\hat{c}_{yy} + f_{xy}\hat{c}_{xy} + f_{xt}\hat{c}_{xt} + f_{yt}\hat{c}_{yt} + f_{tt} \\
D &= \lambda^2 + f_{xx}^2 + f_{yy}^2 + f_{xy}^2 + f_{xt}^2 + f_{yt}^2.
\end{aligned} \tag{4.15}$$

Für jeden Bildpunkt haben wir nun fünf Gleichungen, die zu lösen sind. Aus diesen lässt sich eine Systemmatrix aufstellen, die simultan, z.B. mit dem Gauß-Jordan-Verfahren, die Differenzialgleichungen für alle Punkte löst. Die Anzahl der Zeilen und Spalten der Matrix entspricht dem Fünffachen der Anzahl der Bildpunkte und fällt damit sehr groß aus. Zusätzlich ist sie sehr dünn besetzt. Entsprechend hoch ist damit der Rechenaufwand zum Lösen dieses Systems. Auf Grund der einfachen Struktur von (4.14) bieten sich iterative Lösungsverfahren an. Mit dem Gauß-Seidel-Verfahren errechnet sich die neue Menge der gemischten Geschwindigkeitsparameter $(c_{xx}^{l+1}, \dots, c_{yt}^{l+1})$ aus den Ergebnissen des vorangegangenen Iterationsschrittes $(c_{xx}^l, \dots, c_{yt}^l)$ durch

$$\begin{aligned}
c_{xx}^{l+1} &= \hat{c}_{xx}^l - f_{xx} \frac{P}{D} \\
c_{yy}^{l+1} &= \hat{c}_{yy}^l - f_{yy} \frac{P}{D} \\
c_{xy}^{l+1} &= \hat{c}_{xy}^l - f_{xy} \frac{P}{D} \\
c_{xt}^{l+1} &= \hat{c}_{xt}^l - f_{xt} \frac{P}{D} \\
c_{yt}^{l+1} &= \hat{c}_{yt}^l - f_{yt} \frac{P}{D}
\end{aligned} \tag{4.16}$$

mit

$$\begin{aligned}
P &= f_{xx}\hat{c}_{xx}^l + f_{yy}\hat{c}_{yy}^l + f_{xy}\hat{c}_{xy}^l + f_{xt}\hat{c}_{xt}^l + f_{yt}\hat{c}_{yt}^l + f_{tt} \\
D &= \lambda^2 + f_{xx}^2 + f_{yy}^2 + f_{xy}^2 + f_{xt}^2 + f_{yt}^2.
\end{aligned} \tag{4.17}$$

Initialisiert werden die gemischten Geschwindigkeitsparameter am Anfang mit dem Wert null. Sind die gemischten Geschwindigkeitsparameter nach einer ausreichenden Anzahl von Iterationen bestimmt, so erhalten wir die gesuchten Bewegungsvektoren wieder durch die Nullstellenbestimmung des komplexen Polynoms. Es bleibt jedoch noch die Frage zu klären, wie wir mit den Randpixeln umgehen sollen.

In Bewegungszenen werden die Bewegungen in der Regel über den Bildrand hinaus fortgesetzt, da die Sequenz nur einen Ausschnitt der Realität zeigt, in der meist nur kompakte Objekte vorkommen. Damit bietet sich die Verwendung der natürlichen Randbedingungen an, die besagt, dass die Ableitung der gemischten Geschwindigkeitsparameter in Normalenrichtung zum Rand null ergeben soll. Am Bildrand werden zur Mittelung der gemischten Geschwindigkeitsparameter einige Punkte benötigt, die außerhalb des Bildes liegen. Für diese Punkte verwenden wir einfach die Werte der benachbarten Punkte am Bildrand und erfüllen damit die gewünschte Randbedingung.

4.2 Erweiterung zu N Bewegungen

Mit den gewonnenen Erkenntnissen des vorangegangenen Abschnittes gestaltet sich die Erweiterung zu einer beliebigen Anzahl von überlagerten Bewegungen einfach. Eine Anzahl von N überlagerten Bewegungen ist durch die Gleichung

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - t\mathbf{v}_1) + g_2(\mathbf{x} - t\mathbf{v}_2) + \cdots + g_N(\mathbf{x} - t\mathbf{v}_N). \quad (4.18)$$

beschrieben und die dem Bildmodell entsprechende optische Flussgleichung ist durch

$$\alpha(\mathbf{v}_1) \cdots \alpha(\mathbf{v}_N) f = \sum_I c_I f_I = 0 \quad (4.19)$$

gegeben. Summiert wird hierbei über die Menge $I = \{I_1, \dots, I_M\}$, die aus $M = (N + 1)(N + 2)/2$ geordneten Sequenzen I_i der Länge N besteht, deren Elemente aus dem Alphabet $\{x, y, t\}$ entnommen werden. Die I_i 's werden dabei so gewählt, dass sie mit den partiellen Ableitungen der Bildsequenz übereinstimmen. Als Konvention besteht I_M immer nur aus einer Abfolge des Buchstabens t , d.h. $I_M = t \cdots t$. Im Falle von zwei Bewegungen lautet zum Beispiel in Gleichung (4.4) $I_1 = xx$, $I_4 = xt$ und $I_6 = tt$.

In Analogie zum zweifachen Bewegungsfall minimieren wir jetzt das Energiefunktional

$$E(\mathbf{c}) = \int_{\Omega} \left(\sum_I c_I f \right)^2 + \lambda^2 \sum_{I \setminus I_M} ((\partial_x c_I)^2 + (\partial_y c_I)^2) \mathbf{d}\mathbf{x}. \quad (4.20)$$

Dabei verdeutlicht der Ausdruck $I \setminus I_M$ in der Summation der Nebenbedingungen den mengentheoretischen Ausschluss des Elementes I_M aus I , da per Konvention I_M nur aus einer Abfolge des Buchstaben t besteht und demnach $c_{I_M} = c_{t \cdots t} = 1$ ist und nicht geglättet werden muss. Aus Gleichung (4.20) ergeben sich die $M - 1$ Euler-Lagrangeschen Differenzialgleichungen

$$\left(\sum_I c_I f_I \right) f_{I_i} = \lambda^2 \Delta c_{I_i}, \quad i = 1, \dots, M - 1. \quad (4.21)$$

Verwenden wir wieder die Diskretisierung $\Delta c_{I_i} = \hat{c}_{I_i} - c_{I_i}$, so erhalten wir das Gleichungssystem

$$\left(\sum_{I \setminus I_M} c_I f_I \right) f_{I_i} + \lambda^2 c_{I_i} = \lambda^2 \hat{c}_{I_i} - f_{I_i} f_{I_M}, \quad i = 1, \dots, M - 1 \quad (4.22)$$

Durch Einsetzen können wir leicht nachweisen, dass

$$c_{I_i} = \hat{c}_{I_i} - f_{I_i} \frac{P}{D}, \quad i = 1, \dots, M-1 \quad (4.23)$$

mit

$$P = \sum_I \hat{c}_I f_I \quad \text{und} \quad D = \lambda^2 + \sum_{I \setminus I_M} f_I^2 \quad (4.24)$$

eine Lösung von Gleichung (4.22) ist. Da das obige System positiv definit ist, ist Gleichung (4.23) auch für mehr als nur zwei Bewegungen die einzige mögliche Lösung des Systems (4.22). Wie im Fall von zwei Bewegungen lautet die Iterationsvorschrift für das Gauß-Seidelverfahren

$$c_{I_i}^{l+1} = \hat{c}_{I_i}^l - f_{I_i} \frac{P}{D}, \quad i = 1, \dots, M-1 \quad (4.25)$$

mit

$$P = \sum_I \hat{c}_I^l f_I \quad \text{und} \quad D = \lambda^2 + \sum_{I \setminus I_M} f_I^2. \quad (4.26)$$

4.3 Ergebnisse

Abbildung 4.1 stellt das Ergebnis der Bewegungsbestimmung einer realistischen Bildsequenz dar. Grafik (a) zeigt ein Bild der Monalisasequenz, für das die Bewegungen bestimmt werden sollen. Der Aufbau der Aufnahme ist noch einmal zur Erinnerung in (b) veranschaulicht. Abbildung (e) und (f) zeigen die ermittelten Bewegungsvektoren der jeweiligen Schichten für jeden 10ten Bildpunkt nach 100 Iterationen. In jedem Punkt des Bildes sind zwei Bewegungsvektoren bestimmt worden, die gut mit den Bewegungen im Bild übereinstimmen. Die Bilder (c) und (d) verdeutlichen die Güte der Übereinstimmung auf eine etwas ungewöhnliche Art und Weise. Die ermittelten Bewegungsvektoren der jeweiligen Schichten wurden in die einfache optische Flussgleichung eingesetzt und dargestellt. Alle Anteile, die sich mit der eingesetzten Bewegung bewegen, werden aus der Bildsequenz entfernt. Übrig bleibt die optische Flussgleichung angewendet auf die zweite Schicht bzw. die Anteile, die sich nicht mit der Bewegung bewegen, und ihr ein reliefartiges Aussehen verleihen. In beiden Fällen ist jeweils der Anteil einer Schicht entfernt worden und damit stimmen die ermittelten Bewegungen gut mit den wahren Bewegungen überein. In diesen Beispielen sind die partiellen Ableitungen durch Anwenden des Gaußfilters aus dem vorangegangenen Kapitel ermittelt worden. Für den Parameter λ wählten wir den Wert 5.

Die Abbildungen 4.2, 4.3 und 4.4 zeigen die Resultate einer Fehleranalyse in Analogie zu der Fehleranalyse des vorangegangenen Kapitels. Die verrauschten Sequenzen sowie die Filter zur Berechnung der Ableitungen blieben unverändert, es wurde lediglich der Algorithmus für die Bewegungsbestimmung ausgetauscht. Damit sind beide Verfahren nun direkt vergleichbar. Im Unterschied zu dem Tensorverfahren sind hier die Fehlerkurven nicht für unterschiedliche Integrationsfenstergrößen sondern für unterschiedliche Iterationen angegeben. Der Regularisierungsparameter λ ist in allen Experimenten gleich eins gewählt. Sowohl bei der Verwendung der Binomialfilter (Abbildung 4.2) als auch bei den Filtern von Scharf (Abbildung 4.4) nimmt das Verfahren schnell ein Optimum an, eine



(a)



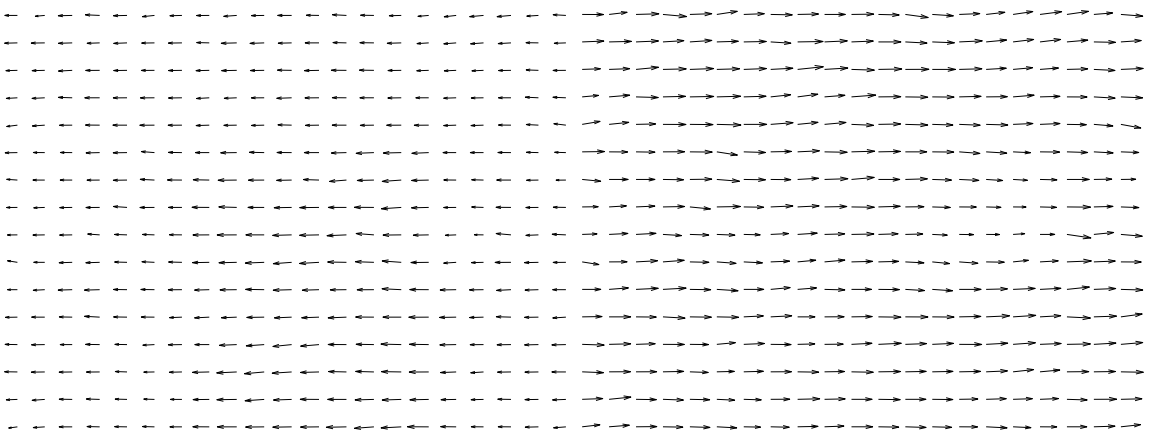
(b)



(c)



(d)



(e)

(f)

Abbildung 4.1: Ergebniss für ein reales Beispiel. (Siehe Text)

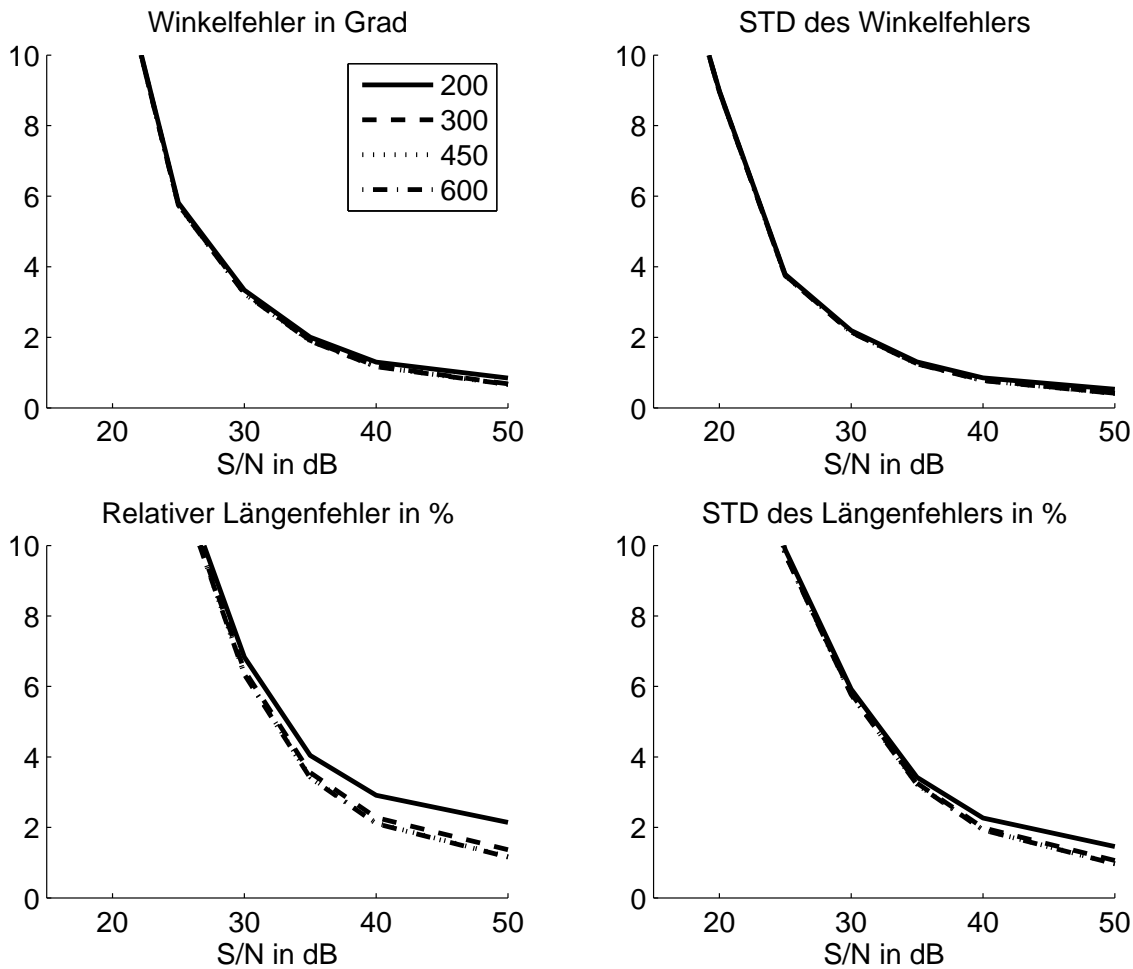


Abbildung 4.2: Ergebnisse der Fehleranalyse gerechnet mit den Binomialfiltern. (Siehe Text)

Steigerung der Iterationszahl bringt nur marginale Verbesserungen. Bei den Gaußfiltern (Abbildung 4.3) dagegen verbessern steigende Iterationszahlen die Ergebnisse. Ähnlich wie bei den Strukturtensorverfahren sind die Binomialfilter und die Filter von Scharf ab einem Signalrauschverhältnis von 35dB und mehr gut geeignet. Mit dem Gaußfilter und 600 Iterationen bekommen wir für den gesamten gemessenen Bereich gute Resultate. Im Vergleich fallen die Ergebnisse für das Tensorverfahren etwas besser aus als bei dem regularisierten Verfahren. Somit bestätigen sich die in [67, 16] für eine Bewegung gefundenen Ergebnisse auch für transparente Bewegungen: die lokalen Methoden sind günstiger als die Globalen. Bei der regularisierten Methode beobachten wir keinen plötzlichen Anstieg in den Längenfehlern wie bei den Tensorverfahren, bei denen gelegentlich das Rauschen die lokalen Strukturen soweit stört, dass die gefundenen Lösungen keine Lösung mehr im Sinne der Bewegung darstellen, obwohl sie die Konfidenzkriterien erfüllen. Durch die Glattheitsforderung der Bewegungsvektorfelder im regularisierten Verfahren werden solche Ausreißer verhindert.

Bisher haben wir immer vorausgesetzt, dass eine bestimmte Anzahl von Bewegungen

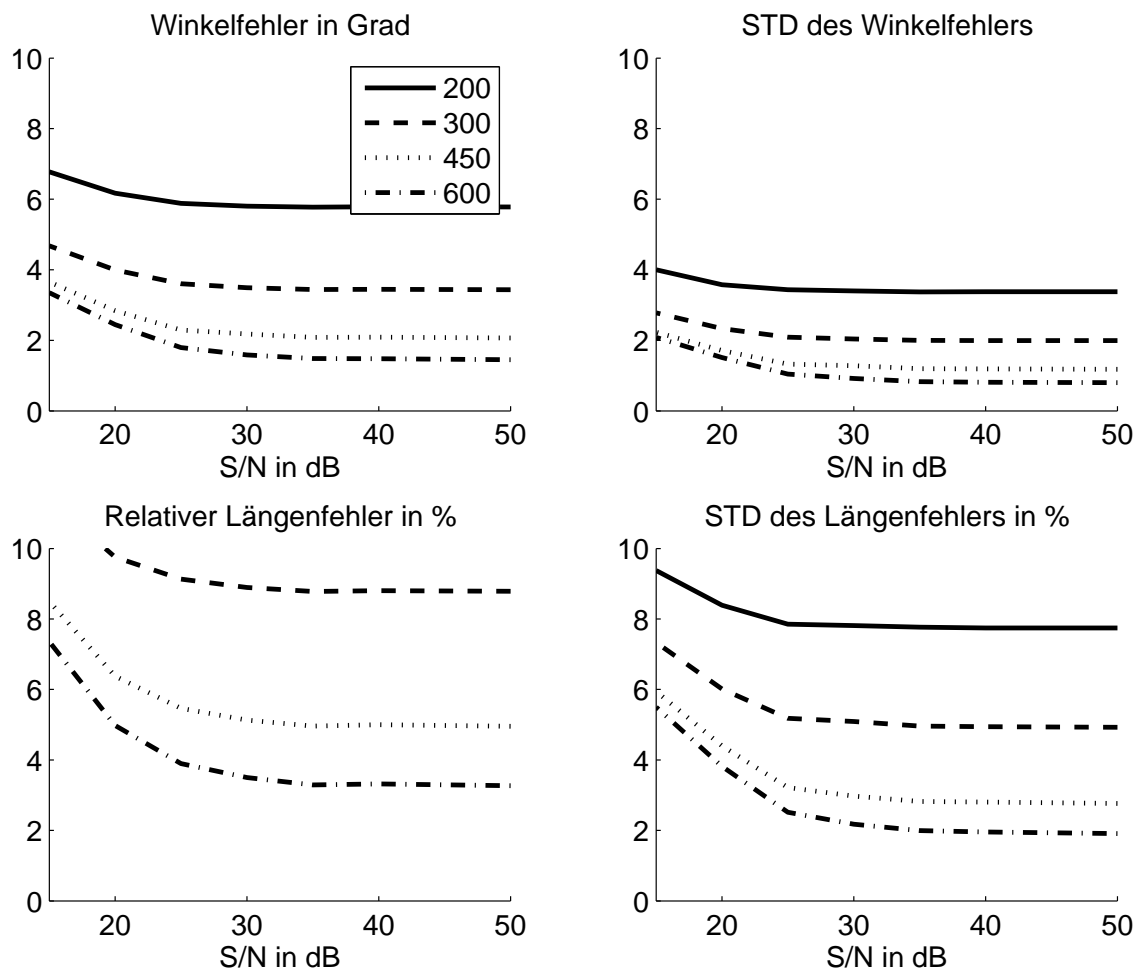


Abbildung 4.3: Ergebnisse der Fehleranalyse gerechnet mit abgeleiteten Gaußfiltern. (Siehe Text)

in der Bildsequenz vorkommt. Kommen mehr als die im Algorithmus vorgesehenen Bewegungen in der Bildsequenz vor, so können wir sie offensichtlich nicht mehr bestimmen. Treten in der Bildsequenz weniger Bewegungen auf als ursprünglich angenommen, so sind ein oder mehrere Bewegungsvektoren in der erweiterten optischen Flussgleichung frei wählbar ohne ihre Erfüllbarkeit zu verletzen. Im folgenden möchten wir daher untersuchen, wie sich das Verfahren verhält, wenn weniger Bewegungen als von dem Algorithmus vorgesehen in der Bildsequenz auftreten.

Abbildung 4.5 zeigt die Ergebnisse zweier Sequenzen, in denen nur eine Bewegung vorkommt und damit der zweite Bewegungsvektor im transparenten Modell frei wählbar ist. Die ermittelten Bewegungsfelder der ersten Sequenz nach 600 Iterationen sind in (a) und (b) für jeden 10ten Bildpunkt dargestellt. Die Sequenz ist durch Verschieben des Bildes aus Abbildung 3.3 (a) mit dem Bewegungsvektor $\mathbf{u} = (1, 0)^T$ entstanden, zu der zusätzlich ein gaußverteiltes Rauschen hinzugefügt wurde, so dass sich ein Signalrauschverhältnis von 25dB ergab. Das in Abbildung 4.5(a) abgebildete Flussfeld stimmt gut mit dem wahren Bewegungsvektor überein. Der mittlere absolute Winkelfehler liegt bei 1.7

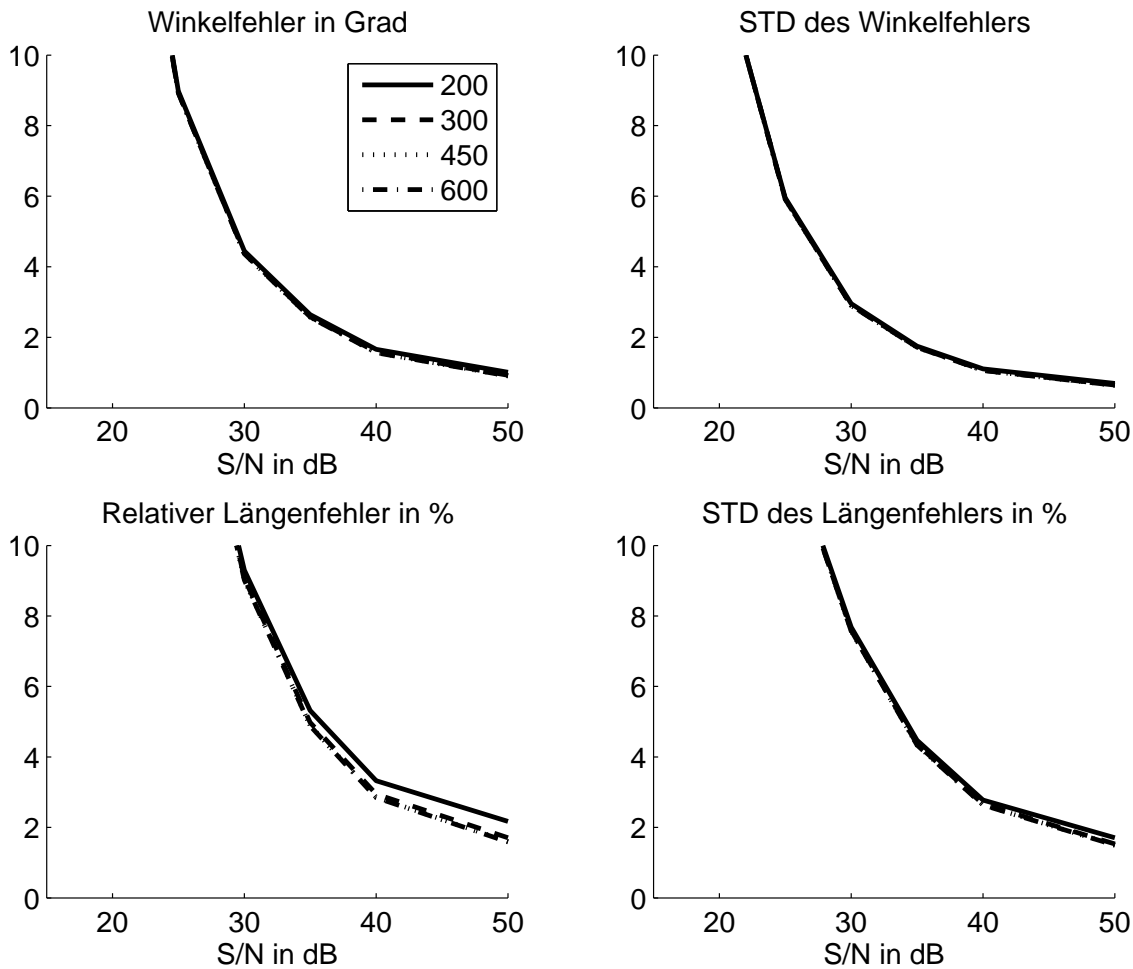


Abbildung 4.4: Ergebnisse der Fehleranalyse mit den Filtern von Scharr. (Siehe Text)

Grad mit einer Standardabweichung von 1.15 Grad, einem mittleren relativen absoluten Längenfehler bei 3.2 Prozent und einer Standardabweichung von ebenfalls 3.2 Prozent. Die ermittelten Bewegungsvektoren des frei wählbaren Feldes fallen deutlich langsamer aus und zeigen in die entgegengesetzte Richtung. Ein ähnliches Verhalten beobachten wir auch im zweiten Beispiel. Die ermittelten Bewegungsvektoren in (c) stimmen trotz des hohen Signalrauschverhältnisses gut mit der wahren Bewegung $\mathbf{v} = (1, 1)^T$ überein. Die mittleren Winkelfehler sind mit 2.6 Grad und einer Standardabweichung von 1.8 Grad etwas höher als im ersten Beispiel, decken sich aber in etwa mit den Fehlern aus den rein transparenten Fällen. Der mittlere relative Längenfehler liegt bei 5.9 Prozent mit einer Standardabweichung von 3.8 Prozent. Die Geschwindigkeiten für das frei wählbare Flussfeld sind kleiner und zeigen tendenziell in die der Bewegung entgegengesetzte Richtung. Eine ähnliche Verhaltensweise beobachteten wir auch für andere Bewegungsrichtungen. Der Parameter λ sowie die verwendeten Ableitungsfilter entsprechen denen aus der Monalisasequenz.

Das Verfahren scheint selbst dann plausible Ergebnisse zu liefern, wenn weniger Bewegungen in der Sequenz vorkommen als im Modell vorgesehen. Mit dem Strukturtensoransatz

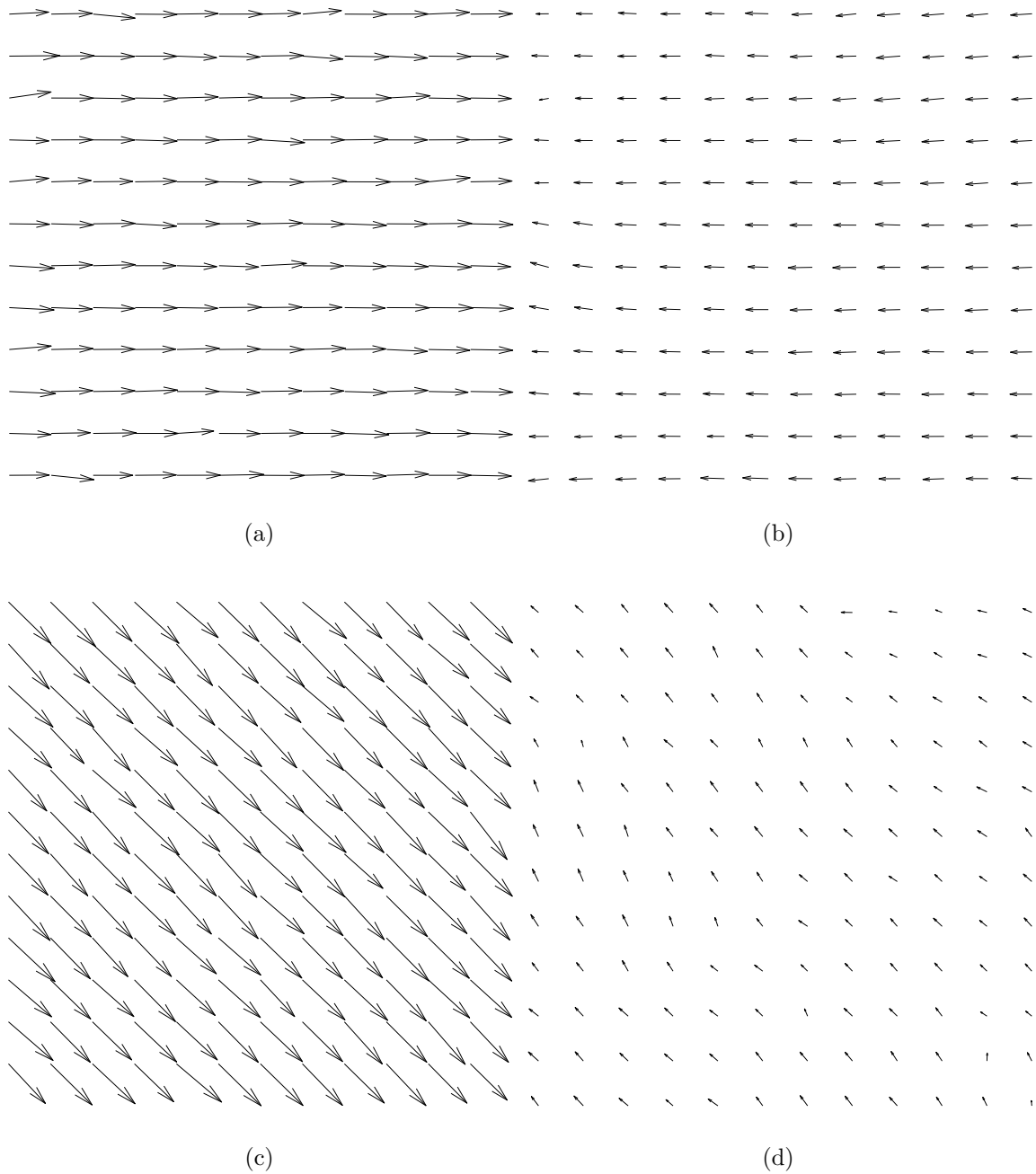


Abbildung 4.5: Ermittelte Flussfelder, in denen nur eine Bewegung vorkommt, mit dem Modell für zwei transparente Bewegungen und einem S/N von 25dB: (a) und (b) zeigen die Flussfelder für eine Sequenz, die sich mit der Geschwindigkeit $\mathbf{u} = (1, 0)^T$ bewegt und (c) und (d) für eine Sequenz mit einem Geschwindigkeitsvektor $\mathbf{u} = (1, 1)^T$.

für zwei Bewegungen können wir dieses Ergebnis nicht erzielen. Wie in [114, 113, 112] gezeigt wurde, besitzt der Strukturtenor für zwei Bewegungen angewendet auf eine Sequenz, in der nur eine Bewegung vorkommt, drei Eigenwerte gleich null und damit einen Lösungsraum, aus dem eine plausible Lösung auszuwählen wäre.

4.4 Zusammenfassung

Mit dem hier vorgestellten Verfahren zur Bestimmung von transparent überlagerten Bewegungen konnten sehr gute Ergebnisse für synthetische als auch reale Daten erzielt werden. Im Unterschied zu den Strukturtenorverfahren nehmen wir dabei eine bestimmte Anzahl von Überlagerungen an, die für den gesamten Bildbereich gilt. Eine hierarchische Vorgehensweise ist nicht erforderlich. Wie die Experimente zeigen, lassen sich selbst dann, wenn weniger Bewegungen in der Sequenz vorkommen als in dem verwendeten Modell angenommen, die beobachtbaren Bewegungen ermitteln. Mit dem lokalen Strukturtenoransatz ist das nicht so einfach möglich. Die resultierenden Flussfelder sind dicht besetzt, d.h. jedem Punkt des Bildbereiches sind ein oder mehrere Bewegungsvektoren zugeordnet. Dies ist insbesondere dann interessant, wenn in der Sequenz nur sehr wenige Punkte vorkommen, an denen alle transparenten Bewegungen mit lokalen Verfahren bestimmbar sind, und ein dichtes Bewegungsfeld gewünscht ist.

Für das Verfahren von Horn & Schunck [84] wurde erst kürzlich von Mitiche & Mansouri [111] die Konvergenz der iterativen Lösung des Differenzialgleichungssystems mit dem Gauß-Seidel-Verfahren nachgewiesen. Nur wenn sich ein Aperturproblem über die gesamte Bildebene erstreckt, konvergiert das Verfahren nicht. Ein vergleichbarer Konvergenznachweis für das transparente Verfahren bleibt noch zu führen. Auf Grund der sehr ähnlichen Struktur der Differenzialgleichungssysteme von Horn & Schnuck [84] und des präsentierten Verfahrens scheint die Konvergenz auch hier recht wahrscheinlich zu sein, könnte aber gegebenenfalls mit mehreren Ausschlussfällen einhergehen.

Kapitel 5

Erweiterungen des optischen Flusses

Die Herleitung der optische Flussgleichung basiert auf der Annahme, dass der Bildinhalt über der Zeit konstant bleibt und die beobachtbaren Veränderungen nur auf die Bewegungen zurückzuführen sind. Mathematisch ausgedrückt bedeutet dies, dass die totale Ableitung der Bildsequenz nach der Zeit null ergibt. Die beobachtbaren Intensitätsänderungen von einem Bild zum nächsten sind damit nur auf die Bewegungen zurückzuführen. Kommt es zu zusätzlichen Einflüssen, die neben der Bewegung Änderungen in den Intensitäten der Bildsequenz bewirken, so kann die Lösung der optischen Flussgleichung eine schlechte oder fehlerbehaftete Approximation der Bewegungen ergeben [35, 34, 175, 51, 46, 75, 119]. Als Ursachen für zusätzliche Intensitätsänderungen kommen zum Beispiel Beleuchtungsänderungen, Schatten, Reflexe, Blitz- und Scheinwerferlicht, Änderungen in der Objektoberflächenorientierung relativ zur gerichteter Beleuchtung [175] sowie plötzliche Änderungen im Erscheinungsbild [35, 34] eines Objektes in Frage. Insbesondere Außenaufnahmen sind einem permanenten Beleuchtungswechsel ausgesetzt. Da diese Helligkeitsänderungen die Bewegungsbestimmung und -segmentierung negativ beeinflussen können, versuchen einige Ansätze durch Intensitätsnormalisierungen [102] oder durch Abziehen der (lokalen) Mittelwerte der Bilder [154], die Bewegungsbestimmung bzw. Segmentierung robuster gegenüber Beleuchtungsänderung zu machen. Letzterer nimmt damit eine additive Zusammensetzung des Reflektanz- und Beleuchtungsanteils in der Bildsequenz an. Nicht selten scheint die Modellbeschreibung der Bildsequenz mit Hilfe von Lamberschen Oberflächen realistischer zu sein. In dieser Modellbeschreibung sind die aufgenommenen Bildintensitäten durch die Beleuchtung, die von den Objektoberflächen der Bildszene reflektiert werden, beschrieben. Für Lambersche Oberflächen ergeben sich demnach die beobachteten Intensitäten aus der Multiplikation des Beleuchtungsanteils mit der Reflektanz (siehe auch [130]). Das multiplikative Modell ist durch Logarithmieren auf das additive Modell übertragbar. Toth et. al. [172, 171] nutzen diesen Zusammenhang, um den Beleuchtungsanteil mit Hilfe von homomorphen Filtern [128] aus der Bildsequenz zu entfernen. Für die Entfernung des Beleuchtungsanteils setzt ihre Methode räumlich langsam variierende und damit tieffrequente Beleuchtungsanteile sowie mittel bis hochfrequente Reflexionsanteile voraus. Für Sequenzen, in denen sich nur der Beleuchtungsanteil über der Zeit verändert, entwarf Weiss [184] eine Methode zur Trennung von Beleuchtungs- und Reflexionsanteilen. Auch er setzt das multiplikative Beleuchtungsmodell voraus. Nach Logarithmieren aller Bilder der Bildsequenz bestimmt er die partiellen Ableitungen in x - und y -Richtung. Anschließend wendet er einen zeitlichen Medianfilter auf die jeweiligen partiellen Ablei-

tungen an und bestimmt das Reflektanzbild durch Invertierung der Ableitungsoperation.

Viele Autoren passten die optische Flussgleichung den gegebenen Beleuchtungsbedingungen durch Einführen weiterer Terme an, [46, 186, 119, 120, 68, 69, 10, 123, 193, 79] um einige zu nennen. Negahdaripour [123] stellt in diesem Zusammenhang ein allgemeines lineares Modell

$$f(\mathbf{x} - \mathbf{u}(\mathbf{x}), t + 1) = m(\mathbf{x}, t)f(\mathbf{x}, t) + c(\mathbf{x}, t) \quad (5.1)$$

vor, das viele der oben genannten Erweiterungen beinhaltet und mit dem sich sowohl multiplikative $m(\mathbf{x}, t)$ als auch additive $c(\mathbf{x}, t)$ Abweichungen von Helligkeitskonstanztannahmen beschreiben lassen. Prinzipiell können die Funktionen $m(\mathbf{x}, t)$ und $c(\mathbf{x}, t)$ beliebige Gestalt annehmen. Da es sich bei diesem Modell um eine Gleichung mit fünf Unbekannten handelt, müssen wir zur Lösung von Gleichung (5.1) weitere Anforderungen stellen. Negahdaripour [123] beschreibt ein Lösungsverfahren für obige Gleichung, bei dem er die lokale Konstanz der Funktionen $m(\mathbf{x}, t)$ und $c(\mathbf{x}, t)$ annimmt.

In einigen Anwendungen sind die zusätzlichen Helligkeitsänderungen nicht durch Änderungen in der Beleuchtung sondern durch physikalische Prozesse bedingt. Zum Beispiel stellen Haussecker [78] und Jähne et. al [68, 69] Erweiterungen der optischen Flussgleichung zur gleichzeitigen Messung der Bewegungs- und Abkühlvorgänge in Infrarotaufnahmen von Wasseroberflächen vor. Wildes et. al. [186] hingegen erweitern die optische Flussgleichung um einen Massenerhaltungsterm, um den Fluss von Flüssigkeiten in Bildsequenzen zu messen. Vereinzelt mag in solchen Anwendungen die Ermittlung der Parameter des jeweiligen physikalischen Vorganges im Vordergrund stehen und die Bewegungsbestimmung eher eine untergeordnete Rolle spielen. Eine simultane Bestimmung der physikalischen Parameter und Bewegungsvektoren erscheint dann sinnvoll. In diesem Zusammenhang entwarfen Haussecker & Fleet [79] Erweiterungen der optischen Flussgleichung zur simultanen Behandlung und Beschreibung von Bewegung und physikalisch motivierten Helligkeitsänderungen, wie z.B. exponentielle Zerfallsraten und Diffusionvorgänge. Unter anderem beschreiben sie einen Vorgang

$$f(\mathbf{x}, t) = m(\mathbf{x} - \mathbf{v}t)g(\mathbf{x} - \mathbf{u}t), \quad (5.2)$$

in dem auch die Beleuchtungsfunktion bewegt wird. Unter der Annahme, dass $m(\mathbf{x} - \mathbf{v}t)$ räumlich sehr glatt ist, können sie die Bewegung von $g(\mathbf{x} - \mathbf{u}t)$ ermitteln. Zwischen diesem Beleuchtungsmodell und dem hier vorgestellten Konzept der überlagerten Bewegungen besteht ein Zusammenhang: durch Logarithmieren kann dieses Modell in das additive transparente Bewegungsmodell überführt werden und beide Bewegungen ermitteln.

Im Unterschied zu allen hier genannten Ansätzen gehen Black et. al. [35] davon aus, dass die Intensitätsänderungen zwischen aufeinander folgenden Bildern durch eine Summen von mehreren gleichzeitig auftretenden Ereignissen entstehen. Daher beschreiben sie die Intensitätsänderungen durch ein statistisches Mixtur-Modell, das unter anderem Bewegungs- und Beleuchtungsänderungen beinhaltet und das sie mit dem EM-Algorithmus lösen.

Wie diese Diskussion zeigt, gibt es viele Gründe für zusätzliche Helligkeitsänderungen in Bildsequenzen, deren Behandlung oft problemspezifisch ist. In dem nun anschließenden

Abschnitten möchten wir zwei der wichtigsten Beleuchtungsmodelle im Zusammenspiel mit den transparenten Bewegungen genauer untersuchen. In Abschnitt 5.1 beginnen wir mit der Analyse eines um eine additive Beleuchtungsfunktion erweiterten transparenten Bewegungsmodells. Anschließend fahren wir in Abschnitt 5.2 mit einem multiplikativen Modell fort, aus dem wir dann in 5.3 ein exponentielles Helligkeitsänderungsmodell ableiten. In allen Fällen beschränken wir uns jedoch auf die Betrachtung von zwei Überlagerungen.

5.1 Additive Helligkeitsänderung

Für die Modellierung der additiven Helligkeitsänderung nehmen wir an, dass der Helligkeitsterm $k(t)$ räumlich lokal konstant ist und damit nur von der Zeit abhängt. Entsprechend ist die transparente Bewegungssequenz durch

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - t\mathbf{u}) + g_2(\mathbf{x} - t\mathbf{v}) + k(t) \quad (5.3)$$

gegeben. Wenden wir nun den Richtungsableitungsoperator $\alpha(\mathbf{u}) = \partial_x u_x + \partial_y u_y + \partial_t$ mit den Geschwindigkeiten \mathbf{u} und \mathbf{v} auf obiges Bildmodell an, so erhalten wir

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t) = \alpha(\mathbf{u})\alpha(\mathbf{v})(g_1(\mathbf{x} - t\mathbf{u}) + g_2(\mathbf{x} - t\mathbf{v})) + \alpha(\mathbf{u})\alpha(\mathbf{v})k(t) = k''(t). \quad (5.4)$$

Für lineare Helligkeitsfunktionen $k(t)$ verschwindet die zweite Ableitung. Demnach ist das transparente Bewegungsbestimmungsverfahren für zwei Bewegungen blind gegenüber linearen Helligkeitsänderungen und eine gesonderte Betrachtung additiver Beleuchtungsfunktionen ist erst ab quadratischen Helligkeitsfunktionen erforderlich. Ähnlich wie zur Bestimmung der Geschwindigkeitsvektoren im rein transparenten Fall können wir Gleichung (5.4) als inneres Produkt zweier Vektoren, d.h.

$$\mathbf{d}f \cdot \mathbf{p} = 0, \quad (5.5)$$

mit

$$\begin{aligned} \mathbf{p} &= (c_{xx}, c_{xy}, c_{yy}, c_{xt}, c_{yt}, 1, k'')^T, \\ \mathbf{d}f &= (\partial_{xx}f, \partial_{xy}f, \partial_{yy}f, \partial_{xt}f, \partial_{yt}f, \partial_{tt}f, -1)^T \end{aligned} \quad (5.6)$$

schreiben und damit mit einem angepassten Tensorverfahren lösen. Erstreckt sich der Integrationsbereich der Tensoreinträge sowohl über den Ort als auch die Zeit, setzen wir voraus, dass der Parametervektor in der räumlichen und zeitlichen Umgebung konstant ist. Folglich muss dann auch die zweite Ableitung von $k(t)$ eine Konstante ergeben und wir können mit diesem Verfahren nur quadratische Helligkeitsfunktionen behandeln. Bei rein örtlichen Integrationsbereichen müssen wir diese Forderung $k''(t)$ gleich konstant nicht stellen.

5.2 Multiplikative Helligkeitsänderung

Bei additiven Helligkeitsänderungen können wir nicht entscheiden, ob sich die Helligkeitsänderung nur auf eine der beiden Schichten bezieht oder aus der Summe der Helligkeitsfunktionen der jeweiligen Schichten entstanden ist. Bei multiplikativen Helligkeitsänderungen sind zwei Situationen denkbar. Zunächst möchten wir den Fall betrachten, in dem die multiplikative Helligkeitsänderung für beide Schichten gleich ist. Nehmen

wir wieder an, dass die Beleuchtungsfunktion lokal nur von der Zeit abhängt, so schiebt sich das Bildmodell als

$$f(\mathbf{x}, t) = (g_1(\mathbf{x} - \mathbf{v}t) + g_2(\mathbf{x} - \mathbf{u}t))k(t). \quad (5.7)$$

Nach Anwendung des optischen Flusses für zwei transparente Bewegungen auf das obige Bildmodell erhalten wir

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f = \frac{k''(t)}{k(t)}f + k'(t)((u_x - v_x)(g_{1x} - g_{2x}) + (u_y - v_y)(g_{1y} - g_{2y})), \quad (5.8)$$

mit $\alpha(\mathbf{u}) = u_x\partial_x + u_y\partial_y + \partial_t$. Im Unterschied zur reinen Transparenz entsteht auf der rechten Seite der Gleichung ein von null verschiedener Ausdruck. Da dort Ausdrücke in Abhängigkeit der partiellen Ableitungen der Schichten entstehen und wir diese nicht kennen, ist es nicht möglich, diese Gleichung direkt zur Berechnung der Bewegungen zu verwenden. Um dieses Problem zu lösen, können wir zunächst folgende Beobachtung machen:

$$f_x(u_x + v_x) + f_y(u_y + v_y) + 2f_t = \frac{2k'(t)}{k(t)}f + k(t)((u_x - v_x)(g_{1x} - g_{2x}) + (u_y - v_y)(g_{1y} - g_{2y})) \quad (5.9)$$

Kombinieren wir nun die Gleichungen (5.8) und (5.9) wie folgt miteinander

$$k(t)\alpha(\mathbf{u})\alpha(\mathbf{v})f - k'(t)(f_x(u_x + v_x) + f_y(u_y + v_y) + 2f_t) = \frac{k(t)k''(t) - 2k'(t)^2}{k(t)}f, \quad (5.10)$$

so enthält das Resultat keine partiellen Ableitungen der jeweiligen Schichten mehr und ist damit lösbar. Zur besseren Lesbarkeit können wir Gleichung (5.10) auch als

$$k(t)\alpha(\mathbf{u})\alpha(\mathbf{v})f - k'(t)(\alpha(\mathbf{v}) + \alpha(\mathbf{u}))f = \frac{k(t)k''(t) - 2k'(t)^2}{k(t)}f, \quad (5.11)$$

schreiben oder äquivalent als das inneres Produkt der Vektoren

$$\mathbf{d}f \cdot \mathbf{p} = 0, \quad (5.12)$$

mit

$$\mathbf{d}f = (f_{xx}, f_{yy}, f_{xy}, f_{xt}, f_{yt}, f_{tt}, f_x, f_y, f_t, f) \quad \text{und} \quad (5.13)$$

$$\mathbf{p} = \left(k(t)c_{xx}, k(t)c_{yy}, k(t)c_{xy}, k(t)c_{xt}, k(t)c_{yt}, k(t), \right. \quad (5.14)$$

$$\left. -k'(t)c_{xt}, -k'(t)c_{yt}, -2k'(t), -\frac{k(t)k''(t) - 2k'(t)^2}{k(t)} \right). \quad (5.15)$$

Damit lässt sich der Parametervektor \mathbf{p} zum Beispiel mit einem angepassten Strukturten-sorverfahren bestimmen. Die c 's entsprechen wieder denen aus dem transparenten Fall, d.h.

$$c_{xx} = u_x v_x, \quad c_{yy} = u_y v_y, \quad c_{xy} = u_x v_y + u_y v_x, \quad c_{xt} = u_x + v_x, \quad c_{yt} = u_y + v_y. \quad (5.16)$$

Innerhalb des Parametervektors \mathbf{p} werden die gemischten Geschwindigkeitsparameter mit der Helligkeitsfunktion $k(t)$ multipliziert. Um die Geschwindigkeitsvektoren aus den c 's zu rekonstruieren, müssen diese durch $k(t)$ geteilt werden, d.h. $\mathbf{p} = \mathbf{p}/\mathbf{p}_6$. Nach dem Skalieren des Parametervektors ist

$$\mathbf{p}_9 = 2 \frac{k'(t)}{k(t)}. \quad (5.17)$$

Um die Beleuchtungsfunktion $k(t)$ zu bestimmen, müssen wir die homogene Differenzialgleichung

$$a(t) = \frac{k'(t)}{k(t)} \quad (5.18)$$

lösen, wobei wir im Vergleich zu obiger Gleichung den zeitabhängigen Ausdruck $\mathbf{p}_9/2$ durch die zeitabhängige Funktion $a(t)$ ersetzt haben. Die Lösung dieser Differenzialgleichung ist durch

$$k(t) = k(0) \exp\left(\int_0^t a(s) ds\right) \quad (5.19)$$

gegeben. Der Anfangswert $k(0)$ wird in der Regel den Wert eins besitzen, da wir zu Beginn der Sequenz keine Helligkeitsänderungen beobachten können und ihn damit als Referenzgröße nutzen. Beabsichtigen wir, den Parametervektor \mathbf{p} durch ein angepasstes Strukturtenorverfahren zu bestimmen, müssen wir voraussetzen, dass der Vektor \mathbf{p} innerhalb des räumlichen und zeitlichen Integrationsbereiches konstant ist. Damit muss dann aber auch

$$\frac{k'(t)}{k(t)} = a = \text{konstant} \quad (5.20)$$

innerhalb dieses Fensters konstant sein. Folglich erhalten wir mit

$$k(t) = k(0) \exp(at) \quad (5.21)$$

einen Schätzwert für den Wert $k(t)$ für das zum Zeitpunkt t betrachtete Fenster. Im nächsten Abschnitt gehen wir davon aus, dass die beiden Schichten jeweils mit einer eigenen Helligkeitsfunktion multipliziert werden. Da wir für die meisten Geschwindigkeitsbestimmungsverfahren lokal konstante Parametervektoren voraussetzen müssen, werden die beiden Helligkeitsfunktionen einen in Analogie zu der hier vorgestellten Lösung exponentiellen Verlauf besitzen. Aus diesem Grund werden wir uns daher im Folgenden auf die Betrachtung exponentieller Helligkeitsfunktionen beschränken.

5.3 Exponentielle Helligkeitsänderung

Werden die beiden bewegten Schichten jeweils mit einer eigenen Helligkeitsfunktion $k_1(t)$ und $k_2(t)$ multipliziert, so können wir die Bildsequenz durch

$$f(\mathbf{x}, t) = g_1(\mathbf{x} - t\mathbf{u})k_1(t) + g_2(\mathbf{x} - t\mathbf{v})k_2(t) \quad (5.22)$$

modellieren. Zur Vereinfachung nehmen wir

$$k_1(t) \propto \exp(a_1 t) \quad \text{und} \quad k_2(t) \propto \exp(a_2 t), \quad (5.23)$$

mit den Konstanten a_1 und a_2 an und damit gilt nach [79] für die jeweiligen Schichten

$$\begin{aligned}\alpha(\mathbf{u})(g_1(\mathbf{x} - t\mathbf{u})k_1(t)) &= a_1(g_1(\mathbf{x} - t\mathbf{u})k_1(t)), \\ \alpha(\mathbf{v})(g_2(\mathbf{x} - t\mathbf{v})k_2(t)) &= a_2(g_2(\mathbf{x} - t\mathbf{v})k_2(t)).\end{aligned}\quad (5.24)$$

Beide Gleichungen stellen eine partielle Differenzialgleichung für die exponentiell abklingenden bewegten Signale dar. Ein exponentielles Abklingen der Grauwertintensitäten können wir zum Beispiel bei dem Abkühlvorgang eines zuvor erhitzten Gegenstandes beobachten [78, 68, 69]. Durch Einführen des Operators

$$\beta(\mathbf{v}, a) = v_x \partial_x + v_y \partial_y + \partial_t - a \quad (5.25)$$

können wir die obigen Gleichungen als

$$\beta(\mathbf{u}, a_1)(g_1(\mathbf{x} - t\mathbf{u})k_1(t)) = 0 \quad \text{und} \quad \beta(\mathbf{v}, a_2)(g_2(\mathbf{x} - t\mathbf{v})k_2(t)) = 0 \quad (5.26)$$

schreiben und folglich ist wegen der Kommutativität des Operators β

$$\beta(\mathbf{u}, a_1)\beta(\mathbf{v}, a_2)f = 0. \quad (5.27)$$

Zur Linearisierung schreiben wir diese erweiterte Flussgleichung wieder als das innere Produkt der Vektoren

$$\begin{aligned}\mathbf{p} &= (c_{xx}, c_{xy}, c_{yy}, c_{xt}, c_{yt}, 1, -u_x a_2 - v_x a_1, -u_y a_2 - v_y a_1, -a_1 - a_2, a_1 a_2)^T \\ \mathbf{d}f &= (f_{xx}, f_{xy}, f_{yy}, f_{xt}, f_{yt}, f_{tt}, f_x, f_y, f_t, f)^T,\end{aligned}\quad (5.28)$$

d.h. $\mathbf{d}f \cdot \mathbf{p} = 0$. Für die Bestimmung der Parameter a_1 und a_2 können wir dasselbe Verfahren anwenden wie zur Bestimmung der Geschwindigkeitsvektoren aus den gemischten Geschwindigkeitsparametern. Mit den reellen Koeffizienten

$$A_0 = a_1 a_2 = \mathbf{p}_{10}, \quad \text{und} \quad A_1 = -a_1 - a_2 = \mathbf{p}_9 \quad (5.29)$$

ergeben sich die gesuchten Parameter a_1 und a_2 aus den Nullstellen des Polynoms

$$x^2 + A_1 x + A_0 = 0. \quad (5.30)$$

Nach dem die beiden Bewegungsvektoren und Zerfallsraten bestimmt sind, stellt sich die Frage, welche Zerfallsrate zu welchem Bewegungsvektor gehört. Für die Zuordnung nutzen wir die verbleibenden Komponenten des Vektors \mathbf{p} . Ist die Zuordnung (u_x, u_y, a_1) und (v_x, v_y, a_2) korrekt, sollte sie die Bedingungen

$$-u_x a_2 - v_x a_1 - \mathbf{p}_7 \approx 0 \quad \text{und} \quad -u_y a_2 - v_y a_1 - \mathbf{p}_8 \approx 0 \quad (5.31)$$

erfüllen. Ist dies nicht der Fall so vertauschen wir a_1 und a_2 .

5.4 Ergebnisse

Abbildung 5.1 zeigt drei synthetische Sequenzen, bei denen neben der Bewegung weitere Helligkeitsänderungen durch die Verwendung unterschiedlicher Beleuchtungsfunktionen

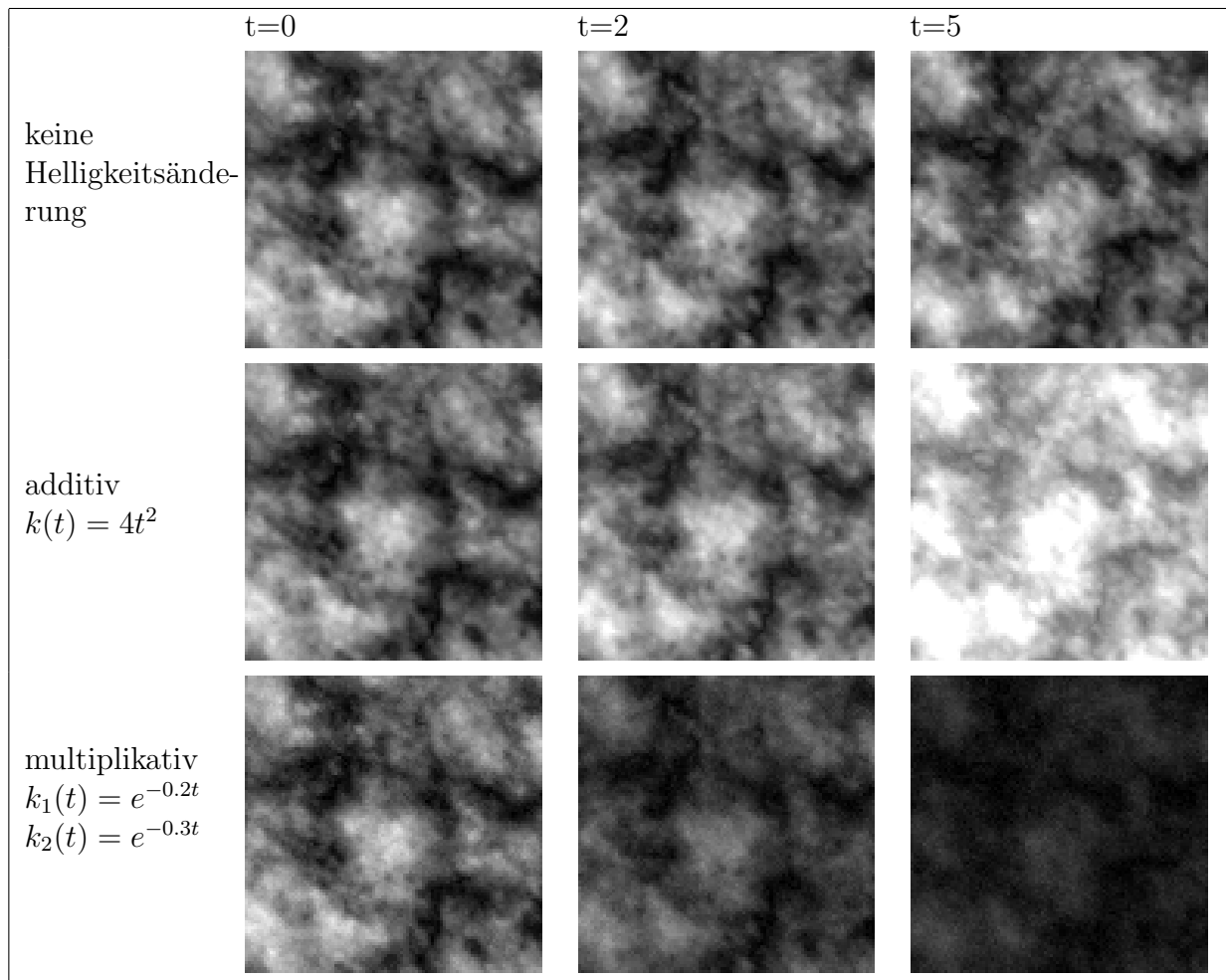


Abbildung 5.1: Jeweils drei Bilder aus den drei Testsequenzen mit keiner, additiver und multiplikativer Beleuchtungsänderung und einem Signalrauschverhältnis von 40 dB.

hervorgerufen wurden. Mit Hilfe dieser synthetischen Sequenzen werden wir die oben entwickelten Verfahren testen. In allen drei Sequenzen bewegen sich die Testmuster mit den globalen Geschwindigkeiten $\mathbf{u} = (0, -1)^T$ und $\mathbf{v} = (1, 1)^T$. Die Sequenzen sind 21 Bilder lang und der Zeitpunkt $t = 0$ befand sich genau in der Sequenzmitte, also bei Bild 11. Zum Zeitpunkt $t = 0$ sehen die Bilder der drei Sequenzen daher identisch aus. Zur Simulation von Kamerarauschen sind die Sequenzen mit einem gaussverteilten weißen Rauschen gestört worden, so dass ein Signalrauschverhältnis von 40dB entstand. Die Bilder in der ersten Zeile von Abbildung 5.1 entsprechen einer reinen transparenten Sequenz, in der es zu keiner zusätzlichen Helligkeitsänderung kommt. Als additive Beleuchtungsänderungsfunktion in der zweiten Zeile verwendeten wir die quadratische Funktion $k(t) = 4 * t^2$ und erhalten damit $k''(t) = 8$. Bei der multiplikativen Beleuchtungsänderungsfunktion wurden die Schichten mit der exponentiellen Funktion $\exp(-a_i t)$ gewichtet, wobei wir für die erste Schicht $a_1 = -0.2$ und für die zweite $a_2 = -0.3$ wählten.

Die ermittelten Ergebnisse für die Sequenzen aus Abbildung 5.1 und drei verschiedene Ableitungs-/Querglättungspaare sind in Abbildung 5.2 dargestellt. Wie die für jeden

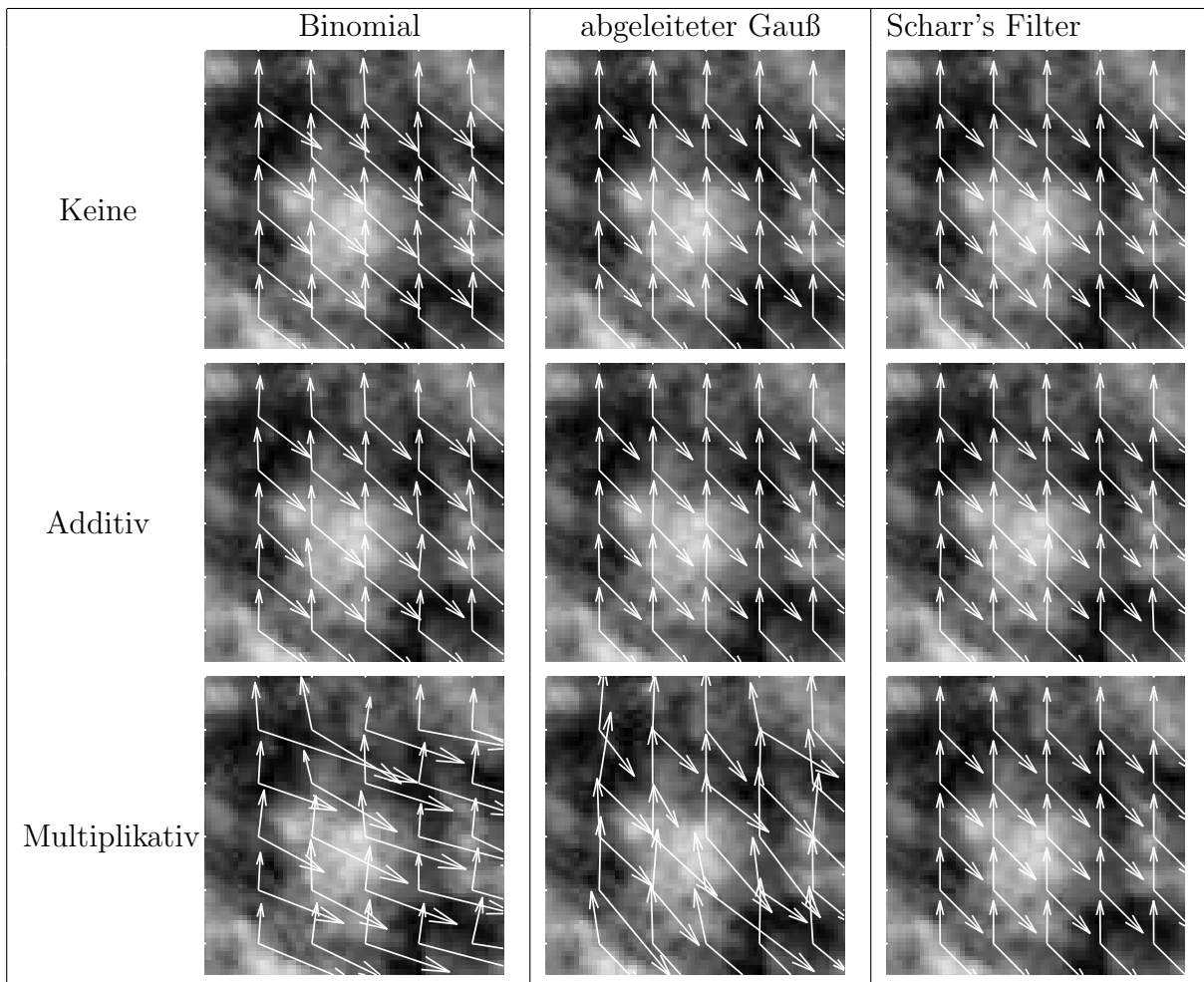


Abbildung 5.2: Testsequenzen mit keiner, additiver und multiplikativer Beleuchtungsänderung und einem Signalrauschverhältnis von 40 dB.

10ten Bildpunkt angegebenen Bewegungsvektoren verdeutlichen, scheint die Wahl der Ableitungs- und Querglättungsfilterpaare bei dem rein transparenten und dem additiven Beleuchtungsänderungsmodell keine größere Auswirkung auf die Ergebnisse zu haben. Im multiplikativen Fall hingegen scheint der Einfluss stärker zu sein. Die besten Ergebnisse wurden mit den 5 Bildpunkte langen und separierbaren Filtern von Scharr (vgl. [134, 135]) erzielt. Im Binomialfall setzen sich die Filterpaare wieder aus den separierbaren Filtermasken $[1, 0, -1]/2$ und $[1, 2, 1]/4$ zusammen, im Gaußfall bestanden die Filtermasken wieder aus sieben Bildpunkte langen abgeleiteten und nicht abgeleiteten Gaußfunktionen mit einer Standardabweichung von 1.5 Bildpunkten. Die partiellen Ableitungen zweiter Ordnung berechneten sich durch die Mehrfachanwendung der Ableitungsoperatoren erster Ordnung. Zur Integration der Tensoreinträge verwendeten wir eine 15×15 Bildpunkte umfassende Umgebung im Ort.

