

# Evaluierung effizienter Methoden zur Berechnung des optischen Flusses

DANIEL FREY & MARKUS ULRICH, München; STEFAN HINZ, Karlsruhe

**Keywords:** Optischer Fluss, Wavelets, Mehrgitterverfahren, Evaluierung

**Summary:** *Evaluation of Efficient Methods for Optical Flow Computation.* The efficient computation of optical flow is a prerequisite for a large number of applications, which led to the development of various different optical flow algorithms in the past years. Optical flow describes for each pixel in the first image its position in the second image. Hence, it provides the basis to analyze ego-motion or motion of objects between two images in a pixel-wise fashion. Despite of the fact that many efficient solutions to calculate the optical flow have been developed in the past, there is a lack of research simultaneously comparing the accuracy and the run-time efficiency of these algorithms systematically.

After the mathematical definition of the optical flow, the basic principles of four selected methods of optical flow computation are explained. The main contribution of this article is the evaluation of these algorithms. The studied algorithms are on the one hand so called multigrid methods and on the other hand approaches that are based on wavelet decomposition. All algorithms have in common that they prospect high performance. The motivation of the evaluation is to help the reader to find the most suitable algorithm for his task.

**Zusammenfassung:** Die effiziente Berechnung des optischen Flusses zwischen zwei Bildern ist Voraussetzung für eine Vielzahl von Anwendungen und daher in der Bildverarbeitung weiterhin aktuelles Forschungsthema. Der optische Fluss beschreibt für jeden Bildpunkt in einem Bild einer Bildfolge dessen Position im darauffolgenden Bild der Bildfolge. Damit lassen sich z. B. Bewegungsabläufe innerhalb einer Bildsequenz analysieren. In den letzten Jahren wurden verschiedene effiziente Lösungsmöglichkeiten entwickelt. Jedoch mangelt es an Arbeiten, die die Laufzeiteffizienzen dieser Algorithmen deren Genauigkeiten systematisch gegenüberstellen.

Nach einer mathematischen Definition des optischen Flusses werden in diesem Artikel die Grundprinzipien von vier ausgewählten Algorithmen erläutert. Der Hauptbeitrag des Artikels ist die Evaluierung dieser Lösungsansätze. Bei den untersuchten Algorithmen handelt es sich um so genannte Mehrgitterverfahren sowie um Verfahren, die auf Wavelets beruhen. Allen gemeinsam ist, dass sie eine hohe Laufzeiteffizienz versprechen. Ziel der Evaluierung ist dabei auch, dem Leser eine Hilfestellung bei der Algorithmenwahl für seine spezielle Anwendung zu geben.

## 1 Einleitung

Die automatische Bestimmung von Objekt- oder Eigenbewegungen aus Bildsequenzen ist schon seit den frühen 1980er Jahren ein intensives Forschungsgebiet der Bildanalyse (HORN & SCHUNCK 1981, DRESCHLER & NAGEL 1982). Typische Anwendungsbeispiele sind die Verkehrsüberwachung, Personenverfolgung, autonome Navigation, sowie medizinische oder industrielle Bildverarbeitung. Die klassischen photogrammetrischen Verfahren gehen dabei typischerweise punktbasierend vor, indem mit

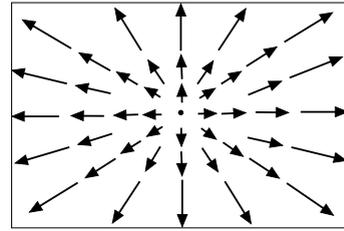
Interests-Operatoren auffällige Punktmerkmale aus den Bildern extrahiert und dann mittels Matching-Verfahren über die Bildsequenz hinweg getrackt werden (HARTLEY & ZISSERMAN 2004). Als Ergebnis erhält man Punkttrajektorien, aus denen ggf. ein lückenloses Bewegungsvektorfeld mittels Interpolationsverfahren abgeleitet wird. Während die Bewegungsvektoren in den homologen Punkten hohe Genauigkeiten aufweisen (bis in den Subpixel-Bereich), verschlechtert sich natürlich die Genauigkeit an interpolierten Positionen.

Komplementiert werden diese Verfahren durch flächenbasierte Methoden, wie z. B. den optischen Fluss. Der Vorteil solcher Verfahren liegt darin, dass die gesamte Bildinformation inklusive der lokalen Nachbarschaftsbeziehungen von Bildpunkten zur Ableitung des Bewegungsvektorfeldes genutzt wird. Als Ergebnis erhält man ein dichtes Bewegungsvektorfeld. Voraussetzung ist allerdings, dass die Bewegungsvektoren nicht größer als der Einzugsbereich der im Verfahren genutzten Filtermasken sind – eine Einschränkung, die bei o.g. Matching-Verfahren nicht gegeben ist – was meist eine hohe Bildwiederholrate bedingt. Daher wird der optische Fluss meist zur Auswertung von Videosequenzen eingesetzt, da hier die Voraussetzung kleiner Bewegungsvektoren oft erfüllt ist.

Eine wesentliche Komponente bei der Bewegungsbestimmung ist heutzutage die Echtzeitfähigkeit. Dies gilt v.a. für industrielle Anwendungen der Bildverarbeitung wie zum Beispiel bei der Qualitätskontrolle. Deshalb wurde in den letzten Jahren das Augenmerk neben der Genauigkeit auch auf die Effizienz der Algorithmen gerichtet. Die fundierte Analyse und vergleichende Evaluierung solcher Verfahren ist Ziel dieses Artikels. Nach einer mathematischen Definition des optischen Flusses werden die Grundprinzipien von vier ausgewählten Algorithmen erläutert und anhand von Testszenen vergleichend numerisch evaluiert. Bei den untersuchten Algorithmen handelt es sich um so genannte Mehrgitterverfahren sowie um Verfahren, die auf Basis von Wavelets beruhen. Allen gemeinsam ist, dass sie eine hohe Laufzeiteffizienz versprechen. Die Evaluierung stellt die Laufzeiteffizienz und die Genauigkeit des resultierenden optischen Flusses der einzelnen Verfahren gegenüber.

## 2 Optischer Fluss

Der optische Fluss wird in den meisten Fällen durch ein Vektorfeld dargestellt. In diesem Flussfeld sieht man sowohl die 2D-Bewegungsrichtung (Orientierung des Vektors) als auch die Geschwindigkeiten (Betrag des Vektors). Um diese Größen zu bestimmen, werden die Grauwerte der beiden hintereinander auf-



**Abb. 1:** Flussfeld einer Vorwärtseigenbewegung.

genommenen Bilder betrachtet. Für jedes Pixel im ersten Bild wird versucht, das korrespondierende Pixel im zweiten Bild zu finden. Der sich daraus ergebende Differenzvektor beinhaltet die Bewegung zwischen den beiden Bildpunkten im Bildraum. In Abb. 1 ist exemplarisch ein Vektorfeld des optischen Flusses für eine Vorwärtseigenbewegung abgebildet.

