

# INAUGURAL-DISSERTATION

ZUR  
ERLANGUNG DER DOKTORWÜRDE  
DER  
NATURWISSENSCHAFTLICH-MATHEMATISCHEN GESAMTFAKULTÄT  
DER  
RUPRECHT-KARLS-UNIVERSITÄT  
HEIDELBERG

vorgelegt von  
Diplom-Informatikerin, Diplom-Mathematikerin Claudia Kondermann  
aus Bocholt

Tag der mündlichen Prüfung: 13. Juli 2009



---

# Postprocessing and Restoration of Optical Flows

1. Gutachter: **PD. Dr. Christoph Garbe**  
Digitale Bildverarbeitung  
Interdisziplinäres Zentrum  
für Wissenschaftliches Rechnen  
Universität Heidelberg

---

2. Gutachter: **Prof. Dr. Rudolf Mester**  
Visuelle Sensorik  
und Informationsverarbeitung  
Universität Frankfurt

---



---

## **Abstract**

The notion “Optical flow” refers to the apparent motion in the image plane produced by the projection of the real 3D motion onto the 2D image plane. The thesis at hand addresses postprocessing and restoration methods for arbitrarily computed optical flow fields. Many motion estimators have been proposed during the last three decades, but all of them suffer from shortcomings in difficult situations. Hence, it is of utmost importance for any optical flow measurement technique to give a prediction of the quality and reliability of each individual flow vector. Yet, a sound, universally applicable, and statistically motivated confidence measure for optical flow measurements is still missing today. Based on such information, erroneous optical flow fields can be restored or improved by means of inpainting techniques.

This thesis introduces three confidence measures, which evaluate the reliability of optical flow vectors. In contrast to previously employed methods, these confidence measures are based on learned motion models and are, thus, statistically motivated, they are independent of the original flow computation method and yield more accurate predictions on the quality of optical flow vectors. The thesis puts a second focus on the restoration of optical flow fields, where it transfers inpainting techniques from the restoration of images to the field of motion recovery. Since the reconstruction process in case of motion fields can use the image sequence as additional source of information, a novel motion inpainting approach is proposed. It combines motion and image information in one functional and, thus, allows to control the orientation of the reconstruction algorithm based on image edges.

## **Zusammenfassung**

Der Begriff “Optischer Fluss” bezeichnet die scheinbare Bewegung auf der Bildebene, die durch die Projektion der realen 3D Bewegung auf die 2D Bildebene erzeugt wird. Die vorliegende Arbeit befasst sich mit der Nachbearbeitung und Wiederherstellung beliebig berechneter Flussfelder. Viele Bewegungsschätzer sind in den letzten drei Jahrzehnten vorgeschlagen worden, aber alle weisen in schwierigen Situationen Mängel auf. Deshalb ist es von höchster Wichtigkeit für jede Flussberechnungsmethode, eine Schätzung der Qualität und Zuverlässigkeit für jeden einzelnen Flussvektor anzugeben. Jedoch fehlt ein solides, allgemein anwendbares und statistisch motiviertes Konfidenzmaß für Flussberechnungen bis heute. Basierend auf diesen Informationen können mittels sogenannter “inpainting” Methoden fehlerhafte Flussfelder wiederhergestellt oder verbessert werden. Im Rahmen dieser Arbeit werden drei Konfidenzmaße vorgeschlagen, die die Zuverlässigkeit von Flussvektoren bewerten. Im Unterschied zu den bisher verwendeten Methoden basieren diese Konfidenzmaße auf gelernten Bewegungsmodellen und sind somit statistisch motiviert, sie sind unabhängig von der zu Grunde liegenden Flussberechnungsmeth-

---

ode und liefern genauere Vorhersagen über die Qualität der Flussvektoren. Ein zweiter Schwerpunkt der Arbeit liegt auf der Wiederherstellung von Flussfeldern. Sie überträgt “inpainting” Methoden von der Bildrestaurierung auf das Feld der Bewegungsrekonstruktion. Da der Rekonstruktionsprozess im Falle von Bewegungsfeldern zusätzlich die Bildsequenz als Informationsquelle benutzen kann, wird ein neuer Ansatz zur Rekonstruktion von Bewegungen vorgeschlagen. Er kombiniert Bewegungs- und Bildinformationen in einem Funktional und erlaubt dadurch die Orientierung des Rekonstruktionsprozesses an Bildkanten.

# Acknowledgements

During the last three years I have worked with people from different mathematical backgrounds, whose ideas and criticism were vital to this work.

First of all I want to thank PD Dr. Christoph Garbe for offering me the PhD position despite the long waiting time, for introducing me to the world of optical flows, for inspiring discussions, his constant support and nightshifts of paper revisions. I am indebted to Prof. Dr. Rudolf Mester, whose support and commitment in the field of statistics constitute a significant contribution to the success of this thesis. It was a pleasure to work with both of you. Many thanks also go to Prof. Dr. Martin Rumpf and Benjamin Berkels for fruitful discussions on the topic of optical flow restoration. Finally, I thank Daniel Kondermann for many hours of interesting discussions.

Last but not least, I want to thank my colleagues for this great working environment at IWR and HCI and all the fun we have had during our joint free time activities. Especially I want to thank Barbara Werner and my office neighbor Nikos Gianniotis for the many enlivening conversations and joyful hours we have spent together.



# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Introduction . . . . .	3
1.2	Thesis Outline . . . . .	6
1.3	Notation . . . . .	8
<b>2</b>	<b>Mathematical Preliminaries</b>	<b>9</b>
2.1	The Calculus of Variations . . . . .	9
2.2	Hypothesis Testing . . . . .	11
2.3	Best Linear Unbiased Estimators . . . . .	14
2.4	Intrinsic Dimensions . . . . .	17
2.5	Principal Component Analysis . . . . .	19
2.6	The Least Squares Method . . . . .	22
<b>3</b>	<b>Optical Flow Estimation</b>	<b>25</b>
3.1	Local Methods . . . . .	25
3.2	Global Methods . . . . .	27
<b>4</b>	<b>Error Analysis</b>	<b>29</b>
4.1	Introduction . . . . .	29
4.2	Discussion of the Angular Error . . . . .	33
4.3	The Joint Distribution of Optical Flow Estimation . . . . .	35
4.4	Experiments and Results . . . . .	37
4.5	Summary and Conclusion . . . . .	44
<b>5</b>	<b>Predictability and Situation Measures</b>	<b>45</b>
5.1	Introduction . . . . .	45
5.2	Classification of Situation Measures . . . . .	48
5.3	Experiments and Results . . . . .	58
5.4	Summary and Conclusion . . . . .	67

<b>6 Surface Measures</b>	<b>69</b>
6.1 Introduction . . . . .	69
6.2 Surface Measures . . . . .	71
6.3 Computational Issues . . . . .	74
6.4 Experiments and Results . . . . .	75
6.5 Summary and Conclusion . . . . .	79
<b>7 Statistical Confidence Estimation</b>	<b>81</b>
7.1 Introduction . . . . .	81
7.2 A Confidence Measure Based on Linear Subspace Projections . . . . .	84
7.3 A Statistical Confidence Measure . . . . .	89
7.4 Applicability of the Test . . . . .	92
7.5 Application to Sparse Vector Fields . . . . .	92
7.6 A Nonlinear Extension . . . . .	93
7.7 Results . . . . .	94
7.8 Summary and Conclusion . . . . .	102
<b>8 A Model Based Optical Flow Algorithm</b>	<b>105</b>
8.1 Introduction . . . . .	105
8.2 Parameter Estimation . . . . .	107
8.3 Confidence Estimation . . . . .	109
8.4 Integration of the Model into a Global Optical Flow Method . . . . .	109
8.5 Results . . . . .	111
8.6 Summary and Conclusion . . . . .	126
<b>9 The Restoration of Optical Flow Fields</b>	<b>129</b>
9.1 Introduction . . . . .	129
9.2 Diffusion Based Motion Inpainting . . . . .	131
9.3 TV Motion Inpainting . . . . .	132
9.4 Image Guided Motion Inpainting . . . . .	133
9.5 Experiments and Results . . . . .	136
9.6 Summary and Conclusion . . . . .	142
<b>10 Conclusions and Perspectives</b>	<b>143</b>
10.1 Summary and Conclusion . . . . .	143
10.2 Future Research . . . . .	145

# Chapter 1

---

## Introduction

### 1.1 Introduction

Optical flow is the apparent motion in the image plane produced by the projection of the real 3D motion onto the 2D image plane. Since the 1980s, optical flow has been an important subject of research in computer vision, and many optical flow estimators have been proposed so far. Despite these long standing research activities, the equally important field of sound quality and reliability measures has often been ignored. A few measures called “confidence measures” have been proposed, yet they either merely estimate the complexity of the image sequence without taking account of the actual flow field at all, or they are derived from and, thus, depend on specific flow computation methods. To the best of our knowledge, no comprehensive approach has been published in literature that assesses the quality of computed flow fields in a postprocessing step independent of the actual flow estimation procedure. Furthermore, none of these measures are statistically substantiated. If sound reliability information was available, motion restoration methods could be employed to reconstruct erroneous flow fields with lower average errors. Such methods for an automatic refinement of computed flow fields would be of high interest for many applications.

In fact, it would also be possible to directly integrate confidence measures into optical flow computation methods, as they basically consist of additional constraints and knowledge on the flow field. Yet, if different constraints are combined, they may conflict or contradict each other or drastically increase the computation time. Just as in the case of NP-problems, it is simpler to verify the accuracy of an already computed solution than to find the solution itself.

Let  $D := \Omega \times [0, T]$  denote a spatio-temporal image domain, where  $\Omega \subseteq \mathbb{R}^d$ ,  $d \in \{2, 3\}$ , stands for the spatial domain of the sequence and  $[0, T], T \in \mathbb{N}$ , for its time interval.

Let, furthermore,  $I$  refer to an image sequence defined as

$$I : D \rightarrow \mathbb{R} . \quad (1.1)$$

Then the notion "optical flow" refers to the displacement field  $\mathbf{u}$  of corresponding pixels in subsequent frames of the image sequence

$$\mathbf{u} : D \rightarrow \mathbb{R}^d . \quad (1.2)$$

Confidence measures can be defined as mappings from the spatio-temporal image domain  $D$ , the image sequence  $I$  and a  $d$ -dimensional displacement vector to the interval of confidence  $[0, 1]$ , where 1 stands for high and 0 for low confidence:

$$\varphi : D \times I \times \mathbb{R}^d \rightarrow [0, 1]. \quad (1.3)$$

Optical flow is employed in many applications today such as medicine, control systems, data compression, robot navigation, as well as pedestrian and vehicle tracking.

In medical applications it is often necessary to screen the movement of a patient over a short period of time or to compare images of an organ before and after treatment with medication. Since the patient and his heart move, the motion must be compensated before further processing steps can be taken. To this end, optical flow fields can be computed to register different frames. E.g. in [91] the intensity of X-ray images is improved by screening the patient for a longer period of time followed by subsequent motion compensation.

Especially in medical applications it is important to gain information on the accuracy of the motion field, since symptoms of diseases are often deduced from extremely small indications or color differences. Incorrect motion fields can easily produce such differences when subsequent frames are compared or registered. Furthermore, to fully register two frames, dense optical flow fields are necessary. Hence, confidence measures combined with an automatic motion restoration algorithm would be desirable in the field of medical imaging.

Optical flow estimation is also important for physical applications and control systems. In [55] the optical flow estimates are used to monitor the motion of flames in order to optimize the combustion process. In [46, 45] the temperature change of the water surface, the heat flux, is estimated to analyze the air-sea gas transfer during the investigation of global climatic changes.

For both applications accurate motion estimates are important for further processing steps. In case of combustion monitoring the heat of the process is regulated based on information drawn from the motion field. Erroneous optical flow measurements can, therefore, have undesired or even fatal effects on these control systems, such as a low effectiveness leading to unnecessarily high energy dissipation and operating expenses. In case of the analysis of the heat flux, incorrect motion estimates can entail experimental

errors and, thus, incorrect scientific conclusions. Depending on the control system or subsequent analysis steps, such as higher order derivatives of the flow field, dense optical flow fields may be required as well. Hence, accuracy measurements and restoration algorithms are also beneficial for physical applications.

Another important field for optical flow is data compression, e.g. the compression of large video sequences [43]. If the first frame and the corresponding motion field is known, a rough version of the following frame can be obtained by means of warping. The resulting version of the following frame can be taken as a reference in the encoding stage. In this way, the required compensation information is reduced, which improves the compression ratio.

For data compression accurate motion estimates are also a prerequisite, since they improve the quality of the computed reference frame and, thus, decrease the necessary compensation information. Furthermore, dense flow fields are required in order to compute the reference frame. Reliable confidence measures together with motion inpainting algorithms for the subsequent restoration of the motion field could, therefore, yield significant improvements of currently used methods in the field of video compression, reducing compression artifacts at increasing compression ratios.

In robot navigation optical flow fields are used for obstacle detection, collision avoidance and to trail moving objects, e.g. in [34]. The basic concept was inspired by the way bees navigate. They try to balance the amount of motion occurring on either side of them. If the robot wants to avoid obstacles, then it should turn away from the side that shows more motion in the optical flow field, since this indicates a possible approach to a stationary obstacle. Similarly, if a target is trailed, the robot should turn to the side that shows more motion. In this way, it keeps the target in focus.

For robot applications accuracy measurements are also important to ensure the robot's safety and unproblematic navigation. In case of incorrect, possibly extremely large motion vectors, errors in the flow field can lead to incorrect navigation commands resulting in accidents, failures or uncontrolled behavior of the robot. Hence, confidence measures are important in this field.

In pedestrian and vehicle tracking optical flow is often employed as well. One example is traffic monitoring in aerial video sequences, where low contrast and occlusions are difficult to handle [76]. Another example is pedestrian detection, e.g. [48], where principal component analysis and a boosting classifier are combined to distinguish between pedestrian motion and that of other objects such as cars. For applications as important as pedestrian detection accuracy measurements are indispensable to ensure the safety of people. Therefore, this thesis is dedicated to the analysis and classification of known confidence measures, to the proposition and evaluation of new confidence measures for optical flow fields and to the automatic reconstruction of erroneous measurements by means of inpainting methods yielding dense, accurate flow fields.

## 1.2 Thesis Outline

After introducing the topic of optical flow and confidence estimation as well as motion restoration and its applications I give an overview of the remaining chapters of this thesis. As some of the methods used in this thesis require some mathematical background knowledge, Chapter 2 contains the necessary preliminaries. For the estimation of optical flow fields many approaches have been proposed since the 1980s. The ones important for this thesis are shortly outlined in Chapter 3. I distinguish between local optical flow methods, which are based on locally restricted image regions, and global methods, which solve the optical flow problem for the whole image sequence within one functional.

In order to evaluate confidence measures, which predict the error of computed flow fields, I first introduce and analyze common error measurements for optical flow fields in chapter 4. Such error measurements are of high importance for the understanding of strengths and weaknesses of optical flow algorithms as well as for scientific and industrial applications. However, their evaluation has been limited to mostly the indication of the average angular error and its variance for a small number of highly artificial test sequences. Hence, I also propose a new evaluation method, which comprehends the assessment of motion estimators as a sampling from a joint probability distribution, namely the joint distribution of the true and the estimated flow, as well as local gray value neighborhoods. Marginals and conditionals of this distribution allow for a detailed assessment of optical flow algorithms.

After introducing common error measurements I come to the analysis of known confidence measures. I find that an important distinction has to be made between measures, which only judge the complexity of the image sequence to obtain information on the accuracy of the flow field, and “real” confidence measures. Since the former measures do not even consider the flow vectors at all, they, in fact, are insufficient for the task of reliability estimation. Such measures will be denoted by “situation measures”, since they only analyze the complexity of the image sequence. In chapter 5 I explain and classify these measures according to the intrinsic dimension of the image sequence they examine. I show that these measures can be successfully applied to the recognition of aperture problems, homogeneous regions and occlusions as well as to detect locations, where the flow vector can be computed reliably.

In chapter 6 I employ the notion of intrinsic dimensionality again to evaluate the accuracy of optical flow fields. However, I do not compute the intrinsic dimension of the image sequence. Instead, since in fact every optical flow method can be formulated as some kind of energy minimization task, I propose to analyze the intrinsic dimension of the energy surface produced by small variations of the computed flow vector. I formulate surface measures, which are able to identify unreliable parameter vectors as well as outliers, and can be used as situation or confidence measures. In this way, sparse, but reliable motion fields with lower errors can be obtained.

In chapter 7 I suggest learned motion models as another option to evaluate the accuracy

of an arbitrarily computed optical flow field. These models are obtained from learning algorithms applied to typical motion fields, e.g. ground truth flow fields, computed or synthetic fields. The model then contains information on common flow vector constellations, which can be used to examine computed flow fields. I propose two different kinds of motion models, one based on linear subspace projections and the other based on a statistical hypothesis test. The resulting confidence measures are statistically motivated, generally applicable independent of the flow computation method and obtain highly accurate results. They can be extended to a nonlinear version and to handle non-dense flow fields.

As confidence measures are based on some kind of knowledge on what a correct flow field should be like, most confidence measures already incorporate the basic idea or constraints for an optical flow computation method. Both concepts, the optical flow algorithms and confidence measures for their evaluation, are, therefore, closely related. Hence, starting out from the confidence measure proposed in Chapter 7, a new optical flow computation method is developed in Chapter 8. It is simple to implement, can easily be parallelized and yields highly accurate results.

Finally, confidence measures can be used to improve optical flow fields. Depending on the estimated accuracy of a given flow vector it can be removed from the flow field and reconstructed from its surrounding neighborhood in a subsequent step. Methods for the restoration of optical flow fields are introduced in Chapter 9. Here, the transfer of known inpainting methods to the reconstruction of vector fields is discussed. To further increase the quality of the restored field, I propose to include the image sequence information in the restoration process. The resulting functional directly combines motion and image information and allows to control the impact of image edges on the motion field reconstruction. In fact, in case of jumps of the motion field, where the jump set coincides with an edge set of the underlying image intensity, an anisotropic TV-type functional acts as a prior in the inpainting model.

To conclude, I combine the most effective confidence measure proposed in Chapter 7 and the image guided motion inpainting approach proposed in Chapter 9 to automatically restore optical flow fields.

### 1.3 Notation

As before, let  $I : D \rightarrow \mathbb{R}$  denote the spatio-temporal image sequence and  $\mathbf{u} : D \rightarrow \mathbb{R}^d$  the optical flow field. In case the ground truth flow is known it will be denoted by  $\mathbf{g} : D \rightarrow \mathbb{R}^d$ .

Furthermore, let  $\|.\|_{l_2}$  denote the  $l_2$ -norm as the square root of the sum of the squared vector/matrix components.  $\mathbf{x}^T$  stands for the transpose of the vector  $\mathbf{x}$ . The gradient is indicated by  $\nabla_z$ , where  $z \subset \{x, y, t\}$  indicates its direction.  $\det(A)$  refers to the determinant of the matrix  $A$ ,  $\text{trace}(A)$  denotes the trace of the matrix  $A$ .

Sometimes vectors are augmented by an additional dimension, which is set to 1. To simplify the notation in these cases, let for any given vector  $\mathbf{v} \in \mathbb{R}^n$

$$\tilde{\mathbf{v}} = (\mathbf{v}_1, \dots, \mathbf{v}_n, 1)^T \in \mathbb{R}^{n+1}. \quad (1.4)$$

Sometimes I need to indicate the image sequence  $I$  warped according to a flow field  $\mathbf{u}$ . Since the flow field is in general not integer numbered, I use linear interpolation. The warped image sequence is indicated by  $I_w$ :

$$I_w(x, y, t) = I(x + \mathbf{u}_1, y + \mathbf{u}_2, t + 1). \quad (1.5)$$

Let, furthermore,  $\mathbb{R}$  denote the set of real numbers,  $\mathbb{N}$  the set of natural numbers and  $\mathbb{N}_n = \{1, \dots, n\}$  the first  $n$  natural numbers.

# Chapter 2

---

## Mathematical Preliminaries

### 2.1 The Calculus of Variations

Let  $L$  be a real-valued mapping depending on five variables, which is twice continuously differentiable. The integral

$$\Phi(u) = \int_0^l \int_0^l L(x, y, u(x, y), u_x(x, y), u_y(x, y)) dy dx \quad (2.1)$$

is to be minimized over the set of all continuously differentiable mappings  $u : \mathbb{R}^2 \rightarrow \mathbb{R}$  fulfilling the boundary value conditions

$$\begin{aligned} u(0, y) &= 0, \quad u(l, y) = 0 \quad \forall y \in \mathbb{R} \\ u(x, 0) &= 0, \quad u(x, l) = 0 \quad \forall x \in \mathbb{R}. \end{aligned} \quad (2.2)$$

The mapping  $u \rightarrow \Phi(u)$  is called functional, and the minimization of such functionals is handled by the calculus of variations.

We assume that there exists a mapping  $u$ , which minimizes the functional  $\Phi(u)$ . Let  $v : \mathbb{R}^2 \rightarrow \mathbb{R}$  be an arbitrary, continuously differentiable mapping with boundary values equivalent to (2.2). Then the mapping

$$\epsilon \rightarrow \Phi(u + \epsilon v) = \int_0^l \int_0^l L(x, y, u + \epsilon v, u_x + \epsilon v_x, u_y + \epsilon v_y) dy dx \quad (2.3)$$

is minimized for  $\epsilon = 0$ . As this function is continuously differentiable, the derivative can be computed within the integral. Using partial integration and the conditions stated in

(2.2) we obtain <sup>1</sup>

$$\begin{aligned}
 & \left[ \frac{\partial}{\partial \epsilon} \Phi(u + \epsilon v) \right]_{\epsilon=0} = 0 \Leftrightarrow \\
 & \int_0^l \int_0^l \underbrace{\frac{\partial L}{\partial x} \frac{\partial x}{\partial \epsilon}}_{=0} + \underbrace{\frac{\partial L}{\partial y} \frac{\partial y}{\partial \epsilon}}_{=0} + \frac{\partial L}{\partial u} \frac{\partial u}{\partial \epsilon} + \frac{\partial L}{\partial u_x} \frac{\partial u_x}{\partial \epsilon} + \frac{\partial L}{\partial u_y} \frac{\partial u_y}{\partial \epsilon} dy dx = 0 \Leftrightarrow \\
 & \int_0^l \int_0^l \frac{\partial L}{\partial u} v + \frac{\partial L}{\partial u_x} v_x + \frac{\partial L}{\partial u_y} v_y dy dx = 0 \Leftrightarrow \\
 & \int_0^l \int_0^l \frac{\partial L}{\partial u} v - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} v - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} v dy dx + \underbrace{\int_0^l \frac{\partial L}{\partial u_x} v \Big|_0^l}_{=0} dy + \underbrace{\int_0^l \frac{\partial L}{\partial u_y} v \Big|_0^l}_{=0} dx = 0 \Leftrightarrow \\
 & \int_0^l \int_0^l \frac{\partial L}{\partial u} v - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} v - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} v dy dx = 0 \Leftrightarrow \\
 & \int_0^l \int_0^l \left( \frac{\partial L}{\partial u} - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} \right) v = 0 . \tag{2.4}
 \end{aligned}$$

**Theorem 1.** Let  $f$  be a continuous, real-valued mapping over the interval  $[0, l] \times [0, l]$ , and let

$$\int_0^l \int_0^l f(x, y) v(x, y) dy dx = 0 \tag{2.5}$$

for every continuously differentiable, real-valued mapping  $v$  fulfilling the boundary conditions equivalent to (2.2). Then it follows that  $f = 0$  over  $[0, l] \times [0, l]$ .