Tabelle 5.1 gibt die zu Abbildung 5.2 korrespondierenden Fehler der Bewegungsbestimmung sowie die Mittelwerte und Standardabweichungen der Parameter der Beleuchtungsfunktionen an. Die ermittelten Winkel- und Längenfehler werden für beide Vektorfelder

Modell	Filter	Winkelfehler RMSE/STD	Relativer Längenfehler RMSE/STD	Parameter Mittelwert/STD
Kein	Binom	\mathbf{u} : 1.335/0.767 \mathbf{v} : 5.55/1.521	\mathbf{u} : 2.7/2.0 \mathbf{v} : 17.6/4.7	–
	Gauß	\mathbf{u} : 0.626/0.391 \mathbf{v} : 0.963/0.496	\mathbf{u} : 1.1/1.0 \mathbf{v} : 2.4/1.0	–
	Scharr	\mathbf{u} : 0.287/0.177 \mathbf{v} : 0.255/0.156	\mathbf{u} : 0.5/0.5 \mathbf{v} : 0.6/0.6	–
Additiv	Binom	\mathbf{u} : 2.473/1.517 \mathbf{v} : 5.3/2.198	\mathbf{u} : 6.0/4.0 \mathbf{v} : 9.6/5.2	k'' : 8.692/0.279
	Gauß	\mathbf{u} : 0.663/ 0.411 \mathbf{v} : 1.075/0.555	\mathbf{u} : 1.1/1.1 \mathbf{v} : 2.9/1.4	k'' : 7.998/0.015
	Scharr	\mathbf{u} : 1.032/0.605 \mathbf{v} : 0.8/0.518	\mathbf{u} : 1.8/1.4 \mathbf{v} : 1.7/1.3	k'' : 8.016/0.077
Multiplikativ	Binom	\mathbf{u} : 8.779/5.128 \mathbf{v} : 25.49/7.421	\mathbf{u} : 15.5/ 15.5 \mathbf{v} : 73.7/ 29.9	a_1 : -0.452/0.133 a_2 : -0.226/0.083
	Gauß	\mathbf{u} : 6.147/ 3.852 \mathbf{v} : 7.521/5.098	\mathbf{u} : 42.13/17.59 \mathbf{v} : 43.82/32.85	a_1 : -0.301/0.054 a_2 : -0.099/0.066
	Scharr	\mathbf{u} : 3.039/1.837 \mathbf{v} : 3.218/1.826	\mathbf{u} : 5.4/5.0 \mathbf{v} : 4.8/4.7	a_1 : -0.289/0.038 a_2 : -0.202/0.038

Tabelle 5.1: Ermittelte Fehlerwerte entsprechend den Ergebnissen aus Abbildung 5.2.

getrennt angegeben. Die Einheit der Winkelfehler ist Grad und die der relativen Längenfehler Prozent. Die Zahlen bestätigen den schon optisch gewonnenen Eindruck, dass die Wahl der Filterpaare im rein transparenten und im additiven Fall relativ unkritisch ist, während sie im multiplikativen Fall die Ergebnisse deutlich beeinflussen. Die Filter von Scharr erzielen jedoch gute Ergebnisse, so dass sie in diesem Fall für die Bestimmung der Bewegungen und Beleuchtungsparameter genutzt werden sollten.

In den bisher betrachteten Beispielen stimmte das verwendete Bewegungsbestimmungsverfahren immer mit dem Helligkeitsänderungsmodell in der Sequenz überein. Wie in Abbildung 5.3 zu sehen ist, erhalten wir, wenn wir die Bewegungsbestimmung mit einem Verfahren durchführen, das nicht der vorliegenden Sequenz entspricht, extrem schlechte Ergebnisse. Ausgenommen sind rein transparente Sequenzen. Bei ihnen sind ebenfalls sehr gute Ergebnisse mit dem additiven und dem multiplikativen Modell erzielbar. Diese Beobachtung ist jedoch wenig verwunderlich, da der rein transparente Fall in beiden Modellen den Sonderfall darstellt, bei dem die Parameter der Beleuchtungsfunktionen null sind. Die Wahl des richtigen Verfahrens ist daher entscheidend. Da wir alle hier vorgestellten Erweiterungen in der Form $\mathbf{d}f \cdot \mathbf{c} = 0$ darstellen können, ist der Parametervektor mit einem angepassten Strukturtenorverfahren bestimmbar. Es gelten dann auch die Konfidenzkriterien aus Abschnitt 3.1.7 und erlauben das richtige Verfahren auszuwählen. Die notwendigen Erweiterungen sind leicht in das hierarchische Verfahren 3.1 integrierbar.

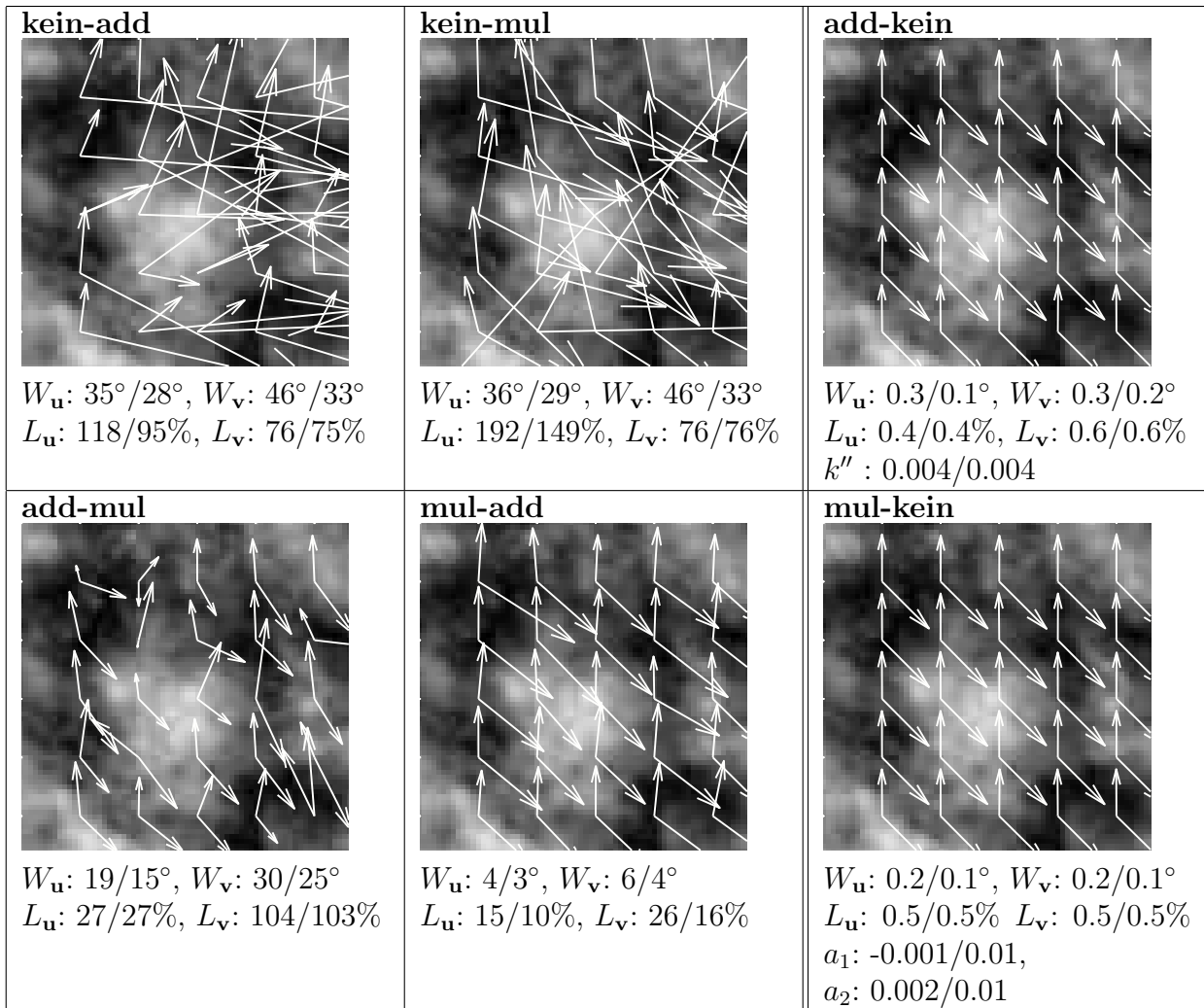


Abbildung 5.3: Ergebnisse für den Fall, dass das verwendete Verfahren nicht mit der Beleuchtungsänderung in der Sequenz konform ist. Die jeweilige Paarung von Modell und Sequenz wird durch die Überschrift **Berechnungsverfahren-Sequenztyp** angegeben.

5.5 Diskussion

Zusätzliche Beleuchtungsänderungen, die nicht auf die Bewegungen in einer Bildsequenz zurückführbar sind, können die Ergebnisse der Bewegungsbestimmung deutlich beeinflussen. Für eine Bewegung sind in der Vergangenheit eine Reihe von Verfahren vorgestellt worden, die die Bewegungsbestimmung robuster gegenüber äußeren Beleuchtungsänderungen machen sollen. Zum Teil sind diese Verfahren physikalisch motiviert und neben den Bewegungen interessieren auch die Parameter des physikalischen Vorgangs, der die Helligkeitsänderungen bewirkt.

In diesem Abschnitt haben wir den transparenten Bewegungsansatz für zwei Überlagerungen um die Modellierung additiver und multiplikativer Helligkeitsänderungen erweitert. In dem hier vorgestellten Verfahren gehen wir davon aus, dass die zusätzlichen Helligkeitsänderungen durch Funktionen beschreibbar sind, die nur von der Zeit abhängen. Fer-

ner implizieren wir damit, dass die Helligkeitsänderungen im Ort nur langsam variieren und damit lokal annähernd konstant sind. Ausgehend aus dieser Beleuchtungsmodellfunktion leiteten wir separate erweiterte Flussgleichungen für zwei transparente Bewegungen mit additiver bzw. multiplikativer Helligkeitsänderung her.

Die Analyse zeigte, dass das rein transparente Bewegungsmodell blind gegenüber additiven linearen Helligkeitsänderungen ist, und somit eine explizite Betrachtung additiver Helligkeitsfunktionen erst ab quadratischen Funktionen erforderlich macht. Nicht selten werden bei der Bestimmung der Bewegungsvektoren mit lokalen Methoden kleine räumliche und zeitliche Umgebungen betrachtet und innerhalb dieser Umgebung alle Parameter als konstant angenommen. Dies impliziert bei dem multiplikativen Ansatz Beleuchtungsänderungen durch exponentielle Funktionen und bei dem additiven Modell durch quadratische Funktionen. Insbesondere die Annahme einer exponentiellen Beleuchtungsfunktion ermöglicht eine einfache Handhabung von Beleuchtungsfunktionen, die sich multiplikativ jeweils getrennt auf eine der beiden Schichten auswirken.

Da die erweiterten Flussgleichungen als inneres Produkt zwischen einem Datenvektor und einem Parametervektor darstellbar sind, können die Bewegungsvektoren sowie die Parameter der Beleuchtungsfunktionen durch ein angepasstes Strukturtenorverfahren bestimmt werden. Anhand des Konfidenzmaßes können wir entscheiden, ob ein Verfahren für die Bestimmung der Bewegungsvektoren sowie der Beleuchtungsparameter geeignet ist. Können Sequenz und Verfahren nicht in Einklang gebracht werden, so gibt es abgesehen von der trivialen Lösung keinen Parametervektor, der das innere Produkt zwischen dem Daten- und Parametervektor zu null setzt. Damit ist der kleinste Eigenwert des entsprechenden Tensors deutlich von null verschieden und die Konfidenz nicht erfüllt. Die Erweiterungen können leicht in den hierarchischen Ablauf von Algorithmus 3.1 integriert werden.

Die Experimente mit synthetischen Sequenzen bestätigen gute Ergebnisse, wenn das verwendete Verfahren der gegebenen Bildsequenz entspricht. Der Versuch, die Bewegungsvektoren mit einem der Sequenz nicht entsprechenden Verfahren zu bestimmen, wird im Allgemeinen scheitern und zu schlechten Ergebnissen führen. Bei dem multiplikativen Modell hat die Wahl der Ableitungs- und Glättungsfilter einen entscheidenden Einfluss auf die Ergebnisse. Gute Ergebnisse ließen sich dabei für synthetische Sequenzen mit geringem Rauschen durch die Filter von Scharr [134, 135] erzielen. Alle anderen Filter, mit denen wir zuvor gute Ergebnisse für reale und synthetische Sequenzen bekamen, führten zu schlechten Resultaten. Um das multiplikative Modell in praktischen Anwendungen einsetzen zu können, ist noch viel Optimierungsarbeit zu leisten. Hingegen beeinflusst die Wahl der Ableitungsfilter im additiven Modell die Ergebnisse nur geringfügig. Ein Test beider Modelle mit realen Daten bleibt noch zu führen.

Kapitel 6

Phasenbasierte Bewegungsbestimmung

Transparent überlagerte Bewegungen lassen sich auch im Frequenzraum bestimmen. In dem einleitenden Kapitel hatten wir einen Ansatz dazu kennen gelernt: Zunächst wird die gesamte Bildsequenz fouriertransformiert und dann nach den Ebenen gesucht, die die Fouriertransformierten der verschobenen Bilder enthält. Eine weitere Möglichkeit, Bewegungen im Frequenzraum zu bestimmen, basiert auf der Beobachtung, dass es durch die Bewegungen zu einer Phasenverschiebung im Frequenzraum kommt. Im Falle einer einfachen Bewegung ermitteln wir durch Teilen der Fouriertransformierten zweier aufeinander folgender Bilder die durch die Bewegung induzierte Phasenverschiebung. Diese Phasenverschiebung kann nun zur Bestimmung des Geschwindigkeitsvektors genutzt werden. Vernon [174] entwickelte eine vergleichbare Lösung zur Bestimmung der jeweiligen Phasenverschiebungen von zwei transparent überlagerten Bewegungen. Für die Bestimmung beider Phasenverschiebungen sind drei aufeinander folgende Bilder nötig.

Alternativ zu der Phasenbestimmung im Frequenzraum bestimmen Jepson & Fleet [61] die Bewegungen anhand lokaler Phasen. Die lokale Phaseninformation ermitteln sie durch Filterung der Sequenz mit einer Menge von räumlich-zeitlichen Quadraturfiltern. Ein besonderer Vorteil der Verwendung der lokalen Phaseninformation eines bandpassgefilterten Signals gegenüber der Amplitudeninformation besteht in ihrer weitgehenden Unabhängigkeit gegenüber Beleuchtungsänderungen und kleineren Störungen [61, 63, 62]. Aus den räumlich-zeitlichen Gradienten der lokalen Phase jedes Filterergebnisses bestimmen sie die normale Geschwindigkeitkomponente senkrecht zum räumlichen Phasengradienten. Aus den normalen Geschwindigkeitskomponenten aller Filterantworten berechnen sie anschließend die gesuchten Geschwindigkeitsvektoren. Wie eine vergleichende Studie verschiedener Bewegungsbestimmungsverfahren von Barron et. al. [16] zeigt, wird mit der Phasenverschiebungsmethode von Jepson & Fleet [61] in vielen Fällen das beste Ergebnis erzielt. Im Gegenzug dazu ist sie mit ihren 22 Filterungen vergleichsweise rechenaufwendig. Gautama & Van Hulle [70] stellen später ein ähnliches Verfahren vor, dass im Unterschied zu dem von Jepson & Fleet [61] nur mit räumlichen Quadraturfiltern auskommt und keine räumlich-zeitlichen Filter benötigt.

Wie wir gesehen haben, teilen sich die phasenbasierten Methoden in zwei wesentliche

Ansatzlinien auf: die eine ermittelt die Geschwindigkeiten anhand der Phasenverschiebungen im Frequenzraum und die andere anhand lokaler Phasen, um eine Entkopplung von der Amplitude des Signals zu erzielen. In diesem Abschnitt werden wir die von Vernon [174] begonnene Arbeit auf eine beliebige Anzahl transparent überlagerter Bewegungen verallgemeinern [163]. Interessanterweise tritt dabei ein ähnliches Problem wie bei der Bewegungsbestimmung im Ortsraum auf. Zunächst lösen wir das Problem einer auftretenden Nichtlinearität durch Einführen der sogenannten *gemischten Phasenterme*. Nachdem die gemischten Phasenterme bestimmt sind, erhalten wir die gesuchten Phasenverschiebungen durch die Berechnung der Nullstellen eines komplexen Polynoms, dessen Koeffizienten sich aus den gemischten Phasentermen zusammensetzen. Ferner lässt sich mit dieser Methode schnell ein Eindruck über die Größenordnung der Bewegungen in der Bildsequenz gewinnen.

6.1 Phasen-Modell

Wie bei den differenziellen Methoden nehmen wir an, dass die Bildsequenz aus einer Superposition von N bewegten Schichten $g_i(\mathbf{x})$, $i = 1, \dots, N$ mit jeweils konstanten Bewegungen \mathbf{v}_i entstanden ist, d.h.

$$f_{t_k}(\mathbf{x}) = f(\mathbf{x}, t_k) = g_1(\mathbf{x} - t_k \mathbf{v}_1) + g_2(\mathbf{x} - t_k \mathbf{v}_2) + \dots + g_N(\mathbf{x} - t_k \mathbf{v}_N). \quad (6.1)$$

Im Unterschied zu vorher liegt die Bildsequenz jedoch durch $t_k = k\Delta t$, $k = 0, \dots$ zeitdiskret vor. Der Ausdruck Δt beschreibt den zeitlichen Abstand zweier aufeinander folgender Bilder der Bildsequenz. Um die Schreibweise zu vereinfachen, werden wir im Folgenden den Zeitpunkt in Form eines Index an der Bildsequenz angeben und schreiben $f(\mathbf{x}, t_k)$ als $f_{t_k}(\mathbf{x})$. Nach Anwendung der zweidimensionalen Fouriertransformation auf jedes Bild der Bildsequenz erhalten wir das Gleichungssystem

$$F_{t_k}(\boldsymbol{\omega}) = \phi_1^k G_1(\boldsymbol{\omega}) + \phi_2^k G_2(\boldsymbol{\omega}) + \dots + \phi_N^k G_N(\boldsymbol{\omega}), \quad k = 0, \dots, \quad (6.2)$$

mit den Phasenausdrücken $\phi_n = e^{-j2\pi\boldsymbol{\omega} \cdot \mathbf{v}_n \Delta t}$, $n = 1, \dots, N$, $j^2 = -1$, und dem Frequenzvektor $\boldsymbol{\omega} = (\omega_x, \omega_y)$. Die großen Buchstaben beschreiben die Fouriertransformierten der entsprechenden kleinbuchstabigen Signale im Ortsraum. So ist zum Beispiel F_{t_k} die Fouriertransformierte von f_{t_k} . Vernon [174] löste Gleichung (6.2) für den einfachen Fall von zwei Bewegungen analytisch, um die Phasenausdrücke ϕ_n zu bestimmen und damit anschließend die bewegten Schichten voneinander zu trennen. In dem folgenden Abschnitt werden wir Gleichung (6.2) für den allgemeinen Fall von N -Bewegungen lösen.

6.2 Bestimmung der Phasenverschiebungen

Bevor wir mit der Phasenbestimmung beginnen, vereinfachen wir Gleichung (6.2) für eine gegebene Frequenz $\boldsymbol{\omega}$ durch Einführen der Notationen $\Phi_k = (\phi_1^k, \dots, \phi_N^k)$ und $\mathbf{G} = (G_1, \dots, G_N)$ zu

$$F_{t_k} = \Phi_k \cdot \mathbf{G}, \quad k = 0, \dots \quad (6.3)$$

Unser Ziel ist jetzt, den gesuchten Phasenvektor $\Phi_1 = (\phi_1, \dots, \phi_N)$ dadurch zu bestimmen, dass wir den unbekanntem Vektor \mathbf{G} , der die Spektralanteile unserer Schichten für

eine gegebene Frequenz ω enthält, aus dem obigen Gleichungssystem auslöschen. Dazu definieren wir das komplexe Polynom

$$p(z) = (z - \phi_1) \cdots (z - \phi_N) = z^N + a_1 z^{N-1} + \cdots + a_N, \quad (6.4)$$

das die Nullstellen ϕ_1, \dots, ϕ_N besitzt und dessen Koeffizient a_1, \dots, a_N durch

$$\begin{aligned} a_0 &= 1 \\ a_1 &= - \sum_{i=1}^N \phi_i \\ a_2 &= \sum_{i<l} \phi_i \phi_l \\ a_3 &= - \sum_{i<l<k} \phi_i \phi_l \phi_k \\ &\vdots \\ a_N &= (-1)^N \phi_1 \phi_2 \cdots \phi_N \end{aligned} \quad (6.5)$$

gegeben sind. Da die Phasenausdrücke ϕ_1, \dots, ϕ_N die Nullstellen von $p(z)$ sind, d.h. $p(\phi_n) = 0$, für $n = 1, \dots, N$, lässt sich der Nullvektor durch die Linearkombination

$$\begin{aligned} \Phi_{m+N} + a_1 \Phi_{m+N-1} + \cdots + a_N \Phi_m &= (\dots, \phi_n^{m+N} + a_1 \phi_n^{m+N-1} + \cdots + a_N \phi_n^m, \dots) = \\ (\dots, \phi_n^m (\phi_n^N + a_1 \phi_n^{N-1} + \cdots + a_N), \dots) &= (\phi_1^m p(\phi_1), \dots, \phi_n^m p(\phi_n), \dots, \phi_N^m p(\phi_N)) = \mathbf{0}. \end{aligned} \quad (6.6)$$

erzeugen.

Setzen wir nun die Fouriertransformierten für $N + 1$ aufeinander folgende Bilder aus Gleichung (6.3) in Gleichung (6.6) ein, so erhalten wir

$$\begin{aligned} F_{t_{m+N}} + a_1 F_{t_{m+N-1}} + \cdots + a_N F_{t_m} &= \\ (\Phi_{m+N} + a_1 \Phi_{m+N-1} + \cdots + a_N \Phi_m) \cdot \mathbf{G} &= \mathbf{0} \cdot \mathbf{G} = 0. \end{aligned} \quad (6.7)$$

Diese Gleichungen lassen sich zu

$$F_{t_{m+N}} = -a_N F_{t_m} - \cdots - a_1 F_{t_{m+N-1}} \quad m = 0, \dots \quad (6.8)$$

umstellen und beschreiben somit, wie sich die Spektralanteile zum Zeitpunkt t_{m+N} aus den Spektralanteilen der N Vorgängerbilder berechnen lassen. Gleichung (6.8) geht für die ersten N Gleichungen in das lineare System

$$\begin{pmatrix} F_{t_N} \\ F_{t_{N+1}} \\ \vdots \\ F_{t_{2N-1}} \end{pmatrix} = - \begin{pmatrix} F_{t_0} & F_{t_1} & \cdots & F_{t_{N-1}} \\ F_{t_1} & F_{t_2} & \cdots & F_{t_N} \\ \vdots & \vdots & & \vdots \\ F_{t_{N-1}} & F_{t_N} & \cdots & F_{t_{2N-2}} \end{pmatrix} \begin{pmatrix} a_N \\ a_{N-1} \\ \vdots \\ a_1 \end{pmatrix}. \quad (6.9)$$

über. Seine Lösung ergibt den Koeffizientenvektor $\mathbf{y} = (a_1, \dots, a_N)$. Nachdem das Gleichungssystem für die *gemischten Phasenterme* a_1, \dots, a_N gelöst wurde, ermitteln wir die gesuchten Phasenausdrücke ϕ_1, \dots, ϕ_N durch eine Nullstellenbestimmung von $p(z)$. Mit der Bezeichnung \mathbf{F}_N für den Vektor auf der linken Seite und \mathbf{A}_N für die Matrix im obigen System, führen wir die Kurzschreibweise $\mathbf{F}_N = -\mathbf{A}_N \mathbf{y}$ ein.

6.3 Singularität der Matrix \mathbf{A}_N

Im Allgemeinen lassen sich die Phasenausdrücke $\Phi_1 = (\phi_1, \dots, \phi_N)$ nur dann aus Gleichung (6.9) bestimmen, wenn die Matrix \mathbf{A}_N nicht singulär ist. Hier ist aber die Berechnung einiger der Phasenverschiebungen selbst bei einer singulären Matrix \mathbf{A}_N möglich. Um dies zu verstehen, analysieren wir zunächst die Umstände, unter denen \mathbf{A}_N singulär wird. Die sogenannte Vandermonde-Matrix

$$\mathbf{B}_N = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \phi_1 & \phi_2 & \cdots & \phi_N \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1^{N-1} & \phi_2^{N-1} & \cdots & \phi_N^{N-1} \end{pmatrix}, \quad (6.10)$$

deren Einträge aus Potenzen der Phasenausdrücke ϕ_1, \dots, ϕ_N bestehen, ermöglicht eine Faktorisierung der Matrix \mathbf{A}_N zu

$$\mathbf{A}_N = \begin{pmatrix} F_{t_0} & F_{t_1} & \cdots & F_{t_{N-1}} \\ F_{t_1} & F_{t_2} & \cdots & F_{t_N} \\ \vdots & \vdots & \ddots & \vdots \\ F_{t_{N-1}} & F_{t_N} & \cdots & F_{t_{2N-2}} \end{pmatrix} = \mathbf{B}_N \begin{pmatrix} G_1 & 0 & \cdots & 0 \\ 0 & G_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & G_N \end{pmatrix} \mathbf{B}_N^T. \quad (6.11)$$

Die Determinante der Vandermonde-Matrix ist durch

$$\det \mathbf{B}_N^T = \prod_{m>n} (\phi_m - \phi_n) = \det \mathbf{B}_N, \quad (6.12)$$

gegeben und wir erhalten für die Determinante von \mathbf{A}_N

$$\det \mathbf{A}_N = G_1 \cdots G_N \prod_{m>n} (\phi_m - \phi_n)^2. \quad (6.13)$$

Aus Gleichung (6.12) wird deutlich, dass es zwei sich gegenseitig nicht ausschließende Fälle gibt, für die \mathbf{A}_N singulär wird:

1. Die Fouriertransformierte einer der Schichten verschwindet für die betrachtete Frequenz ω , d.h. $G_m(\omega) = 0$;
2. Zwei oder mehrere Phasenausdrücke sind identisch, d.h. $\phi_m = \phi_n$ für mindestens ein Paar m, n .

Sind zwei Phasenausdrücke identisch, z.B. $\phi_N = \phi_{N-1}$, reduziert sich die Anzahl der Summanden in Gleichung (6.2) zu

$$F_{t_k} = \phi_1^k G_1 + \phi_2^k G_2 + \cdots + \phi_{N-1}^k (G_{N-1} + G_N) = \phi_1^k G_1 + \phi_2^k G_2 + \cdots + \phi_{N-1}^k \tilde{G}_{N-1}, \quad k = 0, \dots \quad (6.14)$$

wobei $\tilde{G}_{N-1} = G_{N-1} + G_N$ ist. Sind mehr als zwei Phasenausdrücke identisch, so werden die Spektralanteile von zwei oder mehr Schichten in \tilde{G}_i zusammengefasst und wir können daher

$$F_{t_k} = \phi_1^k \tilde{G}_1 + \phi_2^k \tilde{G}_2 + \cdots + \phi_R^k \tilde{G}_R \quad k = 0, \dots \quad (6.15)$$

annehmen, falls die nun verbleibenden Phasenausdrücke ϕ_1, \dots, ϕ_R alle verschieden und alle $\tilde{G}_1, \dots, \tilde{G}_R$ ungleich null sind. Gänzlich ausschließen können wir jedoch nicht, dass ein weiterer Phasenterm in der Gleichung verschwindet, falls die Summe der fouriertransformierten Schichten für eines der \tilde{G}_i null ergibt. Da sich in der Praxis nur der Rang von \mathbf{A}_N beobachten lässt, ist es uns nicht möglich festzustellen, ob eine solche Situation aufgetreten ist oder nicht. Die linke Seite der Gleichung (6.15) bleibt sowohl bei einer Faktorisierung als auch bei einer Neuordnung der Ausdrücke unverändert und wir können Gleichung (6.11) als

$$\mathbf{A}_N = \begin{pmatrix} F_{t_0} & F_{t_1} & \cdots & F_{t_{N-1}} \\ F_{t_1} & F_{t_2} & \cdots & F_{t_N} \\ \vdots & \vdots & & \vdots \\ F_{t_{N-1}} & F_{t_N} & \cdots & F_{t_{2N-2}} \end{pmatrix} = \mathbf{B}_N \begin{pmatrix} \tilde{G}_1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \tilde{G}_2 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & \tilde{G}_R & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 0 \end{pmatrix} \mathbf{B}_N^T \quad (6.16)$$

schreiben. Folglich ist $\text{rang}(\mathbf{A}_N) = \text{rang}(\mathbf{A}_R) = R$ und wir können die Phasenausdrücke ϕ_1, \dots, ϕ_R durch ein erneutes Lösen des auf R Bewegungen verkleinerten Systems

$$\mathbf{F}_R = -\mathbf{A}_R \mathbf{y} \quad (6.17)$$

bestimmen. Im Unterschied zu dem Fall von N -Bewegungen besteht der Vektor $\mathbf{y} = (a_1, \dots, a_R)$ jetzt aus R Elementen. Folglich müssen wir auch das Polynom aus Gleichung (6.4) neu anpassen, um die gesuchten Phasenausdrücke aus dem Vektor \mathbf{y} zu extrahieren: Es muss R Nullstellen besitzen und demnach vom Grad R sein.

6.3.1 Beispiel für zwei Bewegungen

In diesem Abschnitt möchten wir die oben hergeleiteten Ergebnisse anhand eines Beispiels mit zwei Bewegungen veranschaulichen. Bei der Bewegungsbestimmung können an jeder Frequenz ω im Spektralbereich folgende drei Fälle auftreten:

1. $\text{Rang}(\mathbf{A}_2) = 2$: Die Bestimmung der Phasenterme ϕ_1, ϕ_2 ist möglich. Im ersten Schritt bestimmen wir die gemischten Phasenterme a_1, a_2 durch Lösen des Systems $\mathbf{F}_{t_2} = -\mathbf{A}_2 \mathbf{y}$, bzw.

$$\begin{pmatrix} F_{t_2} \\ F_{t_3} \end{pmatrix} = - \begin{pmatrix} F_{t_0} & F_{t_1} \\ F_{t_1} & F_{t_2} \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \end{pmatrix} \quad (6.18)$$

und erhalten

$$a_1 = \frac{F_{t_1} F_{t_2} - F_{t_0} F_{t_3}}{F_{t_0} F_{t_2} - F_{t_1}^2}$$

$$a_2 = \frac{F_{t_1}^2 - F_{t_0} F_{t_2}}{F_{t_0} F_{t_2} - F_{t_1}^2}.$$

Im zweiten Schritt berechnen wir die gesuchten Phasenausdrücke ϕ_1, ϕ_2 durch die Bestimmung der Nullstellen von

$$p(z) = z^2 + a_1 z + a_2. \quad (6.19)$$

2. $\text{Rang}(\mathbf{A}_2) = 1$: In diesem Fall ist entweder $G_1 = 0 \wedge G_2 \neq 0$, $G_1 \neq 0 \wedge G_2 = 0$ oder $\phi_1 = \phi_2 \wedge G_1 + G_2 \neq 0$ und wir bestimmen eine der beiden unterschiedlichen Phasenterme bzw. im letzten Fall die ‘‘Doppelphase’’ durch

$$F_{t_1} = F_{t_0} \phi. \quad (6.20)$$

3. $\text{Rang}(\mathbf{A}_2) = 0$: In diesem Fall ist entweder $G_1 = G_2 = 0$ oder $\phi_1 = \phi_2, \wedge G_1 + G_2 = 0$ und alle Gleichungen in (6.2) sind null, d.h.

$$F_{t_k} = 0, \quad k = 0, \dots, \quad (6.21)$$

und keine der Phasenausdrücke ist bestimmbar.

6.4 Berechnung der Geschwindigkeitsvektoren

Bisher haben wir gesehen, wie die Phasenterme bestimmt werden können, sind aber in den meisten Fällen an den zugrunde liegenden Bewegungsvektoren interessiert und benötigen daher ein Verfahren, mit dem wir die Bewegungsvektoren aus den Phasentermen ermitteln können. Zwischen dem Frequenzvektor $\boldsymbol{\omega}$, dem Bewegungsvektor \mathbf{v} und dem Phasenausdruck $\phi(\boldsymbol{\omega}) = e^{-j2\pi(\omega_x v_x + \omega_y v_y)\Delta t}$ besteht der Zusammenhang

$$\boldsymbol{\omega} \cdot \mathbf{v} = \omega_x v_x + \omega_y v_y = -\frac{1}{2\pi} \arctan(\Im(\phi(\boldsymbol{\omega})), \Re(\phi(\boldsymbol{\omega}))). \quad (6.22)$$

Er beschreibt ein lineares Gleichungssystem mit einer Gleichung und zwei Unbekannten. Um den Bewegungsvektor \mathbf{v} bestimmen zu können, benötigen wir also mindestens eine weitere Gleichung, die wir durch die Auswertung obiger Gleichung für eine weitere Frequenz erhalten. Wichtig dabei ist, dass der zweite Frequenzvektor, den wie mit $\hat{\boldsymbol{\omega}}$ bezeichnen, kein Vielfaches von $\boldsymbol{\omega}$ ist und dass für ihn die Phase ϕ bestimmt werden konnte. Diese beiden Gleichungen erlauben, den gesuchten Bewegungsvektor \mathbf{v} durch Lösen des linearen Gleichungssystems

$$\begin{pmatrix} \omega_x & \omega_y \\ \hat{\omega}_x & \hat{\omega}_y \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix} = -\frac{1}{2\pi} \begin{pmatrix} \arctan(\Im(\Phi(\boldsymbol{\omega})), \Re(\Phi(\boldsymbol{\omega}))) \\ \arctan(\Im(\Phi(\hat{\boldsymbol{\omega}})), \Re(\Phi(\hat{\boldsymbol{\omega}}))) \end{pmatrix} \quad (6.23)$$

zu bestimmen. Folglich benötigen wir für die Bestimmung der Geschwindigkeitsvektoren nur zwei Punkte im Frequenzraum, an denen alle Phasenverschiebungen berechnet werden konnten und deren Verbindungsgerade nicht durch den Ursprung verläuft. Punkte, für die die Matrix \mathbf{A}_N nicht den vollen Rang besitzt, sind für die Geschwindigkeitsbestimmung uninteressant und brauchen nicht weiter berücksichtigt zu werden.

Obiges Gleichungssystem ist von den Frequenzen abhängig und das korrekte Aufstellen dieses Systems setzt eine vorangegangene Phasenabwicklung voraus. Dieser Vorverarbeitungsschritt erhöht zum Einen den notwendigen Rechenaufwand. Desweiteren wird er durch einen weiteren Umstand erschwert: Nach der Bestimmung der Phasenausdrücke aus den Nullstellen des komplexen Polynoms $p(z)$ ist die Zuordnung zwischen den Phasenausdrücken ϕ_1, \dots, ϕ_N und den Bewegungsvektoren $\mathbf{v}_1, \dots, \mathbf{v}_N$ nicht eindeutig. So kann zum Beispiel ϕ_1 für die Frequenz $\boldsymbol{\omega}$ dem Bewegungsvektor \mathbf{v}_1 entsprechen, für die Frequenz $\hat{\boldsymbol{\omega}}$

aber dem Vektor \mathbf{v}_2 . Dieses Zuordnungsproblem lösen wir mittels einer Vorschätzung der Geschwindigkeitsvektoren. Mit den vorgeschätzten Bewegungsvektoren sortieren wir die Phasenverschiebungen so, dass ϕ_i immer dem Bewegungsvektor \mathbf{v}_i entspricht.

Eine Vorschätzung der Bewegung kann aus dem gemischten Phasenterm $a_1 = \sum_{i=1}^N \phi_i$ errechnet werden. Die Rücktransformation dieses Ausdruckes in den Ort ergibt

$$a_1(\boldsymbol{\omega}) = \sum_{i=1}^N \phi_i(\boldsymbol{\omega}) \bullet - \circ \sum_{i=1}^N \delta(\mathbf{x} - \mathbf{v}_i). \quad (6.24)$$

Nach der Rücktransformation erhalten wir eine erste Schätzung der Bewegungsvektoren aus der Position der Dirac-Einheitsimpulse. Liegen die Bewegung allerdings sehr dicht beieinander, so können die zwei Dirac-Einheitsimpulse zu einem verschmelzen, und damit sind die Bewegungen nicht mehr eindeutig schätzbar. Dennoch ermöglicht uns diese Vorgehensweise, einen schnellen Eindruck über die Geschwindigkeiten in der Bildsequenz zu erhalten, eine Information die für weitere Verarbeitungsschritte und auch zur Parametereinstellung von Bewegungsbestimmungsverfahren hilfreich ist. Im Falle von zwei Bewegungen ist einer der Dirac-Stöße für die Sortierung ausreichend. Ferner ist zu beachten, dass die Summe aller Phasenausdrucke (a_1) nicht an allen Punkten berechnet werden kann und wir für diese Werte annehmen müssen. Dies kann Störungen im Ort verursachen. Wie wir bereits gesehen haben, sind einige dieser Bereiche durch Geraden beschrieben, für die wir die Auswirkungen explizit quantifizieren können. Setzen wir zum Beispiel bei zwei Bewegungen alle Anteile von a_1 entlang der Geraden $\delta(\mathbf{n} \cdot \boldsymbol{\omega})$ mit dem Normalenvektor $\mathbf{n} \propto \mathbf{v}_1 - \mathbf{v}_2$ gleich null, so ist a_1 durch den Ausdruck

$$a_1(\boldsymbol{\omega}) = (\phi_1(\boldsymbol{\omega}) + \phi_2(\boldsymbol{\omega}))(1 - \delta(\mathbf{n} \cdot \boldsymbol{\omega})) \quad (6.25)$$

beschrieben. Nach inverser Fouriertransformation entspricht er im Ort

$$\begin{aligned} & \delta(\mathbf{x} - \mathbf{v}_1) + \delta(\mathbf{x} - \mathbf{v}_2) - (\delta(\mathbf{x} - \mathbf{v}_1) + \delta(\mathbf{x} - \mathbf{v}_2)) * \delta(\mathbf{n}^\perp \cdot \mathbf{x}) \\ & = \delta(\mathbf{x} - \mathbf{v}_1) + \delta(\mathbf{x} - \mathbf{v}_2) - \delta(\mathbf{n}^\perp \cdot (\mathbf{x} - \mathbf{v}_1)) - \delta(\mathbf{n}^\perp \cdot (\mathbf{x} - \mathbf{v}_2)). \end{aligned} \quad (6.26)$$

Die Dirac-Geraden $\delta(\mathbf{n}^\perp \cdot (\mathbf{x} - \mathbf{v}_1))$ und $\delta(\mathbf{n}^\perp \cdot (\mathbf{x} - \mathbf{v}_2))$ verlaufen beide durch die Punkte \mathbf{v}_1 und \mathbf{v}_2 , und erscheinen damit wie eine Gerade auf der sich die gesuchten Dirac-Stöße befinden. Die diskrete Fouriertransformation gewichtet die beiden Dirac-Einheitsimpulsgeraden in der Praxis meist mit einer sehr kleinen Zahl. Die inverse diskrete Fouriertransformation eines $N \times M$ großen Spektrums C_{nm} ist durch die Gleichung

$$c(k, l) = \frac{1}{MN} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} C_{nm} \exp(j2\pi(\frac{nk}{N} + \frac{ml}{M})) \quad (6.27)$$

beschrieben. Besitzen die Spektralkoeffizienten C_{nm} entlang der Geraden den Wert eins, so können wir mit $k = l = 0$ direkt die Werte entlang der entsprechenden Geraden im Ort ablesen:

$$\frac{1}{MN} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} C_{nm}. \quad (6.28)$$

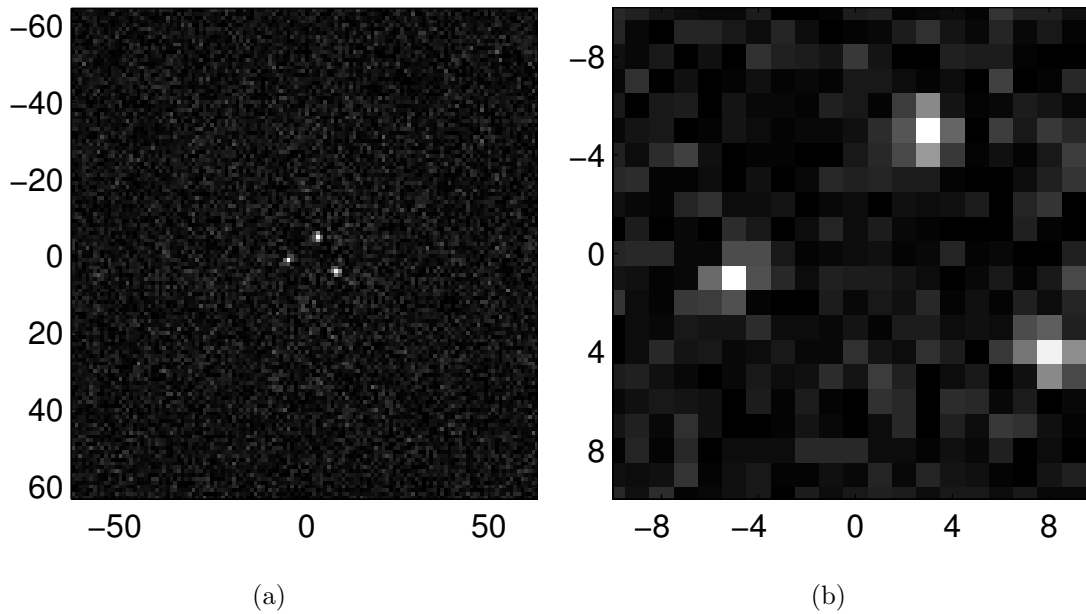


Abbildung 6.1: Bild (a) zeigt die aus $a_1 = \phi_1 + \phi_2 + \phi_3$ ermittelten Dirac-Impulse für eine Sequenz mit den Geschwindigkeiten $\mathbf{u} = (-5, 1)^T$, $\mathbf{v} = (8, 4)^T$ und $\mathbf{w} = (3, -5)^T$ und einem SNR von 40dB. In (b) ist ein kleiner Ausschnitt um die Diracs herum vergrößert dargestellt.

Die Doppelsumme ist als Länge der Geraden aufgefasst, die deutlich kleiner als der Ausdruck $\frac{1}{MN}$ ausfällt. So ergibt sich beispielsweise für $N = M = 32$ und einer Geradenlänge von 32 Punkten ein Skalierungsfaktor von $\frac{1}{M} \approx 0.03$ und ist damit im Verhältnis zu den Einheitsimpulsen klein.

Abbildung 6.1 zeigt ein Beispiel für eine Sequenz in der drei Bewegungen und ein Signalrauschverhältnis von 40dB beobachtet werden. In (a) sind die Dirac-Impulse deutlich zu erkennen. Punkte bei denen nicht alle Phasen bestimmbar waren, wurden zu null gesetzt. Die daraus resultierenden Geraden im Ort haben ein so geringes Gewicht, dass wir sie nicht wahrnehmen können. Aus der Vergrößerung (b) können wir die Geschwindigkeiten ablesen. Zum Erstellen dieser Sequenzen verwendeten wir wieder unsere wolkenförmigen Testmuster aus den vorangegangenen Abschnitten.

Die ermittelten Dirac-Impulse für die Monalisasequenz sind in Abbildung 6.2(a) dargestellt. Da die Bewegungen sehr dicht bei einander liegen, fällt es selbst in der Vergrößerung (b) schwer die beiden Einheitsimpulse zu identifizieren. In der Monalisasequenz treten jedoch maximal nur zwei Bewegungen auf, so dass für die Sortierung der Phasen ein Bewegungsvektor ausreicht. In diesem Fall wählen wir also den Maximalwert an der Position $\mathbf{v} = (1, 0)$ zum Sortieren der Phasen.

Im Anschluss nutzen wir die gefundenen Einheitsimpulse, um eine Zuordnung der Phasen zu den jeweiligen Bewegungsvektoren durchzuführen. Die Übereinstimmung eines Phasen-

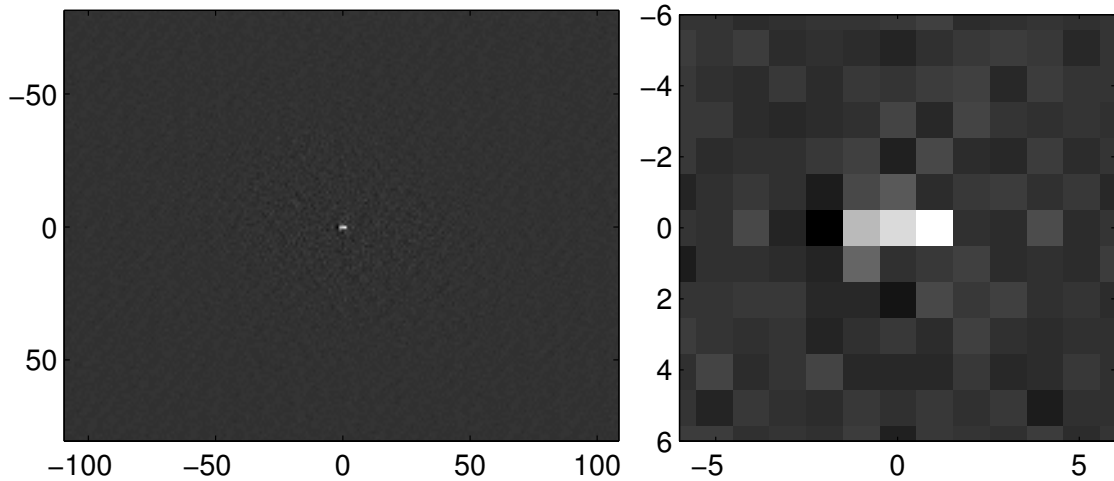


Abbildung 6.2: Bild (a) zeigt die aus $a_1 = \phi_1 + \phi_2$ ermittelten Einheitsimpulse für die Monalisasequenz. In (b) ist ein kleiner Ausschnitt um die Impulse herum vergrößert dargestellt.

wertes ϕ_i an der Frequenz $\boldsymbol{\omega}$ mit dem Bewegungsvektor \mathbf{v} messen wir mit dem Distanzmaß

$$D(\phi_i, \mathbf{v}, \boldsymbol{\omega}) = \text{abs}(\phi_i(\boldsymbol{\omega}) - \exp(-j2\pi\boldsymbol{\omega} \cdot \mathbf{v})), \quad (6.29)$$

das ohne eine Phasenabwicklung auskommt.

Die Vorschätzung erlaubt uns die Angabe der Geschwindigkeiten mit Pixelgenauigkeit. Zur genaueren Bestimmung der Geschwindigkeitsvektoren folgen wir nun einem ähnlichen Ansatz, wie Vernon [174] ihn vorschlägt. Das Verfahren basiert auf der Grundidee der Hough-Transformation. Zunächst legen wir einen Suchraum (hier einen quadratischen Bereich), in dem wir die Bewegung einer bestimmten Phase vermuten, unter der Vorgabe einer bestimmten Genauigkeit fest. Anschließend initialisieren wir eine Matrix, deren Zeilen mit der x -Komponente und deren Spalten mit der y -Komponente der Bewegungsvektoren im Suchraum assoziiert werden, mit null. Für jede Frequenz und jeden Vektor des Suchraumes werten wir für die gerade betrachtete Phase obiges Distanzmaß aus. Ist obige Distanz kleiner als ein vorgegebener Schwellwert (in allen hier angegebenen Beispielen 0.05), so erhöhen wir den zu dem aktuellen Bewegungsvektor korrespondierenden Matrixeintrag um eins. Am Ende der Rechnung entspricht der Vektor, dessen Matrixeintrag am größten ist, dem gesuchten Bewegungsvektor. Diesen Vorgang wiederholen wir nun für alle verbleibenden Bewegungsvektoren.

Lagen die Vektoren zu dicht beieinander, um eine sinnvolle Schätzung mit der Dirac-Methode zu erhalten, bietet sich ein leicht modifiziertes Vorgehen an. Anstatt die Vektoren nun separat zu ermitteln, nutzen wir nun eine Matrix zur Bestimmung aller Bewegungsvektoren. Für jede Frequenz und jeden Vektor werten wir die Distanzen zu allen Phasen aus. Für jede Distanz, die kleiner als ein vorgegebener Schwellwert ist, erhöhen wir den Matrixeintrag um eins. Am Ende der Rechnung entsprechen die N größten Einträge, gegebenenfalls nach einer Nicht-Maximum-Unterdrückung, den gesuchten Bewegungsvektoren.

Schrittweite	\mathbf{u}	\mathbf{v}
0.1	(0.9,0)	(-0.6,0)
0.05	(0.9,0)	(-0.55,0)
0.02	(0.94,0.02)	(-0.56,0)
0.01	(0.9,-0.01)	(-0.61,0.01)

Tabelle 6.1: Ermittelte Ergebnisse der Monalisasequenz bei unterschiedlichen Schrittweiten.

Natürlich hängt die Qualität der Ergebnisse von der gewählten Suchschrittweite zwischen den Bewegungsvektorkomponenten sowie deren Startpunkten, ab denen wir mit der Zählung beginnen, ab. Tabelle 6.1 zeigt einige Resultate der Monalisasequenz für Schrittweiten zwischen 0.1 und 0.01. In diesem Bereich variieren die Ergebnisse leicht. Das Suchintervall für jede Bewegungsvektorkomponente lag im Bereich von minus zwei bis plus zwei. Wie die Tabelle zeigt, variieren die Ergebnisse leicht mit der Schrittweite.

Das hier vorgestellte Verfahren eignet sich besonders gut zur Ermittlung globaler Bewegungen. Die Bestimmung lokaler Bewegungen ist, wie Abbildung 6.3 zeigt, prinzipiell möglich. Die Parametrisierung dieses Verfahrens gestaltet sich, insbesondere wenn zwischen einer und mehreren Bewegungen unterschieden werden soll, als schwierig. Zudem ist der Rechenaufwand recht hoch. Den meisten Rechenaufwand benötigt das Suchverfahren. Setzt sich zum Beispiel das Spektrum einer von zwei Schichten nur aus sehr wenigen Frequenzen zusammen, so gibt es im Frequenzraum nur sehr wenige Punkte, an denen die beiden Phasenanteile bestimmbar sind. Da der überwiegende Anteil nur zur Bestimmung der Bewegung der zweiten Schicht verwendbar ist, würde man hier von einer Bewegung ausgehen. In dem Beispiel aus Abbildung 6.3 ist die *lokale* Bewegungsbestimmung mit einer Fenstergröße von 64 Punkten durchgeführt worden. Leakage-Artefakte reduzierten wir in diesem und auch den anderen Beispielen durch Verwendung des Hanning-Fensters. In Bereichen, in denen die Matrix \mathbf{A}_2 nicht den vollen Rang besaß (falls das Verhältnis zwischen dem kleinsten und dem größten Singulärwert kleiner war als 0.4), haben wir die Phasenanteile der einfachen Bewegung bestimmt und das Ergebnis einer der beiden Phasen zugeordnet, während die andere dort dem Wert null zugewiesen bekam. In Bereichen mit nur einer Bewegung besteht nach der Sortierung eine der Phasenmatrizen im wesentlichen aus den Wert null. In dem entsprechenden Akkumulator dürfte sich demnach auch kein eindeutiges Maximum finden. Als Maximum anerkannt wurden in diesem Beispiel nur Akkumulatorzellen, deren Wert mindestens 200 betrug. Liegt der Wert darunter gehen wir davon aus, dass diese Phase zu keiner Bewegung korrespondiert und setzten den Bewegungsvektor für diese Schicht zu null.

6.5 Zusammenfassung

In diesem Abschnitt haben wir die von Vernon [174] begonnene Arbeit zur Bestimmung von zwei Bewegungen auf eine beliebige Anzahl überlagerter Bewegungen verallgemeinert und Beispiele für zwei und drei Bewegungen gezeigt. Für die Bestimmung von N Bewegungen benötigen wir die Fouriertransformierten von $2N$ aufeinander folgenden Bildern.



Abbildung 6.3: Lokale Bewegungsbestimmung mit 64×64 Bildpunkte großen Fenstern. Angegeben sind die Bewegungsvektoren für jeden achten Bildpunkt.

Aus diesen bestimmen wir ähnlich wie bei den differenziellen Methoden im ersten Schritt den gemischten Phasentermvektor. Die Einträge dieses Vektors setzen sich aus Produkten, Summen und Summen von Produkten der gesuchten Phasenterme zusammen. Anschließend entsprechen die gesuchten Phasenterme, aus denen wir dann die Bewegungsvektoren ermitteln möchten, wieder den Nullstellen eines komplexen Polynoms.

Ferner haben wir gesehen, dass einem der gemischten Phasenterme eine besondere Rolle zukommt. Transformierten wir ihn zurück in den Ort, so erhalten wir aus den Positionen der dort beobachtbaren Dirac-Impulse eine erste Schätzung der Bewegungsvektoren. Diese Vektoren können anschließend zur Sortierung der Phasenterme verwendet werden, aus denen sich die Bewegungen genauer bestimmen lassen.

Da wir für die Berechnung der Bewegungen aus ihren Fouriertransformierten ausreichend große Fenster (32×32 oder mehr) benötigen, ist dieses Verfahren tendenziell eher zur Bestimmung globaler als lokaler Bewegungen geeignet. Die hier gewonnenen Erkenntnisse werden wir im übernächsten Kapitel in den Ort übertragen und daraus ein Block-Matchingverfahren für transparent überlagerte Bewegungen ableiten, das sich besonders gut zur Ermittlung lokaler Bewegungen eignet.

Weiter konnten wir zeigen, dass die Phasenverschiebungen nicht an allen Frequenzen im Frequenzraum bestimmbar sind. Einige dieser Bereiche sind durch Geraden durch den Ursprung beschrieben. Diese Geraden wirken sich auf die Bewegungsbestimmung nur wenig aus, spielen aber im nächsten Kapitel, in dem wir die bewegten Schichten voneinander trennen möchten, eine besondere Rolle.

Kapitel 7

Trennung der bewegten Schichten

In den vorangegangenen Abschnitten haben wir uns mit dem Problem der Bewegungsbestimmung im Falle von überlagerten Bewegungen auseinander gesetzt. In einigen Anwendungen steht jedoch nicht die Bewegungsinformation im Vordergrund, sondern das Interesse gilt den Schichten. Daher möchten wir hier der Frage nachgehen, wie wir die gefundene Bewegungsinformation nutzen können, um die bewegten Schichten voneinander zu trennen. In der Literatur finden sich zu der Trennung additiv überlagelter Schichten eine Reihe von Veröffentlichungen, die sich grob in die zwei Ansatzlinien mit und ohne Berücksichtigung der Bewegungsinformation aufteilen lassen. Viele Ansätze zur Trennung transparent überlagelter Schichten in Bilder ohne zusätzliche Bewegungsinformation setzen zwei Aufnahmen desselben Bildes unter der Verwendung von verschiedenen Polarisationsfiltern voraus [133, 36, 54, 55, 126, 136, 140, 139]. Die Polarisationsfilter bewirken eine Veränderung der Gewichtungen der einzelnen Schichten in dem resultierenden Gesamtbild. Die Gewichtungparameter stellen multiplikative Faktoren für jede Schicht dar, die in beiden Aufnahmen unterschiedlich und in der Regeln unbekannt sind. Aus diesem Grund werden in [54, 55, 36] die unbekanntes Gewichtungparameter durch ein Bild-Source-Separation-Problem beschrieben und die gesuchten Schichten mit Hilfe der Independent-Component-Analyse (ICA) bestimmt. Schechner et. al. [138, 137, 141, 139] trennen die überlagerten Schichten aus zwei Aufnahmen, bei der jeweils eine der beiden Schichten scharfgestellt wird und die andere damit unscharf wird. Sie stellen eine Methode vor, die es erlaubt, aus den Rohdaten die Übertragungsfunktionen für die Defokussierungsoperation der Schichten zu ermitteln und damit die Trennung der Schichten durchzuführen. Diese Ansätze eignen sich normalerweise weniger zur Trennung der bewegten und überlagerten Schichten in Bildsequenzen, da in der Regel keine weiteren Fokussierungs- bzw. Polarisierungsaufnahmen vorliegen. Die Schichtentrennung muss daher unter Berücksichtigung der Bewegungsinformationen erfolgen.

Aus dem einleitenden Kapitel wissen wir, dass bei translativen Bewegungen die Fouriertransformierte Bildsequenz zwei Ebenen ergibt, die durch den Ursprung verlaufen und jeweils eine der Fouriertransformierten Schichten beinhalten. Diese Idee wurde in [49, 48] zur Generierung nullender Filter verwendet, um die Frequenzanteile einer Schicht zu entfernen und somit die andere zu isolieren. Da sich aber je zwei dieser Ebenen in einer Geraden schneiden, die ebenfalls durch den Ursprung verläuft, lassen sich die Frequenzanteile entlang dieser Geraden nicht voneinander trennen.

Szeliski et. al. [164] berechnen im ersten Schritt eine Approximation der bewegten Schichten durch den von ihnen entwickelten min/max-Alternierungs Algorithmus. Im nächsten Schritt nutzen sie die Approximationen der Schichten als Initialisierung eines durch die Bewegungen beschriebenen Kleinstquadrat Optimierungsproblems unter der zusätzlichen Nebenbedingung, dass die Grauwerte der jeweiligen Schichten immer einen Wert größer oder gleich null besitzen sollen. Die Nebenbedingung soll den Verlust der Mittelwerte der Schichten weiter einschränken und sicherstellen, dass der Grauwertbereich der Bilder nicht verlassen wird. Voraussetzung für eine verlustfreie Rekonstruktion ist jedoch, dass ausreichend viele Schichten einen Punkt mit dem Grauwert null besitzen.

Irani & Peleg [85, 86] trennen zwei transparent überlagerte Schichten durch zeitliche Integration. Am Anfang werden beide Schichten mit dem ersten Originalbild initialisiert. Anschließend wird jede dieser Schichten um ihre Bewegung verschoben und der Mittelwert zwischen der verschobenen Schicht und dem nächsten Bild in der Sequenz gebildet. Iterativ wiederholen sie diesen Vorgang über einige aufeinander folgende Bilder der Bildsequenz. Mit fortschreitender Iteration wird jeweils eine der Schichten immer deutlicher hervorgehoben während die andere durch die Mittelwertbildung verschmiert und abgeschwächt wird.

Sarel & Irani [133] stellen ein Verfahren vor, das sie als *layer information exchange* Algorithmus bezeichnen und das erlaubt, zwei transparent überlagerte unbewegte Schichten aus zwei Aufnahmen, die durch die Verwendung unterschiedlicher Polarisationsfilter entstanden sind, zu rekonstruieren. Der iterative Informationsaustausch basiert auf einem lokalen normalisierten Grauwertkorrelationsmaß. Zur Behandlung von bewegten Szenen kompensieren sie zunächst eine der beiden Bewegungen und wenden auf diese Sequenz das Verfahren von Weiss [184] an, um eine erste Approximation der bewegungskompensierten Schicht zu erhalten. Die zweite Schicht erhalten sie durch Abziehen der geschätzten Schicht von der bewegungskompensierten Bildsequenz, die gegebenenfalls noch geringe Anteile der ersten Schicht beinhalten und somit wie eine Aufnahme durch ein Polarisationsfilter wirken. Die so generierte Schicht und das Originalbild nutzen sie dann als Eingabebilder für den *layer information exchange* Algorithmus und erhalten eine weitere Verbesserung der Ergebnisse.

Das Verfahren von Weiss [184] ist ursprünglich zur Bestimmung von intrinsischen Bildern, d.h. die Aufteilung der Bilder in einen Reflexions- und Beleuchtungsanteil, in Bildsequenzen konzipiert, eignet sich aber auch zur Trennung von transparent überlagerten Schichten, wenn man die eine Schicht als den Reflexionsanteil und die andere als den Beleuchtungsanteil nach Logarithmierung betrachtet. Für sein Verfahren nimmt er an, dass der Reflexionsanteil über der Zeit gleich bleibt und sich nur der Beleuchtungsanteil mit der Zeit verändert. In unseren Kontext übertragen ist demnach eine der beiden Schichten unbewegt, während in der anderen Bewegungen vorkommen. Für jedes Bild berechnet er die Ableitungen in x - und y -Richtung und wendet anschließend einen zeitlichen Medianfilter auf die jeweiligen Ableitungen über alle Bilder an. Abschließend errechnet er das Reflektanzbild durch Invertierung der Ableitungsoperationen.

Toro et. al.[168] nutzen zwei aufeinander folgende Bilder der Bildsequenz zur Trennung beider transparent überlagelter Bewegungen. Zunächst kompensieren sie eine der beiden Bewegungen und ziehen das kompensierte Bild von dem Vorgängerbild ab. Damit verschwinden alle Anteile der bewegungskompensierten Schicht und übrig bleibt ein Differenzbild der zweiten Schicht $g_2(\mathbf{x})$ der Form

$$D(\mathbf{x}) = g_2(\mathbf{x}) - g_2(\mathbf{x} - \mathbf{v} + \mathbf{u}) \quad (7.1)$$

mit dem Bewegungsvektor \mathbf{u} für die erste und \mathbf{v} für die zweite Schicht. Zur Rekonstruktion der Schicht g_2 integrieren sie das Differenzbild $D(\mathbf{x})$ entlang paralleler Geraden in Richtung $\mathbf{u} - \mathbf{v}$. Da jedes dieser Integrale bis auf eine additive Konstante bestimmt ist, führen sie noch weitere Regularisierungsterme hinzu, die hinreichend glatte Lösungen bevorzugen.

Vernon [174] und Zhou & Kambhamettu [194] nutzen die durch die Bewegungen entstandenen Phasenverschiebungen der Fouriertransformierten einiger aufeinander folgender Bilder der Bildsequenz zur Trennung der Spektralanteile der Schichten. Beide Verfahren werden jedoch nur für den Fall von zwei Überlagerungen angegeben. Ähnlich wie bei der Bestimmung der Phasenverschiebungen aus dem vorangegangenen Abschnitt treten auch hier Bereiche auf, in denen die beiden Phasenverschiebungen identisch sind und daher keinen eindeutigen Rückschluss auf die zugrunde liegenden Spektralanteile der Schichten erlauben. Vernon [174] löste dieses Problem, indem er die fehlenden Spektralinformationen aus den umgebenen Spektralanteilen der Schichten interpolierte. Zhou & Kambhamettu [194] bestimmen ähnlich wie Vernon zuerst alle trennbaren Spektralanteile. Dieses Ergebnis nutzen sie als Initialisierung für einen weiteren Optimierungsvorgang, der die Betragsgleichheit der Spektralanteile der Schichten über mehrere Bilder hinweg berücksichtigt.

Wie die Diskussion gezeigt hat, existieren bereits viele Verfahren zur Trennung von überlagerten Schichten in Bildern bzw. Bildsequenzen. Alle hier vorgestellten Verfahren zur bewegungsbasierten Schichtentrennung, haben eine Gemeinsamkeit: Es stehen nicht genügend Gleichungen für eine perfekte Trennung zur Verfügung, und die Schichten können nur approximativ ermittelt werden. In Anlehnung an das vorangegangene Kapitel möchten wir hier die phasenbasierten Verfahren zur Trennung der bewegten Schichten ([174, 194]) aufgreifen und die notwendigen Schritte für die Erweiterung zu mehr als zwei Überlagerungen vorstellen [163, 116]. Mit Hilfe der Phasenverschiebungen berechnen wir zunächst die Spektren der bewegten Schichten und erhalten dann das gesuchte Ergebnis durch Rücktransformation in den Ortsraum. Leider ist die Trennung der Schichten nicht für alle Frequenzen möglich. Wie wir zeigen werden, sind die Bereiche, an denen sich nicht alle Spektralanteile der Schichten voneinander trennen lassen, durch Geraden bestimmt, die durch den Ursprung im Frequenzraum verlaufen. Nach der Trennung der Spektralanteile interpolieren wir die fehlenden Spektralanteile aus den umliegenden Spektralanteilen der jeweiligen Schicht.

7.1 Trennung global bewegter Schichten im Fourier- raum

Die Trennung der global bewegten Schichten läuft in zwei Stufen ab. Zunächst müssen alle Bewegungen bestimmt werden. Dabei ist es nicht wichtig, mit welchem Verfahren die Bewegungen berechnet wurden, da die Phasenverschiebungen sich auf einfache Art und Weise aus den Bewegungsvektoren berechnen lassen. Nachdem die Berechnung der Phasenverschiebungen abgeschlossen ist, beginnen wir mit der Trennung der fouriertransformierten Schichten im Frequenzraum.

Im vorangegangenen Abschnitt hatten wir bereits gesehen, dass sich die Fouriertransformierten der Bilder

$$F_{t_k}(\boldsymbol{\omega}) = \phi_1^k G_1(\boldsymbol{\omega}) + \phi_2^k G_2(\boldsymbol{\omega}) + \cdots + \phi_N^k G_N(\boldsymbol{\omega}), \quad k = 0, \dots, \quad (7.2)$$

mit $\phi_n = e^{-j2\pi\boldsymbol{\omega} \cdot \mathbf{v}_n \Delta t}$, $n = 1, \dots, N$, jeweils aus einer Summe von Produkten zwischen den Fouriertransformierten der Schichten mit den ihnen zugeordneten Phasenverschiebungen ergibt.

Wir fassen die ersten N Gleichungen aus (7.2) zu dem linearen Gleichungssystem (vgl. (6.10))

$$\begin{pmatrix} F_{t_0} \\ F_{t_1} \\ \vdots \\ F_{t_{N-1}} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \phi_1 & \phi_2 & \cdots & \phi_N \\ \vdots & \vdots & \vdots & \vdots \\ \phi_1^{N-1} & \phi_2^{N-1} & \cdots & \phi_N^{N-1} \end{pmatrix} \begin{pmatrix} G_1 \\ G_2 \\ \vdots \\ G_N \end{pmatrix}, \quad (7.3)$$

zusammen, dessen Lösungsvektor die Spektralanteile der Schichten für eine gegebene Frequenz $\boldsymbol{\omega}$ enthält. In Kurzform schreiben wir dieses Gleichungssystem als $\mathbf{F}_{t_0} = \mathbf{B}_N \mathbf{G}$, mit $\mathbf{F}_{t_0} = (F_{t_0}, \dots, F_{t_{N-1}})$ und $\mathbf{G} = (G_1, \dots, G_N)$ und \mathbf{B}_N obiger Matrix, deren Einträge sich aus Phasenverschiebungen zusammensetzen. Die Matrix \mathbf{B}_N ist invertierbar, falls alle ihre Einträge ϕ_1, \dots, ϕ_N verschieden sind. Ist dies der Fall, so erlauben uns N aufeinander folgende Bilder der Bildsequenz die Schichten voneinander zu trennen, in dem wir \mathbf{B}_N invertieren, d.h.

$$\mathbf{G} = \mathbf{B}_N^{-1} \mathbf{F}_{t_0}. \quad (7.4)$$

Nachdem die Schichtentrennung im Frequenzraum für alle Frequenzen durchgeführt wurde, erhalten wir mit Hilfe der inversen Fouriertransformation die gesuchten Schichten im Ortsraum. Leider ist die Trennung mit dem hier vorgestellten Verfahren nicht für alle Frequenzen möglich.