Im Folgenden wird das bekannte mathematische Modell für den optischen Fluss hergeleitet, das zur so genannten Flussgleichung führt. Ein Bild, das zu einer bestimmten Zeit aufgenommen wurde, kann als eine Funktion  $I(x,y,t)$  von Intensitätswerten aufgefasst werden, wobei die Bildkoordinaten  $x(t)$  und  $y(t)$  eines abgebildeten Objektpunktes wiederum von der Zeit abhängig sind. Ein Bild, das zum Zeitpunkt  $t+dt$  aufgenommen wird, kann demnach mit der Funktion  $I(x+dx,y+dy,t+dt)$  beschrieben werden. Aus der Verschiebung der Pixel ergibt sich der zweidimensionale optische Fluss  $u = dx/dt$  und  $v = dy/dt$ . Setzt man (ohne Beschränkung der Allgemeinheit) den zeitlichen Abstand zwischen zwei Aufnahmen auf  $dt = 1$ , ergibt sich  $u = dx$  und  $v = dy$ . Um zur Flussgleichung zu gelangen, wird (zunächst) die grundlegende Annahme gemacht, dass sich die Intensitätswerte von Objekten zwischen zwei Bildern nicht ändern.

$$I(x + dx, y + dy, t + dt) = I(x, y, t) \quad (1)$$

Nach einer Taylerentwicklung ersten Grades an der Stelle  $(x,y,t)$  und einigen Umformungen erhält man die von Horn und Schunk (HORN & SCHUNCK 1981) entwickelte Flussgleichung:

$$I_x u + I_y v + I_t = 0 \quad (2)$$

Hierbei bezeichnen  $I_x$ ,  $I_y$  und  $I_t$  die Ableitung der Intensitätswerte nach  $x$ ,  $y$  und  $t$ . Das Ziel

ist, die beiden Komponenten des optischen Flusses  $u$  und  $v$  so zu schätzen, dass diese Gleichung erfüllt ist. Die Annahme der konstanten Intensitätswerte in realen Szenen ist jedoch so gut wie nie erfüllt. Die Ursachen für Helligkeits- bzw. Intensitätsvariationen zwischen zwei Bildern können vielseitig sein. Zum Beispiel führen Verdeckungen von Objekten, Beleuchtungsveränderungen, Blickrichtungsänderung durch Bewegung des Aufnahmeapparates oder Helligkeitsvariation des Objektes selbst (z. B. durch Wolken) zu einer Verletzung der Annahme. Zudem ist die Flussgleichung (2) für einen einzigen Bildpunkt unterbestimmt, also ein schlecht-gestelltes Problem. Zur Lösung müssen zusätzliche Bedingungen eingeführt werden, die im folgenden Abschnitt vorgestellt werden.

### 3 Lösungsansätze für den optischen Fluss

In den vergangenen Jahren wurde eine Reihe von Lösungsmöglichkeiten für den optischen Fluss entwickelt. Eine Möglichkeit zur Gruppierung der Ansätze wurde in (BARRON et al. 1994) durchgeführt. Einer der bekanntesten und grundlegendsten Ansätze, der ein 100% dichtes Flussfeld erzeugt, stammt von HORN & SCHUNCK (1981). Sie führen als weitere Bedingung einen so genannten Glattheitsterm ein. Dieser Term beruht auf der Annahme, dass die Bewegungen benachbarter Pixel voneinander abhängig sind und das gleiche Geschwindigkeitsmuster aufweisen. Die Änderung des Vektorfeldes in der Umgebung des jeweils betrachteten Punktes soll möglichst gering sein. Diese Bedingung dient zur Regularisierung des schlecht-gestellten Problems, weshalb an dieser Stelle auch häufig von Regularisierern gesprochen wird. Mathematisch kann die von HORN und SCHUNCK aufgestellte Glattheitsbedingung wie folgt dargestellt werden:

$$E_s(u, v) = |\nabla u|^2 + |\nabla v|^2 \rightarrow \min \quad (3)$$

Neben der unterbestimmten Flussbedingung

$$E_d(u, v) = I_x u + I_y v + I_t \rightarrow \min \quad (4)$$

besitzt man mit dieser Glattheitsbedingung eine zweite Gleichung, die erfüllt werden soll. Um eine optimale Lösung für beide Gleichungen zu erlangen, werden sie in einem Energiefunktional zusammengefasst.

$$E(u, v) = \iint_{x, y} (E_d + \alpha \cdot E_s) dy dx \quad (5)$$

Diese zu minimierende Gleichung ist die Grundstruktur vieler Lösungsansätze, wobei  $E(u, v)$  aus dem Datenterm  $E_d$  (d für "data") und dem Glattheitsterm  $E_s$  (s für smoothness) besteht. Alle folgenden Ansätze können in diese zwei Terme aufgeteilt werden. Im Falle der Methode von HORN und SCHUNCK ergibt sich das Energiefunktional:

$$E_{HS}(u, v) = \iint_{x, y} (I_x u + I_y v + I_t)^2 + \alpha \cdot (|\nabla u|^2 + |\nabla v|^2) dy dx \quad (6)$$

Die Variable  $\alpha$  ist ein Gewichtungsfaktor, mit dem der Einfluss der beiden Bedingungen auf das Minimierungsproblem reguliert werden kann. Hierbei muss beachtet werden, dass die partiellen Ableitungen der Flussbedingung  $I_x$ ,  $I_y$  und  $I_t$  auch negativ werden können. Dies ist der Grund, weshalb dieser Term quadriert wird. Das Minimierungsproblem kann mit Hilfe der Variationsrechnung gelöst werden (LEITMANN 1981).

#### 3.1 FDRIG-Methode

Die von BROX et al. (2004) entwickelte Methode FDRIG ist nach ihren Charakteristiken *flussgetrieben (flow-driven)*, *robust*, *isotrop*, *gradientenkonstant* benannt und unterscheidet sich von HORN und SCHUNCKS Methode im Wesentlichen durch einen modifizierten Datenterm. Der Datenterm kann in zwei Einheiten  $E_{d1}$  und  $E_{d2}$  zerlegt werden,

$$E_{FDRIG, d} = E_{FDRIG, d1} + E_{FDRIG, d2} \quad (7)$$

wobei  $E_{FDRIG, d1}$  dem nicht linearisierten Datenterm von HORN und SCHUNCK entspricht. Da die alleinige Annahme von konstanten Intensitätswerten, wie sie in Gleichung (1) impliziert ist, aber sehr stark anfällig gegen Helligkeits-

variationen zwischen den Szenen ist, führt Brox eine zusätzliche Bedingung ein. Ein Kriterium, das additive Veränderungen in den Grauwerten zulässt, ist die Konstanz des Grauwertgradienten. Somit ergibt sich die zweite Annahme, die invariant gegenüber monotonen Grauwertvariationen ist, zu:

$$\nabla I(x + dx, y + dy, t + dt) = \nabla I(x, y, t) \quad (8)$$

Für den zweiten Teil des Datenterms ergibt sich in nicht linearisierter Vektorschreibweise, wenn  $\mathbf{x} := (x, y, t)^T$  und  $\mathbf{v} := (u, v, l)^T$  ist und quadriert wird:

$$E_{FDRIG,d2} = |\nabla I(\mathbf{x} + \mathbf{v}) - \nabla I(\mathbf{x})| \quad (9)$$