*Proof.* Assume, on the contrary, that  $f$  does not vanish over  $[0, l] \times [0, l]$ . Since  $f$  is continuous, there exists a point  $(x_0, y_0) \in [0, l] \times [0, l]$  and an  $\epsilon > 0$  such that  $f(x, y) \neq 0$  for  $(x, y) \in [x_0 - \epsilon, x_0 + \epsilon] \times [y_0 - \epsilon, y_0 + \epsilon]$ . Let first  $f(x, y) > 0$ . We now choose

$$v(x, y) = 
 \begin{cases} 
 [2\epsilon^2 - (x - x_0)^2 - (y - y_0)^2]^2, & (x, y) \in [x_0 - \epsilon, x_0 + \epsilon] \times [y_0 - \epsilon, y_0 + \epsilon] \\
 0, & \text{otherwise.}
 \end{cases}$$

Then  $v$  is continuously differentiable and fulfills the assumed boundary conditions. It follows

$$f(x, y) v(x, y) \begin{cases} > 0, & (x, y) \in [x_0 - \epsilon, x_0 + \epsilon] \times [y_0 - \epsilon, y_0 + \epsilon] \\ = 0, & \text{otherwise.} \end{cases} \tag{2.6}$$

---

<sup>1</sup>In the following,  $\frac{\partial L}{\partial x}$  denotes the derivative with respect to the first variable of  $L$ ,  $\frac{\partial L}{\partial u}$  the derivative with respect to the third variable of  $L$  and so on. It does not denote the derivative with respect to the function  $u$ .

Hence, we would obtain

$$\int_0^l \int_0^l f(x, y)v(x, y) dy dx = \int_{x_0-\epsilon}^{x_0+\epsilon} \int_{y_0-\epsilon}^{y_0+\epsilon} f(x, y)v(x, y) dy dx > 0, \quad (2.7)$$

which contradicts the assumption. The case  $f(x_0, y_0) < 0$  is handled in the same way. Hence, it follows that  $f(x, y) = 0$  for  $x \in (0, l) \times (0, l)$  and thus for all  $x \in [0, l] \times [0, l]$ . This concludes the proof.  $\square$

Based on this theorem we can now conclude from equation (2.4)

$$\int_0^l \int_0^l \left( \frac{\partial L}{\partial u} - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} \right) v = 0 \quad (2.8)$$

that

$$\frac{\partial L}{\partial u} - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} = 0. \quad (2.9)$$

This partial differential equation is also called the Euler-Lagrange equation. Hence, the mapping  $u$ , which minimizes the functional in (2.1), can be obtained by solving the Euler-Lagrange equation in (2.9).

## 2.2 Hypothesis Testing

Hypothesis tests are used to judge if a given sample realization can stem from a hypothetical distribution (also called null hypothesis). Let  $(\mathbb{H}, \mathcal{H}, \mathcal{W})$  be a statistical experiment, where  $\mathbb{H}$  refers to the  $n$ -dimensional sample space,  $\mathcal{H}$  to the  $\sigma$ -algebra over the sample space and  $\mathcal{W}$  to a set of probability measures. Let, furthermore,  $\Gamma$  refer to a parameter space, which is partitioned into two sets  $\Gamma_1$  and  $\Gamma_2$ . This partitioning of the parameter space corresponds to a partitioning of the set of probability measures  $\mathcal{W}$  into  $\mathcal{W}_1$  and  $\mathcal{W}_2$ , since each parameter in  $\Gamma$  defines one probability measure in  $\mathcal{W}$ . The quintuple  $(\mathbb{H}, \mathcal{H}, \mathcal{W}, \mathcal{W}_1, \mathcal{W}_2)$  is called testing experiment.

$\mathcal{W}_1$  is called the hypothesis,  $\mathcal{W}_2$  the alternative. A hypothesis test decides for each possible realization of a sample  $X = (X_1, \dots, X_n)$  if the distribution of  $X$  can be described by a distribution in  $\mathcal{W}_1$ .

Let  $(\mathbb{H}, \mathcal{H}, \mathcal{W}, \mathcal{W}_1, \mathcal{W}_2)$  be a testing experiment, and let  $\mathcal{B}$  denote the Borel  $\sigma$ -algebra. Then every  $(\mathcal{H} - [0, 1] \cap \mathcal{B})$ -measurable function  $\phi : \mathbb{H} \rightarrow [0, 1]$  defines a hypothesis test. Here,  $\phi(x) = 0$  means that the hypothesis is accepted, whereas  $\phi(x) = 1$  means that the hypothesis is rejected.

$$A := \phi^{-1}(0) = \{x \in \mathbb{H} | \phi(x) = 0\} \quad (2.10)$$

is called acceptance region of the test. If for a sample realization we have  $x \in A$  then the hypothesis is not rejected.

$$A^c := \phi^{-1}(1) = \{x \in \mathbb{H} | \phi(x) = 1\} \quad (2.11)$$

is called critical or rejection region of the test. If for a sample realization we have  $x \in A^c$  then the hypothesis is rejected. The choice of acceptance and rejection region is crucial for the quality of the test.

### 2.2.1 The Quality of a Hypothesis Test

The quality of a hypothesis test is measured based on the probability of an incorrect decision.

#### Type 1 and Type 2 Errors

There are two types of incorrect decisions in a hypothesis test

- Type 1 error: the hypothesis is rejected even though it is correct,
- Type 2 error: the hypothesis is not rejected even though it is incorrect.

Let there be a test problem with  $\mathcal{W}_1 = \{P_1\}$  denoting the hypothesis and  $\mathcal{W}_2 = \{P_2\}$  the alternative. Let  $A$  refer to the acceptance region and  $A^c$  to the rejection region of a test  $\phi$ . Then the probability of a type 1 error is  $P_1(A^c)$  and that of a type 2 error  $P_2(A)$ . Figure 2.1 demonstrates this relation for a simple example.

#### Quality Function and Power of a Test

Quantitative statements on the quality of a test can be obtained by means of the quality function. Let  $(\mathbb{H}, \mathcal{H}, \mathcal{W}, \mathcal{W}_1, \mathcal{W}_2)$  be a testing experiment and  $\phi : \mathbb{H} \rightarrow \{0, 1\}$  a hypothesis test. Then the function

$$\beta_\phi : \Gamma \rightarrow [0, 1] \quad (2.12)$$

$$\beta_\phi(\gamma) = E_\gamma(\phi) = \int \phi dP_\gamma \quad (2.13)$$

is called quality function of  $\phi$ . For each parameter  $\gamma$  of the parameter space  $\Gamma$  this function computes the expected value of the test, that is the probability for the rejection of the hypothesis. Hence, it is important to design the test  $\phi$  such that  $\beta_\phi(\gamma)$  is minimal for  $\gamma \in \Gamma_1$  and maximal for  $\gamma \in \Gamma_2$ . The probability for a type 1 error can be determined by means of the restriction  $\beta_\phi|_{\Gamma_1}$ .

To control the type 1 error, a significance level  $\alpha$  is chosen for hypothesis tests. The

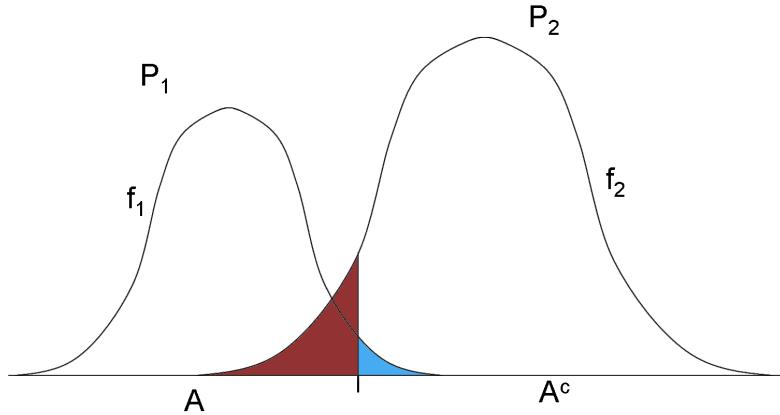


Figure 2.1: Graphical representation of the probability of a type 1 error ( $P_1(A^c)$ , blue) and a type 2 error ( $P_2(A)$ , red),  $\{P_1\}$  refers to the hypothesis with corresponding probability density  $f_1$ ,  $\{P_2\}$  to the alternative with corresponding probability density  $f_2$ .

test  $\phi$  is designed in such a way that the probability for a type 1 error is limited by  $\alpha$  in case the hypothesis is true, that means

$$P_\gamma\{x \in \mathbb{H}|\phi(x) = 1\} = \beta_\phi(\gamma) \leq \alpha, \gamma \in \Gamma_1 . \quad (2.14)$$

Tests fulfilling (2.14) are called  $\alpha$ -tests or tests with significance level  $\alpha$ . An important quality criterion, which can also be computed from the quality function  $\beta_\phi$ , is the power of a test, which is defined as the restriction  $\beta_\phi|_{\Gamma_2}$ . It indicates the probability that the hypothesis is rejected in case that the distribution underlying the data does not belong to the hypothesis, that is the absence of a type 2 error.

## 2.2.2 p-Values

**Definition 1.** Let  $(\mathbb{H}, \mathcal{H}, \mathcal{W}, \mathcal{W}_1, \mathcal{W}_2)$  refer to a testing experiment. For  $\alpha \in (0, 1)$  let  $\phi_\alpha : \mathbb{H} \rightarrow \{0, 1\}$  be a deterministic test with significance level  $\alpha$ , that means

$$\int_{\mathbb{H}} \phi_\alpha dP_\gamma \leq \alpha, P_\gamma \in \mathcal{W}_1 . \quad (2.15)$$

For  $\alpha \in (0, 1)$  let  $K_\alpha := \{x \in \mathbb{H}|\phi_\alpha(x) = 1\}$  indicate the rejection region of  $\phi_\alpha$ . Furthermore, let

$$0 < \alpha_1 < \alpha_2 < 1 \Rightarrow K_{\alpha_1} \subset K_{\alpha_2} . \quad (2.16)$$

Then the p-value function  $\Pi : \mathbb{H} \rightarrow [0, 1]$  belonging to the family of tests  $(\phi_\alpha)_{\alpha \in (0, 1)}$  is defined by

$$\Pi(x) = \inf \{\alpha | x \in K_\alpha\} . \quad (2.17)$$

Let  $x$  be a sample realization, then the value  $\Pi(x)$  is called the p-value of the sample  $x$ . It corresponds to the minimum significance level, for which the sample  $x$  is rejected.

### 2.3 Best Linear Unbiased Estimators

**Theorem 2.** Let  $\mathbf{x}$  be an  $n \times 1$  parameter vector that is to be estimated from an  $m \times 1$  vector  $\mathbf{y}$  of observations. Let, furthermore,  $\mu_{\mathbf{x}}$  and  $\mu_{\mathbf{y}}$  denote the expected values of  $\mathbf{x}$  and  $\mathbf{y}$ , and let  $\mathbf{C}_{\mathbf{xx}}$ ,  $\mathbf{C}_{\mathbf{xy}}$ ,  $\mathbf{C}_{\mathbf{yx}}$  and  $\mathbf{C}_{\mathbf{yy}}$  denote the corresponding covariance matrices. Then the best linear unbiased estimator (BLUE) for  $\mathbf{x}$  given the observations  $\mathbf{y}$  is

$$\hat{\mathbf{x}} = \mu_{\mathbf{x}} + \mathbf{C}_{\mathbf{xy}} \mathbf{C}_{\mathbf{yy}}^{-1} (\mathbf{y} - \mu_{\mathbf{y}}), \quad (2.18)$$

$$\text{Var}(\hat{\mathbf{x}}) = \mathbf{C}_{\mathbf{xy}} \mathbf{C}_{\mathbf{yy}}^{-1} \mathbf{C}_{\mathbf{yx}}, \quad (2.19)$$

$$\text{Var}(\mathbf{x} - \hat{\mathbf{x}}) = \mathbf{C}_{\mathbf{xx}} - \mathbf{C}_{\mathbf{xy}} \mathbf{C}_{\mathbf{yy}}^{-1} \mathbf{C}_{\mathbf{yx}}. \quad (2.20)$$

*Proof.* To prove this statement we introduce the following four theorems. Their proofs can be found in [61]. Let  $\mathbf{x}$  and  $\mathbf{y}$  denote two random variables. Then

**Theorem 3.**  $\text{Var}(A\mathbf{x} + \mathbf{b}) = A \text{Var}(\mathbf{x}) A^T$

**Theorem 4.**  $\text{Cov}(A\mathbf{x} + a, B\mathbf{y} + b) = A \text{Cov}(\mathbf{x}, \mathbf{y}) B^T$

**Theorem 5.**  $\text{Cov}(\mathbf{x} - \mathbf{y}, \mathbf{z}) = \text{Cov}(\mathbf{x}, \mathbf{z}) - \text{Cov}(\mathbf{y}, \mathbf{z})$

**Theorem 6.**  $\text{Var}(\mathbf{x} + \mathbf{y}) = \text{Var}(\mathbf{x}) + \text{Cov}(\mathbf{x}, \mathbf{y}) + \text{Cov}(\mathbf{y}, \mathbf{x}) + \text{Var}(\mathbf{y})$

Let  $\mathbf{b} \in \mathbb{R}^n$ . Then we want to estimate (as customary in most textbooks) an arbitrary linear function  $\widehat{\mathbf{b}^T \mathbf{x}}$  such that the following assumptions hold:

**Assumption 1.**  $\widehat{\mathbf{b}^T \mathbf{x}}$  is linearly predictable from  $\mathbf{y}$

$$\widehat{\mathbf{b}^T \mathbf{x}} = \mathbf{c}^T \mathbf{y} + \mathbf{d}, \quad \mathbf{c}, \mathbf{d} \in \mathbb{R}^m.$$

**Assumption 2.**  $\widehat{\mathbf{b}^T \mathbf{x}}$  is unbiased

$$E(\mathbf{c}^T \mathbf{y} + \mathbf{d}) = E(\widehat{\mathbf{b}^T \mathbf{x}}), \text{ which equals the condition } \mathbf{c}^T \mu_{\mathbf{y}} + \mathbf{d} - \mathbf{b}^T \mu_{\mathbf{x}} = 0.$$

**Assumption 3.** The squared error of the estimator is to be minimized

$$\text{Var}(\widehat{\mathbf{b}^T \mathbf{x}} - \mathbf{b}^T \mathbf{x}) \rightarrow \min.$$

From assumption 3 and 1 together with theorem 3 and 4 it follows that

$$\begin{aligned}
 \text{Var}(\widehat{\mathbf{b}}^T \mathbf{x} - \mathbf{b}^T \mathbf{x}) &= \\
 \text{Var}(\mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}) &= \\
 E((\mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x} - E(\mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}))^2) &= \\
 E(((\mathbf{c}^T \mathbf{y} - E(\mathbf{c}^T \mathbf{y})) - (\mathbf{b}^T \mathbf{x} - E(\mathbf{b}^T \mathbf{x})))^2) &= \\
 E((\mathbf{c}^T \mathbf{y} - E(\mathbf{c}^T \mathbf{y}))^2 - 2(\mathbf{c}^T \mathbf{y} - E(\mathbf{c}^T \mathbf{y}))(\mathbf{b}^T \mathbf{x} - E(\mathbf{b}^T \mathbf{x})) + (\mathbf{b}^T \mathbf{x} - E(\mathbf{b}^T \mathbf{x}))^2) &= \\
 \text{Var}(\mathbf{c}^T \mathbf{y}) - 2 \text{Cov}(\mathbf{c}^T \mathbf{y}, \mathbf{b}^T \mathbf{x}) + \text{Var}(\mathbf{b}^T \mathbf{x}) &= \\
 \mathbf{c}^T \mathbf{C}_{yy} \mathbf{c} - 2\mathbf{c}^T \mathbf{C}_{yx} \mathbf{b} + \mathbf{b}^T \mathbf{C}_{xx} \mathbf{b} &\rightarrow \min.
 \end{aligned}$$

To minimize this expression with regard to assumption 2 we use the Lagrange multiplier  $-2\lambda$

$$w(\mathbf{c}, \mathbf{d}) := \mathbf{c}^T \mathbf{C}_{yy} \mathbf{c} - 2\mathbf{c}^T \mathbf{C}_{yx} \mathbf{b} + \mathbf{b}^T \mathbf{C}_{xx} \mathbf{b} - 2\lambda(\mathbf{c}^T \mu_y + \mathbf{d} - \mathbf{b}^T \mu_x) \quad (2.21)$$

and solve the system of equations

$$\begin{aligned}
 \frac{\partial w(\mathbf{c}, \mathbf{d})}{\partial \mathbf{d}} &= -2\lambda = 0, \\
 \frac{\partial w(\mathbf{c}, \mathbf{d})}{\partial \mathbf{c}} &= 2\mathbf{C}_{yy} \mathbf{c} - 2\mathbf{C}_{yx} \mathbf{b} - 2\lambda \mu_y = 0, \\
 \frac{\partial w(\mathbf{c}, \mathbf{d})}{\partial \lambda} &= \mathbf{c}^T \mu_y + \mathbf{d} - \mathbf{b}^T \mu_x = 0.
 \end{aligned}$$

It follows that

$$\lambda = 0 \quad (2.22)$$

$$\mathbf{c} = \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b} \quad (2.23)$$

$$\mathbf{d} = \mathbf{b}^T \mu_x - \mathbf{c}^T \mu_y .$$

Using these results and theorem 3, the variance of the estimation error comes down to

$$\begin{aligned}
 \text{Var}(\mathbf{b}^T \mathbf{x} - \widehat{\mathbf{b}}^T \mathbf{x}) &= \quad (2.24) \\
 \text{Var}(\mathbf{b}^T \mathbf{x} - \mathbf{c}^T \mathbf{y} - \mathbf{d}) &= \text{Var}(\mathbf{b}^T \mathbf{x} - \mathbf{c}^T \mathbf{y} - \mathbf{b}^T \mu_x + \mathbf{c}^T \mu_y) = \\
 \mathbf{b}^T \mathbf{C}_{xx} \mathbf{b} - \mathbf{c}^T \mathbf{C}_{yy} \mathbf{c} &= \mathbf{b}^T \mathbf{C}_{xx} \mathbf{b} - (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b})^T \mathbf{C}_{yy} (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b}) = \\
 \mathbf{b}^T \mathbf{C}_{xx} \mathbf{b} - \mathbf{b}^T \mathbf{C}_{xy} \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b} &= \mathbf{b}^T (\mathbf{C}_{xx} - \mathbf{C}_{xy} \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx}) \mathbf{b}.
 \end{aligned}$$

We now show that the linear estimator defined by (2.22) and (2.23) has minimum error. To this end, let  $\widehat{\mathbf{z}} = \mathbf{e}^T \mathbf{y} + \mathbf{f}$ , for  $\mathbf{e}, \mathbf{f} \in \mathbb{R}^m$  be any other linear estimator. Using theorem 6

and the fact that some expressions equal out to 0, we obtain for the variance

$$\begin{aligned} \text{Var}(\widehat{\mathbf{z}} - \mathbf{b}^T \mathbf{x}) &= \\ \text{Var}(\mathbf{e}^T \mathbf{y} + \mathbf{f} - \mathbf{b}^T \mathbf{x}) &= \text{Var}(\mathbf{e}^T \mathbf{y} + \mathbf{f} - \mathbf{c}^T \mathbf{y} - \mathbf{d} + \mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}) = \\ \text{Var}((\mathbf{e} - \mathbf{c})^T \mathbf{y} + \mathbf{f} - \mathbf{d} + \mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}) &= \\ \text{Var}((\mathbf{e} - \mathbf{c})^T \mathbf{y} + \mathbf{f} - \mathbf{d}) + \text{Var}(\mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}). \end{aligned} \quad (2.25)$$

Bearing in mind that the variance is non-negative we, finally, obtain

$$\text{Var}(\widehat{\mathbf{z}} - \mathbf{b}^T \mathbf{x}) \geq \text{Var}(\mathbf{c}^T \mathbf{y} + \mathbf{d} - \mathbf{b}^T \mathbf{x}) = \text{Var}(\widehat{\mathbf{b}^T \mathbf{x}} - \mathbf{b}^T \mathbf{x}). \quad (2.26)$$

Hence, the error of the estimator  $\widehat{\mathbf{b}^T \mathbf{x}}$  is minimal.