7.2 Invertierbarkeit der Vandermonde-Matrix

Sind zwei oder mehrere Phasenverschiebungen identisch, so kann die Trennung nicht durchgeführt werden, da \mathbf{B}_N nicht mehr den vollen Rang besitzt und damit nicht mehr invertierbar ist. Die Bereiche im Frequenzraum, in denen zwei oder mehr Phasen identisch sind, werden durch die Gleichung

$$\phi_m = \phi_n \iff e^{j2\pi(\mathbf{v}_m - \mathbf{v}_n) \cdot \boldsymbol{\omega} \Delta t} = 1 \iff (\mathbf{v}_m - \mathbf{v}_n) \cdot \boldsymbol{\omega} = 2k\pi, \quad k = 0, \dots \quad (7.5)$$

bestimmt. Folglich können entlang der durch obige Gleichung beschriebenen Geraden die Frequenzanteile von mindestens zwei Schichten nicht mehr voneinander getrennt werden. Die Faktoren $2k\pi$, $k = 0, \dots$ auf der rechten Seite der Geradengleichung beschreiben die periodische Wiederholung der Geraden im Frequenzraum, die durch die Zeitdiskretisierung entstanden ist. Die größtmögliche Anzahl von Geraden erhalten wir, wenn alle Differenzvektoren $\mathbf{v}_m - \mathbf{v}_n$, $m, n = 1, \dots, N$, $m \leq n$ in unterschiedliche Richtungen zeigen und damit kein Differenzvektor ein Vielfaches des anderen ist. In diesem Fall ergeben sich $\binom{N}{2}$ solcher Geraden. Zeigen zwei Differenzvektoren in dieselbe bzw. in die entgegengesetzte Richtung, beschreiben beide Vektoren die gleiche Gerade und die Anzahl der Geraden reduziert sich um eins. Je nach Konstellation der Bewegungsvektoren kann die Anzahl der Geraden geringer ausfallen und obige Zahl stellt eine obere Schranke für die Anzahl der Geraden dar. Befinden wir uns im Frequenzraum an einem Punkt, an dem zwei Phasen identisch sind, so können wir durch Verringerung des Modells um eine Bewegung die Spektralanteile der übrigen Schichten bestimmen. Sind z.B. $\phi_1 = \phi_2$ so lautet das reduzierte Gleichungssystem

$$\begin{pmatrix} F_{t_0} \\ F_{t_1} \\ \vdots \\ F_{t_{N-2}} \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \phi_1 & \phi_3 & \cdots & \phi_N \\ \vdots & \vdots & & \vdots \\ \phi_1^{N-1} & \phi_3^{N-1} & \cdots & \phi_N^{N-1} \end{pmatrix} \begin{pmatrix} G_1 + G_2 \\ G_3 \\ \vdots \\ G_N \end{pmatrix}, \quad (7.6)$$

mit dem sich die Spektralanteile von G_3, \dots, G_N berechnen lassen. Sind drei Phasen identisch, reduziert sich das Gleichungssystem erneut um eins. Die korrekte Aufstellung der reduzierten Gleichungssysteme erfordert genaue Kenntnisse der identischen Phasen. Da wir die Phasen ohnehin zur Aufstellung der Vandermonde-Matrix benötigen, können wir auch leicht feststellen, welche Phasen an der betrachteten Frequenz identisch sind.

Zum Schluss erhalten wir die gesuchten Schichten im Ortsraum, in dem wir die inverse Fouriertransformation auf alle vollständig bestimmten Spektren der Schichten anwenden. Dazu fehlen uns aber noch die Spektralanteile der Frequenzen, für die die Trennung aufgrund von Singularitäten der Vandermondematrix nicht möglich war.

Eine Lösung für dieses Problem besteht in der Interpolation der fehlenden Frequenzanteile aus den benachbarten Frequenzanteilen der Schichten, für die die Trennung zuvor erfolgreich durchgeführt werden konnte. Dieses Vorgehen erfordert Approximationen und führt zu Fehlern in den Ergebnissen. In den kommenden Beispielen haben wir die fehlenden Frequenzen durch Mittelwertbildung der benachbarten Frequenzanteile interpoliert.

Bei Phasengleichheit zweier oder mehrerer Schichten enthält eine der Lösungen des um eine oder mehrere Schichten reduzierten Gleichungssystems die Summe aller Spektralanteile, deren Phasen identisch sind. Entsprechend können wir als Alternative zur Interpolation diese Summe durch die Anzahl der Summanden teilen und den resultierenden Wert als Spektralanteil in allen Schichten mit identischer Phase einsetzen. In dem obigen reduzierten Gleichungssystem würde beispielsweise den Spektralanteilen von G_1 und G_2 jeweils der Wert $(G_1 + G_2)/2$ zugewiesen. Dies führt zwar nach wie vor zu Fehlern

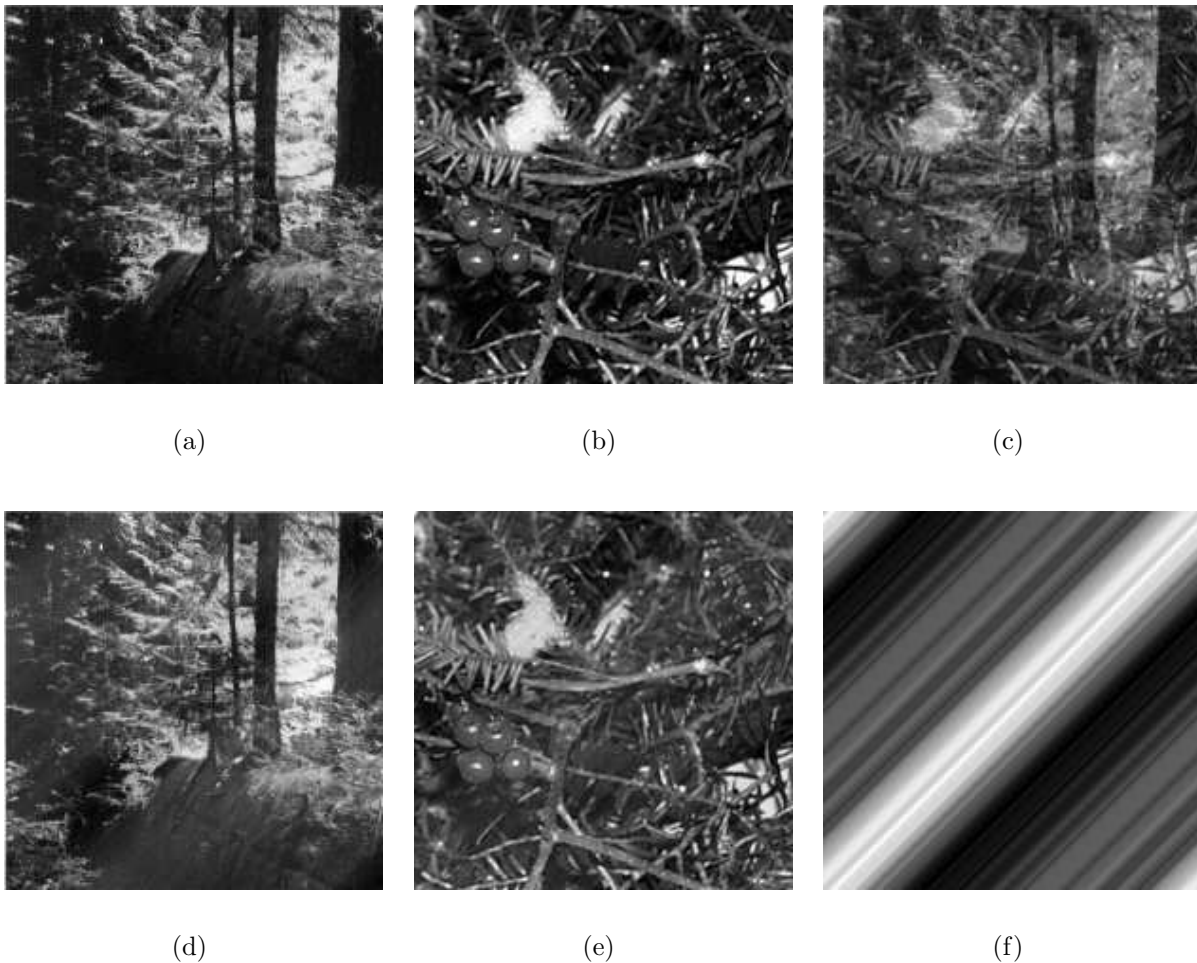


Abbildung 7.1: Trennung der bewegten Schichten (siehe Text).

in den resultierenden Bildern, stellt aber sicher, dass die Summe aller Bilder wieder das Originalbild der Sequenz ergibt.

7.2.1 Beispiele

In Abbildung 7.1 und 7.2 ist jeweils ein synthetisches Beispiel für die Trennung zweier bewegter Schichten dargestellt. Interpoliert wurden die fehlenden Spektralanteile durch Mittelbildung aus benachbarten Spektralanteilen der Schichten. Die Grafiken (a) und (b) zeigen dabei die Originalschichten, die in (c) überlagert wurden, wobei sich die eine Schicht mit der Geschwindigkeit $(1, 0)$ und die andere mit $(0, 1)$ bewegt. In (d) und (e) sind die Ergebnisse nach der Trennung der Schichten zu sehen. Vergleichen wir die Ergebnisse mit den Originalbildern, so fallen Unterschiede auf, die auf Interpolationsfehler zurückzuführen sind. Besonders deutlich wird dieser Fehler in Abbildung 7.1, wenn wir das Bild (b) von dem Bild (e) abziehen und das Ergebnis in (f) normiert darstellen. In Abbildung 7.2 zeigt (f) das Differenzbild zwischen (a) und (d). In beiden Beispielen weist das Fehlerbild ein orientiertes Muster auf und verdeutlicht damit, dass die Ursachen der Unterschiede hauptsächlich durch die Interpolation hervorgerufen wurden. Interpoliert



Abbildung 7.2: Trennung der bewegten Schichten (siehe Text).

wurde dabei nur entlang der Geraden im Frequenzraum, für die die Trennung der Frequenzanteile der Schichten nicht möglich war. Das durch die Interpolation entstandene Fehlerspektrum ist damit nur auf diese Gerade beschränkt, und folglich muss das Fehlerbild im Ort eine orientierte Struktur aufweisen. Optisch scheint jedoch die Auswirkung der Interpolationsfehler auf die Ergebnisse in Abbildung 7.1 weniger ausgeprägt zu sein als in Abbildung 7.2. Damit wird deutlich, dass die Qualität der Ergebnisse zusätzlich noch von den Bildern selbst abhängt.

Ein Ergebnis für die Trennung von drei transparenten Bewegungen ist in Abbildung 7.3 dargestellt. Bild (a) zeigt das Originalbild, das sich aus drei Bildern der vorangegangenen Beispiele zusammensetzt. Die Schichten bewegten sich mit den Geschwindigkeiten $(1, 0)$, $(0, 1)$ und $(1, 1)$ Bildpunkte pro Zeitschritt. Auf eine Schätzung der Bewegungen haben wir hier verzichtet. Die Separierungsergebnisse für den rauschfreien Fall sind in (b), (c) und (d) dargestellt. Da in jeder Schicht eine Interpolation der Frequenzanteile entlang zweier Geraden erforderlich ist, sind Fehlermuster in den Resultaten nicht mehr nur in einer Richtung orientiert, sondern zeigen eine Überlagerung zweier orientierter Muster.

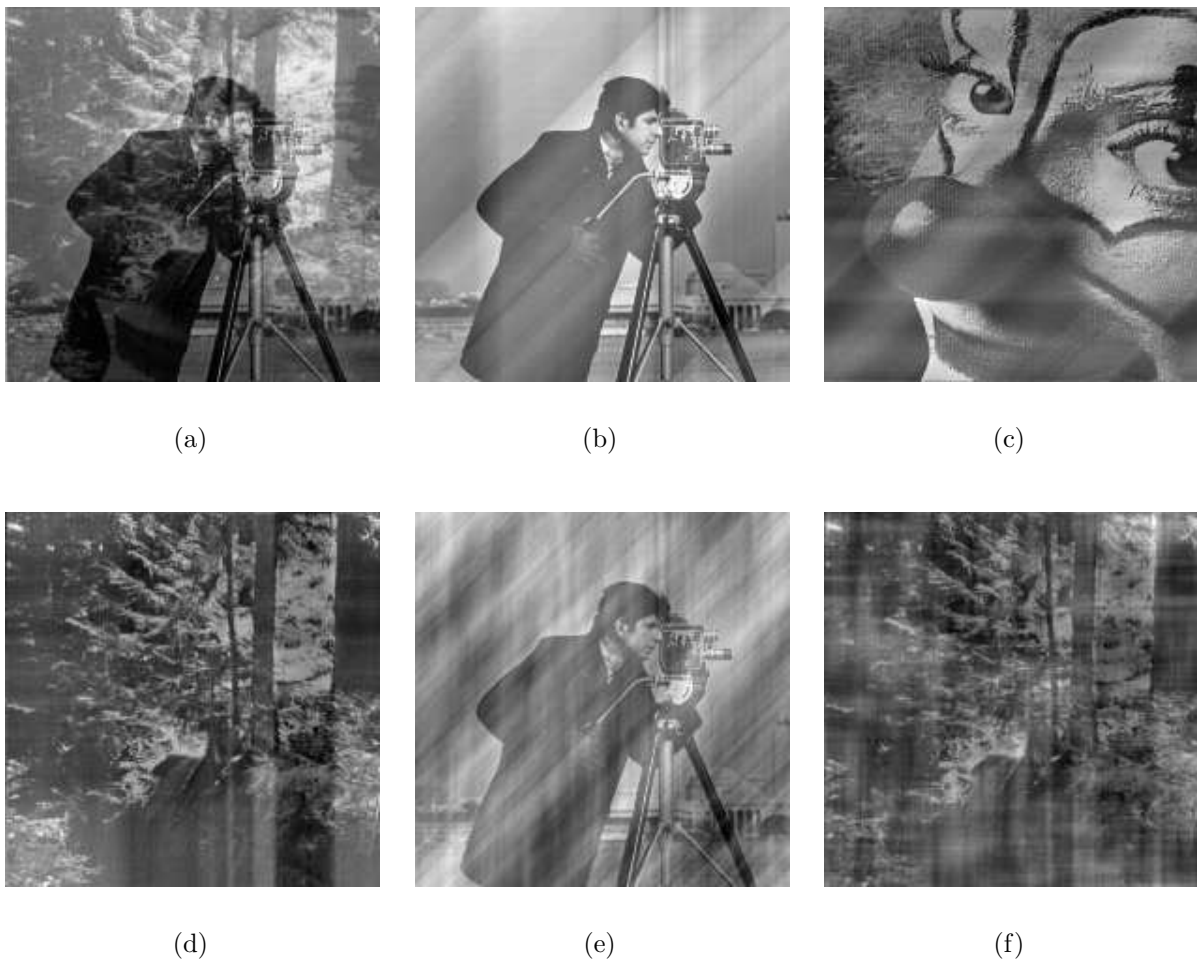


Abbildung 7.3: Trennung von drei überlagerten Schichten (siehe Text)

Die Abbildungen 7.3 (e) und (f) zeigen das Separierungsergebnis von zwei der drei Schichten für die gleiche aber mit einem Signalrauschverhältnis von 40dB verrauschten Sequenz. Nach den Ergebnissen nimmt die Qualität der Separierung mit dem Rauschen deutlich ab und äußert sich auch in einer Frequenzerhöhung der Fehlermuster. Zhou & Kambhamettu [194] bemerkten bereits die Rauschanfälligkeit der phasenbasierten Methode bei der Trennung von zwei Überlagerungen. Zur Reduktion der Rauscheffekte nutzen sie die beiden mit der phasenbasierten Methode ermittelten Schichten als Initialisierung für einen Optimierungsalgorithmus, der unter anderem die Betragsgleichheit der Spektralanteile der jeweiligen Schichten über mehrere Bilder hinweg berücksichtigt, und konnten damit die Rauscheffekte deutlich verringern. In wieweit sich ihre Methode auch auf die Trennung von drei Überlagerungen anwenden lässt, bleibt noch zu untersuchen.

Zum Abschluss möchten wir noch ein Ergebnis für die Schichtentrennung einer realen Bildsequenz zeigen, auch wenn in ihr nur zwei Bewegungen vorkommen. Abbildung 7.4 (a) zeigt ein Originalbild der Monalisasequenz, (b) und (c) das Ergebnis der Schichtentrennung. Für die Schichtentrennung sind zunächst die lokalen Bewegungsvektoren mit Hilfe des regularisierten Bewegungsbestimmungsverfahrens ermittelt worden. Für die Be-



Abbildung 7.4: Schichtentrennung einer realen Bildsequenz. Bild (a) zeigt das Originalbild der Sequenz, (b) und (c) das Separierungsergebnis.

rechnung der Phasenverschiebungen sind die Mittelwerte der Bewegungsvektorfelder der jeweiligen Schichten verwendet worden. Erst nach der Trennung werden die Strukturen auf der Müslipackung optisch erkennbar. Die Interpolationseffekte sind hier jedoch nicht so deutlich zu erkennen wie bei den synthetischen Beispielen. Dies ist wahrscheinlich auf weitere Fehlerstrukturen zurückzuführen, die durch Variationen im Bewegungsfeld entstanden sind.

7.3 Ausblick

Das hier vorgestellte Verfahren erlaubt die Trennung einer beliebigen Anzahl überlagerter und bewegter Schichten, und es wurden zum ersten Mal Trennungsergebnisse für mehr als nur zwei Überlagerungen gezeigt. Ähnlich wie in vielen der anderen bekannten Verfahren stehen jedoch nicht genügend Gleichungen zur Verfügung, um die Schichten exakt voneinander trennen zu können, so dass die Ergebnisse nur eine Approximation der Originale darstellen. Die durch die Interpolation entstandenen Fehler ließen sich eventuell durch eine verbesserte Interpolation reduzieren, die gegebenenfalls spektrale Eigenschaften der Schichten berücksichtigen. Ferner wurde gezeigt, dass die phasenbasierte Methode rauschanfällig ist. In wie weit die Nachverarbeitungsmethode zur Reduktion der Rauscheffekte von Zhou & Kambhamettu [194] für den Fall von zwei Überlagerungen auf drei Überlagerungen übertragbar ist, bleibt noch zu klären. Dennoch können diese Ergebnisse als Initialisierung für andere Verfahren (z.B. [164]) im Ort genutzt werden. In den gezeigten synthetischen Beispielen waren die Bewegungen an jedem Bildpunkt gleich. In der Praxis wird dies jedoch in der Regel nicht der Fall sein, und die Schichtentrennung muss lokal durchgeführt werden.

Kapitel 8

Block Matching für überlagerte Bewegungen

Als Alternative zu den differentiellen Bewegungsbestimmungsverfahren werden Blockvergleichsmethoden (Block Matching) in vielen Anwendungen zur Bewegungsbestimmung genutzt und sind Bestandteil gängiger Videokomprimierungsstandards wie der H.261, H.263 Standard und MPEG-Familie [42, 173, 150, 149]. Den Bewegungsvektor für einen Bildpunkt erhalten wir, indem wir einen Bereich oder Block um den Bildpunkt herum ausschneiden, und anschließend in dem nachfolgenden Bild nach der neuen Position dieses Blockes suchen. Um zwei Blöcke miteinander vergleichen zu können, werden entsprechende Übereinstimmungsmaße benötigt. Zu diesem Zweck finden beispielsweise Kreuzkorrelationsmaße sowie Maße basierend auf der Summe absoluter oder quadrierter Differenzen [124, 30, 157, 12] Verwendung. Die Übereinstimmungsmaße geben uns nur Auskunft darüber, wie gut der zu suchende Block mit dem gerade betrachteten Block im nachfolgenden Bild korreliert. Folglich müssen die Übereinstimmungsmaße für alle Bewegungsvektoren, die in der Sequenz vorkommen könnten, ausgewertet werden. Anschließend entspricht der Vektor, für den die Übereinstimmung am besten ausfällt, dem gesuchten Bewegungsvektor. Das vollständige Durchsuchen des Lösungsraums gestaltet sich als sehr rechenaufwendig und dementsprechend wurden eine Vielzahl von Algorithmen entworfen, die den Rechenaufwand reduzieren sollen. Der Dreistufensuch- [100], der Diamantensuchalgorithmus [166] sowie das sukzessive Eliminierungsverfahren [101] stellen die bekanntesten Suchverfahren dar. Eine Gegenüberstellung dieser und anderer Suchmethoden findet sich zum Beispiel in [103, 89]. Die ersten beiden Verfahren [100, 166] und viele ihrer Artverwandten basieren auf der Annahme, dass die Übereinstimmungsmaße mit der Entfernung von dem minimalen Punkt steigen und damit eine konvexe Funktion bilden. Da nachgewiesenerweise diese Annahme nicht immer in realen Sequenzen zutrifft [45, 43], besteht die Gefahr, dass diese Verfahren in einem lokalen Minimum stecken bleiben. Mit der vollständigen Suche findet man jedoch immer das Optimum. Bei den meisten Suchverfahren wird die Geschwindigkeitsbestimmung im ersten Schritt mit Integerpräzision durchgeführt und bei Bedarf in einem zweiten Schritt um Subpixelschritte verfeinert. Sollen die Geschwindigkeitsvektoren auch im Subpixelbereich bestimmt werden, erfordert dies weitere Interpolationsschritte. Davis et. al. [50] zeigten in diesem Zusammenhang, dass Block-Matching Verfahren, die für die Subpixelgeschwindigkeitsbestimmung bilineare Interpolationsmethoden und den mittleren quadratischen Fehler als Übereinstimmungsmaß

verwenden, eine äquivalente Darstellung der optischen Flussgleichung basierend auf den Ableitungen erster Ordnung sind. Um einen gesteigerten Zeitaufwand durch zusätzliche Interpolationsmaßnahmen und Auswertungsschritte des Übereinstimmungsmaßes zu umgehen, versuchen einige Ansätze das Übereinstimmungsmaß direkt aus den Werten an den Integerpositionen zu interpolierten [167, 144, 44]. Trotz dieser Einschränkungen ist Block Matching eine häufig verwendete Methode zur Bewegungsbestimmung. Die große Beliebtheit verdankt diese Methode ihrer einfachen Implementierbarkeit, ihrer Eignung zur Bestimmung von schnellen Bewegungen, wie sie häufig in Videosequenzen auftreten, sowie ihrer Robustheit gegenüber Rauschen und kleinen Störungen.

In diesem Kapitel machen wir es uns zur Aufgabe, ein Block Matching Verfahren zur Bestimmung von transparent überlagerten Bewegungen herzuleiten [159, 161, 162]. Im Unterscheid zu dem konventionellen Block Matching, benötigen wir dazu mehrere Blöcke, die über mehrere Bilder verteilt sind. Die Anzahl der Blöcke und Bilder ist von der Anzahl der Bewegungen und die Anordnung der Blöcke von den Bewegungsvektoren selbst abhängig.

Entsprechend der Anzahl der Bewegungen ist für jeden Bildpunkt das zugehörige Bewegungsmodell auszuwählen. Um zwischen dem Auftreten von einer, zwei oder mehrerer Bewegungen unterscheiden zu können, benötigen wir daher entsprechende Konfidenzkriterien. In diesem Zusammenhang entwerfen wir ein stochastisches Übereinstimmungskriterium, mit dem wir entscheiden können, ob der Übereinstimmungswert durch das Rauschen erklärbar ist oder nicht und ein hierarchisches Vorgehen der Bewegungsbestimmung ermöglicht.

Übereinstimmungsmaße, die auf der Differenzbildung der Bildpunkte des Korrespondenzmit denen des Referenzbildes basieren, eignen sich besonders gut zur stochastischen Beschreibung und Modellierung des Rauschens. Zum Beispiel nutzen Aach & Kunz [4] diese Möglichkeit zur Bayes'schen Bewegungsbestimmung in Röntgenaufnahmen, in dem sie das Rauschen durch ein generalisiertes Gaußmodell beschreiben und apriori Erwartungen an die räumliche und zeitliche Glattheit des Bewegungsfeldes durch ein Markov-Zufallsfeld beschreiben. Stiller [156] nutzt ebenfalls ein generalisiertes Gaußmodell zur Beschreibung des Rauschens in der bewegungskompensierten Differenz und ein Markov-Zufallsfeld zur Formulierung der räumlichen und zeitlichen Glattheitsbeziehungen des Bewegungsfeldes. Simultan zur Bestimmung des Bewegungsfeldes ermittelt er eine Objektsegmentierung. Die Segmentierung erlaubt es ihm, Diskontinuitäten im Bewegungsfeld, entstanden durch Verdeckungen, zu erkennen und damit die räumliche Glättung des Bewegungsfeldes über die Verdeckungsgrenzen hinaus zu unterbinden.

Das im ersten Teil entwickelte Blockmatchingverfahren zur Bestimmung transparenter Bewegungen berücksichtigt keine räumlichen und zeitlichen Beziehungen der Bewegungsvektoren. Zur weiteren Erhöhung der Rauschrobustheit integrieren wir die Erkenntnisse des ersten Abschnittes im zweiten Teil des Kapitels in einem Bayes'sche Ansatz zur Bewegungsbestimmung, der Regularisierungen der Bewegungsfelder ermöglicht. Ähnlich wie in [156, 4] formulieren wir die räumliche und zeitliche Glattheit der Bewegungsfelder durch ein Markov-Zufallsfeld. Simultan mit den Bewegungen bestimmen wir eine Segmentie-

nung, die die Anzahl der beobachteten Bewegungen in jedem Bildpunkt beschreibt und damit die Auswahl des jeweils zutreffenden Bewegungsmodells ermöglicht.

8.1 Die Block Matching Bedingung

Die Verwendung eines einzelnen Blockes ist für die transparent überlagerten Bewegungen nicht mehr ausreichend, da er nur die Kompensation einer Bewegung erlaubt. Vielmehr benötigen wir eine von den Bewegungsvektoren abhängige Vorschrift, die angibt, wie wir mehrere Blöcke über mehrere aufeinander folgende Bilder anordnen bzw. kombinieren müssen, um dann mit Hilfe eines Übereinstimmungsmaßes feststellen zu können, ob es sich hierbei um die zu testenden Bewegungen handelt.

Wie bei der Bestimmung der Phasenverschiebungen gehen wir wieder von dem zeitdiskreten Bildmodell

$$f_k(\mathbf{x}) = f(\mathbf{x}, k) = g_1(\mathbf{x} - k\mathbf{v}_1) + g_2(\mathbf{x} - k\mathbf{v}_2) + \cdots + g_N(\mathbf{x} - k\mathbf{v}_N), \quad k = 0, 1, \dots \quad (8.1)$$

aus, wobei wir jetzt zur Vereinfachung $\Delta t = 1$ annehmen. Wenden wir die zweidimensionale Fouriertransformation auf jedes Bild obiger Gleichung an, so erhalten wir

$$F_k(\boldsymbol{\omega}) = \phi_1^k G_1(\boldsymbol{\omega}) + \phi_2^k G_2(\boldsymbol{\omega}) + \cdots + \phi_N^k G_N(\boldsymbol{\omega}), \quad k = 0, 1, \dots \quad (8.2)$$

mit den durch die Bewegungen hervorgerufenen Phasenverschiebungen $\phi_n = e^{-j2\pi\boldsymbol{\omega}\cdot\mathbf{v}_n}$, $n = 1, \dots, N$ und dem Frequenzvektor $\boldsymbol{\omega} = (\omega_x, \omega_y)$. Durch Einführen des komplexen Polynoms

$$p(z) = (z - \phi_1) \cdots (z - \phi_N) = a_0 z^N + a_1 z^{N-1} + \cdots + a_N \quad (8.3)$$

mit den Koeffizienten

$$\begin{aligned} a_0 &= 1 \\ a_1 &= -\sum_{i=1}^N \phi_i \\ a_2 &= \sum_{i<l} \phi_i \phi_l \\ a_3 &= -\sum_{i<l<k} \phi_i \phi_l \phi_k \\ &\vdots \\ a_N &= (-1)^N \phi_1 \phi_2 \cdots \phi_N. \end{aligned}$$

hatten wir in Kapitel 6 folgenden Zusammenhang hergeleitet:

$$a_0 F_N + a_1 F_{N-1} + \cdots + a_N F_0 = 0. \quad (8.4)$$

Wenden wir die inverse Fouriertransformation auf Gleichung (8.4) an, so erhalten wir

$$\begin{aligned} e(f, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N) &:= (-1)^N f_0(\mathbf{x} - \mathbf{v}_1 - \cdots - \mathbf{v}_N) + \cdots \\ &- \sum_{i<l} f_{N-2}(\mathbf{x} - \mathbf{v}_i - \mathbf{v}_l) + \sum_i f_{N-1}(\mathbf{x} - \mathbf{v}_i) - f_N(\mathbf{x}) = 0, \end{aligned} \quad (8.5)$$

denn die Produkte der Phasenverschiebungen führen zu einer Aneinanderreihung der Bewegungsvektoren im Ortsraum. Da jedes a_n aus einer Summe von $\binom{N}{n}$ Ausdrücken besteht, setzt sich der mittlere Teil von Gleichung (8.5) aus $\sum_{n=0}^N \binom{N}{n} = 2^N$ Ausdrücken zusammen.

Gleichung (8.5) beschreibt, wie sich das $N + 1$ -te Bild unter der Verwendung der Bewegungsvektoren $\mathbf{v}_1, \dots, \mathbf{v}_N$ aus den N Vorgängerbildern erzeugen lässt. Damit stellt diese Gleichung die Basis der Block Matching Methoden für eine beliebige Anzahl von überlagerten Bewegungen dar. Im Falle einer Bewegung reduziert sich Gleichung (8.5) zu der klassischen Block Matching Bedingung

$$e(f, \mathbf{x}, \mathbf{v}) = f_0(\mathbf{x} - \mathbf{v}) - f_1(\mathbf{x}) = 0 \quad (8.6)$$

und für zwei transparente Bewegungen zu

$$e(f, \mathbf{x}, \mathbf{v}_1, \mathbf{v}_2) = f_0(\mathbf{x} - \mathbf{v}_1 - \mathbf{v}_2) - f_1(\mathbf{x} - \mathbf{v}_1) - f_1(\mathbf{x} - \mathbf{v}_2) + f_2(\mathbf{x}) = 0. \quad (8.7)$$

8.1.1 Kausalität und Verfeinerung

Die nach Gleichung (8.5) bestimmten Bewegungsvektoren werden dem Bild f_N zugeordnet. Damit setzen wir voraus, dass die ermittelten Bewegungsvektoren dem Bild zugeordnet werden, dessen Argument in der Block Matching Bedingung unabhängig von den Bewegungsvektoren ist. Obige Bedingung nutzt N aufeinander folgende Bilder, um die Bewegungsvektoren für das $N + 1$ -te Bild zu bestimmen und ist daher kausal. Für einen Zeitpunkt k können wir jedoch ältere Ergebnisse verfeinern, indem wir die Block Matching Bedingung in einer anderen Weise nutzen. Wie wir gleich zeigen werden, erlauben einfache Substitutionen die Bestimmung der Bewegungsvektoren für jedes der verbleibenden N Bilder f_0, \dots, f_{N-1} . Die Bewegungsbestimmung für diese Bilder ist zwar nicht mehr kausal, führt aber nur zu einer konstanten Verzögerung. Um die Bewegungen für das Bild f_0 zu bestimmen, verwenden wir die Substitution $\mathbf{y} = \mathbf{x} - \mathbf{v}_1 - \dots - \mathbf{v}_N$ und erhalten für Gleichung (8.5)

$$\begin{aligned} e(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N) := & (-1)^N f_0(\mathbf{y}) + (-1)^{N-1} \sum_{i < l} f_1(\mathbf{y} + \mathbf{v}_i) + \\ & (-1)^{N-2} \sum_i f_2(\mathbf{y} + \mathbf{v}_i + \mathbf{v}_j) + \dots - f_N(\mathbf{y} + \mathbf{v}_1 + \dots + \mathbf{v}_N) = 0. \end{aligned} \quad (8.8)$$

Die Bewegungen werden immer dem Bild zugeordnet, dessen Argument unabhängig von den Bewegungsvektoren ist. Damit die Bewegungen dem Bild f_k zugeordnet werden, wählen wir für die Substitution das Argument des Bildes aus Gleichung (8.5). Da in Gleichung (8.5) viele Bilder mehrfach vorkommen, können mehrere unterschiedliche Substitutionen für dieses Bild gewählt werden. Durch Anwendung der Substitutionen

$$\begin{aligned} \mathbf{y} &= \mathbf{x} - \mathbf{v}_i, & i &= 1, \dots, N \\ \mathbf{y} &= \mathbf{x} - \mathbf{v}_i - \mathbf{v}_j, & i &< j \\ \mathbf{y} &= \mathbf{x} - \mathbf{v}_i - \mathbf{v}_j - \mathbf{v}_k, & i &< j < k \\ &\vdots \\ \mathbf{y} &= \mathbf{x} - \mathbf{v}_1 - \dots - \mathbf{v}_N. \end{aligned} \quad (8.9)$$

Block Matching Bedingung	Substitution
$f_0(\mathbf{x} - \mathbf{v}_1 - \mathbf{v}_2) - f_1(\mathbf{x} - \mathbf{v}_1) - f_1(\mathbf{x} - \mathbf{v}_2) + f_2(\mathbf{x}) = 0$	
$f_0(\mathbf{y} - \mathbf{v}_2) - f_1(\mathbf{y}) - f_1(\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2) + f_2(\mathbf{y} + \mathbf{v}_1) = 0$	$\mathbf{y} = \mathbf{x} - \mathbf{v}_1$
$f_0(\mathbf{y} - \mathbf{v}_1) - f_1(\mathbf{y} - \mathbf{v}_1 + \mathbf{v}_2) - f_1(\mathbf{y}) + f_2(\mathbf{y} + \mathbf{v}_2) = 0$	$\mathbf{y} = \mathbf{x} - \mathbf{v}_2$
$f_0(\mathbf{y}) - f_1(\mathbf{y} + \mathbf{v}_2) - f_1(\mathbf{y} + \mathbf{v}_1) + f_2(\mathbf{y} + \mathbf{v}_1 + \mathbf{v}_2) = 0$	$\mathbf{y} = \mathbf{x} - \mathbf{v}_1 - \mathbf{v}_2$

Tabelle 8.1: Weitere Block Matching Bedingungen, die sich aus den angegebenen Substitutionen ergeben.

erhalten wir alle Block Matching-Bedingungen, die aus $N + 1$ aufeinander folgenden Bildern erzeugt werden können. In Tabelle 8.1 sind alle möglichen Block Matching Bedingungen für zwei Bewegungen aufgelistet, die sich durch die Anwendung obiger Substitutionen auf Gleichung (8.5) ergeben. In Gleichung (8.5) treten 2^N unterschiedliche Verschiebungen auf und damit auch $2^N - 1$ mögliche Substitutionen. Folglich existieren für N -Bewegungen 2^N unterschiedliche Block Matching Bedingungen. Weiter tritt jedes Bild f_k $\binom{N}{k}$ -mal in Gleichung (8.5) auf und wir haben daher $\binom{N}{k}$ Möglichkeiten, die Bewegungen für dieses Bild zu bestimmen. Mit Hilfe dieser zusätzlichen Bedingungen lassen sich die Endergebnisse des Block Matchings verbessern. Bezeichne zum Beispiel $\{e_1(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N), \dots, e_l(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N)\}$ eine Menge von l ausgewählten Block Matching Bedingungen, so könnte eine Bedingung, die diese Gleichungen berücksichtigt, wie folgt lauten:

$$e(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N) = \sum_{i=1}^l e_i(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N) = 0. \quad (8.10)$$

Die Verwendung der zusätzlichen Block Matching Bedingungen hat noch einen weiteren Vorteil. Sie ermöglichen uns die Bestimmung der Geschwindigkeitsvektoren sowohl für das erste als auch für das letzte Bild unserer Sequenz. Die Ergebnisse, die wir hier vorstellen werden, sind jedoch nur mit der initialen Bedingung aus Gleichung (8.5) berechnet worden. Die Frage welcher Mehrgewinn durch die gleichzeitige Verwendung mehrerer Block Matching Bedingungen zu erzielen ist, bleibt daher noch zu klären.

8.2 Hierarchischer Algorithmus für transparente und verdeckte Bewegungen

Aus den im vorangegangenen Abschnitt entwickelten Block Matching Bedingungen lassen sich eine Reihe unterschiedlicher Block Matching Verfahren für die Bestimmung von N Bewegungen ableiten. Wir werden einen hierarchischen Algorithmus vorstellen, der aus einer Kombination statistischer Modellbeschreibung und hierarchischer Entscheidungsfindung besteht. Zuerst wenden wir das einfache Bewegungsmodell mittels einer vollständigen Suche auf die Bildsequenz an. Ist die ermittelte Übereinstimmung nicht ausreichend genug, so wird die einfache Bewegungshypothese verworfen, und der Algorithmus fährt mit dem Test auf zwei transparent überlagerte Bewegungen fort. Anderenfalls werden die einfachen Bewegungen bestimmt und verwendet. Muss die Annahme zweier transparenter Bewegungen ebenfalls verworfen werden, so versucht der Algorithmus ein Verdeckungsmodell, das wir in diesem Abschnitt entwickeln werden, mit der Bildsequenz in Übereinstimmung

zu bringen. Diese Methode ist auf eine beliebige Anzahl transparenter Bewegungen erweiterbar. Der Einfachheit halber nehmen wir für das Bildmodell weißes gaußverteilt Rauschen an, das zu einem auf der χ^2 -Verteilung basierenden Signifikanztest führt.

8.2.1 Stochastisches Bildmodell

Neben Verdeckungen und weiteren Störungen wird die Block Matching Bedingung durch Rauschen verletzt. Zusätzliche Information über die Rauschverteilung kann bei der Entscheidungsfindung helfen, ob der beobachtete Fehlerwert nach dem Block Matching durch das Rauschmodell erklärbar ist. Unterschiedliche Bewegungstypen führen zu unterschiedlichen Verteilungen der Übereinstimmungsmaße und sind daher für die Auswahl des wahrscheinlichsten Bewegungsmodells von großer Bedeutung.

Im Folgenden nehmen wir an, dass die beobachteten Bildintensitäten an jeder räumlichen und zeitlichen Position durch Rauschen gestört sind. Das ermittelte $f_k(\mathbf{x})$ setzt sich aus dem tatsächlichen Wert $\bar{f}_k(\mathbf{x})$ und einem Rauschanteil $\epsilon_k(\mathbf{x})$ zusammen:

$$f_k(\mathbf{x}) = \bar{f}_k(\mathbf{x}) + \epsilon_k(\mathbf{x}), \quad \epsilon_k(\mathbf{x}) \sim \mathcal{N}(0, \sigma^2) \quad (8.11)$$

für $k = 0, 1, \dots$. Der Ausdruck $\mathcal{N}(0, \sigma^2)$ beschreibt dabei die Gaußverteilung mit Mittelwert null und Varianz σ^2 . Setzen wir dieses Bildmodell in Gleichung (8.5), so erhalten wir

$$e(f, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N) = e(\bar{f}, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N) + \varepsilon_N(\mathbf{x}), \quad (8.12)$$

mit

$$\begin{aligned} \varepsilon_N(\mathbf{x}) = & (-1)^N \epsilon_0(\mathbf{x} - \mathbf{v}_1 - \dots - \mathbf{v}_N) + \dots \\ & - \sum_{i < l} \epsilon_{N-2}(\mathbf{x} - \mathbf{v}_i - \mathbf{v}_l) + \sum_i \epsilon_{N-1}(\mathbf{x} - \mathbf{v}_i) - \epsilon_N(\mathbf{x}). \end{aligned} \quad (8.13)$$

Die rechte Seite obiger Gleichung besteht aus 2^N Ausdrücken. Unter der Annahme, dass diese Ausdrücke statistisch unabhängig sind und dass die korrekten Bewegungsvektoren verwendet wurden, erhalten wir

$$\varepsilon_N(\mathbf{x}) \sim \mathcal{N}(0, 2^N \sigma^2). \quad (8.14)$$

Der beobachtete Fehlerwert (Residual) der Block Matching Bedingung ist demnach normalverteilt mit Mittelwert null und Varianz $2^N \sigma^2$.

Allerdings gilt die Annahme der Rauschunabhängigkeit nicht, wenn zwei oder mehrere Argumente, die auf das gleiche Bild f_n in Gleichung (8.5) angewendet werden, identisch sind. Bei drei oder weniger Bewegungen kann diese Situation nicht eintreten. Bei vier Bewegungen könnte zum Beispiel $\mathbf{v}_1 + \mathbf{v}_2 = \mathbf{v}_3 + \mathbf{v}_4$ sein. Diesen Fall können wir jedoch während des Suchvorganges detektieren und die Varianz entsprechend anpassen. Für eine perfekte Übereinstimmung mit dem transparenten Bewegungsmodell, können wir daher das bewegungskompensierte Residual durch

$$e(f, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N) = \varepsilon_N(\mathbf{x}) \sim \mathcal{N}(0, 2^N \sigma^2). \quad (8.15)$$

modellieren, wobei wir an dieser Stelle nicht weiter auf die oben genannten Sonderfälle eingehen möchten, da sie aufgrund der hohen Bewegungsanzahl eher von akademischer Natur sind. Folglich ist die Summe der quadrierten Differenzen BM_N des Blockes χ^2 verteilt mit $|\mathbf{B}|$ Freiheitsgeraden, d.h.

$$BM_N(\mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N) = \frac{1}{2^N \sigma^2} \sum_{\mathbf{y} \in \mathbf{B}} e_N(f, \mathbf{y}, \mathbf{v}_1, \dots, \mathbf{v}_N)^2 \sim \chi^2(|\mathbf{B}|), \quad (8.16)$$

wobei \mathbf{B} die Menge der Pixel des gerade betrachteten Blockes und $|\mathbf{B}|$ die Anzahl der Elemente in \mathbf{B} beschreibt.

Im Folgenden werden wir obige Gleichung als Basis für einen Block Matching Algorithmus verwenden. Anstelle der quadratischen Funktion können jedoch auch andere strikt monotone Funktionen auf das bewegungskompensierte Residual angewendet und als Block Matching Kriterium verwendet werden. BM_N wird dann aber im Allgemeinen nicht mehr χ^2 -verteilt sein.

8.2.2 Einfache und zweifach transparente Bewegungen

Im Falle einfacher Bewegungen ist die Block Matching Bedingung durch die Differenz des bewegungskompensierten Bildes mit dem nachfolgenden Bild beschrieben und wir bestimmen den Bewegungsvektor durch minimieren der Funktion

$$BM_1(\mathbf{x}, \mathbf{v}) = \frac{1}{2\sigma^2} \sum_{\mathbf{y} \in \mathbf{B}} (f_0(\mathbf{y} - \mathbf{v}) - f_1(\mathbf{y}))^2. \quad (8.17)$$

Ähnlich ist im Falle von zwei Bewegungen die Funktion

$$BM_2(\mathbf{x}, \mathbf{v}_1, \mathbf{v}_2) = \frac{1}{4\sigma^2} \sum_{\mathbf{y} \in \mathbf{B}} e(f, \mathbf{y}, \mathbf{v}_1, \mathbf{v}_2)^2 \quad (8.18)$$

hinsichtlich der Vektoren \mathbf{v}_1 und \mathbf{v}_2 zu minimieren.

Tritt innerhalb des Blockes \mathbf{B} nur eine Bewegung auf, zum Beispiel $f_1(\mathbf{x}) = f_0(\mathbf{x} - \mathbf{v})$, so wird der Wert $BM_1(\mathbf{x}, \mathbf{v})$ im rauschfreien Fall für den korrekten Bewegungsvektor \mathbf{v} klein sein. Enthält \mathbf{B} jedoch zwei Bewegungen, so ist der Wert BM_1 für jeden Vektor \mathbf{v} deutlich von null verschieden, da ein Vektor nicht in der Lage ist, zwei Bewegungen gleichzeitig zu kompensieren. Der Wert von $BM_2(\mathbf{x}, \mathbf{v}_1, \mathbf{v}_2)$ hingegen ist für die korrekten Bewegungsvektoren \mathbf{v}_1 und \mathbf{v}_2 klein. Anhand der Block Matching Werte können wir daher entscheiden, welcher der beiden Bewegungstypen in dem gerade betrachteten Block vorkommt. Um eine solche Entscheidung durchführen zu können, benötigen wir Konfidenzmaße ähnlich wie die Maße H , K , und S bei den hierarchischen Verfahren aus Kapitel 3. Das hier verwendete Gaußmodell wurde zum Beispiel schon in [1, 4], eingesetzt, könnte alternativ jedoch durch ein verallgemeinertes Gaußmodell [2, 156] ersetzt werden.

8.3 Verdeckte Bewegungen

Verdeckte Bewegungen erfüllen an der Verdeckungsgrenze weder Gleichung (8.6) noch (8.7), da sie nicht durch das additive Modell (8.1) beschrieben werden können. Ist die Bewegungsbestimmung an einer Verdeckungsgrenze erwünscht, ist eine Anpassung des Bildmodells zur Beschreibung verdeckender Bewegungen erforderlich. Zu diesem Zweck modellieren wir die Verdeckung der Schicht g_2 durch g_1 durch das Bildmodell

$$f_k(\mathbf{x}) = \gamma(\mathbf{x} - k\mathbf{v}_1)g_1(\mathbf{x} - k\mathbf{v}_1) + (1 - \gamma(\mathbf{x} - k\mathbf{v}_1))g_2(\mathbf{x} - k\mathbf{v}_2). \quad (8.19)$$

Die Funktion γ hat für Positionen, an denen g_1 die Schicht g_2 verdeckt, den Wert eins und sonst den Wert null (siehe auch [64]). Wenden wir nun Gleichung (8.7) für transparente Bewegungen auf das Verdeckungsmodell an, so erhalten wir

$$e(f, \mathbf{x}, \mathbf{v}_1, \mathbf{v}_2) = (\gamma(\mathbf{x} - 2\mathbf{v}_1) - \gamma(\mathbf{x} - \mathbf{v}_1 - \mathbf{v}_2)) \quad (8.20) \\ (g_2(\mathbf{x} - \mathbf{v}_1 - \mathbf{v}_2) - g_2(\mathbf{x} - 2\mathbf{v}_1)),$$

wobei die Schicht g_1 in der rechten Seite der Gleichung nicht vorkommt. In Abhängigkeit von den Bewegungsvektoren kann die Differenz zwischen den beiden γ -Funktionen in obiger Gleichung einen von null verschiedenen Wert ergeben, was insbesondere in der Nähe der Verdeckungsgrenze der Fall sein wird. Beabsichtigen wir die transparente Block Matching Bedingung zur Bestimmung der beiden Bewegungsvektoren um die Verdeckungsgrenze herum einzusetzen, so werden wir selbst für die korrekten Bewegungsvektoren \mathbf{v}_1 und \mathbf{v}_2 hohe Werte für $BM_2(\mathbf{x}, \mathbf{v}_1, \mathbf{v}_2)$ bekommen. Die Größe dieses Bereiches um die Verdeckungsgrenze herum hängt dabei von der Differenz der Bewegungsvektoren ab. Die Anwendung der Substitution $\mathbf{y} = \mathbf{x} - 2\mathbf{v}_1$ auf die rechte Seite in obiger Gleichung ergibt

$$e(f, \mathbf{x}, \mathbf{v}_1, \mathbf{v}_2) = (\gamma(\mathbf{y}) - \gamma(\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2))(g_2(\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2) - g_2(\mathbf{y})), \quad (8.21)$$

und verdeutlicht, dass der ‘‘Störbereich’’ einen Streifen mit der maximalen Breite von $|\mathbf{v}_1 - \mathbf{v}_2|$ um die Verdeckungsgrenze herum bildet. Breiter kann dieser Bereich nicht werden, da der Auswertungsabstand zwischen beiden γ -Funktionen genau $|\mathbf{v}_1 - \mathbf{v}_2|$ beträgt. Es besteht jedoch die Möglichkeit, dass er kleiner ausfällt. Zum Beispiel hat im Falle einer geradlinig verlaufenden Verdeckungsgrenze dieser Bereich eine Breite von $|\mathbf{N} \cdot (\mathbf{v}_1 - \mathbf{v}_2)|$, wobei es sich bei dem Vektor \mathbf{N} um den Einheitsnormalenvektor der Verdeckungsgrenze handelt. In Abbildung 8.1 ist ein Beispiel für die Breite des Störbereiches bei einer geradlinig verlaufenden Verdeckungsgrenze dargestellt. Der Bereich erstreckt sich dabei von der gestrichelten Linie bis zur Verdeckungsgrenze. Mit Hilfe der Auswertungspunkte \mathbf{y} und $\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2$ für Gleichung (8.21) soll die Einleitung des Störbereiches verdeutlicht werden. Analog beschreiben die Auswertungspunkte \mathbf{y}' und $\mathbf{y}' + \mathbf{v}_1 - \mathbf{v}_2$ das Verlassen des Störbereiches. Die Grenzen des Störbereiches sind parallel zur Verdeckungsgrenze. Die Breite des Störbereiches lässt sich aus dem Dreieck, das durch die Punkte \mathbf{y} , $\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2$ und \mathbf{y}' festgelegt ist, bestimmen. Der kleinste Winkel α zwischen den Vektoren $\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2 - \mathbf{y} = \mathbf{v}_1 - \mathbf{v}_2$ und $\mathbf{y}' - \mathbf{y}$ stimmt mit dem kleinsten Winkel zwischen $\mathbf{v}_1 - \mathbf{v}_2$ und \mathbf{N} überein. Nach den Cosinusetzen ist

$$\cos(\alpha) = \frac{|\mathbf{N} \cdot (\mathbf{v}_1 - \mathbf{v}_2)|}{|\mathbf{v}_1 - \mathbf{v}_2|} = \frac{|(\mathbf{y} + \mathbf{v}_1 - \mathbf{v}_2) - \mathbf{y}|}{|\mathbf{y}' - \mathbf{y}|} \quad (8.22)$$

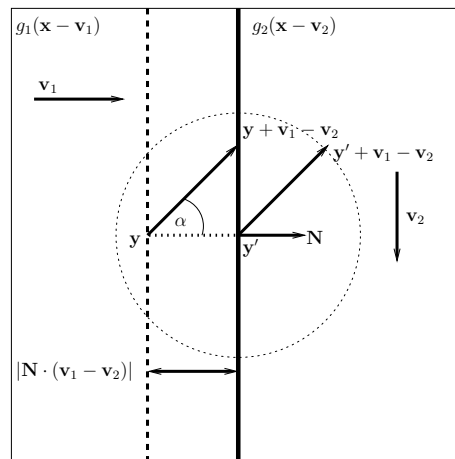


Abbildung 8.1: Dargestellt ist ein Beispiel für die Größe des Störbereiches im Falle einer geradlinig verlaufenden Verdeckungsgrenze. Dabei verdeckt die Schicht g_1 die Schicht g_2 , deren Verdeckungsgrenze durch die schwarze Gerade beschrieben ist. Der Störbereich beginnt bei der gestrichelten Linie und endet an der Verdeckungsgrenze. Der Kreis soll verdeutlichen, dass die Breite des Störbereiches in diesem Fall kleiner als $|\mathbf{v}_1 - \mathbf{v}_2|$ ist.

und folglich ist der Abstand zwischen \mathbf{y} und \mathbf{y}' durch $|\mathbf{y}' - \mathbf{y}| = |\mathbf{N} \cdot (\mathbf{v}_1 - \mathbf{v}_2)|$ gegeben.

Aufgrund dieser Störung ist nicht mehr garantiert, dass das Minimum von BM_2 mit den korrekten Bewegungsvektoren übereinstimmt. Lassen wir jedoch diesen Störbereich nicht mit in die Berechnung des transparenten Bewegungsmodells einfließen, reduziert sich das Problem der Bewegungsbestimmung an der Verdeckungsgrenze auf das der transparenten Bewegungsbestimmung. Allerdings müssen dafür die Position und die Lage der Verdeckungsgrenze im Voraus bekannt sein. In den meisten Fällen wird uns diese Information nicht zur Verfügung stehen. Lösen werden wir dieses Problem mittels eines Zweiphasenansatzes, den wir im Folgenden vorstellen werden.

8.3.1 Wahl des adäquaten transparenten Bewegungsmodells

Eine Möglichkeit, das wahrscheinlichste Bewegungsmodell auszuwählen, bietet die Verwendung von Diskriminanzfunktionen oder, im Falle von zwei Bewegungen, ein Likelihood-Ratio-Test. In beiden Fällen müssen jedoch die minimalen Block Matching Werte für alle Bewegungsmodelle bestimmt werden, bevor der Test durchgeführt werden kann. Da die Berechnungen in der Summe relativ viel Rechenzeit beanspruchen, verwenden wir eine effizientere Vorgehensweise, die auf einer hierarchischen Bewegungsbestimmung basiert.

In der Diskussion des vorangegangenen Abschnittes hatten wir bereits gesehen, dass $BM_N(\mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N)$ χ^2 -verteilt ist mit $|\mathbf{B}|$ Freiheitsgraden. Erlauben wir einen Prozentsatz α an falsch positiver Fehlklassifikationen, so erhalten wir den für einen Signifikanztest notwendigen Schwellenwert T_N , der auf BM_N angewendet wird, durch folgende Schritte: Die Nullhypothese H_0 nimmt an, dass das Modell für N transparente Bewegungen korrekt

ist. Den für den Signifikanztest notwendigen Schwellenwert T_N bestimmen wir durch

$$\text{Prob}(BM_N > T_N | H_0) = \alpha. \quad (8.23)$$

Die Nullhypothese H_0 wird abgelehnt, wenn $BM_N > T_N$ ist, und das Bewegungsmodell wird damit als nicht zutreffend betrachtet.

Entnehmen können wir den korrekte Schwellenwert T_N auch aus den Tabellen der χ^2 -Verteilung, die sich in vielen Formelsammlungen finden lassen.

8.3.2 Der hierarchische Algorithmus

In diesem Abschnitt werden wir zeigen, wie wir die obigen Betrachtungen in einen hierarchischen Algorithmus, mit dem sich einfache, transparente und verdeckte Bewegungen behandeln lassen, integrieren können. Zunächst werden dabei die Konfidenzmaße in steigender Komplexität der Bewegungsmodelle ausgewertet und anschließend die Bewegungsbestimmung für das adäquateste Bewegungsmodell durchgeführt. Beschreiben werden wir den hierarchischen Algorithmus nur bis zu einer Anzahl von zwei Bewegungen. Eine Erweiterung zu mehr als zwei Bewegungen läuft nach demselben Schema ab.

Im ersten Schritt bestimmt der Algorithmus mittels einer vollständigen Suche den Bewegungsvektor \mathbf{v} , für den BM_1 minimal ist. Anschließend testet er, ob der beobachtete Block Matching Wert durch das zugrunde liegende Rauschmodell erklärbar ist. Ist $BM_1(\mathbf{x}, \mathbf{v}) < T_1$, so wird der Bewegungsvektor \mathbf{v} der derzeitigen Position zugeordnet. Anderenfalls sucht er nach den Bewegungsvektoren \mathbf{v}_1 und \mathbf{v}_2 , die BM_2 minimieren, und testet, ob $BM_2(\mathbf{x}, \mathbf{v}_1, \mathbf{v}_2) < T_2$ ist. Werden beide Bewegungsmodelle abgelehnt, so wird die Position als ein Pixel, das zu der Verdeckungsgrenze gehört, markiert. In der zweiten Iteration wird die Bewegungsbestimmung für alle markierten Bildpunkte wiederholt. Dazu wird iterativ der Block solange vergrößert, bis genügend nicht markierte Bildpunkte für eine erfolgreiche Bewegungsbestimmung in dem Block vorhanden sind oder eine maximale Anzahl von Wiederholungen erreicht wurde. Die Bestimmung der Bewegungsvektoren basiert dabei nur auf den nicht markierten Bildpunkten, da die markierten Bildpunkte die Annahme von einfachen oder zwei additiven Bewegungen verletzen und daher würde die Minimierung von BM_1 und BM_2 nicht die korrekten Bewegungsvektoren liefern. Zu beachten ist jedoch, dass bei jeder Iteration die Schwellenwerte T_1 und T_2 neu an die Anzahl der nicht markierten Pixel in dem Block (Freiheitsgerade) angepasst werden müssen, um eine korrekte Entscheidungsfindung nach dem Rauschmodell zu gewährleisten. Dieser Zweiphasenansatz erlaubt uns, die zwei Bewegungen an der Verdeckungsgrenze zu bestimmen, indem es die von null verschiedenen Ausdrücke aus Gleichung (8.21) nicht mit in die Berechnung einfließen lässt. Algorithmus 2 fasst diesen Vorgang noch einmal in Form eines Pseudocodeprogrammes zusammen.

Algorithm 2 Hierarchischer Algorithmus

```
1: Berechne die Schwellenwerte  $T_1$  und  $T_2$ 
2: for all Pixel do
3:   Bestimme den minimalen Wert von  $BM_1$  und den korrespondierenden Bewegungs-
     vektor.
4:   if  $BM_1 < T_1$  then
5:     Wähle das einfache Bewegungsmodell
6:   else
7:     Bestimme den minimalen Wert von  $BM_2$  und die zwei Bewegungsvektoren
8:     if  $BM_2 < T_2$  then
9:       Wähle das transparente Bewegungsmodell
10:    else
11:      Markiere den Pixel.
12:    end if
13:  end if
14: end for
15: Erhöhe die Blockgröße und wiederhole die Zeilen 3 bis 14 für alle markierten Pixel.
     Ignoriere die markierten Pixel innerhalb des Blocks und berechne die neuen Schwel-
     lenwerte  $T_1$  und  $T_2$  in Abhängigkeit von der Anzahl nicht-markierter Pixel innerhalb
     des Blocks.
```

8.4 Bewegungsbestimmung mittels Markov-Zufallsfeldern

Der in den vorangegangenen Abschnitten vorgestellte Algorithmus basiert auf räumlichen und zeitlichen Beziehungen zwischen den Bildintensitäten, betrachtet aber keine räumlichen oder zeitlichen Beziehungen zwischen den Bewegungsvektoren. Dies kann unzureichend sein, denn Regionen, die zu bewegten Objekten korrespondieren, tendieren zu einer geschlossenen Form mit einem glatten Bewegungsfeld. Das Auftreten einzelner Punkte mit nicht glatten Bewegungsvektoren ist in natürlichen Bildern weniger wahrscheinlich. Regularisierung der Bewegungsvektorfelder wird häufig zur Bestimmung des optischen Flusses eingesetzt, wir hatten sie in Kapitel 4 für die Bestimmung überlagerter Bewegungen genutzt. Da wir hier die Bewegungsbestimmung im Sinne von statistischen Beobachtungen betrachten, erhöhen wir die Rauschrobustheit der Bewegungsbestimmung, indem wir eine stochastische Methode basierend auf den Markov-Zufallsfeldern mit den Block Matching Bedingungen kombinieren, ähnlich wie es in [2] zur Bewegungssegmentierung und in [4, 156] zur Bestimmung einfacher Bewegungen genutzt wurde. Diese Vorgehensweise hat drei wesentliche Vorteile: Zum Einen erlaubt sie, das wahrscheinlichste Bewegungsmodell unter dem Einfluss von Rauschen, im Sinne der Anzahl beobachteter Bewegungen in einem Punkt, auszuwählen. Zum Anderen wirkt sich die Einbeziehung benachbarter Punkte in die Berechnung positiv auf die Bewegungsbestimmung unter schwierigen Bedingungen aus. Zusätzlich wird simultan zur Berechnung eine Bewegungssegmentierung zur Verfügung gestellt, die für jeden Bildpunkt die lokale Anzahl der Bewegungen beschreibt. Im Folgenden werden wir einen Algorithmus für die Bestimmung von bis zu zwei Bewegungen herleiten. Eine Erweiterung zu einer beliebigen Anzahl von transparent

überlagerten Bewegungen ist prinzipiell möglich, erfordert aber Anpassungen im Bereich der räumlichen Glattheitsterme.

8.4.1 Bayes'sche Formulierung für zwei Bewegungen

Für jeden Bildpunkt \mathbf{x} und jeden Zeitpunkt k suchen wir nach den beiden Bewegungsvektoren $\mathbf{v}_1(\mathbf{x})$ und $\mathbf{v}_2(\mathbf{x})$ und einen Segmentierungswert $s(\mathbf{x}) \in \{0, 1\}$, der die Anzahl der beobachteten Bewegungen in diesem Bildpunkt beschreibt. Die Segmentierung hat dabei den Wert $s(\mathbf{x}) = 0$ in Bereichen mit einer Bewegung und $s(\mathbf{x}) = 1$ in Bereichen, in denen zwei transparente Bewegungen auftreten. Statische Szenen stellen wir durch das einfache Bewegungsmodell mit der Geschwindigkeit null dar. Unser Ziel ist es nun, das Tupel

$$\mathbf{u}_k(\mathbf{x}) = (\mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x}), s(\mathbf{x})) \quad (8.24)$$

mit Hilfe von $N + 1 = 3$ aufeinander folgenden Bildern für jeden Bildpunkt \mathbf{x} zu bestimmen. Entsprechend dem Maximum a posteriori Prinzip beabsichtigen wir, die wahrscheinlichste Segmentierung und Bewegungsvektoren für das gerade betrachtete Bild in Abhängigkeit von den gegebenen Beobachtungen f_k, f_{k-1}, f_{k-2} zu bestimmen. Erhalten können wir dieses Tupelfeld $\mathbf{u}_k = \{\mathbf{u}_k(\mathbf{x})\}$, indem wir mit

$$\mathbf{u}_k = \arg \max_{\mathbf{u}} p(\mathbf{u} | f_k, f_{k-1}, f_{k-2}), \quad (8.25)$$

nach dem Feld \mathbf{u} suchen, für das die posteriori Verteilungsdichtefunktion $p(\mathbf{u} | f_k, f_{k-1}, f_{k-2})$ unter den gegebenen Beobachtungen f_k, f_{k-1}, f_{k-2} maximal wird. Weiter erlaubt uns das Bayes-Theorem, obige Gleichung als

$$\mathbf{u}_k = \arg \max_{\mathbf{u}} p(f_k, f_{k-1}, f_{k-2} | \mathbf{u}) p(\mathbf{u}) \quad (8.26)$$

zu schreiben. Die a priori Verteilungsdichtefunktion $p(\mathbf{u})$ stellt sicher, dass das Ergebnis mit der erwarteten Glattheitsbedingung konsistent ist, während die bedingte Verteilungsdichtefunktion $p(f_k, f_{k-1}, f_{k-2} | \mathbf{u})$ die Beziehung zwischen den beobachteten Bildern und den unbekanntem Bewegungen beschreibt.

8.4.2 Das Beobachtungsmodell

Das Segmentierungsergebniss beschreibt die Anzahl der Bewegungen für jeden Bildpunkt. In Abhängigkeit von dieser Segmentierung müssen wir das korrespondierende Bewegungsmodell für die Spezifikation der Likelihood-Funktion $p(f_k, f_{k-1}, f_{k-2} | \mathbf{u})$ heranziehen. Aus Abschnitt 8.2.1 wissen wir bereits, dass die bewegungskompensierte Differenz der Block Matching Bedingung (8.5) $\mathcal{N}(0, 2^N \sigma^2)$ -verteilt ist und demnach jeder Differenzwert $e(f, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N)$ der Normalverteilungsdichte

$$\mathcal{N}(0, 2^N \sigma^2) = \frac{1}{\sqrt{2^{N+1} \sigma^2 \pi}} e^{-e(f, \mathbf{x}, \mathbf{v}_1, \dots, \mathbf{v}_N)^2 / (2^{N+1} \sigma^2)} \quad (8.27)$$

unter der Voraussetzung genügt, dass die gewählten Bewegungsvektoren korrekt sind. Teilt uns die Segmentierung s mit, dass an dem betrachteten Punkt nur eine Bewegung vorhanden ist ($s = 0$), so verwenden wir den Block Matcher BM_1 für eine Bewegung

zur Aufstellung des Likelihoodterms. Anderenfalls gehen wir zum Modell für zwei Bewegungen (BM_2) über. Mit Hilfe der vorangegangenen Betrachtungsweisen spezifizieren wir den Likelihoodterm, der das Umschalten zwischen den beiden Bewegungsmodellen in Abhängigkeit von der Segmentierung erlaubt, durch die Verbundverteilungsdichte

$$p(f_k, f_{k-1}, f_{k-2} | \mathbf{u}) = \frac{1}{Z_L} \prod_{\mathbf{x}} [(1 - s(\mathbf{x})) (4\pi\sigma^2)^{-|\mathbf{B}|/2} e^{-BM_1(\mathbf{x}, \mathbf{v}_1(\mathbf{x}))/2} + s(\mathbf{x}) (8\pi\sigma^2)^{-|\mathbf{B}|/2} e^{-BM_2(\mathbf{x}, \mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x}))/2}], \quad (8.28)$$

wobei die Normalisierungskonstante Z_L sicherstellen soll, dass $\sum_{f_k, f_{k-1}, f_{k-2}} p(f_k, f_{k-1}, f_{k-2} | \mathbf{u}) = 1$ ist. Im Allgemeinen werden wir diese Konstante jedoch nicht bestimmen können, da uns die Wahrscheinlichkeiten aller möglichen Bilder nicht bekannt sind. Problematisch ist dies jedoch nicht, da es sich hier um eine Konstante handelt, die durch eine spätere Logarithmierung in eine additive Konstante, die keinen Einfluss auf die weitere Berechnung hat, überführt wird und somit vernachlässigt werden kann. Für uns ist also die Aussage, dass der angegebene Ausdruck proportional zu der Likelihoodfunktion ist, völlig ausreichend.