Der gesamte Datenterm in Vektorschreibweise ist schließlich

$$E_{FDRIG,d} = \Psi \left( |I(\mathbf{x} + \mathbf{v}) - I(\mathbf{x})|^2 + \gamma |\nabla I(\mathbf{x} + \mathbf{v}) - \nabla I(\mathbf{x})|^2 \right) \quad (10)$$

Dabei wird zum einen wieder ein Gewichtungsfaktor  $\gamma$  eingeführt, der den Einfluss der beiden Datenterme reguliert und zum anderen wird auf den gesamten Ausdruck eine Straffheitsfunktion  $\Psi$  angewandt. Diese von HUBER (1981) vorgeschlagene Funktion  $\Psi(s^2) = \sqrt{s^2 + \varepsilon^2}$  mit  $\varepsilon = 0.001$  reduziert den Einfluss von Ausreißern, die die Konstanzannahmen in (1) und (8) stark verletzen. Sie beschreibt eine lineare Bestrafung der Ausreißer und führt so zu einem robusteren Ergebnis. PAPANBERG et al. (2006) untersuchen neben den Annahmen der Konstanz der Grauwerte und der Konstanz der zugehörigen Gradienten auch die Konstanz höherer Ordnung, wie die der Hesse-Matrix oder der Laplace-Matrix. Je höher jedoch die Ordnung wird, desto schwieriger und komplexer wird es, das Energiefunktional zu lösen und desto größer wird die Anfälligkeit gegenüber Bildrauschen. Der Glattheitsterm  $E_s$  ist mit dem von Horn und Schunk entwickelten Term bis auf die Multiplikation mit der schon oben erwähnten Funktion  $\Psi$ , die für die Erhaltung von Diskontinuitäten im Flussfeld verantwortlich ist, identisch (RUDIN et al. 1992, ACAR & VOGEL 1994). Diese

Art von Glattheitsterm wird auch flussgetrieben und isotrop genannt. Somit ergibt sich abschließend das Energiefunktional zu:

$$E_{FDRIG}(\mathbf{v}) = \iint_{x,y} \Psi \left( |I(\mathbf{x} + \mathbf{v}) - I(\mathbf{x})|^2 + \gamma |\nabla I(\mathbf{x} + \mathbf{v}) - \nabla I(\mathbf{x})|^2 \right) + \alpha \cdot \Psi \left( |\nabla u|^2 + |\nabla v|^2 \right) dy dx \quad (11)$$

Die Minimierung dieses Funktionals kann mit Hilfe von Euler-Lagrange-Gleichungen durchgeführt werden. Damit diese Methode auch für größere Pixelbewegungen korrekte Ergebnisse liefert, wird eine grob-zu-fein Strategie bei der Berechnung der optischen Flusskomponenten auf Basis von Bildpyramiden durchgeführt.

### 3.2 CLG-Methode

Die von BRUHN et al. (2005b) entwickelte CLG-Methode ("Combined-Local-Global") ist eine Kombination der schon beschriebenen globalen Methode von HORN und SCHUNK und des lokalen Ansatzes von LUCAS & KANADE (1981). Die meisten lokalen Methoden zeichnen sich durch ein hohes Maß an Robustheit aus. Jedoch haben sie den großen Nachteil, dass sie keine dichten optischen Flussfelder liefern. Globale Methoden, wie diejenigen von HORN und SCHUNK, erzeugen dagegen ein 100% dichtes Flussfeld, sind aber anfälliger gegenüber Rauschen. Aus diesem Grund wird bei der CLG-Methode versucht, die jeweiligen Vorteile der Methoden zu kombinieren. Durch eine Einbettung der lokalen Methode von LUCAS und KANADE in die globale Methode von HORN und SCHUNK wird ein Ergebnis erzielt, das sowohl robust gegenüber Rauschen ist, als auch eine Dichte von 100% aufweist. Die Kombination der Methoden lässt sich durch Einführung von  $\mathbf{v} = (u, v, l)^T$  und  $|\nabla \mathbf{v}|^2 = |\nabla u|^2 + |\nabla v|^2$  und folgende Notation in Matrixschreibweise herleiten:

$$\nabla_3 I = (I_x, I_y, I_t)^T \quad (12)$$

$$J_\rho(\nabla_3 I) = K_\rho * (\nabla_3 I \nabla_3 I^T) \quad (13)$$

Hierbei bezeichnet  $K_\rho$  einen Glättungsfilter, der auf die Flussgleichung angewendet wird. Die von LUCAS und KANADE zu minimierende Funktion in quadratischer Form ergibt sich dann zu

$$E_{LK}(\mathbf{v}) = \mathbf{v}^T J_\rho(\nabla_3 I) \mathbf{v}. \quad (14)$$

Hier wird angenommen, dass der unbekannte optische Flussvektor innerhalb einer Umgebung der Größe  $\rho$  konstant ist. Wird der Glättungsterm  $K_\rho$  weggelassen, werden lediglich die Komponenten parallel zu den Bildgradienten bestimmt. Analog wird aus dem Energiefunktional von HORN und SCHUNK:

$$E_{HS}(\mathbf{v}) = \iint_{x \ y} \left( \mathbf{v}^T J_0(\nabla_3 I) \mathbf{v} + \alpha |\nabla \mathbf{v}|^2 \right) dy dx \quad (15)$$

Um daraus das kombinierte Energiefunktional abzuleiten, wird die Matrix  $J_0(\nabla_3 I)$  durch den Strukturtenor  $J_\rho(\nabla_3 I)$  mit  $\rho > 0$  ersetzt. Dadurch erhält man das Energiefunktional der CLG-Methode:

$$E_{CLG}(\mathbf{v}) = \iint_{x \ y} \left( \mathbf{v}^T J_\rho(\nabla_3 I) \mathbf{v} + \alpha |\nabla \mathbf{v}|^2 \right) dy dx \quad (16)$$

Die Minimierung dieses Funktionals wird – wie im Falle der FDRIG-Methode – mit Euler-Lagrange-Gleichungen durchgeführt. In (BRUHN et al. 2005a) wird beschrieben, wie die aus der CLG-Methode entstehenden Gleichungssysteme mit Hilfe von bidirektionalen Mehrgitterverfahren gelöst werden können. Diese numerischen Verfahren zählen zu den schnellsten Algorithmen zur Lösung von linearen und nicht linearen Gleichungssystemen. Ein Nachteil der CLG-Methode ist jedoch, dass nur kleine Bewegungen in der Größenordnung weniger Pixel detektiert werden können.