We, thus, come to the following conclusions for the BLUE:

$$\begin{aligned} \widehat{\mathbf{b}^T \mathbf{x}} &= \mathbf{c}^T \mathbf{y} + \mathbf{d} = (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b})^T \mathbf{y} + \mathbf{b}^T \mu_x - (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b})^T \mu_y = \\ &= (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b})^T (\mathbf{y} - \mu_y) + \mathbf{b}^T \mu_x = \mathbf{b}^T (\mu_x + \mathbf{C}_{xy} \mathbf{C}_{yy}^{-1} (\mathbf{y} - \mu_y)) . \end{aligned} \quad (2.27)$$

Due to (2.24) the error of the estimator results in

$$\text{Var}(\mathbf{b}^T \mathbf{x} - \widehat{\mathbf{b}^T \mathbf{x}}) = \mathbf{b}^T (\mathbf{C}_{xx} - \mathbf{C}_{xy} \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx}) \mathbf{b} \quad (2.28)$$

and is minimal due to (2.26). According to theorem 3 and the results in (2.22) and (2.23) its variance is

$$\begin{aligned} \text{Var}(\widehat{\mathbf{b}^T \mathbf{x}}) &= \text{Var}(\mathbf{c}^T \mathbf{y} + \mathbf{d}) = \mathbf{c}^T \text{Var}(\mathbf{y}) \mathbf{c} = \\ &= (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b})^T \mathbf{C}_{yy} (\mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b}) = \\ &= \mathbf{b}^T \mathbf{C}_{xy} \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \mathbf{b}. \end{aligned} \quad (2.29)$$

Finally setting  $\mathbf{b}^T = (1, 0, \dots, 0), \dots, (0, \dots, 0, 1)$  in (2.27), (2.28) and (2.29) we obtain the original statements in theorem 2. The formulas for the estimator  $\widehat{\mathbf{x}}$  and  $\text{Var}(\mathbf{x} - \widehat{\mathbf{x}})$  can also be obtained from the normal distribution, which is conditioned on  $\mathbf{y}$ . Here  $\widehat{\mathbf{x}}$  and  $\text{Var}(\mathbf{x} - \widehat{\mathbf{x}})$  correspond to the expectation and covariance of the conditional normal distribution of  $\mathbf{x}$  given the observation vector  $\mathbf{y}$ .

□

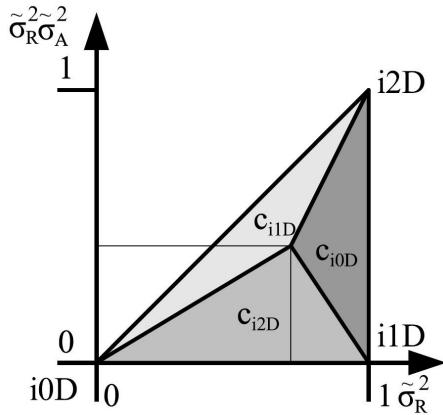


Figure 2.2: Triangular topology of the intrinsic dimensionality in barycentric coordinates, taken from [38].

## 2.4 Intrinsic Dimensions

According to [17] the notion “intrinsic dimension” is defined as follows: “a data set in  $n$  dimensions is said to have an intrinsic dimensionality equal to  $d$  if the data lies entirely within a  $d$ -dimensional subspace”. It has first been applied to image processing by Zetsche and Barth in [101] in order to distinguish between edge-like and corner-like structures in an image. Such information can be used to identify reliable locations for optical flow computation, tracking and registration, e.g. corners in an image sequence. An equivalent definition of “intrinsic dimension” in image patches is based on its spectrum assigning the identifier

- i0d, if the spectrum consists of a single point, which corresponds to a homogeneous image patch
- i1d, if the spectrum is a line through the origin, which corresponds to an edge in the image patch
- i2d otherwise, which corresponds to highly textured regions.

The intrinsic dimension of a given dataset can be computed as the rank of the structure tensor [16]. As, of course, the eigenvalues of the structure tensor are never exactly 0, thresholds have to be applied to obtain estimates of discrete intrinsic dimensions. Hence, since no situation exclusively belongs to only one of the intrinsic dimensions, Felsberg et al. [38] gave a continuous formulation of intrinsic dimensions smaller and equal to two and showed that the underlying topology of the intrinsic dimension space corresponds to a triangle (see Figure 2.2). Their formulation has been used in [60] to analyze errors of local optical flow computation methods based on the intrinsic dimensionality of the

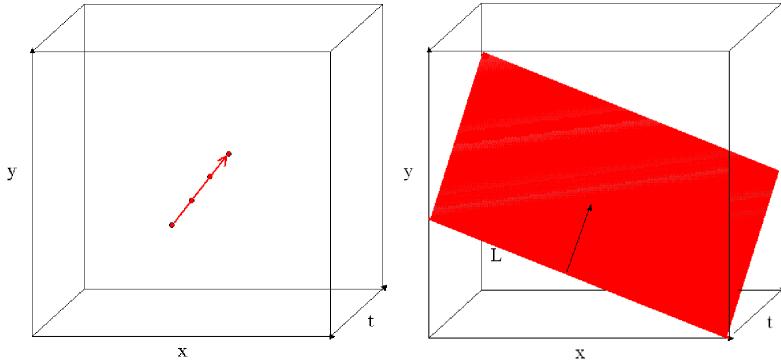


Figure 2.3: Left: example for i2d: a point moving in 3d-space, the dimension of the subspace with constant intensity along the line is 1, thus yielding intrinsic dimension 2. Right: example for i1d: a moving edge leading to a 2d-plane of constant intensity, thus yielding intrinsic dimension 1.

underlying image.

In [11, 13] Barth has introduced the intrinsic dimensionality of three-dimensional image sequences and applied it to motion estimation, especially for multiple and transparent motions. Provided that the assumption of constant brightness over time holds, motion of a single point corresponds to a line of constant brightness in the image sequence volume. Intuitively speaking, the notion “intrinsic dimension” refers to the dimension of the examined image region (here three) minus the number of dimensions with the same intensity as the current pixel. Thus, the intrinsic dimension of locations in image sequences, where motion is present, is lower or equal to two. In case of intrinsic dimension three the brightness constancy assumption is violated which can be due to e.g. noise, occlusions or transparent structures, since the trajectory of constant intensity in the temporal dimension is intersected. If unambiguous movement (e.g. of a corner) is present, a unique trajectory of the current pixel in temporal direction exists, which corresponds to intrinsic dimension two. If we have an aperture problem, there is one additional spatial direction with constant intensity leading to intrinsic dimension one. A homogeneous region contains the same intensity in two spatial and the temporal dimension leading to intrinsic dimension zero. If no consistent movement exists, we have intrinsic dimension three. Figure 2.3 demonstrates this concept for intrinsic dimension one and two.

Occlusions play a special role here. If a motion vector with intrinsic dimension between zero and two is occluded, the temporal direction no longer contains constant intensity values. Consequently the vector gains one intrinsic dimension. Thus, estimators for the intrinsic dimensionality cannot distinguish between a situation of a certain intrinsic dimension and an occluded situation of a lower intrinsic dimension. This leads to problems

for confidence estimators, which rely on intrinsic dimension estimates.

## 2.5 Principal Component Analysis

### 2.5.1 Origin and Objectives

The origins of Principal Component Analysis (PCA) reach back to the 1930s, when PCA was originally used in psychology to study intelligence. It is also called "Karhunen-Loeve transform" and refers to a mathematical way of expressing an n-dimensional dataset in a new coordinate system that shows the properties of the data samples most clearly along the coordinate axes.

Today PCA is used in a wide range of applications from computer science to psychology, biology, botanics and medicine, wherever the reduction of dimensionality is needed. There are also applications, where the dimensionality of the data is unknown and PCA is used to define as small a number of principal axes as possible.

There are several goals of PCA:

- the reduction (or discovery) of the dimensionality of a dataset with many interrelated variables,
- the conservation of as much of the variation in the dataset as possible despite the compression,
- the reduction of noise and redundancy in the dataset,
- the emphasis of the variation in the dataset,
- the identification of new, underlying, explanatory variables.

In Figure 2.4 the original dataset in 2-dimensional space can be seen on the left within the original coordinate system. On the right the same dataset has been reexpressed using a "better" coordinate or basis system in order to achieve the goals of PCA. The variables have been decorrelated as much as possible. Most of the variance of the dataset can now be found along the axis  $y_1$ . To reduce the dimensionality of the data we could remove the axis  $y_2$  and would only lose as few information as possible, since most of the variance is conserved on the other axis.

To perform PCA a large sample dataset is needed. The question remains how to choose the new basis system for the representation of the original  $n$ -dimensional data samples in order to fulfill the goals stated above.

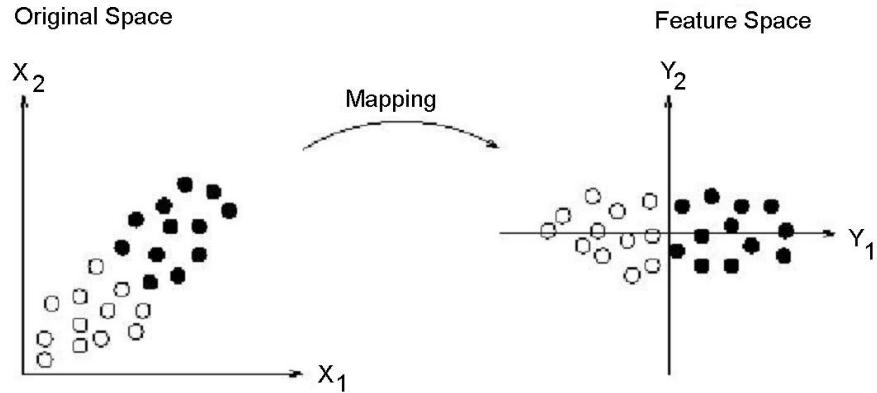


Figure 2.4: Principal Component Analysis for 2D-data.

### 2.5.2 Achieving the goals of PCA

The covariance of two random variables expresses the redundancy between these variables. A high value for an off-diagonal element of the covariance matrix indicates high correlation and, thus, redundancy between the corresponding variables. We show that the goals of PCA will be achieved by finding a way to diagonalize the covariance matrix. First, redundancy reduction is obtained, since the cross-covariances of all variables are zero after the diagonalization. So no linear redundancy is left in the dataset. Furthermore, information on the variation of the variables can be obtained from the main diagonal of the covariance matrix after diagonalization. The higher the variance of the variable the more information of the original dataset is contained in it and the more important is the variable. Noise reduction goes together with the goal of dimensionality reduction. Noise can usually be found in variables with low variance, whereas variables with higher variances represent important dynamics of the dataset. The latter are desirable components needed to discriminate between the different data samples, whereas the variables with lower variances, the noise, make the process of distinguishing between the data samples more difficult. These variables are removed during the process of dimensionality reduction. Hence, when reducing the dimensions of the dataset the noise is reduced at the same time. Thus, it is clear that all of the above stated goals of PCA can be obtained by the diagonalization of the covariance matrix.

### 2.5.3 Diagonalization of the Covariance Matrix using Eigenvector Decomposition

The assumption that the original basis and the desired principal components form orthonormal bases is necessary to find a rather simple solution for the diagonalization

problem using linear algebra.

Let  $X$  be the matrix containing the mean-adjusted data samples in its columns. Our goal is to find a matrix  $P$  where  $Y = P^T X$  in such a way, that the covariance matrix  $D = \frac{1}{n-1} Y Y^T$  of the resulting matrix  $Y$  is a diagonal matrix. Let  $\bar{M}$  refer to the row-wise mean vector of a given matrix  $M$ , and let  $C$  denote the covariance matrix of the original dataset matrix containing the samples in its columns. Then we can express the covariance matrix  $D$  after diagonalization in the following way

$$\begin{aligned} D &= \frac{1}{n-1} (Y - \bar{Y})(Y - \bar{Y})^T \\ &= \frac{1}{n-1} (P^T(X - \bar{X}))(P^T(X - \bar{X}))^T \\ &= P^T \underbrace{\frac{1}{n-1}(X - \bar{X})(X - \bar{X})^T}_{}_{:=C} P \\ &= P^T C P. \end{aligned}$$

So, the final redefinition of the problem is to find a matrix  $P$  such that  $P^T C P$  is diagonal.

The following steps are based on several theorems from linear algebra:

- Every matrix multiplied by its transposed is symmetric.
- Eigenvectors of every symmetric matrix are orthogonal.
- For every orthogonal matrix  $M$  the inverse of this matrix is its transpose, so  $M^{-1} = M^T$ .
- Consequently the transposed of every matrix  $S$  containing the normalized eigenvectors of a symmetric matrix  $M$  in its columns is equal to  $S^{-1}$ .

Hence, we know that every symmetric matrix  $M$  can be diagonalized by a matrix containing the eigenvectors of  $M$  in its columns. Therefore  $SMS^{-1} = SMS^T$  is a diagonal matrix. The covariance matrix  $C$  is symmetric.  $P$  must, therefore, contain the eigenvectors of  $C$  in its columns in order to diagonalize  $C$ . To find the eigenvectors of a symmetric matrix e.g. Givens rotations can be used. These eigenvectors of  $C$  are the new basis vectors, which we will use for the transformation of the data samples.  $D$  is the desired diagonal matrix, the new covariance matrix of the transformed dataset, containing the eigenvalues of  $C$  on the main diagonal.

Since the eigenvalues on the main diagonal correspond to the variances of each single variable, the eigenvalues corresponding to the principal components can be used to determine their importance. They describe the variance of the dataset along the corresponding principal component, the eigenvector.

In order to reduce the dimensionality of the data to a meaningful subspace, the axes representing the least information (the smallest variance) of the dataset can be removed. These are the eigenvectors with the smallest eigenvalues. We can select the number of eigenvectors containing the fraction  $\delta$  of the information of the original dataset by choosing  $k$  of the  $n$  eigenvectors sorted by decreasing eigenvalue  $\lambda_i$  such that

$$k = \min\{r \mid \frac{\sum_{i=1}^r \lambda_i}{\sum_{i=1}^n \lambda_i} \geq \delta\}. \quad (2.30)$$

#### 2.5.4 Data transformation

The following transformation will be performed to express the original dataset in terms of the new basis system. Let  $X$  be the matrix of the original dataset with each single data vector (or sample image) arranged in one of the columns. Let  $P$  be the transformation matrix containing the new basis vectors in its columns. Then the transformation

$$P^T(X - \bar{X}) = Y$$

expresses the original dataset in the new coordinate system. Thus,  $Y_i$  (the  $i$ -th column of  $Y$ ) is the projection of the  $i$ -th data sample onto the basis given in the columns of  $P$ . To retransform the data from the eigenspace to the original sample space the inverse transformation is used

$$X = PY + \bar{X}.$$

## 2.6 The Least Squares Method

Let there be  $n$  data points  $(x_i, y_i), i \in \mathbb{N}_n$ , forming an overdetermined system of equations with unknown parameters  $\alpha$  and  $\beta$

$$y_i = \alpha + \beta x_i, i \in \mathbb{N}_n. \quad (2.31)$$

To estimate  $\alpha$  and  $\beta$  we want to minimize the  $l_2$ -norm of the residual vector  $r$

$$F(\alpha, \beta) := \|r\|_2^2 = \sum_{i=1}^n r_i^2 = \sum_{i=1}^n (\alpha + \beta x_i - y_i)^2. \quad (2.32)$$

To obtain the minimum we have to solve the following system of equations

$$\frac{\partial F(\alpha, \beta)}{\partial \alpha} = \sum_{i=1}^n 2(\alpha + \beta x_i - y_i) = 0, \quad (2.33)$$

$$\frac{\partial F(\alpha, \beta)}{\partial \beta} = \sum_{i=1}^n 2(\alpha + \beta x_i - y_i)x_i = 0. \quad (2.34)$$

These equations are called normal equations and can be rewritten in the following way

$$\sum_{i=1}^n (\alpha + \beta x_i) = \sum_{i=1}^n y_i \quad (2.35)$$

$$\sum_{i=1}^n (\alpha + \beta x_i) x_i = \sum_{i=1}^n y_i x_i. \quad (2.36)$$

Equivalently, we can use matrices to rewrite these equations

$$\begin{pmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n y_i x_i \end{pmatrix}. \quad (2.37)$$

It is obvious that the matrix of the normal equations is real-valued and symmetric. Let

$$A := \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \in \mathbb{R}^{n \times 2}, \quad y := \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \in \mathbb{R}^n. \quad (2.38)$$

Then the system of normal equations in (2.37) can be written in the following way

$$A^T A \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = A^T y. \quad (2.39)$$

For the Hessian matrix  $H$  containing the second derivatives of  $F(\alpha, \beta)$  we obtain

$$H = \begin{pmatrix} \frac{\partial F(\alpha, \beta)}{\partial^2 \alpha} & \frac{\partial F(\alpha, \beta)}{\partial \alpha \partial \beta} \\ \frac{\partial F(\alpha, \beta)}{\partial \alpha \partial \beta} & \frac{\partial F(\alpha, \beta)}{\partial^2 \beta} \end{pmatrix} = \begin{pmatrix} 2n & 2 \sum_{i=1}^n x_i \\ 2 \sum_{i=1}^n x_i & 2 \sum_{i=1}^n x_i^2 \end{pmatrix} = 2A^T A. \quad (2.40)$$

Furthermore, we have  $\det(H) = \det(2A^T A) = 2^n \det(A^T A) \geq 0$ . And in case there is a pair  $i, j \in \mathbb{N}_n$  of indices with  $x_i \neq x_j$  the determinant of  $H$  is positive. Thus, in this case the solution of the normal equations given in (2.39) yields the minimum residual solution for the parameters  $\alpha$  and  $\beta$ .



# Chapter 3

---

## Optical Flow Estimation

Optical flow computation is usually based on the assumption that the brightness of a moving pixel remains constant over time. If  $\mathbf{x} : [0, T] \rightarrow \mathbb{R}^2$  describes the trajectory of a point of an object we can model a constant brightness intensity as  $I(\mathbf{x}(t), t) = \text{const}$ . A first order approximation yields the brightness constancy constraint equation (BCCE)

$$\frac{dI}{dt} = 0 \Leftrightarrow \mathbf{u} \cdot \nabla_{\mathbf{x}} I + \frac{\partial I}{\partial t} = 0 , \quad (3.1)$$

where  $\nabla$  is the gradient operator with respect to parameters given as indices.

### 3.1 Local Methods

In case of a local method, the optical flow is estimated for each pixel individually based on a specific flow model, which is only locally valid.

Therefore, local methods are usually simple to implement and fast as they can be easily parallelized. Furthermore, they have proven relatively robust to noise [56]. Yet, as only local image information is used for the computation, such methods often suffer from problems in regions, where none or only little image information is available, such as in homogeneous regions or in case of aperture problems. In contrast, global methods solve the optical flow problem by minimizing an energy term formulated on the whole image domain.

#### 3.1.1 The Lukas and Kanade Method

Lukas and Kanade [70] proposed to solve the optical flow problem by assuming brightness constancy and a constant flow within small neighborhoods. This leads to an overdeter-

mined system of brightness constancy equations for each pixel

$$\underbrace{\nabla_{\mathbf{x}} I}_{A} \mathbf{u} = \underbrace{-\nabla_t I}_{b}. \quad (3.2)$$

To solve this system of equations, the least squares method described in chapter 2 is used.

### 3.1.2 The Bigün Method

The problem with the method by Lukas and Kanade is that the least squares method assumes errors only in the right hand side of the equation system, in the observation vector, that is in the temporal image gradient. Since errors also occur in the spatial image gradient, it would be more appropriate to minimize the error of the true and the measured spatio-temporal image gradients. This approach was proposed by Bigün [16] in 1991.

Let  $D = [A, b]$  and  $\mathbf{r} = [\mathbf{u}, -1]^T$ . Then the idea of the total least squares method is to solve the following optimization problem:

$$\min \|D\mathbf{r}\|_2^2, \quad s.t. \quad \mathbf{r}^T \mathbf{r} = 1. \quad (3.3)$$

Using Lagrange multipliers

$$L(\mathbf{r}, \lambda) = \|D\mathbf{r}\|_2^2 + \lambda(1 - \mathbf{r}^T \mathbf{r}) = \mathbf{r}^T \underbrace{D^T D}_{J} \mathbf{r} + \lambda(1 - \mathbf{r}^T \mathbf{r}) \quad (3.4)$$

we obtain the following system of equations

$$\frac{\partial L}{\partial \mathbf{r}} = 2J\mathbf{r} - 2\lambda\mathbf{r} = 0, \quad (3.5)$$

$$\frac{\partial L}{\partial \lambda} = 1 - \mathbf{r}^T \mathbf{r} = 0. \quad (3.6)$$

Thus, the minimization of (3.3) reduces to an eigenvalue problem of the matrix  $J$ , which is called the structure tensor. As  $J$  is symmetric the eigenvalues are positive and real-valued. Hence, to minimize (3.3) let  $\mathbf{r}$  be an eigenvector of  $J$ . Then we have

$$\min \|D\mathbf{r}\|_2^2 = \mathbf{r}^T J \mathbf{r} = \mathbf{r}^T \lambda \mathbf{r} = \lambda. \quad (3.7)$$

Therefore, it is obvious that the minimum of (3.3) is obtained if  $\mathbf{r}$  corresponds to the eigenvector of  $J$  with the smallest eigenvalue  $\lambda$ . To obtain the optical flow vector for the current pixel, the eigenvector has to be renormalized. To this end, it is divided by its last entry, which is removed afterwards.

### 3.1.3 Handling Covariances Caused by Derivative Filters

In general, one can say that established total least squares methods estimate the most likely corrections  $A_e$  and  $b_e$  to a given data matrix  $[A, b]$  perturbed by additive Gaussian noise, such that there exists a solution  $\mathbf{u}$  with  $[A + A_e, b + b_e][\mathbf{u}, -1] = 0$ . In practice, regression imposes a more restrictive constraint, namely the existence of a solution  $x$  with  $[A + A_e]x = [b + b_e]$ . In addition, more complicated correlations arise canonically from the use of linear filters, e.g. derivative filters. In [4] we, therefore, propose a maximum likelihood estimator for regression in the general case of arbitrary positive definite covariance matrices. This leads to an unconstrained minimization of a multivariate polynomial which can, in principle, be carried out by means of a Gröbner basis.

There exist several extensions of the structure tensor method leading to more accurate results, for example the integration of brightness variations [50] and the consideration of outliers [6]. Other local methods have been proposed by Anandan [3], who uses a block matching approach to compute flow vectors, and Farnebäck [37], who introduces orientation tensors and a region segmentation method to obtain fast and accurate velocity estimates.

## 3.2 Global Methods

Global methods are formulated as energy optimization problems consisting of a data term and a regularization term. The data term usually ensures some constancy constraint, such as the brightness or gradient constancy. The regularizers are used to obtain a spatio-temporal relation of neighboring flow estimates. In this way, information is 'transported' into regions, where otherwise no or only little image information is available. This effect is called the filling-in effect of global methods. Advantages of global methods are that they yield dense flow fields and mostly avoid aperture problems. Disadvantages are that they are usually more complicated to implement than local methods and require higher computation effort. Furthermore, they are more sensitive to noise [10].

Numerous global methods are known today. An overview can be found in [10, 74].