Das Umschalten zwischen beiden Modellen haben wir durch die Funktionen $s(\mathbf{x})$ und $(1 - s(\mathbf{x}))$ realisiert. Tritt an dem betrachteten Punkt nur eine Bewegung auf, so wird durch $(1 - s(\mathbf{x})) = 1$ das einfache Bewegungsmodell ein- und durch $s(\mathbf{x}) = 0$ das transparente Bewegungsmodell ausgeschaltet. Bei zwei Bewegungen wird dieses Verhalten umgekehrt, d.h. $(1 - s(\mathbf{x})) = 0$ und $s(\mathbf{x}) = 1$.

Für eine Erweiterung auf mehr als zwei Bewegungen müssen die Schalterfunktionen $s(\mathbf{x})$ und $(1 - s(\mathbf{x}))$ neu angepasst werden, um sicherzustellen, dass für jeden Segmentierungswert immer nur ein Bewegungsmodell genutzt wird. Im nächsten Schritt geben wir die Verbundverteilungsdichte $p(\mathbf{u})$ an, um das Beobachtungsmodell zu vervollständigen.

8.4.3 Räumliche Glattheit

Die Spezifikation der Verbundverteilungsdichte $p(\mathbf{u})$ aller Tupel $\mathbf{u}(\mathbf{x})$ sollte sicherstellen, dass das zu erwartende Tupelfeld wahrscheinlicher als alle anderen ist. Die Verbundverteilungsdichte $p(\mathbf{u})$ lässt sich mit Hilfe der Markov-Zufallsfelder lokal beschreiben, wenn sie der Gibbs-Verteilungsdichtefunktion

$$p(\mathbf{u}) = \frac{1}{Z} e^{-\lambda E(\mathbf{u})} \quad (8.29)$$

genügt [71, 24], wobei Z eine Normalisierungskonstante ist und der Parameter λ den Einfluss der späteren Glättung kontrolliert. Die Energie $E(\mathbf{u})$ sollte für glatte Bewegungsvektor- und Segmentierungsfelder klein sein, damit die Wahrscheinlichkeit für diese Felder möglichst groß wird. Weiter ermöglicht die Markov-Annahme mit

$$E(\mathbf{u}) = \sum_{\mathbf{x}} E_L(\mathbf{x}, \mathbf{u}) \quad (8.30)$$

die Aufteilung der Energie $E(\mathbf{u})$ in eine Summe lokaler Energien $E_L(\mathbf{x}, \mathbf{u})$. Die lokale Energie $E_L(\mathbf{x}, \mathbf{u})$ hängt sowohl von den Bewegungsvektoren und den Segmentierungswerten des Bildpunktes \mathbf{x} als auch von den Punkten in seiner Nachbarschaft $N_{\mathbf{x}}$ ab. In

unserem Fall besteht die Nachbarschaft $N_{\mathbf{x}}$ aus den acht Punkten, die den Punkt \mathbf{x} umgeben. Um sowohl glatte Bewegungs- als auch Segmentierungsfelder beschreiben zu können, teilen wir die lokalen Energien in zwei Anteile auf:

$$E_L(\mathbf{x}, \mathbf{u}) = E_{L_s}(\mathbf{x}, \mathbf{u}) + E_{L_v}(\mathbf{x}, \mathbf{u}). \quad (8.31)$$

Der Ausdruck E_{L_s} misst dabei die Glattheit des Segmentierungsfeldes und E_{L_v} die Glattheit der Bewegungsvektorfelder. Damit wir lokal möglichst glatte Bewegungsvektorfelder erhalten, bestrafen wir Unterschiede in den Bewegungsvektoren zweier benachbarter Positionen \mathbf{x} und \mathbf{y} . Treten an beiden Positionen zwei Bewegungen auf, so bezeichnen wir mit demselben Index jeweils die Vektoren, die einander am ähnlichsten sind. Demnach stimmt $\mathbf{v}_1(\mathbf{x})$ besser mit $\mathbf{v}_1(\mathbf{y})$ überein als mit $\mathbf{v}_2(\mathbf{y})$. Problematisch wird dieses Vorgehen zum Beispiel aufgrund von Objektgrenzen, wenn die Anzahl der Bewegungsvektoren innerhalb der Nachbarschaft variiert. Da es sich hier aber um transparente Bewegungen handelt, wissen wir jedoch, dass der Bewegungsvektor eines Punktes mit nur einer Bewegung mit einem der beiden Bewegungsvektoren eines Punktes mit zwei Bewegungen übereinstimmen muss. Folglich können wir in solchen Fällen die Glattheitsbedingung nur auf diese beiden Punkte anwenden. Im Folgenden bezeichnen wir den Bewegungsvektor von beiden, der am besten mit dem einfachen Bewegungsvektor übereinstimmt, mit \mathbf{v}_1 . Mit dieser Konvention können wir nun die Glattheitsbedingung als

$$E_{L_v}(\mathbf{x}, \mathbf{u}) = \sum_{\mathbf{y} \in N_{\mathbf{x}}} (\|\mathbf{v}_1(\mathbf{x}) - \mathbf{v}_1(\mathbf{y})\|^2 + s(\mathbf{x})s(\mathbf{y})\|\mathbf{v}_2(\mathbf{x}) - \mathbf{v}_2(\mathbf{y})\|^2). \quad (8.32)$$

schreiben, wobei wir die Ähnlichkeit zweier Vektoren durch die quadrierte Norm des Differenzvektors messen. Die Unterscheidung, ob jeweils nur ein Vektor oder zwei Vektoren miteinander verglichen werden sollen, wird durch den Ausdruck $s(\mathbf{x})s(\mathbf{y})$ gesteuert.

Zur Spezifikation der Funktion E_{L_s} nutzen wir dieselbe Vorgehensweise, wie sie in [2] zur Bewegungssegmentierung verwendet wurde. Dazu werden alle Punkte in der Nachbarschaft $N_{\mathbf{x}}$ gezählt, die den gleichen Segmentierungswert besitzen wie der Punkt \mathbf{x} . Anschließend ziehen wir diese Werte von der maximal erreichbaren Anzahl gleicher Segmentierungswerte, die in unserem Fall acht ist, ab. Damit definieren wir die lokale Segmentierungsenergie durch

$$E_{L_s}(\mathbf{x}, \mathbf{u}) = 8 - w_{N_{\mathbf{x}}}, \quad (8.33)$$

wobei der Ausdruck $w_{N_{\mathbf{x}}}$ die Anzahl der Punkte innerhalb der Nachbarschaft $N_{\mathbf{x}}$ beschreibt, die den gleichen Segmentierungswert besitzen wie der Punkt \mathbf{x} . Die Energie ist minimal, wenn alle Punkte innerhalb der Nachbarschaft vom selben Bewegungstyp wie der betrachtete Punkt sind. Folglich steigt mit steigender Anzahl $w_{N_{\mathbf{x}}}$ auch die Wahrscheinlichkeit des Punktes \mathbf{x} , von diesem Bewegungstyp zu sein.

8.4.4 Optimierungsalgorithmus

Die gesuchte Lösung entspricht dem Argument, dass Gleichung (8.26) maximiert. Wie wir bereits gesehen haben, setzt sich Gleichung (8.26) aus dem Produkt der beiden Gleichungen (8.28) und (8.29) mit den zugrunde liegenden Energien (8.32) und (8.33) zusammen.

Durch Anwenden des negativen Logarithmus wird die Maximierung von Gleichung (8.26) äquivalent zu der Minimierung von

$$\mathbf{u}_k = \arg \min_{\mathbf{u}} \left[-\log(p(f_k, f_{k-1}, f_{k-2}|\mathbf{u})) - \log(p(\mathbf{u})) \right]. \quad (8.34)$$

Wenden wir den negativen Logarithmus auf $p(f_k, f_{k-1}, f_{k-2}|\mathbf{u})$ an, so erhalten wir

$$\begin{aligned} -\log(p(f_k, f_{k-1}, f_{k-2}|\mathbf{u})) &= \log(Z_L) \\ &+ \sum_{\mathbf{x}} [(1 - s(\mathbf{x}))(BM_1(\mathbf{x}, \mathbf{v}_1(\mathbf{x})) - \log((4\pi\sigma^2)^{-|\mathbf{B}|/2}) \\ &+ s(\mathbf{x})(BM_2(\mathbf{x}, \mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x})) - \log((8\pi\sigma^2)^{-|\mathbf{B}|/2})] \end{aligned} \quad (8.35)$$

und nach Anwendung auf $p(\mathbf{u})$,

$$-\log(p(\mathbf{u})) = \lambda E(\mathbf{u}) + \log(Z). \quad (8.36)$$

Durch Einsetzen der Gleichungen (8.35) und (8.36) in Gleichung (8.34) können wir das Optimierungskriterium als

$$\begin{aligned} C(f_2, f_1, f_0|\mathbf{u}) &= \sum_{\mathbf{x}} [(1 - s(\mathbf{x}))(BM_1(\mathbf{x}, \mathbf{v}_1(\mathbf{x})) - \log((4\pi\sigma^2)^{-|\mathbf{B}|/2}) \\ &+ s(\mathbf{x})(BM_2(\mathbf{x}, \mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x})) - \log((8\pi\sigma^2)^{-|\mathbf{B}|/2})] \\ &+ \lambda E(\mathbf{u}) + \log(Z) + \log(Z_L) \end{aligned} \quad (8.37)$$

schreiben. Im nächsten Schritt möchten wir alle Ausdrücke, die keinen Einfluss auf die Minimierung haben, aus dem Optimierungskriterium entfernen. Zu diesen Ausdrücken zählen die beiden additiven Konstanten $\log(Z)$ und $\log(Z_L)$. Weitere streichbare Ausdrücke erhalten wir durch eine genauere Betrachtung von $\log((4\pi\sigma^2)^{-|\mathbf{B}|/2})$ und $\log((8\pi\sigma^2)^{-|\mathbf{B}|/2})$. Eine erneute Anwendung der Logarithmengesetze auf die beiden Ausdrücke ergibt

$$\begin{aligned} \log((4\pi\sigma^2)^{-|\mathbf{B}|/2}) &= -\frac{|\mathbf{B}|}{2} (4 \log(\sqrt{2}) + \log(\pi) + \log(\sigma^2)), \\ \log((8\pi\sigma^2)^{-|\mathbf{B}|/2}) &= -\frac{|\mathbf{B}|}{2} (6 \log(\sqrt{2}) + \log(\pi) + \log(\sigma^2)). \end{aligned} \quad (8.38)$$

Da in beiden Gleichungen die konstanten Terme $\log(\pi)$ und $\log(\sigma^2)$ auftreten haben sie keinen Einfluss auf die Minimierung von (8.37) und können daher vernachlässigt werden. Der Ausdruck $\frac{|\mathbf{B}|}{2} 4 \log(\sqrt{2})$ geht nur mit in die Berechnung ein, wenn $(1 - s(\mathbf{x})) = 1$ ist. Umgekehrt wird $\frac{|\mathbf{B}|}{2} 6 \log(\sqrt{2})$ nur dann verwendet, wenn $s(\mathbf{x}) = 1$ ist. Unser Ziel ist nun, diese beiden Ausdrücke zu einem einzigen additiven Term zusammenzufassen. In Abhängigkeit des Segmentierungswertes $s(\mathbf{x})$ wird entweder der eine oder der andere Term verwendet, d.h.

$$\begin{aligned} &\sum_{\mathbf{x}} [(1 - s(\mathbf{x})) \frac{|\mathbf{B}|}{2} 4 \log(\sqrt{2}) + \frac{|\mathbf{B}|}{2} 6 \log(\sqrt{2}) s(\mathbf{x})] = \\ &\sum_{\mathbf{x}} \left[\frac{|\mathbf{B}|}{2} 4 \log(\sqrt{2}) + \frac{|\mathbf{B}|}{2} 2 \log(\sqrt{2}) s(\mathbf{x}) \right] = \\ &\sum_{\mathbf{x}} \frac{|\mathbf{B}|}{2} 4 \log(\sqrt{2}) + \sum_{\mathbf{x}} |\mathbf{B}| \log(\sqrt{2}) s(\mathbf{x}). \end{aligned} \quad (8.39)$$

Die erste Summe obiger Gleichung hängt nicht von der Segmentierung ab und stellt damit eine additive Konstante dar, die keinen Einfluss auf die Optimierung hat. Die zweite Summe hängt jedoch von der Segmentierung ab und kann daher nicht vernachlässigt werden. Diese Betrachtungen erlauben uns die Vereinfachung des Optimierungskriteriums (8.37) zu

$$C(f_2, f_1, f_0|\mathbf{u}) = \sum_{\mathbf{x}} \left[(1 - s(\mathbf{x})) BM_1(\mathbf{x}, \mathbf{v}_1(\mathbf{x})) + s(\mathbf{x}) BM_2(\mathbf{x}, \mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x})) \right] + \lambda E(\mathbf{u}) + \log(\sqrt{2})|\mathbf{B}||s| \quad (8.40)$$

mit $|s| = \sum_{\mathbf{x}} s(\mathbf{x})$. Optimieren lässt sich diese Gleichung zum Beispiel mit dem deterministischen Relaxationsverfahren, das in [23] vorgestellt wurde. Es garantiert zwar nicht, das globale Minimum zu finden, läuft dafür aber verhältnismäßig schnell. Geman und Geman [71] stellten ein Optimierungsverfahren vor, mit dem das Finden des globalen Optimums garantiert ist. Da es jedoch viel Rechenzeit beansprucht, ist es für praktische Anwendungen nur bedingt einsetzbar. Zur Berechnung der Ergebnisse verwenden wir daher Besag's ICM-Verfahren [23].

Für die Berechnung der Bewegungsvektoren nutzen wir die gefundenen Ergebnisse des vorangegangenen Bildes zur Initialisierung des Optimierungsalgorithmus. Im Falle schneller Bewegungen kann diese Initialisierung allerdings unzureichend sein. Solange sich die Bewegungen von einem Bild zum nächsten nicht abrupt ändern, kommt eine solche Vorhersage den wahren Bewegungen sehr nahe und stellt eine Form der zeitlichen Regularisierung dar. In einigen Anwendungen kann jedoch eine explizite zeitliche Glättung zur einer Verbesserung der Ergebnisse führen. Im Folgenden möchten wir kurz die dafür notwendigen Modifikationen des Algorithmus beschreiben.

8.4.5 Zeitliche Glattheit

Optional können wir die Bewegungsbestimmung um eine zeitliche Regularisierung der Bewegungsfelder erweitern. Für die Bestimmung des Tupelfeldes \mathbf{u}_k liegen die Ergebnisse \mathbf{u}_{k-1} des vorangegangenen Bildes bereits vor und können daher zum Entwurf einer zeitlichen Glattheitsbedingung genutzt werden. Um den Entwurf einer solchen Bedingung zu vereinfachen, nehmen wir weiter an, dass die aufeinander folgenden Bilder f_k, f_{k-1}, f_{k-2} und \mathbf{u}_{k-1} statistisch bedingt unabhängig voneinander sind. Diese Annahmen erlauben uns, das Beobachtungsmodell $p(f_k, f_{k-1}, f_{k-2}, \mathbf{u}_{k-1}|\mathbf{u})$ als

$$p(f_k, f_{k-1}, f_{k-2}, \mathbf{u}_{k-1}|\mathbf{u}) = p(f_k, f_{k-1}, f_{k-2}|\mathbf{u}) p(\mathbf{u}_{k-1}|\mathbf{u}). \quad (8.41)$$

zu schreiben und wir passen die MAP-Bestimmung durch

$$\mathbf{u}_k = \arg \max_{\mathbf{u}} p(f_k, f_{k-1}, f_{k-2}|\mathbf{u}) p(\mathbf{u}_{k-1}|\mathbf{u}) p(\mathbf{u}). \quad (8.42)$$

den neuen Gegebenheiten an. Im Vergleich zu dem vorherigen Optimierungskriterium ist (8.42) um eine weitere Komponente $p(\mathbf{u}_{k-1}|\mathbf{u})$, die die Beziehung zwischen dem vorangegangenen Tupel \mathbf{u}_{k-1} und dem aktuellen beschreibt, erweitert worden.

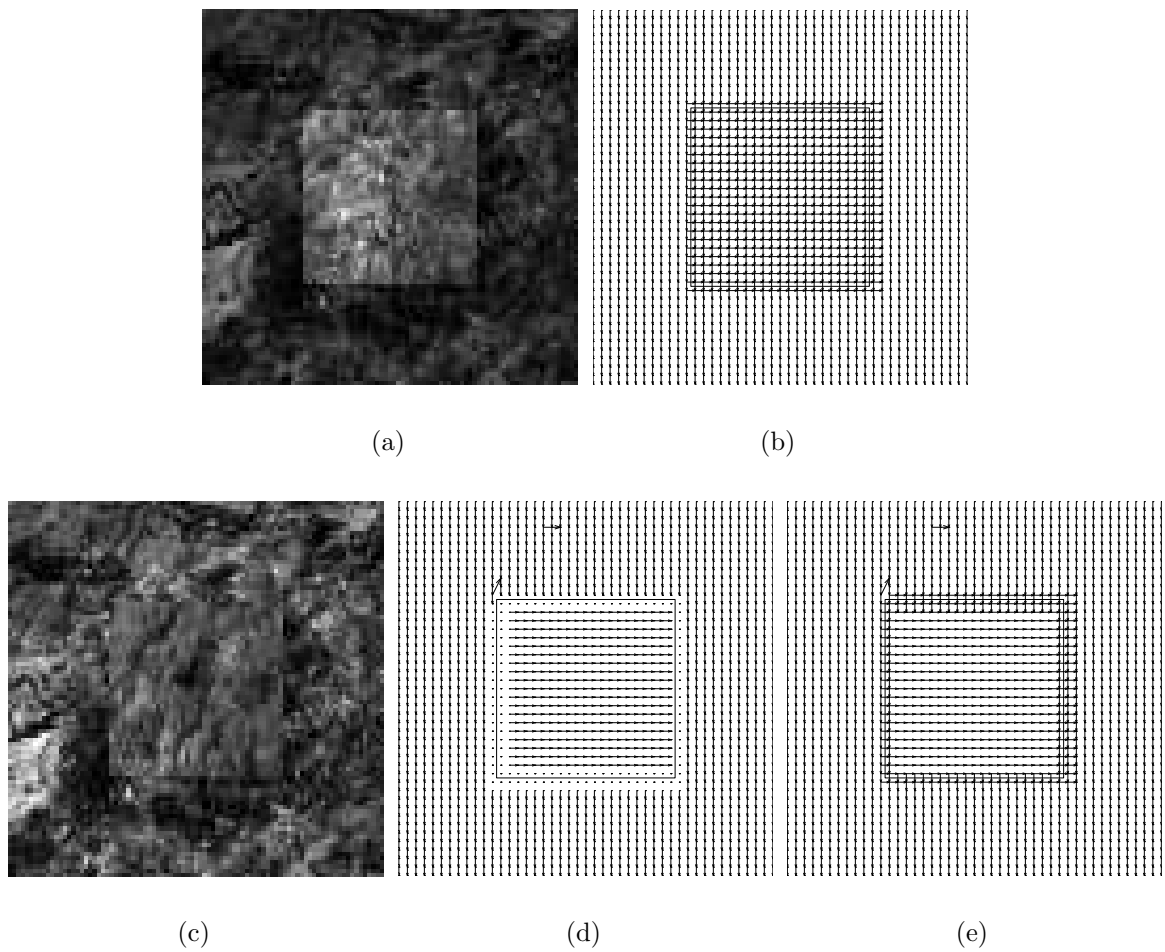


Abbildung 8.2: Beispiele für transparente und verdeckende Bewegungen. Bild (a) zeigt ein Bild einer transparenten Testsequenz und (b) die ermittelten Bewegungsvektoren. In (c) ist ein Bild einer Testsequenz mit verdeckenden Bewegungen, (d) und (e) die ermittelten Bewegungsvektoren nach der ersten bzw zweiten Phase dargestellt.

Für die Spezifikation von $p(\mathbf{u}_{k-1}|\mathbf{u})$ nutzen wir wieder ein Gauß-Modell und machen zwei weitere Vereinfachungen: Erstens, jeder Vektor $\mathbf{v}^k = \mathbf{v}^k(\mathbf{x})$ hängt nur von seinem Vorgänger $\mathbf{v}^{k-1}(\mathbf{x} + \mathbf{v}^k)$ entlang der Bewegungstrajektorie ab. Zweitens setzen wir voraus, dass die Bewegungsvektoren eines Bildes statistisch unabhängig voneinander sind. Diese Vereinfachungen erlauben uns den Likelihood-Term als

$$p(\mathbf{u}_{k-1}|\mathbf{u}) = \prod_{\mathbf{x}} p(\mathbf{v}_1^{k-1}(\mathbf{x} + \mathbf{v}_1^k)|\mathbf{v}_1^k) p(\mathbf{v}_2^{k-1}(\mathbf{x} + \mathbf{v}_2^k)|\mathbf{v}_2^k), \quad (8.43)$$

zu schreiben, wobei die bedingten Verteilungsdichten

$$p(\mathbf{v}_i^{k-1}(\mathbf{x} + \mathbf{v}_i^k)|\mathbf{v}_i^k) = \frac{1}{Z_T} \exp(-\lambda_T \|(\mathbf{v}_i^{k-1}(\mathbf{x} + \mathbf{v}_i^k) - \mathbf{v}_i^k)\|^2) \quad (8.44)$$

für $i = 1, 2$ gaußverteilt und die Faktoren Z_T und λ_T Normalisierungs- bzw. Regularisierungskonstanten sind. Bei Bedarf können diese Ausdrücke in das Optimierungskriteri-

um (8.40) integriert werden.

8.5 Ergebnisse

Abbildung 8.2 zeigt je ein Beispiel für transparente und verdeckende Bewegungen. In (a) ist das mittlere Bild einer Bildsequenz abgebildet, in der Bereiche mit einer und zwei Bewegungen vorkommen. Der Bereich mit zwei transparenten Bewegungen ist gut durch den helleren kastenförmigen Bereich im Bild identifizierbar. Eine Schicht bewegt sich mit einer Geschwindigkeit von einem Bildpunkt pro Bild nach rechts und die andere mit einer Geschwindigkeit von einem Bildpunkt pro Bild nach unten. Die ermittelten Bewegungsvektoren für jeden zweiten Bildpunkt sind in (b) dargestellt. Das Rechteck in (b) markiert die Umrandung des Bereichs mit zwei Bewegungen. In beiden Bereichen wurden die Bewegungen korrekt erkannt. In diesem Beispiel verwendeten wir eine Blockgröße von 5×5 Bildpunkten. Bild (c) zeigt das mittlere Bild einer Verdeckungssequenz und die Bilder (d) und (e) die Ergebnisse nach der ersten bzw. zweiten Phase des Algorithmus für jeden zweiten Bildpunkt. Die Bewegungen in beiden Regionen sind abgesehen von einigen wenigen Ausreißern gut bestimmt. Nach der zweiten Phase sind in einem Bereich um die Verdeckungsgrenze herum, wo in der ersten Phase keine Bewegungen ermittelt werden konnten, zwei Bewegungen erkannt worden. Für die erste und zweite Phase verwendeten wir Fenstergrößen von 5×5 und 9×9 Bildpunkten. Zu beiden Beispielen wurde ein gaußverteiltes Rauschen addiert, so dass sich ein Signalrauschverhältnis von 35dB ergab. Für den Signifikanztest setzten wir $\alpha = 0.001$. Die Bewegungen sind hier und in den folgenden Beispielen mittels einer Vollständigensuche über den Bereich von -5 bis 5 Bildpunkte in ganzzahligen Schritten pro Bewegungsvektorkomponente bestimmt worden.

Abbildung 8.3 zeigt ein reales Szenario mit verdeckenden Bewegungen. Ein Spielzeugzug, der eine Textur transportiert, bewegt sich nach rechts vor einer Textur, die sich nach links bewegt. Beide Bewegung sind ungefähr ein Bildpunkt pro Bild schnell. Das Rechteck in (a) markiert den Bereich, für den wir die ermittelten Bewegungsvektoren angeben. Bild (b) zeigt diesen Ausschnitt sowie die Ränder des Bereiches, in dem nach der ersten Phase aufgrund der Verdeckungsgrenze keine Bewegungen berechnet werden konnten. Die Größe dieser Region entspricht in etwa der Blockgröße von 7×7 Bildpunkten. Die ermittelten Bewegungsvektoren nach der ersten Phase sind in (c) abgebildet. Die Vektoren des Endergebnisses (nach der zweiten Phase) sind zur besseren Visualisierung in zwei separate Bilder aufgeteilt. Bild (d) zeigt die linksgerichteten und (e) die rechtsgerichteten Vektoren. Beide Bewegungen in der Region um die Verdeckungsgrenze sind gut bestimmt worden. Für die zweite Phase verwendeten wir eine Blockgröße von 18×18 Bildpunkten in der ersten Phase setzten wir $\alpha = 0.001$ für eine Bewegung und $\alpha = 0.05$ für zwei Bewegungen, da wir hier keine transparenten Bewegungen erwarteten. In der zweiten Phase setzten wir für beide Fälle $\alpha = 0.001$.

Abbildung (8.4) zeigt ein Ergebnis für den Markov-Zufallsfeldansatz. Als Testsequenz verwendeten wir die gleiche Sequenz wie bei dem vorangegangenen Beispiel für den hierarchischen Algorithmus nur mit dem Unterschied, dass die Sequenz nun ein Signalrauschverhältnis von 17dB aufweist. Initialisiert wurde das Verfahren mit einer Bewegung und

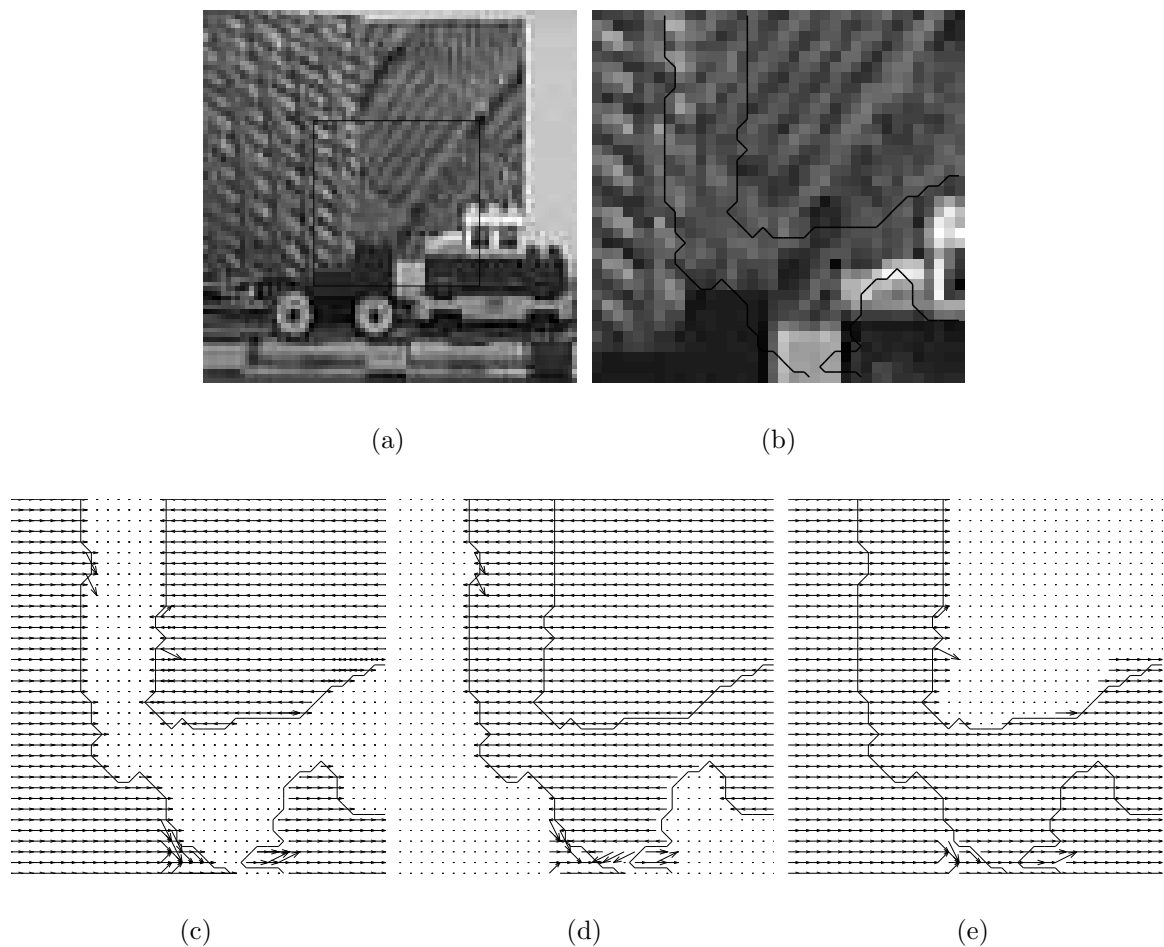


Abbildung 8.3: Ergebnis für ein reales Video mit verdeckenden Bewegungen. Bild (a) zeigt ein Bild der Sequenz, (b) den Bereich für den die Bewegungsvektoren gezeigt werden, (c) die ermittelten Bewegungsvektoren nach der ersten Phase, (d) und (e) die ermittelten Vektoren nach der zweiten Phase, eingeteilt in zwei Bilder zur besseren Visualisierung.

der Geschwindigkeit null in allen Punkten. Bild (a) zeigt ein Bild der Testsequenz, (b) die ermittelten Bewegungsvektoren des MZF-Verfahrens für jeden zweiten Bildpunkt nach drei Iterationen und (c) ermittelte zum Vergleich die Bewegungsvektoren mit dem hierarchischen Verfahren. Für beide Verfahren verwendeten wir ein Blockgröße von 3×3 Bildpunkten. Für den MZF-Ansatz benutzten wir den Parameter $\lambda = 1$ sowie drei Iterationen und für das hierarchische Verfahren den Parameter $\alpha = 0.001$. Bei dem MZF-Verfahren werden deutlich weniger Ausreißer beobachtet als bei dem hierarchischen. In einigen kleineren Regionen blieben die mit dem MRF-Verfahren ermittelten Bewegungen wie initialisiert null. Die Ergebnisse acht Bilder später sind in der zweiten Zeile von Abbildung 8.4 dargestellt. Erneut beobachten wir bei dem MZF-Verfahren weniger Ausreißer und die Regionen mit der Geschwindigkeit null sind verschwunden.

Abbildung 8.5 verdeutlicht, dass das hierarchische Block-Matching-Verfahren auch unter realen Bedingungen funktioniert. Im Unterschied zu den vorangegangenen Beispielen

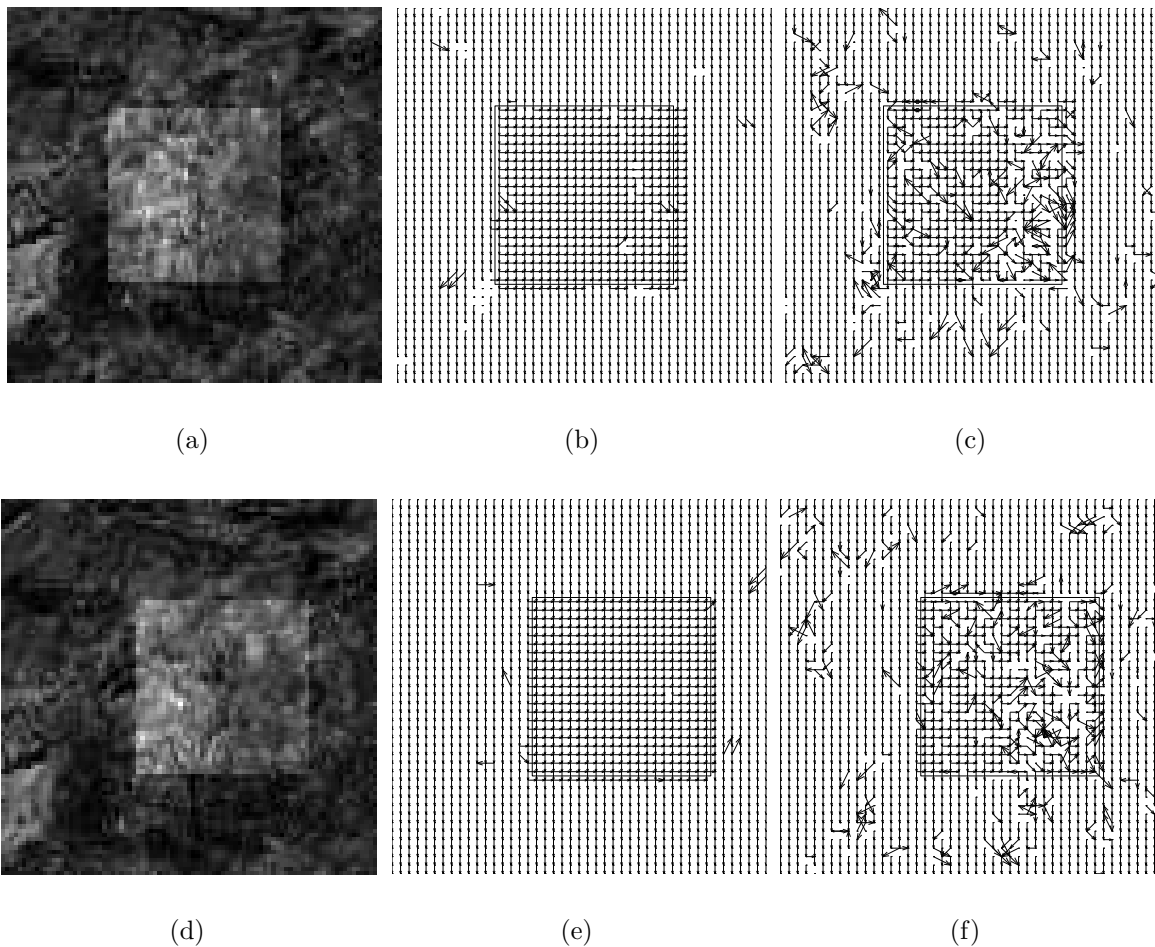


Abbildung 8.4: Ergebnis für transparente Bewegungen, ermittelt mit dem Markov-Zufallsfeldverfahren (MZF): In (a) ist ein Bild einer Sequenz mit transparenten Bewegungen und einem Signalrauschverhältnis von 17 dB, in (b) die ermittelten Bewegungsvektoren mit dem MZF-Verfahren und (c) die Ergebnisse des hierarchischen Verfahrens zum Vergleich dargestellt. Die zweite Zeile zeigt die ermittelten Bewegungen acht Bilder später.

mit der Monalisasequenz, haben wir die Sequenz hier noch einmal unterabgetastet, um den nötigen Rechenaufwand zu reduzieren. Durch die Unterabtastung werden die Bewegungen langsamer. Um die Bewegungsbestimmung mit dem Block-Matching-Verfahren nicht komplett im Subpixelbereich durchzuführen, haben wir durch Weglassen von je zwei aufeinander folgenden Bildern die Bewegungen beschleunigt. Damit ist eine der Geschwindigkeiten ca. ein Pixel pro Bild schnell, während die andere etwas schneller ist. Abbildung 8.5(a) zeigt ein Bild dieser Sequenz. In Grafik (b) sind die ermittelten Bereiche mit einer und zwei Bewegungen in Form einer Farbkodierung dargestellt. Zur Berechnung der Bewegungen verwendeten wir in diesem Beispiel eine, in der Praxis häufig verwendete Blockgröße, von 16×16 Bildpunkten. Aufgrund dieser Blockgröße ragt der Bereich mit zwei Bewegungen (weiß) deutlich in den Bereich mit nur einer Bewegung (schwarz) hinein. Die Berechnung der Bewegungsvektoren erfolgte hier im Unterschied zu den vorangegangenen Beispielen mit einer Genauigkeit von 0.25 Bildpunktschritten über

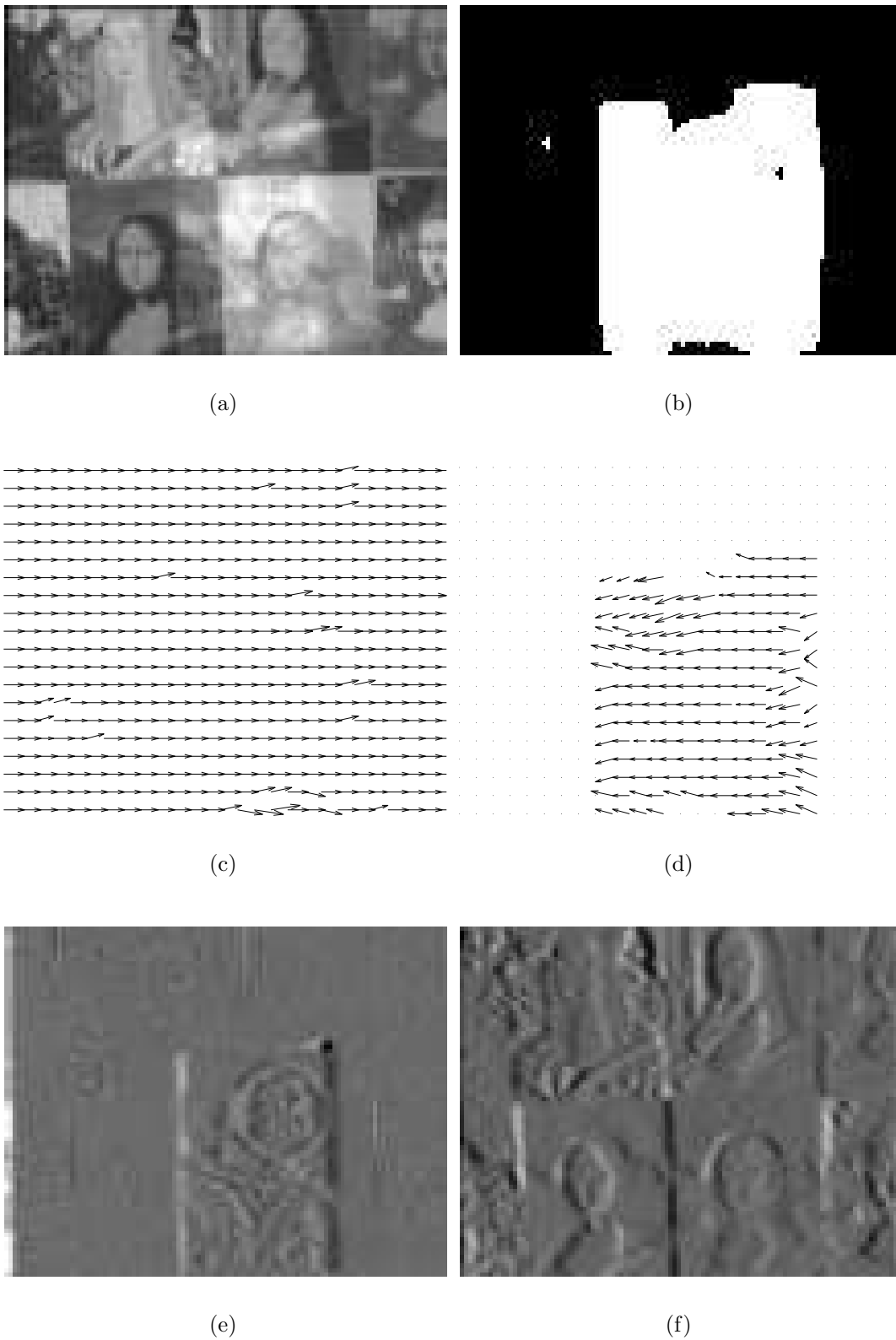


Abbildung 8.5: Ergebnis des hierarchischen Algorithmus für die Monalisasequenz. (Siehe Text)

dem Intervall von -2 bis 2 für jede Bewegungsvektorkomponente. Für den Signifikanztest setzten wir $\alpha = 0.001$ bei einer Varianz von $\sigma^2 = 4$. Die Piktogramme (c) und (d) zeigen die ermittelten Bewegungsvektoren für jeden vierten Bildpunkt. Als mittlerer Geschwindigkeitsvektor für die Monaliskaschicht ergab sich der Vektor $\mathbf{v} = (1.24, 0.001)^T$ und für die Müslipackung der Vektor $\mathbf{u} = (1.06, 0.02)^T$. Rechnen wir diese Geschwindigkeiten auf die Originalmonalissesequenz um, so entsprechen sie den Vektoren $\mathbf{v} = (0.83, 0.00)^T$ und $\mathbf{u} = (-0.71, 0.01)^T$. Die umgerechneten Bewegungsvektoren weichen damit etwas von den Ergebnissen der anderen Bewegungsbestimmungsverfahren ab (vergleiche z.B. mit den Ergebnissen aus dem vorangegangenen Kapitel), was im Wesentlichen auf die Diskretisierungsschrittweite der Bewegungsbestimmung und die Unterabtastung zurückzuführen ist. Die Bilder (e) und (f) zeigen die Auswertung der einfachen Block-Matching-Bedingung für jeden Bildpunkt unter Verwendung der gemittelten Bewegungsvektoren. Alle Anteile, die sich mit dieser Bewegung bewegen, werden aus der Sequenz entfernt. Da in beiden Bildern nur Anteile jeweils einer der beiden Schichten sichtbar sind, konnten die Anteile der jeweils anderen Schicht entfernt werden und die berechneten Bewegungsvektoren stimmen im Mittel gut mit den Bewegungen in der Sequenz überein.

Das Block-Matchingresultat des hierarchischen Verfahrens haben wir in einem weiteren Verarbeitungsschritt zur Initialisierung für das Block-Matchingverfahren basierend auf einem Markov-Zufallsfeld genutzt. Die erste Spalte von Abbildung 8.6 zeigt das Ergebnis für das gleiche Bild nach drei Iterationen des MZF-Verfahrens und die zweite Spalte das Ergebnis sechs Bilder später. Im Vergleich zu dem vorangegangenen Beispiel verkleinerte sich der Bereich mit zwei Bewegungen schon nach drei Iterationen deutlich und näherte sich damit der Größe der Müslibox an. Im Gegenzug dazu entstanden einige Löcher, in denen nur eine Bewegung beobachtet wird. Die verwendete Blockgröße betrug hier nur 3×3 Bildpunkte und daher ist es wenig verwunderlich, dass die neu entstandenen Löcher in Bereichen auftreten, in denen entweder die Müslipackung oder die Monaliskaschicht eine homogene Gauwertstruktur aufweist. Die mittleren Bewegungsvektoren der Monalisa- und Müslischicht lagen bei $\mathbf{v} = (1.2, 0.01)$ bzw $\mathbf{u} = (-1.1, 0.01)$ und weichen dabei nur wenig von dem Ergebnis des hierarchischen Verfahrens ab. Die Suchschrittweitengenauigkeit lag auch hier bei 0.25 Bildpunkten. Die zweite Spalte zeigt das Ergebnis nach sechs weiteren Bildern. Jedes Bild wurde mit dem Ergebnis des Vorgängerbildes initialisiert und anschließend drei Iterationen durchgeführt. Wie das Ergebnis zeigt, stimmen die Bewegungen auch nach sechs weiteren Bildern gut mit den Bewegungen in der Sequenz überein. Der ermittelte Bereich mit zwei Bewegungen deckt sich gut mit der Position der Müslipackung und es konnten sogar einige der Löcher wieder geschlossen werden. Lediglich am unteren Rand entsteht ein kleiner nachziehender Schleier. Die Ursache für diesen Schleier liegt in der Implementierung. Die Bewegungsbestimmung wurde erst ab einem Mindestabstand von 5 Bildpunkten vom Rand durchgeführt, die initialisierten Werte am Rand blieben daher unverändert. Die mittleren Bewegungsvektoren beider Schichten lagen in diesem Bild bei $\mathbf{v} = (1.21, 0.01)$ bzw $\mathbf{u} = (-1.13, 0.01)$. Für den Regularisierungsparameter λ wählten wir den Wert fünf.



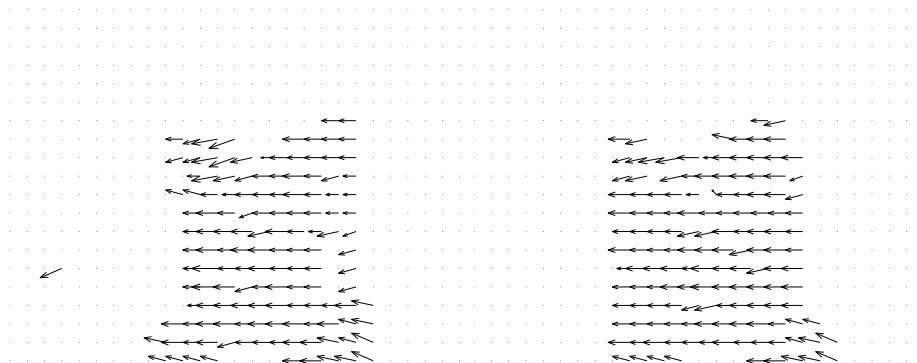
(a)

(b)



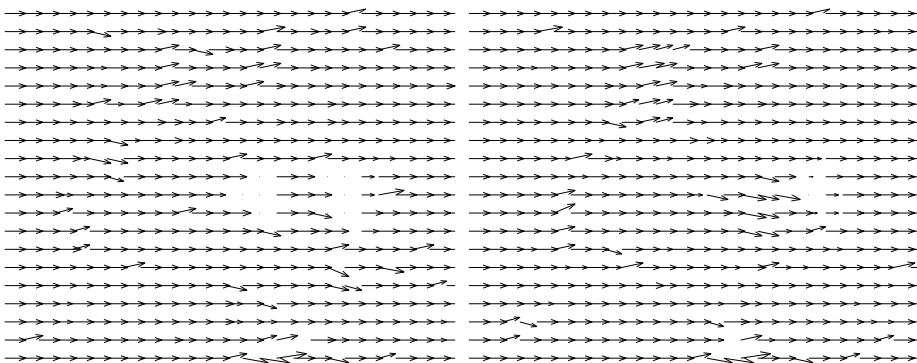
(c)

(d)



(e)

(f)



(g)

(h)

Abbildung 8.6: Ergebnis des Markov-Zufallsfeld-Algorithmus für die Monalisasequenz. (Siehe Text)

8.6 Zusammenfassung

Das sogenannte Block-Matching-Verfahren hat sich in den vergangenen Jahren zu einer beliebten Methode zur Bestimmung einfacher Bewegungen in Bildsequenzen etabliert und unter anderem in verschiedenen Videokompressionsverfahren Einzug gehalten. In diesem Kapitel stellen wir daher ihre Erweiterung hinsichtlich der Bestimmung transparent überlagelter Bewegungen vor.

Aus den Erkenntnissen des vorangegangenen Kapitels leiteten wir zunächst eine Block-Matching-Bedingung für eine beliebige Anzahl von transparent überlagerten Bewegungen her. Um N Bewegungen zu bestimmen, sind $N + 1$ Bilder und 2^N Blöcke nötig. Anschließend analysierten wir das Verhalten der Block-Matchingbedingung in der Nähe von Verdeckungsgrenzen.

Basierend auf diesen theoretischen Untersuchungen entwickelten wir einen hierarchischen Algorithmus, der die Bestimmung von einfachen, mehrfachen transparenten und verdeckenden Bewegungen erlaubt. Die Bestimmung von verdeckenden Bewegungen wird in einer zweiten Phase durchgeführt, bei der alle Bildpunkte in der Nähe der Verdeckungsgrenze ausgeschlossen bleiben, die in der ersten Phase gefunden wurden.

Der hierarchische Algorithmus wurde anhand von realen und synthetischen verrauschten Beispielen getestet. In allen Fällen lieferte er gute Ergebnisse. Für einige Applikationen werden, zum Beispiel aus dem Bereich der medizinischen Bildgebung, robustere Verfahren benötigt. Aus diesem Grund haben wir hier noch eine regularisierte Version des Block-Matchingverfahrens für transparente Bewegungen vorgestellt und gezeigt, dass dieses Verfahren bei zunehmendem Rauschen die Robustheit erhöht. Dieses Verfahren basiert auf Markov-Zufallsfeldern und erlaubt eine simultane Bestimmung der Bewegungen und einer Segmentierung, die die Anzahl beobachteter Bewegungen in jedem Bildpunkt angibt.

Kapitel 9

Verdeckung

Szenarien, bei denen ein Vordergrundobjekt ein Hintergrundobjekt verdeckt, lassen sich in nahezu jeder Bildsequenz beobachten. Die exakte Ermittlung der Bewegungsvektoren an dem Grenzübergang der Objekte bildet daher einen der Hauptschwerpunkte im Bereich der Bewegungsanalyse. Die Trennung von bewegten Vorder- und den Hintergrundobjekten ist insbesondere für die Videokompressions- und Analysezwecke von hoher Bedeutung. An dem Grenzübergang zwischen dem Vordergrund- und Hintergrundobjekt kommt es zu Diskontinuitäten im Fluss/Bewegungsfeld, die einen schnellen Umbruch im Bewegungsfeld ergeben. Standard-Bewegungsbestimmungsverfahren versagen an dem Grenzübergang, da zum Einen in einem kleinen Bereich um den Grenzübergang zwei Bewegungen beobachtet werden und zum Anderen aufgrund der Verdeckungseigenschaft Bildinhalte verschwinden bzw. neue hervortreten können. Die Annahme lokal konstanter Bewegungen, die in der Regel von den Bewegungsbestimmungsverfahren vorausgesetzt wird, ist an diesen Stellen nicht erfüllt.

Entsprechend viel Forschungsarbeit zur Beschreibung und Behandlung der Diskontinuitäten an der Verdeckungsgrenze ist in der Vergangenheit geleistet worden. Einige Ansätze [11, 122, 52, 181, 180, 143, 37, 91] behandeln Diskontinuitäten im Flussfeld, indem sie den quadratischen Glattheitsterm von Horn & Schunck [84] durch Glattheitsterme ersetzen, die stückweise glatte Bewegungsfelder in den Resultaten erlauben. In anderen Ansätzen [47, 142, 106, 82] werden die Diskontinuitäten durch geschlossene Kurven [47, 142, 106, 82, 99] oder Segmentierungen [96, 156, 56, 125, 82] beschrieben, die simultan zur Bewegung bestimmt werden. Letztere basieren meist auf Markov-Zufallsfeldern und sprechen in gewissem Sinne den Dualismus zwischen Bewegungsbestimmung und -segmentierung an: Eine gute Bewegungssegmentierung führt zu einer guten Bewegungsbestimmung und umgekehrt. Zusätzlich zur räumlichen Behandlung von Diskontinuitäten im Flussfeld werden in verschiedenen Verfahren [31, 29, 53, 121, 182, 33] zeitliche Bedingungen zur Verbesserung der Bewegungsbestimmung und Prädiktion der durch die Verdeckungen hervorgerufenen Diskontinuitäten verwendet. Ähnlich wie bei den transparenten Bewegungen werden Schichtenmodelle auch zur Modellierung verdeckender Bewegungen genutzt. Wang & Adelson [177] beschreiben verdeckende Bildbereiche durch eine Menge von überlappenden Schichten. Zunächst berechnen sie in kleinen Regionen die lokalen Bewegungen durch einen Kleinstquadratansatz. Anschließend gruppieren sie die Bewegungen mit Hilfe eines K-Means-Clustering-Verfahrens in Regionen ähnlicher Bewegungen. In diesem

Kontext werden auch gerne Mixture-Modelle [13, 90, 170, 183, 185] zur Einteilung der Bildsequenz in Schichten ähnlicher Bewegungen eingesetzt. Ein interessanter Klassifikationsansatz zur Einteilung der Bildsequenz in Bereiche ähnlicher Bewegungen wird von Vidall & Ma [176] vorgestellt. Der Ausgangspunkt der Klassifikation stellt eine Menge von Korrespondenzpunkten bzw. Bewegungsvektoren dar, die es in N Cluster ähnlicher Bewegungen einzuteilen gilt. Für die Clusterung der Daten nutzen sie einen vergleichbaren multiplikativen Ansatz, wie wir ihn noch bei den verdeckenden Orientierungen im Kapitel 10 verwenden werden. Durch Multiplikation der N Bewegungsmodelle erhält man eine Gleichung, die alle Modelle beinhaltet, von denen aber nur eins zur Zeit zutrifft. Diese Gleichung ist als komplexes Polynom darstellbar, dessen Koeffizienten eine indirekte Darstellung der Clusterparameter sind und aus den zuvor ermittelten Bewegungen errechnet werden. Durch Ableiten des Polynoms und Einsetzen einiger Messwerte erhält man die Clusterparameter, die dann für die Segmentierung im Sinne der kleinsten Euklidischen Distanz verwendet werden.

Als Alternative zu den räumlichen Methoden, behandeln einige Autoren die Bewegungsbestimmung an der Verdeckungsgrenze als ein Ebenensuchproblem im Frequenzraum [187, 20, 151, 21, 190, 189] und setzen es damit den transparenten Bewegungen gleich. Übersichten über die verschiedenen Ansätze im Bereich der Bewegungsbestimmung in Verbindung mit Verdeckungen finden sich in [110, 157, 32].

In diesem Abschnitt möchten wir genauer untersuchen, warum auch die Superpositionsmodelle (transparente Modelle) [174, 147] an der Verdeckungsgrenze versagen. Als Ergebnis leiten wir eine neue räumliche Bedingungsgleichung für den von der Verdeckungsgrenze induzierten Störterm her [117, 18, 19]. Basierend auf dieser analytischen Beschreibung zeigen wir, dass der Störterm einen hyperbolischen Verlauf im Frequenzraum aufweist, unabhängig von der Form der Verdeckungsgrenze. Ferner leiten wir für den Spezialfall einer geradlinig verlaufenden Verdeckungsgrenze den exakten Ausdruck des Störterms her [117]. Dieser Ausdruck zeigt, dass das Spektrum des Störterms entlang bestimmter paralleler Ebenen das gleiche aber gewichtete und verschobene Profil aufweist wie das Spektrum des verdeckten Signals. Für die Analyse verwenden wir das Verdeckungsmodell von Fleet & Langley [64], die ebenfalls verdeckende Bewegungen im Frequenzraum analysierten und deren Ergebnisse von einer Reihe von Autoren [190, 187, 20, 21, 116] weiterentwickelt wurde.

9.1 Verdeckende Bewegungen

Für die Beschreibung der verdeckenden Bewegungen nutzen wir das Modell von Fleet & Langley [64]. Dabei bewegt sich das Vordergrundobjekt mit der konstanten Geschwindigkeit \mathbf{u} und das Hintergrundobjekt mit der konstanten Geschwindigkeit \mathbf{v} . Die Bildsequenz wird durch die Superposition

$$f(\mathbf{x}, t) = \chi(\mathbf{x} - t\mathbf{u})g_1(\mathbf{x} - t\mathbf{u}) + [1 - \chi(\mathbf{x} - t\mathbf{u})]g_2(\mathbf{x} - t\mathbf{v}) \quad (9.1)$$

modelliert, wobei

$$\chi(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega \\ 0 & \text{sonst} \end{cases} \quad (9.2)$$

die charakteristische Funktion des Vordergrundobjektes ist und Ω dessen Bereich darstellt. Der Richtungsableitungsoperator

$$\alpha(\mathbf{u}) = u_x \partial_x + u_y \partial_y + \partial_t \quad (9.3)$$

erfüllt die Bewegungsbedingungen $\alpha(\mathbf{u})f(\mathbf{x}, t) = 0$ innerhalb von Ω und $\alpha(\mathbf{v})f(\mathbf{x}, t) = 0$ außerhalb von Ω . Damit ist die Gleichung $\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t) = 0$ überall gültig mit Ausnahme an den Verdeckungsgrenzen. In den folgenden Abschnitten soll der auftretende Störterm an der Verdeckungsgrenze, der sich z.B. als Residuum der transparenten Bewegungsgleichung ergibt, genauer untersucht werden.

9.2 Räumliche Analyse der Bewegung an der Verdeckungsgrenze

Im Folgenden möchten wir den Bewegungsoperator $\alpha(\mathbf{u})\alpha(\mathbf{v})$ auf das Bewegungsmodell an der Verdeckungsgrenze anwenden. Um die Kommutativität der Ableitungsoperatoren zu gewährleisten, seien "lokal konstante Bewegung" vorausgesetzt.

Wir beginnen die Analyse mit der Anwendung des Operators $\alpha(\mathbf{u})$ auf Gleichung (9.1) und erhalten die Gleichung

$$\alpha(\mathbf{u})f(\mathbf{x}, t) = [1 - \chi(\mathbf{x} - t\mathbf{u})]\alpha(\mathbf{u})g_2(\mathbf{x} - t\mathbf{v}). \quad (9.4)$$

Der Ausdruck $\alpha(\mathbf{u})g_2(\mathbf{x} - t\mathbf{v})$ lässt sich mit Hilfe von Gleichung (9.3) und $\frac{\partial}{\partial t}g_2(\mathbf{x} - t\mathbf{v}) = -\mathbf{v} \cdot \nabla g_2(\mathbf{x} - t\mathbf{v})$ weiter zu

$$\alpha(\mathbf{u})g_2(\mathbf{x} - t\mathbf{v}) = (\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v}) \quad (9.5)$$

vereinfachen und ermöglicht obige Gleichung als

$$\alpha(\mathbf{u})f(\mathbf{x}, t) = [1 - \chi(\mathbf{x} - t\mathbf{u})](\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v}). \quad (9.6)$$

zu schreiben.

Die Anwendung des Vordergrundoperators $\alpha(\mathbf{v})$ auf die Gleichung (9.6) ergibt

$$\alpha(\mathbf{v})\alpha(\mathbf{u})f(\mathbf{x}, t) = -\alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})](\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v}). \quad (9.7)$$

Bei der Differenzierung tritt das Problem auf, dass wir Ableitungen an der Verdeckungsgrenze bilden müssen. Zur Ableitung an der Diskontinuität von χ wenden wir die Distributionstheorie an. Die Differenzierung an der Diskontinuität wird über eine partielle Integration auf die Differenzierung einer glatten und kompakten Testfunktion¹ ϕ übertragen:

$$\left\langle \frac{\partial \chi}{\partial x}, \phi \right\rangle = - \int \chi(\mathbf{x}) \frac{\partial \phi}{\partial x}(\mathbf{x}) d\mathbf{x}. \quad (9.8)$$

Da der Störterm nur an der Verdeckungsgrenze auftritt, ist die Anwendung des Operators $\alpha(\mathbf{v})$ auf die Unstetigkeit von besonderem Interesse und wir werden den folgenden Zusammenhang beweisen:

¹Eine Testfunktion $\phi(x)$ geht einschließlich ihrer Ableitungen schneller als eine Potenz von $1/|x|$ für $|x| \rightarrow \infty$ gegen null.

Lemma 1 *An der Verdeckungsgrenze χ gilt*

$$-\alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})] = (\mathbf{u} - \mathbf{v}) \cdot \nabla\chi(\mathbf{x} - t\mathbf{u}). \quad (9.9)$$

Beweis:

Genaugenommen ist die zu zeigende Aussage eine direkte Konsequenz von Gleichung (9.5). Da die Funktion χ jedoch nicht differenzierbar ist, weisen wir die zu zeigende Aussage zur Vollständigkeit mathematisch korrekt im Sinne der Distributionstheorie nach:

Durch partielle Integration erhalten wir mit (9.3)

$$-\langle \alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})], \phi \rangle = \int \chi(\mathbf{x} - t\mathbf{u}) [\mathbf{v} \cdot \nabla\phi(\mathbf{x}, t) + \phi_t(\mathbf{x}, t)] d\mathbf{x}dt. \quad (9.10)$$

Die Substitution $\mathbf{y} = \mathbf{x} - t\mathbf{u}$ führt zu

$$-\langle \alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})], \phi \rangle = \int \chi(\mathbf{y}) [\mathbf{v} \cdot \nabla\phi(\mathbf{y} + t\mathbf{u}, t) + \phi_t(\mathbf{y} + t\mathbf{u}, t)] d\mathbf{y}dt. \quad (9.11)$$

Die totale Differenziation von $\phi(\mathbf{y} + t\mathbf{u}, t)$ ergibt

$$\frac{d\phi(\mathbf{y} + t\mathbf{u}, t)}{dt} = \mathbf{u} \cdot \nabla\phi(\mathbf{y} + t\mathbf{u}, t) + \phi_t(\mathbf{y} + t\mathbf{u}, t) \quad (9.12)$$

und daher können wir die zeitliche Ableitung von $\phi(\mathbf{y} + t\mathbf{u}, t)$ als

$$\phi_t(\mathbf{y} + t\mathbf{u}, t) = -\mathbf{u} \cdot \nabla\phi(\mathbf{y} + t\mathbf{u}, t) + \frac{d\phi(\mathbf{y} + t\mathbf{u}, t)}{dt} \quad (9.13)$$

schreiben. Eingesetzt in Gleichung (9.11) ergibt dies

$$\begin{aligned} -\langle \alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})], \phi \rangle &= \iint \chi(\mathbf{y})(\mathbf{v} - \mathbf{u}) \cdot \nabla\phi(\mathbf{y} + t\mathbf{u}, t) d\mathbf{y}dt \\ &\quad + \int \chi(\mathbf{y}) \left(\int \frac{d}{dt} \phi(\mathbf{y} + t\mathbf{u}, t) dt \right) d\mathbf{y} \\ &= \iint \chi(\mathbf{y})(\mathbf{v} - \mathbf{u}) \cdot \nabla\phi(\mathbf{y} + t\mathbf{u}, t) d\mathbf{y}dt \\ &\quad + \int \chi(\mathbf{y}) [\phi(\mathbf{y} + t\mathbf{u}, t)]_{-\infty}^{\infty} d\mathbf{y} \\ &= \iint (\mathbf{u} - \mathbf{v}) \cdot \nabla\chi(\mathbf{y}) \phi(\mathbf{y} + t\mathbf{u}, t) d\mathbf{y}dt, \end{aligned} \quad (9.14)$$

da $\phi(\mathbf{y} + t\mathbf{u}, t)$ über t integriert null ergibt, denn die Testfunktion ϕ verschwindet an den Integrationsgrenzen. Eine Rücksubstitution ergibt die zu zeigende Aussage. \square

Lemma 1 ermöglicht uns eine weitere Verfeinerung von Gleichung (9.7). Im Folgenden sei $B(\mathbf{x})$ eine glatte Indikatorfunktion, die an der Grenze von Ω verschwindet und für die gilt $|\nabla B(\mathbf{x})| = 1$. Solch ein $B(\mathbf{x})$ erhält man z.B. durch die Konstruktion einer Distanzfunktion, die für jeden Punkt \mathbf{y} den minimalen Abstand zu der Kurve $B(\mathbf{x}) = 0$, die uns interessiert, angibt. Solche Distanzfunktionen werden unter anderem auch bei den sogenannten *Aktiven Konturen*, z.B. den *Level Sets* von Sethian [145], eingesetzt. Wir zeigen:

Lemma 2 *Ist eine Bildsequenz durch Gleichung (9.1) gegeben und beschreibt $B(\mathbf{x}) = 0$ die Grenze des Bereiches Ω und $\mathbf{N}(\mathbf{x}) = \nabla B(\mathbf{x})$ die Normale dieser Grenze, so gilt:*

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t) = q(\mathbf{x}, t, \mathbf{u}, \mathbf{v})\delta(B(\mathbf{x} - t\mathbf{u})) \quad (9.15)$$

mit

$$q(\mathbf{x}, t, \mathbf{u}, \mathbf{v}) = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x} - t\mathbf{u})(\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v}). \quad (9.16)$$

Beweis: Zunächst möchten wir den Ausdruck $(\mathbf{u} - \mathbf{v}) \cdot \nabla \chi(\mathbf{x})$ aus Lemma 1 genauer untersuchen. Wir betrachten diese Aussage im Zweidimensionalen, da der zeitliche Anteil nur eine Verschiebung der zweidimensionalen Funktion $\chi(\mathbf{x})$ bewirkt. Zur vereinfachten Schreibweise setzen wir im folgenden $\mathbf{w} = \mathbf{u} - \mathbf{v}$. Nach Lemma 1 gilt

$$\langle \mathbf{w} \cdot \nabla \chi, \phi \rangle = - \int \chi(\mathbf{x}) \mathbf{w} \cdot \nabla \phi(\mathbf{x}) d\mathbf{x} = - \int_{\Omega} \mathbf{w} \cdot \nabla \phi(\mathbf{x}) d\mathbf{x}. \quad (9.17)$$

Da für den Ausdruck innerhalb des Integrals $\mathbf{w} \cdot \nabla \phi(\mathbf{x}) = \operatorname{div}(\mathbf{w} \phi(\mathbf{x}))$ ² gilt, können wir die Gaußsche-Integralformel für die Ebene anwenden und obige Gleichung zu

$$\langle \mathbf{w} \cdot \nabla \chi, \phi \rangle = - \int_{B(\mathbf{x})=0} \mathbf{w} \cdot \mathbf{N}(\mathbf{x}) \phi(\mathbf{x}) ds \quad (9.18)$$

umschreiben, wobei s die Bogenlänge von B ist und $B(\mathbf{x}) = 0$ den äußeren Rand von Ω beschreibt. Entsprechend der Distributionstheorie beschreibt $\delta(B(\mathbf{x}))$ die Diskontinuität an der Kante von Ω

$$\langle \delta(B(\mathbf{x})), \xi \rangle = \int_B \xi(\mathbf{x}) ds, \quad (9.19)$$

wir setzen $\xi = \mathbf{w} \cdot \mathbf{N}(\mathbf{x}) \phi(\mathbf{x})$, damit gilt

$$\langle \mathbf{w} \cdot \nabla \chi, \phi \rangle = - \langle \delta(B(\mathbf{x})), \mathbf{w} \cdot \mathbf{N}(\mathbf{x}) \phi \rangle = - \langle \mathbf{w} \cdot \mathbf{N}(\mathbf{x}) \delta(B(\mathbf{x})), \phi \rangle. \quad (9.20)$$

Wir haben nun gezeigt, dass

$$(\mathbf{u} - \mathbf{v}) \cdot \nabla \chi = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x}) \delta(B(\mathbf{x})) \quad (9.21)$$

ist und damit auch für das zeitlich verschobene $\chi(\mathbf{x} - t\mathbf{u})$

$$\alpha(\mathbf{v})[\chi(\mathbf{x} - t\mathbf{u})] = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x} - t\mathbf{u}) \delta(B(\mathbf{x} - t\mathbf{u})). \quad (9.22)$$

Zum Schluss setzen wir diesen Ausdruck in die Gleichung (9.7) ein und erhalten die zu zeigende Aussage

$$\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t) = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x} - t\mathbf{u}) \delta(B(\mathbf{x} - t\mathbf{u})) (\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v}). \quad (9.23)$$

□

Gleichung (9.15) hat eine einfache und intuitive Bedeutung: $\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t)$ ist fast

² $\phi(\mathbf{x})$ ist ein skalares Feld und \mathbf{w} ein Vektor. Für die Divergenz gilt mit der Produktregel $\operatorname{div}(\mathbf{w} \phi(\mathbf{x})) = \phi(\mathbf{x}) \operatorname{div} \mathbf{w} + \mathbf{w} \cdot \nabla \phi(\mathbf{x})$ und außerdem $\operatorname{div} \mathbf{w} = 0$, da es sich hier um lokal konstante Bewegungen handelt und \mathbf{w} damit nicht von \mathbf{x} abhängt.