### 3.3 DDRAW-Methode

Die dritte hier vorgestellte Methode DDRAW wurde von NAGEL & ENKELMANN (1986) entwickelt. Wiederum beinhaltet das Akronym die wichtigsten Eigenschaften: *datengetrieben (data-driven)*, *robust*, *anisotrop*, *fortlaufende Kompensation (warping)*. Wie die vorangegangenen Ansätze unterscheidet auch dieser

einen Daten- und einen Glattheitsterm. Der Datenterm ist dem Term von HORN und SCHUNK sehr ähnlich. Zum einen aber ist der Datenterm nicht linearisiert, damit auch größere Bewegungen detektiert werden können, und zum anderen wird wieder eine Straffheitsfunktion eingeführt, wie sie bereits in der FDRIG-Methode beschrieben wurde. Der Datenterm ist:

$$E_{DDRAW,d} = \Psi \left( \left| I(\mathbf{x} + \mathbf{v}) - I(\mathbf{x}) \right|^2 \right) \quad (17)$$

Gegenüber dem einfachen Datenterm ist hier der Glattheitsterm komplexer. Wie auch beim FDRIG-Algorithmus sollen Diskontinuitäten erhalten bleiben. Allerdings wird hier nur parallel zur Kantenrichtung geglättet, während senkrecht zu den Kanten keine Glättung erfolgt. Diese Art von Glattheitsterm wird auch datengetrieben oder anisotrop genannt:

$$E_{DDRAW,s} = \nabla u^T P_{NE} \nabla I \nabla u + \nabla v^T P_{NE} \nabla I \nabla v \quad (18)$$

Die normalisierte Projektionsmatrix orthogonal zu  $\nabla I$  ist hierbei  $P_{NE} \nabla I$

$$P_{NE} \nabla I = \frac{1}{|\nabla I|^2 + \varepsilon_s^2} \begin{pmatrix} I_x^2 + \varepsilon_s^2 & -I_x I_y \\ -I_x I_y & I_y^2 + \varepsilon_s^2 \end{pmatrix} \quad (19)$$

Diese Matrix beschreibt die richtungsabhängige Glättung. Dies hat zur Folge, dass Diskontinuitäten erhalten bleiben. Die Konstante  $\varepsilon_s$  wird wiederum mit 0,001 angegeben. Dieser Regulierungsparameter verhindert, dass die Projektionsmatrix singularär wird. Das gesamte Energiefunktional dieser Methode ergibt sich schließlich zu:

$$E_{DDRAW} = \iint_{x \ y} \Psi \left( \left| I(\mathbf{x} + \mathbf{v}) - I(\mathbf{x}) \right|^2 \right) + \alpha \left( \nabla u^T P_{NE} \nabla I \nabla u + \nabla v^T P_{NE} \nabla I \nabla v \right) dy dx \quad (20)$$

Wiederum kann mit Hilfe des Parameters  $\alpha$  der Einfluss des Glattheitsterms auf das Minimierungsproblem gesteuert werden. Auch bei dieser Methode wird auf eine grob-zu-fein Strategie zurückgegriffen, wobei eine fortlaufende Verfeinerung der Komponenten des optischen Flusses durchgeführt wird.

### 3.4 WAVE-Methode

Als letzte Methode wird der in (BERNARD 2001) beschriebene Algorithmus (WAVE) betrachtet. Dieser Ansatz beruht auf der Verwendung von Waveletfunktionen, die für den Aufbau einer pyramidalen Struktur geeignet sind. Diese Repräsentation kann sehr effizient berechnet werden (STRANG 1996, MALLAT 1999). Ausgangspunkt ist wiederum die in Abschnitt 2 hergeleitete Flussgleichung (2). Diese Gleichung ist eine partielle Differentialgleichung erster Ordnung mit zwei Unbekannten. Eine oft angewandte Methode, um partielle Differentialgleichungen zu lösen, ist das Ergebnis als Linearkombination von Basisfunktionen auszudrücken. Typische Basisfunktionen sind harmonische Schwingungen oder finite Elemente. Nach der Diskretisierung der partiellen Differentialgleichung mittels solcher Funktionen, kann eine numerische Lösung gewonnen werden. In WAVE werden als Basisfunktionen zweidimensionale Waveletfunktionen verwendet. Sie besitzen gegenüber anderen gängigen Basisfunktionen eine Reihe von Vorteilen. Der größte Vorteil besteht in der Möglichkeit, verschiedene Auflösungen an unterschiedlichen Orten darzustellen. Somit kann die Auflösung des resultierenden Vektorfeldes an die Auflösung der jeweilig vorhandenen Bildinformation angepasst werden. Ein weiterer Vorteil besteht in der Erweiterbarkeit der Gleichung durch die Verwendung weiterer Waveletfunktionen. Die daraus entstehenden Gleichungen sind zwar untereinander korreliert, trotzdem steigt die Genauigkeit mit jeder hinzugefügten Gleichung an, ohne numerische Instabilitäten zu erzeugen. Dieser Vorteil gegenüber anderen Basisfunktionen und eine ausführliche Diskussion über die Lösung von partiellen Differentialgleichungen mit Hilfe von Wavelets als Basisfunktionen kann in (GOEDECKER & IVANOV 1998) nachgelesen werden. Werden 2D-Waveletfunktionen  $\Psi_{s,k_1,k_2}^n(x,y)$  – z. B. die wohlbekannten Haar-Wavelets – mit verschiedenen Richtungen  $n$  auf die optische Flussgleichung angewendet, ergibt sich ein Gleichungssystem mit  $n$  Gleichungen. Hierbei bezeichnet  $s$  den Skalierungsparameter und  $k_1, k_2$  die Translationsparameter in  $x$ - und  $y$ -Rich-

tung. Durch die Redundanz ist es möglich, die zwei Unbekannten  $u$  und  $v$  zu bestimmen.

$$\begin{aligned} \iint_{x,y} \left( \frac{\partial I}{\partial x} u(x,y) + \frac{\partial I}{\partial y} v(x,y) + \frac{\partial I}{\partial t} \right) \Psi_{s,k_1,k_2}^1(x,y) dx dy &= 0 \\ \iint_{x,y} \left( \frac{\partial I}{\partial x} u(x,y) + \frac{\partial I}{\partial y} v(x,y) + \frac{\partial I}{\partial t} \right) \Psi_{s,k_1,k_2}^2(x,y) dx dy &= 0 \\ &\vdots \\ \iint_{x,y} \left( \frac{\partial I}{\partial x} u(x,y) + \frac{\partial I}{\partial y} v(x,y) + \frac{\partial I}{\partial t} \right) \Psi_{s,k_1,k_2}^n(x,y) dx dy &= 0 \end{aligned} \quad (21)$$

Formel (21) lässt sich in kompakter Schreibweise zusammenfassen zu:

$$\iint_{x,y} \left( \frac{\partial I}{\partial x} u(\mathbf{x}) + \frac{\partial I}{\partial y} v(\mathbf{x}) + \frac{\partial I}{\partial t} \right) \Psi_{s,\mathbf{k}}^n(\mathbf{x}) dx dy = 0 \quad \forall n = 1, \dots, N \quad (22)$$

Unter Beachtung, dass die Wavelet-Funktionen limitierten und symmetrischen Support (Bereich einer Funktion, der von 0 verschieden ist) haben, folgt:

$$\begin{aligned} \iint_{x,y} \frac{\partial I}{\partial x} u(2^s \mathbf{k}) \Psi_{s,\mathbf{k}}^n(\mathbf{x}) dx dy \\ + \iint_{x,y} \frac{\partial I}{\partial y} v(2^s \mathbf{k}) \Psi_{s,\mathbf{k}}^n(\mathbf{x}) dx dy \\ + \iint_{x,y} \frac{\partial I}{\partial t} \Psi_{s,\mathbf{k}}^n(\mathbf{x}) dx dy = 0 \quad \forall n = 1, \dots, N \end{aligned} \quad (23)$$