### 3.2.1 Horn and Schunck

The first global method was proposed in 1981 by Horn and Schunck. They minimized the following energy functional consisting of the brightness constancy constraint and a simple smoothness assumption relating neighboring flow vectors

$$E(\mathbf{u}) = \underbrace{\int_{\Omega} (I_x \mathbf{u}_1 + I_y \mathbf{u}_2 + I_t)^2}_{\text{data term}} + \lambda \underbrace{(\|\nabla_2 \mathbf{u}_1\|^2 + \|\nabla_2 \mathbf{u}_2\|^2)}_{\text{regularization term}} dx dy. \quad (3.8)$$

The energy is minimized using the calculus of variations described in chapter 2, which leads to the following Euler-Lagrange equations

$$(I_x \mathbf{u}_1 + I_y \mathbf{u}_2 + I_t) I_x - \mathbf{u}_{1xx} - \mathbf{u}_{1yy} = 0, \quad (3.9)$$

$$(I_x \mathbf{u}_1 + I_y \mathbf{u}_2 + I_t) I_y - \mathbf{u}_{2xx} - \mathbf{u}_{2yy} = 0. \quad (3.10)$$

To solve this linear system of equations, for example the conjugate gradient method or the Gauss-Seidel method in combination with successive overrelaxation (SOR) can be used.

### 3.2.2 Bruhn et al.

In [28], Bruhn et al. proposed the combined local global (CLG) method, which integrates the advantages of local methods into a global framework. There are two variants of this method: a linear one and a nonlinear one. The nonlinear method explicitly allows for discontinuities in the flow field. Let  $J_\rho(\nabla_3 I) = K_\rho * (\nabla_3 I)(\nabla_3 I)^T$  denote the structure tensor, where  $K_\rho$  indicates the convolution with a Gaussian kernel of standard deviation  $\rho$ , and let  $\tilde{\mathbf{u}}$  denote the vector  $\mathbf{u}$  with an additional last coordinate set to 1. Then the following energy is minimized in the linear case

$$E(\mathbf{u}) = \int_{\Omega} \tilde{\mathbf{u}} J(\nabla_3 I) \tilde{\mathbf{u}}^T + \lambda (\|\nabla \mathbf{u}_1\|^2 + \|\nabla \mathbf{u}_2\|^2) dx dy . \quad (3.11)$$

We obtain the following system of Euler-Lagrange equations

$$J(\nabla_3 I) \tilde{\mathbf{u}}^T - \mathbf{u}_{1xx} - \mathbf{u}_{1yy} = 0, \quad (3.12)$$

$$J(\nabla_3 I) \tilde{\mathbf{u}}^T - \mathbf{u}_{2xx} - \mathbf{u}_{2yy} = 0, \quad (3.13)$$

which can again be solved using linear solvers such as SOR.

In the nonlinear case, a nonlinear function  $\psi$  is introduced to handle outliers in the data term as well as in the regularization term

$$E(\mathbf{u}) = \int_{\Omega} \psi_1(\tilde{\mathbf{u}} J(\nabla_3 I) \tilde{\mathbf{u}}^T) + \lambda \psi_2(\|\nabla \mathbf{u}_1\|^2 + \|\nabla \mathbf{u}_2\|^2) dx dy . \quad (3.14)$$

An example for  $\psi$  is Charbonnier's function, which is convex in  $s$

$$\psi(s^2) = 2\beta^2 \sqrt{1 + \frac{s^2}{\beta^2}} . \quad (3.15)$$

In this case we obtain a nonlinear system of Euler-Lagrange equations

$$\begin{aligned} \psi'_1(\tilde{\mathbf{u}} J(\nabla_3 I) \tilde{\mathbf{u}}^T)(J_{11}\mathbf{u}_1 + J_{12}\mathbf{u}_2 + J_{13}) - \lambda \operatorname{div}(\psi'_2(\|\nabla \mathbf{u}\|^2)\nabla \mathbf{u}_1) &= 0, \\ \psi'_1(\tilde{\mathbf{u}} J(\nabla_3 I) \tilde{\mathbf{u}}^T)(J_{21}\mathbf{u}_1 + J_{22}\mathbf{u}_2 + J_{23}) - \lambda \operatorname{div}(\psi'_2(\|\nabla \mathbf{u}\|^2)\nabla \mathbf{u}_2) &= 0. \end{aligned}$$

This system can be solved using gradient descent methods.

# Chapter 4

---

## Error Analysis

### 4.1 Introduction

The estimation of optical flow has been a fundamental problem in computer vision ever since the pioneering work of Fennema and Thompson [39]. Over the last three decades, the importance of this problem has spawned the development of algorithms and their usage to a wide range of disciplines, ranging from robot vision to scientific and measurement applications. Along with the diversity of methods comes the need for a thorough quantitative evaluation of their accuracy and applicability to different scenes. In order to evaluate confidence measures, which predict the error of computed flow fields, first commonly used error measurements of optical flow fields need to be introduced and analyzed. Such error measures are of high importance for the understanding of strengths and weaknesses of these algorithms, for scientific and industrial applications and for confidence estimation. However, the evaluation has been limited to mostly the indication of the average angular error and its variance for a small number of highly artificial test sequences. I propose to comprehend the evaluation of motion estimators as a sampling from a joint probability distribution, namely the joint distribution of the true and the estimated flow, as well as local gray value neighborhoods. Marginals and conditionals of this distribution allow for a detailed assessment of optical flow algorithms. For the ranking of motion estimators, a new indicator is suggested, a scalar measure, which overcomes conceptual difficulties of the average angular error. The expressiveness of the proposed method is shown for five different flow estimators and five test sequences.

#### 4.1.1 Motivation

Seminal work has been dedicated to the task of accuracy evaluation by Barron and Fleet [10], was followed by Mitiche and Bouthemy [74], Stiller and Konrad [94], Kalkan et al. [59] and others. Expressive measures of errors along with a detailed investigation of

estimated flow fields, in conjunction with the image sequences used in testing, will help to better understand and judge the specific strengths and weaknesses of flow estimators. The choice of ground truth test sequences has tremendous influence on the significance of comparative studies. So far, most quantitative studies on optical flow are restricted to the indication of the average angular error [42] and its variance. This error is defined as the angle between two three dimensional vectors, namely the true and the estimated displacement vector augmented to 3D by setting the third coordinate to one. Recently, Baker et al. [8] have suggested to report quantiles of the statistics of the endpoint error, i.e. the length of the difference vector between the estimated and the true flow vector. Unlike relative error measures, which suffer from possible divisions by zero, the angular error and the endpoint error are well-defined. This advantage is complemented by the representation of the error in a single scalar value, which usually is computed by taking the average over the whole flow field. A careful analysis reveals several crucial drawbacks of the angular error which are summarized in section 4.2.

A generalized view on the evaluation of optical flow estimators is proposed, which is essentially understood as a sampling from the joint PDF of the ground truth and computed flow field, as well as the gray value structure of local pixel neighborhoods. Relevant properties of the estimator can be derived as marginals, conditionals and scalar descriptors from this distribution. The more different test sequences are used to sample from the PDF, the more generally valid become statements on the accuracy of flow estimators. By means of marginalization and conditioning of the PDF, the relation between error measures and local gray value structure can be assessed. E.g., one can ask for the distribution of a scalar error measure within homogeneous regions, or conversely for the gray value structure of local neighborhoods for a given error interval. The results of the operations on the PDF can be visually and quantitatively examined. Hence, new insights into the quality and drawbacks of a single estimator as well as quantitative comparisons between different estimators can be made.

An important aim of comparative studies is the ranking of optical flow estimators. While complex descriptors of error distributions are useful to reveal detailed properties of the estimator, they are inadequate for ranking. Instead, a scalar indicator is desirable to impose an order on the estimators. Based on the endpoint error, which naturally results from the PDF framework, the integral over the cumulative distribution function is computed in order to rank different methods.

In the results section the quality of five well-established optical flow estimators are evaluated and compared. In this way, it is demonstrated how the proposed general framework can be applied to obtain all kinds of statements on the quality of an estimator based on a single PDF. In addition, results for known estimators are provided, which can be used to compare against the quality of new estimators in the future.

### 4.1.2 Related Work

The following measures have been proposed as measures of discrepancy between the ground truth vector  $\mathbf{g} \in \mathbb{R}^d$  and the estimated flow vector  $\mathbf{u} \in \mathbb{R}^d$ .

#### Angular Error

The most widely used error measure is the angular error,

$$E_\alpha(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \frac{180}{\pi} \arccos \frac{\tilde{\mathbf{u}}(\mathbf{x}) \cdot \tilde{\mathbf{g}}(\mathbf{x})}{\|\tilde{\mathbf{u}}(\mathbf{x})\| \|\tilde{\mathbf{g}}(\mathbf{x})\|}, \quad (4.1)$$

which was suggested by Barron and Fleet [10] and dates back to prior work by Fleet and Jepson [42]. A brief discussion of the angular error can be found in [51].

#### Endpoint Error

The length of the difference vector between the true and the computed vector,

$$E_2(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \|\mathbf{u}(\mathbf{x}) - \mathbf{g}(\mathbf{x})\|, \quad (4.2)$$

was proposed by Otte and Nagel [80] and revived as endpoint error by Baker et al. [8]. This measure discounts errors in regions of small flow.

#### Angle Error

The angle between the correct and the estimated flow vector,

$$E_\phi(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \frac{180}{\pi} \arccos \frac{\mathbf{u}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x})}{\|\mathbf{u}(\mathbf{x})\| \|\mathbf{g}(\mathbf{x})\|}, \quad (4.3)$$

is usually referred to as the angle error. Since it does not take into account the error in length, it is usually indicated together with the magnitude error of the flow vector.

#### Magnitude Error

The magnitude error is defined as the absolute difference of the magnitudes

$$E_m(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \|\|\mathbf{u}(\mathbf{x})\| - \|\mathbf{g}(\mathbf{x})\|\|. \quad (4.4)$$

This error measure does not account for errors in direction and, hence, is usually indicated together with the angle error.

### Relative Magnitude Error

The relative magnitude error

$$E_\mu(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \frac{|\|\mathbf{u}(\mathbf{x})\| - \|\mathbf{g}(\mathbf{x})\||}{\|\mathbf{g}(\mathbf{x})\|} \quad (4.5)$$

relates the absolute magnitude error to the length of the ground truth vector. Due to divisions by values close to zero this error measure is problematic for ground truth vectors of very small magnitude [73].

### Error Normal to the Gradient

In order to measure how effectively an algorithm compensates for the aperture problem, Galvin et al. [44] propose to measure the error normal to the gradient

$$E_\perp(\mathbf{g}(\mathbf{x}), \mathbf{u}(\mathbf{x})) = \|(\mathbf{u}(\mathbf{x}) - \mathbf{g}(\mathbf{x})) \mathbf{f}^\perp(\mathbf{x})\|, \quad \mathbf{f}^\perp(\mathbf{x}) = (-\partial_y I(\mathbf{x}), \partial_x I(\mathbf{x}))^T. \quad (4.6)$$

### Gray Value Differences

Error measures based on the squared gray value differences between the original frame and the frame warped by the computed motion field have been proposed by Baker et al. [8]. The problem with these measures is their strong dependence on a frame interpolation algorithm. Furthermore, the error depends on the homogeneity of the scene, because any incorrect flow vector pointing to a location with identical gray value is considered correct.

#### 4.1.3 Contribution

My contribution is threefold.

First, due to several shortcomings of the angular error a generalized evaluation of optical flow methods is suggested, which is based on marginals and conditionals of the PDF comprising the ground truth flow, the computed flow and the gray value neighborhood. To compute the PDF as many test sequences as possible should be used to gain independence from the image sequence. In this way, many questions on the quality of estimators can be answered, e.g. questions concerning the best estimator in case of high velocities or typical gray value structures in case of high errors. Thus, an evaluation method is suggested, which comprises previously used analysis methods and at the same time allows for various, more specific questions. The results can be visually examined in order to gain new insights into special problems or advantages of estimators, but quantitative statements can be made as well to allow for comparisons between estimators.

Second, a scalar indicator is proposed to rank different optical flow methods. This indicator is based on the cumulative distribution function of the absolute error of the flow

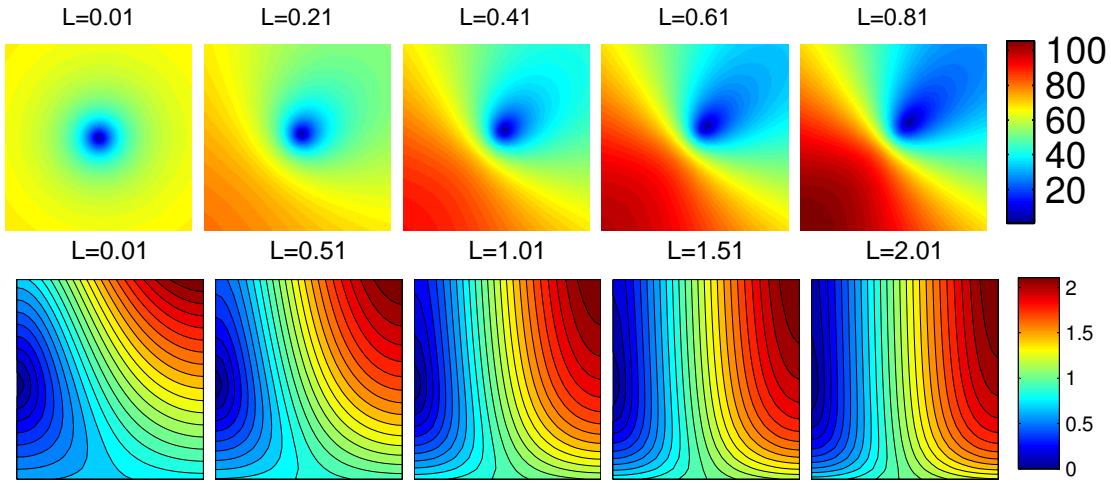


Figure 4.1: Top: Angular error for  $(\mathbf{u}_1, \mathbf{u}_2)^T \in [-10, 10] \times [-10, 10]$  for different ground truth lengths  $\mathbf{g} = L \cdot (1, 1)^T$ . Bottom: Angular error isocontours for increasing angle error ( $[0, \pi]$ , horizontal axis) and relative magnitude error ( $[0, 2]$ , vertical axis), for different ground truth lengths  $L$ .

field.

And third, results are shown for the comparison of five known optical flow estimators within the proposed framework. These results can be understood as starting point for the comparison between known and new estimators. Besides, interesting statements on specific questions such as the most reliable method in case of high velocities are made. This approach has been submitted [63].

## 4.2 Discussion of the Angular Error

The angular error has been used as a scalar indicator for the ranking of optical flow estimators. However, given the accuracy of recent algorithms, conceptual shortcomings of this measure become apparent. This section is devoted to the discussion of these drawbacks.

**The angular error depends non-linearly on the true magnitude.** The top row of Figure 4.1 depicts the angular error for all kinds of flow vectors within the interval  $[-10, 10] \times [-10, 10]$  for a given ground truth flow vector  $L \cdot (1, 1)^T$ . Here  $L$  determines the length of the ground truth vector and is increased from left to right. It can be seen that, for a small magnitude  $L$  of the true displacement, the angular error is governed by the deviation in magnitude, because the error homogeneously grows in all directions. Conversely, for higher magnitudes  $L$  of

$$\begin{aligned}
 & \lim_{L \rightarrow \infty} E_\alpha(\mathbf{x}) \\
 &= \lim_{L \rightarrow \infty} \arccos \frac{L (\mathbf{u}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x}) + \frac{1}{L})}{L \left\| \begin{pmatrix} \mathbf{u}_1(x) \\ \mathbf{u}_2(x) \\ \frac{1}{L} \end{pmatrix} \right\| \|\tilde{\mathbf{g}}(\mathbf{x})\|} \\
 &= \arccos \frac{\mathbf{u}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x})}{\|\mathbf{u}(\mathbf{x})\| \|\tilde{\mathbf{g}}(\mathbf{x})\|}
 \end{aligned}$$

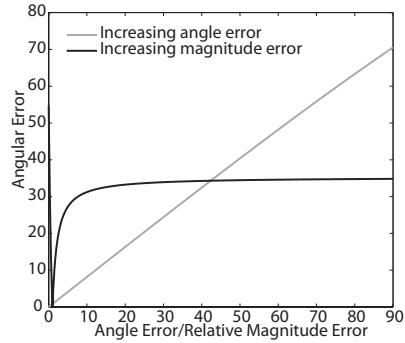


Figure 4.2: The angular error is bounded with respect to magnitude errors, but not with respect to angle errors.

the true displacement, the angle error dominates. This relation is non-linear and rather arbitrary.

**The angular error is bounded for magnitude errors, not for angle errors.** Let  $\mathbf{u}(\mathbf{x})$  be a computed displacement vector, which is varied in length by multiplication with a factor  $L$ . Then the limit for  $L$  tending to infinity is given in Figure 4.2. On the right hand side the angular error is shown for a true displacement vector  $(1, 1)^T$  and a computed vector with a) increasing angle error and b) increasing magnitude error. The plot shows that the angular error is bounded for increasing deviations in magnitude, but not for deviations in angle. Hence, a vector of length infinity with zero angle error is seen as equally “correct” as a vector with thirty degrees angle error.

**The angular error is not invariant against the sign of magnitude deviations.** An estimated flow vector parallel to the true displacement but being too short gives rise to a different angular error than a parallel vector that overestimates the displacement by the same percentage (see bottom row Figure 4.1). Yet, both cases should yield the same error value, since a vector that is 10% too short and a vector that is 10% too long are, in fact, equally “correct”.

**The influence of the magnitude and angle error interdepend.** The bottom row of Figure 4.1 shows isocontours for angle errors around 90 degrees, which are nearly parallel to the vertical axis. This means that for angle errors around 90 degrees the magnitude error does not influence the angular error at all. For smaller and larger angle errors the influence of the magnitude error increases again. This problem is especially apparent for larger speeds of the ground truth vector. In contrast, for angle errors around 0 and 180 degrees the isocontours are almost parallel to

the horizontal axis. This shows that for very small and very large angle errors the angular error almost exclusively depends on the magnitude error.

**The average angular error and its variance are insufficient characteristics.**

Another problem when comparing the quality of different optical flow techniques is that papers in this area are mostly limited to the indication of the average angular error and its variance, the first and second order moments of the error distribution. For comparisons and rankings of different methods, the variance is even often not taken into account. This is insufficient because specific problems or advantages are neglected, and a single outlier can distort the comparison.

Due to these shortcomings of the angular error a new policy and error measurement for the evaluation of optical flow estimators is proposed.

## 4.3 The Joint Distribution of Optical Flow Estimation

I comprehend the evaluation of optical flow estimates from a set of test sequences as a sampling from a joint PDF, from which relevant properties of the estimator can be derived as marginals, conditionals and scalar descriptors, respectively. The application of any motion estimator to (possibly several) sequences of images results in a set of samples, one for each pixel, which are assumed independent

$$\forall \mathbf{x} \in D : (\mathbf{g}_1(\mathbf{x}), \mathbf{g}_2(\mathbf{x}), \mathbf{u}_1(\mathbf{x}), \mathbf{u}_2(\mathbf{x}), J_1(\mathbf{x}), \dots, J_n(\mathbf{x})) \in \mathbb{R}^{n+4} . \quad (4.7)$$

Herein,  $(\mathbf{g}_1(\mathbf{x}), \mathbf{g}_2(\mathbf{x}))^T$  and  $(\mathbf{u}_1(\mathbf{x}), \mathbf{u}_2(\mathbf{x}))^T$  shall denote the true and the estimated displacement field, respectively, and  $(J_1(\mathbf{x}), \dots, J_n(\mathbf{x}))^T$  shall be the vector of gray values in a local neighborhood of the pixel  $\mathbf{x}$ . These are samples from the PDF, which would result from the application of the estimator to *all* sequences of images. Although this PDF is unavailable in practice, approximations based on the set of samples can be obtained by means of density estimation. To this end, exemplary results are shown for Parzen kernel density estimates [82]. Once an estimate of the PDF has been computed, marginal and conditional distributions as well as scalar descriptors derived from this PDF open a generic and direct view to evaluating the quality of optical flow estimators.

### 4.3.1 Marginals and Conditionals

#### Image Sequences

The representativeness of the estimated PDF highly depends on the choice of test sequences. It is, therefore, important to consider the distribution of the true displacement accumulated over the set of test sequences, ideally in conjunction with spatio-temporal gray value context, namely to consider the marginal distribution of  $(\mathbf{g}_1, \mathbf{g}_2, J_1, \dots, J_n)$ .

Based on this distribution, marginals and conditionals can be computed, which allow for answers to various kinds of questions concerning the quality of the optical flow estimator. To this end, a low-dimensional representation of the gray values of local neighborhoods can be obtained by PCA [57], ICA [32], the local variance and local entropy [90], respectively.

### Flow Field Discrepancy

The discrepancy between the true and the estimated flow field is a very important factor in the evaluation of optical flow estimators. A thorough quantitative analysis of this discrepancy is, therefore, indispensable. Let us consider the bivariate PDF of  $(\mathbf{u}_1 - \mathbf{g}_1, \mathbf{u}_2 - \mathbf{g}_2)^T$ , i.e. the distribution of the difference vector between the true and the estimated flow. Due to its fundamental importance, I recommend that estimates of this distribution be visualized (e.g. Figure 4.4) when analyzing the quality of optical flow estimators.

In addition, the (bivariate) distribution of the magnitudes of the true and the estimated flow field is of interest (e.g. Figure 4.6), namely the marginal distribution of  $(\|\mathbf{u}\|, \|\mathbf{g}\|)$ . In this way, systematic errors for specific lengths can be observed for different optical flow methods. For special applications, e.g. driver assistance systems, the performance of the methods for very large flow magnitudes are of specific interest and can be investigated in this way.

### Causes of Discrepancy

Once the discrepancy of the true and the estimated flow field has been quantified, the question is vital, in which local context the highest errors occur. In fact, conditioning the joint PDF on a certain interval of the endpoint error (4.2) yields a PDF of local neighborhoods of gray values. Conversely, a distribution of the endpoint error is obtained by conditioning the joint PDF on an interval of a scalar texture descriptor (i.e., the local gray value entropy). In this way the distribution of the error in case of special gray value structures becomes apparent.

### Ranking of Estimators

To obtain a scalar indicator suited for comparing different optical flow methods I propose to compare the integral over the cumulative distribution functions of the endpoint error (4.2). This scalar indicator is not bounded with respect to large magnitude errors, it is invariant against the sign of deviation in magnitude, it does not depend on the ground truth magnitude and it is a statistical measure reflecting the error distribution in a more detailed way than the indication of average error and variance. Hence, this indicator evades the difficulties reported for the average angular error.