überall null nur nicht an der Verdeckungsgrenze, wo sie aufgrund der Diskontinuität des Grenzübergangs der Maske χ zu einer Dirac-Distribution wird. Daher versagt die Bewegungsbestimmung bei Verdeckungen, denn an der Verdeckungsgrenze sind sowohl Gleichung $\alpha(\mathbf{v})f(\mathbf{x}, t) = 0$ als auch Gleichung $\alpha(\mathbf{v})\alpha(\mathbf{u})f(\mathbf{x}, t) = 0$ nicht erfüllt. Richtig ist jedoch Gleichung (9.15). Um die zwei verdeckenden Bewegungen zu bestimmen, können wir entweder Gleichung $\alpha(\mathbf{v})\alpha(\mathbf{u})f(\mathbf{x}, t) = 0$ verwenden und dabei nicht die Punkte an der Verdeckungsgrenze in die Berechnung eingehen lassen [18, 19] oder wir lösen direkt Gleichung (9.15).

Aus Gleichung (9.15) und Gleichung (9.16) lassen sich folgende Schlüsse ziehen: der Störterm ist

1. auf die Verdeckungsgrenze beschränkt,
2. minimal, wenn die Normale der Verdeckungsgrenze senkrecht zur relativen Bewegung (Differenz zwischen Vordergrund- und Hintergrundbewegung) ist und maximal, wenn die beiden Vektoren in die gleiche Richtung zeigen,
3. proportional zu der Intensität des Gradienten der Hintergrundbewegung.

9.3 Verdeckung im Frequenzraum

Die Beobachtung, dass eine einfache Bewegung im Ortsraum zu einer Ebene im Frequenzraum korrespondiert, hat in der Vergangenheit zu einer Reihe von verschiedenen Ansätzen zur Bewegungsbestimmung geführt. Bisher beschränkte sich die Fourier-Analyse von verdeckenden Bewegungen nur auf den Fall von geradlinig verlaufenden Verdeckungsgrenzen. Im Gegensatz dazu ermöglichen die Ergebnisse des vorangegangenen Abschnittes diese Einschränkungen aufzuheben, wie wir jetzt zeigen werden. Zunächst setzen wir zur vereinfachenden Schreibweise

$$h(\mathbf{x}) = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x})\delta(B(\mathbf{x})). \quad (9.24)$$

Mit dieser Notation ergibt das Residual auf der rechten Seite von Gleichung (9.16)

$$r(\mathbf{x}, t) = (\mathbf{u} - \mathbf{v}) \cdot \nabla g_2(\mathbf{x} - t\mathbf{v})h(\mathbf{x} - t\mathbf{u}) \quad (9.25)$$

und in den Frequenzraum übertragen

$$\begin{aligned} R(\boldsymbol{\xi}, \xi_t) &= 2\pi j [(\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\xi} G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)] * [H(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)] \\ &= 2\pi j \iint (\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\omega} G_2(\boldsymbol{\omega})\delta(\mathbf{v} \cdot \boldsymbol{\omega} + \omega_t) \\ &\quad H(\boldsymbol{\xi} - \boldsymbol{\omega})\delta(\mathbf{u} \cdot (\boldsymbol{\xi} - \boldsymbol{\omega}) + \xi_t - \omega_t) d\boldsymbol{\omega} d\omega_t, \end{aligned} \quad (9.26)$$

wobei $G_2(\boldsymbol{\xi})$ die Fouriertransformierte von $g_2(\mathbf{x})$ ist und $H(\boldsymbol{\xi})$ die von $h(\mathbf{x})$. Bei der Auswertung der Faltung kommt es unter anderem zur Integration des Ausdrucks

$$\delta(\mathbf{v} \cdot \boldsymbol{\omega} + \omega_t)\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t - \mathbf{u} \cdot \boldsymbol{\omega} - \omega_t) \quad (9.27)$$

in Abhängigkeit der Faltungsvariablen $\boldsymbol{\omega}$ und ω_t . Dieser Ausdruck ist nur dann ungleich null, wenn die Argumente beider Dirac-Stöße gleichzeitig null ergeben. Die Menge L der Punkte, für die dies gleichzeitig gilt, ist gegeben als Lösung des Gleichungssystems

$$\begin{aligned}\mathbf{v} \cdot \boldsymbol{\omega} + \omega_t &= 0 \\ \mathbf{u} \cdot \boldsymbol{\omega} + \omega_t &= \mathbf{u} \cdot \boldsymbol{\xi} + \xi_t.\end{aligned}\tag{9.28}$$

In Anhang D zeigen wir, dass die Schnittmenge zweier δ -Ebenen eine δ -Gerade ergibt. Im obigen Fall verläuft diese Gerade in Richtung $(\mathbf{v} + \mathbf{e}_t) \times (\mathbf{u} + \mathbf{e}_t)$ und besitzt den Querschnitt³

$$d = |(\mathbf{u} + \mathbf{e}_t) \times (\mathbf{v} + \mathbf{e}_t)|,\tag{9.29}$$

wobei \mathbf{e}_t den zeitlichen Einheitsvektor darstellt.

Aus dem Gleichungssystem (9.28) folgt, dass $(\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\omega} = \mathbf{u} \cdot \boldsymbol{\xi} + \xi_t$ gilt, nicht explizit von ω_t abhängt und vor dem Integral stehen kann

$$R(\boldsymbol{\xi}, \xi_t) = \frac{2\pi j}{d} (\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) \int_L G_2(\boldsymbol{\omega}) H(\boldsymbol{\xi} - \boldsymbol{\omega}) \mathbf{d}s,\tag{9.30}$$

wobei s die Bogenlänge von L ist. Bemerkenswert ist, dass die Gerade L eine Gerade im dreidimensionalen Raum ist, im Faltungsintegral aber nur die 2D-Anteile $\boldsymbol{\omega}$ und $\boldsymbol{\xi}$ verwendet werden. Der Anteil ω_t geht demnach nicht in die Integration ein. Zurückzuführen ist dies auf die Tatsache, dass G_2 und H zweidimensional sind. Damit können wir nun folgende Aussage über die Fouriertransformierten der Bildsequenz treffen.

Satz 8 *Das Spektrum der Bildsequenz $f(\mathbf{x}, t)$ ist gegeben durch*

$$F(\boldsymbol{\xi}, \xi_t) = A(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) + B(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t) + C(\boldsymbol{\xi}, \xi_t)\tag{9.31}$$

mit

$$C(\boldsymbol{\xi}, \xi_t) = \frac{1}{2d\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} \int_L G_2(\boldsymbol{\omega}) H(\boldsymbol{\xi} - \boldsymbol{\omega}) \mathbf{d}s,\tag{9.32}$$

wobei die Spektren $A(\boldsymbol{\xi})$ und $B(\boldsymbol{\xi})$ noch zu bestimmen sind.

Beweis:

Um diese Aussage nachweisen zu können, muss die Operation $\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t)$ rückgängig gemacht werden, um $f(\mathbf{x}, t)$ zu erhalten. Im Frequenzraum schreibt sich diese Operation als $2\pi j(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)2\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)F(\boldsymbol{\xi}, \xi_t)$. Demnach ergibt sich das Spektrum des Störterms mit (9.30)

$$\begin{aligned}C(\boldsymbol{\xi}, \xi_t) &= \frac{1}{2\pi j(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)2\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} R(\boldsymbol{\xi}, \xi_t) \\ &= \frac{1}{2d\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} \int_L G_2(\boldsymbol{\omega}) H(\boldsymbol{\xi} - \boldsymbol{\omega}) \mathbf{d}s.\end{aligned}\tag{9.33}$$

³Der Querschnitt einer δ -Geraden und auch einer δ -Ebene entspricht dem Wert des Flächenintegrals in Richtung des Normalenvektors der Geraden bzw. Ebene. Ist die Gerade oder Ebene in Hessescher Normalform, so ist der Querschnitt ungeachtet ihrer Richtung eins.

Ferner wissen wir aus der Analyse transparenter Bewegungen, dass die Operation $2\pi j(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)2\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)F(\boldsymbol{\xi}, \xi_t)$ alle Anteile, die auf einer der beiden Ebenen $\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)$ und $\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)$ liegen, entfernt. Demnach kommen noch zwei weitere Anteile, die durch $A(\boldsymbol{\xi})$ und $B(\boldsymbol{\xi})$ beschrieben seien, zu dem Spektrum $F(\boldsymbol{\xi}, \xi_t)$ hinzu:

$$F(\boldsymbol{\xi}, \xi_t) = A(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) + B(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t) + C(\boldsymbol{\xi}, \xi_t). \quad (9.34)$$

□

Aus Gleichung (9.1) ergibt sich

$$A(\boldsymbol{\xi}) = \hat{\chi}(\boldsymbol{\xi}) * G_1(\boldsymbol{\xi}) \quad (9.35)$$

und entspricht dem Anteil $\chi(\mathbf{x} - t\mathbf{u})g_1(\mathbf{x} - t\mathbf{u})$ im Ortsraum, wobei $\hat{\chi}(\boldsymbol{\xi})$ die Fouriertransformierte von $\chi(\mathbf{x})$ ist. Der Anteil $g_2(\mathbf{x} - t\mathbf{v})$ wird im Frequenzraum auf die Ebene $\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)$ abgebildet und daher

$$B(\boldsymbol{\xi}) = a G_2(\boldsymbol{\xi}), \quad (9.36)$$

wobei a ein unbekannter Faktor ist. Nach dem Ausdruck $\chi(\mathbf{x} - t\mathbf{u})g_2(\mathbf{x} - t\mathbf{v})$ in Gleichung (9.1) ist der Faktor a ungleich eins, falls das Spektrum von $\chi(\mathbf{x})$ Dirac-Anteile der Form $\delta(\boldsymbol{\xi})$ enthält. Die Faltung von $G_2(\boldsymbol{\xi})$ mit einer dieser Dirac-Funktionen verändert G_2 , abgesehen von einer möglichen Skalierung, nicht und addiert sich daher zu dem schon vorhandenen Spektrum $G_2(\boldsymbol{\xi})$ in der Ebene $\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)$ des 3D-Spektrums. Solche Anteile treten, wie wir noch sehen werden, z.B. bei geradlinig verlaufenden Verdeckungsgrenzen auf und wir ordnen sie $B(\boldsymbol{\xi})$ zu. Da im Allgemeinen weder die Funktion $\chi(\mathbf{x})$ noch ihr Spektrum im Voraus bekannt sind, können ohne diese Zusatzinformation $A(\boldsymbol{\xi})$ und $B(\boldsymbol{\xi})$ nicht exakt angegeben werden.

9.3.1 Verlauf des Störterms

Die vorausgegangene Analyse lässt noch die Frage offen, wie der Verlauf des Störterms im Frequenzraum aussieht. Wir werden nun zeigen, dass das Spektrum des Störterms einen hyperbolischen Verlauf (mit $1/x$ auslaufend) hat, wenn $|H(\boldsymbol{\omega})|$ beschränkt bzw. nach oben abschätzbar ist. Zunächst stellen wir anhand des Anhangs D fest, dass der Integrationsbeich L für jeden Punkt $\boldsymbol{\xi} + \xi_t \mathbf{e}_t$ unverändert bleibt: dieser Punkt liegt in einer Ebene, die durch $\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t = c$ beschreibbar ist und damit nur eine Änderung in der Verschiebungskonstante in der Produktbildung unserer beiden Dirac-Stöße bewirkt. Diese Änderung hat, wie wir jedoch wissen, keine Auswirkung auf den Integrationsbereich L . Für die Fouriertransformierte von $h(\mathbf{x})$ aus Gleichung (9.24) gilt

$$H(\boldsymbol{\omega}) = - \int (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x}) \delta(B(\mathbf{x})) e^{(-j\boldsymbol{\omega} \cdot \mathbf{x})} \mathbf{d}\mathbf{x} \quad (9.37)$$

$$= - \int_{B(\mathbf{x})=0} (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}(\mathbf{x}) e^{(-j\boldsymbol{\omega} \cdot \mathbf{x})} \mathbf{d}s. \quad (9.38)$$

Da aber nach Voraussetzung $|\mathbf{N}(\mathbf{x})| = |\nabla B(\mathbf{x})| = 1$ ist, können wir die folgende Abschätzung machen

$$|H(\boldsymbol{\omega})| \leq \int_{B(\mathbf{x})=0} |\mathbf{u} - \mathbf{v}| \mathbf{d}s = |\mathbf{u} - \mathbf{v}| \int_{B(\mathbf{x})=0} \mathbf{d}s. \quad (9.39)$$

In vielen praktischen Anwendungen können wir davon ausgehen, dass die Länge der Verdeckungsgrenze endlich und damit $|H(\boldsymbol{\omega})|$ beschränkt ist und wir erhalten die Abschätzung

$$|H(\boldsymbol{\omega})| \leq |\mathbf{u} - \mathbf{v}| \text{length}(B(\mathbf{x}) = 0). \quad (9.40)$$

Damit haben wir gezeigt, dass der Störterm $C(\boldsymbol{\xi}, \xi_t)$ einen hyperbolischen Verlauf hat. Den Betrag des Störterms können wir schreiben als

$$\begin{aligned} |C(\boldsymbol{\xi}, \xi_t)| &= \left| \frac{1}{2d\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} \int_L G_2(\boldsymbol{\omega}) H(\boldsymbol{\xi} - \boldsymbol{\omega}) \mathbf{d}s \right| \\ &\leq \left| \frac{|\mathbf{u} - \mathbf{v}| \text{length}(B(\mathbf{x})=0)}{2d\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} \right| \left| \int_L G_2(\boldsymbol{\omega}) \mathbf{d}s \right| \end{aligned} \quad (9.41)$$

Die Integration von $G_2(\boldsymbol{\omega})$ ist unabhängig von $\boldsymbol{\xi} + \xi_t \mathbf{e}_t$ und ergibt einen endlichen Wert, falls $G_2(\boldsymbol{\omega})$ bandbegrenzt ist, was für jede praktische Anwendung vorausgesetzt werden kann. Offensichtlich bestimmt der Nenner $2\pi j(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)$ das hyperbolische Abklingverhalten.

Die bisherige Betrachtung gilt nur für den Fall geschlossener Verdeckungsgrenzen endlicher Länge. In dem folgenden Abschnitt werden wir der Frage nachgehen, ob wir vergleichbare Aussagen über das Abklingverhalten des Störterms im Fall geradlinig verlaufender Verdeckungsgrenzen herleiten können, die in diesem Fall unendlich lang und nicht geschlossen sind.

9.4 Gerade Verdeckungsgrenze

Die Ergebnisse der vorangegangenen Abschnitte gelten für beliebige Verläufe der Verdeckungsgrenzen. Hier möchten wir nun den Fall von geradlinig verlaufenden Verdeckungsgrenzen genauer betrachten, da sie in der Praxis häufig auftreten. Das Spektrum von verdeckenden Bewegungen wurde von Beauchemin et al. [21, 20, 189, 190, 187] unter der Annahme einer geradlinig verlaufenden Verdeckungsgrenze untersucht. Sie fanden heraus, dass in diesem Falle die Spektren aus Satz 8 gegeben sind durch

$$\begin{aligned} A(\boldsymbol{\xi}) &= \left[\pi \delta(\boldsymbol{\xi}) + \frac{1}{j \mathbf{N} \cdot \boldsymbol{\xi}} \delta(\mathbf{N}^\perp \cdot \boldsymbol{\xi}) \right] * G_1(\boldsymbol{\xi}) \\ B(\boldsymbol{\xi}) &= (1 - \pi) G_2(\boldsymbol{\xi}), \end{aligned} \quad (9.42)$$

wobei \mathbf{N}^\perp der Einheitsvektor senkrecht zu \mathbf{N} ist. Sie bemerkten auch das hyperbolische Abklingen des Störterms $C(\boldsymbol{\xi}, \xi_t)$, die genaue Gleichung des Störterms blieb aber offen. Wir wollen nun die Resultate der vorangegangenen Abschnitte nutzen, um den Störterm $C(\boldsymbol{\xi}, \xi_t)$ auch für nicht geradlinig verlaufende Verdeckungsgrenzen zu bestimmen. Beschreibt der Bereich Ω des verdeckenden Signals eine Halbebene, so ist die Verdeckungsgrenze durch die Gerade $B(\mathbf{x}) = \mathbf{N} \cdot \mathbf{x} = 0$ gegeben und wir können das Integral in Gleichung (9.32) bestimmen. Entsprechend dieser Modellannahme ist die Indikatorfunktion $\chi(\mathbf{x})$ aus Gleichung (9.1) durch die Sprungfunktion

$$\chi(\mathbf{x}) = \begin{cases} 1 & \mathbf{N} \cdot \mathbf{x} \geq 0 \\ 0 & \text{sonst} \end{cases} \quad (9.43)$$

gegeben und ihre Fouriertransformierte (siehe Anhang E) lautet

$$\hat{\chi}(\boldsymbol{\xi}) = \frac{1}{2}\delta(\boldsymbol{\xi}) + \frac{1}{2\pi j \mathbf{N} \cdot \boldsymbol{\xi}}\delta(\mathbf{N}^\perp \cdot \boldsymbol{\xi}). \quad (9.44)$$

Die Fouriertransformierte unseres Bewegungsmodells (9.1) ist durch

$$\begin{aligned} F(\boldsymbol{\xi}, \xi_t) &= [\hat{\chi}(\boldsymbol{\xi}) * G_1(\boldsymbol{\xi})]\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) + G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t) \\ &\quad - [\hat{\chi}(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)] * [G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)] \\ &= A(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) + B(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t) \\ &\quad - \left[\frac{1}{2\pi j \mathbf{N} \cdot \boldsymbol{\xi}}\delta(\mathbf{N}^\perp \cdot \boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) \right] * [G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)] \end{aligned} \quad (9.45)$$

gegeben mit

$$\begin{aligned} A(\boldsymbol{\xi}) &= \left[\frac{1}{2}\delta(\boldsymbol{\xi}) + \frac{1}{2\pi j \mathbf{N} \cdot \boldsymbol{\xi}}\delta(\mathbf{N}^\perp \cdot \boldsymbol{\xi}) \right] * G_1(\boldsymbol{\xi}), \\ B(\boldsymbol{\xi}) &= \left(1 - \frac{1}{2}\right)G_2(\boldsymbol{\xi}). \end{aligned} \quad (9.46)$$

Die Unterschiede in den Ausdrücken in Gleichung (9.42) und (9.46) sind lediglich auf die unterschiedlichen Konventionen der verwendeten Fouriertransformationen zurückzuführen. Wir verwenden die Fouriertransformation, die in der Signalverarbeitung üblich ist und die Einheit Hertz besitzt, während Beauchemin die in der Mathematik gebräuchliche Fouriertransformation benutzt. In beiden Konventionen sind Anteile des Störterms mit in $B(\boldsymbol{\xi})$ eingeflossen, nämlich der Anteil $[1/2\delta(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)] * [G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)]$ der Faltung $[\hat{\chi}(\boldsymbol{\xi})\delta(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)] * [G_2(\boldsymbol{\xi})\delta(\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)]$ aus Gleichung (9.45).

Im Folgenden möchten wir den Spektralanteil des Störterms genauer untersuchen. Dazu verwenden wir die gewonnenen Erkenntnisse aus den vorangegangenen Abschnitten. Mit den oben eingeführten Modellannahmen ist

$$h(\mathbf{x}) = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} \delta(\mathbf{N} \cdot \mathbf{x}) \quad (9.47)$$

und die Fouriertransformierte von $h(\mathbf{x})$

$$H(\boldsymbol{\xi}) = -(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} \delta(\mathbf{N}^\perp \cdot \boldsymbol{\xi}). \quad (9.48)$$

Setzen wir $H(\boldsymbol{\xi})$ in Gleichung (9.26) ein, so erhalten wir für das Residual

$$\begin{aligned} R(\boldsymbol{\xi}, \xi_t) &= 2\pi j \int (\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\omega} G_2(\boldsymbol{\omega}) \delta(\mathbf{v} \cdot \boldsymbol{\omega} + \omega_t) (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} \delta(\mathbf{N}^\perp \cdot (\boldsymbol{\xi} - \boldsymbol{\omega})) \\ &\quad \delta(\mathbf{u} \cdot \boldsymbol{\omega} + \omega_t - \mathbf{u} \cdot \boldsymbol{\xi} - \xi_t) d\boldsymbol{\omega}. \end{aligned} \quad (9.49)$$

Wie vorher auch ist der Bereich, in dem der Integrand ungleich null wird, beschrieben durch das System

$$\mathbf{v} \cdot \boldsymbol{\omega} + \omega_t = 0 \quad (9.50)$$

$$\mathbf{N}^\perp \cdot \boldsymbol{\omega} = \mathbf{N}^\perp \cdot \boldsymbol{\xi} \quad (9.51)$$

$$\mathbf{u} \cdot \boldsymbol{\omega} + \omega_t = \mathbf{u} \cdot \boldsymbol{\xi} + \xi_t. \quad (9.52)$$

Im ersten Schritt werden wir den Querschnitt des Schnittes der drei Dirac-Funktionen bestimmen. Nach Anhang D und Gleichung (9.29) errechnet sich der Querschnitt d anhand der Formel

$$d = |[(\mathbf{u} + \mathbf{e}_t) \times (\mathbf{v} + \mathbf{e}_t)] \cdot \mathbf{N}^\perp| \quad (9.53)$$

$$= |(\mathbf{u} \times \mathbf{v} + (\mathbf{u} - \mathbf{v}) \times \mathbf{e}_t) \cdot \mathbf{N}^\perp| \quad (9.54)$$

Da \mathbf{u} , \mathbf{v} und \mathbf{N}^\perp in einer Ebene liegen und $\mathbf{u} \times \mathbf{v}$ senkrecht auf \mathbf{u} und \mathbf{v} steht, muss dies auch für \mathbf{N}^\perp gelten und wir haben $(\mathbf{u} \times \mathbf{v}) \cdot \mathbf{N}^\perp = 0$. Damit ergibt sich der Querschnitt zu

$$d = |((\mathbf{u} - \mathbf{v}) \times \mathbf{e}_t) \cdot \mathbf{N}^\perp|. \quad (9.55)$$

Der Ausdruck $((\mathbf{u} - \mathbf{v}) \times \mathbf{e}_t) \cdot \mathbf{N}^\perp$ ist ein Spatprodukt und zahlenmäßig gleich dem Volumen des von den drei Vektoren gebildeten Parallelepipeds. Da \mathbf{e}_t die Länge eins hat, ist das Volumen zahlenmäßig gleich der durch $(\mathbf{u} - \mathbf{v}) \times \mathbf{N}^\perp$ gebildeten Fläche und für den Querschnitt gilt

$$d = |(\mathbf{u} - \mathbf{v}) \times \mathbf{N}^\perp|. \quad (9.56)$$

Betrachten wir nun den Ausdruck innerhalb der Betragsstriche etwas genauer:

$$\begin{aligned} (\mathbf{u} - \mathbf{v}) \times \mathbf{N}^\perp &= |\mathbf{u} - \mathbf{v}| |\mathbf{N}^\perp| \sin(90 - \theta) \\ &= |\mathbf{u} - \mathbf{v}| |\mathbf{N}| \cos(\theta) = (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}, \end{aligned} \quad (9.57)$$

wobei θ der Winkel zwischen den Vektoren $(\mathbf{u} - \mathbf{v})$ und \mathbf{N} ist. Der Querschnitt ergibt sich demnach zu

$$d = |(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}|. \quad (9.58)$$

Um die Bestimmung von $C(\boldsymbol{\xi}, \xi_t)$ abschließen zu können, benötigen wir den Punkt, für den die drei Dirac-Funktionen innerhalb des Faltungsintegrals (9.49) in Abhängigkeit von $\boldsymbol{\xi}$ und ξ_t gleichzeitig ungleich null sind. Dieser Punkt wird durch die Gleichungen (9.50), (9.51) und (9.52) beschrieben. Subtrahieren wir die Gleichung (9.50) von der Gleichung (9.52), so ergibt sich das Gleichungssystem

$$(\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\omega} = \mathbf{u} \cdot \boldsymbol{\xi} + \xi_t \quad (9.59)$$

$$\mathbf{N}^\perp \cdot \boldsymbol{\omega} = \mathbf{N}^\perp \cdot \boldsymbol{\xi}. \quad (9.60)$$

Beide Gleichungen beschreiben jeweils eine Gerade im $\boldsymbol{\omega}$ -Raum, wobei die erste Gleichung in Richtung $(\mathbf{u} - \mathbf{v})^\perp$ verläuft und die zweite in Richtung \mathbf{N} . Unter der Voraussetzung, dass die beiden Vektoren nicht parallel sind, bilden sie eine Basis, anhand derer wir $\boldsymbol{\omega}$ bestimmen können. Im Folgenden nehmen wir an, die beiden Vektoren seien nicht parallel. Anderenfalls wäre das Residuum nach Gleichung (9.47) null und ließe sich mit den Mitteln der transparenten Bewegungsbestimmung behandeln. Damit ergeben sich die gesuchten $\boldsymbol{\omega}$'s als Linearkombination beider Geraden, d.h.

$$\boldsymbol{\omega} = x \mathbf{N} + y (\mathbf{u} - \mathbf{v})^\perp \quad (9.61)$$

mit den noch zu bestimmenden Unbekannten x und y . Die Anwendung von Gleichung (9.61) auf Gleichung (9.59) ergibt

$$\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t = \boldsymbol{\omega} \cdot (\mathbf{u} - \mathbf{v}) = y \mathbf{N} \cdot (\mathbf{u} - \mathbf{v}) \quad (9.62)$$

und daraus folgt

$$x = \frac{\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})}. \quad (9.63)$$

Die Anwendung von (9.61) auf (9.59) ergibt

$$\mathbf{N}^\perp \cdot \boldsymbol{\xi} = \mathbf{N}^\perp \cdot \boldsymbol{\omega} = y \mathbf{N}^\perp \cdot (\mathbf{u} - \mathbf{v})^\perp = y \mathbf{N} \cdot (\mathbf{u} - \mathbf{v}), \quad (9.64)$$

da eine Rotation von zwei Vektoren um 90 Grad keinen Einfluss auf ihr Skalarprodukt hat, damit ist y bestimmt durch

$$y = \frac{\mathbf{N}^\perp \cdot \boldsymbol{\xi}}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})}. \quad (9.65)$$

Setzen wir nun x und y in Gleichung (9.61) ein, so erhalten wir die gesuchten Schnittpunkte in Abhängigkeit von $\boldsymbol{\xi} + \mathbf{e}_t \xi_t$

$$\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t) = \frac{(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) \mathbf{N} + \mathbf{N}^\perp \cdot \boldsymbol{\xi} (\mathbf{u} - \mathbf{v})^\perp}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})}. \quad (9.66)$$

Damit erhalten wir für das Residuum (9.49)

$$\begin{aligned} R(\boldsymbol{\xi}, \xi_t) &= \frac{2\pi j}{d} (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} (\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) \int G_2(\boldsymbol{\omega}) \delta(\boldsymbol{\omega} - \boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t)) d\boldsymbol{\omega} \\ &= \frac{2\pi j (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} (\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t)}{|\mathbf{u} - \mathbf{v}| |\mathbf{N}|} G_2(\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t)) \\ &= 2\pi j \operatorname{sign}((\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}) (\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) G_2(\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t)) \end{aligned} \quad (9.67)$$

und das Fourierspektrum $C(\boldsymbol{\xi}, \xi_t)$ (Gleichung (9.32)) des Störterms, in dem wir obige Gleichung durch den Ausdruck $(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) (\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)$ teilen, also

$$C(\boldsymbol{\xi}, \xi_t) = \frac{\operatorname{sign}((\mathbf{u} - \mathbf{v}) \cdot \mathbf{N})}{2\pi j (\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t)} G_2(\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t)). \quad (9.68)$$

Interessanterweise ist auch in diesem Fall das Profil des Störterms hyperbolisch entlang der Geraden $\mathbf{N} - \mathbf{u} \cdot \mathbf{N} \mathbf{e}_t = \textit{konstant}$, wie wir jetzt zeigen werden.

Es genügt die Richtung dieser Geraden für einen Geradenpunkt zu bestimmen, denn die Richtung einer Geraden ist überall gleich. Wir wählen den Punkt $\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t) = 0$ und erhalten aus Gleichung (9.66)

$$(\mathbf{u} \cdot \boldsymbol{\xi} + \xi_t) \mathbf{N} = -\mathbf{N}^\perp \cdot \boldsymbol{\xi} (\mathbf{u} - \mathbf{v}). \quad (9.69)$$

Setzen wir $\boldsymbol{\xi} = \mathbf{N}$, so ist die rechte Seite null und die Gleichung lautet

$$(\mathbf{u} \cdot \mathbf{N} + \xi_t) \mathbf{N} = 0. \quad (9.70)$$

Da auch die linke Seite null ergeben muss, folgt $\xi_t = -\mathbf{u} \cdot \mathbf{N}$. Damit haben wir die Richtung von $\boldsymbol{\omega}_0(\boldsymbol{\xi}, \xi_t) = 0$ am Punkt $\boldsymbol{\xi} = \mathbf{N}, \xi_t = -\mathbf{u} \cdot \mathbf{N}$ bestimmt. Ferner gilt nach Gleichung (9.70) $\boldsymbol{\omega}_0(\mathbf{N}, -\mathbf{u} \cdot \mathbf{N}) = \boldsymbol{\omega}_0(\lambda \mathbf{N}, -\lambda \mathbf{u} \cdot \mathbf{N}) = 0$ für alle $\lambda \in \mathbb{R}$ und $\boldsymbol{\omega}_0$ ist

konstant in Richtung $\mathbf{N} - \mathbf{u} \cdot \mathbf{N} \mathbf{e}_t$.

Zum Abschluss dieses Abschnittes möchten wir zeigen, dass der Störterm dasselbe Profil wie das verdeckte Signal $G_2(\boldsymbol{\xi})$ entlang der Ebenen $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = c$ hat. Für eine Ebene mit $\xi_t = -\mathbf{v} \cdot \boldsymbol{\xi} + c$ gilt

$$\begin{aligned} \omega_0(\boldsymbol{\xi}, -\mathbf{v} \cdot \boldsymbol{\xi} + c) &= \frac{((\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\xi} + c) \mathbf{N} + \mathbf{N}^\perp \cdot \boldsymbol{\xi} (\mathbf{u} - \mathbf{v})^\perp}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})} \\ &= \frac{(\mathbf{u} - \mathbf{v}) \cdot \boldsymbol{\xi} \mathbf{N} + \mathbf{N}^\perp \cdot \boldsymbol{\xi} (\mathbf{u} - \mathbf{v})^\perp + c \mathbf{N}}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})} \\ &= \hat{\omega}_0(\boldsymbol{\xi}) + \frac{c \mathbf{N}}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})} \end{aligned} \quad (9.71)$$

wobei $\hat{\omega}_0(\boldsymbol{\xi}) = \omega_0(\boldsymbol{\xi}, -\mathbf{v} \cdot \boldsymbol{\xi})$ der Darstellung von ω_0 für die Ebene $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = 0$ entspricht. Der Ausdruck $\hat{\omega}_0(\boldsymbol{\xi})$ bleibt für alle anderen Parallelebenen gleich, hinzu kommt jedoch der Term $c \mathbf{N} / \mathbf{N} \cdot (\mathbf{u} - \mathbf{v})$. Vergleichen wir den Ausdruck $G_2(\omega_0(\boldsymbol{\xi}, -\mathbf{v} \cdot \boldsymbol{\xi} + c)) = G_2(\hat{\omega}_0(\boldsymbol{\xi}) + \frac{c \mathbf{N}}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})})$ der Ebene $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = c$ mit dem Ausdruck $G_2(\hat{\omega}_0(\boldsymbol{\xi}))$ entsprechend der Ebene $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = 0$, so fällt auf, dass das Profil der Spektralschnitte entlang der Ebenen unverändert bleibt, lediglich um den Anteil $\frac{c \mathbf{N}}{\mathbf{N} \cdot (\mathbf{u} - \mathbf{v})}$ verschoben. Nach Gleichung (9.68) gilt dies bis auf die Skalierung mit dem Faktor $\frac{\text{sign}((\mathbf{u} - \mathbf{v}) \cdot \mathbf{N})}{2\pi j c}$ auch für den Störterm. Zu beachten ist jedoch, dass die obige Argumentation über $G_2(\omega_0)$ sich nur auf die Betrachtung der Ebenen bezieht, nicht aber auf den Vergleich mit $G_2(\boldsymbol{\xi})$, und dass der Störterm für die Ebene $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = 0$ nicht definiert ist. Die Abbildung $\omega_0(\boldsymbol{\xi}, \xi_t)$ für eine dieser Ebenen ist zwar linear, könnte jedoch dazu führen, dass $G_2(\omega_0(\boldsymbol{\xi}, \xi_t))$ im Vergleich zu dem Originalspektrum $G_2(\boldsymbol{\xi})$ gedehnt oder gestaucht wird. Wir wollen jetzt zeigen, dass entlang dieser Ebenen, abgesehen von der Verschiebung, die Gleichheit gilt. Es genügt die Betrachtung für $\hat{\omega}_0$ durchzuführen, denn wie wir gezeigt haben, sind die Spektren mit jeder weiteren Parallelebene identisch aber verschoben. Wir überführen die lineare Transformation $\hat{\omega}_0(\boldsymbol{\xi})$ mittels der beiden Vektoren $\mathbf{e}_1 = (1, 0)^T$ und $\mathbf{e}_2 = (0, 1)^T$ in die *Standarddarstellung*

$$\begin{aligned} \mathbf{A} &= [\hat{\omega}_0(1, 0) \mid \hat{\omega}_0(0, 1)] \\ &= \frac{1}{(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}} \begin{bmatrix} (\mathbf{u} - \mathbf{v})_x \mathbf{N}_x + \mathbf{N}_x^\perp (\mathbf{u} - \mathbf{v})_x^\perp & (\mathbf{u} - \mathbf{v})_y \mathbf{N}_x + \mathbf{N}_y^\perp (\mathbf{u} - \mathbf{v})_x^\perp \\ (\mathbf{u} - \mathbf{v})_x \mathbf{N}_y + \mathbf{N}_x^\perp (\mathbf{u} - \mathbf{v})_y^\perp & (\mathbf{u} - \mathbf{v})_y \mathbf{N}_y + \mathbf{N}_y^\perp (\mathbf{u} - \mathbf{v})_y^\perp \end{bmatrix} \end{aligned} \quad (9.72)$$

Setzen wir $\mathbf{N}^\perp = (-\mathbf{N}_y, \mathbf{N}_x)^T$ und $(\mathbf{u} - \mathbf{v})^\perp = (-(\mathbf{u} - \mathbf{v})_y, (\mathbf{u} - \mathbf{v})_x)^T$, so vereinfacht sich die Standardform zu

$$\mathbf{A} = \frac{1}{(\mathbf{u} - \mathbf{v}) \cdot \mathbf{N}} \begin{bmatrix} (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} & 0 \\ 0 & (\mathbf{u} - \mathbf{v}) \cdot \mathbf{N} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (9.73)$$

Da es sich bei der Transformationsmatrix \mathbf{A} um die Einheitsmatrix handelt, haben wir nachgewiesen, dass $\hat{\omega}_0(\boldsymbol{\xi}) = \boldsymbol{\xi}$ ist und damit auch, dass der Störterm $\mathbf{C}(\boldsymbol{\xi}, \xi_t)$ entlang der Ebenen $\mathbf{v} \cdot \boldsymbol{\xi} + \xi_t = c$ das identische aber gewichtet und verschobene Profil hat wie das Spektrum $G_2(\boldsymbol{\xi})$.

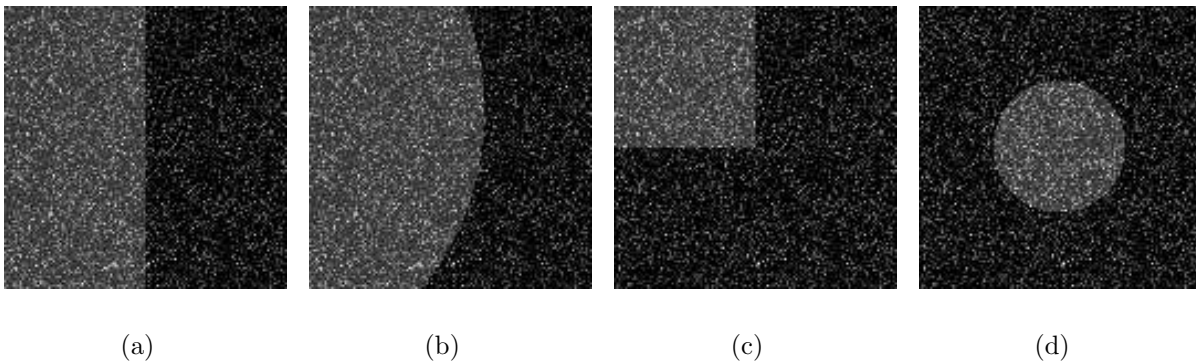


Abbildung 9.1: Veranschaulicht Formen verschiedener Objekte mit Rauschtexturen (heller Bereich), die sich vor einem Hintergrund (dunkler Bereich) bewegen, der ebenfalls eine Rauschtextur trägt. Die Überlagerung ist entweder additiv oder verdeckend.

9.5 Experimentelle Verifikation

Anhand von experimentellen Resultaten möchten wir die gefundenen Ergebnisse verifizieren. Für die Analyse nutzen wir synthetische Beispiele, um definierte Ausgangssituationen zu gewährleisten.

Abbildung 9.1 zeigt die verwendeten Formen der Vorder- und Hintergrundobjekte. Der Radius des kleinen Kreises beträgt 30 und der des großen Kreises 128 Bildpunkte. In allen Testszenarien bewegen sich sowohl das Vordergrund- als auch das Hintergrundobjekt. Die Richtungen und Geschwindigkeiten der Bewegungen werden in den Abbildungen 9.2 und 9.3 angegeben. Die Sequenzen tragen eine Rauschtextur als Muster, die hellere Untermahlung in den Abbildungen dient nur zur besseren Visualisierung. Zur Simulation realer Sequenzen sind die Sequenzen mit einem Signalrauschverhältnis von 30dB verrauscht worden. Die Testsequenzen besitzen in allen drei Dimensionen eine Größe von 128 Pixel und wurden vor der Fouriertransformation mit einem Hanning-Fenster gefenstert. Um die Auswirkungen der Verdeckungen deutlicher hervorzuheben, sind in allen Abbildungen die Differenzen zwischen den verdeckten und den transparenten Spektren bei gleichen Bewegungen dargestellt. Nach Gleichung (9.32) entspricht die verschmiertere Ebene jeweils dem Spektrum der verdeckten Schicht. Deutlich zu erkennen ist dieser Effekt in den Abbildungen 9.2(c), (b) und (d) sowie 9.4(b), (c) und (f). Zur besseren Visualisierung sind die Amplitudenspektren in einer logarithmischen Skala dargestellt.

Abbildung 9.2 zeigt die ermittelten Ergebnisse für den Fall einer geradlinig verlaufenden Verdeckungsgrenze entsprechend Abbildung 9.1(a). Der Vergleich von (a) und (b) verdeutlicht, dass nach Gleichung (9.16) transparente und verdeckende Bewegungen äquivalent sind, falls die relative Bewegung der Schichten senkrecht zu der Normalen der Verdeckungsgrenze ist und damit der Störterm verschwindet. Die relative Bewegung bezeichnet die Differenz zwischen den Bewegungsvektoren der verdeckenden und verdeckten Schicht. In Grafik (c) ist die Störung für die geradlinig verlaufende Verdeckungsgrenze am stärksten, da die relative Bewegung senkrecht zur Verdeckungsgrenze verläuft. In (d) wird

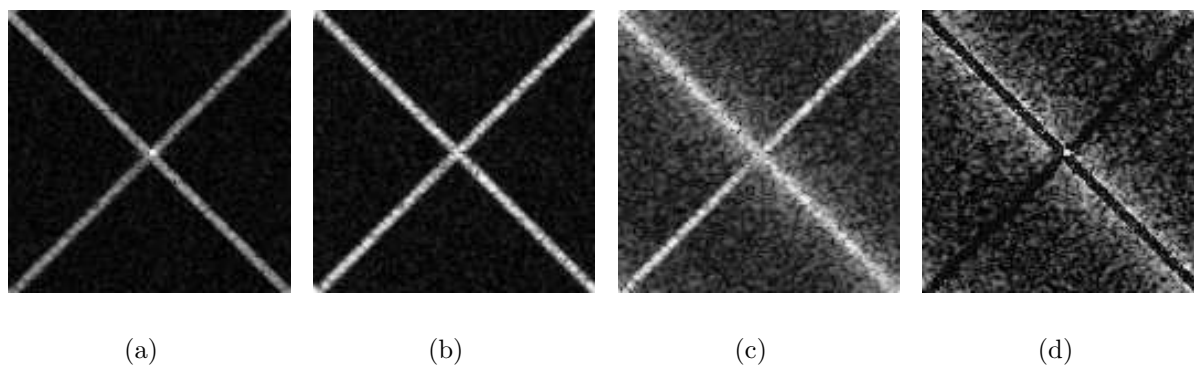


Abbildung 9.2: Ergebnisse bei geradliniger Verdeckungsgrenze: Dargestellt sind die Amplitudenspektren ($\xi_t = 64$) für (a) eine transparente Sequenz mit den Geschwindigkeiten $\mathbf{u} = (1, 1)$ und $\mathbf{v} = (-1, 1)$; (b) eine verdeckende Sequenz mit geradliniger Verdeckungsgrenze und den Geschwindigkeiten $\mathbf{u} = (1, 1)$ für den Vordergrund und $\mathbf{v} = (1, -1)$ für den Hintergrund; (c) eine verdeckende Sequenz mit geradliniger Verdeckungsgrenze und den Geschwindigkeiten $\mathbf{u} = (1, 1)$ und $\mathbf{v} = (-1, 1)$; (d) Differenz zwischen (a) und (c). Zu beobachten ist, dass der durch die Verdeckung hervorgerufene Störterm von der relativen Bewegung und der Normalen der Verdeckungsgrenze abhängt. Dargestellt sind die Extreme (b) keine Störung und (c) maximale Störung. Ebenfalls wird in (d) der hyperbolische Verlauf des Störterms deutlich.

durch Differenzbildung von (a) und (c) die Auswirkung des Störterms veranschaulicht.

Abbildung 9.3 zeigt in Analogie zu Abbildung 9.2(b)-(d) die Ergebnisse für die gekrümmten Verdeckungsgrenzen (siehe Abbildung 9.1 (b)). Im Unterschied zu Abbildung 9.2(b) ist in 9.3(b) nur eine Störung zu beobachten. Wie Abbildung (c) deutlich macht, scheint der Störeinfluss zu steigen, wenn die relative Bewegung in etwa senkrecht zu der “gemittelten” Verdeckungsgrenze ist. Die durch die Verdeckung entstandene Störung ist noch einmal in (d) hervorgehoben. Vergleicht man Abbildung 9.2 (d) mit 9.3(d), lässt sich kein substantieller Unterschied zwischen geraden und gekrümmten Verdeckungsgrenzen beobachten.

Resultate für stark gekrümmte Verdeckungsgrenzen sind in Abbildung 9.4 dargestellt. Sie veranschaulichen, dass intrinsische 2D-Merkmale wie Punkte gute Merkmale für Tracking-Anwendungen sind, selbst im Falle von verdeckten Bewegungen – siehe Abbildung 9.1 (c) und (d). Für Ecken ist die Störung geringer als für geradlinige Verdeckungsgrenzen und variiert weniger mit der Richtung der Bewegung, entsprechend dem Vergleich von Abbildung 9.2(c) und 9.3(c) mit 9.4(b) und (c). Für den kleinen Kreis in (f) gibt es keine Abhängigkeiten des Störterms von den relativen Bewegungen, da die Normalen der Verdeckungsgrenzen in alle Richtungen zeigen. Die durch die Verdeckungen verursachten Störungen für die Ecke und den kleinen Kreis sind in (d) bzw. (g) dargestellt.

Im Anschluss an die Fourieranalyse möchten wir die gefundenen Ergebnisse im Ort verifizieren. Abbildung 9.5(a) zeigt ein Bild einer Testsequenz mit einer geradlinig verlaufenden

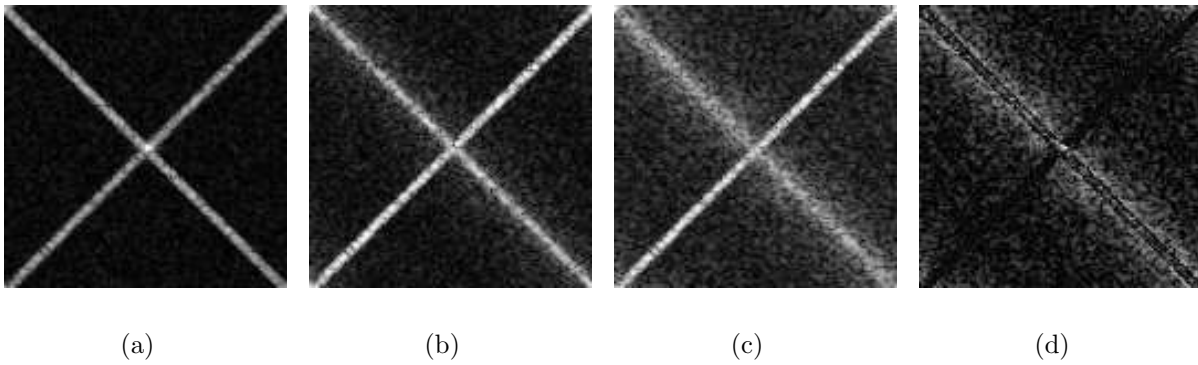


Abbildung 9.3: Ergebnisse bei gekrümmten Verdeckungsgrenzen: Dargestellt sind die Amplitudenspektren ($\xi_t = 64$) für (a) eine transparente Sequenz mit einem großen kreisförmigen Vordergrundobjekt, das sich mit der Geschwindigkeit $\mathbf{u} = (1, 1)$ vor einem mit der Geschwindigkeit $\mathbf{v} = (-1, 1)$ bewegten Hintergrund bewegt; (b) wie in (a) nur mit Verdeckung; (c) wie in (b) nur mit $\mathbf{v} = (1, -1)$; (d) Differenz zwischen (c) und (a). Maximale und minimale Störung sind ebenfalls wieder beobachtbar.

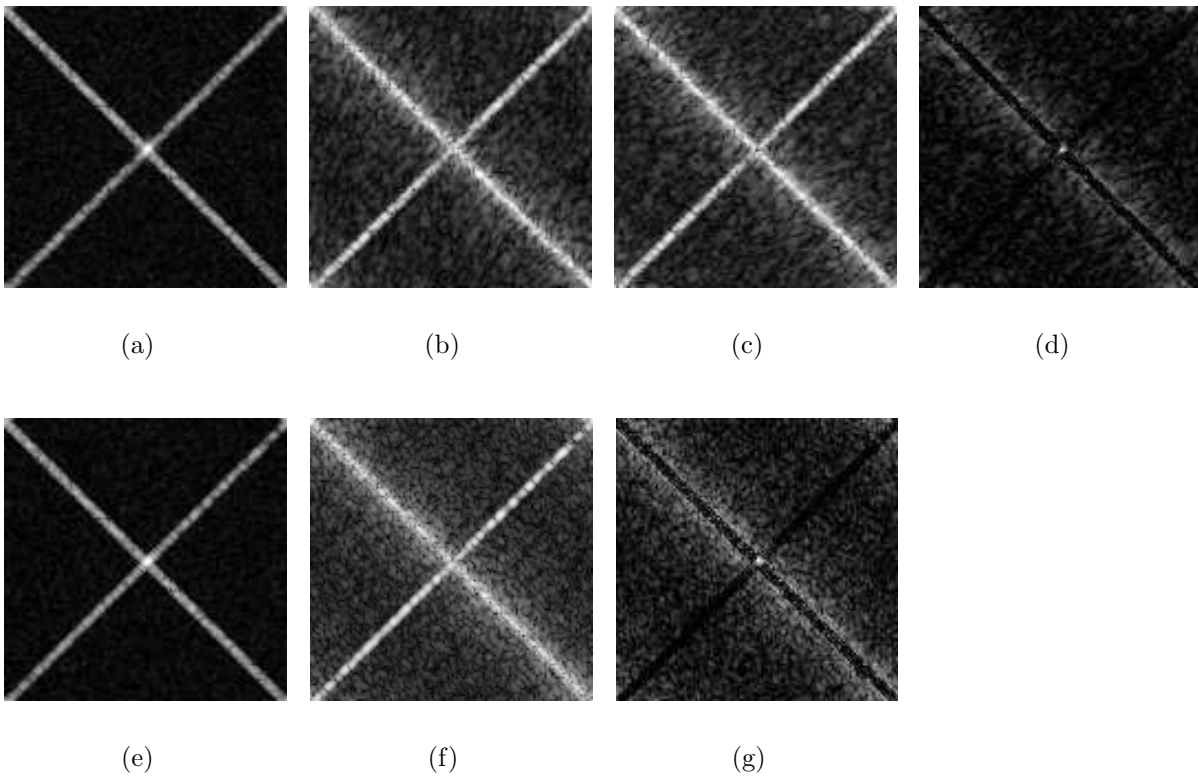


Abbildung 9.4: Amplitudenspektren ($\xi_t = 64$) für (a) eine transparente Sequenz mit einem Eckenformigen Vordergrundobjekt, mit den Geschwindigkeiten $\mathbf{u} = (1, 1)$ und $\mathbf{v} = (1, -1)$; (b) wie in (a) nur mit Verdeckung; (c) wie in (b) nur mit $\mathbf{v} = (-1, 1)$; (d) Differenz zwischen (c) und (a); (e) transparente Sequenz mit einem kleinen kreisförmigen Vordergrundobjekt und den Geschwindigkeiten $\mathbf{u} = (1, 1)$ und $\mathbf{v} = (-1, 1)$; (f) wie in (e) nur mit Verdeckung; (g) Differenz zwischen (f) und (e).

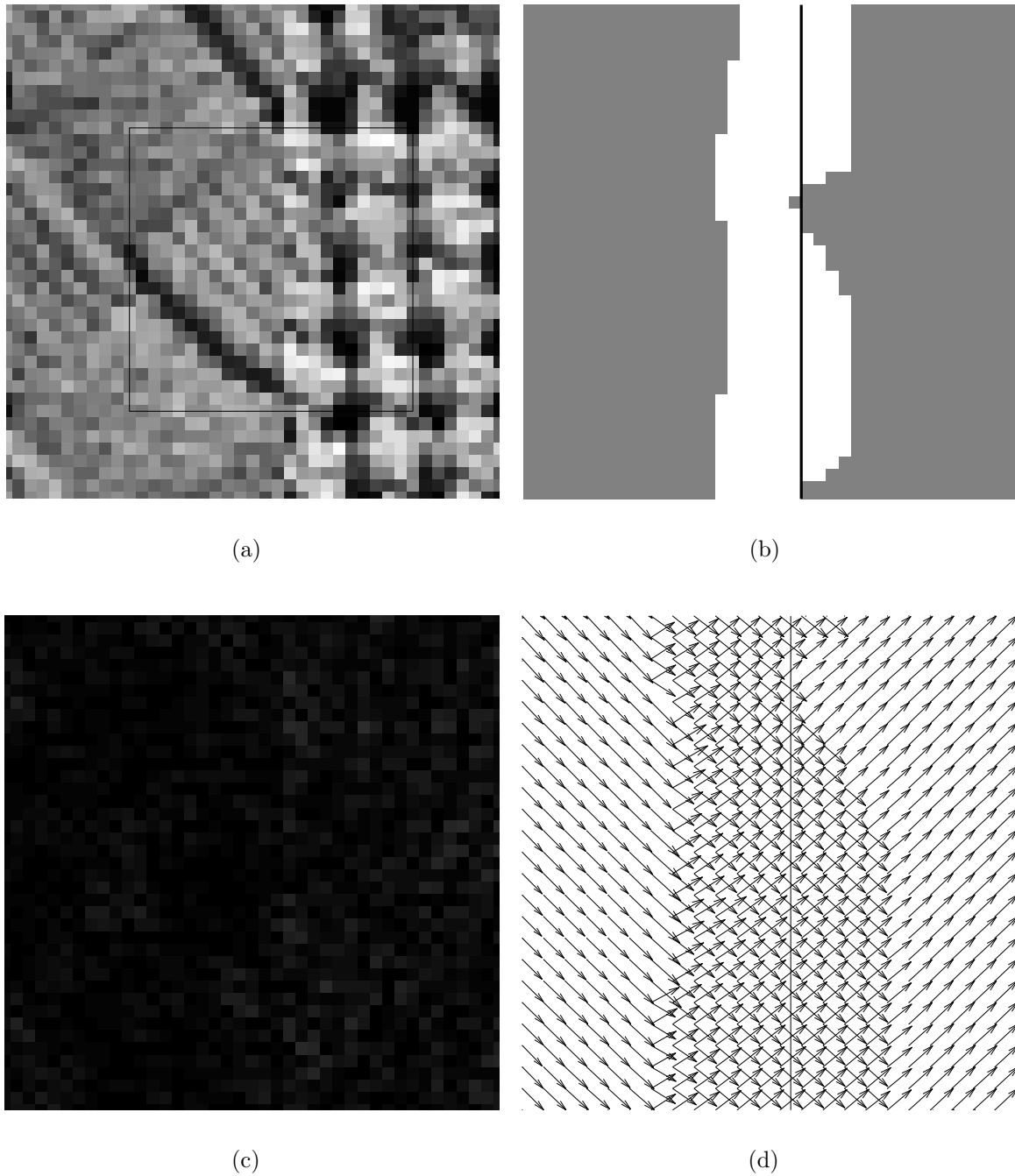


Abbildung 9.5: Beispiel für eine geradlinig verlaufende Verdeckungsgrenze im Ort: (a) Sequenz mit den Bewegungen $\mathbf{u} = (1, -1)^T$ und $\mathbf{v} = (1, 1)^T$, (b) Anzahl gefundener Bewegungen farbkodiert, (c) Werte der erweiterten optischen Flussgleichung und (d) ermittelte Bewegungsvektoren.

Verdeckungsgrenze, für die es die Bewegungen zu bestimmen gilt. Vorder- und Hintergrundobjekt unterscheiden sich in ihrer Textur und bewegen sich mit den Geschwindigkeiten $\mathbf{u} = (1, -1)^T$ und $\mathbf{v} = (1, 1)^T$. Grafik (b) gibt die Anzahl der mit dem hierarchischen Strukturtenorverfahren ermittelten Bewegungen für jeden Bildpunkt farbkodiert an: schwarz entspricht keiner, grau einer und weiss zwei Bewegungen. Wie die Segmentierung (b) verdeutlicht, wurden in einem Bereich um die Verdeckungsgrenze herum zwei Bewegungen ermittelt. Die berechneten Bewegungsvektoren für den in (a) markierten Ausschnitt sind in Abbildung (d) dargestellt. Bild (c) zeigt das Ergebnis der erweiterten optischen Flussgleichung ($\alpha(\mathbf{u})\alpha(\mathbf{v})f(\mathbf{x}, t) = 0$). An dem Grenzübergang beider Schichten ist kein Störeinfluss der Verdeckungsgrenze beobachtbar und das additive Bewegungsmodell eignet sich in Analogie zu den Ergebnissen im Frequenzraum unter dieser Bewegungsvektorkombination zur Bestimmung der Bewegungen an der Verdeckungsgrenze.

Die Ergebnisse für die gleiche Bildsequenz mit den Geschwindigkeiten $\mathbf{u} = (-1, 1)^T$ und $\mathbf{v} = (1, 1)^T$ für Vorder- und Hintergrundobjekte zeigt Abbildung 9.6. Wie in Grafik (a) und (c) zu erkennen ist, konnten in einem Bereich um die Verdeckungsgrenze herum keine Bewegungen ermittelt werden, da die Konfidenzkriterien dort nicht erfüllt sind. Die Konfidenzschranken lagen in beiden Experimenten bei $c_1 = 0.45$ und $c_2 = 0.55$. Die Ursache dafür liegt in dem Störterm, der beim Auswerten der erweiterten optischen Flussgleichung in Bild (b) deutlich zu erkennen ist. In (d) ist im Unterschied zu (c) die Bestimmung von zwei Bewegungen in dem Bereich um die Verdeckungsgrenze herum erzwungen worden. Die gefundenen Bewegungsvektoren stimmen in diesem Fall schlecht mit den wahren Bewegungen an der Verdeckungsgrenze überein. In beiden Beispielen sind die partiellen Ableitungen durch Anwenden der Prewitt Operatoren ermittelt worden und für die Integration der Tensoreinträge diente ein 7×7 Bildpunkte umfassender Mittelwertfilter.

9.6 Zusammenfassung

Die Bewegungsbestimmungsproblematik an Verdeckungsgrenzen ist in der Vergangenheit intensiv untersucht worden. Im Rahmen dieser Arbeit ergänzten wir die früheren Erkenntnisse um einen fehlenden Aspekt: eine Gleichung, die sowohl die Bewegungen an der Verdeckungsgrenze als auch den Ausdruck für den durch die Verdeckung hervorgerufenen Störterm beschreibt. Mit der Analyse im Ortsraum konnten wir zeigen, welche exakten Störungen zu erwarten sind, wenn wir das transparente Bewegungsmodell zur Bestimmung der Bewegungen an der Verdeckungsgrenze nutzen. Dieses Ergebnis bildete die Basis für die anschließende Analyse im Frequenzbereich.

In vorangegangenen Analysen [20, 21, 189, 190, 187] konnte für geradlinige Verdeckungsgrenzen ein hyperbolischer Verlauf des Störterms im Frequenzbereich nachgewiesen werden. Hier konnten wir zeigen, dass dieses Verhalten nicht nur auf geradlinig verlaufende Verdeckungsgrenzen beschränkt ist, sondern auch für gekrümmte Verdeckungsgrenzen gilt. Weiter konnten wir hier sogar den exakten Ausdruck für den Störterm im Falle geradlinig verlaufender Verdeckungsgrenzen angeben. Entlang bestimmter Parallelebenen im Frequenzbereich besitzt der Störterm das identische aber mit einer hyperbolischen Funktion gewichtete und verschobene Profil wie das Spektrum des verdeckten Signals.

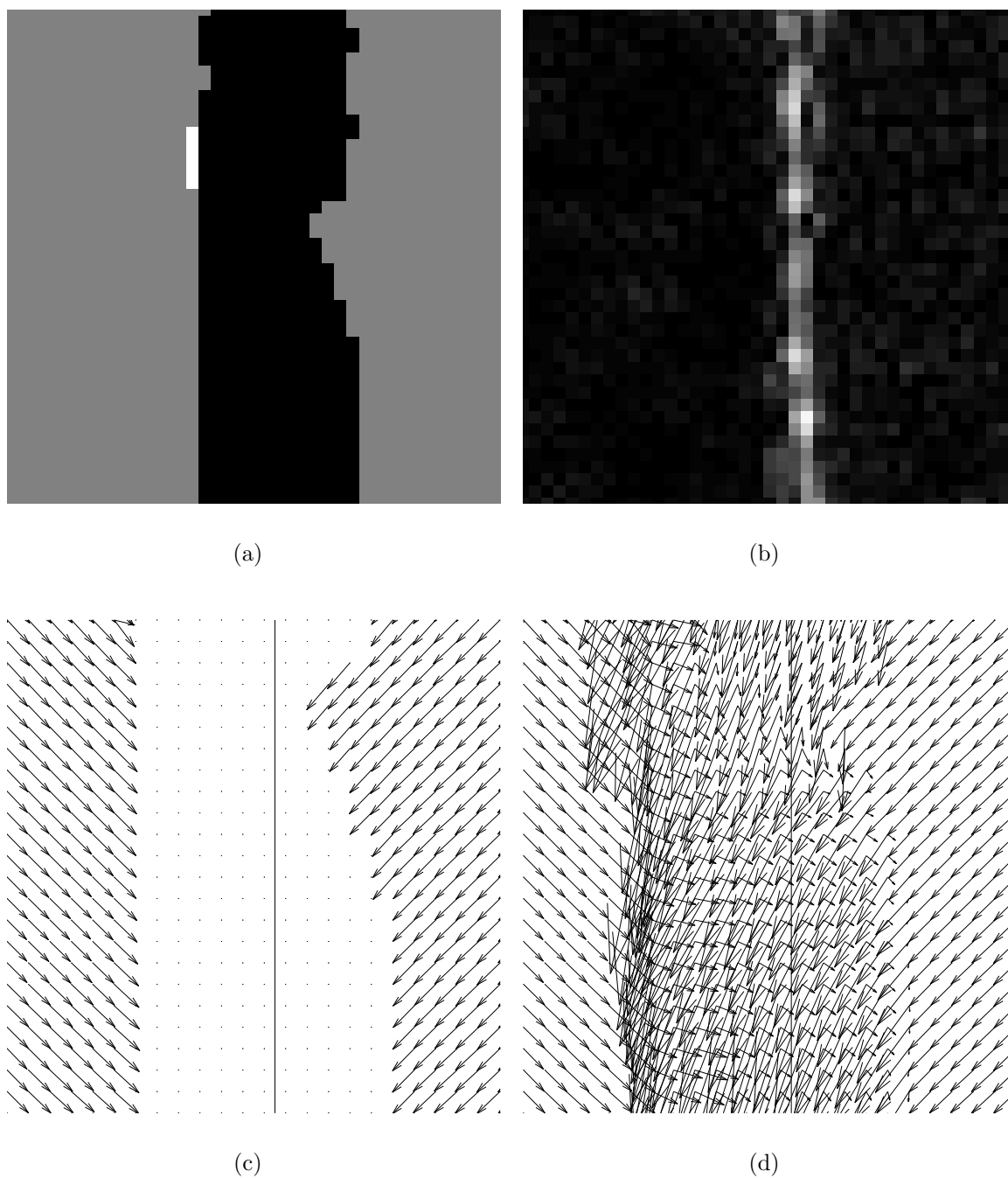


Abbildung 9.6: Ergebnisse für dieselbe Sequenz aus dem vorangegangenen Beispiel, jetzt aber mit den Geschwindigkeiten $\mathbf{u} = (-1, 1)^T$ und $\mathbf{v} = (1, 1)^T$: (a) Anzahl gefundenen Bewegungen farbkodiert, (b) Werte der erweiterten optischen Flussgleichung, (c) ermittelte Bewegungsvektoren und (d) Resultat wenn die Bestimmung von zwei Bewegungen an der Verdeckungsgrenze erzwungen wird.

Anhand synthetischer Beispiele bestätigten wir die theoretischen Ergebnisse. So konnten wir auch experimentell nachweisen, dass verdeckende Bewegungen unter bestimmten Voraussetzungen nur zwei Ebenen im Frequenzbereich besetzen und sich damit wie das transparente Bewegungsmodell verhalten. Die Bestimmung der Bewegungen an der Verdeckungsgrenze ist dann ohne weitere Einschränkungen mit dem transparenten Modell möglich. Als Voraussetzung dafür muss allerdings die relative Bewegung orthogonal zu der Normalen der Verdeckungsgrenze sein. Weiter zeigen die Experimente, dass der Störterm für gerade und gekrümmte Verdeckungsgrenzen ähnliche Formen aufweist, die Störungen für gekrümmte Grenzen jedoch etwas kleiner ausfallen, da sich die Normalen innerhalb des Integrationsbereiches verändern. Ferner scheint die Bewegungsbestimmung bei gekrümmten Verdeckungsgrenzen und Ecken besser möglich zu sein, da der Störterm in diesen Fällen kleiner ausfällt.

Wie die Ergebnisse zeigen, werden Verfahren, die auf Superpositionsmodellen oder einer Ebenenbestimmung im Fourierraum basieren, stark von der relativen Bewegung und der Form der Verdeckungsgrenze beeinflusst. Auch in dem speziellen aber häufigen Fall eines stationären Hintergrundes tritt ein Störterm auf, dessen Einfluß jedoch nur von der Richtung der Bewegung relativ zur Orientierung der Verdeckungsgrenze abhängt.

Kapitel 10

Überlagerte Orientierungen in Bildern

Die Orientierungsanalyse in Bildern aber auch in mehrdimensionalen Daten ist die Grundlage für viele Anwendungen, zum Beispiel dem richtungsabhängigen Filtern [3, 5], der richtungsabhängigen Interpolation [83], der Merkmalsextraktion für die Mustererkennung [92, 95], dem Konzept der intrinsischen Dimension [192] und zur Detektion von eindimensionalen Strukturen wie Kanten oder Linien [25, 41].