Zieht man die von  $x$  und  $y$  unabhängigen Teile ( $u, v, t$ ) aus dem Intergral heraus und schreibt das Integral als inneres Produkt

$$\int f(t) g^*(t) = \langle f, g \rangle \quad (24)$$

ergibt sich daraus:

$$\begin{aligned} \left\langle I, \frac{\Psi_{s,\mathbf{k}}^n}{\partial x} \right\rangle u(2^s \mathbf{k}) + \left\langle I, \frac{\Psi_{s,\mathbf{k}}^n}{\partial y} \right\rangle v(2^s \mathbf{k}) \\ + \frac{\partial}{\partial t} \langle I, \Psi_{s,\mathbf{k}}^n \rangle = 0 \\ \forall n = 1, \dots, N. \end{aligned} \quad (25)$$

In diskreter Schreibweise mit  $dt = 1$  entsteht das folgende Gleichungssystem:

$$\begin{aligned} & \left\langle \frac{I(t+1)+I(t)}{2}, \frac{\Psi_{s,k}^n}{\partial x} \right\rangle u(2^s \mathbf{k}) \\ & + \left\langle \frac{I(t+1)+I(t)}{2}, \frac{\Psi_{s,k}^n}{\partial y} \right\rangle v(2^s \mathbf{k}) \\ & + (I(t+1) - I(t), \Psi_{s,k}^n) = 0 \quad \forall n=1, \dots, N. \quad (26) \end{aligned}$$

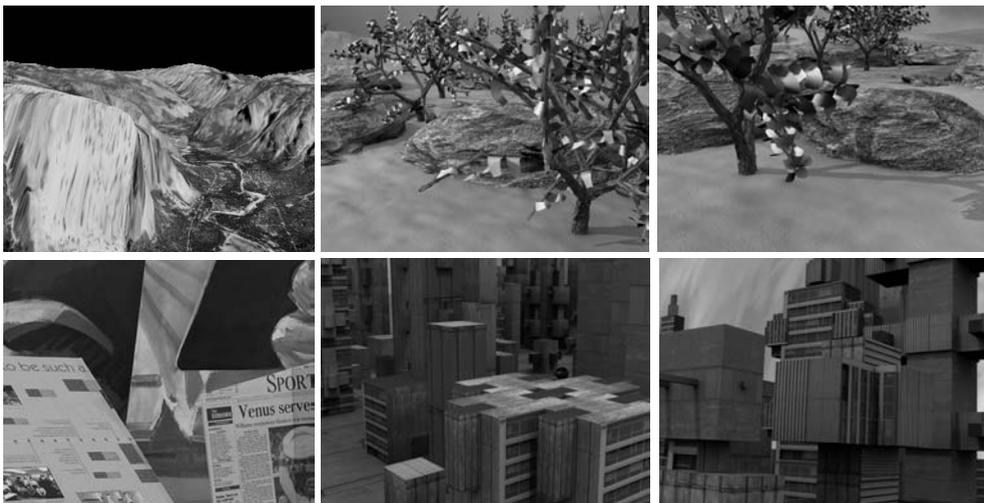
Dieses Gleichungssystem lässt sich nach dem Unbekanntenpaar  $(u, v)$  auflösen. Details zu Herleitung und Lösung des optischen Flusses mittels Wavelets finden sich in (BERNARD 1999). Als Wavelet-Kerne wurden sowohl Deslaurier-Dubuc-Filterkerne (DESLAURIERS & DUBUC 1989) als auch Haarfilterkerne untersucht. Ähnliche Wavelet-basierte Ansätze werden auch bei (MAGAREY & KINGSBURY 1998) aufgezeigt.

## 4 Evaluation

Die Implementierung der Methoden FDRIG, DDRAW und CLG sind Bestandteil der HALCON-Bibliothek (MVTec 2009). Die WAVE-Methode wurde im Rahmen einer Diplomarbeit als Extension-Package für HALCON implementiert.

Um ein Urteil über die Ergebnisse abgeben zu können, benötigt man ein Maß für die Genauigkeit bezüglich Referenzdaten. Für die Bewertung des optischen Flusses hat sich das Maß des Winkelfehlers fest etabliert (FLEET & JEPSON 1990). Dieser Fehler beschreibt die Richtungsabweichung des geschätzten optischen Flusses vom wahren Fluss. Voraussetzung hierfür ist, dass der wahre optische Fluss bekannt ist. Aus diesem Grund wurden für die Genauigkeitsuntersuchungen die in Abb. 2 dargestellten Szenen verwendet. Neben der Yosemite-Sequenz, die auf der Homepage von M.J. Black ([www.cs.brown.edu/people/black/](http://www.cs.brown.edu/people/black/)) zur Verfügung steht, wurden die Methoden an fünf weiteren Szenen getestet, die aus der Middlebury-Datenbank entnommen wurden ([vision.middlebury.edu/flow/](http://vision.middlebury.edu/flow/) – BAKER et al. 2007).

Der Winkelfehler eines Bildes ergibt sich über die Mittelbildung aller Winkelfehler zwischen den geschätzten Flussvektoren  $v_c$  und den wahren Flussvektoren  $v_t$ . Diese Vektoren werden als raumzeitliche Vektoren  $v = (u, v, I)$  dargestellt, wobei die zeitliche Komponente zu Eins (1 Frame) gesetzt wird. Durch die Einführung dieser dritten Komponente werden auch Vektorbeträge, die voneinander abweichen, bestraft. Zunächst werden die Vektoren normiert:



**Abb. 2:** Verwendete Test-Sequenzen von links oben nach rechts unten (Yosemite, Grove2, Grove3, Venus, Urban2, Urban3).

$$\mathbf{v}_e = \frac{1}{\sqrt{u_e^2 + v_e^2 + 1}} \begin{bmatrix} u_e \\ v_e \\ 1 \end{bmatrix} \quad (27)$$

$$\mathbf{v}_i = \frac{1}{\sqrt{u_i^2 + v_i^2 + 1}} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \quad (28)$$

Der Winkelfehler  $\varepsilon$  ergibt sich dann zu:

$$\varepsilon = \arccos(\mathbf{v}_e \cdot \mathbf{v}_i) \quad (29)$$

Eine ausführlich Beschreibung und Bewertung des Winkelfehlers kann in (FLEET 1992) nachgelesen werden. Neben der Evaluation der Genauigkeit wird auch die Laufzeit der einzelnen Algorithmen untersucht. Im Vordergrund steht der relative Vergleich und somit das Verhältnis zwischen den Algorithmen, wohingegen die unten angegebenen absoluten Laufzeiten eine untergeordnete Rolle spielen.