## 4.4 Experiments and Results

Five optical flow estimators and five standard test sequences are considered and various information from the proposed joint PDF is derived. The following results are intended as a demonstration of expressiveness of the proposed method rather than as a comprehensive assessment of recent motion estimators. The algorithms by Nir et al. [78], Farnebäck [37], Bruhn et al. [28] (2D linear CLG), Horn and Schunck [53] and Bigün et al. [16] (for some of them compare Chapter 3) are considered, which are applied to the Marble [80], Street [73], Office [73], Yosemite [52] and Rubber Whale [8] sequence. For the method of Nir et al., the estimated flow field for the Yosemite sequence was kindly provided by the authors. For the Farnebäck method the Matlab implementation provided online by the authors [35] is used. The other algorithms had to be re-implemented and relevant parameters were chosen as follows. The regularization strength, the Gaussian pre-smoothing variance and the spatial integration of the structure tensor were jointly optimized for each sequence. The structure tensor was computed by means of isotropy-optimized 7x7x7 Scharr filters [89]. Unless otherwise noted, results are accumulated over all considered test sequences. Note especially that for Nir's method, results are restricted to the Yosemite sequence. Where errors are visualized by shading, the scale of the shading is logarithmic.

### Comparing the Marginal Distributions of $(\mathbf{u})$ and $(\mathbf{g})$

To begin with, by marginalizing over the gray values  $J_1, \dots, J_n$  and either the estimated  $\mathbf{u}$  or the true flow  $\mathbf{g}$ , the nature of these flow fields becomes apparent and the two can be compared (see Figure 4.3). In fact, these distributions are quite similar as can be expected from a rather accurate flow estimator such as Farnebäck's. Deviations are mainly due to inaccurate results obtained on the Rubber Whale sequence.

### Comparing the Marginal Distribution of $(\mathbf{u} - \mathbf{g})$

In order to precisely assess accuracy, I marginalize over the gray values  $J_1, \dots, J_n$  and analyze the distribution of the deviations  $(\mathbf{d}_1, \mathbf{d}_2)^T = (\mathbf{u}_1 - \mathbf{g}_1, \mathbf{u}_2 - \mathbf{g}_2)^T$ . Figure 4.4 shows Parzen estimates of the  $x$  and  $y$  velocity deviations. For a perfect flow estimator, one would expect a single point at  $(0,0)$ . Farnebäck's and Nir's methods outperform the other algorithms as the distributions are more centered than the others. It can be seen that the distribution of the deviations are similar for local (Farnebäck, Bigün) and global (Bruhn, Horn-Schunck) methods, especially in case that all test sequences are used. Local methods yield distributions that tend to look normal and only yield a small bias in the vertical component. In contrast, global methods often yield systematic deviations in specific directions and exhibit a larger bias in the vertical component. These systematic errors for global methods may be due to regularization. Yet, for the Yosemite sequence the methods tend to underestimate the vertical component. Furthermore, the CLG

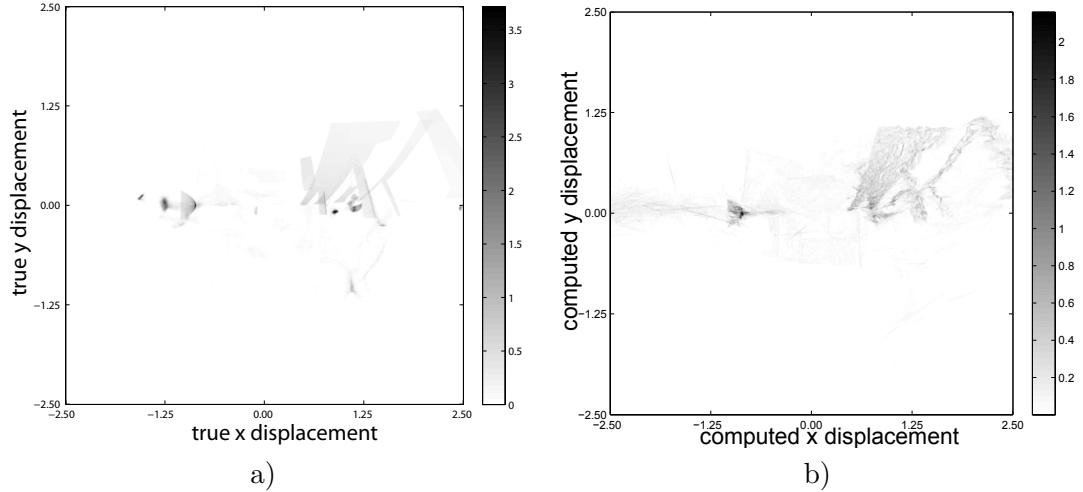


Figure 4.3: a) Distribution of ground truth velocities for all test sequences, b) Distribution of flow estimates from Farnebäck's method for all test sequences.

and Horn-Schunck methods mostly either underestimate the vertical or overestimate the horizontal component. The Nir method yields the most centered distribution but also shows systematic errors in specific directions.

#### 4.4.1 Comparing Scalar Descriptors

A numerical comparison of different flow estimators based on a single value is problematic, since it is hardly possible to include the most important characteristics of the flow field in one figure only. To avoid simple averaging of the error, I propose to integrate over the cumulative distribution function of the endpoint error (4.2). Figure 4.5 shows the cumulative distribution functions over the angular and endpoint error based on all test sequences and based on the Yosemite sequence only. A bin size of 0.0005 was used to compute the cumulative distribution function. The corresponding tables in the same figure compare the integral over the cdf to the respective error measurement.

The resulting order naturally corresponds to that inferred by the angular error, which again confirms the known quality ranking of these estimators. However, the relation between the results of different methods differ, e.g. when comparing the results for all sequences for the methods by Horn and Schunck and by Bigün. Here, the structure tensor result shows an angular error that exceeds the one of the Horn-Schunck method by 2.79. In contrast, the integrals over the corresponding cdfs differ only marginally.

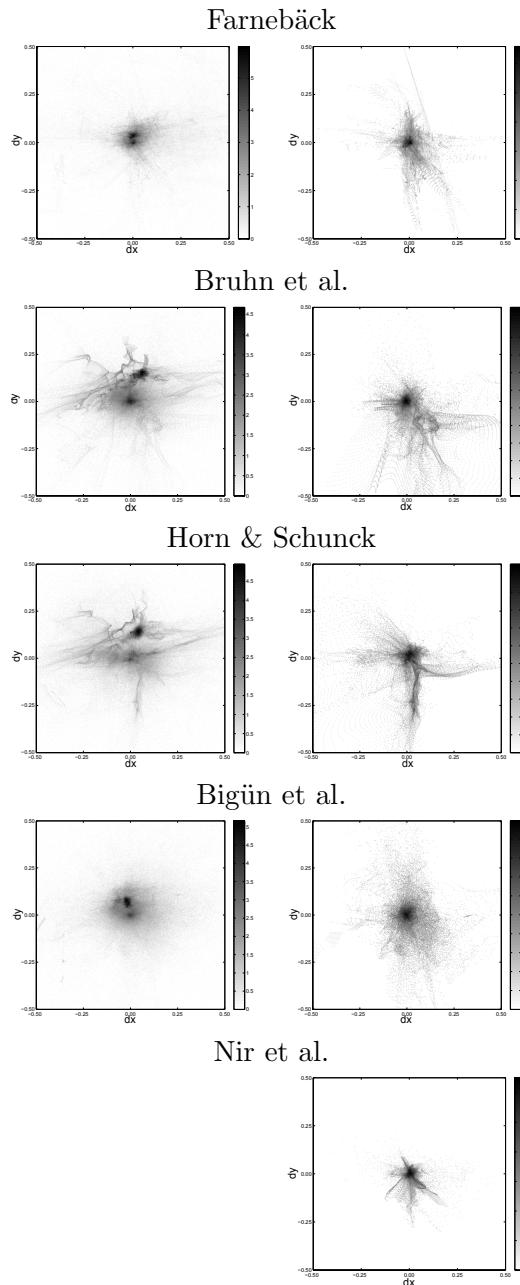


Figure 4.4: Parzen estimate of the distribution of the x and y velocity deviations for different flow estimators over all test sequences (left) and only Yosemite (right).

## 4 Error Analysis

---

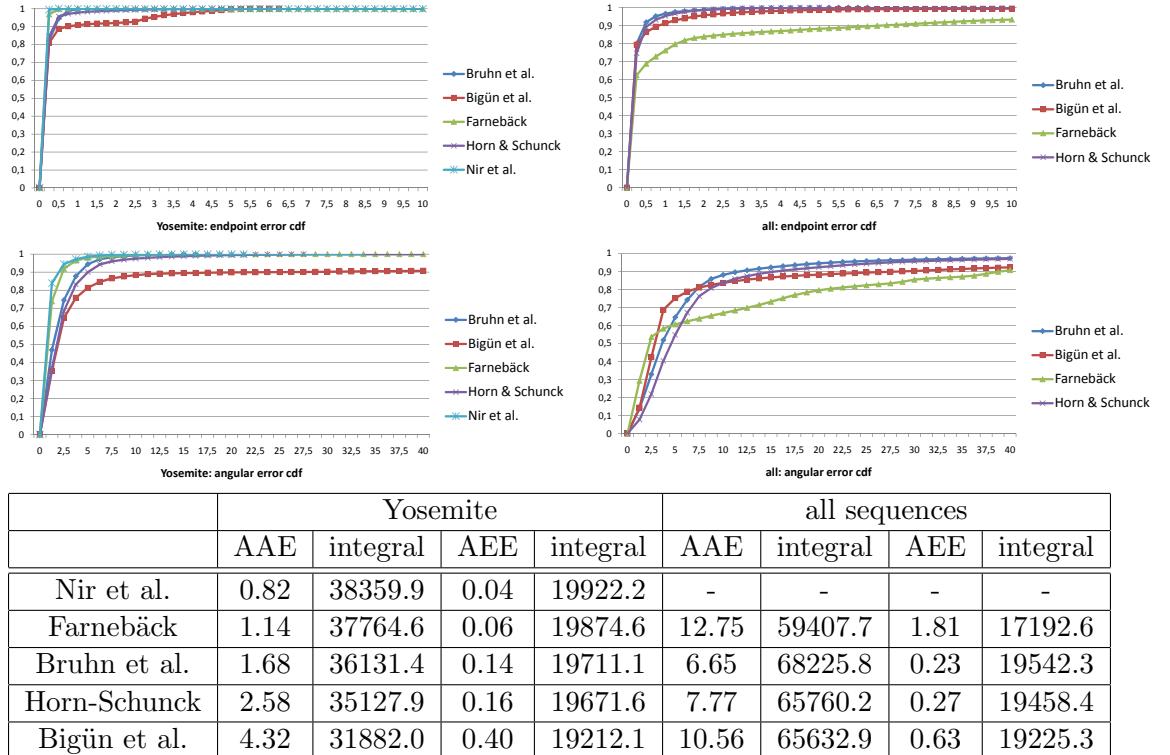


Figure 4.5: Cumulative distribution functions (cdf) of the angular and endpoint error for the compared flow methods and table displaying the average angular error (AAE), the average endpoint error (AEE) and the integral over the cdf for the Yosemite sequence and for all test sequences.

### 4.4.2 The Marginal Distribution of $(\|u\|, \|g\|)$

To test optical flow algorithms for a bias in magnitude of the estimated flow field, I marginalize over all gray values and estimate the joint distribution of the magnitude of the true and the estimated displacement (Figure 4.6). Note that, for an ideal flow estimator, the density should be concentrated around the angle bisector. For none of the methods a severe bias can be observed for the Yosemite sequence. Yet, except for the Nir method, all estimators exhibit difficulties with large flows. This can be observed in Figure 4.6, because the scatter for larger ground truth magnitudes is so severe that the distribution becomes invisible. This is due to the latter methods linearizing the brightness constancy equation, which becomes less reliable for larger flow vectors. The vertical line for ground truth lengths around 1.25 in the Farnebäck result is probably due to the nature of the errors in this flow field, as these almost exclusively occur at the principal motion boundary, where the flow length is approximately 1.25.

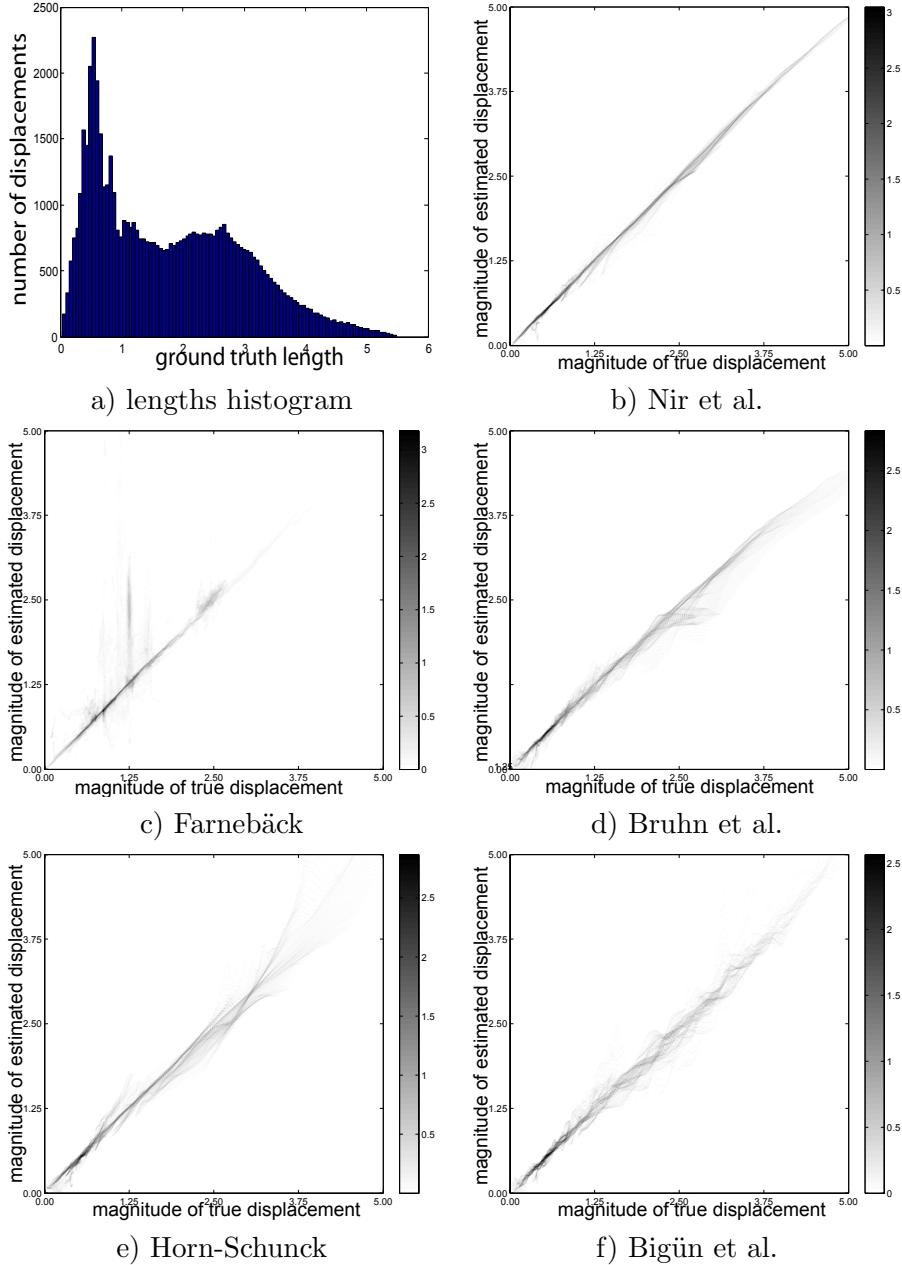


Figure 4.6: Distribution of the ground truth flow length (a) compared to the computed flow length (b-f) for all tested methods on the Yosemite sequence. The results show that only Nir's method yields reliable results for large displacements.

#### 4.4.3 Conditioning on Error Intervals

In contrast to common accuracy measurement techniques the proposed joint PDF allows to investigate the causes of discrepancy. I condition on a certain endpoint interval and marginalize over the flow vectors and thereby obtain the distribution of the corresponding gray value neighborhoods. As the result is multidimensional, PCA is applied in order to obtain a lower dimensional subspace, which still preserves most of the variance of the data. The resulting eigencomponents correspond to the main axes of the PDF. Figure 4.7 shows the five eigencomponents associated with the highest variance, for four estimators based on the Marble sequence. It was conditioned on very high (left) and very low (right) endpoint errors, respectively. It can be seen that the eigencomponents of the structure tensor method significantly differ from those of the other algorithms.

The former clearly indicate that edges pose problems to the structure tensor, the reason being that these often coincide with discontinuities of the displacement field, which violate the assumption of constant flow within the integration area. In contrast, well-textured or smooth regions yield low error values. For the Farnebäck method the results are similar. Here edges pose problems as well, whereas highly textured regions yield lower errors. The first three eigencomponents are even almost identical for the structure tensor method by Bigün and the Farnebäck method. The second and third eigencomponent of Farnebäck's method as well as the second to fifth component of Bigün's method hint at high errors in case of aperture problems. The CLG method by Bruhn et al. also shows problems in case of edges due to motion boundaries, which are oversmoothed by the regularization term. In contrast, for the Horn and Schunck method the eigencomponents of the PDF conditioned on high and low endpoint errors look very similar. This suggests that the gray value structure has only minor influence on errors. This may be due to the regularization, which is independent of the image sequence in case of the Horn-Schunck method.

#### 4.4.4 Conditioning on Gray Value Structures

Finally, I condition on gray value structure and investigate the distribution of the endpoint error. Using the standard deviation of gray values in local neighborhoods as well as the local gray value entropy, bivariate densities relating these scalar descriptors of texture to the endpoint error (cf. Figure 4.8) are obtained. It becomes apparent that the endpoint error is larger for high gray value entropy and small gray value standard deviations, respectively. That means that high frequency structures and rather low frequency regions tend to produce higher endpoint errors.

Many of the statements made for single estimators have been known before. However, automatically computing qualitative and quantitative results at the same time within a unified framework over a number of test sequences is a new concept and will help to

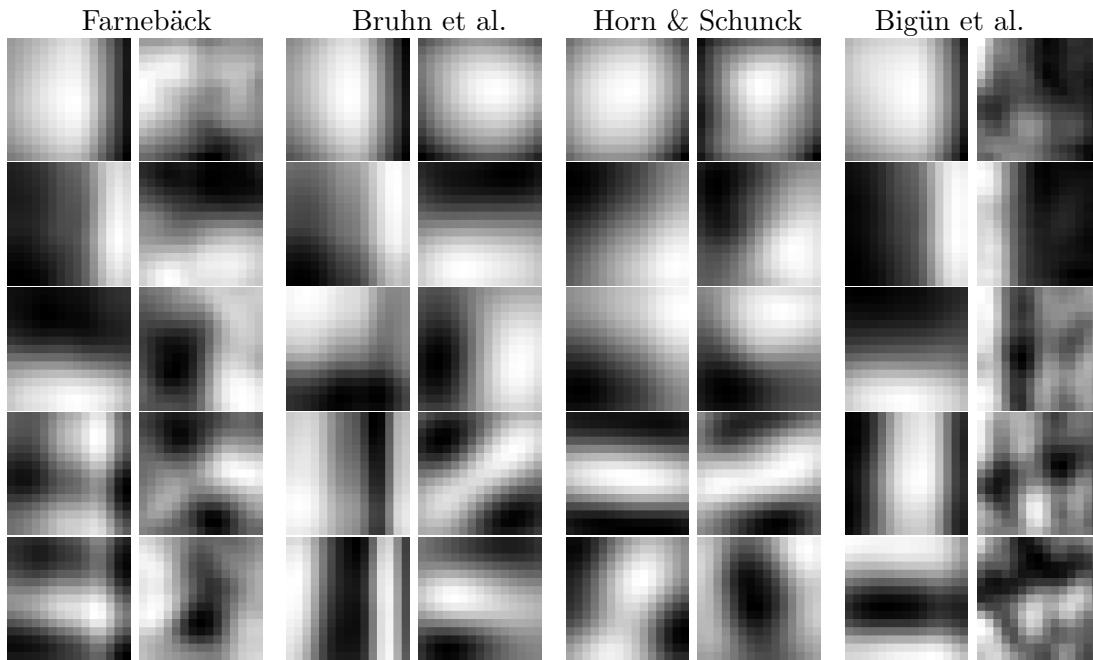


Figure 4.7: Eigencomponents with highest variance of pdf conditioned on high endpoint errors (left column) and low endpoint errors (right column) for four different estimators based on the Marble sequence.

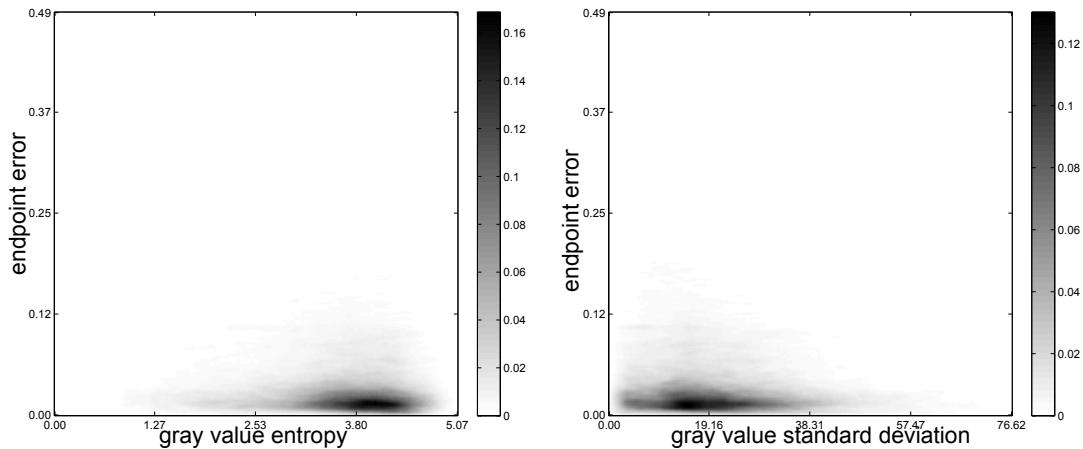


Figure 4.8: Left: Distribution of the neighborhood entropy and the endpoint error, Right: Distribution of the neighborhood gray value standard deviation and the endpoint error, both for the Nir method.

improve the analysis of optical flow estimators.