Die lokalen Orientierungen in einem Bild sind als die Richtung definiert, in denen die Grauwerte innerhalb einer kleinen Nachbarschaft am wenigsten variieren. Sie können zum Beispiel mit steuerbaren richtungsselektiven Filtern [107, 188, 152] oder Tensoren [57, 25, 92, 95, 97, 58] ermittelt werden. Bei den Tensorverfahren sind die Grauwertvariationen entlang und senkrecht zu der Orientierung durch die Eigenwerte beschrieben, während die korrespondierenden Eigenvektoren ihre Richtungen angeben. Die Elemente des Tensors werden durch Filterung der beobachteten Bilddaten berechnet, nämlich durch Differenziation oder durch Quadraturfilter und nichtlineare Operatoren wie zum Beispiel dem Quadrieren [92, 95, 26, 74, 98, 66]. Diese Ansätze setzen aber das Vorhandensein einer Orientierung voraus. Im Falle von zwei oder mehr überlagerten Orientierungen repräsentieren die Eigenvektoren nicht mehr die Orientierungen im Bild. Solche Nachbarschaften, zum Beispiel durch das Vorhandensein einer Ecke entstanden, werden durch große Werte des kleineren Eigenwertes charakterisiert und verdeutlichen, dass keine Achsenrichtung mit geringer Grauwertvarianz gefunden werden kann. Mehrfache Orientierungen entstehen zum Beispiel durch eine Überlagerung von durchsichtigen Strukturen, bei Röntgenaufnahmen oder durch Ecken, Kreuzungen, Bifurkationen und bei gewebten Strukturen mit mehreren Orientierungen. Sie lassen sich nicht durch die Annahme einer einzelnen Vorzugsrichtung beschreiben.

Da Ecken und sogenannte Junctions [7] reichhaltige Informationen für das Bild- und Szenenverständnis liefern, ist ihre Detektion und ihre Analyse für viele Bildverarbeitungsanwendungen relevant und dementsprechend auch Forschungsgegenstand vieler wissenschaftlicher Arbeiten [129, 152, 132, 98, 76, 65, 77, 87, 57]. Abbildung 10.1 (a) zeigt einige der wichtigsten Junctionstypen, die in dem Beispielobjekt (b) auftreten, und den damit verbundenen Zusammenhang zur Mehrfachorientierung. Die Namensgebung der Junctions

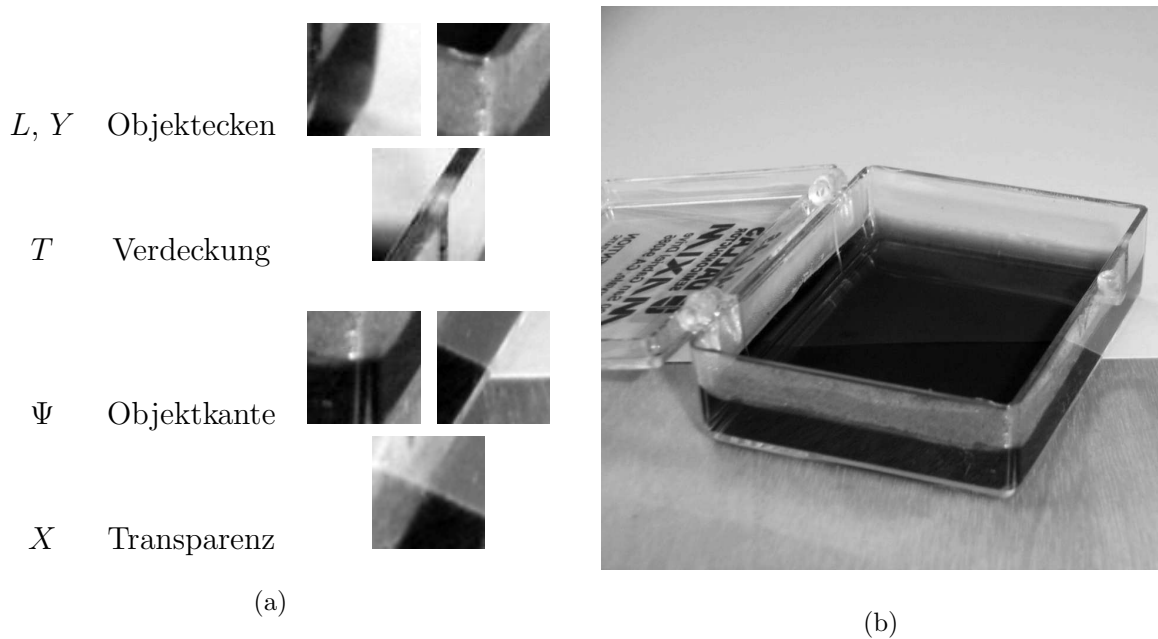


Abbildung 10.1: Klassifikation von Junctions

orientiert sich in der Regel an den Buchstaben und Symbolen, die die Anordnung und die Form der Flächen am Besten beschreiben. T -Junctions sind häufig mit Objektverdeckungen assoziiert, L - und Y -Junctions mit Objektecken, X -Junctions mit der Erscheinung von Transparenz und Ψ -Junction mit gebogenen Flächen [115].

Wie die Diskussion zeigt, ist die Detektion und Analyse von Orientierungen für viele Anwendungen von hoher Bedeutung. Aus diesem Grund möchten wir im Folgenden ein Konzept vorstellen [160, 6, 115], das es erlaubt, mehrfache lokale Orientierungen zu bestimmen. Nach einer kurzen Einführung in die einfache Orientierungsbestimmung formulieren wir das Problem der additiven Überlagerung, ausgehend von dem Konzept der transparenten Bewegungen und mehrfachen Orientierungsfelder [146]. Wie bei den transparenten Bewegungen ermitteln wir durch eine Eigensystemanalyse eines entsprechenden Tensors die gemischten Orientierungsparameter als implizite Darstellung der Orientierungen. Ist explizit nach den Orientierungen gefragt, so erfordert dies eine Zerlegung der gemischten Orientierungsparameter in seine individuellen Bestandteile. Für die Separierung von zwei Orientierungen aus den gemischten Orientierungsparametern werden wir drei unterschiedliche Varianten vorstellen. Die ursprünglich für die Bewegungen entworfenen Konfidenzkriterien aus Kapitel 3 gelten auch für die Orientierungsanalyse und erlauben die Angabe eines hierarchischen Verfahrens zum Test auf keine, eine und zwei Orientierungen. Anschließend stellen wir ein weiteres alternatives Orientierungsmodell für verdeckende Orientierungen vor [115], das nur mit partiellen Ableitungen erster Ordnung auskommt und sich für die Orientierungsanalyse von L und T -Junctions eignet. X -Junctions entsprechen eher dem additiven Modell [9, 7].

10.1 Einfache Orientierung

Um die Orientierungen in Bildern bestimmen zu können, benötigen wir ein entsprechendes Bildmodell aus dem sich die Orientierungsbedingungen ableiten. Wir bezeichnen ein Grauwertbild $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ in einer offenen Region Ω als ideal orientiert, wenn ein eindimensionaler Unterraum E des \mathbb{R}^2 existiert, so dass

$$f(\mathbf{v} + \mathbf{x}) = f(\mathbf{x}) \quad (10.1)$$

für alle \mathbf{x}, \mathbf{v} mit $\mathbf{x}, \mathbf{x} + \mathbf{v} \in \Omega, \mathbf{v} \in E$ ist. Anders formuliert handelt es sich um eine orientierte Struktur, wenn nur eine Richtung existiert, in der die Grauwerte konstant sind. Entsprechend verschwindet die Richtungsableitung nur entlang dieser Richtung und wir können obige Formulierung äquivalent als

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{v}} = \alpha(\theta_1) f(\mathbf{x}) = 0 \quad \text{für alle } \mathbf{x} \in \Omega, \mathbf{v} \in E \quad (10.2)$$

schreiben [66, 26], mit $\mathbf{v} = (v_x, v_y)^T = (\cos \theta, \sin \theta)^T$ und dem Richtungsableitungsoperator $\alpha(\theta_1) = \cos \theta_1 \partial_x + \sin \theta_1 \partial_y = \mathbf{v} \cdot \nabla$. Da diese Gleichung der optischen Flussgleichung sehr ähnlich ist, ist die Vorgehensweise zur Bestimmung des Bewegungsvektors mittels des Tensorsansatzes auf die Orientierungsbestimmung übertragbar. Wie bei dem Bewegungsfall ermitteln wir den gesuchten Orientierungsvektor durch Minimieren des Fehlerfunktional

$$E(\mathbf{v}) = \int_{\Omega} (\alpha(\theta_1) f)^2 \mathbf{d}\Omega = \int_{\Omega} (\mathbf{v} \cdot \nabla f)^2 \mathbf{d}\Omega. \quad (10.3)$$

Unter der Annahme, dass der Orientierungsvektor \mathbf{v} in der Region Ω konstant ist, können wir obiges Fehlerfunktional als

$$E(\mathbf{v}) = \mathbf{v}^T \mathbf{J} \mathbf{v} \quad (10.4)$$

mit

$$\mathbf{J}(f) = \int (\nabla f)(\nabla f)^T \mathbf{d}\Omega \quad (10.5)$$

schreiben. In Analogie zu dem Strukturtenorverfahren aus Kapitel 3 entspricht der gesuchte Orientierungsvektor dem Eigenvektor zum kleinsten Eigenwert. Orientierung liegt nur dann vor, wenn ein Eigenwert von \mathbf{J} null und der andere von null verschieden ist. Sind beide Eigenwerte gleich null, liegt eine homogene Grauwertstruktur vor, sind beide Eigenwerte ungleich null, ändern sich die Grauwerte in allen Richtungen. Letztere Situation tritt beispielsweise auf, wenn in der betrachteten Region eine Ecke, mehrere Orientierungen oder eine transparente Überlagerung orientierter Muster beobachtet wird.

10.2 Transparente Orientierungen

Setzt sich das Bild f aus einer additiven (transparenten) Überlagerung zweier ideal orientierter Bilder g_1 und g_2 mit

$$\alpha(\theta_1)g_1 = 0 \quad \text{und} \quad \alpha(\theta_2)g_2 = 0, \theta_1 \neq \theta_2 \quad (10.6)$$

zusammen, d.h. $f(\mathbf{x}) = g_1(\mathbf{x}) + g_2(\mathbf{x})$, so gilt auch hier in Analogie zum dreidimensionalen Fall das Richtungsableitungsoperatorkonzept von Shizawa & Mase [147, 148]

$$\alpha(\theta_1)\alpha(\theta_2)f = f_{xx}c_{xx} + f_{xy}c_{xy} + f_{yy}c_{yy} = 0, \quad (10.7)$$

mit den gemischten Orientierungsparametern

$$\begin{aligned} c_{xx} &= u_x v_x = \cos\theta_1 \cos\theta_2, \\ c_{xy} &= u_x v_y + u_y v_x = \sin(\theta_1 + \theta_2), \\ c_{yy} &= u_y v_y = \cos\theta_1 \cos\theta_2, \end{aligned} \quad (10.8)$$

die sich aus den Komponenten der Orientierungsvektoren $\mathbf{u} = (u_x, u_y)^T = (\cos\theta_1, \sin\theta_1)^T$ und $\mathbf{v} = (v_x, v_y)^T = (\cos\theta_2, \sin\theta_2)^T$ ergeben. Äquivalent ist Gleichung (10.7) als inneres Produkt $\mathbf{d}f \cdot \mathbf{c} = 0$ der Vektoren

$$\mathbf{d}f = (f_{xx}, f_{xy}, f_{yy})^T \quad \text{und} \quad \mathbf{c} = (c_{xx}, c_{xy}, c_{yy})^T \quad (10.9)$$

darstellbar und der gesuchte gemischte Orientierungsvektor \mathbf{c} ergibt sich, wie bei der einfachen Orientierung, aus dem Eigenvektor korrespondierend zu dem kleinsten Eigenwert von

$$\mathbf{J}_2 = \int (\mathbf{d}f)(\mathbf{d}f)^T d\Omega. \quad (10.10)$$

Indirekt repräsentiert der gemischte Orientierungsvektor \mathbf{c} die in dem Bild vorkommenden Orientierungen und könnte zum Beispiel als ein Merkmalsvektor für Klassifikationsaufgaben dienen. Ist jedoch explizit nach den Orientierungen gefragt, so erfordert dies eine Zerlegung des Vektors \mathbf{c} in die zugrunde liegenden Orientierungsvektoren \mathbf{u} und \mathbf{v} bzw. deren Winkel.

10.2.1 Separierung der Orientierungsvektoren

Die Eigenvektoranalyse des erweiterten Orientierungstensors \mathbf{J}_2 in Gleichung (10.10) ergibt den gemischten Orientierungsparametervektor \mathbf{c} , der bis auf einen unbekanntem Skalierungsfaktor definiert ist. Da per Konvention die Eigensystemanalyse die Eigenvektoren normalisiert, ist der gemischte Orientierungsparametervektor

$$\mathbf{c}^T = R(\cos(\theta_1)\cos(\theta_2), \sin(\theta_1 + \theta_2), \sin(\theta_1)\sin(\theta_2)) = (c_{xx}, c_{xy}, c_{yy}) \quad (10.11)$$

bis auf die Normalisierungskonstante R bestimmt. Für die Extraktion der Orientierungen θ_1 und θ_2 aus dem Vektor \mathbf{c} machen wir zunächst die Beobachtung

$$c_{xx} \mp c_{yy} = R \cos(\theta_1 \pm \theta_2), \quad c_{xy} = R \sin(\theta_1 + \theta_2). \quad (10.12)$$

Auf den ersten Blick scheint es, als könnten diese Ausdrücke invertiert werden um $\theta_1 + \theta_2$ und $\theta_1 - \theta_2$ zu erhalten. Die Schwierigkeit ist jedoch, dass selbst wenn θ_1 und θ_2 auf das Intervall $(-\pi/2, \pi/2]$ beschränkt sind, $-\pi < \theta_1 \pm \theta_2 \leq \pi$ gilt. Die Invertierung des Kosinus und Sinus in diesem Bereich ist leider nicht eindeutig und damit nicht für die Bestimmung der Orientierungen ausreichend.

Ein alternativer Ansatz, ähnlich wie bei der Bestimmung transparenter Bewegungen, besteht in der Suche nach den Nullstellen eines Polynoms, dessen Koeffizienten sich aus den Komponenten c_{xx} , c_{xy} und c_{yy} zusammensetzen. Nutzen wir

$$\begin{aligned}\frac{c_{xy}}{c_{xx}} &= \frac{\sin(\theta_1 + \theta_2)}{\cos(\theta_1)\cos(\theta_2)} = \tan(\theta_1) + \tan(\theta_2), \\ \frac{c_{yy}}{c_{xx}} &= \frac{\sin(\theta_1)\sin(\theta_2)}{\cos(\theta_1)\cos(\theta_2)} = \tan(\theta_1) \cdot \tan(\theta_2)\end{aligned}\quad (10.13)$$

als Koeffizienten für die Konstruktion des Polynoms

$$P(z) = z^2 - \frac{c_{xy}}{c_{xx}}z + \frac{c_{yy}}{c_{xx}} = (z - \tan(\theta_1))(z - \tan(\theta_2)), \quad (10.14)$$

so besitzt $P(z)$ die Nullstellen $\tan(\theta_1)$ und $\tan(\theta_2)$. Die Tangensfunktion kann eindeutig in dem Intervall $(-\pi/2, \pi/2]$ invertiert werden. Sonderbehandlungen sind im Falle der Singularität $c_{xx} = 0$ nötig. Ist $a = 0$, so bedeutet dies $\theta_1 = \pi/2$ oder $\theta_2 = \pi/2$.

Im Folgenden werden wir drei weitere Alternativen zur Separierung der Orientierungen aus den gemischten Orientierungsparametern vorstellen, die die obigen Schwierigkeiten umgehen.

Methode 1

Bei der ersten Methode handelt es sich um den schon mehrfach angesprochenen Ansatz von Shizawa & Mase [147, 148] bzw. deren explizite Darstellung für die Orientierungsanalyse in Bildern [87, 146]. Nach der Ermittlung des gemischten Orientierungsparametervektors werden dessen Parameter in der Matrix

$$C_s = \begin{pmatrix} c_{xx} & c_{xy}/2 \\ c_{xy}/2 & c_{yy} \end{pmatrix} \quad (10.15)$$

und dessen Eigenwerte λ_1 und λ_2 , ($\lambda_1 > \lambda_2$) sowie die korrespondierenden Eigenvektoren \mathbf{e}_1 und \mathbf{e}_2 ermittelt. Die gesuchten Orientierungsvektoren ergeben sich anschließend aus

$$\begin{aligned}\mathbf{u} &= \sqrt{\frac{\lambda_1}{\lambda_1 - \lambda_2}}\mathbf{e}_1 + \sqrt{\frac{-\lambda_2}{\lambda_1 - \lambda_2}}\mathbf{e}_2 \\ \mathbf{v} &= \sqrt{\frac{\lambda_1}{\lambda_1 - \lambda_2}}\mathbf{e}_1 - \sqrt{\frac{-\lambda_2}{\lambda_1 - \lambda_2}}\mathbf{e}_2\end{aligned}\quad (10.16)$$

Methode 2

Als weitere Alternative, die die oben genannten Probleme umgeht, stellen wir hier eine Separierungsmethode vor, die zunächst $(\cos(2\theta_1), \sin(2\theta_1))$ und $(\cos(2\theta_2), \sin(2\theta_2))$ als Schnittpunkte einer Geraden mit dem Einheitskreis findet und dann aus diesen die Orientierungen ermittelt.

Im ersten Schritt ersetzen wir die Variable z von $P(z)$ durch einer ihrer Nullstellen, die

wir hier mit y/x bezeichnen, wobei $x = \cos \theta$ und $y = \sin \theta$ ist. Gleichung (10.14) ist damit äquivalent zu dem Ausdruck

$$c_{yy}x^2 - c_{xy}xy + c_{xx}y^2 = 0. \quad (10.17)$$

Nun können wir zum Beispiels für den Winkel θ_1 beobachten, dass

$$\begin{aligned} x^2 &= \cos^2 \theta_1 = \frac{1}{2} + \frac{1}{2} \cos 2\theta_1, \\ y^2 &= \sin^2 \theta_1 = \frac{1}{2} - \frac{1}{2} \cos 2\theta_1, \\ xy &= \sin \theta_1 \cos \theta_1 = \frac{1}{2} \sin 2\theta_1 \end{aligned} \quad (10.18)$$

gilt und daher die beiden Punkte $P = (\cos 2\theta_1, \sin 2\theta_1)$ und $Q = (\cos 2\theta_2, \sin 2\theta_2)$ Lösungen der Geradengleichung

$$mX + nY + r = 0, \quad (10.19)$$

mit den Parametern

$$m = \frac{c_{yy} - c_{xx}}{\sqrt{1 - 2c_{xx}c_{yy}}}, \quad n = \frac{-c_{xy}}{\sqrt{1 - 2c_{xx}c_{yy}}}, \quad r = \frac{c_{xx} + c_{yy}}{\sqrt{1 - 2c_{xx}c_{yy}}}. \quad (10.20)$$

sind. Die Geradenparameter m, n wurden dabei so normalisiert, dass $m^2 + n^2 = 1$ ergibt. Sowohl Gleichung (10.19) als auch die Bedingung $|\mathbf{c}| = 1$ beinhalten zwei wichtige Aspekte: Auf der einen Seite implizieren sie, dass $1 - 2c_{xx}c_{yy}$ immer positiv ist und auf der anderen Seite, dass die Punkte P und Q immer Schnittpunkte der Geradengleichung mit dem Einheitskreis sind. Der Vektor $\mathbf{e}_1 = (m, n)$ entspricht dem Normalenvektor der Geraden, während $\mathbf{e}_2 = (-n, m)$ in Richtung der Gerade zeigt. Da r den Abstand der Geraden vom Ursprung entspricht, sind die Punkte P und Q durch die Linearkombination

$$P, Q = r\mathbf{e}_1 \mp s\mathbf{e}_2 \quad (10.21)$$

bestimmt, wobei sich $s = \sqrt{1 - r^2}$ nach dem Satz des Pythagoras berechnet. Abbildung 10.2 zeigt die geometrische Interpretation des Separierungsvorganges. Die Punkte P und Q können nun zur Berechnung der Orientierungswinkel gemäß $\theta_1 = \tan^{-1}(\sin(2\theta_1)/\cos(2\theta_1))/2$ und $\theta_2 = \tan^{-1}(\sin(2\theta_2)/\cos(2\theta_2))/2$ genutzt werden.

Methode 3

In der dritte Alternative bilden wir das dyadische Tensorprodukt der Vektoren \mathbf{u} und \mathbf{v} und erhalten die Matrix

$$\mathbf{M} = \mathbf{u} \otimes \mathbf{v} = \begin{pmatrix} u_x v_x & u_x v_y \\ u_y v_x & u_y v_y \end{pmatrix} = \begin{pmatrix} c_{xx} & z_1 \\ z_2 & c_{yy} \end{pmatrix}. \quad (10.22)$$

Die Zeilen der Matrix \mathbf{M} repräsentieren abgesehen von einem Skalierungsfaktor den Orientierungsvektor \mathbf{v} , d.h. $u_x \mathbf{v}^T$ in der ersten und $u_y \mathbf{v}^T$ in der zweiten Zeile. Umgekehrt kodieren die Spalten den Vektor \mathbf{u} , d.h. $v_x \mathbf{u}$ in der ersten und $v_y \mathbf{u}$ in der zweiten Spalte. Die Hauptdiagonalelemente von \mathbf{M} können direkt aus dem gemischten Orientierungsparametervektor \mathbf{c} entnommen werden. Zur vollständigen Aufstellung der Matrix müssen die noch

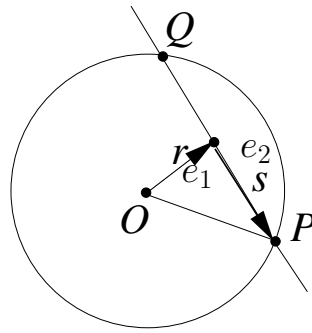


Abbildung 10.2: Geometrische Interpretation der Vektorseparierung.

fehlenden Nichtdiagonalelemente bestimmt werden. Mit der Beobachtung $z_1 z_2 = c_{xx} c_{yy}$ und $z_1 + z_2 = c_{xy}$ ergeben sich die gesuchten Elemente z_1 und z_2 als Nullstellen des Polynoms

$$p(z) = z^2 - c_{xy}z + c_{xx}c_{yy}, \quad (10.23)$$

dessen Koeffizienten aus den Elementen des gemischten Orientierungsvektors bestehen. Da beide Vektoren die Länge eins besitzen sollen, d.h. $\mathbf{u}^T \mathbf{u} = 1$ und $\mathbf{v}^T \mathbf{v} = 1$, ist

$$\mathbf{u} = \frac{(c_{xx}, z_2)^T}{\sqrt{c_{xx}^2 + z_2^2}} = \frac{(z_1, c_{yy})^T}{\sqrt{z_1^2 + c_{yy}^2}} \quad \text{und} \quad \mathbf{v} = \frac{(c_{xx}, z_1)^T}{\sqrt{c_{xx}^2 + z_1^2}} = \frac{(z_2, c_{yy})^T}{\sqrt{z_2^2 + c_{yy}^2}}. \quad (10.24)$$

Entsprechend bestimmen wir die Orientierungswinkel durch

$$\theta_1 = \arctan\left(\frac{z_2}{c_{xx}}\right) = \arctan\left(\frac{c_{yy}}{z_1}\right) \quad \text{und} \quad \theta_2 = \arctan\left(\frac{z_1}{c_{xx}}\right) = \arctan\left(\frac{c_{yy}}{z_2}\right), \quad (10.25)$$

die in dem Intervall von $(-\pi/2, \pi/2]$ liegen. Alternativ können wir auch beide Zeilen bzw. Spalten gleichzeitig mittels einer gewichteten Summe zur Bestimmung der Orientierungsvektoren nutzen und damit die numerische Stabilität erhöhen. Bezeichnen wir mit \mathbf{l}_1^T und \mathbf{l}_2^T die erste und die zweite Zeile der Matrix \mathbf{M} , so ist

$$\mathbf{v} = \frac{\|\mathbf{l}_1\| \mathbf{l}_1 + \|\mathbf{l}_2\| \mathbf{l}_2}{\|\mathbf{l}_1\|^2 + \|\mathbf{l}_2\|^2} \quad (10.26)$$

falls $\mathbf{l}_1^T \mathbf{l}_2 \geq 0$ gilt und anderenfalls

$$\mathbf{v} = \frac{\|\mathbf{l}_1\| \mathbf{l}_1 - \|\mathbf{l}_2\| \mathbf{l}_2}{\|\mathbf{l}_1\|^2 + \|\mathbf{l}_2\|^2}. \quad (10.27)$$

Das gleiche Vorgehen können wir auch für die Bestimmung des Vektors \mathbf{u} anwenden, wir müssen lediglich die Zeilenvektoren durch die Spaltenvektoren ersetzen.

10.2.2 Der Winkel zwischen beiden Orientierungsvektoren

Bei Doppelorientierungen kommt dem Winkel zwischen den beiden Vektoren eine besondere Rolle zu. Er eignet sich beispielsweise als ein Merkmal für Klassifikationsaufgaben.

Auch lässt sich mit ihm die relative Lage zweier orientierter und überlagerter Muster zueinander ermitteln. Sind die beiden Orientierungswinkel bestimmt, so ergibt sich der Winkel durch Differenzbildung. Alternativ können wir ihn auch direkt aus dem gemischten Orientierungsvektor \mathbf{c} bestimmen und damit auf die Berechnung der Orientierungswinkel verzichten. Der Winkel δ zwischen den zwei Vektoren $\mathbf{u} = a(\cos \theta_1, \sin \theta_1)^T$ und $\mathbf{v} = b(\cos \theta_2, \sin \theta_2)^T$ der Längen a bzw. b ergibt sich aus der Gleichung

$$\begin{aligned} \cos \delta &= \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \|\mathbf{v}\|} = \frac{a b (\cos \theta_1 \cos \theta_2 + \sin \theta_1 \sin \theta_2)}{a b} \\ &= \frac{(\cos \theta_1 \cos \theta_2 + \sin \theta_1 \sin \theta_2)}{\sqrt{\cos^2 \theta_1 + \sin^2 \theta_1} \sqrt{\cos^2 \theta_2 + \sin^2 \theta_2}} \\ &= \frac{c_{xx} + c_{yy}}{\sqrt{c_{xx}^2 + c_{xy}^2 + c_{yy}^2 - 2c_{xx}c_{yy}}}, \end{aligned} \quad (10.28)$$

die ihrerseits komplett durch die Komponenten des gemischten Orientierungsparametervektors $\mathbf{c} = (c_{xx}, c_{xy}, c_{yy})^T = (\cos \theta_1 \cos \theta_2, \sin(\theta_1 + \theta_2), \sin \theta_1 \sin \theta_2)^T$ darstellbar ist. Wegen $\cos \delta = -\cos(180^\circ - \delta)$ erhalten wir durch die Betragsbildung

$$\cos \delta = \frac{|c_{xx} + c_{yy}|}{\sqrt{c_{xx}^2 + c_{xy}^2 + c_{yy}^2 - 2c_{xx}c_{yy}}} \quad (10.29)$$

den spitzen Winkel, der von zwei Geraden in Richtung \mathbf{u} und \mathbf{v} eingeschlossen wird.

Alternativ kann der Winkel auch aus den beiden Eigenwerten λ_1 und λ_2 , mit $\lambda_1 > 0 > \lambda_2$, der Matrix (10.15) berechnet werden:

$$\cos \delta = \frac{|\lambda_1 + \lambda_2|}{\lambda_1 - \lambda_2}. \quad (10.30)$$

Schnell werden die Vorzüge des Winkels deutlich: Er ist eine skalare Größe, die die relative Lage zweier orientierter Muster zueinander beschreibt und skalierungs- und rotationsinvariant ist. Letztere Eigenschaft ist insbesondere dann interessant, wenn nur die relative Orientierung zwischen beiden Mustern benötigt wird, die konkrete Anordnung im Bild aber nur eine untergeordnete Rolle spielt. Somit eignet er sich als ein Merkmal für Vermessungs- und Klassifizierungsaufgaben.

10.2.3 Orientierung in Farbbildern

In vielen Anwendungen liegen die Bilder in Farbe vor. Zwar lassen sich Farbbilder in Grauwertbilder umrechnen und damit die bisher entwickelten Verfahren zur Orientierungsanalyse auch auf sie anwenden, durch die Berücksichtigung der Farbinformation können jedoch weitere Gleichungen gewonnen werden, die zu einer Verbesserung der Orientierungsanalyse führen können. Soll die Farbinformation in die Berechnung einfließen, erfordert dies eine Anpassung der Orientierungsanalyseverfahren.

Für Farb- bzw. Multispektralbilder mit

$$\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^q, \quad q > 1 \quad (10.31)$$

können wir unter der Annahme, dass die Orientierungsbedingung

$$\mathbf{c} \cdot \mathbf{d}f_i = 0 \quad (10.32)$$

für jeden Farbkanal f_i , $i = 1, \dots, q$ gilt, den Ansatz von Di Zenzo [191] verfolgen und den Strukturtensor für Farbbilder durch die Summe des grauwertigen Tensoren angewandt auf jeden Farbkanal

$$\mathbf{J}(\mathbf{f}) = \sum_{i=1}^q \mathbf{J}(f_i), \quad \mathbf{J}(f) = \int (\mathbf{d}f)(\mathbf{d}f)^T \mathbf{d}\Omega \quad (10.33)$$

darstellen. Diese Vorgehensweise ist für jede beliebige Anzahl additiv überlagerter orientierter Strukturen anwendbar und auch auf die Orientierungsanalyse im n -dimensionalen Raum oder die Bewegungsbestimmung übertragbar. Der farbwertige Tensor besitzt dieselben Eigenschaften wie der grauwertige: er ist positiv semidefinit und für ihn gelten auch die in Abschnitt 3.1.7 entwickelten Konfidenzkriterien. Folglich entspricht der Eigenvektor zum Eigenwert null von $\mathbf{J}(\mathbf{f})$ wieder dem gesuchten gemischten Orientierungsparametervektor.

10.2.4 Konfidenz und Invarianten

Orientierungen im Bild sind nur dann bestimmbar, wenn entweder der Tensor des einfachen oder des transparenten Modells einen einzigen Eigenwert gleich null besitzt. Um festzustellen, ob einer der beiden Tensoren einen Eigenwert gleich null besitzt, müssen erst alle Eigenwerte bestimmt werden. Die Tensoren \mathbf{J}_1 und \mathbf{J}_2 sind beide positiv semidefinit. Damit gelten die Konfidenzkriterien der Bewegungsbestimmung aus Satz 5 auch für die Orientierungsanalyse.

In Anhang A haben wir für die tensorbasierte Bewegungsbestimmung gezeigt, dass die Konfidenzmaße basierend auf den Tensorinvarianten, wie z.B. der Determinante und der Spur, rein durch die Unterdeterminanten ausgedrückt werden können. Ebenfalls ist bei Matrizen mit nur einem Eigenwert gleich null, der Eigenvektor zu diesem Eigenwert rein durch die Unterdeterminanten ausdrückbar. Die Verwendung der Unterdeterminanten zur Bestimmung der Konfidenzmaße sowie dem Eigenvektor ist besonders für 2×2 und 3×3 Matrizen interessant, da sich die Unterdeterminantenbestimmung bei diesen Matrizen einfach gestaltet und damit schnell und einfach implementieren lassen.

Der einfache Orientierungstensor \mathbf{J}_1 besitzt die beiden Invarianten:

$$\begin{aligned} K_1 &= \det(\mathbf{J}_1) = \lambda_1 \lambda_2 = J_{11}J_{22} - J_{12}^2 \\ H_1 &= \text{trace}(\mathbf{J}_1) = \lambda_1 + \lambda_2 = J_{11} + J_{22} \end{aligned}$$

Die Spur ist gleich null ($H_1 = 0$), wenn $\lambda_1 = \lambda_2 = 0$ gilt. Solche Situationen treten jedoch nur in homogenen Bereichen auf, in denen die Ableitungen verschwinden und keine Orientierung bestimmbar ist. Im Falle einer einfachen Orientierung übersetzt sich das Rangdefizientkriterium in $H_1 > 0$ und $K_1 = 0$. Nach Satz 5 ist $K_1 \leq \frac{1}{4}H_1^2$, wobei die Gleichheit nur in homogenen Bereichen erreicht wird. Diese Beobachtungen erlauben uns,

das Konfidenzkriterium als $K_1 \ll H_1^2$, oder äquivalent $K_1 \leq \epsilon_1 H_1^2$, zu schreiben, wobei $\epsilon_1 \leq \frac{1}{4}$ eine kleine Konstante ist, die sicherstellen soll, dass die Determinante nahe bei null liegt.

In Regionen, in denen das einfache Orientierungsmodell abgelehnt wurde, fahren wir mit dem Test auf doppelte Orientierungen fort. Der Orientierungstensor \mathbf{J}_2 besitzt die Invarianten

$$\begin{aligned} K_2 &= \det(J_2) = \lambda_1 \lambda_2 \lambda_3 = J_{11}M_{31} - J_{12}M_{32} + J_{13}M_{33} \\ S_2 &= (\lambda_1 \lambda_2 + \lambda_1 \lambda_3 + \lambda_2 \lambda_3) = M_{11} + M_{22} + M_{33}, \end{aligned} \quad (10.34)$$

wobei die M_{ij} die Unterdeterminanten von \mathbf{J}_2 sind, die durch Streichen der i -ten Zeile und der j -ten Spalte ermittelt werden. Ähnlich wie im einfachen Orientierungsfall besteht zwischen den Invarianten der Zusammenhang $K_2^2 \leq \frac{1}{27} S_2^3$ (vergleiche Satz 5) und wir schreiben das Konfidenzkriterium als $K_2^2 \leq \epsilon_2 S_2^3$ mit $\epsilon_2 \leq \frac{1}{27}$.

Nach dem ein Orientierungsmodell von dem Konfidenzkriterium akzeptiert wurde, gilt es, den Eigenvektor zum Eigenwert null zu ermitteln. Für eine Orientierung bedeutet dies, dass wir den Vektor suchen, der senkrecht zu einer der Zeilen von \mathbf{J}_1 steht. Die Zeile kann beliebig gewählt werden und wir haben die beiden Alternativen

$$\mathbf{u} = \frac{(-J_{12}, J_{11})^T}{\sqrt{J_{11}^2 + J_{12}^2}} \quad \text{und} \quad \mathbf{u} = \frac{(J_{22}, -J_{12})^T}{\sqrt{J_{22}^2 + J_{12}^2}}. \quad (10.35)$$

Bei zwei Orientierungen muss der gesuchte Vektor zu je zwei Zeilen der Matrix \mathbf{J}_2 orthogonal sein, da die Matrix rangreduziert ist und eine Zeile eine Linearkombination der anderen ist. Somit kann der Eigenvektor zum Eigenwert null durch das Kreuzprodukt zweier Zeilen bestimmt werden. Ferner existieren wie bereits mehrfach erwähnt drei Möglichkeiten, den Eigenvektor durch die Unterdeterminanten von \mathbf{J}_2

$$\mathbf{c}_i = R_i(M_{i3}, -M_{i2}, M_{i1})^T, \quad \text{für } i = 1, 2, 3 \quad (10.36)$$

auszudrücken, wobei der Skalierungsfaktor R_i nach Gleichung (10.12) durch $(c_{xx} + c_{yy})^2 + c_{xy}^2 = 1$ gefunden werden kann.

10.3 Bestimmung verdeckender Doppelorientierung

In den vorangegangenen Abschnitten sind wir davon ausgegangen, dass die Orientierungen entweder multiplikativ oder additiv überlagert wurden. In der Praxis finden sich jedoch viele Beispiele, in denen es zu einem plötzlichen Umbruch in der lokalen Orientierung kommt, zum Beispiel durch Verdeckungen oder Zusammenführen orientierter Muster. Setzt sich das Signal $f(\mathbf{x})$ aus zwei orientierten Signalen $g_1(\mathbf{x})$ und $g_2(\mathbf{x})$ zusammen, die sich verdecken, gliedert sich das Bild in den Bereich Ω_1 mit $f(\mathbf{x}) = g_1(\mathbf{x})$ und den komplementären Bereich Ω_2 mit $f(\mathbf{x}) = g_2(\mathbf{x})$. Demnach lautet das entsprechende Bildmodell

$$f(\mathbf{x}) = \begin{cases} g_1(\mathbf{x}) & \forall \mathbf{x} \in \Omega_1 \\ g_2(\mathbf{x}) & \forall \mathbf{x} \in \Omega_2 \end{cases} \quad (10.37)$$

mit $\Omega = \Omega_1 \cup \Omega_2$ und $\Omega_1 \cap \Omega_2 = \emptyset$. Weiter seien die Teilbilder g_1 und g_2 ideal in Richtung \mathbf{u} bzw. \mathbf{v} orientiert und somit $\partial f(\mathbf{x})/\partial \mathbf{u} = 0$ für alle Punkte \mathbf{x} in Ω_1 und $\partial f(\mathbf{x})/\partial \mathbf{v} = 0$ für alle Punkte \mathbf{x} in Ω_2 . Alternativ zu der transparenten Orientierungsbedingung (10.7), die auch in diesem Fall gilt, können wir aus der Beobachtung, dass immer eine der Richtungsableitungen null ergibt, schließen, dass

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{u}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{v}} = 0 \quad (10.38)$$

ebenfalls gültig ist. Diese Bedingung gilt für alle Punkte $\mathbf{x} \in \Omega$ außer an der Grenze zwischen den Regionen Ω_1 und Ω_2 , wo Werte ungleich null auftreten können. Die partiellen Ableitungen an der Grenze sind streng genommen nicht definiert und somit betrifft dieses Problem gleichermaßen die transparente Orientierungsbedingung. Nach Einsetzen von $\partial f/\partial \mathbf{v} = f_x v_x + f_y v_y$ in Gleichung (10.38) erhalten wir

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{u}} \frac{\partial f(\mathbf{x})}{\partial \mathbf{v}} = c_{xx} f_x f_x + c_{xy} f_x f_y + c_{yy} f_y f_y = 0 \quad (10.39)$$

mit den gemischten Orientierungsparametern

$$c_{xx} = u_x v_x, \quad c_{xy} = u_x v_y + u_y v_x, \quad c_{yy} = u_y v_y. \quad (10.40)$$

Interessanterweise ist diese Bedingung der transparenten sehr ähnlich, die gemischten Orientierungsparameter sind identisch und nur die partiellen Ableitungen zweiter Ordnung durch die Produkte partieller Ableitungen erster Ordnung ersetzt. Obige Gleichung ist als inneres Produkt $\mathbf{d}f \cdot \mathbf{c} = 0$ mit den Vektoren $\mathbf{d}f = (f_x f_x, f_x f_y, f_y f_y)^T$ und $\mathbf{c} = (c_{xx}, c_{xy}, c_{yy})^T$ darstellbar. Zur Bestimmung des gemischten Orientierungsparametervektors $\mathbf{c} = (c_{xx}, c_{xy}, c_{yy})^T$ verwenden wir wieder einen Kleinstquadrat-Ansatz und stellen das Energiefunktional

$$E(\mathbf{c}) = \int_{\Omega} (\mathbf{d}f \cdot \mathbf{c})^2 \mathbf{d}\Omega \quad (10.41)$$

auf. Unter der Annahme, dass der gemischte Orientierungsparametervektor \mathbf{c} in der Region Ω konstant und damit vom Ort unabhängig ist, können wir das Energiefunktional als

$$E(\mathbf{c}) = \mathbf{c}^T \mathbf{J}_2 \mathbf{c} \quad (10.42)$$

schreiben, mit dem Tensor

$$\mathbf{J}_2 = \int_{\Omega} \begin{pmatrix} f_x^4 & f_x^3 f_y & f_x^2 f_y^2 \\ f_x^2 f_y & f_x^2 f_y^2 & f_x f_y^3 \\ f_x^2 f_y^2 & f_x f_y^3 & f_y^4 \end{pmatrix} \mathbf{d}\Omega. \quad (10.43)$$

In Analogie zur dem transparenten Fall ergibt sich der gesuchte Orientierungsparametervektor \mathbf{c} wieder durch den Eigenvektor korrespondierend zu dem Eigenwert null. Für \mathbf{J}_2 gelten auch die Eigenschaften aus Satz 4 und das auf den Invarianten basierende Konfidenzkriterium aus Abschnitt 3.1.7 ist auf ihn übertragbar. Der Parametervektor \mathbf{c} ist für den transparenten Fall und den verdeckenden Fall identisch. Daher kann jedes der drei Verfahren aus Abschnitt 10.2.1 für die Separierung der Orientierungen aus den gemischten Orientierungen genutzt werden. Bei Bedarf kann dieses Modell auch für die Orientierungsbestimmung bzw. Bewegungsbestimmung in höheren Dimensionen genutzt werden.

10.3.1 Orientierungsbestimmung in Farbbildern

Für die Erweiterung der Orientierungsbestimmung in Farbbildern, d.h. $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^q$, können wir den Ansatz von Di Zenzo [191] hier leider nicht verwenden. In Analogie zum Grauwertfall ist $\partial\mathbf{f}(\mathbf{x})/\mathbf{u} = \mathbf{0}$ falls \mathbf{x} innerhalb von Ω_1 und $\partial\mathbf{f}(\mathbf{x})/\mathbf{v} = \mathbf{0}$ falls \mathbf{x} innerhalb von Ω_2 liegt. Das Ergebnis beider Richtungsableitungen ergibt den Nullvektor. Aus dieser Beobachtung schließen wir, dass

$$\frac{\partial\mathbf{f}(\mathbf{x})}{\mathbf{u}} \otimes \frac{\partial\mathbf{f}(\mathbf{x})}{\mathbf{v}} = \mathbf{0} \quad (10.44)$$

überall gelten muss, abgesehen von der Grenze zwischen Ω_1 und Ω_2 , wo Werte ungleich null auftreten können. Das Tensorprodukt in Gleichung (10.44) sollte symmetrisch in \mathbf{u} und \mathbf{v} sein. Da das Tensorprodukt dieser Forderung nicht gerecht wird, zerlegen wir den resultierenden Tensor in einen symmetrischen und einen schiefsymmetrischen Tensor. Stellen wir an den symmetrischen Tensor die Forderung, dass er die Nullmatrix ergibt, so entspricht der schiefsymmetrische Tensor ebenfalls der Nullmatrix und Gleichung (10.44) ist erfüllt. Dementsprechend erhalten wir nach der Expandierung des symmetrischen Tensors die Bedingung

$$c_{xx}\mathbf{f}_x \otimes \mathbf{f}_x + \frac{c_{xy}}{2}(\mathbf{f}_x \otimes \mathbf{f}_y + \mathbf{f}_y \otimes \mathbf{f}_x) + c_{yy}\mathbf{f}_y \otimes \mathbf{f}_y = \mathbf{0}, \quad (10.45)$$

wobei die gemischten Orientierungsparameter c_{xx} , c_{xy} und c_{yy} denen aus Gleichung (10.40) entsprechen. Gleichung (10.38) und (10.45) sind identisch, falls $q = 1$ ist. Für jedes Bild mit q -Spektralkomponenten beinhaltet der symmetrische Tensor $q(q+1)/2$ Gleichungen und Gleichung (10.45) ist für $q > 2$ überbestimmt.

Zur Bestimmung des gemischten Orientierungsparametervektors \mathbf{c} nutzen wir wieder einen Kleinstequadrate-Ansatz, in dem wir das Energiefunktional

$$E(\mathbf{c}) = \int_{\Omega} \left\| c_{xx}\mathbf{f}_x \otimes \mathbf{f}_x + \frac{c_{xy}}{2}(\mathbf{f}_x \otimes \mathbf{f}_y + \mathbf{f}_y \otimes \mathbf{f}_x) + c_{yy}\mathbf{f}_y \otimes \mathbf{f}_y \right\|_F^2 d\Omega \quad (10.46)$$

minimieren. Im Unterschied zu den vorangegangenen Verfahren nutzen wir für die Matrixnorm die Frobenius-Norm, die durch $\|A\|_F = \sqrt{\text{tr}(AA^T)}$ definiert ist und es uns ermöglicht, die Minimierungsaufgabe (10.46) in eine Eigenystemanalyse zu überführen. Zur Herleitung des entsprechenden Systems werten wir im ersten Schritt die Norm aus und führen zur Vereinfachung die Schreibweise $\mathbf{A} = \mathbf{f}_x \otimes \mathbf{f}_x$, $\mathbf{B} = \mathbf{f}_x \otimes \mathbf{f}_y + \mathbf{f}_y \otimes \mathbf{f}_x$ und $\mathbf{C} = \mathbf{f}_y \otimes \mathbf{f}_y$ ein. Da der Tensor symmetrisch ist, ist er identisch mit seiner Transponierten

und wir können die quadrierte Norm

$$\begin{aligned}
\|c_{xx}\mathbf{A} + \frac{c_{xy}}{2}\mathbf{B} + c_{yy}\mathbf{C}\|_F^2 &= \text{tr}((c_{xx}\mathbf{A} + \frac{c_{xy}}{2}\mathbf{B} + c_{yy}\mathbf{C})(c_{xx}\mathbf{A} + \frac{c_{xy}}{2}\mathbf{B} + c_{yy}\mathbf{C})) \\
&= \text{tr}(c_{xx}^2\mathbf{A}\mathbf{A} + \frac{c_{xx}c_{xy}}{2}(\mathbf{A}\mathbf{B} + \mathbf{B}\mathbf{A}) + c_{xx}c_{yy}(\mathbf{A}\mathbf{C} + \mathbf{C}\mathbf{A}) + \frac{c_{xy}^2}{4}\mathbf{B}\mathbf{B} \\
&\quad + \frac{c_{xy}c_{yy}}{2}(\mathbf{B}\mathbf{C} + \mathbf{C}\mathbf{B}) + c_{yy}^2\mathbf{C}\mathbf{C}) \\
&= c_{xx}^2\text{tr}(\mathbf{A}\mathbf{A}) + c_{xx}c_{xy}\text{tr}(\mathbf{A}\mathbf{B}) + 2c_{xx}c_{yy}\text{tr}(\mathbf{A}\mathbf{C}) + \frac{c_{xy}^2}{4}\text{tr}(\mathbf{B}\mathbf{B}) \\
&\quad + c_{xy}c_{yy}\text{tr}(\mathbf{B}\mathbf{C}) + c_{yy}^2\text{tr}(\mathbf{C}\mathbf{C}) \\
&= \mathbf{c}^T \begin{pmatrix} \text{tr}(\mathbf{A}\mathbf{A}) & \frac{1}{2}\text{tr}(\mathbf{A}\mathbf{B}) & \text{tr}(\mathbf{A}\mathbf{C}) \\ \frac{1}{2}\text{tr}(\mathbf{A}\mathbf{B}) & \frac{1}{4}\text{tr}(\mathbf{B}\mathbf{B}) & \frac{1}{2}\text{tr}(\mathbf{B}\mathbf{C}) \\ \text{tr}(\mathbf{A}\mathbf{C}) & \frac{1}{2}\text{tr}(\mathbf{B}\mathbf{C}) & \text{tr}(\mathbf{C}\mathbf{C}) \end{pmatrix} \mathbf{c}
\end{aligned} \tag{10.47}$$

als eine quadratische Form schreiben, da für zwei symmetrische Matrizen \mathbf{A} und \mathbf{B} $\text{tr}(\mathbf{A}\mathbf{B}) = \text{tr}(\mathbf{B}\mathbf{A})$ gilt. Im zweiten Schritt werten wir die Spuren aus, wobei wir uns hier exemplarisch auf einige Fällen beschränken wollen, da die übrigen analog berechnet werden. So erhalten wir beispielsweise für

$$\begin{aligned}
\text{tr}(\mathbf{A}\mathbf{A}) &= \sum_{i=1}^q \sum_{k=1}^q a_{ik}a_{ki} = \sum_{i=1}^q \sum_{k=1}^q (\mathbf{f}_x)_i (\mathbf{f}_x)_k (\mathbf{f}_x)_k (\mathbf{f}_x)_i \\
&= \sum_{i=1}^q (\mathbf{f}_x)_i (\mathbf{f}_x)_i \sum_{k=1}^q (\mathbf{f}_x)_k (\mathbf{f}_x)_k = \|\mathbf{f}_x\|_2^4, \\
\text{tr}(\mathbf{A}\mathbf{B}) &= \sum_{i=1}^q \sum_{k=1}^q a_{ik}b_{ki} = \sum_{i=1}^q \sum_{k=1}^q (\mathbf{f}_x)_i (\mathbf{f}_x)_k ((\mathbf{f}_x)_k (\mathbf{f}_y)_i + (\mathbf{f}_y)_k (\mathbf{f}_x)_i) \\
&= \sum_{i=1}^q (\mathbf{f}_x)_i (\mathbf{f}_y)_i \sum_{k=1}^q (\mathbf{f}_x)_k (\mathbf{f}_x)_k + \sum_{i=1}^q (\mathbf{f}_x)_i (\mathbf{f}_x)_i \sum_{k=1}^q (\mathbf{f}_x)_k (\mathbf{f}_y)_k = 2\|\mathbf{f}_x\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y, \\
\text{tr}(\mathbf{A}\mathbf{C}) &= \sum_{i=1}^q \sum_{k=1}^q a_{ik}c_{ki} = \sum_{i=1}^q \sum_{k=1}^q (\mathbf{f}_x)_i (\mathbf{f}_x)_k (\mathbf{f}_y)_k (\mathbf{f}_y)_i \\
&= \sum_{i=1}^q (\mathbf{f}_x)_i (\mathbf{f}_y)_i \sum_{k=1}^q (\mathbf{f}_x)_k (\mathbf{f}_y)_k = (\mathbf{f}_x \cdot \mathbf{f}_y)^2.
\end{aligned} \tag{10.48}$$

Äquivalent können wir das Energiefunktional (10.46) mit den vorangegangenen Betrachtungen als

$$E(\mathbf{c}) = \int_{\Omega} \mathbf{c}^T \begin{pmatrix} \|\mathbf{f}_x\|_2^4 & \|\mathbf{f}_x\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & (\mathbf{f}_x \cdot \mathbf{f}_y)^2 \\ \|\mathbf{f}_x\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & \frac{1}{2}(\|\mathbf{f}_x\|_2^2 \|\mathbf{f}_y\|_2^2 + (\mathbf{f}_x \cdot \mathbf{f}_y)^2) & \|\mathbf{f}_y\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y \\ (\mathbf{f}_x \cdot \mathbf{f}_y)^2 & \|\mathbf{f}_y\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & \|\mathbf{f}_y\|_2^4 \end{pmatrix} \mathbf{c} \, d\Omega \tag{10.49}$$

schreiben und nach dem Satz von Courant-Fischer [73] ergibt sich der gesuchte Vektor \mathbf{c} aus dem Eigenvektor korrespondierend zu dem kleinsten Eigenwert von

$$\mathbf{J}_2 = \int_{\Omega} \begin{pmatrix} \|\mathbf{f}_x\|_2^4 & \|\mathbf{f}_x\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & (\mathbf{f}_x \cdot \mathbf{f}_y)^2 \\ \|\mathbf{f}_x\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & \frac{1}{2}(\|\mathbf{f}_x\|_2^2 \|\mathbf{f}_y\|_2^2 + (\mathbf{f}_x \cdot \mathbf{f}_y)^2) & \|\mathbf{f}_y\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y \\ (\mathbf{f}_x \cdot \mathbf{f}_y)^2 & \|\mathbf{f}_y\|_2^2 \mathbf{f}_x \cdot \mathbf{f}_y & \|\mathbf{f}_y\|_2^4 \end{pmatrix} d\Omega. \tag{10.50}$$

Die Matrix \mathbf{J}_2 ist positiv semidefinit und somit gelten ebenfalls die in Abschnitt 3.1.7 entworfenen Konfidenzkriterien.

10.4 Schichtentrennung und Regionenbestimmung

Für einige Applikationen kann die Verwendung der Orientierungsinformation zur Trennung der Muster von Interesse sein. Sowohl im transparenten als auch im verdeckenden Fall können wir mit Richtungsableitungen eine Art Signaltrennung erzielen.

Wenden wir die Richtungsableitung in Richtung einer der Orientierungen auf das transparente Modell an, so erhalten wir

$$\alpha(\theta_1)f(\mathbf{x}) = \alpha(\theta_1)g_2(\mathbf{x}) \text{ bzw. } \alpha(\theta_2)f(\mathbf{x}) = \alpha(\theta_2)g_1(\mathbf{x}). \quad (10.51)$$

Jede Richtungsableitung entfernt alle Anteile in Richtung dieser Orientierung und übrig bleibt die Richtungsableitung angewendet auf das zweite Muster, die, falls das Originalmuster gesucht ist, invertiert werden muss. Die Invertierung kann zum Beispiel im Frequenzraum durchgeführt werden.

10.4.1 Trennung transparenter Muster im Frequenzraum

Ideal orientierte Bilder besetzen im Spektralbereich eine Gerade durch den Ursprung, deren Normalenvektor dem Orientierungsvektor entspricht. Jedes weitere transparente Muster ergibt eine zusätzliche Gerade im Frequenzraum. Um die Fouriertransformierte eines der zwei Muster zu erhalten, müssen alle Frequenzanteile auf der Geraden des zweiten Musters entfernt werden. Dies kann zum Beispiel durch die Spezifikation orientierter Filter erfolgen.

In diesem Abschnitt stellen wir eine Lösung vor, die partielle Ableitungen zur Trennung nutzt und daher ohne die Spezifikation solcher Filter auskommt. Wenden wir die Richtungsableitung $\alpha(\theta_1)$ auf $f(\mathbf{x})$ an, so ergibt dies im Frequenzbereich,

$$\alpha(\theta_1)f(\mathbf{x}) = \alpha(\theta_1)g_2(\mathbf{x}) \circ \bullet j2\pi\boldsymbol{\omega} \cdot \mathbf{u} G_2(\boldsymbol{\omega}) =: \bar{G}_2(\boldsymbol{\omega}). \quad (10.52)$$

An Frequenzen bei denen $\boldsymbol{\omega} \cdot \mathbf{u} \neq 0$, $\mathbf{u} = (\cos \theta_1, \sin \theta_2)^T$ ist, erhalten wir $G_2(\boldsymbol{\omega})$ durch

$$G_2(\boldsymbol{\omega}) = \frac{\bar{G}_2(\boldsymbol{\omega})}{j2\pi\boldsymbol{\omega} \cdot \mathbf{u}}. \quad (10.53)$$

Die Frequenzen mit $\boldsymbol{\omega} \cdot \mathbf{u} = 0$ korrespondieren zu $G_1(\boldsymbol{\omega})$ und werden daher mit

$$G_2(\boldsymbol{\omega} \cdot \mathbf{u}) = 0. \quad (10.54)$$

zu null gesetzt. Da beide Geraden durch den Ursprung verlaufen, ist es unmöglich, die Frequenzanteile im Ursprung von einander zu trennen. Damit verlieren wir die Mittelwerte der Muster.

Das hier beschriebene Separierungsverfahren setzt konstante Orientierungen in dem Bild voraus. Bei Bildern, in denen es zu graduellen Änderungen in den Orientierungsfeldern kommt, kann die Trennung mit diesem Verfahren nicht mehr global für das ganze Bild sondern nur noch lokal durchgeführt werden.

10.4.2 Regionenermittlung bei verdeckenden Orientierungen

Wenden wir die Richtungsableitungen auf das verdeckende Modell an, erhalten wir ähnlich wie im transparenten Fall

$$\begin{aligned} \alpha(\theta_1)f(\mathbf{x}) &= \begin{cases} 0 & \forall \mathbf{x} \in \Omega_1 \\ \alpha(\theta_1)g_2(\mathbf{x}) & \forall \mathbf{x} \in \Omega_2 \end{cases} \quad \text{bzw.} \\ \alpha(\theta_2)f(\mathbf{x}) &= \begin{cases} \alpha(\theta_2)g_1(\mathbf{x}) & \forall \mathbf{x} \in \Omega_1 \\ 0 & \forall \mathbf{x} \in \Omega_2 \end{cases}. \end{aligned} \quad (10.55)$$

Die Gleichungen ermöglichen die Ermittlung der Bereiche Ω_1 und Ω_2 . Für die Regionszuordnung werten wir an jedem Punkt die quadrierte Richtungsableitung entsprechend Gleichung (10.55) für beide Orientierungen θ_1 und θ_2 aus und summieren sie über eine kleine Nachbarschaft (z.B. 3×3 Pixel), d.h.

$$E_{\theta_1}(\mathbf{x}) = \sum_{\mathbf{y} \in N_{\mathbf{x}}} (\alpha(\theta_1)f(\mathbf{y}))^2, \quad E_{\theta_2}(\mathbf{x}) = \sum_{\mathbf{y} \in N_{\mathbf{x}}} (\alpha(\theta_2)f(\mathbf{y}))^2, \quad (10.56)$$

wobei wir mit $N_{\mathbf{x}}$ die Nachbarschaft um den Punkt \mathbf{x} bezeichnen. Aus dem Vergleich beider Werte bekommen wir die gesuchte Zuordnung:

- Ist $E_{\theta_1}(\mathbf{x}) < E_{\theta_2}(\mathbf{x})$, so ist $\mathbf{x} \in \Omega_1$ und in Richtung θ_1 orientiert.
- Ist $E_{\theta_1}(\mathbf{x}) > E_{\theta_2}(\mathbf{x})$, so ist $\mathbf{x} \in \Omega_2$ und in Richtung θ_2 orientiert.

Die Regularisierung über die Nachbarschaft soll Fehlentscheidungen, die z.B. durch Rauschen in den Bildern hervorgerufen wurden, reduzieren. Für Farbbilder ist lediglich der quadrierte Ausdruck in Gleichung (10.56) durch die quadrierte Norm dieses Ausdruckes zu ersetzen.

10.5 Ergebnisse

Abbildung 10.3 zeigt ein synthetisches Beispiel für additive Überlagerungen. Die beiden Originalschichten (a) und (b) sind in (c) addiert und mit gaußverteilterm weißen Rauschen verrauscht worden, so dass ein Signalrauschverhältnis von 15dB entstand. Die ermittelten Orientierungsvektoren sind in (f) für jeden 15ten Bildpunkt dargestellt und stimmen optisch gut mit den Orientierungen im Bild überein. Die Bilder (d) und (e) bestätigen den schon optisch gewonnenen Eindruck und zeigen jeweils Richtungsableitungen nach einer der beiden Orientierungsfelder entsprechend Gleichung (10.51). Die Anteile der Orientierungen in Ableitungsrichtung konnten entfernt werden und es bleiben nur die richtungsabgeleiteten Anteile des zweiten Musters übrig. Zu bemerken ist, dass das Orientierungsbestimmungsverfahren zwar an jedem Punkt zwei Orientierungen liefert, diese aber nicht eindeutig den Mustern zugeordnet sind, und somit eine Sortierung der Orientierungen nach den Mustern für die Berechnung der Richtungsableitungen erforderlich macht. In einem Orientierungswinkelhistogramm aller Orientierungen der Bildebene lieferte uns der größte Peak über einen einfachen Schwellwert die 45 Gradorientierung. Anschließend nutzten wir diese Orientierung für die Sortierung der Orientierungsfelder.

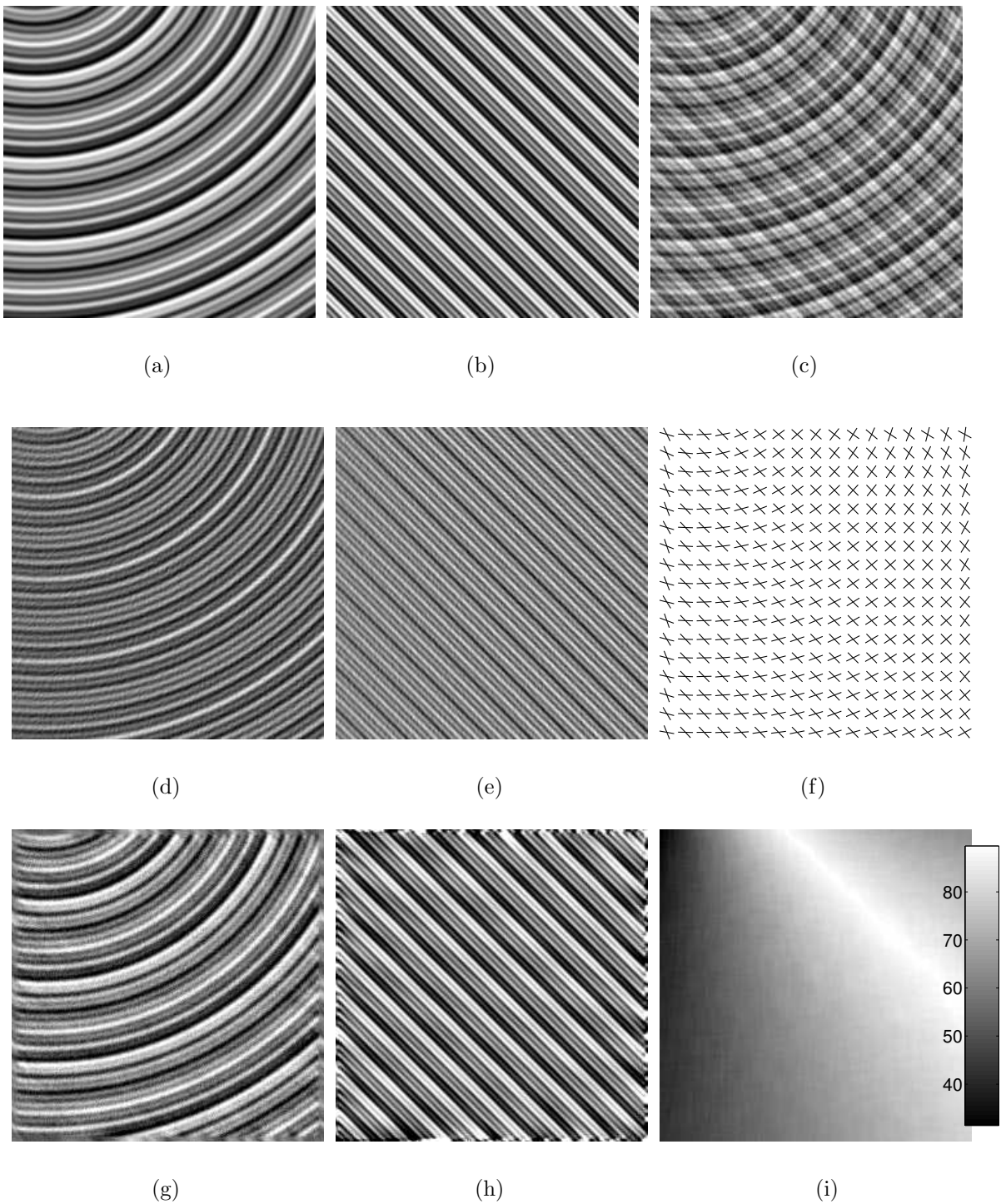


Abbildung 10.3: Test mit additiver Überlagerung. (a) und (b) Originalschichten, die in (c) überlagert sind, (f) ermittelte Orientierungen, (d) und (e) Ergebnis nach dem Nullen einer der Orientierungen, (g) und (h) Rekonstruktion der Schichten im Fourierraum, (i) Winkel zwischen den Orientierungen als Invariante.

Für jeden Punkt ordneten wir die Orientierung, die am dichtesten an der 45 Grad Orientierung lag, dem 45 Grad Muster zu. Die Qualität der gefundenen Ergebnisse ist weiter durch Zahlenwerte bestätigt: der RMSE-Wert sowie die Standardabweichung fällt für das Kreismuster mit 0.53 bzw. 0.01 Grad und für das 45 Grad Muster mit 0.66 und 0.02 Grad sehr klein aus.

Die Bilder (g) und (h) zeigen die aus (c) mit Hilfe der Orientierungsinformation rekonstruierten Schichten. Dabei wurde im ersten Schritt die Richtungsableitung des richtungsabgeleiteten Halbkreismusters aus (d) im Frequenzraum invertiert (folglich Abschnitt 10.4.1) und die Rücktransformierte in (g) dargestellt. Abbildung (h) entstand durch Abziehen der rekonstruierten Schicht (g) von dem Originalbild und verdeutlicht, dass eine Schichtentrennung auch dann möglich ist, wenn nur eines der beiden Muster ein konstantes Orientierungsfeld über dem Bildbereich aufweist. Die in Abschnitt 10.4.1 vorgestellten Methoden können nur angewendet werden, wenn die Richtungsableitungen über dem betrachteten Bereich konstant und nicht gekrümmt sind. Mit Hilfe der Matlab Edgetaper-Funktion wurden die Übergänge zur nächsten Periode geglättet und die Abweichungen am Rand erklärt.

Abschließend ist in (i) der von beiden Orientierungsvektoren eingeschlossene Winkel nach Gleichung (10.29) abgebildet. Auch er stimmt optisch gut mit den wahren Zwischenwinkeln im Bild überein und beschreibt wie erwartet einen sehr kontinuierlichen und glatten Übergang.

Die partiellen Ableitungen sind in diesem Beispiel mit den separierbaren Filtern $[-1\ 0\ 1]/2$ in Ableitungsrichtung und $[1\ 3\ 1]/5$ in Glättungsrichtung berechnet worden. Die partiellen Ableitungen zweiter Ordnung ermittelten wir durch Hintereinanderausführung der partiellen Ableitungen erster Ordnung. Die Integration der Tensoreinträge ist durch einen 15×15 Bildpunkte großen Mittelwertfilter realisiert. Für die Konfidenzkriterien nutzten wir die Parameter $\epsilon = 0.00001$, $c_1 = 0.02$ und $c_2 = 0.025$.