Die Messungen wurden mit Hilfe der Software HALCON durchgeführt. Die verwendete Hardware bestand aus einem AMD Athlon(tm) X2 Dual mit 2,21 GHz. Bei allen Algorithmen gibt es die Möglichkeit eine Vielzahl von Parametern einzustellen. Aus diesem Grund wurden für jeden Algorithmus Beispielparametersätze ausgewählt, die den Zusammenhang zwischen Geschwindigkeit und Genauigkeit deutlich machen sollen. Untersuchungen zu den einzelnen Parametern und deren exakten Einfluss auf die Berechnung des Flussfeldes, kann für die FDRIG Methode in (BROX et al. 2004) und für die CLG Methode in (BRUHN et al. 2005b) nachgelesen werden. In den Beispielparametersätzen weisen die Zusätze (Va = Very accurate, A = Accurate, Fa = Fast accurate, F = Fast) zur Bezeichnung des Algorithmus auf verschiedene Parametereinstellungen hin. Die Unterschiede dieser Parametersätze werden im Folgenden für jeden Algorithmus kurz erläutert:

- Bei der WAVE-Methode wurden für „Va“ und „A“ Deslaurier-Dubuc-Filter verwendet. Für „Fa“ und „F“ kamen die kompakteren Haarfilter zum Einsatz. Bei den Parametersätzen „Va“ und „Fa“ wurde aufgrund des nicht vorhandenen Regularisierungsterms zusätzlich eine Glättung der Kompo-

nenten des optischen Flusses mit einem 4x4-Medianfilter durchgeführt.

- Bei der FDRIG-Methode wurde der Regularisierungsparameter  $\alpha$  und der Gewichtungsfaktor  $\gamma$ , der den Einfluss der beiden Datenterme steuert, bei allen Parametersätzen identisch gewählt ( $\alpha=20$ ,  $\gamma=5$ ). Der Unterschied der Parametersätze liegt in der Coarse-to-Fine-Warping-Strategie. Das Unterscheidungsmerkmal zwischen „Va“ bzw. „A“ auf der einen und „Fa“ bzw. „F“ auf der anderen Seite ist die AuflösungsEbene, bis zu der das Warping durchgeführt wird. Bei „Va“ und „A“ wird im Gegensatz zu „Fa“ und „F“ die feinste Auflösung berechnet. Die Parametersätze „Va“ und „A“ bzw. „Fa“ und „F“ unterscheiden sich bezüglich der verwendeten Zyklen und der in den Zyklen durchgeführten Iterationen. Während in „Va“ und „Fa“ W-Zyklen verwendet werden, kommen bei „A“ und „F“ V-Zyklen zum Einsatz. W-Zyklen versprechen bei einer höheren Berechnungsdauer eine höhere Genauigkeit im Vergleich zu V-Zyklen.
- Auf die gleiche Weise wie bei der FDRIG-Methode werden auch die Parametersätze der DDRAW-Methode unterschieden. Der einzige Unterschied stellt der Regularisierungsparameter  $\alpha$  dar, der auf 1000 gesetzt wurde.
- Auch bei der CLG-Methode ist das Unterscheidungsmerkmal zwischen „Va“ und „A“ identisch wie bei der FDRIG-Methode. Die Ziffern 1 bzw. 2 geben bei der CLG-Methode folgende unterschiedliche Parameter an: Für „Va1“ und „A1“ wurde  $\rho = 1$  gewählt, während für „Va2“ und „A2“  $\rho = 1.8$  gewählt wurde.

In Abb. 3 wird die Relation zwischen der Geschwindigkeit und der Genauigkeit verschiedener Algorithmen grafisch veranschaulicht. Mit Hilfe der ID-Nummer ist der entsprechende Algorithmus in Tab. 1 referenzierbar. Zusätzlich sind in Tab. 1 neben den absoluten Genauigkeits- und Laufzeitmessungen auch relative Angaben aufgelistet. Um die Vergleichbarkeit der Algorithmen zu erleichtern wurde dem schnellsten bzw. genauesten Algorithmus der Wert 1,0 zugewiesen und alle weiteren Algorithmen als Faktoren dargestellt. In

**Tab. 1:** Evaluation verschiedener Algorithmen mit unterschiedlichen Parametersätzen.

ID	Algorithmus + Parametersätze	Yosemite				Venus			
		Winkelfehler		Laufzeit		Winkelfehler		Laufzeit	
		$\varepsilon$ in Grad	rel.	in ms	rel.	$\varepsilon$ in Grad	rel.	in ms	rel.
1	WAVE - Va	5,01°	1,4	780	13,0	12,05°	1,5	1337	11,5
2	WAVE - A	8,31°	2,3	610	10,2	16,62°	2,0	1076	9,3
3	WAVE - Fa	8,20°	2,3	398	6,6	10,70°	1,3	646	5,6
4	WAVE - F	12,87°	3,6	249	4,2	17,91°	2,2	399	3,4
5	FDRIG - Va	3,54°	1,0	308	5,1	8,15°	1,0	653	5,6
6	FDRIG - A	3,54°	1,0	154	2,6	9,70°	1,2	315	2,7
7	FDRIG - Fa	4,94°	1,4	98	1,6	8,99°	1,1	213	1,8
8	FDRIG - F	4,91°	1,4	60	1,0	10,75°	1,3	116	1,0
9	DDRAW - Va	9,94°	2,8	744	12,4	10,28°	1,3	1665	14,4
10	DDRAW - A	9,41°	2,7	281	4,7	11,08°	1,4	605	5,2
11	DDRAW - Fa	17,52°	5,0	191	3,2	11,07°	1,4	433	3,7
12	DDRAW - F	15,98°	4,5	83	1,4	12,24°	1,5	182	1,6
13	CLG - Va1	7,37°	2,1	110	1,8	20,42°	2,5	230	2,0
14	CLG - A1	7,94°	2,2	75	1,3	26,59°	3,3	151	1,3
15	CLG - Va2	3,73°	1,1	119	2,0	13,44°	1,6	249	2,1
16	CLG - A2	3,76°	1,1	85	1,4	18,96°	2,3	168	1,4