## **4.5 Summary and Conclusion**

In this chapter I proposed to analyze the accuracy of optical flow estimators within a uniform framework by means of a single joint probability distribution over gray value neighborhoods, ground truth and computed flow fields. I have shown that by means of marginalization and conditioning interesting, additional information on the quality and properties of optical flow estimators can be obtained from the suggested distribution. For example, the results showed that all estimators except the Nir method have difficulties with large displacements. Furthermore, principal components showing typical gray value structures in case of very high or very low endpoint errors were obtained. By this statistical viewpoint, I hope to further improve optical flow evaluation methods by a) systematically analyzing certain problematic areas such as large displacements through conditioning and marginalization, b) making the evaluation of optical flow estimators independent of the underlying test sequences and c) offering an improved scalar error measure in order to overcome the described shortcomings of the average angular error. As all statistics can be computed on any data sets, this approach becomes highly adaptable to application-dependent requirements, e.g. car driving sequences. The method was applied to the analysis of five exemplary optical flow estimators on five different test sequences. For each method the distribution of the velocity deviations was depicted and the proposed scalar valued indicator was computed which allowed for the ranking of the analyzed estimators. Here, the order of the methods was similar to that inferred by the angular error, but not identical. By means of the proposed analysis method many insights into the quality of the tested flow estimators could be obtained.

# Chapter 5

---

## Predictability and Situation Measures

### 5.1 Introduction

Even though many optical flow estimators exist, none of them solves the optical flow problem satisfactorily. In fact, all of these methods are prone to errors in specific situations. Hence, it is important to identify the unreliable flow vectors prior to further processing steps. To this end, situation and confidence measures can be used. Both types of measures assign a level of reliability to each single motion vector. Situation measures mostly only take into account the image sequence and judge the hypothetical complexity of estimating the optical flow accurately based on the image data. They can be defined as mappings from the image domain and the image intensity to the interval of confidence  $[0, 1]$ , where 1 stands for high and 0 for low reliability:

$$\varphi : D \times I \rightarrow [0, 1] . \quad (5.1)$$

In contrast, confidence measures evaluate the reliability of a given optical flow field and, thus, map from the image domain, the image intensity and the flow field to the interval  $[0, 1]$

$$\varphi : D \times I \times \mathbb{R}^d \rightarrow [0, 1] . \quad (5.2)$$

There are several fields of application for situation and confidence measures. In general, both kinds of measures are important for every application that uses optical flow and whose results depend to any extent on the accuracy of the displacement field. There are mainly two large fields of application for optical flow methods:

- areas, where an accurate result is important such as medical applications [49], data compression [43] or particle image velocimetry [83],

- areas, where real-time optical flow results are required, such as robot navigation [67] and pedestrian and vehicle tracking [47].

For both fields accuracy measurements are valuable, since for accurate applications results can be improved afterwards, and for real-time applications less accurate and, thus, faster flow computation methods can be used.

Reliable situation and confidence measures also provide valuable information for the improvement of optical flow methods themselves. For example, displacement vectors classified as unreliable could be left out in the result leading to a sparser but more accurate displacement field. Here it also might make sense to ignore these regions completely during the flow field computation or to derive flow information from surrounding pixels, where the flow can be estimated reliably [92]. Another possibility to improve the field in global methods could be to let the parameter that controls the smoothness of the field depend on the confidence or situation. In this way, vectors with higher confidence could exert higher influences on vectors with low confidence, and the smoothing of vectors with high confidence could be reduced.

In this chapter I classify, analyze and compare situation measures based on the intrinsic dimension of the image sequence (see Chapter 2.4) they examine. These measures can be successfully applied to the recognition of aperture problems, homogeneous regions and occlusions as well as to the detection of locations, where the flow vector can be computed reliably. Since these measures are affected by image noise, their stability for increasing noise levels is also investigated.

### 5.1.1 Motivation

As described before in Chapter 2.4, the intrinsic dimension of locally confined regions of the image sequence is important for the assessment of the accuracy of optical flow fields. The relationship between all situations based on their intrinsic dimension is shown in Figure 5.1. The first distinction between intrinsic dimension two and three is characterized by the existence of any spatial and/or temporal directed structure. If movement is present, there is always a temporal directed structure in the image marking the trajectory of the object. However, general directed structures can also be spatially directed. In this case, intrinsic dimensions smaller than two are observed which lead to aperture problems (edges or homogeneous regions). Therefore, movement can only be identified reliably in the presence of a temporal directed structure and the simultaneous absence of a spatial directed structure, that is in the case of intrinsic dimension two.

The problem with the assignment of reliability levels to specific intrinsic dimensions is the case of occlusion, which increases the intrinsic dimension by one as explained in Chapter 2.4. In order to examine the degree of correctness of the flow field, the augmented intrinsic dimension of occluded structures is undesired as it leads to ambiguous statements on the feasibility of accurate flow computation. For example in the case of intrinsic dimension two, one either observes a uniquely defined translation and flow

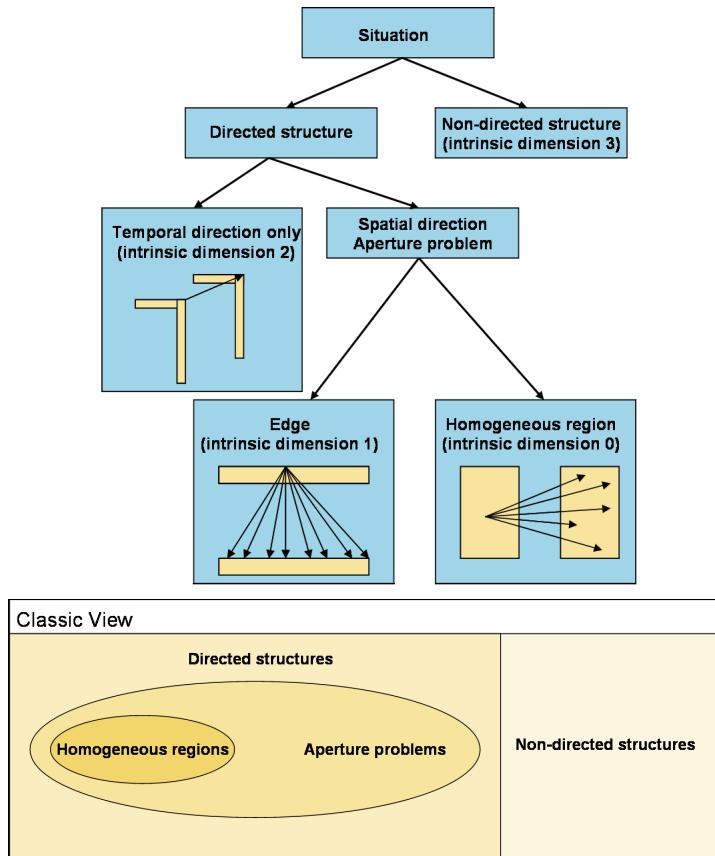


Figure 5.1: Tree and diagram illustrating the relation between all situations that appear in optical flow problems. The situations of augmented intrinsic dimensions due to occlusions are neglected for the sake of clarity.

vector or an occluded aperture problem. In the first case, the flow can be computed reliably, while in the second case it cannot. The exception of occluded regions with lower intrinsic dimension, therefore, interferes with the notion of reliability. Table 5.1 shows different situations occurring in image sequences and the corresponding intrinsic dimensions. As the intrinsic dimension is ambiguous due to occlusions, it is important to detect occlusions in order to resolve this situation. Consequently, situation measures for the detection of the following six situations are compared:

- intrinsic dimension  $\leq 2$  (directed structures),
- intrinsic dimension 2 (temporal directed structures),
- intrinsic dimension  $\leq 1$  (aperture problems),

	image sequence situation
i0d	aperture problem (homogeneous region)
i1d	aperture problem (edge) occluded homogeneous region
i2d	unique translation of corners occluded edge aperture problem
i3d	undirected or transparent structures noise occluded translation of corners

Table 5.1: Typical situations in optical flow problems assigned to intrinsic dimensions

- intrinsic dimension 1 (edge aperture problems),
- intrinsic dimension 0 (homogeneous regions),
- occlusions (increased intrinsic dimension).

There are other situations as well that would be beneficial to detect, for example changes in lighting, which could for example be recognized via heavily averaged difference images.

### 5.1.2 Related Work

Related work on intrinsic dimensions has been described in the corresponding section in Chapter 2.4.

### 5.1.3 Contribution

Situation measures have been proposed before in literature, but no one has come up with a summary and comparison. To fill this gap, I have composed and tested previously proposed situation measures. My contribution is three-fold: First, I propose a classification scheme for known situation measures, which is based on intrinsic dimensionality. Second, new measures that I felt were missing are added. And third, the performance of the most important situation measures is compared.

## 5.2 Classification of Situation Measures

### 5.2.1 Data and Output Functions

To handle the situation measures in a unified way, each of them is subdivided into 1) a data function and 2) an output function. The data function is used to acquire the reliability data, for example the gradient of the image at a certain location. But this

data alone cannot be used as a situation measure for the following two reasons: firstly, it does not range between 0 and 1, so comparability is not given for different measures, and secondly the relation between data and situation is often inverse, which means that a large data value often corresponds to a low probability for the situation. Therefore, we need monotonous, non-negative output functions mapping any kind of data function values to situation values in the interval  $[0, 1]$ . The following four output functions are used to obtain situation measure values for data function values  $d$ :

- (a) If the data function  $d$  is naturally bounded by a fixed range  $[\min, \max]$  (e.g.  $[0, 2\pi]$ ), and there is no inverse relation between data and situation:

$$c_1(d) := \frac{d - \min}{\max - \min}. \quad (5.3)$$

- (b) If the data function  $d$  is naturally bounded by a fixed range  $[\min, \max]$ , and there is an inverse relation between data and situation:

$$c_2(d) := 1 - c_1(d) = \frac{\max - d}{\max - \min}. \quad (5.4)$$

- (c) If the data function  $d$  is not bounded by a fixed range (but non-negative as assumed here), and there is no inverse relation between data and situation:

$$c_3(d) := \frac{d^2}{1 + d^2}. \quad (5.5)$$

- (d) If the data function  $d$  is not bounded by a fixed range, and there is an inverse relation between data and situation:

$$c_4(d) := 1 - c_3(d) = \frac{1}{1 + d^2}. \quad (5.6)$$

### 5.2.2 Principal Currently Known Situation Measures

I now present all principal known situation measures classified according to their intrinsic dimensionality. For each measure a short explanation of the concept behind it and the data and its output function is given.

#### Intrinsic Dimension $\leq 2$ (Directed Structures)

This situation is present for any directed structures at a given location. These structures can be temporal or spatial. Thus, this is the most general of the situations based on the intrinsic dimension. The two following situation measures can be used for its detection.

**structEv3** structEv3 stands for the minimum eigenvalue of the structure tensor (see Chapter 3). It is derived from [51] and is based on the following concept: The nearer to zero the smallest eigenvalue  $\lambda_3$  of the structure tensor is, the more likely exists a main direction of constant intensity within the integration area of the structure tensor. This is the case if movement at constant velocity (0 velocity as well) is present, but dimensions of constant intensity also exist in case of an aperture problem or within homogeneous regions. Hence, this measure is not able to distinguish between temporal directed structures due to movement and spatial directed structures due to aperture problems or homogeneous regions.

$$\begin{aligned} \text{Data function: } & d = \lambda_3 \\ \text{Output function: } & c_4(d) \end{aligned}$$

**structCt** structCt stands for the total coherency measure of the structure tensor. It is based on the same idea as the previous measure structEv3 and is also derived from [51]. The advantage of this measure compared to the structEv3 measure is that it is able to distinguish between temporal directed structures due to motion and spatial directed structures due to homogeneous regions. However, structCt is not able to distinguish motion from edge aperture problems, since in both cases  $\lambda_1 \gg \lambda_3$ . The data function is maximal if  $\lambda_1 \gg \lambda_3 = 0$  (in case of i2d or i1d) and minimal if  $\lambda_1 = \lambda_3$  (in case of i0d or i3d).

$$\begin{aligned} \text{Data function: } & d = \begin{cases} \left(\frac{\lambda_1 - \lambda_3}{\lambda_1 + \lambda_3}\right)^2 & \text{if } \lambda_1 \neq 0, \\ 0 & \text{if } \lambda_1 = 0. \end{cases} \\ \text{Output function: } & c_1(d) \end{aligned}$$

### Intrinsic Dimension 2 (Directed Temporal Structures)

This situation is present if the intrinsic dimension at a given position equals two, which means that a directed structure in temporal direction only or an occluded aperture problem exists. In the first case motion can be estimated reliably. A difficult task for these measures is the distinction between directed structures in temporal and spatial direction. The following measures have been applied to this task.

**structMinors** The structMinors method stands for the structure tensor measure examining the existence of multiple motions in the same location. The basic idea underlying this measure has been mentioned by Barth [12]. I suggest to use the minors of the structure tensor to derive four different expressions for the same motion vector. Ideally, these expressions should be identical in case of a unique motion vector of intrinsic dimension

two, but for other spatio-temporal patterns the calculated motion vectors  $u_1, u_2, u_3$  and  $u_4$  differ. Based on a chosen error measure  $e$  (see section 4.1.2) for the comparison of these vectors different situation measures can be defined.

$$\begin{aligned} \text{Data function: } & d = \sum_{i=1}^3 \sum_{j=i+1}^4 e(u_i, u_j) \\ \text{Output function: } & c_4(d) \end{aligned}$$

**structCc** structCc stands for the corner measure of the structure tensor proposed in [51]. Its data function is defined as the difference between the total coherency measure (structCt) and the spatial coherency measure (structCs) data function. In this way, structCc returns high values in locations, where structCt is large (which is the case for i2d and i1d structures) and structCs is small (which is the case for i3d, i2d and i0d structures). This means that structCc is only large for intrinsic dimension two, that is in case of directed temporal structures. It is bounded by the range  $[0, 1]$ , since  $0 \leq Cs \leq Ct \leq 1$ .

$$\begin{aligned} \text{Data function: } & d := Ct - Cs = \left( \frac{\lambda_1 - \lambda_3}{\lambda_1 + \lambda_3} \right)^2 - \left( \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} \right)^2 \\ \text{Output function: } & c_1(d) \end{aligned}$$

**structMultipleMotion** This measure is based on the structure tensor measure proposed by [75], which examines the existence of multiple motions in the same location. It relies on the assumption that for a reliable motion vector estimate a temporal and no spatial directed structure should exist. For the eigenvalues of the structure tensor  $J$  this means

$$\lambda_1 \geq \lambda_2 \gg \lambda_3 = 0 . \quad (5.7)$$

Therefore, the product  $K = \lambda_1 \lambda_2 \lambda_3$  of the eigenvalues of  $J$  is compared to the average diminished product of the eigenvalues

$$S = \frac{1}{3}(\lambda_2 \lambda_3 + \lambda_1 \lambda_3 + \lambda_1 \lambda_2) . \quad (5.8)$$

Here  $K = 0$  indicates  $\lambda_3 = 0$  and, thus, if any directed temporal or spatial structure exists. In this case we have intrinsic dimension two, one or zero. In contrast to that  $S = 0$  means for the two smallest eigenvalues  $\lambda_3 = \lambda_2 = 0$  and, thus, indicates an aperture problem of intrinsic dimension one or zero. Therefore, a reliable motion vector of intrinsic dimension two can be identified by a small value of  $K$  and a large value of  $S$  at the same time. To adjust scales  $\sqrt[3]{K}$  is compared to  $\sqrt{S}$ . Hence in the case of intrinsic dimension two it follows  $\sqrt[3]{K} \ll \sqrt{S}$ .

$$\begin{aligned} \text{Data function: } & d = \sqrt{S} - \sqrt[3]{K} \\ \text{Output function: } & c_3(d) \end{aligned}$$

### Intrinsic Dimension $\leq 1$ (Aperture Problems)

In the situation of an aperture problem (edge or homogeneous region), which is defined by an intrinsic dimension smaller or equal to 1, the local context is not sufficient to calculate a unique displacement vector. An example is a long edge of a rectangle moving downwards, since here from a local point of view, any flow vector with the correct vertical and an arbitrary horizontal component would be correct. In the case of an edge only the component orthogonal to the edge that causes the aperture problem can be estimated reliably, whereas in the case of homogeneous regions any flow vector is possible. To detect such situations the following situation measures can be used.

**detHessian, evHessian, condHessian** These measures stand for different Hesse matrix measurements: the determinant, the smallest eigenvalue and the condition number. The Hesse matrix  $H$  is defined as

$$H = \begin{pmatrix} \partial_{xx}I & \partial_{xy}I \\ \partial_{yx}I & \partial_{yy}I \end{pmatrix}. \quad (5.9)$$

Its condition number has been proposed in [97], but in [10] the determinant of the Hesse matrix has been found more reliable. The determinant can be expressed as the product of the eigenvalues  $\gamma_1$  and  $\gamma_2$  of  $H$

$$\det(H) = \gamma_1 \gamma_2. \quad (5.10)$$

Since the curvature of a function can be approximated by the second derivative, the entries of the Hesse matrix describe the curvature of the sequence in different directions. This measure can be used especially for the identification of homogeneous regions and the aperture problem:  $H$  is a symmetric positive definite matrix. Therefore, its eigenvectors are orthonormal and form a basis of  $\mathbb{R}^2$ . The curvature in a certain direction  $\mathbf{x}$  can be computed as  $\mathbf{x}^T H \mathbf{x}$ . If  $\mathbf{v}_i, i \in \{1, 2\}$ , is a normalized eigenvector of  $H$  with corresponding eigenvalue  $\gamma_i$ , then the curvature along this eigenvector can be computed as follows

$$\mathbf{v}_i^T H \mathbf{v}_i = \mathbf{v}_i^T \gamma_i \mathbf{v}_i = \gamma_i. \quad (5.11)$$

For this reason the eigenvalues  $\gamma_i, i \in \{1, 2\}$ , are equal to the curvature along the main axes, the eigenvectors. Thus, aperture problems and homogeneous regions amount to  $\det(H)=0$ . For these reasons two different data functions for the recognition of intrinsic dimensions one and zero can be defined. One data function is based on  $\gamma_2$ , whereas the other is based on the determinant of  $H$ .

$$\begin{aligned} \text{Data function: } & d_1 := \gamma_2, \quad d_2 := \det(H) \\ \text{Output function: } & c_4(d) \end{aligned}$$

**LOGHessian** LOGHessian stands for the measure examining the curvature of the edge map of the image sequence. It has been proposed by Waxman, Wu and Bergholm in [98]. The idea is similar to the detHessian measure. But instead of using the image sequence directly, the authors use an edge map  $E$  of the image sequence convolved with a spatio-temporal Gaussian kernel called “activation profile”  $A$ . This edge map is computed by a convolution of a DOG filter with the sequence and the identification of the zero-crossings  $z$

$$E = z(DOG * I), \quad (5.12)$$

$$A = G(\sigma_x, \sigma_y, \sigma_t) * E. \quad (5.13)$$

Then instead of the Hesse matrix of the image sequence the Hesse matrix of the activation profile is computed and its determinant is used as data function.

$$\begin{aligned} \text{Data function: } & d = \det \begin{pmatrix} \partial_{xx} A & \partial_{xy} A \\ \partial_{yx} A & \partial_{yy} A \end{pmatrix} \\ \text{Output function: } & c_4(d) \end{aligned}$$

**SSDSurface** This measurement is based on Anandan’s proposal [3] to examine the SSD (sum of squared differences) surface. It is created by the repeated modification of the flow vector at the current position and the computation of a new SSD value each time. These SSD values at different modified locations make up the surface. If the minimum SSD value of the surface,  $S_{min}$ , is rather high, no good match in the SSD sense exists for the current vector. To detect aperture problems, the curvature of the surface along the maximum and the minimum principal axis,  $C_{max}$  and  $C_{min}$ , is computed. In homogeneous regions the curvature is low leading to small values for  $C_{max}$  and  $C_{min}$ . At edges the curvature along the maximum principal axis is high, whereas that along the minimum principal axis is low. Anandan defines the two measurements  $g_1$  and  $g_2$  in order to quantify these values:

$$g_1 := \frac{C_{max}}{k_1 + k_2 S_{min} + k_3 C_{max}}, \quad (5.14)$$

$$g_2 := \frac{C_{min}}{k_1 + k_2 S_{min} + k_3 C_{max}}, \quad (5.15)$$

$$(5.16)$$

where  $k_1, k_2$  and  $k_3$  are constants.  $k_1$  prevents 0 in the denominator,  $k_2$  determines the punishment for high values of  $S_{min}$ , and  $k_3$  bounds the result to the range  $(0, \frac{1}{k_3})$ .

Following Anandan the parameters are chosen as follows:  $k_1 = 150$ ,  $k_2 = 1$  and  $k_3 = 0$ . The final measurement data function  $d$  is defined as the product of  $g1$  and  $g2$ .

$$\begin{aligned} \text{Data function: } & d := g_1 \ g_2 \\ \text{Output function: } & c_3(d) \end{aligned}$$

**SinghSurface** SinghSurface stands for the measure examining the velocity distribution at each position in the sequence. The measure has been proposed by Singh and can be found in [10]. It is based on a two-stage computation of SSD values using the previous and following frame combined with the displacement field in positive and negative direction:

$$\begin{aligned} SSD_s(x, y, t) = & \quad (5.17) \\ (I(x + u_1, y + u_2, t + 1) - I(x, y, t))^2 + \\ (I(x - u_1, y - u_2, t - 1) - I(x, y, t))^2. \end{aligned}$$

In this way, spurious minima due to noise or periodic texture are averaged out. The idea behind this measure is to calculate the  $SSD_s$  surface for varying integer displacements for each motion vector  $\mathbf{u}$ . The resulting surface is then converted to a probability distribution by

$$R(\mathbf{u}_1 + i, \mathbf{u}_2 + j) = \exp(-k \ SSD_s(\mathbf{u}_1 + i, \mathbf{u}_2 + j)), \quad (5.18)$$

where  $k = -\frac{\ln(0.95)}{\min(SSD_s(\mathbf{u}))}$ . From this distribution the velocity  $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2)$  can be obtained as its mean

$$\mathbf{v}_1 = \frac{\sum_{i,j=1}^n R(\mathbf{u}_1 + i, \mathbf{u}_2 + j)(\mathbf{u}_1 + i)}{\sum_{i,j=1}^n R(\mathbf{u}_1 + i, \mathbf{u}_2 + j)}, \quad (5.19)$$

$$\mathbf{v}_2 = \frac{\sum_{i,j=1}^n R(\mathbf{u}_1 + i, \mathbf{u}_2 + j)(\mathbf{u}_2 + j)}{\sum_{i,j=1}^n R(\mathbf{u}_1 + i, \mathbf{u}_2 + j)}. \quad (5.20)$$

This only works well if the distribution is nearly symmetrical about the true velocity and has few maxima. Therefore, the eigenvalues of the covariance matrix of this distribution are examined. If both eigenvalues are large, we have a homogeneous region, since many displacements lead to high probabilities. If one eigenvalue is large and one small, we have an aperture problem, since the probability is high in many places along the axis corresponding to the larger eigenvalue. If both eigenvalues are small, we have a corner, where the displacement field can be computed reliably. Let  $\lambda_1$  denote the larger eigenvalue. Only if  $\lambda_1$  is small, both eigenvalues are small and the measure is reliable.

$$\begin{aligned} \text{Data function: } & d := \lambda_1 \\ \text{Output function: } & c_4(d) \end{aligned}$$

**structCc** As explained before this measure indicates the existence of a situation of intrinsic dimension two. By exchanging the output function to  $c_2(d)$  the measure indicates the existence of situations of intrinsic dimension two and one. In case of intrinsic dimension three the result can also be large leading to incorrect statements.