Abbildung 10.4(a) zeigt eine Röntgenaufnahme eines Autoreifens. Die Karkassen, in mehreren Schichten aufgebracht, sind deutlich als orientierte Strukturen erkennbar. Die ermittelten Orientierungsvektoren sind für eine und zwei Orientierungen in (b) für jeden zehnten Bildpunkt überlagert dargestellt und zeigen eine gute Übereinstimmung mit den Orientierungen im Bild. Auch die Übergängen zwischen einer und zwei bzw. zwei und drei Orientierungen sind gut getroffen worden. Nur in dem Bereich mit drei Orientierungen konnte keine Orientierungsbestimmung durchgeführt werden, da sich die hier vorgestellten Verfahren nur für die Bestimmung von zwei Orientierungen eignen. Die Anzahl der gefundenen Orientierungen in jedem Bildpunkt sind in (c) farbkodiert dargestellt: grau steht für eine Orientierung, weiß für zwei und schwarz für keine oder mehr als zwei Orientierungen. In diesem Anwendungsfall kann beispielsweise aus dem Verlauf der Orientierungsgrenzen in (c) sowie den gefundenen Orientierungen selber auf Produktionsfehler geschlossen werden. Die Bilder (d) und (e) zeigen die Ergebnisse der Richtungsableitungen in Richtung jeweils einer Orientierung. Da in beiden Fällen jeweils eine der Orientierung entfernt werden konnte, stimmen die berechneten und wahren Orientierungen sehr gut überein. Bild (f) zeigt ein Ergebnis zur Entfernung eines Musters. Der Mittelwert alle Richtungen eines

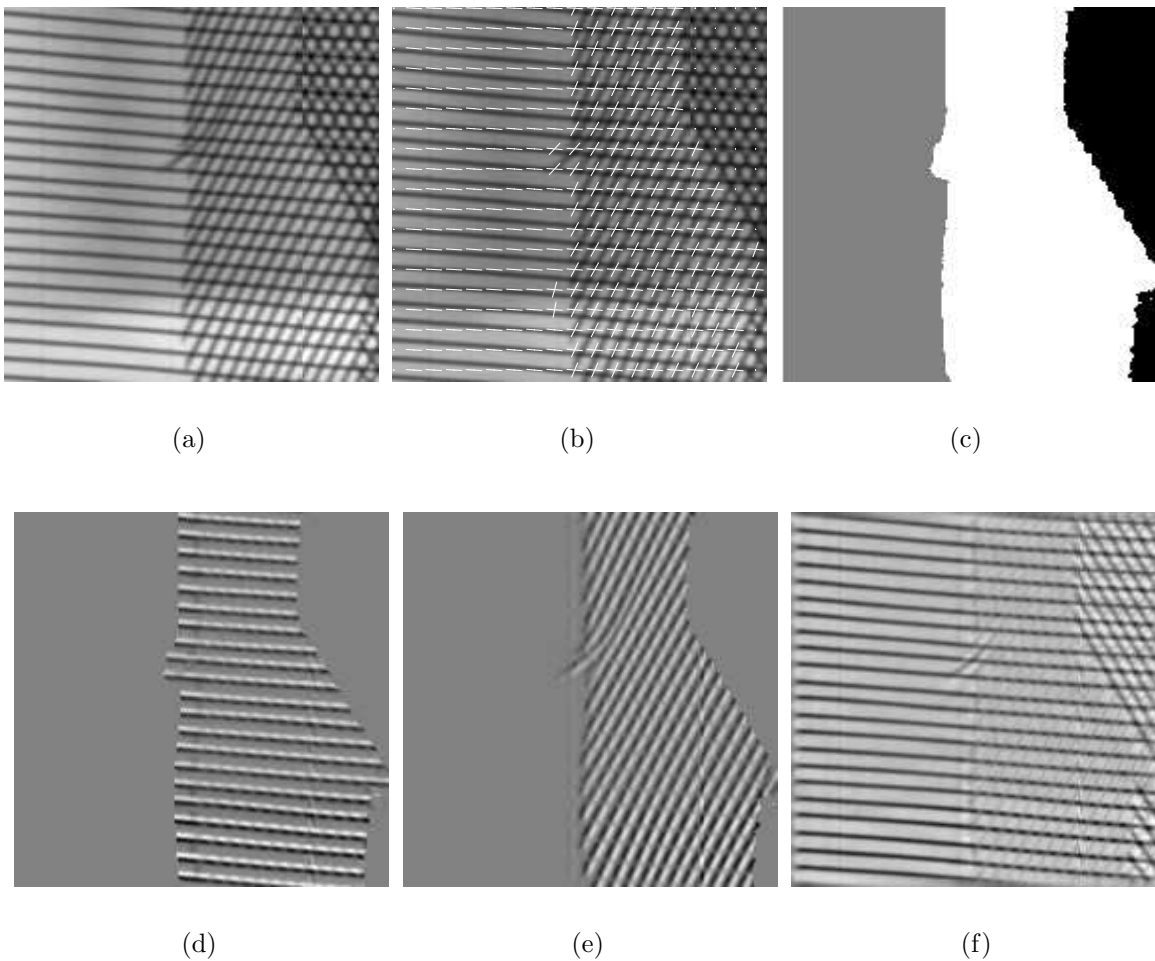
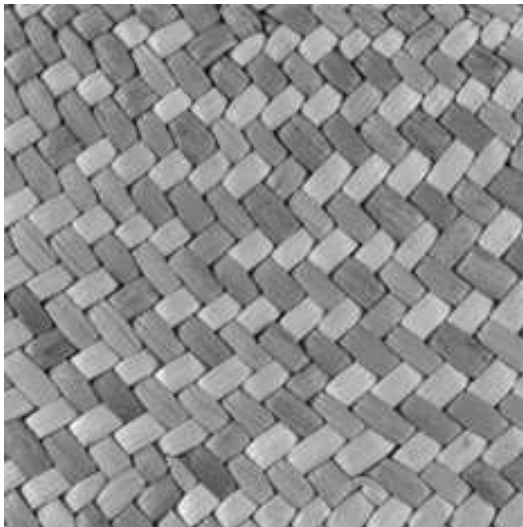


Abbildung 10.4: (a) Röntgenaufnahme eines Autoreifens, (b) ermittelte Einfach- und Doppelorientierung überlagert, (c) Anzahl gefundener Orientierungen farbkodiert, (d) und (e) Richtungsableitungen im Bereich für zwei Orientierungen, (d) Entfernung einer Orientierung und Rekonstruktion mittels der Fouriertransformation.

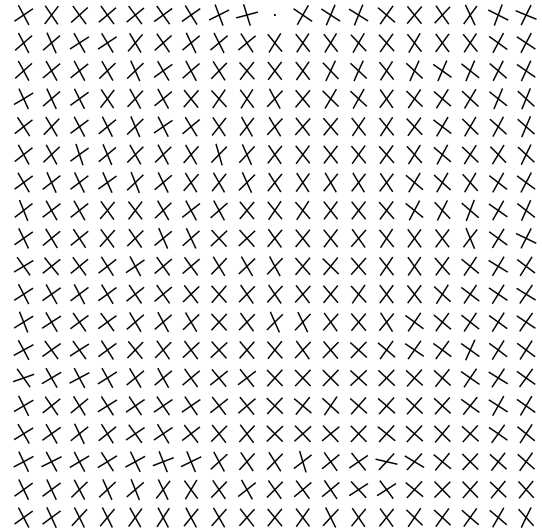
Musters ist hier zur Bildung der Richtungsableitung über den gesamten Bild sowie deren Invertierung genutzt worden. Da es im Orientierungsfeld zu Variationen kam, ist die Entfernung und die Rekonstruktion im Frequenzraum nicht perfekt und Reste der Struktur bleiben erkennbar.

Abbildung 10.5 zeigt zwei Anwendungsbeispiele des additiven Orientierungsmodells für reale Texturen. Trotz der Tatsache, dass die Orientierungen nicht additiv überlagert sondern verdeckt sind, können mit diesem Verfahren brauchbare Ergebnisse erzielt werden. Die Integrationsfenstergröße lag bei 19×19 Bildpunkten und die Parameter $c_1 = 0.01$, $c_2 = 0.025$ und $\epsilon = 0.00001$.

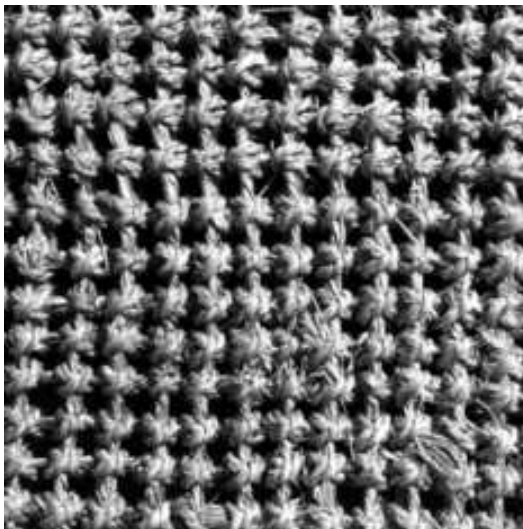
Die Abbildungen 10.6, 10.7 und 10.8 zeigen einige Messergebnisse für den Zwischenwinkel von zwei additiv überlagerten Sinusmustern und einem Signalrauschverhältnis von 30 dB. In allen drei Abbildungen sind die RMSE-Werte aller Zwischenwinkel mit ihren Standard-



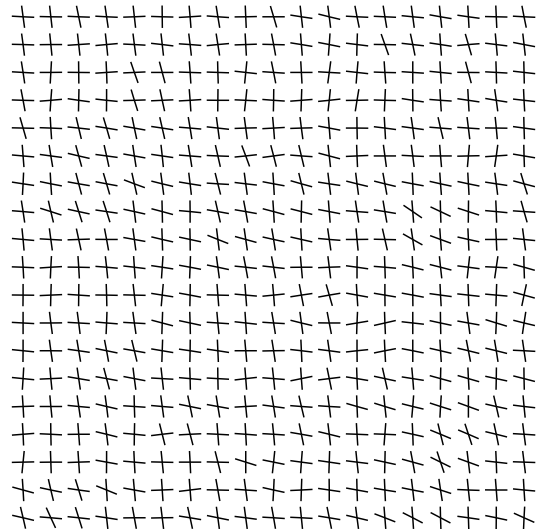
(a)



(b)

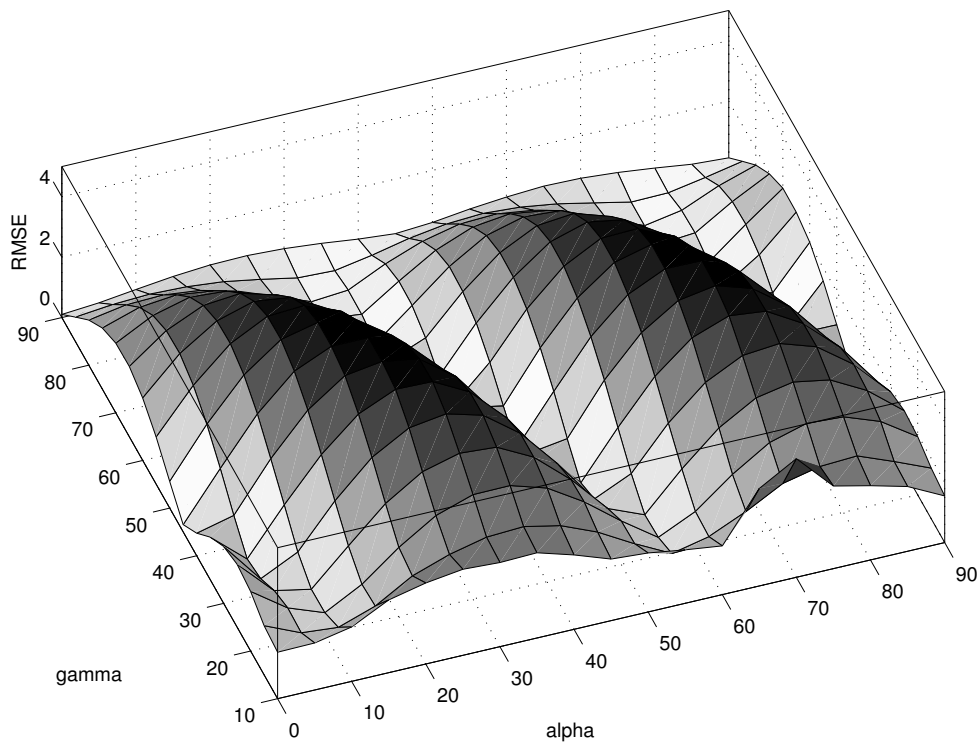


(c)

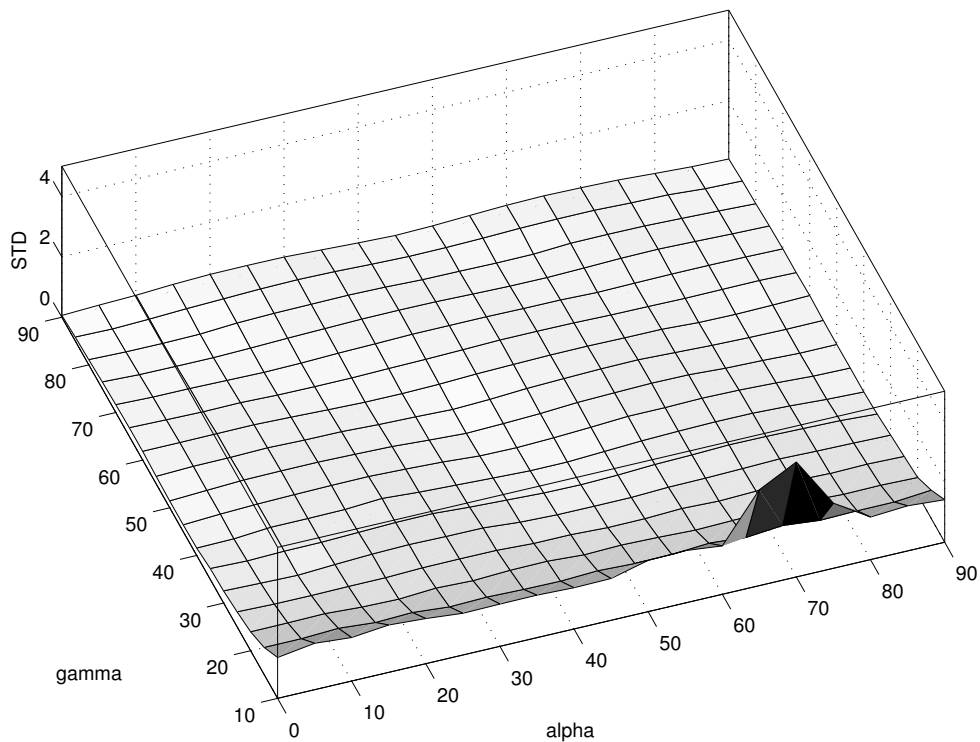


(d)

Abbildung 10.5: Beispielergebnis für zwei Texturen mit dem additiven Orientierungsmodell.

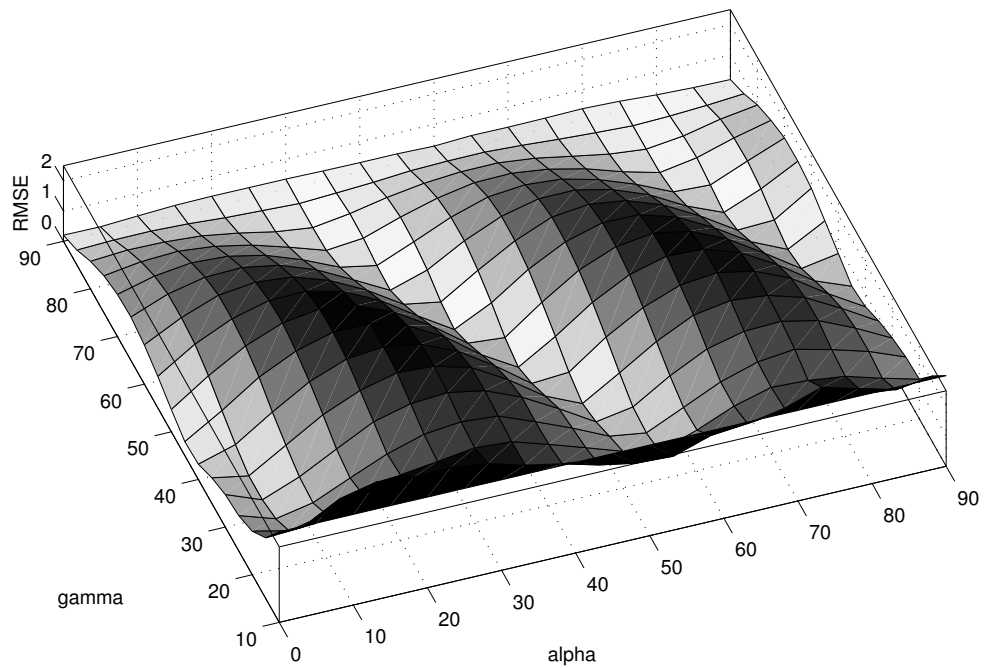


(a)

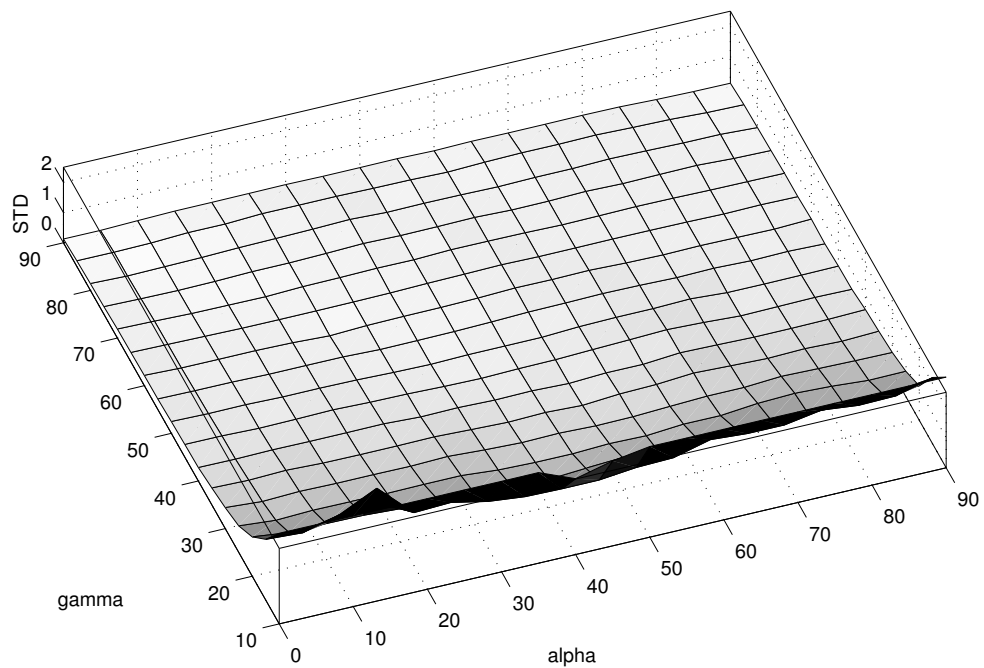


(b)

Abbildung 10.6: (a) RMSE des Zwischenwinkels δ , (b) Standardabweichung über verschiedene Winkel bei einer Grenzfrequenz der Schichten $f_g = 1/6$.

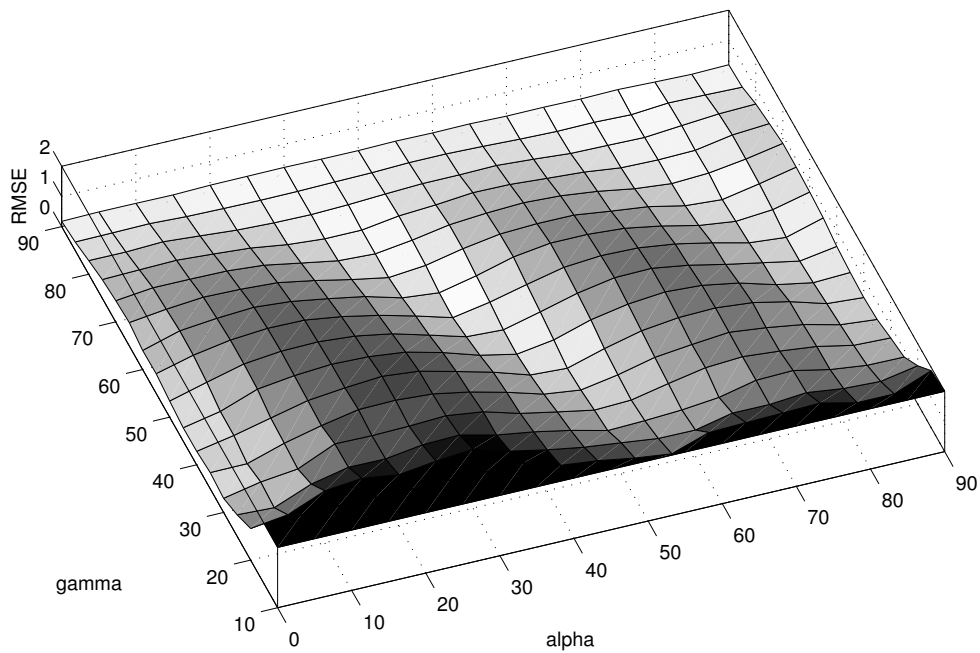


(a)

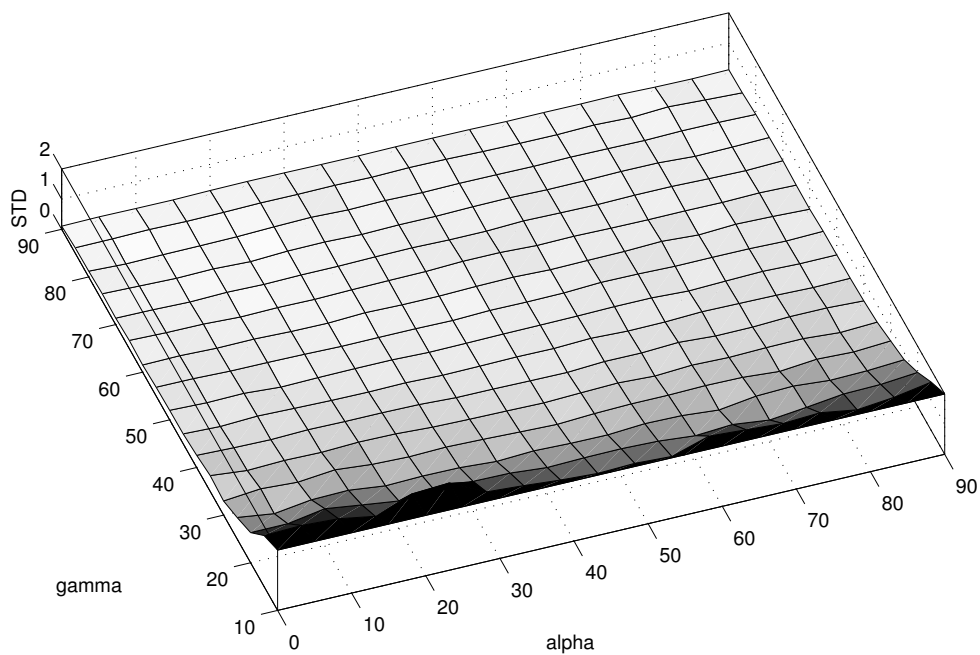


(b)

Abbildung 10.7: (a) RMSE des Zwischenwinkels δ , (b) Standardabweichung über verschiedene Winkel bei einer Grenzfrequenz der Schichten $f_g = 1/8$.



(a)



(b)

Abbildung 10.8: (a) RMSE des Zwischenwinkels δ , (b) Standardabweichung über verschiedene Winkel bei einer Grenzfrequenz der Schichten $f_g = 1/6$ und Aufwärtsabtastung.

abweichungen über den Winkeln α und γ in 5 Gradschritten abgebildet. Eines der Muster ist in α orientiert während das andere Muster in Richtung $\beta = \alpha + \gamma$ orientiert ist. Der Winkel γ beschreibt demnach den Winkel zwischen beiden Mustern. Zur Erzeugung der Muster wurden verschiedene Sinusfunktionen unterschiedlicher Frequenzen überlagert, so dass orientierte Muster ähnlich wie in Abbildung 10.3 (b) entstanden. Bei den Ergebnissen in den Abbildungen 10.6 und 10.8 liegt die Grenzfrequenz f_g der Muster bei $1/6$ und in 10.7 bei $1/8$. Interessanterweise scheint die Grenzfrequenz der Muster eine Auswirkung auf die Qualität der Ergebnisse zu haben, denn die Fehler bei einer Grenzfrequenz von $1/8$ fallen deutlich kleiner aus als bei $1/6$. Ferner fällt in allen Fällen auf, dass es Winkelkombinationen gibt, in denen die Fehler größer sind als bei anderen. Damit wird deutlich, dass der Zwischenwinkel in der Praxis nicht hundertprozentig rotationsinvariant sein muß. Die Standardabweichungen in allen drei Fällen sind sehr klein und verdeutlichen, dass es sich insbesondere bei den großen Fehlern um systematische Fehler handelt. Am Kleinsten fallen die Fehler in Abbildung 10.8 aus und bestätigen damit den von Köthe [97] zuvor für eine Orientierung angemerkten Effekt, dass sich durch Überabtastung Aliaseffekte, die bei der Produktbildung der Tensoreinträge entstehen können, reduzieren lassen. Die Integrationsfenstergröße fiel mit 17×17 in dem letzten Beispiel knapp doppelt so groß aus, wie in den 9×9 Bildpunkte umfassenden Fenster der vorangegangenen Beispiele, entspricht dann aber in etwa dem gleichen aber höher aufgelösten Bildausschnitt.

Abbildung 10.9 zeigt ein synthetisches und ein reales Beispiel für die einfache und die verdeckende Orientierungsbestimmung. Im Bild (a) sind zwei sinusförmig orientierte Muster mit einem Signalrauschverhältnis von 25 dB so angeordnet, dass sie eine T-Junction entlang der Hauptdiagonalen formen. Die ermittelten Orientierungsvektoren für einen ausgewählten Ausschnitt sind in (b) dargestellt. Deutlich zu erkennen ist, dass in einem streifenförmigen Bereich um die Junction herum zwei Orientierungen pro Bildpunkt ermittelt wurden, eine für das diagonal verlaufende und eine für das kreisförmige Muster. Außerhalb dieses Bereiches konnten nur einfache Orientierungen ermittelt werden. In (c) sind die Vektoren nach dem Ausdünnvorgang dargestellt, bei dem immer nur der Vektor beibehalten wird, der am besten die Orientierung in dem betrachteten Punkt beschreibt. Der nach der Ausdünnung entstandene Umbruch im Orientierungsfeld und die Position der Junction stimmen sehr gut überein. Dieser Umbruch erlaubt nun beispielsweise, die beiden unterschiedlich orientierten Regionen voneinander zu trennen und damit den Verlauf und die Position der Junction zu bestimmen.

Abbildung (d) zeigt ein reales Beispiel mit vielen Orientierungen und Junctionen. Die ermittelten Orientierungen für die in (d) ausgewählte Region sind in (e) dargestellt. Die Orientierungen nach dem Ausdünnvorgang in (f) stimmen gut mit den jeweiligen Orientierungen im Bild überein. In beiden Beispielen wurden für die Nachbarschaftsintegration ein 9×9 großer Mittelwertfilter und für den Konfidenztest die Konstanten $c_1 = c_2 = 0.025$ und $\epsilon = 0.001$ verwendet. Für die Berechnung der Ableitungen haben wir die gleichen Filter aus dem vorangegangenen Beispiel eingesetzt.

Das zweite Beispiel in Abbildung 10.10 zeigt einige Resultate für L-Junctionen mit unterschiedlichen Winkeln. Bild (a) zeigt den Buchstaben 'A' mit additivem gaußverteilterm Rauschen und einem Signalrauschabstand von 25 dB. Eine Segmentierung, die die An-

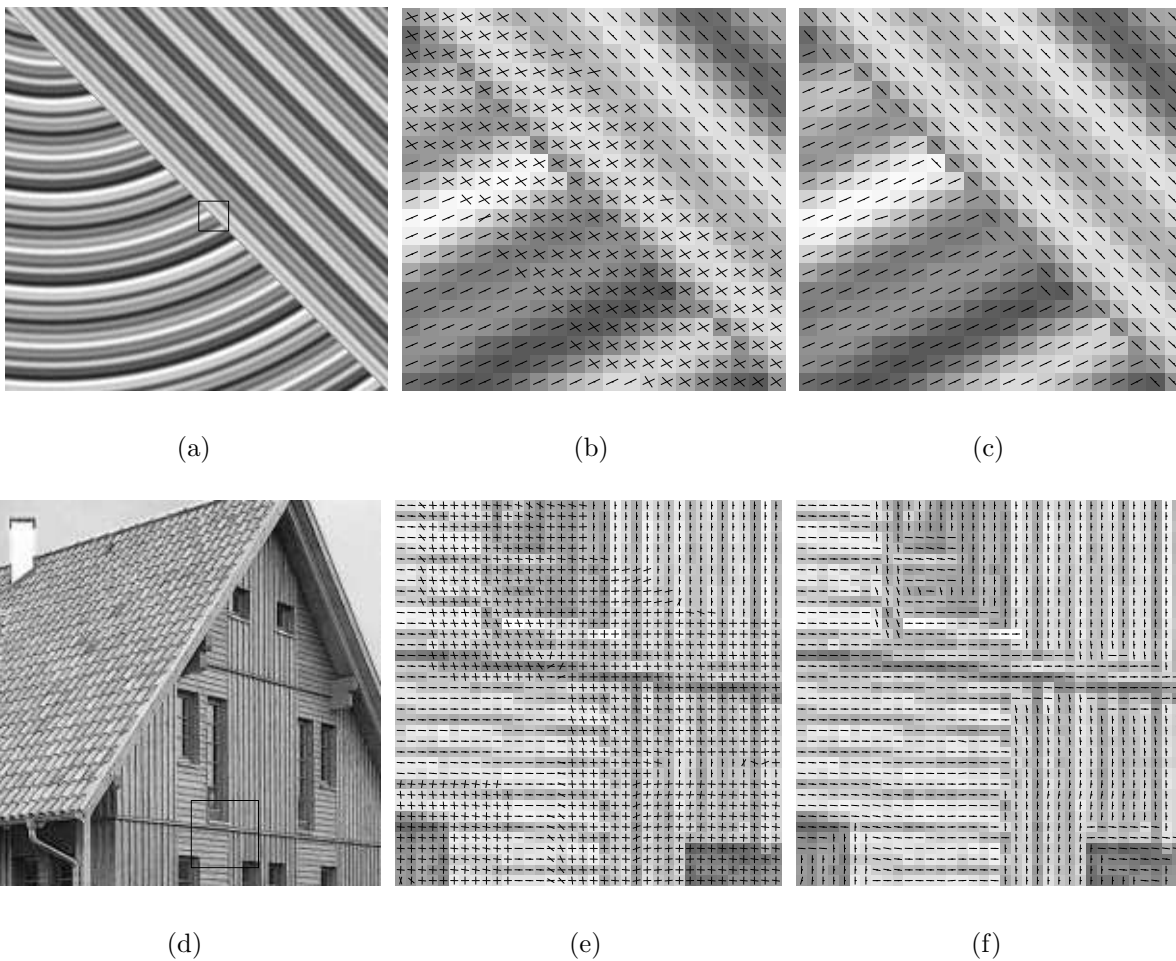


Abbildung 10.9: Synthetisches Beispiel: Bild (a) zeigt ein sinusförmiges Muster (b) die ermittelten Orientierungen für die markierte Region (c) die Orientierungen nach dem Ausdünnen. Reales Beispiel: Bild (d) zeigt ein Bild eines Hauses; (e) und (f) sind analog zu (b) und (c).

zahl der beobachteten Orientierungen in jedem Bildpunkt wiedergibt, ist in (d) dargestellt: weiß bedeutet keine, schwarz eine und grau steht für zwei Orientierungen. Um jede Ecke des Buchstabens ist ein Bereich mit zwei Orientierungen ermittelt worden. Die bestimmten Orientierungen für die obere linke Ecke sind in (b) vor und in (c) nach dem Ausdünnen dargestellt. Abbildung (e) zeigt die Orientierungen für die Ecke des Buchstabens mit dem kleinsten Winkel. Für alle Punkte mit zwei Orientierungen um die Ecke herum wurde zur Lokalisierung der Eckenposition das Verfahren von Förstner & Gülch [66], das ebenfalls auf einem Tensor basiert, herangezogen. Das Kreuz in (f) repräsentiert die so ermittelte Eckenposition und zeigt gleichzeitig die zu diesem Punkt korrespondierenden Orientierungen. Auch in diesem Beispiel entsprechen die verwendeten Ableitungsoperatoren denen aus den vorangegangenen Beispielen. Weiter sind in diesem Beispiel ein Integrationsfenster von 7×7 Bildpunkten und die Konstanten $c_1 = 0.15$, $c_2 = 0.025$ und $\epsilon = 0.01$ für den Konfidenztest verwendet worden.

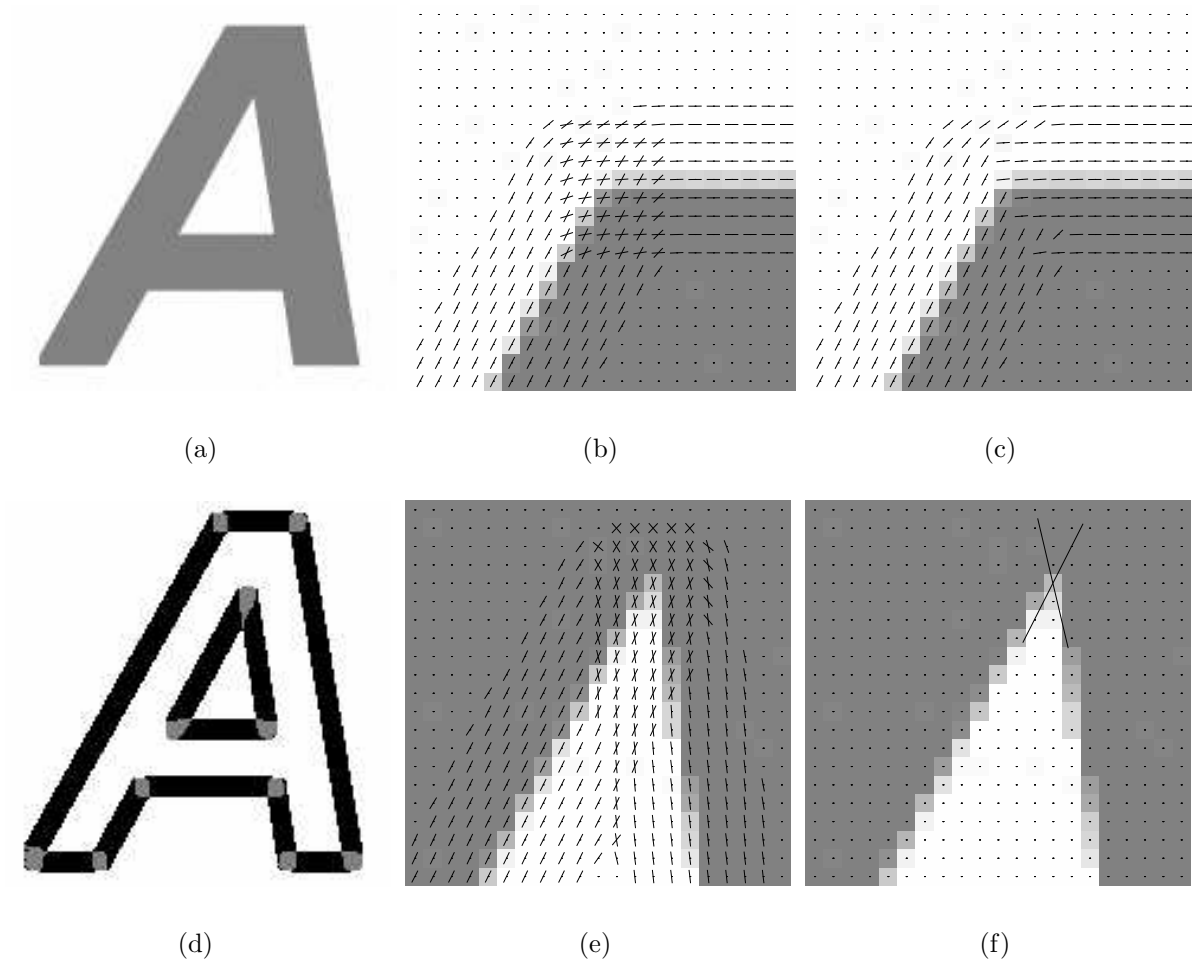


Abbildung 10.10: Abbildung (a) zeigt den Buchstaben 'A' als Eingabebild. Die ermittelten Orientierungen vor und nach dem Ausdünnvorgang sind in (b) und (c) für die linke obere Ecke des 'A' dargestellt. Bild (d) zeigt eine Segmentierung in Sinne der Anzahl beobachteter Orientierungen, (e) die ermittelten Orientierungen für die Ecke mit dem kleinsten Winkel und (f) die nach Förstners Methode ermittelte Eckenposition mit den dort auftretenden Orientierungen.

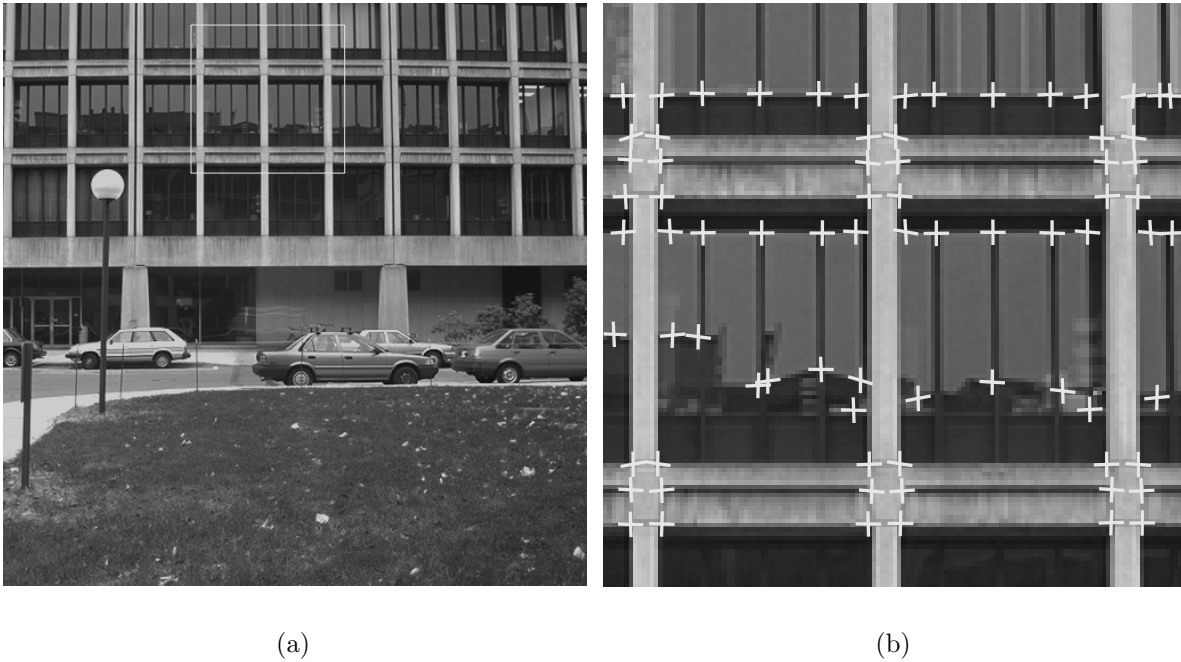


Abbildung 10.11: Abbildung (a) zeigt ein Bild eines Hochhauses, in dem viele L - und T -Junctions vorkommen. Die für den Ausschnitt ermittelten Eckpositionen mit den zu ihnen korrespondierenden Orientierungen sind in (b) durch die weißen Kreuze beschrieben.

Abschließend zeigt Abbildung 10.11 noch ein reales Ergebnis für die Orientierungsanalyse an relevanten Merkmalen wie Ecken, das mit dem verdeckenden Modell berechnet wurde. In (a) sehen wir ein Foto eines Hochhauses. Die weißen Kreuze in (b) repräsentieren die gefundenen Eckpositionen mit den korrespondierenden Orientierungen innerhalb des dargestellten Bildausschnittes. Der Algorithmus hat die meisten Ecken genau lokalisieren können und auch die ermittelten Orientierungen stimmen gut mit den Orientierungen im Bild überein. Für die Eckenlokalisierung haben wir im ersten Schritt den Wert des größten aller kleinsten Eigenwerte des einfachen Orientierungstensors über dem kompletten Bildbereich ermittelt. Alle Punkte, deren kleinster Eigenwert größer als 20 Prozent des zuvor ermittelten maximalen Eigenwertes ist, wurden als potentielle Eckenkandidaten markiert. Anschließend wurden durch eine Nicht-Maximum-Unterdrückung nur die Punkte als Eckenkandidaten beibehalten, die das Maximum in einer 5×5 Nachbarschaft darstellen. Da die so ermittelten Eckpunkte in einigen Fällen geringfügig neben der eigentlichen Eckposition lagen, haben wir im zweiten Schritt die Eckposition mit den Verfahren von Förstner & Gülch [66] unter der Hinzunahme einer 5×5 Integrationsnachbarschaft um jeden dieser Punkte verfeinert. Die Identifikation der Bereiche, in denen sich eine Ecke befindet, konnte hier nicht wie im vorangegangenen Beispiel anhand der Bereiche mit zwei Orientierungen bestimmt werden, da in diesem Beispiel die Bereiche zum Teil mehrerer Ecken dicht nebeneinander liegen. Für die Bestimmung der Orientierungen erwies sich ein gaußförmiges 11×11 Bildpunkte umfassendes Integrationsfenster mit einer Standardabweichung von 2.5 Bildpunkten als günstig. Für den Konfidenztest verwendeten wir die Konstanten $c_1 = 0.25$, $c_2 = 0.015$ und $\epsilon = 0.001$.

10.6 Zusammenfassung

In diesem Abschnitt haben wir Verfahren zur Bestimmung überlagerter Orientierungen in Bildern in Anlehnung an die zuvor vorgestellten Modelle zur Bewegungsbestimmung entwickelt. Auch bei überlagerten Orientierungen in Bildern ergeben sich die gemischten Orientierungsparameter aus der Eigensystemanalyse von Tensoren. Für zwei Überlagerungen stellten wir drei Verfahren zur Extraktion der Orientierungsparameter aus dem gemischten Orientierungsparametervektor vor und zeigten, dass der Winkel zwischen beiden Orientierungsvektoren rotations- und linear skalierungsinvariant ist und sich damit als ein gutes Merkmal für Klassifikations-, Matching- und Trackinganwendungen darstellt.

Die Konfidenzkriterien aus Abschnitt 3 konnten wir auf die Orientierungsanalyse in Bildern übertragen und somit Testkriterien auf keine, einfach oder doppelt orientierte Nachbarschaften angeben.

Neben dem additiven bzw. transparenten Überlagerungsmodell haben wir noch ein weiteres verdeckendes Modell vorgestellt, das in Zukunft auch für die Bewegungsbestimmung von Interesse sein könnte. Die Unterschiede beider Verfahren äußern sich nur in den Tensoreinträgen, die gemischten Orientierungsparameter sind in beiden Fällen identisch. Im verdeckenden Modell setzen sich die Tensoreinträge aus Produkten partieller Ableitungen erster Ordnung zusammen, während sie im transparenten Fall aus Produkten partieller Ableitungen zweiter Ordnung bestehen. Wie wir gesehen haben eignet sich das additive Modell auch für die Bestimmung verdeckender Orientierungen.

Ferner haben wir Methoden zur Trennung der orientierten Muster bzw. Regioneneinteilung vorgestellt und ihre Anwendbarkeit anhand einiger Beispiele demonstriert. Sowohl die Regioneneinteilung im verdeckenden Fall als auch eine Art Schichtentrennung in additiven Überlagerungen lassen sich durch die Richtungsableitungen in Richtung einer der gefundenen Orientierungen ermitteln. Zur Invertierung der Richtungsableitung und damit zur Rekonstruktion der Schichten stellten wir ein Verfahren im Frequenzraum vor.

Selbst mit einfachen Ableitungsfiltern konnten wir sehr gute Ergebnisse für reale und synthetische Beispiele erzielen. Dennoch könnten sich in Zukunft durch optimierte Filter, wie z.B. [134], die Rauschrobustheit erhöhen und die Ergebnisqualität weiter verbessern lassen.

In diesem Kapitel haben wir nur Lösungen für zwei Orientierungen vorgestellt. Eine Erweiterung zu mehr Orientierungen bleibt daher eine interessante Aufgabe, die es noch zu lösen gilt.

Kapitel 11

Diskussion

In dieser Arbeit wurden mehrere Verfahren zur Bestimmung transparent überlagerter Bewegungen in Bildsequenzen entwickelt. Dabei gelang es, bekannte Konzepte für einfache Bewegungen auf die Bestimmung einer beliebigen Anzahl transparenter Bewegungen auszuweiten und einen theoretischen Rahmen für die transparente Bewegungsbestimmung zu bilden. Für jede der drei Gruppen der differenziellen, vergleichs- und fourierbasierten Methoden der Bewegungsbestimmung konnten entsprechende Verfahren entworfen werden.

Bei einer beliebigen Anzahl transparenter Bewegungen treten in der erweiterten optischen Flussgleichung von Shizawa & Mase [147, 148] Nichtlinearitäten auf. Einen zentralen Punkt zur Lösung bildet die Linearisierung mit Hilfe der *gemischten Geschwindigkeitsparameter*. Nach ihrer Ermittlung aus den Bilddaten ergeben sich die gesuchten Bewegungsvektoren aus den Nullstellen eines komplexen Polynoms, dessen Koeffizienten durch die *gemischten Geschwindigkeitsparameter* beschrieben sind. Mit dieser Erkenntnis gelang es, Limitierungen auf zwei überlagerte Bewegungen, wie sie in der Vergangenheit oft notwendig waren, zu überwinden.

Diese Linearisierungsschritte und die damit verbundenen Nullstellenbestimmungen beschränken sich dabei nicht nur auf die differenziellen Methoden, sondern ermöglichen auch die Verallgemeinerung der phasenbasierten Methode von Vernon [174] auf eine beliebige Anzahl von Bewegungen sowie die Herleitung eines Blockmatchingverfahrens für transparente Bewegungen.

Durch diese Linearisierung werden allerdings gegebenenfalls auch Lösungen gefunden, die keine Lösung im Sinne der Bewegungen sind. Dies betrifft jedoch in der Regel nur die Fälle, in denen ein Aperturproblem in einer oder mehreren Schichten vorliegt und die Bestimmung aller Bewegungen ohnehin nicht eindeutig möglich ist. In den Bereichen, in denen alle Bewegungen bestimmbar sind, treten solche Lösungen nicht auf. Für zwei Bewegungen gelang es, ein Entscheidungskriterium zur Überprüfung, ob eine gefundene Lösung eine Lösung im Sinne der Bewegung ist, anzugeben. Die Herleitung entsprechender Kriterien für mehr als zwei Bewegungen bleibt jedoch noch Forschungsgegenstand für die Zukunft.

Im Bereich der differenziellen Methoden wurde hier zwei Verfahren vorgestellt. Bei dem

hierarchischen Verfahren werden lokale Energieausdrücke optimiert, während bei dem globalen Verfahren ein globaler Energieausdruck minimiert wird, um damit nicht einzelne Bewegungsvektoren, sondern dicht besetzte Bewegungsvektorfelder über der gesamten Bildebene zu bestimmen. Betont sei an dieser Stelle, dass das globale Energiekriterium letztlich lokal optimiert wird, die resultierenden Bewegungsfelder damit "hinreichend" glatt aber nicht grundsätzlich ideal glatt, d.h. konstant werden.

Mit dem hierarchischen Verfahren kann gegenüber dem globalen Verfahren direkt an jedem Punkt die Anzahl der gefundenen Bewegungen angegeben werden. Bei dem globalen Verfahren wird eine Anzahl von Bewegungen angenommen, die dann für jeden Punkt in der Sequenz gilt. Die Anzahl der Bewegungen in jedem Punkt ist somit erst durch Nachverarbeitungsschritte ermittelbar. Anhand einiger Experimente konnte auch für die transparenten Bewegungen ein schon für einfache Bewegung bekannter Effekt [67, 16] nachgewiesen werden: Die lokale Methode liefert qualitativ etwas bessere Ergebnisse als die globale, ergibt unter Umständen jedoch nur dünn besetzte Bewegungsfelder. Ferner wurde gezeigt, dass die Wahl der Ableitungsfilter in beiden Fällen die Qualität der Ergebnisse beeinflusst und die Filter daher sorgfältig für die jeweilige Anwendung auszuwählen sind.

Der Rechenaufwand der lokalen Methode ist in der Regel größer als für die globale, da für die Aufstellung der Tensoren viele Faltungen bzw. Integrationen notwendig sind. Zusätzlich werden bei der lokalen Methode die Ergebnisse durch die Einstellungen mehrerer Parameter, den Konfidenzparametern und denen des Integrationsfensters, beeinflusst, während die globale mit nur einem Regularisierungsparameter und einem Parameter, der die Anzahl der Iterationen angibt, auskommt. Für die Berechnung der Beispiele wurden die Parameter experimentell ermittelt, die Frage nach der optimalen Wahl wurde hier nicht verfolgt und bleibt noch Gegenstand nachfolgender Untersuchungen.

Anhand synthetischer und realer Beispiele wurde gezeigt, dass mit beiden Verfahren bei geeigneter Parameterwahl gute Ergebnisse erzielbar sind. Die differenziellen Methoden ermöglichen eine genaue Bewegungsbestimmung und eignen sich besonders für die Bestimmung langsamer Geschwindigkeiten. Bei schnellen Bewegungen (mehrere Bildpunkte pro Frame) sind Multiskalenansätze möglich.

Der differenzielle Ansatz wurde auch hinsichtlich Beleuchtungsänderungen erweitert. Dabei steigt die Qualität der Bewegungsbestimmung, wenn die Helligkeitsänderung gut mit dem angenommenen Beleuchtungsmodell übereinstimmt, wie sich bei der Untersuchung von additiven und multiplikativen Beleuchtungsmodellen ergab. Die hier betrachteten Beleuchtungsmodelle erlauben eine simultane Bestimmung der Bewegungen und der Parameter der Beleuchtungsfunktionen und lassen sich ohne großen Aufwand in das hierarchische Verfahren integrieren. Weiter zeigte sich, dass beim additiven Modell die Wahl der Ableitungsfilter einen vergleichbaren Einfluss auf die Ergebnisse hat wie bei dem transparenten Fall ohne Beleuchtungsänderung. Wesentlich stärker wirkt sich die Wahl der Filter im multiplikativen Modell aus. Die besten Ergebnisse konnten hier bei synthetischen Sequenzen mit den Filtern von Scharr [134, 135] erzielt werden, dennoch besteht weiterer Optimierungsbedarf, wenn dieser Ansatz auch für reale Daten genutzt werden soll.

Die phasenbasierte Methode zur Bestimmung einer beliebigen Anzahl transparenter Bewegungen eignet sich besonders zur Bewegungsbestimmung in Sequenzen, in denen nur globale Bewegungen zu ermitteln sind. Für eine lokale Bewegungsbestimmung ist sie weniger gut geeignet, da die Phasenverschiebungen nicht an allen Frequenz bestimmbar sind und somit größere Bildausschnitte mit konstanten Bewegungen fouriertransformiert werden müssen. Sie erlaubt, einen schnellen Eindruck über die Größenordnung der Geschwindigkeiten in einer Bildsequenz zu gewinnen, zumal sich mit ihr sowohl schnelle als auch langsame Bewegungen unter einer vorgegebenen Genauigkeit bestimmen lassen.

Im Anschluss an die Herleitung der phasenbasierten Methode wurde gezeigt, wie die gewonnenen Phaseninformationen für die Trennung der bewegten Schichten im Frequenzraum genutzt werden können. Leider ist die Trennung nicht an allen Frequenzen möglich, so dass die fehlenden Frequenzanteile der Schichten interpoliert werden müssen. Die Interpolation im Frequenzraum führt zu Fehlermustern der Schichten im Ortsraum und es bleibt die Frage zu klären, wie die fehlende Information optimal interpoliert werden kann. In der Praxis empfiehlt sich, die Schichtentrennung aufgrund von Änderungen im Bewegungsfeld eher lokal als nur global durchzuführen. Entsprechende Verbesserungen in dieser Richtung wären daher wünschenswert.

Blockmatching ist eine weit verbreitete Methode zur Bestimmung einfacher Bewegungen in Bildsequenzen und gilt als robust aber wenig genau. Die phasenbasierten Methoden bilden die Basis zur Herleitung einer Blockmatchingbedingung für eine beliebige Anzahl transparenter Bewegungen. Für diese Gleichungen stellten wir zwei Lösungsmethoden vor: eine hierarchische, bei der die Entscheidung über Annahme oder Ablehnung eines bestimmten Bewegungsmodells durch einen Signifikanztest erfolgt, und eine zweite basierend auf einem Markov-Zufallsfeld, die auch getroffene Entscheidungen für benachbarte Bildpunkte einschließt und bei der Optimierung berücksichtigt. Erwartungsgemäß erwies sich die Markov-Zufallsfeld-Methode in den Experimenten robuster als die hierarchische, ihr Aufwand fällt aber durch mehrfache Iterationen größer aus. Die in beiden Verfahren verwendete Optimierungsstrategie mittels einer vollständigen Suche garantiert zwar bei der hierarchischen Methode, die im Sinne des vorgegebenen Suchraums optimale Lösung zu finden, erweist sich aber als extrem rechenaufwendig. Um diese Methoden in praktischen Anwendungen einsetzbar zu machen, sind schnellere Optimierungsverfahren notwendig.

Ferner zeigte sich, dass die transparenten Methoden in der Regel nicht für die Bestimmung von Bewegungen an der Verdeckungsgrenze geeignet sind, da am Grenzübergang nicht verschwindende Störterme auftreten. Dies gilt sowohl für Verfahren im Ort als auch im Frequenzraum. Nur unter ganz bestimmten Konstellationen, die von den Bewegungen relativ zur Verdeckungsgrenze abhängen, verschwinden die Störterme und das verdeckende Modell verhält sich wie das transparente.

In dieser Arbeit ist die theoretische Basis für die Bestimmung transparenter Bewegungen in Bildsequenzen gelegt worden. Gleichzeitig wurde gezeigt, dass die Bestimmung transparenter Bewegungen nicht viel schwieriger ist als die Bestimmung einfacher Bewegungen

und dass bekannte Lösungsansätze für eine Bewegung auf mehrere transparente Bewegungen übertragbar sind, wie wir hier indirekt durch die Angabe der differentiellen, der phasenbasierten und der Blockmatichungsmethode gezeigt haben. Die jeweiligen Methoden besitzen daher auch ähnliche Eigenschaften wie ihre Gegenstücke aus der einfachen Bewegungsbestimmungsproblematik. Die Auswahl der jeweils am besten geeigneten Methode ist jedoch meist anwendungsabhängig. Anhand synthetischer und realen Beispielen wurden ihre praktischen Anwendbarkeiten demonstriert.

Aus der Erweiterung ergeben sich eine Reihe neuer Fragestellungen wie zum Beispiel die Wahl optimaler Ableitungsfiler und Parametereinstellungen, effizientere Suchmethoden für die blockbasierten Verfahren oder aber auch grundsätzlich neue Aufgaben zur Behandlung von Aperturproblemen in einer oder mehreren Schichten. Eine theoretische Basis ist gelegt, der Schritt in die praktischen Anwendungen muss noch vollzogen werden.

Während des Fortschreitens dieser Arbeit stellte sich Frage, ob das Konzept überlagerter Bewegungen auch auf die Bestimmung überlagerter Orientierungen in Bildern anwendbar ist. Eine Übertragung für zwei Orientierungen erwies sich als relativ einfach, die Bestimmung von mehr als zwei Orientierungen blieb jedoch ungelöst. Es wurden zwei Modelle zur Bestimmung von Doppelorientierungen in Bildern vorgestellt, ein transparentes, das auf partiellen Ableitungen zweiter Ordnung, und ein verdeckendes, das auf partiellen Ableitungen erster Ordnung basiert. Mit beiden Verfahren sind gute Ergebnisse für viele reale und synthetische Beispiele erreicht worden. Diese Beispiele verdeutlichen die hohe Relevanz der Mehrfaktororientierungsanalyse für eine Vielzahl von Anwendungen im Bereich der industriellen Bildverarbeitung und Computer Vision. Dieser Zweig wird wohl Forschungsgegenstand vieler weiterer Arbeiten werden.

Anhang A

Analytische Bewegungsbestimmung

Interessanterweise kann das Strukturtensorverfahren analytisch für bis zu vier Bewegungen gelöst werden. Dies klingt zunächst überraschend, denn für die Bewegungsbestimmung müssen Eigenwerte und -vektoren der Tensoren berechnet werden. Schon ab zwei Bewegungen erfordert die Eigenwertbestimmung von \mathbf{J}_2 numerische Methoden, da die Bestimmung der Eigenwerte in diesem Fall mit der Nullstellenbestimmung eines Polynoms vom Grad sechs einhergeht. Formeln zur Nullstellenbestimmung sind nur für Polynome bis zum Grad vier bekannt. Offensichtlich ist der auf vier Bewegungen limitierende Faktor nicht die Eigenwertbestimmung der Tensoren, sondern die Separierung der Bewegungsvektoren aus den gemischten Geschwindigkeitsparametervektoren. In diesem Abschnitt werden wir zeigen, wie sich der Eigenvektor zum Eigenwert null und die Konfidenzmaße ohne eine explizite Bestimmung der Eigenwerte der Tensoren berechnen lassen.

A.1 Berechnung des Eigenvektors

Neben der klassischen numerischen Möglichkeit kann der Eigenvektor einer Matrix zum Eigenwert null auch durch die Unterdeterminanten bestimmt werden. Die Bestimmung aller Unterdeterminanten einer Matrix ist in den meisten Fällen aufwendiger als die Eigenwertbestimmung und auch aus numerischer Sicht nicht immer empfehlenswert. Bei Matrizen bis zur einer Größe von 3×3 lassen sich die Unterdeterminanten einfach und schnell berechnen. Beispielsweise gelang es Barth [17], das Tensorverfahren für eine Bewegung durch Berechnung der Unterdeterminanten zu beschleunigen. Im Falle von zwei oder mehr überlagerten Bewegungen sind die Tensoren größer als 3×3 und damit steigt auch der Rechenaufwand für die Bestimmung der Unterdeterminanten.

Die Determinante einer Matrix wird nach einer ihrer Zeilen entwickelt. Diese Determinante kann als inneres Produkt zweier Vektoren, dem Zeilenvektor der Matrix und einem Vektor \mathbf{X} , der als Elemente die Unterdeterminanten mit ihren Kofaktoren enthält, aufgefasst werden

$$\mathbf{X} = (M_{im}, -M_{im-1}, \dots, (-1)^m M_{i1})^T, \quad i = 1, \dots, \text{ord}(\mathbf{J}) = m. \quad (\text{A.1})$$

Genau wie bei dem Strukturtensor für eine Bewegung muss die letzte Komponente des gemischten Geschwindigkeitsvektors \mathbf{X} eins ergeben. Teilt man den Vektor \mathbf{X} durch seine

letzte Komponente, so erhält man die gesuchten gemischten Geschwindigkeitskomponenten

$$\mathbf{X}_{\text{gesucht}} = \left(\frac{M_{im}}{(-1)^m M_{i,1}}, \frac{-M_{im-1}}{(-1)^m M_{i,1}}, \dots, 1 \right)^T, \quad i = 1, \dots, \text{ord}(\mathbf{J}) = m, \quad (\text{A.2})$$

aus denen dann die Einzelgeschwindigkeiten \mathbf{v}_i zu bestimmen sind. Zum Abschluss dieses Abschnittes sei der Satz angegeben:

Satz 9 *Sei \mathbf{A} eine quadratische Matrix der Größe $n \times n$ mit dem minimalen Eigenwert null, so kann der Eigenvektor zum Eigenwert null durch die Unterdeterminanten ausgedrückt werden.*

Beweis

Mit A_i $i = 1, \dots, n$ werden die Zeilen der Matrix A bezeichnet

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_n \end{pmatrix}. \quad (\text{A.3})$$

Der Eigenvektor \mathbf{X} zum Eigenwert null muss orthogonal zu allen Zeilen in der Matrix sein, d.h.

$$\mathbf{A}_1 \mathbf{X} = 0, \dots, \mathbf{A}_n \mathbf{X} = 0. \quad (\text{A.4})$$

Ist ein Eigenwert null, so ist auch die Determinante der Matrix A gleich null. Wird nun die Determinante nach der letzten Zeile der Matrix entwickelt, so gilt

$$\det(\mathbf{A}) = \sum_{i=1}^n (-1)^{i+n} a_{in} M_{(n-i+1)1} = 0, \quad (\text{A.5})$$

wobei M_{ij} die Unterdeterminante durch Streichen der $(n - i + 1)$ ten Zeile und der $(n - j + 1)$ ten Spalte ist. Diese Summe kann als inneres Produkt zweier Vektoren geschrieben werden

$$\mathbf{A}_n \mathbf{X} = 0, \quad \text{mit } \mathbf{X} = ((-1)^{1+n} M_{n1}, (-1)^{2+n} M_{(n-1)1}, \dots, (-1)^{n+n} M_{11})^T. \quad (\text{A.6})$$

Ist ein Eigenwert null, so gibt es eine linear abhängige Zeile in der Matrix A . Ersetzt man die letzte Zeile A_n durch die erste Zeile A_1 , so gilt für die Determinante

$$\det(\mathbf{A}) = \begin{vmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_{n-1} \\ \mathbf{A}_1 \end{vmatrix} = 0. \quad (\text{A.7})$$

Damit sind der Vektor \mathbf{X} und die Zeile \mathbf{A}_1 orthogonal zueinander. Anstelle von \mathbf{A}_1 können in der letzten Zeile die Zeilenvektoren $\mathbf{A}_2, \dots, \mathbf{A}_{n-1}$ eingetragen werden und die Determinante ist jedesmal null. Damit ist gezeigt worden, dass der Vektor \mathbf{X} zu allen Zeilenvektoren \mathbf{A}_i orthogonal ist und daher einen Eigenvektor zum Eigenwert null darstellt. Diese Vorgehensweise kann nun für jede Zeile der Matrix angewandt werden und somit kann der Eigenvektor zum Eigenwert null durch die Unterdeterminanten der Matrix \mathbf{A} ausgedrückt werden. \square

A.2 Der gemittelte Eigenvektor

Bisher ist nur ein Eigenvektor nach einer beliebigen Zeile bestimmt worden. Wird nun für jede Zeile ein Eigenvektor bestimmt, so erhält man $m = \text{ord}(\mathbf{J}_N)$ Eigenvektoren, aus denen ein Mittelwert gebildet werden kann. Vor der Mittelwertbestimmung sollen noch einige Eigenschaften der Geschwindigkeitsvektoren, die aus den Eigenvektoren bestimmt worden sind, dargestellt werden. Dies wird anhand eines Beispiels für eine Bewegung beschrieben: In diesem Fall erhalten wir die drei Eigenvektoren

$$\begin{aligned} \mathbf{e}_1 &= (M_{31}, -M_{21}, M_{11})^T \\ \mathbf{e}_2 &= (M_{23}, -M_{22}, M_{12})^T \\ \mathbf{e}_3 &= (M_{33}, -M_{23}, M_{13})^T. \end{aligned} \tag{A.8}$$

Da die letzten Komponenten der Geschwindigkeitsvektoren eins ergeben müssen, müssen die Eigenvektoren jeweils durch ihre letzte Komponente geteilt werden. Damit lauten die Geschwindigkeitsvektoren

$$\begin{aligned} \mathbf{v}_1 &= \left(\frac{M_{31}}{M_{11}}, -\frac{M_{21}}{M_{11}}, 1 \right)^T \\ \mathbf{v}_2 &= \left(\frac{M_{23}}{M_{12}}, -\frac{M_{22}}{M_{12}}, 1 \right)^T \\ \mathbf{v}_3 &= \left(\frac{M_{33}}{M_{13}}, -\frac{M_{23}}{M_{13}}, 1 \right)^T. \end{aligned} \tag{A.9}$$

Kritisch wird diese Vorgehensweise, wenn die Bewegung entlang einer Achsenrichtung (x - oder y -Achse) erfolgt. Nehmen wir nun an, dass eine Bewegung entlang der x -Achse vorliegt, z.B. $\mathbf{v} = (1, 0, 1)^T$, so ist die M_{21} -Komponente null. Aufgrund der Symmetrie der Unterdeterminanten ist $M_{21} = M_{12}$. Damit ist die letzte Komponente des Eigenvektors \mathbf{e}_2 null und eine Division ist nicht mehr möglich. Der Geschwindigkeitsvektor \mathbf{v}_2 kann in diesem Fall nicht bestimmt werden. Gesucht ist nun ein Verfahren, das den Mittelwert unter Berücksichtigung dieses Aspektes bildet.

An dieser Stelle wird die Vorgehensweise für beliebige Dimensionen angegeben. Sei m die Dimension von \mathbf{J}_N und seien \mathbf{e}_i mit $i = 1, \dots, m$ die Eigenvektoren entwickelt nach den m Zeilen, so kann der gemittelte Komponentenvektor \mathbf{X} nach der Formel

$$\mathbf{X} = a_1 \mathbf{e}_1 + a_2 \mathbf{e}_2 + \dots + a_m \mathbf{e}_m \tag{A.10}$$

ermittelt werden, wobei die a_i Gewichtungsfaktoren darstellen. Die Frage ist, wie diese Gewichtungsfaktoren zu wählen sind, damit der gewünschte Effekt erreicht wird. Hierbei hilft die Tatsache, dass die letzte Komponente des Geschwindigkeitsvektors eins ergeben muss. Für die letzte Komponente von \mathbf{X} muss die Gleichung

$$\begin{aligned} 1 &= a_1 M_{11} + a_2 M_{21} + \dots + a_m M_{m1} \\ &= a_1 M_{11} + a_2 M_{12} + \dots + a_m M_{1m} \end{aligned} \tag{A.11}$$

gelten. Aus dieser Gleichung können nun sofort die Faktoren a_i abgelesen werden

$$\begin{aligned} a_1 &= \frac{M_{11}}{M_{11}^2 + M_{12}^2 + \cdots + M_{1m}^2} \\ a_2 &= \frac{M_{12}}{M_{11}^2 + M_{12}^2 + \cdots + M_{1m}^2} \\ &\vdots \\ a_m &= \frac{M_{1m}}{M_{11}^2 + M_{12}^2 + \cdots + M_{1m}^2}. \end{aligned} \tag{A.12}$$

Sollte nun eine Komponente M_{1i} klein oder null sein, so ist der zugehörige Gewichtungsfaktor ebenfalls klein oder gar null und der entsprechende Eigenvektor geht nicht mehr so stark in die Mittelwertbildung ein oder wird im Falle der Null sogar aus der Mittelwertbildung ausgeschlossen.