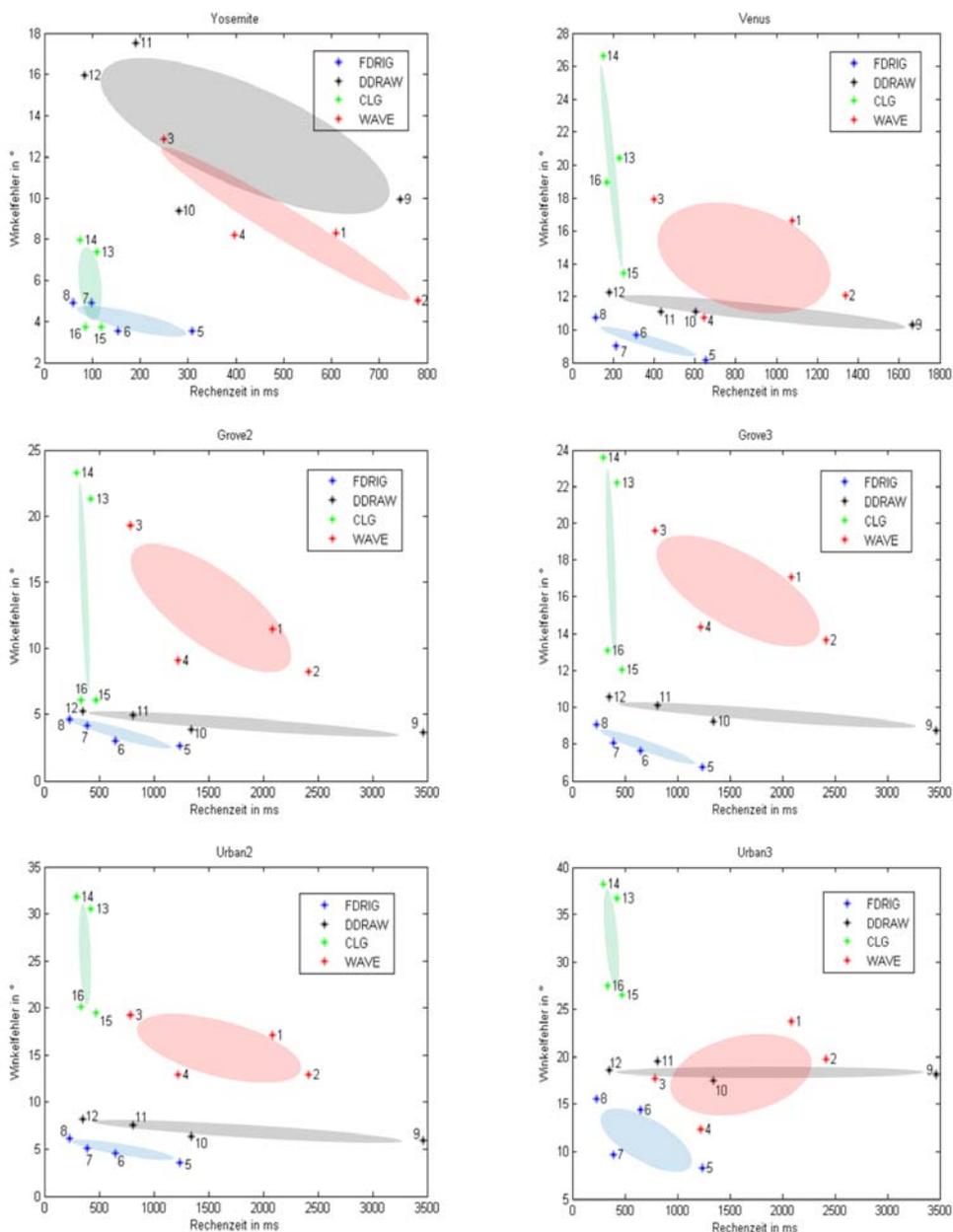
ID	Algorithmus + Parametersätze	Grove2		Grove3		Grove2/Grove3	
		Winkelfehler		Winkelfehler		Laufzeit	
		$\varepsilon$ in Grad	rel.	$\varepsilon$ in Grad	rel.	in ms	rel.
1	WAVE - Va	8,28°	3,1	13,62°	2,0	2413	10,4
2	WAVE - A	11,44°	4,3	17,09°	2,5	2086	9,0
3	WAVE - Fa	9,08°	3,4	14,35°	2,1	1218	5,3
4	WAVE - F	19,29°	7,3	19,62°	2,9	789	3,4
5	FDRIG - Va	2,65°	1,0	6,73°	1,0	1238	5,4
6	FDRIG - A	3,01°	1,1	7,65°	1,1	646	2,8
7	FDRIG - Fa	4,16°	1,6	8,07°	1,2	389	1,7
8	FDRIG - F	4,62°	1,7	9,07°	1,3	231	1,0
9	DDRAW - Va	3,61°	1,4	8,72°	1,3	3461	15,0
10	DDRAW - A	3,89°	1,5	9,21°	1,4	1345	5,8
11	DDRAW - Fa	4,94°	1,9	10,11°	1,5	812	3,5
12	DDRAW - F	5,25°	2,0	10,55°	1,6	347	1,5
13	CLG - Va1	21,25°	8,0	22,18°	3,3	427	1,8
14	CLG - A1	23,26°	8,8	23,56°	3,5	295	1,3
15	CLG - Va2	6,07°	2,3	12,07°	1,8	472	2,0
16	CLG - A2	6,10°	2,3	13,10°	1,9	332	1,4

ID	Algorithmus + Parametersätze	Urban2		Urban3		Urban2/Urban3	
		Winkelfehler		Winkelfehler		Laufzeit	
		$\varepsilon$ in Grad	rel.	$\varepsilon$ in Grad	rel.	in ms	rel.
1	WAVE - Va	12,91°	3,5	19,79°	2,4	2413	10,4
2	WAVE - A	17,08°	4,7	23,78°	2,9	2086	9,0
3	WAVE - Fa	12,93°	3,6	12,35°	1,5	1218	5,3
4	WAVE - F	19,22°	5,3	17,70°	2,2	789	3,4
5	FDRIG - Va	3,64°	1,0	8,23°	1,0	1238	5,4
6	FDRIG - A	4,55°	1,3	14,40°	1,7	646	2,8
7	FDRIG - Fa	5,12°	1,4	9,70°	1,2	389	1,7
8	FDRIG - F	6,13°	1,7	15,58°	1,9	231	1,0
9	DDRAW - Va	5,92°	1,6	18,13°	2,2	3461	15,0
10	DDRAW - A	6,43°	1,8	17,48°	2,1	1345	5,8
11	DDRAW - Fa	7,62°	2,1	19,51°	2,4	812	3,5
12	DDRAW - F	8,16°	2,2	18,53°	2,3	347	1,5
13	CLG - Va1	30,56°	8,4	36,70°	4,5	427	1,8
14	CLG - A1	31,78°	8,7	38,25°	4,6	295	1,3
15	CLG - Va2	19,45°	5,3	26,49°	3,2	472	2,0
16	CLG - A2	20,16°	5,5	27,53°	3,3	332	1,4

Abb. 3 ist deutlich zu erkennen, dass einige Algorithmen eine diagonale Struktur aufweisen, die von links oben nach rechts unten führt, wie die schematisch in Abb. 3 eingezeichneten Ellipsen zeigen. Dies bestätigt das zu erwartende Verhalten, dass mit steigender

Genauigkeit gleichzeitig die Rechenzeit zunimmt.

Insgesamt betrachtet erzielt der FDRIG-Algorithmus die besten Ergebnisse. Er liefert in allen Test-Sequenzen sowohl die schnellsten, als auch die genauesten Ergebnisse. Der



**Abb. 3:** Relation zwischen Rechenzeit und Winkelfehler verschiedener Algorithmen.

DDRAW-Algorithmus erzielt sowohl in der Genauigkeit als auch in der Geschwindigkeit geringfügig schlechtere Resultate. Bemerkenswert ist, dass sich die Genauigkeit bei längerer Rechenzeit kaum steigert, wobei jedoch auffällt, dass der DDRAW-Algorithmus in der Yosemite-Sequenz bezüglich der Genauigkeit recht schlecht abschneidet. Ein Grund für die deutlich unterschiedlichen Ergebnisse zwischen der Yosemite-Sequenz und den restlichen untersuchten Sequenzen liegt in der Glattheit des optischen Flusses. Im Gegensatz zu dem kontinuierlichen Verlauf des Flusses in der Yosemite-Sequenz, treten im Flussfeld der anderen Szenen häufig Diskontinuitäten auf. Diese szenenspezifische Eigenschaft führt auch dazu, dass der CLG-Algorithmus in der Yosemite-Sequenz eine höhere Genauigkeit aufweist als in den restlichen Szenen. In allen weiteren Szenen erreicht der CLG-Algorithmus, dessen Rechenzeiten immer sehr gering sind, jedoch niemals die Genauigkeit des FDRIG- bzw. DDRAW-Algorithmus.