### Intrinsic Dimension 1 (Edge Aperture Problems)

Edges pose problems for flow computation methods, since they cause an aperture problem. Here, the motion vector is no longer uniquely defined from a local point of view and, thus, cannot be estimated reliably. Only one measure is known, which distinguishes between aperture problems caused by edges and those caused by homogeneous regions.

**structCs** structCs stands for the spatial coherency measure of the structure tensor mentioned in [51]. If we have an aperture problem and assume the brightness constancy equation holds, there are two directions of constant gray values: the temporal direction and the direction along the object that causes the aperture problem. Therefore, the two smallest eigenvalues  $\lambda_2 \geq \lambda_3$  of the structure tensor are equal to 0. This property can be measured by the spatial coherency measure data function structCs. It reaches its maximum for intrinsic dimension 1 (since  $\lambda_1 \gg \lambda_2 = \lambda_3 = 0$ ). For all other types of motion it is smaller than 1 and, thus, bounded by  $[0, 1]$ .

$$\begin{aligned} \text{Data function: } & d = \begin{cases} \left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\right)^2 & \text{if } \lambda_1 \neq 0, \\ 0 & \text{if } \lambda_1 = 0 \end{cases} \\ \text{Output function: } & c_1(d) \end{aligned}$$

### Intrinsic Dimension 0 (Homogeneous Regions)

Homogeneous regions pose problems for many flow computation methods due to the lack of image structure. From a local point of view any displacement vector can be correct in these situations. The following measures can be used for their detection.

**grad** The idea behind the gradient measurement is that the displacement field can be computed the more reliably the more texture is contained in the image. There are different ways to compute the image gradient. Here the central differences scheme (grad) and the forward differences scheme (gradFD) is employed.

$$\begin{aligned} \text{Data function: } & d := \nabla_2 I \\ \text{Output function: } & c_4(d) \end{aligned}$$

**structTrace** structTrace stands for the trace of the structure tensor  $J$  (see Chapter 3). In homogeneous regions all image gradients tend to 0. The same applies to the structure tensor, its eigenvalues and, thus, the sum of its eigenvalues. Since the trace of a matrix is invariant under coordinate transformations, the sum of the eigenvalues of the structure tensor equals its trace.

$$\begin{aligned} \text{Data function: } & d = \text{trace}(J) \\ \text{Output function: } & c_4(d) \end{aligned}$$

### Occlusion Detection

Occlusion situations pose difficulties for flow computation methods, since they contain pixels with undefined flow vectors. This situation is especially important for the detection of higher intrinsic dimensions caused by occlusion.

Many investigations have been conducted in the field of occlusion detection. However, these methods are not applicable to situations, where only a monocular image sequence and a flow field is given. Several approaches use stereo images and disparity maps in order to derive occlusion information [102]. Other techniques employ initialization images and compute difference images to detect occlusions [71]. There are also flow computation methods that integrate occlusion detection into global energy terms [1, 24]. Furthermore, there are several statistical approaches [69, 68] that are not considered here due to complexity and variety.

In my opinion, situation measures do not have to be based on the image sequence necessarily, but can just as well derive their information only from the flow field. Therefore, I suggest the use of regularizers of global optical flow computation methods as situation measures. Regularizers can be used to identify locations, where the flow field does not correspond to the regularization model. In this way, for example occlusions could be detected. Here four classes of regularizers (image-driven, flow-driven, isotropic and anisotropic) defined in [99] are examined plus the homogeneous regularizer used in [53], a space-time regularizer defined in [28] and a total variation regularizer proposed in [88]. I also propose a new type of regularizer I found missing in this collection: a purely temporal regularizer.

To detect occlusion situations the following measures can be used.

**homReg** homReg stands for the homogeneous regularizer. It has been used in [53] and examines the smoothness of the flow field  $\mathbf{u}$  by computing its spatial gradient.

$$\begin{aligned} \text{Data function: } & d := \|\nabla_{\{x,y\}} \mathbf{u}_1\|_{l_2}^2 + \|\nabla_{\{x,y\}} \mathbf{u}_2\|_{l_2}^2 \\ \text{Output function: } & c_3(d) \end{aligned}$$

**isoFlowReg** isoFlowReg stands for the isotropic flow-driven regularizer. It was proposed in [99] and examines the smoothness of the flow field, but allows for exceptions at flow edges. Let  $\psi(s^2)$  be a differentiable and increasing function that is convex in  $s$ , e.g.

$$\psi(s^2) = \epsilon s^2 + (1 - \epsilon) \lambda^2 \sqrt{1 + \frac{s^2}{\lambda^2}} \quad (5.21)$$

mentioned in [99] with  $\lambda = \epsilon = 0.1$ .

$$\begin{aligned} \text{Data function: } & d := \psi(\|\nabla_{\{x,y\}} \mathbf{u}_1\|_{l_2}^2 + \|\nabla_{\{x,y\}} \mathbf{u}_2\|_{l_2}^2) \\ \text{Output function: } & c_3(d) \end{aligned}$$

**tvReg** tvReg stands for the total variation regularizer. It was originally proposed by Rudin et al. in [88] and is a special case of the isotropic flow-driven regularizer with

$$\psi(s^2) = \sqrt{s^2 + \epsilon^2}. \quad (5.22)$$

**anisoFlowReg** anisoFlowReg stands for the anisotropic flow-driven regularizer presented in [99]. It also examines the smoothness of the flow field except at flow edges. Here it only assumes smoothness along the edge, whereas no smoothness is assumed across the edge. Let  $\phi$  be a matrix valued function, which uses the function  $\psi$  defined as in isoFlowReg:

$$\phi(U) := \sum_{i=1}^2 \psi(\lambda_i) \mathbf{v}_i \mathbf{v}_i^T. \quad (5.23)$$

Here  $U$  is a symmetric positive semidefinite matrix. Thus, it has two orthonormal eigenvectors  $\mathbf{v}_i, i \in \{1, 2\}$ , with corresponding eigenvalues  $\lambda_i$ . Due to the special choice of  $U$  in the data function, the eigenvalues specify the contrast of the image in the directions  $\mathbf{v}_1$  and  $\mathbf{v}_2$ , respectively.

$$\begin{aligned} \text{Data function: } & d := \text{trace}(\phi(\nabla_{\{x,y\}} \mathbf{u}_1 \nabla_{\{x,y\}} \mathbf{u}_1^T + \nabla_{\{x,y\}} \mathbf{u}_2 \nabla_{\{x,y\}} \mathbf{u}_2^T)) \\ \text{Output function: } & c_3(d) \end{aligned}$$

**timeReg** Among the variety of regularizers I found missing a purely temporal regularizer, which assumes only temporal smoothness of the flow field at each pixel. Thus, it may be well suited for the detection of occlusions.

$$\begin{aligned} \text{Data function: } & d := \|\nabla_t \mathbf{u}_1\|_{l^2}^2 + \|\nabla_t \mathbf{u}_2\|_{l^2}^2 \\ \text{Output function: } & c_3(d) \end{aligned}$$

**spaceTimeReg** spaceTimeReg was proposed in [28] and assumes temporal and spatial smoothness of the flow field at each pixel.

$$\begin{aligned} \text{Data function: } & d := \|\nabla_{\{x,y,t\}} \mathbf{u}_1\|_{l^2}^2 + \|\nabla_{\{x,y,t\}} \mathbf{u}_2\|_{l^2}^2 \\ \text{Output function: } & c_3(d) \end{aligned}$$

**structMinors** In case of occlusions the four different motion vectors computed from the minors of the structure tensor will differ. Thus, occlusions can be detected by high errors between these estimates. The measure has been described before and can be found in the section on directed temporal structures. To detect occlusions, the output function has to be changed to  $c_3(d)$ .

## 5.3 Experiments and Results

### 5.3.1 Comparison Technique

To examine the performance of the situation measures an artificial sequence (Figure 5.2) containing various intrinsic dimensions in combination with occlusion has been generated. It consists of four parts:

- (a) noise as example for undirected structures (i3d),
- (b) a moving two-dimensional sine as example for temporal directed structures (i2d),
- (c) a moving one-dimensional sine as example for an edge aperture problem (i1d),
- (d) a homogeneous region (i0d).

To evaluate the effect of occlusions of different intrinsic dimensions on all measures the lower half of the sequence is occluded by a sine pattern in the following frame (see Figure 5.2). Thus, we obtain four regions for each intrinsic dimension, each of which consists of an occluded and a non-occluded region. To test the situation measures the detection accuracy of the corresponding intrinsic dimensions is evaluated. According to the situation and the dimension of the image region examined by the measures each of the eight areas in Figure 5.2 is either to be detected as belonging to the situation or not. Measures not using temporal information are not influenced by occlusion in the next frame and, thus, have the same ground truth values for the upper and lower half of the sequence.

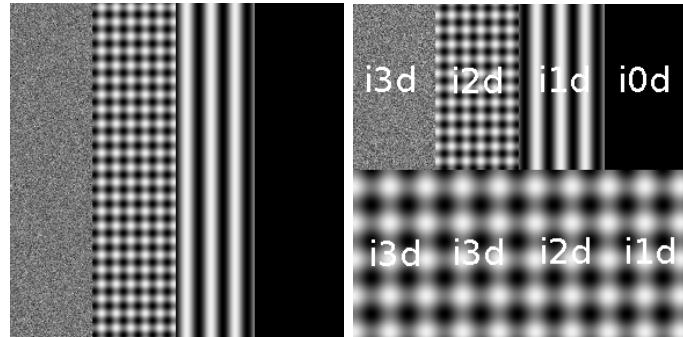


Figure 5.2: First and second frame of the artificial test sequence for the comparison of all situation measures. The lower half is occluded in the second frame, which increases the original intrinsic dimension by one in these regions.

To evaluate the performance of a given situation measure  $S$  is used as the set of pixels within the situation,  $T$  as the set of pixels outside, that means in all other situations, and  $g_i \in \{0, 1\}$  as the ground truth value at pixel  $i$ . This value  $g_i$  is set to 0 for pixels in  $S$  and to 1 for pixels in  $T$ . Now the quality of the measures is expressed in two characteristic values: the average identification error within the situation (inSit) and that outside the situation (outSit)

$$\text{inSit} = \frac{\sum_{i \in S} |c_i - g_i|}{|S|}, \quad (5.24)$$

$$\text{outSit} = \frac{\sum_{i \in T} |c_i - g_i|}{|T|}, \quad (5.25)$$

where  $c_i \in [0, 1]$  denotes the result of the situation measure at pixel  $i$ . Now different measures can be compared based on the sum of these values. The effect of image noise on the quality of the measures will be especially considered in the evaluation and final decision on the best measure. In each Figure the horizontal axis shows the current image noise level and the vertical axis shows the sum of the error measures inSit and outSit for the compared situation measures.

### 5.3.2 Results for Intrinsic Dimensions $\leq 2$ (Directed Structures)

For this situation both considered situation measures yield comparable, moderate results. The results of the structEv3 method are slightly better than those of the structCt measure. The methods strongly depend on the size of the integration area of the structure tensor and the size of the filter mask for the derivatives. Both measures perform only moderately due to the fact that large parts of the noise (i3d) are recognized as directed structures as well, which leads to a high error outside the situation. The slightly

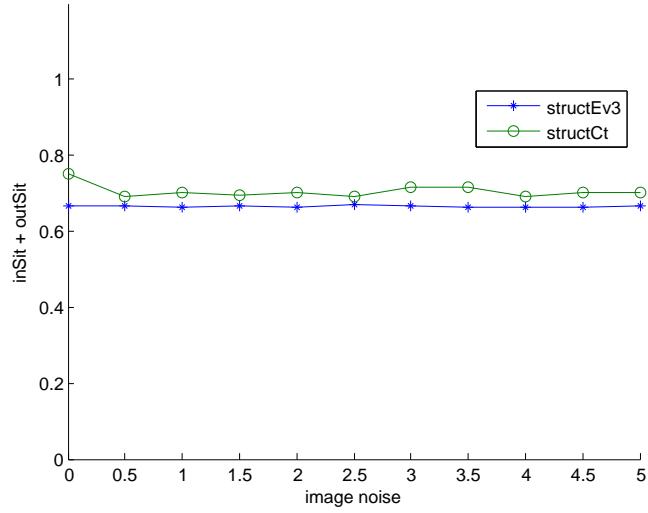


Figure 5.3: Dependency of the directed structures situation measures on image noise.

lower quality of the structCt measure is due to the reason that this measure is not able to detect homogeneous regions as directed structures and, thus, suffers from a considerable error within the situation. Both methods are robust to noise as shown in Figure 5.3.

### 5.3.3 Results for Intrinsic Dimension 2 (Directed Temporal Structures)

In this situation only directed temporal structures, that means the two-dimensional sine part, should be recognized by the measures specializing on this situation. As all measures also use temporal information, occluded edge aperture problems have intrinsic dimension two as well and, thus, are also part of the i2d situation. For the structMinors measure the amplitude error measure (structMinorsAmplitude), the angle error measure (structMinorsAngle) and the angular error measure (structMinorsAngular) defined in section 4.1.2 are chosen for the comparison of the four computed flow vectors.

The structMultipleMotion and the structMinorsAmplitude measure yield good results, whereas the results of the structMinorsAngle and the structMinorsAngular measure show high errors outside the situation. Figure 5.4 shows that the structMinorsAmplitude method is absolutely non-robust towards noise and already yields high errors for a noise level of  $\sigma = 0.5$  despite the integration area of the structure tensor. In contrast to that the other measures are much less susceptible to noise. Therefore, one would favor the structMultipleMotion measure for this situation.

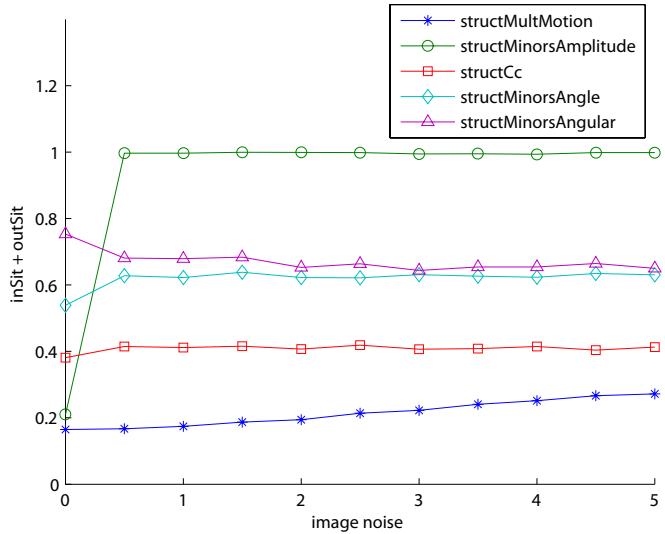


Figure 5.4: Dependency of the directed temporal structures situation measures on image noise.

### 5.3.4 Results for Intrinsic Dimension $\leq 1$ (Aperture Problems)

For this situation most aperture problem measures yield almost optimal recognition rates within the situation, so here the average error outside the situation determines the quality of the measures. The dependency on image noise is very interesting for these measures, since many which show very good results if no noise is present already deteriorate dramatically for small noise levels. All of these measures are tested with different noise levels of  $\sigma \in [0, 5]$  standard deviation. The results in Figure 5.5 show that measures using second image derivatives, such as the smallest eigenvalue of the Hessian, are highly dependent on the image noise level. In contrast, the logHessian measure is rather noise resistant from smaller noise levels of  $\sigma = 0.5$  on due to the convolution of the edge map with a spatio-temporal Gaussian filter. The ssdSurface and the singhSurface are rather robust to noise as well, which is probably due to the reason that their decision is based on a larger set of measurements.

I come to the conclusion that one would find the detHessian measure preferable.

### 5.3.5 Results for Intrinsic Dimension 1 (Edge Aperture Problems)

For the situation of edge aperture problems just one situation measure, structCs, is known. Figure 5.6 shows that the structCs measure yields rather good results and is not influenced by the noise levels tested.

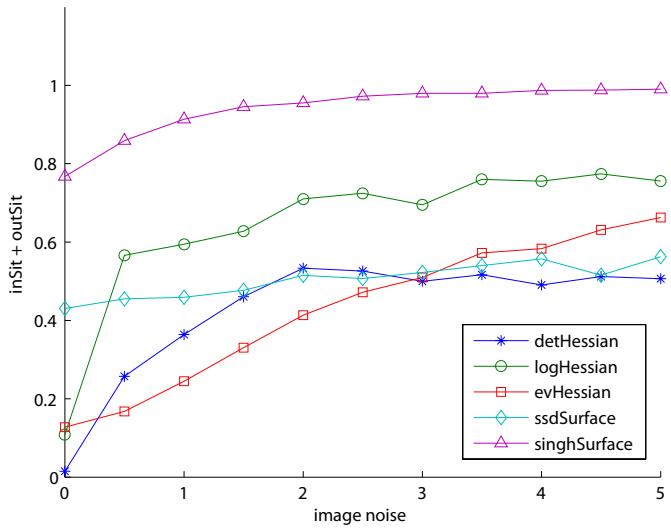


Figure 5.5: Dependency of the aperture problem measures on image noise.

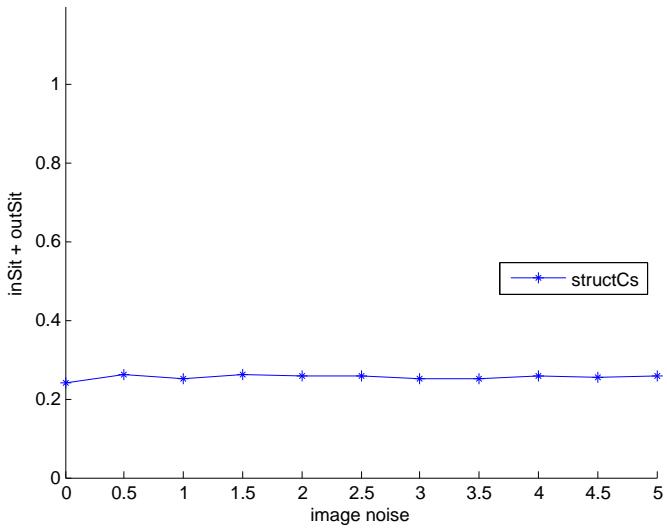


Figure 5.6: Dependency of the edge aperture problem measure structCs on image noise.

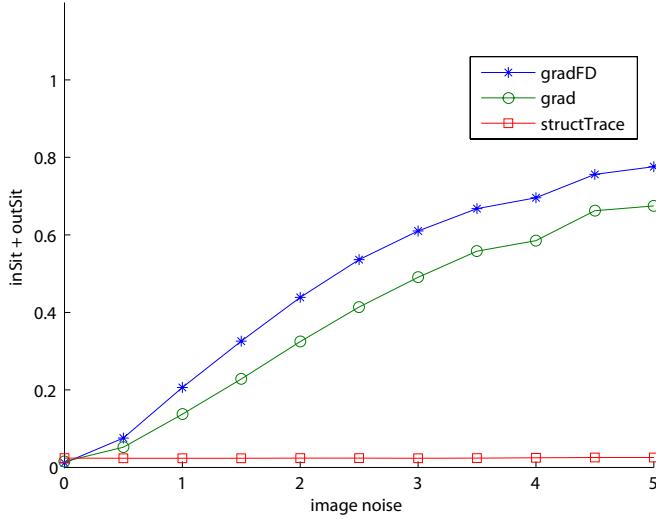


Figure 5.7: Dependency of the homogeneous regions measures on image noise.

### 5.3.6 Results for Intrinsic Dimension 0 (Homogeneous Regions)

The measures detecting homogeneous regions are compared in Figure 5.7. The curves show that measures using image derivatives, the gradient measures, are a lot more susceptible to noise than the structTrace method, which relies on the structure tensor with its larger integration scale. The gradient measures yield good results if no noise is present, but already for a noise level of  $\sigma = 0.5$  the structTrace method is superior. The grad measure performs slightly superior to the gradFD measure, since it is based on the central differences scheme, which averages two forward differences gradients and, thus, makes the result less prone to noise. In contrast to the gradient measures the structTrace measure remains stable for all higher noise levels and, thus, easily outperforms the other two measures. Therefore, one would favor the structTrace method for this situation.

### 5.3.7 Results for Occlusion

The measures detecting occlusions should be able to recognize the lower half of the four regions of the test sequence. Here the distinction into four regions is interesting as the consequences of different occluded intrinsic dimensions can be examined. Since the regularizers depend on a computed flow field two different flow fields are used: one field computed by the structure tensor method [16] and one field computed by the Horn-Schunck method [53].

For both flow computation methods the results are quite different. For the flow field calculated by the Horn-Schunck method the timeReg measure yields the best results,

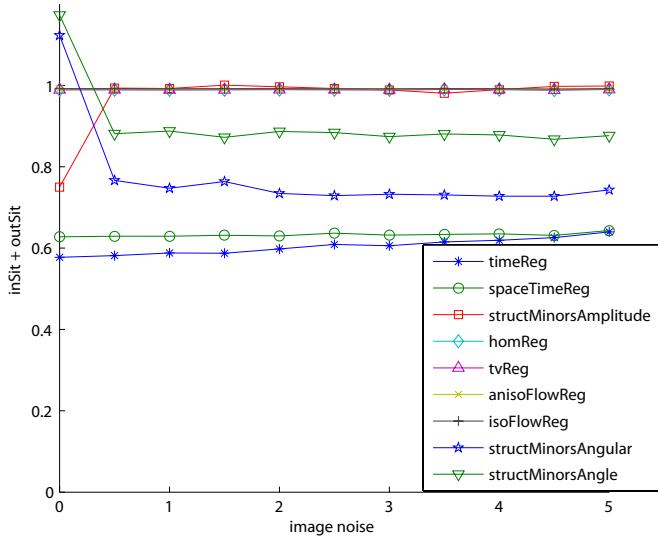


Figure 5.8: Dependency of the occlusion measures on image noise for a flow field computed by the Horn-Schunck method.

which are also rather robust to noise. In contrast to that the structMinorsAngular method yields the best results for the structure tensor flow field. The influence of noise on the measures applied to the Horn-Schunck method is presented in Figure 5.8. For the structure tensor method the regularizer results are very similar, only about 0.3 higher. The diagram shows that the timeReg measure as well as the structMinorsAngular measure are rather robust to noise. Therefore, one would favor the timeReg measure for Horn-Schunck flow fields and the structMinorsAngular measure for structure tensor flow fields. However, due to the high error rates none of these measures is really effective.