A.3 Analytische Konfidenzmaßberechnung

Wie wir gerade gesehen haben kann der Eigenvektor zum einzigen Eigenwert null durch die Unterdeterminanten der Matrix bestimmt werden. Diese Aussage trifft nur dann zu, wenn die Matrix genau einen Eigenwert gleich null besitzt, eine Voraussetzung die wir ebenfalls für die erfolgreiche Bewegungsbestimmung benötigen. Überprüfen können wir die Erfüllbarkeit der Voraussetzung mit den Konfidenzmaßen

$$K = \det(\mathbf{J}_N) = \lambda_1 \lambda_2 \cdots \lambda_m \tag{A.13}$$

$$H = \frac{1}{m} \text{trace}(\mathbf{J}_N) = \frac{1}{m} (\lambda_1 + \cdots + \lambda_m) \tag{A.14}$$

$$S = \frac{1}{m} \text{trace}(\mathbf{M}) = \frac{1}{m} \sum_i \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m, \tag{A.15}$$

wobei die λ_i die Eigenwerte des Systems sind und $\lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_m = \lambda_1 \cdots \lambda_{i-1} \lambda_{i+1} \cdots \lambda_m$ ist. Die Matrix \mathbf{M} entspricht der Unterdeterminantenmatrix von \mathbf{J}_N , deren Einträge \mathbf{M}_{ij} durch Streichen der $(m - i + 1)$ -ten Zeile und $(m - j + 1)$ -ten Spalte von \mathbf{J}_N entstanden sind. Die Spur von \mathbf{J}_N sowie die Determinante sind analytisch bestimmbar, entsprechend gilt dann auch für die Spur der Unterdeterminantenmatrix \mathbf{M} .

Damit haben wir gezeigt, dass für bis zur vier transparent überlagerte Bewegungen eine analytische Lösung des Tensorverfahrens existiert. Der auf vier Bewegungen limitierende Faktor ist die Separierung der Bewegungsvektoren aus dem gemischten Geschwindigkeitsparametervektor, die auf einer Nullstellenbestimmung eines komplexen Polynoms basiert. Mit der Anzahl der Bewegungen wächst die Ordnung des Polynoms, dessen Nullstellen ab fünf Bewegungen mit numerischen Verfahren ermittelt werden müssen, da für Polynome vom Grad fünf oder mehr keine geschlossenen Formeln für die Nullstellenbestimmung bekannt sind.

Anhang B

Separierungsverfahren von Shizawa & Mase

Satz 10 Erfüllt \mathbf{c} angeordnet in der Matrix \mathbf{C}_s die Bedingungen (2.56), so ergeben sich die gesuchten Orientierungsvektoren \mathbf{u} und \mathbf{v} , mit $\|\mathbf{u}\| = \|\mathbf{v}\| = 1$ durch

$$\begin{aligned}\mathbf{u} &= \sqrt{\lambda_1} \mathbf{e}_1 + \sqrt{-\lambda_2} \mathbf{e}_2 \\ \mathbf{v} &= \sqrt{\lambda_1} \mathbf{e}_1 - \sqrt{-\lambda_2} \mathbf{e}_2,\end{aligned}\tag{B.1}$$

wobei λ_1, λ_2 die von null verschiedenen und $\mathbf{e}_1, \mathbf{e}_2$ die korrespondierenden Eigenvektoren der Matrix \mathbf{C}_s aus Gleichung (2.55) sind.

Beweis:

Wie wir bereits in der Fourieranalyse in Gleichung (2.48) und (2.49) gesehen haben, muss an den Stellen mit $F(\boldsymbol{\omega}) \neq 0$ der Ausdruck $\boldsymbol{\omega}^T \frac{1}{2} (\mathbf{u} \mathbf{v}^T + \mathbf{v} \mathbf{u}^T) \boldsymbol{\omega} = \boldsymbol{\omega}^T \mathbf{C}_s \boldsymbol{\omega} = 0$ sein. Er stellt eine quadratische Form dar, die mittels ihrer Eigenvektoren diagonalisiert werden kann. Zur Diagonalisierung verwenden wir die Matrix \mathbf{P} , deren Spaltenvektoren mit den Eigenvektoren von \mathbf{C}_s übereinstimmen. Nach der Transformation auf Diagonalfom seien die x_i die neuen Variablen und Gleichung (2.48) schreibt sich als

$$\mathbf{x}^T \mathbf{P}^T \mathbf{C}_s \mathbf{P} \mathbf{x} F(\mathbf{P} \mathbf{x}) = 0,\tag{B.2}$$

mit der Diagonalmatrix $\mathbf{P}^T \mathbf{C}_s \mathbf{P}$. Von Interesse sind allerdings nur die x_i mit Eigenwerten von \mathbf{C}_s ungleich null. Wir bezeichnen sie im Folgenden mit x_1 und x_2 . In dem neuen Koordinatensystem schreibt sich die quadratische Gleichung unter der Berücksichtigung, dass λ_2 negativ ist, als

$$\lambda_1 x_1^2 + \lambda_2 x_2^2 = 0 \Leftrightarrow (\sqrt{\lambda_1} x_1 + \sqrt{-\lambda_2} x_2)(\sqrt{\lambda_1} x_1 - \sqrt{-\lambda_2} x_2) = 0,\tag{B.3}$$

d.h. $(\sqrt{\lambda_1} x_1 + \sqrt{-\lambda_2} x_2) = 0$ oder $(\sqrt{\lambda_1} x_1 - \sqrt{-\lambda_2} x_2) = 0$ falls $F(\mathbf{P} \mathbf{x}) \neq 0$, damit Gleichung (B.2) erfüllt ist. Der Vergleich mit Gleichung (2.48) zeigt, dass es sich hier im Wesentlichen um die gesuchten Orientierungsvektoren handelt, nur in unterschiedlichen Koordinatensystemen. In beiden Gleichungen werden zwei Skalarprodukte miteinander multipliziert. In dem neuen Koordinatensystem sind die beiden Vektoren \mathbf{u} und \mathbf{v} durch

$$\begin{aligned}\mathbf{u} &= (\sqrt{\lambda_1}, \sqrt{-\lambda_2}) \\ \mathbf{v} &= (\sqrt{\lambda_1}, -\sqrt{-\lambda_2})\end{aligned}\tag{B.4}$$

gegeben ¹. Wird die Koordinatentransformation mittels der Eigenvektoren, d.h. mit $\boldsymbol{\omega} = \mathbf{P}\mathbf{x}$, rückgängig gemacht, so sind

$$\begin{aligned}\mathbf{u} &= \sqrt{\lambda_1}\mathbf{e}_1 + \sqrt{-\lambda_2}\mathbf{e}_2 \\ \mathbf{v} &= \sqrt{\lambda_1}\mathbf{e}_1 - \sqrt{-\lambda_2}\mathbf{e}_2\end{aligned}\tag{B.5}$$

die Orientierungsvektoren in den alten Koordinaten. Ferner entsprechen $\mathbf{e}_1 = (\mathbf{u} + \mathbf{v})/\|\mathbf{u} + \mathbf{v}\|$ und $\mathbf{e}_2 = (\mathbf{u} - \mathbf{v})/\|\mathbf{u} - \mathbf{v}\|$ den normalisierten Eigenvektoren aus Satz 3.

□

¹An dieser Stelle sei bemerkt, dass wir im Bereich der Orientierungsanalyse nur an Vektoren der Länge eins interessiert sind. Sollten die Einträge der Matrix \mathbf{C}_s eine Lösung repräsentieren, in der die Vektoren \mathbf{u} und \mathbf{v} nicht die Länge eins besitzen, so bietet sich an dieser Stelle eine Normalisierung mit $\|\mathbf{u}\| = \|\mathbf{v}\| = \sqrt{\lambda_1 - \lambda_2}$ an.

Anhang C

Algorithmen für eine, zwei und drei Bewegungen

Einen hierarchischen Algorithmus, der die Bestimmung von ein, zwei und drei Bewegungen ermöglicht, hatten wir in Kapitel 3 kennengelernt. In diesem Abschnitt geben wir alle Algorithmen noch einmal als eigenständige Algorithmen an. Insbesondere legen wir bei ihrer Angabe Wert auf die Beschreibung der für die Separierung der Bewegungsvektoren aus dem gemischten Geschwindigkeitsparametervektor notwendigen Schritte, die in dem hierarchischen Algorithmus nicht gegeben wurden.

C.1 Algorithmus für eine Bewegung

Das Vorgehen für eine Bewegung ist in Algorithmus (3) aufgeführt. In Zeile 1 werden erst alle partiellen Ableitungen gebildet, die dann in Zeile 2 miteinander multipliziert und über eine kleine Nachbarschaft gemittelt, d.h. mit einem kleinen Faltungskern gefaltet werden. Die Bestimmung aller $H(x, y, t)$ erfolgt in Zeile 3. Diese H 's lassen sich leicht durch die Formel $f_x f_x + f_y f_y + f_t f_t$ berechnen. Es ist daher nicht notwendig, den Strukturtenor \mathbf{J} aufzustellen, um H zu berechnen. Die Zeilen 5 bis 17 bilden die Hauptschleife für die Bewegungsbestimmung, die für jeden Punkt in der Sequenz durchlaufen wird. In Zeile 6 überprüfen wir, ob $H(x, y, t)$ größer als ein kleiner vorgegebener Wert ist, um zu verhindern, dass die Bewegungsbestimmung in einem Bereich mit homogener Grauwertstruktur durchgeführt wird. In Zeile 10 werden die Geschwindigkeiten durch Teilen der ersten beiden Komponenten des Eigenvektors zum kleinsten Eigenwert λ_1 durch die letzte bestimmt. Soll anstelle des Bewegungsvektors ein Orientierungsvektor ermittelt werden, so repräsentiert \mathbf{e} bereits den Orientierungsvektor.

C.2 Algorithmus für zwei Bewegungen

Der Algorithmus (4) für zwei Bewegungen ist dem für eine sehr ähnlich. In Zeile 10 werden die Geschwindigkeitskomponenten c_I bestimmt, in dem sie durch die Komponente c_{tt} geteilt werden. Danach werden in den Zeilen 11 bis 14 die Geschwindigkeitsvektoren aus den gemischten Geschwindigkeitskomponenten c_I bestimmt.

Algorithm 3 Eine Bewegung

```

1: Bestimme die ersten partiellen Ableitungen  $f_x, f_y$  und  $f_t$ 
2: Bestimme die Produkte  $f_x f_x, f_x f_y, \dots, f_t f_t$  und middle diese um eine kleine Nachbarschaft (Faltung)
3:  $H(x, y, t) = \text{trace}(J_1(x, y, t))$  für alle Punkte  $(x, y, t)$ 
4: for all Punkte  $(x, y, t)$  do
5:   Bestimme die Eigenwerte  $\lambda_1 \leq \lambda_2 \leq \lambda_3$  und den zu  $\lambda_1$  korrespondierenden Eigenvektor  $\mathbf{e}$  von  $J_1(x, y, t)$ 
6:   if  $H(x, y, t) > \epsilon_1$  then
7:      $K = \det(J_1(x, y, t)) = \lambda_1 \lambda_2 \lambda_3$ 
8:      $S = \lambda_1 \lambda_2 + \lambda_1 \lambda_3 + \lambda_2 \lambda_3$ 
9:     if  $K^{\frac{1}{3}} < \epsilon_2 S^{\frac{1}{2}}$  then
10:       $\mathbf{v}(x, y, t) = \begin{pmatrix} \mathbf{e}_x & \mathbf{e}_y \\ \mathbf{e}_t & \mathbf{e}_t \end{pmatrix}^T$ 
11:    else
12:       $\mathbf{v}(x, y, t) = \mathbf{0}$ 
13:    end if
14:  else
15:     $\mathbf{v}(x, y, t) = \mathbf{0}$ 
16:  end if
17: end for

```

Algorithm 4 Zwei Bewegungen

```

1: Bestimme die zweiten partiellen Ableitungen  $f_{xx}, f_{xy}, f_{xt}, f_{yy}, f_{yt}$  und  $f_{tt}$ 
2: Berechne die Produkte  $f_{xx} f_{xx}, f_{xx} f_{xy}, \dots, f_{tt} f_{tt}$  und middle diese über eine Nachbarschaft (Faltung)
3: Berechne  $H(x, y, t) = \text{trace}(J_2(x, y, t))$  für alle Punkte  $(x, y, t)$ 
4: for all Punkte  $(x, y, t)$  do
5:   Bestimme alle Eigenwerte  $\lambda_1 \leq \dots \leq \lambda_6$  und den zu  $\lambda_1$  korrespondierenden Eigenvektor  $\mathbf{c}$  von  $J_2(x, y, t)$ 
6:   if  $H(x, y, t) > \epsilon_1$  then
7:      $K = \det(J_2(x, y, t))$ 
8:      $S = \sum_{i=1}^6 \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_6$ , mit  $\hat{\lambda}_i = 1$ 
9:     if  $K^{\frac{1}{6}} < \epsilon_2 S^{\frac{1}{5}}$  then
10:       $c_{xx} = c_{xx}/c_{tt}, c_{yy} = c_{yy}/c_{tt}, c_{xt} = c_{xt}/c_{tt}, c_{yt} = c_{yt}/c_{tt}, c_{xy} = c_{xy}/c_{tt}$ 
11:       $a = c_{xt} + i c_{yt}$ 
12:       $b = c_{xx} - c_{yy} + i c_{xy}$ 
13:      Bestimme die Nullstellen  $(x_1, x_2)$  von  $x^2 - ax + b$ 
14:       $\mathbf{v}_1(x, y, t) = \begin{pmatrix} \text{Re}(x_1) \\ \text{Im}(x_1) \end{pmatrix}, \mathbf{v}_2(x, y, t) = \begin{pmatrix} \text{Re}(x_2) \\ \text{Im}(x_2) \end{pmatrix}$ 
15:    else
16:       $\mathbf{v}_1(x, y, t) = \mathbf{0}, \mathbf{v}_2(x, y, t) = \mathbf{0}$ 
17:    end if
18:  else
19:     $\mathbf{v}_1(x, y, t) = \mathbf{0}, \mathbf{v}_2(x, y, t) = \mathbf{0}$ 
20:  end if
21: end for

```

C.3 Algorithmus für drei Bewegungen

Der Algorithmus (5) für drei Bewegungen ist dem Algorithmus für zwei Bewegungen sehr ähnlich. Die Hauptunterschiede liegen in der Ordnung der partiellen Ableitungen, der Größe des Tensors, sowie dem Verfahren zur Separierung der Bewegungsvektoren aus dem gemischten Geschwindigkeitsparametervektor.

Algorithm 5 Drei Bewegungen

- 1: Bestimme die dritten partiellen Ableitungen $f_{xxx}, f_{xxt}, f_{xtt}, f_{yyy}, f_{yyt}, f_{ytt}, f_{xyt}, f_{xxy}, f_{xyy}$ und f_{ttt}
 - 2: Berechne die Produkte $f_{xxx}f_{xxx}, f_{xxx}f_{xxy}, \dots, f_{ttt}f_{ttt}$ und middle sie über eine Nachbarschaft (Faltung)
 - 3: Berechne $H(x, y, t) = \text{trace}(J_3(x, y, t))$ für alle Punkte (x, y, t)
 - 4: **for all** Punkte (x, y, t) **do**
 - 5: Bestimme alle Eigenwerte $\lambda_1 \leq \dots \leq \lambda_{10}$ und den zu λ_1 korrespondierenden Eigenvektor \mathbf{c} von $J_3(x, y, t)$
 - 6: **if** $H(x, y, t) > \epsilon_1$ **then**
 - 7: $K = \det(J_3(x, y, t))$
 - 8: $S = \sum_{i=1}^{10} \lambda_1 \cdots \hat{\lambda}_i \cdots \lambda_{10}$, mit $\hat{\lambda}_i = 1$
 - 9: **if** $K^{\frac{1}{10}} < \epsilon_2 S^{\frac{1}{9}}$ **then**
 - 10: $c_{xxx} = c_{xxx}/c_{ttt}, c_{xxt} = c_{xxt}/c_{ttt}, c_{xtt} = c_{xtt}/c_{ttt}, c_{yyy} = c_{yyy}/c_{ttt}, c_{yyt} = c_{yyt}/c_{ttt},$
 $c_{ytt} = c_{ytt}/c_{ttt}, c_{xyt} = c_{xyt}/c_{ttt}, c_{xxy} = c_{xxy}/c_{ttt}, c_{xyy} = c_{xyy}/c_{ttt}$
 - 11: $a = c_{xtt} + ic_{ytt}$
 - 12: $b = c_{xxt} - c_{yyt} + ic_{xyt}$
 - 13: $c = c_{xxx} - c_{xyy} + i(c_{xxy} - c_{yyy})$
 - 14: Bestimme die Nullstellen (x_1, x_2, x_3) des Polynoms $x^3 - ax^2 + bx - c$
 - 15: $\mathbf{v}_1(x, y, t) = \begin{pmatrix} \text{Re}(x_1) \\ \text{Im}(x_1) \end{pmatrix}, \mathbf{v}_2(x, y, t) = \begin{pmatrix} \text{Re}(x_2) \\ \text{Im}(x_2) \end{pmatrix}, \mathbf{v}_3(x, y, t) = \begin{pmatrix} \text{Re}(x_3) \\ \text{Im}(x_3) \end{pmatrix}$
 - 16: **else**
 - 17: $\mathbf{v}_1(x, y, t) = \mathbf{0}, \mathbf{v}_2(x, y, t) = \mathbf{0}, \mathbf{v}_3(x, y, t) = \mathbf{0}$
 - 18: **end if**
 - 19: **else**
 - 20: $\mathbf{v}_1(x, y, t) = \mathbf{0}, \mathbf{v}_2(x, y, t) = \mathbf{0}, \mathbf{v}_3(x, y, t) = \mathbf{0}$
 - 21: **end if**
 - 22: **end for**
-

Anhang D

Mehrdimensionale δ -Funktionen

In dem Abschnitt über verdeckende Bewegungen (Kapitel 10) treten mehrdimensionale δ -Funktionen auf. Einige dort benutzte Eigenschaften seien hier kurz dargestellt, weiterführende Aussagen finden sich zum Beispiel in [15].

Die dort symbolisch als Produkte von δ -Funktionen bezeichneten Ausdrücke stellen keine Produkte im engeren Sinne dar, denn Produkte von Distributionen sind im Allgemeinen nicht erklärt. Diese Ausdrücke beziehen sich im Zweidimensionalen auf die Schnittpunkte von Linien und im Dreidimensionalen auf die Schnittlinien von Flächen.

Eine δ -Funktionen ist normiert

$$\int \delta(x) dx = 1. \quad (\text{D.1})$$

Werden im Argument Substitutionen vorgenommen, so ist nach der Integration der Wert im Allgemeinen nicht mehr eins. Die δ -Funktion durch diesen Wert geteilt führt dann wieder zu einer Normierung. In Anlehnung an die Geometrie bezeichnet man diesen Wert auch als Querschnitt der Schnittpunkte oder Schnittlinien.

Zunächst möchten wir $\delta(a_1(\mathbf{x}))\delta(a_2(\mathbf{x}))$ genauer erläutern. Im Zweidimensionalen beschreiben die beiden Funktionen jeweils eine Linie, die sich in einem Punkt oder bei gekrümmten Linien auch in mehreren schneiden. Dieser Ausdruck ist in einem betrachteten Punkt x_0 nur dann ungleich null, wenn x_0 innerhalb der Schnittmenge P_S der beiden Linien liegt, d.h.

$$\delta(a_1(\mathbf{x}_0))\delta(a_2(\mathbf{x}_0)) \neq 0 \quad (\text{D.2})$$

für $\mathbf{x}_0 \in P_S$. Der Querschnitt des Schnittpunktes zweier δ -Linien ist an dem Punkt $\mathbf{x}_0 \in P_S$ durch

$$d(\mathbf{x}_0 \in P_S) = |\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)| \quad (\text{D.3})$$

gegeben und wir können das Produkt der beiden Dirac-Linien vereinfacht schreiben durch

$$\delta(a_1(\mathbf{x}))\delta(a_2(\mathbf{x})) = \frac{1}{|\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)|} \delta(\mathbf{x} - \mathbf{x}_0). \quad (\text{D.4})$$

Das Kreuzprodukt beschreibt die von zwei Vektoren eingeschlossene Fläche. Zu beachten ist jedoch, dass die Gradientenvektoren jeweils um eine dritte Komponente, die null ist,

erweitert wurden, damit obige Formel gültig ist. Im Dreidimensionalen stellen die beiden δ -Funktionen zwei Flächen dar, die sich in einer oder mehreren δ -Linien S schneiden. Analog zu dem Schnitt zweier δ -Linien errechnet sich der Querschnitt der Schnittlinien in einem betrachteten Punkt $\mathbf{x}_0 \in S$ durch

$$d(\mathbf{x}_0 \in P_S) = |\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)| \quad (\text{D.5})$$

und der Richtungsvektor \mathbf{l} der Schnittlinie durch

$$\mathbf{l}(\mathbf{x}_0 \in S) = \nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0). \quad (\text{D.6})$$

Demnach sind die δ -Einheitspunkte bzw. δ -Einheitslinien von der Form

$$|\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)| \delta(a_1(\mathbf{x})) \delta(a_2(\mathbf{x})). \quad (\text{D.7})$$

Anschließend betrachten wir den Schnitt dreier δ -Flächen $\delta(a_1(\mathbf{x}))$, $\delta(a_2(\mathbf{x}))$ und $\delta(a_3(\mathbf{x}))$ in dem Punkt \mathbf{x}_0 . Die ersten beiden Flächen bilden eine δ -Linie, die im Schnittpunkt \mathbf{x}_0 den Richtungsvektor $\mathbf{l}(\mathbf{x}_0)$ hat und den Querschnitt $|\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)|$. Weiter beschreibt ϕ den Winkel zwischen dem Richtungsvektor $\mathbf{l}(\mathbf{x}_0)$ und dem Normalenvektor $\nabla a_3(\mathbf{x}_0)$. Das Volumen des δ -Punktes ist abhängig von dem Winkel und dem Normalenvektor und damit ergibt sich der Querschnitt

$$\begin{aligned} (|\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)| |\nabla a_3(\mathbf{x}_0)| \cos \phi(\mathbf{x}_0)) \\ = |(\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)) \cdot \nabla a_3(\mathbf{x}_0)|. \end{aligned} \quad (\text{D.8})$$

Das Produkt der drei Dirac-Linien lässt sich vereinfacht schreiben durch

$$\delta(a_1(\mathbf{x})) \delta(a_2(\mathbf{x})) \delta(a_3(\mathbf{x})) = \frac{1}{|(\nabla a_1(\mathbf{x}_0) \times \nabla a_2(\mathbf{x}_0)) \cdot \nabla a_3(\mathbf{x}_0)|} \delta(\mathbf{x} - \mathbf{x}_0). \quad (\text{D.9})$$

Das Spatprodukt beschreibt das von drei Vektoren eingeschlossene Volumen.

Anhang E

Spektrum der Indikatorfunktion

Im Verdeckungsabschnitt (Kapitel 9) sind wir immer davon ausgegangen, dass das Spektrum der Funktion

$$\chi(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \cdot \mathbf{N} \geq 0 \\ 0 & \text{sonst} \end{cases} \quad (\text{E.1})$$

mit $\mathbf{x} = (x, y)^T$ gleich

$$\hat{\chi}(\boldsymbol{\omega}) = \frac{1}{2}\delta(\boldsymbol{\omega}) + \frac{1}{j2\pi\mathbf{N} \cdot \boldsymbol{\omega}}\delta(\mathbf{N}^\perp \cdot \boldsymbol{\omega}). \quad (\text{E.2})$$

ist. Hier möchten wir kurz diese Beziehung nachweisen.

Im Eindimensionalen besitzt die Sprungfunktion

$$\epsilon(t) = \begin{cases} 1 & t \geq 0 \\ 0 & \text{sonst} \end{cases} \quad (\text{E.3})$$

das Spektrum

$$\frac{1}{2}\delta(f) + \frac{1}{j2\pi f}. \quad (\text{E.4})$$

Die Fouriertransformation der zweidimensionalen Funktion

$$f(\mathbf{x}) = f(x, y) = \epsilon(x) \quad (\text{E.5})$$

ist durch

$$F(\boldsymbol{\omega}) = F(\omega_x, \omega_y) = \left(\frac{1}{2}\delta(\omega_x) + \frac{1}{j2\pi\omega_x}\right)\delta(\omega_y). \quad (\text{E.6})$$

gegeben. Im nächsten Schritt möchten wir die Funktion $\chi(\mathbf{x})$ durch die Funktion $f(\mathbf{x})$ darstellen. Dazu ordnen wir die beiden Vektoren $\mathbf{N} = (N_x, N_y)^T$ und $\mathbf{N}^\perp = (N_y, -N_x)^T$ in der Matrix

$$\mathbf{A} = \begin{pmatrix} \mathbf{N} \\ \mathbf{N}^\perp \end{pmatrix} \quad (\text{E.7})$$

an und erhalten durch

$$f(\mathbf{Ax}) = f(\mathbf{N} \cdot \mathbf{x}, \mathbf{N}^\perp \cdot \mathbf{x}) = \epsilon(\mathbf{N} \cdot \mathbf{x}) = \chi(\mathbf{x}) \quad (\text{E.8})$$

die gesuchte Darstellung der Funktion $\chi(\mathbf{x})$. Die Determinante der Matrix \mathbf{A} hat den Wert minus eins. Interessanterweise ist \mathbf{A} symmetrisch und zu sich selbst invers, d.h. $\mathbf{A}^{-1} = \mathbf{A} = \mathbf{A}^T$. Zur Bestimmung des Spektrums von $f(\mathbf{A}\mathbf{x})$ nutzen wir aus, dass sich $f(\mathbf{A}\mathbf{x})$ zu $|\det \mathbf{A}|^{-1}F((\mathbf{A}^{-1})^T\boldsymbol{\omega})$ transformiert (siehe [15]), und erhalten schließlich für das gesuchte Spektrum von $\chi(\mathbf{x})$

$$\begin{aligned}\hat{\chi}(\boldsymbol{\omega}) &= F((\mathbf{A}^{-1})^T\boldsymbol{\omega}) = \left(\frac{1}{2}\delta(\mathbf{N} \cdot \boldsymbol{\omega}) + \frac{1}{j2\pi\mathbf{N} \cdot \boldsymbol{\omega}}\right)\delta(\mathbf{N}^\perp \cdot \boldsymbol{\omega}) \\ &= \frac{1}{2}\delta(\boldsymbol{\omega}) + \frac{1}{j2\pi\mathbf{N} \cdot \boldsymbol{\omega}}\delta(\mathbf{N}^\perp \cdot \boldsymbol{\omega}).\end{aligned}\tag{E.9}$$

Anhang F

Symbol-Tabelle

Ω	Bildbereich, Teilmenge des \mathbb{R}^2
$\mathbf{x} = (x, y)^T$	Ortsvektor, $\mathbf{x} \in \Omega$
$\mathbf{v} = (v_x, v_y)^T$	Geschwindigkeitsvektor, $\mathbf{v} \in \mathbb{R}^2$ In einigen Fällen ist \mathbf{v} in homogenen Koordinaten dargestellt, d.h. $\mathbf{v} = (v_x, v_y, 1)^T$
$\mathbf{u} = (u_x, u_y)^T$	Geschwindigkeitsvektor, $\mathbf{u} \in \mathbb{R}^2$ In einigen Fällen ist \mathbf{u} in homogenen Koordinaten dargestellt, d.h. $\mathbf{v} = (u_x, u_y, 1)^T$
$\boldsymbol{\omega} = (\omega_x, \omega_y)^T$	Ortsfrequenzvektor, $\boldsymbol{\omega} \in \mathbb{R}^2$
ω_t	Zeitfrequenzvariable, $\omega_t \in \mathbb{R}$
\mathbf{J}_N	Strukturtenor für N Bewegungen Gl. (3.15)
K	Konfidenzmaß, Strukturtenor, Gl. (3.37)
S	Konfidenzmaß, Strukturtenor, Gl. (3.37)
H	Konfidenzmaß, Strukturtenor, Gl. (3.37)
$I = \{I_1, \dots, I_m\}$	Menge von m geordneten Sequenzen I_m , der Länge N auf dem Alphabet $\{x, y, t\}$. Gl. (2.23) Konvention: $I_m = t \dots t$
$\mathbf{c} = (\mathbf{c}_I)_I$	Gemischter Geschwindigkeitsparametervektor, Gl. (2.26),(2.23)
$f(\mathbf{x}, t)$	Bildsequenz: $f : \mathbb{R}^2 \times \mathbb{R} \rightarrow \mathbb{R}$
$g_i(\mathbf{x})$	Transparente Schicht, die sich mit der Geschwindigkeit \mathbf{v}_i bewegt. $g : \mathbb{R}^2 \rightarrow \mathbb{R}$
$F(\boldsymbol{\omega}, \omega_t)$	Fouriertransformierte der Sequenz $f(\mathbf{x}, t)$
$G_i(\boldsymbol{\omega})$	Fouriertransformierte Schicht $g_i(\mathbf{x})$
$\phi_i(\boldsymbol{\omega}) = e^{j2\pi\boldsymbol{\omega} \cdot \mathbf{v}_i}$	Phasenverschiebung durch \mathbf{v}_i .
$\nabla f = (\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial t})^T$	Gradient von $f(\mathbf{x}, t)$
$\alpha(\mathbf{v}) = \mathbf{v} \cdot \nabla$	Richtungsableitungsoperator in Richtung \mathbf{v} . Gl. (2.2)
$\mathbf{d}f$	Vektor höherer Ableitungen der Bildsequenz, Gl. (2.22), (5.12),(5.28)
$\delta(\mathbf{x})$	Diracdistribution
$\Delta g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$	Laplace-Operator von g .
BM_N	Blockmatcher für N Bewegungen, Gl. (8.16)
\mathbf{B}	Block bzw. Region um einen Punkt, Gl. (8.16)
$ \mathbf{B} $	Anzahl Bildpunkte im Block

Literaturverzeichnis

- [1] T. Aach. *Bayes-Methoden zur Bildsegmentierung, Änderungsdetektion und Verschiebungsvektorschätzung*. Fortschrittberichte VDI Reihe 10, Nr. 261, VDI Verlag, Düsseldorf, 1993. Dissertation, RWTH Aachen.
- [2] T. Aach and A. Kaup. Bayesian algorithms for adaptive change detection in image sequences using Markov random fields. *Signal Processing: Image Communication*, 7(2):147–160, 1995.
- [3] T. Aach and D. Kunz. Anisotropic spectral magnitude estimation filters for noise reduction and image enhancement. In *Proc. IEEE Internat. Conf. on Image Proc. ICIP96*, pages 335–338, Lausanne, Switzerland, 1996.
- [4] T. Aach and D. Kunz. Bayesian motion estimation for temporally recursive noise reduction in X-Ray fluoroscopy. *Philips Journal of Research*, 51(2):231–251, 1998.
- [5] T. Aach and D. Kunz. A lapped directional transform for spectral image analysis and its application to restoration and enhancement. *Signal Processing*, 80(11):2347–2364, 2000.
- [6] T. Aach, I. Stuke, C. Mota, and E. Barth. Estimation of multiple local orientations in image signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume III, pages 553–556, Montreal, Canada, May 17-21 2004.
- [7] E. Adelson. Lightness perception and lightness illusions. In M. Gazzaniga, editor, *In The New Cognitive Neurosciences, 2nd ed., Chapter 24*, pages 339–351. MA : MIT Press, Cambridge, 2000.
- [8] E. Adelson and J. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of Optical Society of America*, 2:284–321, 1985.
- [9] T. Adelson and P. Anandan. Ordinal characteristics of transparency. In *Proc. AAAI-90 Workshop on Qualitative Vision*, pages 77–81, Boston, MA, 1990.
- [10] Y. Altunbasak, R. Mersereau, and A. Patti. A fast parametric motion estimation algorithm with illumination and lens distortion correction. *IEEE Transactions on Image Processing*, 12(4):395–408, 2003.
- [11] L. Alvarez, J. Esclarn, M. Lefbure, and J. Sanchez. A PDE model for computing the optical flow. In *Proceedings of the XVI Congreso de Ecuaciones Diferenciales y Aplicaciones*, pages 1349–1356, Las Palmas de Gran Canaria, Spain, 1999.

- [12] P. Anandan. A computational framework and algorithm for measurement of visual motion. *International Journal on Computer Vision*, 2:283–310, 1989.
- [13] S. Ayer and H. S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and mdl encoding. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, pages 777–784, Washington, DC, USA, 1995. IEEE Computer Society.
- [14] A. Bainbridge-Smith and R. G. Lane. Determining optical flow using a differential method. *Image Vision Comput.*, 15(1):11–22, 1997.
- [15] R. Bamler. *Mehrdimensionale Lineare Systeme*. Springer, Berlin, 1989.
- [16] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *Int. J. Comput. Vision*, 12(1):43–77, 1994.
- [17] E. Barth. The minors of the structure tensor. In G. Sommer, N. Krüger, and C. Perwass, editors, *DAGM-Symposium*, Informatik Aktuell, pages 221–228. Springer, 2000.
- [18] E. Barth, I. Stuke, T. Aach, and C. Mota. Spatio-temporal motion estimation for transparency and occlusion. In *Proceedings of the IEEE International Conference on Image Processing*, volume III, pages 69–72, Barcelona, Spain, September 2003.
- [19] E. Barth, I. Stuke, and C. Mota. Analysis of motion and curvature in image sequences. In *Proceedings of the 5th IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 206–210, Santa Fe, New Mexico, 2002.
- [20] S. S. Beauchemin and J. L. Barron. The frequency structure of 1D occluding image signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(2):200–206, 2000.
- [21] S. S. Beauchemin, K. Daniilidis, and R. Bajcsy. Computing multiple image motions. In *Vision Interface 99*, pages 544–51, Trois-Rivieres, Canada, May 18-21 1999.
- [22] J. R. Bergen, P. J. Burt, R. Hingorani, and S. Peleg. A three-frame algorithm for estimating two-component image motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(9):886–895, 1992.
- [23] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Stat. Soc.*, 48(3):259–302, 1986.
- [24] J. E. Besag. Spatial interaction and the statistical analysis of lattice systems. *J. Roy. Stat. Soc.*, B 36:192–236, 1974.
- [25] J. Bigün. Frequency and orientation sensitive texture measures using linear symmetry. *Signal Process.*, 29(1):1–16, 1992.
- [26] J. Bigün and G. H. Granlund. Optimal orientation detection of linear symmetry. In *Proceedings of the IEEE First International Conference on Computer Vision*, pages 433–438, Washington DC, June 1987.

- [27] J. Bigün and G. H. Granlund. Optical flow based on the inertia matrix of the frequency domain. In *Proceedings from SSAB Symposium on Picture Processing*, pages 132–135, Lund University, Sweden, March 1988. SSAB. Report LiTH-ISY-I-0934, Computer Vision Laboratory, Linköping University, Sweden, 1988.
- [28] J. Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, August 1991. Report LiTH-ISY-I-1148, Linköping University, Sweden, 1990.
- [29] M. Black. Recursive non-linear estimation of discontinuous flow fields. In *Proc. of the Third European Conf. on Computer Vision, ECCV'94*, pages 138–145, Stockholm, Sweden, 1994.
- [30] M. J. Black. *Robust Incremental Optical Flow*. PhD thesis, Yale University, Department of Computer Science, Research Report YALEU-DCS-RR-923, 1992.
- [31] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. Computer Vision and Pattern Recognition, CVPR-91*, pages 296–302, Maui, Hawaii, 1991.
- [32] M. J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Comput. Vis. Image Underst.*, 63(1):75–104, 1996.
- [33] M. J. Black and D. J. Fleet. Probabilistic detection and tracking of motion boundaries. *International Journal of Computer Vision*, 38(3):231–245, 2000.
- [34] M. J. Black, D. J. Fleet, and Y. Yacoob. A framework for modeling appearance change in image sequences. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, pages 660–667, Washington, DC, USA, 1998. IEEE Computer Society.
- [35] M. J. Black, D. J. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. *Comput. Vis. Image Underst.*, 78(1):8–31, 2000.
- [36] M. M. Bronstein, A. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi. Separation of reflections via sparse ica. In *Proceedings of the IEEE International Conference on Image Processing (ICIP 2003)*, volume 1, pages 313–316, Barcelona, Catalonia, Spain, September 14-18, 2003.
- [37] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In T. Pajdla and J. Matas, editors, *ECCV (4)*, volume 3024 of *Lecture Notes in Computer Science*, pages 25–36. Springer, 2004.
- [38] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr. Real-time optic flow computation with variational methods. In N. Petkov and M. A. Westenberg, editors, *CAIP*, volume 2756 of *Lecture Notes in Computer Science*, pages 222–229. Springer, 2003.

- [39] A. Bruhn, J. Weickert, and C. Schnörr. Combining the advantages of local and global optic flow methods. In L. J. V. Gool, editor, *DAGM-Symposium*, volume 2449 of *Lecture Notes in Computer Science*, pages 454–462. Springer, 2002.
- [40] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *Int. J. Computer Vision*, 61(3):211–231, 2005.
- [41] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.
- [42] CCITT. Recommendation H.261 : Video codec for audiovisual services at p x 64 kbit/s. *COM XV-R 37-E*, 1989.
- [43] C. Cheung and L. Po. Normalized partial distortion search algorithm for block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(2):417–422, 2000.
- [44] T.-K. Chiew, J. T. H. Chung-How, D. R. Bull, and C. N. Canagarajah. Interpolation-free subpixel refinement for block-based motion estimation. In *Visual Communications and Image Processing 2004*, S. Panchanathan and B. Vasudev, eds., *Proceedings of SPIE*, volume 5308, pages 1261–1269, 2004.
- [45] K. H. K. Chow and M. L. Liou. Generic motion search algorithm for video compression. *IEEE Trans. Circuits Syst. Video Technol.*, 3:440–445, 1993.
- [46] N. H. Cornelius and T. Kanade. Adapting optical-flow to measure object motion in reflectance and X-Ray image sequences. In *Proc. of ACM SIGGRAPH/SIGART Workshop on Motion: Representation and Perception*, pages 50–58, Toronto, 1983.
- [47] D. Cremers and C. Schnörr. Motion competition: Variational integration of motion segmentation and shape regularization. In *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, pages 472–480, London, UK, 2002. Springer-Verlag.
- [48] T. Darrell and E. Simoncelli. Nulling filters and the separation of transparent motions. In *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*, pages 738–739, New York, June 14–17 1993.
- [49] T. Darrell and E. Simoncelli. Separation of transparent motion into layers using velocity-tuned mechanisms. In *Proceedings of the European Conference on Computer Vision*, Stockholm, Sweden, 1994. Springer-Verlag.
- [50] C. Q. Davis and D. M. Freeman. Equivalence of subpixel motion estimators based on optical flow and block matching. In *ISCV '95: Proceedings of the International Symposium on Computer Vision*, pages 7–12, Washington, DC, USA, 1995. IEEE Computer Society.
- [51] A. Del Bimbo, P. Nesi, and J. Sanz. Analysis of optical flow constraints. *IEEE Transactions on Image Processing*, 4(4):460–469, 1995.

- [52] R. Deriche, P. Kornprobst, and G. Aubert. Optical-flow estimation while preserving its discontinuities: A variational approach. In *Proc. of the Second Asian Conference on Computer Vision*, pages 71–80, London, UK, 1996. Springer-Verlag.
- [53] M. Elad and A. Feuer. Recursive optical flow estimation - adaptive filtering approach. *The Journal of Visual Comm. and Image Representation*, 9:119–138, 1998.
- [54] H. Farid and E. H. Adelson. Separating reflections and lighting using independent components analysis. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR '99)*, pages 1262–1267, Ft. Collins, CO, USA, 23-25 June 1999, 1999.
- [55] H. Farid and E. H. Adelson. Separating reflections from images by use of independent component analysis. *Journal of the Optical Society of America*, 16(9):2136–2145, 1999.
- [56] G. Farnebäck. Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. In *Proceedings of the Eighth IEEE International Conference on Computer Vision*, volume I, pages 171–177, Vancouver, Canada, July 2001.
- [57] M. Felsberg and G. Granlund. POI detection using channel clustering and the 2D energy tensor. In *Pattern Recognition: 26th DAGM Symposium*, volume 3175 of *LNCS*, pages 103–110, Tübingen, Germany, 2004. Springer Berlin.
- [58] M. Felsberg and U. Köthe. Get: The connection between monogenic scale-space and gaussian derivatives. In R. Kimmel, N. Sochen, and J. Weickert, editors, *Scale Space and PDE Methods in Computer Vision*, volume 3459 of *LNCS*, pages 192–203. Springer, 2005.
- [59] C. Fermuller, D. Shulman, and Y. Aloimonos. The statistics of optical flow. *Computer Vision and Image Understanding*, 82(1):1–32, 2001.
- [60] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Amer. A*, 4(12):2379–2394, 1987.
- [61] D. J. Fleet and A. D. Jepson. Computation of component image velocity from local phase information. *Int. J. Comput. Vision*, 5(1):77–104, 1990.
- [62] D. J. Fleet and A. D. Jepson. Stability of phase information. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(12):1253–1268, 1993.
- [63] D. J. Fleet, A. D. Jepson, and M. R. M. Jenkin. Phase-based disparity measurement. *CVGIP: Image Underst.*, 53(2):198–210, 1991.
- [64] D. J. Fleet and K. Langley. Computational analysis of non-fourier motion. *Vision Research*, 34(22):3057–3079, 1995.
- [65] W. Förstner. A feature based correspondence algorithm for image matching. In *International Archives of Photogrammetry and Remote Sensing*, volume 26-3/3, pages 150–166, Vancouver, Canada, 1986.

- [66] W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners, and centers of circular features. In *Proc. Intercommision Conf. on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, 1987.
- [67] B. Galvin, B. McCane, K. Novins, D. Mason, and S. Mills. Recovering motion fields: An evaluation of eight optical flow algorithms. In *Proceedings of the British Machine Vision Conference 1998, BMVC 1998*, Southampton, UK, 1998.
- [68] C. S. Garbe and B. Jähne. Reliable estimates of the sea surface heat flux from image sequences. In *Proceedings of the 23rd DAGM-Symposium on Pattern Recognition*, pages 194–201, London, UK, 2001. Springer-Verlag.
- [69] C. S. Garbe, H. Spies, and B. Jähne. Estimation of surface flow and net heat flux from infrared image sequences. *J. Math. Imaging Vis.*, 19(3):159–174, 2003.
- [70] T. Gautama and M. Van Hulle. A phase-based approach to the estimation of the optical flow field using spatial filtering. *IEEE Trans. Neural Networks*, 13(5):1127–1136, 2002.
- [71] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.
- [72] F. Glazer. Multilevel relaxation in low-level computer vision. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis*, pages 312–330. Springer Verlag, 1983.
- [73] G. H. Golub and C. F. van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore MD, 3 edition, 1996.
- [74] G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision*. Kluwer Academic Publishers, 1995. ISBN 0-7923-9530-1.
- [75] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(10):1025–1039, 1998.
- [76] M. Hahn and N. Krüger. Junction detection and semantic interpretation using hough lines. In *2nd Int. ICSC Symposium on Engineering of Intelligent Systems (EIS'2000)*, pages 109–115, University of Paisley, Scotland, 2000.
- [77] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey88 Vision Conf.*, pages 189–192, Manchester, 1988.
- [78] H. W. Haußecker. Physics from IR sequences: Quantitative analysis of transport models and parameters of air-sea gas transfer. In *Proc. Int. Symp. Gas Transfer at Water Surfaces*, 2000.
- [79] H. W. Haußecker and D. J. Fleet. Computing optical flow with physical models of brightness variation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6):661–673, 2001.

- [80] H. W. Haußecker and B. Jähne. A tensor approach for local structure analysis in multi-dimensional images. In *Proc. 3D Image Analysis and Synthesis '96*, pages 171–178, Universität Erlangen-Nürnberg, November 18-19 1996.
- [81] H. W. Haußecker and B. Jähne. A tensor approach for precise computation of dense displacement vector fields. In *Proceedings Mustererkennung 1997, 19. DAGM-Symposium*, pages 199–208, London, UK, 1997. Springer-Verlag.
- [82] F. Heitz and P. Bouthemy. Motion estimation and segmentation using a global bayesian approach. In *15th IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pages 2305–2308, Albuquerque (USA), 1990.
- [83] J. Hladuvka and E. Gröller. Direction-driven shape-based interpolation of volume data. In *Proceedings of the Vision Modeling and Visualization Conference 2001 (VMV-01)*, pages 113–120, Stuttgart, Germany, 2001.
- [84] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.
- [85] M. Irani and S. Peleg. Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation (JVCIP)*, 4(4):324–335, 1993.
- [86] M. Irani and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision (IJCV)*, 12(1):5–16, 1994.
- [87] T. Iso and M. Shizawa. Detecting L-, T-, and X-Junctions from low-order image derivatives. In *Proc. SPIE Conference on Visual Communications and Image Processing 93*, volume 2094, pages 1185–1197, 1993.
- [88] B. Jähne. Performance characteristics of low-level motion estimators in spatiotemporal images. In *Proceedings of DAGM-Workshop Performance Characteristics and Quality of Computer Vision Algorithms*, pages 139–152, Braunschweig, 18. September, 1997.
- [89] S. Jamkar, S. Belhe, S. Dravid, and M. S. Sutaone. A comparison of block-matching search algorithms in motion estimation. In *ICCC '02: Proceedings of the 15th international conference on Computer communication*, pages 730–739, Washington, DC, USA, 2002. International Council for Computer Communication.
- [90] A. Jepson and M. Black. Mixture models for optical flow computation. In *IEEE Conf. on Computer Vision and Pattern Recognition CVPR-93*, pages 760–761, New York, NY, 1993.
- [91] S. Ju, M. J. Black, and A. D. Jepson. Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, CVPR'96*, pages 307–314, San Francisco, CA, 1996.

- [92] M. Kass and A. Witkin. Analyzing oriented patterns. *Computer Vision, Graphics, and Image Processing*, 37(3):362–385, 1987.
- [93] J. K. Kearney, W. B. Thompson, and D. L. Boley. Optical flow estimation: an error analysis of gradient-based methods with local optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(2):229–244, 1987.
- [94] L. Kitchen and A. Rosenfeld. Gray level corner detection. *Patt. Recog. Lett.*, 1(2):95–102, December 1982.
- [95] H. Knutsson and G. H. Granlund. Texture Analysis Using Two-Dimensional Quadrature Filters. In *IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management - CAPAIDM*, Pasadena, October 1983.
- [96] J. Konrad and C. Stiller. On Gibbs-Markov models for motion computation. In H. Li, S. Sun, and H. Derin, editors, *Video Compression for Multimedia Computing - Statistically Based and Biologically Inspired Techniques*, pages 121–154. Kluwer Academic Publishers, 1997.
- [97] U. Köthe. Edge and junction detection with an improved structure tensor. In G. K. B. Michaelis, editor, *Pattern Recognition, Proc. of 25th DAGM Symposium*, volume 2781 of *Lecture Notes in Computer Science*, pages 25–32, Heidelberg, 2003. Springer. Awarded the main prize of the German Pattern Recognition Society (DAGM) 2003.
- [98] U. Köthe. Integrated edge and junction detection with the boundary tensor. In *ICCV 03, Proc. of 9th Intl. Conf. on Computer Vision*, volume 1, pages 424–431. IEEE Computer Society, 2003.
- [99] A. Kumar, A. R. Tannenbaum, and G. J. Balas. Optical flow: a curve evolution approach. *IEEE Transactions on Image Processing*, 5(4):598–610, 1996.
- [100] R. Li, B. Zeng, and M. Liou. A new three-step search algorithm for fast block motion estimation. *IEEE Transactions on Circuits and System for Video Technology*, 4:438–442, 1994.
- [101] W. Li and E. Salari. Successive elimination algorithm for motion estimation. *IEEE Transactions on Image Processing*, 4(1):105–107, 1995.
- [102] R. Lillestrand. Techniques for change detection. *IEEE Transactions on Computers*, 21(7):654–659, 1972.
- [103] Y. Liu and S. Orintara. Complexity comparison of fast block-matching. motion estimation algorithms. In *Proc. of IEEE International on Acoustics, Speech, and Signal Processing*, pages 341–344, Quebec, Canada, 2004.
- [104] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, Vancouver, Canada, 1981.

- [105] B. D. Lucas. *Generalized image matching by the method of differences*. PhD thesis, 1984.
- [106] E. Memin and P. Perez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Transactions on Image Processing*, 7(5):703–719, 1998.
- [107] M. Michaelis and G. Sommer. Junction classification by multiple orientation detection. In J.-O. Eklundh, editor, *Proc. Third European Conference on Computer Vision*, pages 101–108. Stockholm, LNCS, Vol. 800, Springer-Verlag, 1994.
- [108] P. Milanfar. Projection-based, frequency-domain estimation of superimposed translational motions. *Journal of Optical Society of America*, 13(11):2151–2162, 1996.
- [109] P. Milanfar. Two-dimensional matched filtering for motion estimation. *IEEE Transaction on Image Processing*, 8(3):438–443, 1999.
- [110] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: a synopsis of current problems and methods. *Int. J. Comput. Vision*, 19(1):29–55, 1996.
- [111] A. Mitiche and A. Mansouri. On convergence of the Horn and Schunck optical-flow estimation method. *IEEE Transactions on Image Processing*, 13(6):848–852, June 2004.
- [112] C. Mota, M. Dorr, I. Stuke, and E. Barth. Categorization of transparent-motion patterns using the projective plane. In W. Dosch and R. Y. Lee, editors, *Proceedings of the ACIS 4th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 03)*, pages 363–369, Luebeck, Germany, October 2003.
- [113] C. Mota, M. Dorr, I. Stuke, and E. Barth. Analysis and synthesis of motion patterns using the projective plane. In B. E. Rogowitz and T. N. Pappas, editors, *Human Vision and Electronic Imaging IX*, volume 5292, pages 174–181, San Jose, California, USA, 2004. SPIE.
- [114] C. Mota, M. Dorr, I. Stuke, and E. Barth. Categorization of transparent-motion patterns using the projective plane. *International Journal of Computer & Information Science*, 5(2):129 – 140, 2004.
- [115] C. Mota, I. Stuke, T. Aach, and E. Barth. Estimation of multiple orientations at corners and junctions. In *DAGM04: 26th Pattern Recognition Symposium*, pages 163–170, Tuebingen, Aug. 30 – Sept. 1 2004. German Association for Pattern Recognition.
- [116] C. Mota, I. Stuke, T. Aach, and E. Barth. Divide-and-conquer strategies for estimating multiple transparent motions. In *Proceedings of the 1st International Workshop on Complex Motion*, Schloss Reinsburg, Germany, 2005. Lecture Notes on Computer Science, LNCS 3417.

- [117] C. Mota, I. Stuke, T. Aach, and E. Barth. Spatio-temporal motion estimation for transparency and occlusions. *Signal Processing: Image Communication. Elsevier Science*, 20(6):529–536, 2005.
- [118] C. Mota, I. Stuke, and E. Barth. Analytic solutions for multiple motions. In *Proceedings of the International Conference on Image Processing*, pages 917–920, Thessaloniki, Greece, October 7–10 2001.
- [119] N. Mukawa. Estimation of shape, reflection coefficients and illumination direction from image sequences. In *Proceedings of the Third International Conference on Computer Vision*, pages 507–512, Osaka, Japan, 1990.
- [120] H.-H. Nagel. On a constraint equation for the estimation of displacement rates in image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(1):13–30, 1989.
- [121] H.-H. Nagel. Extending the 'oriented smoothness constraint' into the temporal domain and the estimation of derivatives of optical flow. In O. D. Faugeras, editor, *ECCV*, volume 427 of *Lecture Notes in Computer Science*, pages 139–148. Springer, 1990.
- [122] H. H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(5):565–593, 1986.
- [123] S. Negahdaripour. Revised definition of optical flow: Integration of radiometric and geometric cues for dynamic scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(9):961–979, 1998.
- [124] P. Nillius and J.-O. Eklundh. Fast block matching with normalized cross-correlation. Technical Report ISRN KTH/NA/P-02/11-SE, KTH, 2002.
- [125] J.-M. Odobez and P. Bouthemy. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 6(2):143–155, 1998.
- [126] N. Ohnishi, K. Kumaki, T. Yamamura, and T. Tanaka. Separating real and virtual objects from their overlapping images. In *Proceedings of the 4th European Conference on Computer Vision*, volume 2, pages 636–646, Cambridge, UK, April 15-18 1996.
- [127] N. Ohta. Uncertainty models of the gradient constraint for optical flow computation. *IEICE Transactions on Information and Systems (D)*, E79-D(7):958–964, 1996.
- [128] A. V. Oppenheim, R. W. Schafer, and T. G. Stockham. Nonlinear filtering of multiplied and convolved signals. *Proceedings of the IEEE*, 56(8):1264–1291, 1968.
- [129] L. Parida, D. Geiger, and R. A. Hummel. Kona: A multi-junction detector using minimum description length principle. In M. Pelillo and E. R. Hancock, editors, *Energy Minimization Methods in Computer Vision and Pattern Recognition, First International Workshop, EMMCVPR'97, Venice, Italy, May 21-23, 1997, Proceedings*, volume 1223 of *Lecture Notes in Computer Science*, pages 51–65. Springer, 1997.

- [130] B. T. Phong. Illumination for computer generated pictures. *Commun. ACM*, 18(6):311–317, 1975.
- [131] E. Pingault, M Bruno and D. Pellerin. A robust multiscale B-spline function decomposition for estimating motion transparency. *IEEE Transactions on Image Processing*, 12(11):1416– 1426, November 2003.
- [132] K. Rohr. Recognizing corners by fitting parametric models. *Int. J. Comput. Vision*, 9(3):213–230, 1992.
- [133] B. Sarel and M. Irani. Separating transparent layers through layer information exchange. In *Proceedings on 8th European Conference on Computer Vision*, pages 328–341, Prague, Czech Republic, May 11-14, 2004.
- [134] H. Scharr. Optimal filters for extended optical flow. In *International Workshop on Complex Motion, LNCS 3417*, 2005.
- [135] H. Scharr, I. Stuke, C. Mota, and E. Barth. Estimation of transparent motions with physical models for additional brightness variation. In *13th European Signal processing Conference*, Antalya, Turkey, Sep. 4-8 2005.
- [136] Y. Schechner, N. Kiryati, and J. Shamir. Separation of transparent layers by polarization analysis. In *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*, volume 1, pages 235–242, Kangerlussuaq, Greenland, 1999.
- [137] Y. Y. Schechner, N. Kiryati, and R. Basri. Separation of transparent layers using focus. In *Proceedings of the International Conference On Computer Vision (ICCV)*, pages 1061–1066, Bombay, 1998.
- [138] Y. Y. Schechner, N. Kiryati, and R. Basri. Separation of transparent layers using focus. *International Journal of Computer Vision*, 39(1):25–39, 2000.
- [139] Y. Y. Schechner, N. Kiryati, and J. Shamir. Multi-valued images and their separation. In *Proceedings of the 10th International Workshop on Theoretical Foundations of Computer Vision*, pages 129–141, London, UK, 2001. Springer-Verlag.
- [140] Y. Y. Schechner, J. Shamir, and N. Kiryati. Polarization-based decorrelation of transparent layers: The inclination angle of an invisible surface. In *Proceedings off International Conference on Computer Vision (ICCV)*, pages 814–819, Kerkyra, Greece, 1999.
- [141] Y. Y. Schechner, J. Shamir, and N. Kiryati. Blind recovery of transparent and semireflected scenes. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1038–1043, Hilton Head, SC, USA, 13-15 June 2000.
- [142] C. Schnörr. Computation of discontinuous optical flow by domain decomposition and shape optimization. *Int. J. Comput. Vision*, 8(2):153–165, 1992.

- [143] C. Schnörr. Segmentation of visual motion by minimizing convex non-quadratic functionals. In *Proc.12th Int. Conf. on Pattern Rec.*, pages 661–663, Jerusalem, Israel, October 1994.
- [144] Y. Senda, H. Harasaki, and M. Yano. Theoretical background and improvement of a simplified half-pel motion estimation. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, volume 6, pages 263–266, Atlanta, Georgia, 1996.
- [145] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, 2nd edition, 1999.
- [146] M. Shizawa and T. Iso. Direct representation and detecting of multi-scale, multi-orientation fields using local differentiation filters. In *Proceedings IEEE Conference on of Computer Vision and Pattern Recognition (CVPR)*, pages 508–514, New York, NY, USA, 1993.
- [147] M. Shizawa and K. Mase. Simultaneous multiple optical flow estimation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 274–8, Atlantic City, June 1990.
- [148] M. Shizawa and K. Mase. A unified computational theory for motion transparency and motion boundaries based on eigenenergy analysis. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 289–95, Maui, Hawaii, June 1991.
- [149] T. Sikora. The MPEG-4 video standard verification model. *IEEE Trans. on Circuits and Systems for VideoTechnology*, 7(1):19–31, 1997.
- [150] T. Sikora. MPEG digital video-coding standards. *IEEE Signal Processing Magazine*, 14(5):82–100, 1997.
- [151] E. P. Simoncelli. Distributed representation and analysis of visual motion. Technical Report 209, MIT Media Laboratory, Cambridge, MA, 1993.
- [152] E. P. Simoncelli. A rotation-invariant pattern signature. In *Third Int. Conf. on Image Proc.*, volume III, pages 185–188, Lausanne, September 1996. IEEE Sig Proc Society.
- [153] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger. Probability distributions of optical flow. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 310–315, Maui, Hawaii, 1991. IEEE Computer Society.
- [154] K. Skifstad and R. Jain. Illumination independent change detection for real world image sequences. *Comput. Vision Graph. Image Process.*, 46(3):387–399, 1989.
- [155] M. V. Srinivasan. Generalized gradient schemes for the measurement of two-dimensional image motion. *Biol Cybernetics*, 63:421–31, 1990.
- [156] C. Stiller. Object-based estimation of dense motion fields. *IEEE Transactions on Image Processing*, 6(2):1111–1117, 1997.

- [157] C. Stiller and J. Konrad. Estimating motion in image sequences. *IEEE Signal Processing Magazine*, 16(4):70–91, 1999.
- [158] I. Stuke. Bestimmung überlagerter Bewegungen in Bildsequenzen. Master's thesis, Institut für Signalverarbeitung, Universität zu Lübeck, 2001.
- [159] I. Stuke, T. Aach, E. Barth, and C. Mota. Estimation of multiple motions by block matching. In W. Dosch and R. Y. Lee, editors, *Proceedings of the ACIS Fourth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD'03)*, pages 358–362, Luebeck, Germany, October 2003.
- [160] I. Stuke, T. Aach, E. Barth, and C. Mota. Analysing superimposed oriented patterns. In *6th IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 133 – 137, Lake Tahoe, NV, 2004.
- [161] I. Stuke, T. Aach, E. Barth, and C. Mota. Estimation of multiple motions using block-matching and Markov random fields. In S. Panchanathan and B. Vasudev, editors, *Visual Communications and Image Processing 2004*, volume 5308, pages 486–496, San Jose, California, USA, 2004. SPIE.
- [162] I. Stuke, T. Aach, E. Barth, and C. Mota. Multiple-motion estimation by block-matching using Markov random fields. *International Journal of Computer & Information Science*, 5(2):141 – 152, 2004.
- [163] I. Stuke, T. Aach, C. Mota, and E. Barth. Estimation of multiple motions: regularization and performance evaluation. In B. Vasudev, R. Hsing, A. Tescher, and T. Ebrahimi, editors, *SPIE/IST Electronic Imaging 2003*, volume 5022, San Jose, California, USA, 2003. SPIE.
- [164] R. Szeliski, S. Avidan, and P. Anandan. Layer extraction from multiple images containing reflections and transparency. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR 2000)*, volume 1, pages 246–253, Hilton Head, SC, USA, 13–15 June 2000.
- [165] D. Terzopoulos. Image analysis using multigrid relaxation methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(2):129–139, 1986.
- [166] J. Y. Tham, S. Ranganath, M. Ranganth, and A. A. Kassim. A novel unrestricted center-biased diamond search algorithm for block motion estimation. *IEEE Trans. Circuits and Systems for Video Technology*, 8(4):369–377, 1998.
- [167] T. Toivonen and J. Heikkilä. Efficient method for half-pixel block motion estimation using block differentials. In N. N. García, J. M. Martínez, and L. Salgado, editors, *VLBV*, volume 2849 of *Lecture Notes in Computer Science*, pages 225–232. Springer, 2003.
- [168] J. Toro, R. Medina, and D. Ziou. Recovery of two transparent primitive images from two frames. In *Proceedings of the IEEE International Conference on Acoustics*,

- Speech, and Signal Processing (ICASSP '04)*, volume 3, pages 225–228, Quebec, Canada, 2004.
- [169] J. Toro, F. Owens, and R. Medina. Multiple motion estimation and segmentation in transparency. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 2087–2090, Istanbul, Turkey, 2000.
- [170] P. H. S. Torr, R. Szeliski, and P. Anandan. An integrated bayesian approach to layer extraction from image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3):297–303, 2001.
- [171] D. Toth, T. Aach, and V. Metzler. Bayesian spatio-temporal motion detection under varying illumination. In *European Signal Processing Conference (EUSIPCO)*, pages 2081–2084, Tampere, Finland, 2000.
- [172] D. Toth, T. Aach, and V. Metzler. Illumination-invariant change detection. In *SSIAI*, pages 3–7. IEEE Computer Society, 2000.
- [173] UTU-T. Recommendation H.263 : Video codec for narrow telecommunicatin channels at < 64 kbit/s, 1989.
- [174] D. Vernon. Decoupling Fourier components of dynamic image sequences: A theory of signal separation, image segmentation, and optical flow estimation. In *Proceedings of the European Conference on Computer Vision ECCV '98*, pages 69–85, Freiburg, Germany, 1998.
- [175] A. Verri and T. Poggio. Motion field and optical flow: Qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-11(5):490–498, 1989.
- [176] R. Vidal and Y. Ma. A unified algebraic approach to 2-d and 3-d motion segmentation. In T. Pajdla and J. Matas, editors, *ECCV (1)*, volume 3021 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2004.
- [177] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *The IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638, September 1994.
- [178] A. B. Watson and A. J. Ahumada. Look at motion in the frequency domain. In *Proc. of ACM SIGGRAPH/SIGART Workshop on Motion: Representation and Perception*, pages 1–10, Toronto, 1983.
- [179] A. B. Watson and A. J. Ahumada. Model of human visual-motion sensing. *Journal of Optical Society of America*, 2(2):322–342, February 1985.
- [180] J. Weickert, A. Bruhn, N. Papenberg, and T. Brox. Variational optic flow computation: From continuous models to algorithms. In L. Alvarez, editor, *International Workshop on Computer Vision and Image Analysis, IWCVIA'03*, Las Palmas de Gran Canaria, Dec. 2003.

- [181] J. Weickert and C. Schnörr. A theoretical framework for convex regularizers in PDE-based computation of image motion. *International Journal of Computer Vision*, 45(3):245–264, 2001.
- [182] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *J. Math. Imaging Vis.*, 14(3):245–255, 2001.
- [183] Y. Weiss. Smoothness in layers: Motion segmentation using nonparametric mixture estimation. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 520–527, Washington, DC, USA, 1997. IEEE Computer Society.
- [184] Y. Weiss. Deriving intrinsic images from image sequences. In *Proceedings of the Eighth International Conference On Computer Vision (ICCV-01)*, volume 2, pages 68–75, Vancouver, British Columbia, Canada, July 7-14., 2001.
- [185] Y. Weiss and E. Adelson. A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 321–326, Washington, DC, USA, 1996. IEEE Computer Society.
- [186] R. P. Wildes, M. J. Amabile, A.-M. Lanzillotto, and T.-S. Leu. Recovering estimates of fluid flow from image sequence data. *Comput. Vis. Image Underst.*, 80(2):246–266, 2000.
- [187] W. Yu, K. Daniilidis, S. Beauchemin, and G. Sommer. Detection and characterization of multiple motion points. In *IEEE Conf. Computer Vision and Pattern Recognition*, volume I, pages 171–177, June 1999. Fort Collins, CO.
- [188] W. Yu, K. Daniilidis, and G. Sommer. Approximate orientation steerability based on angular Gaussians. *IEEE Transactions on Image Processing*, 10(2):193–205, 2001.
- [189] W. Yu, K. Daniilidis, and G. Sommer. Multiple motion analysis: in spatial or in spectral domain? *Computer Vision and Image Understanding*, 90(2):129–152, 2003.
- [190] W. Yu, G. Sommer, S. Beauchemin, and K. Daniilidis. Oriented structure of the occlusion distortion: is it reliable? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1286–1290, September 2002.
- [191] S. D. Zeno. A note on the gradient of a multi-image. *Comput. Vision Graph. Image Process.*, 33(1):116–125, 1986.
- [192] C. Zetsche and E. Barth. Fundamental limits of linear filters in the visual processing of two-dimensional signals. *Vision Research*, 30:1111–1117, 1990.
- [193] L. Zhang, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo. In *Proceedings of IEEE the 9th International Conference on Computer Vision*, pages 618–625, Washington, DC, USA, 2003. IEEE Computer Society.

- [194] W. Zhou and C. Kambhamettu. Separation of reflection component by Fourier decoupling. In *Proceedings of the Asian Conference on Computer Vision*, pages 27–30, Jeju Island, Korea, 2004.

Lebenslauf

Persönliche Daten

Name	Ingo Stuke
Geburtsdatum	18. November 1974
Geburtsort	Bremen
Wohnort	Lübeck
Familienstand	verheiratet, ein Kind

Bildungsweg

1981 – 1994	Grundschule und Gymnasium Horn, Bremen Abschluss: Abitur
Oktober 1995 – März 2001	Studium der Informatik an der Universität zu Lübeck
22 März 2001	Studienabschluß zum Diplom-Informatiker an der Universität zu Lübeck

Wehrersatzdienst

Juli 1994 – September 1995	Wehrersatzdienst beim Deutschen Roten Kreuz Bremen
----------------------------	--

Beruflicher Werdegang

seit April 2001	Wissenschaftlicher Mitarbeiter am Institut für Signalverarbeitung und Prozessrechentechnik der Universität zu Lübeck.
-----------------	---