Weiterhin ist zu bemerken, dass die Größe des Glättungskerns  $\rho$  eine größere Auswirkung auf die Genauigkeit hat als die verwendete Auflösungsebene bis zu der das Warping durchgeführt wird. Dies spiegelt sich in dem Ergebnis, dass der Parametersatz CLG-A2 eine höhere Genauigkeit als der Parametersatz CLG-Va1 erreicht. Ähnliche Effekte sind auch bei anderen Parametersätzen zu beobachten. Der WAVE-Algorithmus schneidet unter Ausnahme der Urban3-Sequenz, in der die zweitbeste Genauigkeit erreicht wurde, im Vergleich am schlechtesten ab. Abschließend sollte darauf hingewiesen werden, dass die Parametersätze nicht auf die Szenen optimiert wurden, sondern Standardparameter gewählt wurden, die erfahrungsgemäß gute Ergebnisse für unterschiedliche Sequenzen liefern. Dies bedeutet, dass durch Anpassung der Parameter auf einzelne Szenen durchaus bessere Genauigkeiten erzielt werden können.

## 5 Zusammenfassung

In diesem Artikel wurden nach einer Beschreibung der grundlegenden Prinzipien vier verschiedene Methoden zur Berechnung des

optischen Flusses miteinander verglichen. Alle Ansätze zeichnen sich durch ihre Effizienz aus. Jedoch konnte anhand verschiedener Testszene gezeigt werden, dass sich die untersuchten Methoden hinsichtlich Genauigkeit, Geschwindigkeit und Auflösung deutlich voneinander unterscheiden. Am besten schnitten die Multigridverfahren FDRIG und CLG ab.

## Danksagung

Diese Arbeit wurde von der MVTec Software GmbH im Rahmen einer Diplomarbeit unterstützt, wofür sich der Erstautor herzlich bedanken möchte.

## Referenzen

- ACAR, R. & VOGEL, C., 1994: Analysis of Bounded Variation Penalty Methods for Ill-Posed Problems. – *Inverse Problems* **10** (6): 1217–1229.
- BAKER, S., SCHARSTEIN, D., LEWIS, J.P., ROTH, S., BLACK, M.J. & SZELISKI, R., 2007: A Database and Evaluation Methodology for Optical Flow. – 11TH IEEE International Conference on Computer Vision **5**: 1–8.
- BARRON, J.L., FLEET, D.J. & BEAUCHEMIN, S.S., 1994: Performance of Optical Flow Techniques. – *International Journal of Computer Vision* **12** (1): 43–77.
- BERNARD, C.P., 2001: Discrete Wavelet Analysis for Fast Optic Flow Computation. – *Applied and Computational Harmonic Analysis* **11** (1): 32–63.
- BERNARD, C.P., 1999: Wavelets and Ill Posed Problems: Optic Flow and Scattered Data Interpolation. – PhD Thesis, Ecole Polytechnique.
- BROX, T., BRUHN, A., PAPPENBERG, A. & WEICKERT, J., 2004: High Accuracy Optical Flow Estimation Based on a Theory for Warping. – 8th European Conference on Computer Vision **4**: 25–36.
- BRUHN, A., WEICKERT, J., FEDDERN, C., KOHLBERGER, T. & SCHNÖRR, C., 2005a: Variational Optical Flow Computation in Real Time. – *IEEE Transactions on Image Processing* **14** (5): 608–615.
- BRUHN, A., WEICKERT, J. & SCHNÖRR, C., 2005b: Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods. – *International Journal of Computer Vision* **61** (3): 211–231.

- DESLAURIERS, G. & DUBUC, S., 1989: Symmetric Iterative Interpolation Processes. – *Constructive Approximation* **5** (1): 49–68.
- DRESCHLER, L. & NAGEL, H.H., 1982: Volumetric model and trajectory of a moving car derived from monocular TV frame sequence of a street scene. – *Computer Graphics and Image Processing* **20**: 199–228.
- FLEET, D.J., 1992: Measurement of Image Velocity. – Kluwer, Norwell, MA, USA.
- FLEET, D.J. & JEPSON, A., 1990: Computation of Component Image Velocity from Local Phase Information. – *International Journal of Computer Vision* **5** (1): 77–104.
- GOEDECKER, S. & IVANOV, O., 1998: The Solution of Multi-Scale Partial Differential Equations using Wavelets. – *Computers in Physics* **12** (6): 548–555.
- HARTLEY, R. & ZISSERMAN, A., 2004: Multiple View Geometry in Computer Vision. – Cambridge University Press, 2. Edition.
- HORN, B.K.P. & SCHUNCK, B., 1981: Determining Optical Flow. – *Artificial Intelligence* **17** (1–3): 185–203.
- HUBER, P.J., 1981: Robust statistics. – Wiley, New York, USA.
- LUCAS, B.D. & KANADE T., 1981: An Iterative Image Registration Technique with an Application to Stereo Vision. – 7th International Joint Conference on Artificial Intelligence: 674–679.
- MAGAREY, J. & KINGSBURY, N., 1998: Motion Estimation using a Complex-Valued Wavelet Transform. – *IEEE Transactions on Signal Processing* **46** (4): 1069–1084.
- MALLAT, S., 1999: A Wavelet Tour of Signal Processing. – Academic Press.
- MVTec, 2009: HALCON 9.0.1 –Reference Manual, MVTec Software GmbH.
- LEITMANN, G., 1981: The Calculus of Variations and Optimal Control. – *Mathematical Concepts and Methods in Science and Engineering* (24). – Plenum Press, New York, USA.
- NAGEL, H.H. & ENKELMANN, W., 1986: An Investigation of Smoothness Constraints for the Estimation of Displacement Vectorfields from Image Sequences. – *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8** (5): 565–593.
- PAPENBERG, N., BRUHN A., BROX, T., DIDAS S. & WEICKERT J., 2006: Highly Accurate Optic Flow Computation with Theoretically Justified Warping. – *International Journal of Computer Vision* **67** (2): 141–158.
- RUDIN, L., OSHER S. & FATEMI E., 1992: Nonlinear Total Variation Based Noise Removal Algorithms. – *Physica D: Nonlinear Phenomena* **60** (1–4): 259–268.
- STRANG, G., 1996: Wavelets and Filter Banks. – Wellesley Cambridge Press.

#### Anschriften der Autoren:

Dipl.-Ing. DANIEL FREY, Technische Universität München, Lehrstuhl für Methodik der Fernerkundung, D-80333 München, Tel.: +49-89-289-22673, e-mail: Daniel.Frey@bv.tu-muenchen.de  
 Dr.-Ing. MARKUS ULRICH, MVTec Software GmbH, Neherstr.1, D-81675 München, Tel.: +49-89-4576950, e-mail: ulrich@mvtec.com  
 Prof. Dr.-Ing. STEFAN HINZ, Universität Karlsruhe, Lehrstuhl für Fernerkundung und Bildverarbeitung, D-76131 Karlsruhe, Tel.: +49-721-608-2314, e-mail: Stefan.Hinz@ipf.uni-karlsruhe.de

Manuskript eingereicht: März 2009  
 Angenommen: November 2009