### 5.3.8 Application to Real-World sequences

To test the presented situation measures on real-world-scenes, they are applied to the Marble sequence. The question I seek to answer is, whether the situation measures that perform best on artificial sequences also yield good results on noisy real-world-sequences. The Marble sequence (Figure 5.9a) is chosen, since it contains many different situations such as aperture problems of intrinsic dimension 1 (diagonal line on the table, parts of the flagstone edges) and 0 (background, parts of the marble blocks), directed temporal structures (the main part of the image) and occlusion (at the edges of the marble blocks).

Figure 5.9 b) shows the result of the best measure for the examination of directed structures (structEv3) applied to the Marble sequence. Temporal directed structures

as well as aperture problems, such as the edges of the marble blocks and homogeneous regions in the background, are recognized well. Occluded regions of intrinsic dimension two, thus becoming intrinsic dimension three at the edges of the marble blocks, are partially recognized as non-directed structures. However, several pixels of the table pattern in the foreground are not recognized as directed structures even though a unique motion vector exists.

The application of the best measure for the recognition of directed temporal structures, `structMultipleMotion`, to the Marble sequence is shown in Figure 5.9 c). The measure seems to work quite well for real-world sequences, too, as aperture problems (e.g. the diagonal line on the table, parts of the flagstone boundaries) and homogeneous regions (the background, larger patterns on the table) are detected as situations, where no unique direction of motion exists. However, larger parts of the marble blocks with mainly homogeneous texture pose problems for this measure. In contrast, the parts of the block texture, where darker structures appear, are detected correctly.

For the recognition of aperture problems the `detHessian` measure showed the best results if noise was present in the scene. For real-world-sequences the measure is very noisy. The result can be seen in Figure 5.9 d). It recognizes aperture problems such as the diagonal line on the table, the borders of the flagstones, the background and even some structures on the table. These structures are detected as well, because they form small but homogeneous regions. However, like the `structMultipleMotion` measure, this measure also detects large parts of the marble blocks as aperture problems, which is due to the homogeneity of the block texture. Hence, we can see that several measures depend on well-textured regions for correct decisions.

The application of the only measure for the recognition of edge aperture problems of intrinsic dimension one, the `structCs` measure, to the Marble sequence yields the result in Figure 5.9 e). We can see that the measure is well able to detect regions with edge aperture problems, especially the flagstone border regions, the diagonal table line, edges of the blocks and the table edge. However, false positives appear for small structures on the table, which locally appear as aperture problems.

Figure 5.9 f) shows the result of the application of the best situation measure for the detection of homogeneous regions (`structTrace`) to the Marble sequence. We can see that the measure is well suited even for real-world-sequences, since all larger homogeneous regions are detected. Here as well large parts of the marble block texture are classified as homogeneous. The undetected border of some homogeneous regions is due to the reason that the minimum distance from the edge of a homogeneous region for the current pixel to be detected is determined by the integration area of the structure tensor. This also affects the minimum size of a homogeneous region below which it cannot be detected. For the occlusion sequence the `timeReg` measure is the only one to yield acceptable results provided an artificial sequence and the Horn Schunck method are used to calculate the flow field. However, the application to the Marble sequence shows that in fact none of the occlusion situations is detected. A similar statement accounts for the best measure

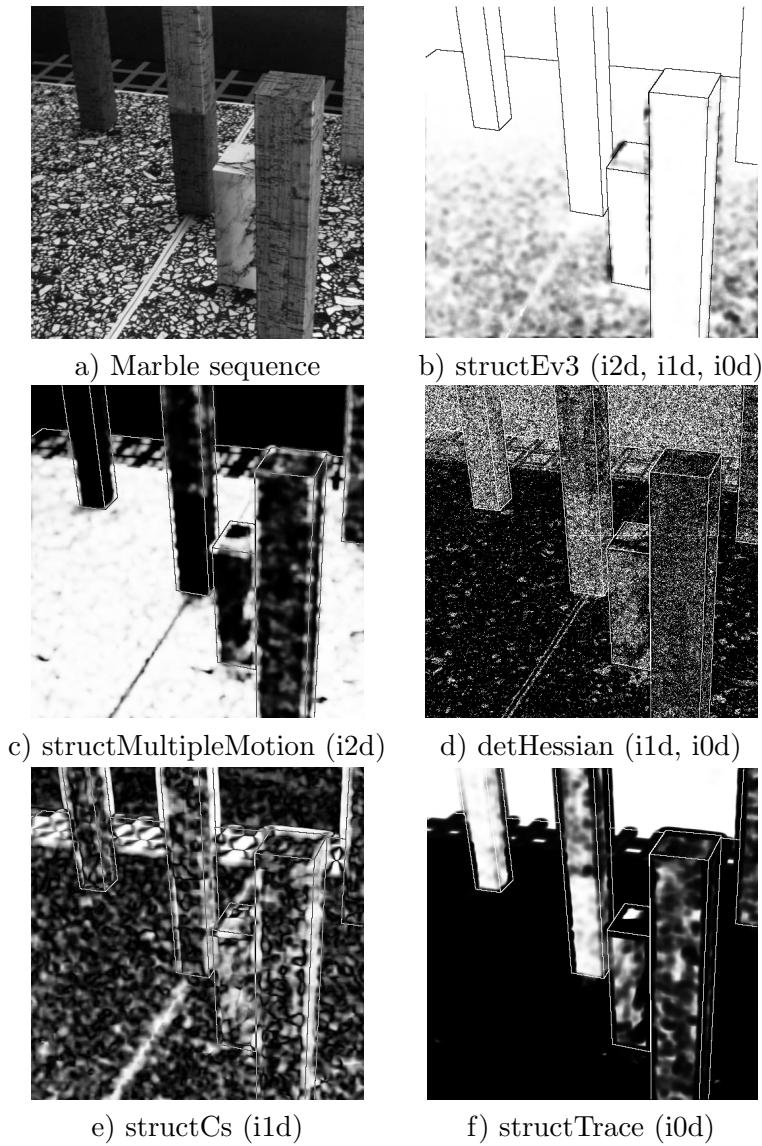


Figure 5.9: Best situation measures for the detection of the compared situations applied to the real-world Marble sequence.

for structure tensor fields, the structMinorsAngular measure. Here several of the marble blocks, the background and smaller parts of the table are recognized as occluded. Real occlusion situations are partly recognized, but by far the largest part of the detections does not correspond to occlusions.

## 5.4 Summary and Conclusion

In this chapter I have summarized and examined typical situation measures which are used to identify image sequence locations, where a reliable optical flow estimation is difficult or impossible. For each situation a measure was identified that yields the best results and is still robust in the presence of noise. These measures are shown in Table 5.2.

id	situation	best measure
i2d - i0d	directed structures	structEv3
i2d	temp. directed structures	structMultipleMotion
i1d - i0d	general aperture problems	detHessian
i1d	edge aperture problems	structCs
i0d	homogeneous regions	structTrace
i+1d	occlusion (HS-field)	timeReg
i+1d	occlusion (ST-field)	structMinorsAngular

Table 5.2: Best situation measure for each situation based on its intrinsic dimension.

The quality of the presented situation measures is different. If no noise is present in the scene, there is always a measure available that yields good or at least moderate results. In case of higher noise levels the results show that larger integration scales like those of the structure tensor or smoothing as is applied by the logHessian measure partially remove the influence of noise on the measures and make them robust. In contrast, measures based on unsmoothed image gradients become unreliable for small noise scales already. Therefore, for almost all of the situations a measure of the structure tensor has finally been chosen with respect to noise robustness. However, the application to the Marble sequence shows that several of the measures are not able to cope with real-world situations.



# Chapter 6

---

## Surface Measures

### 6.1 Introduction

In the previous chapter I have presented and compared the most important situation measures classified according to their intrinsic dimensionality. Yet, since none of these measures accounts for the computed flow field none of them is, in fact, qualified to estimate the accuracy of optical flow vectors. Instead of examining the intrinsic dimension of the image sequence we, therefore, propose to examine the intrinsic dimension of the energy surface of optical flow estimators based on an arbitrary number of parameters. By means of surface functions I was able to simultaneously derive a situation and a confidence measure for optical flows. I show that based on the information of the proposed so-called “surface measures” the average error of optical flow fields can be significantly reduced by a basic motion restoration algorithm.

#### 6.1.1 Motivation

In general, all global and local optical flow computation methods can be formulated as a parameter optimization problem, which consists of the minimization of an energy. This is also true for many other image processing problems such as registration, segmentation, image restoration and reconstruction. Global methods minimize an energy consisting of a data term ensuring brightness or gradient constancy etc. along the flow trajectory and a regularization term, which enforces the smoothness of the resulting flow field in order to obtain unique solutions [53, 25, 28, 78]. Local methods employ energy optimization by means of e.g. least squares or total least squares estimators [70, 16]. In several cases additional parameters such as intensity changes [20, 50] or various model parameters [78] are estimated as well. The aim of such energy optimization problems is to find parameters fulfilling the requirements expressed in the energy formulation as good as possible. Yet, despite recent progress in optical flow approaches, the algorithms

are still facing difficult problems, which lead to errors in the resulting flow fields. Hence, if an optimum of the parameter optimization problem has been found by an optical flow algorithm, the question of the quality of this optimum and the corresponding optical flow parameters still remains.

Based on the intrinsic dimension of surface functions a new confidence and a new situation measure are simultaneously proposed. The only difference between both measures is the underlying flow field. In case a computed flow field is used we obtain a confidence measure, which evaluates the accuracy of the flow vectors. In case a zero flow field is used we obtain a situation measure, which evaluates the feasibility of accurate flow estimation.

### 6.1.2 Related Work

Related work on intrinsic dimensions has been introduced in chapter 2.4.

To estimate the intrinsic dimension of the energy of the parameter optimization problem I propose surface functions. The notion surface function refers to a generalization of correlation surfaces to arbitrary energy functions. Correlation surfaces have widely been used for the detection of features in images. In terms of motion estimation, they have been applied to measure the similarity between regions in subsequent image frames in order to estimate the displacement between moving objects. To this end, a similarity measure relating different image intensities for alternative locations in the sequence is computed. Typical similarity measures are for example the sum of squared differences or the normalized cross correlation. Correlation surfaces have for example been applied by Anandan in optical flow computation [3], who proposed a confidence measure to detect aperture problems in order to find the optimal scale for his hierarchical block matching algorithm for the computation of the optical flow, by Rosenberg and Werman [84] to detect locations where motion cannot be represented by a Gaussian random variable and by Irani and Anandan in [54] to align images obtained from different sensors. Due to the dependence on image intensities, a drawback of correlation surfaces is their susceptibility to image noise. This can be overcome by choosing large correlation regions, which in turn leads to problems caused by occlusions and motion boundaries.

Surface functions differ from correlation surfaces in one main aspect: They do not operate on the image, but on energies derived from parameter optimization methods. Hence, they are not limited to simple intensity based image similarity measures but depend on all parameters occurring in the optimization problem, such as the horizontal and vertical flow component and parameters estimating brightness changes etc.

I want to analyze the intrinsic dimension of energy surfaces. So far, in image processing the intrinsic dimension of image sequences has naturally been restricted to equal to or smaller than two [38] or three [11]. As we are dealing with energy surfaces and an arbitrarily large set of parameters, the intrinsic dimension of the surface functions can be arbitrarily large. Furthermore, as has been stated in [38], in fact for images and

energies there is no situation that is exclusively of one intrinsic dimension. Instead, a probability can be assigned to each situation for it to be of a given intrinsic dimension. This led to a continuous formulation of the intrinsic dimension within the topology of a cone [38]. Therefore, to be able to apply the theory of intrinsic dimensions to the case of an arbitrary number  $d$  of parameters, a generalized continuous formulation of the intrinsic dimension up to dimension  $d$  is required.

### 6.1.3 Contribution

In previous approaches (see Chapter 5) only the intrinsic dimension of the image sequence has been used. To apply the theory of intrinsic dimensions to energy surface functions depending on an arbitrary number of parameters, I give a generalized continuous formulation of the intrinsic dimension up to dimension  $d$ , which can be represented as a simplex. In addition to the intrinsic dimension of the energy surface function my formulation also comes with a simple method to detect outliers. For these, either the energy could not be optimized at all or the optimization process got stuck in a local optimum. From both sources of information, the intrinsic dimensionality of the energy surface function and the outlier detection, a new confidence measure can be derived. In case a zero motion field instead of a computed motion field is used, we, in fact, obtain a situation measure, which allows for statements on the feasibility of accurate optical flow estimation. Part of this work was published in [62].

## 6.2 Surface Measures

### 6.2.1 Energy Functions

The computation of optical flow fields usually amounts to the solution of energy minimization problems. Based on arbitrary energy formulations I will now derive surface functions. Such energy formulations will be described by mappings  $c$  from the image domain, the image sequence and a parameter vector from the  $d$ -dimensional parameter space  $\mathbb{R}^d$  to the set of non-negative real numbers:

$$c : D \times I \times \mathbb{R}^d \rightarrow \mathbb{R}_0^+. \quad (6.1)$$

The parameter vector  $\mathbf{u} \in \mathbb{R}^d$  usually consists of the horizontal and vertical flow field component together with any additional parameters in the flow computation problem. The set of such energy formulations will be denoted by  $C$ . These energies can be derived directly from the flow computation method, or they can be arbitrarily defined on the given flow field. Depending on the employed flow field (the computed flow field or an artificial zero flow field) either statements on the accuracy of the given flow field or on the feasibility of an accurate flow computation can be made. Examples for typical energy formulations appearing in global methods are derived from image invariants under the

displacement field, e.g. the constancy of the brightness, the intensity, the gradient or the curvature at a given location  $\mathbf{x}$  in the original image and the image after warping by means of the parameter vector  $\mathbf{u}$ :

$$\text{brightnessConst: } c(\mathbf{x}, I, \mathbf{u}) = (\nabla_{\{x,y\}} I \mathbf{u} + \nabla_t I)^2 = 0,$$

$$\text{ssdConst: } c(\mathbf{x}, I, \mathbf{u}) = \|I(\mathbf{x}) - I_w(\mathbf{x})\|_{l_2}^2 = 0,$$

$$\text{gradConst: } c(\mathbf{x}, I, \mathbf{u}) = \|\nabla_{\{x,y\}} I(\mathbf{x}) - \nabla_{\{x,y\}} I_w(\mathbf{x})\|_{l_2}^2 = 0,$$

$$\text{hessConst: } c(\mathbf{x}, I, \mathbf{u}) = \|H(\mathbf{x}) - H_w(\mathbf{x})\|_{l_2}^2 = 0,$$

where  $I_w$  and  $H_w$  denote the image sequence and the Hessian of the image sequence, which are warped by the computed parameter vector  $\mathbf{u}$ . In order to obtain a bounded interval for these energies they are mapped to the interval  $[0, 1]$  using the transformation  $\frac{1}{1+c(\mathbf{x}, I(\mathbf{x}), \mathbf{u})^2}$ . Note, that the energy minimum is turned into a maximum. To make the resulting energy function robust to image noise the energy function is scaled by multiplying it by a value  $\kappa_\sigma \geq 1$  depending on the noise level  $\sigma$  and cutting the result to the interval  $[0, 1]$ .

### 6.2.2 Surface Functions

A surface function for a given  $d$ -dimensional parameter vector  $\mathbf{u}$  reflects the variation of the energy  $c \in C$  over the set of modifications of the current parameter vector:

$$S_{\mathbf{x}, \mathbf{u}, c} : \mathbb{R}^d \rightarrow [0, 1], \quad S_{\mathbf{x}, \mathbf{u}, c}(\mathbf{p}) := c(\mathbf{x}, I(\mathbf{x}), \mathbf{u} + \mathbf{p}). \quad (6.2)$$

It can be understood as an indicator for possible alternatives to the current parameter vector as it shows the effect of slight parameter changes  $\mathbf{p}$  on the given energy.

If the parameter changes but the surface function,  $S_{\mathbf{x}, \mathbf{u}, c}(\mathbf{p})$ , remains almost constantly high a rather small reliability is assigned to this optimum, since neighboring parameters yield almost equally low energies. In such cases the surface function shows an aperture problem or homogeneous region, which makes a reliable parameter optimization impossible without further information. In the case of occlusion, transparent structures and noise the maximum of the surface function is usually small indicating that no good estimate is possible at all. Such outliers can make a parameter estimation arbitrarily bad, for example in the case of least squares estimators, which are used in many local methods.

Hence, the intuition is that the computed parameters in the optimum are reliable only if two requirements are fulfilled:

- (a) No other parameter constellation with a similar energy exists. If there are different constellations of parameters yielding very similar surface function values, the solution to the parameter optimization problem is not unique and, thus, unreliable.
- (b) The surface function is sufficiently high at the maximum. If this is not the case we either got stuck in a local energy minimum or, especially if the problem is

convex, there is no satisfying solution to the parameter optimization problem. In this case we have come across an outlier, for which the energy cannot be optimized satisfactorily. In both cases, the optimum is unreliable.

In contrast, a single surface function peak suggests a unique, reliable optimum. Hence, we can now investigate the quality of the energy optimum by examining the intrinsic dimension and the maximum of the surface function.

### 6.2.3 A Continuous Formulation of the Intrinsic Dimension as Simplex Structure

In the discrete formulation, the intrinsic dimension of the surface function at location  $\mathbf{x}$ ,  $S_{\mathbf{x}, \mathbf{u}, c}$ , corresponds to the dimension of the subspace of non-constant values. Let  $d$  be the number of parameters in the parameter optimization problem. We need to examine the variance of the surface function in the  $(d + 1)$ -dimensional space. To this end, the curvature along the main axes of the surface function is computed. Following Felsberg [38] I use a continuous formulation of the intrinsic dimensionality (see Figure 2.2 in Chapter 2.4). Let  $t \in \mathbb{R}_+$  define a threshold indicating a very large curvature value, and let  $\mathbf{v}$  stand for the  $d$ -dimensional curvature vector with its entries sorted in descending order. These entries are normalized to the range  $[0, 1]$  by

$$\mathbf{v}_i = \min\left\{\frac{\mathbf{v}_i}{t}, 1\right\}, i \in \mathbb{N}_d. \quad (6.3)$$

Then each entry can be understood as a barycentric coordinate in the intrinsic dimension simplex, e.g.  $\mathbf{v}_1$  indicates the first coordinate

$$(1 - \mathbf{v}_1)i0d + \mathbf{v}_1 i1d, \quad (6.4)$$

$\mathbf{v}_2$  indicates the second coordinate

$$(1 - \mathbf{v}_2)i1d + \mathbf{v}_2 i2d, \quad (6.5)$$

and so on. Due to  $\mathbf{v}_i \geq \mathbf{v}_{i+1}$  the resulting coordinates always lie within the  $d$ -dimensional simplex defined by the edges  $i0d, i1d, \dots, id$ . This approach can, therefore, be understood as a generalization of the triangle formulation by Felsberg [38].

### 6.2.4 Outlier Detection

After computing the intrinsic dimension vector corresponding to the surface function I come to the second point, the detection of outliers. Outliers are locations in the image sequence, where it is not possible to optimize the energy of the optical flow estimator. One often has to do with optimization problems, for which the result of the estimator can already be arbitrarily bad in case of a single outlier in the data, as is for example the

case in least squares methods (see section 2.6), which are used for local flow computation methods, such as [70, 16]. Hence, it would be beneficial to detect outliers. A simple but effective method for the detection of outliers is to examine the maximum of the surface function  $S_0 \in [0, 1]$ . If the value is sufficiently close to 1, the energy can be optimized by the corresponding parameter set, otherwise the optimization was unsuccessful. This can be the case if the energy cannot be optimized or if the optimization process got stuck in a local minimum. Both cases are not desirable for reliable parameters. In optical flow estimation, especially the difficult situations of occlusions, severe noise, transparent structures or incoherent motion can, thus, be detected.

### 6.2.5 Surface Measures

Based on the proposed intrinsic dimension estimator and the outlier detection a single function  $\varphi$  can now be defined as confidence or situation measure. The situation, where the  $d$  optimized parameters for the optical flow problem are reliable, demands the existence of a high intrinsic dimension with a high maximum value of the surface function,  $S_0$ , at the same time. To combine the intrinsic dimension of the surface function and its maximum value  $S_0$   $\varphi$  is defined in the following way:

$$\varphi : D \times \mathbb{R}^d \rightarrow [0, 1], \varphi(\mathbf{x}, \mathbf{u}) := \phi(S_{\mathbf{x}, \mathbf{u}, c}). \quad (6.6)$$

The function  $\phi$  derives the situation measure value based on properties of the surface function. It will be defined based on the following theoretical considerations. The case of lower intrinsic dimensionality of the surface function in the optimum can be detected by a low minimum curvature value  $\mathbf{v}_n$ . In this way, homogeneous regions and aperture problems in different dimensions of the energy surface are recognized. In the case of an outlier the surface function yields a low maximum value  $S_0$ . Therefore, the value of the function  $\phi$  should always be close to 1 if  $\mathbf{v}_n$  and  $S_0$  are high. Let  $S$  be the set of surface functions defined in Equation (6.2). Then the function  $\phi$  can be defined by

$$\phi : S \rightarrow [0, 1], \phi(S_{\mathbf{x}, \mathbf{u}, c}) := S_0 \cdot \left(1 - \frac{1}{1 + \tau \mathbf{v}_n^2}\right), \quad (6.7)$$

where  $\tau \in \mathbb{R}^+$  is used to scale the influence of the intrinsic dimensionality on the value of  $\phi$ . Here  $\tau$  was set to 60.

## 6.3 Computational Issues

The discretization of the surface function has a large influence on the quality of the estimation of the intrinsic dimension. To discretize a surface function  $S_{\mathbf{x}, \mathbf{u}, c}$  a step size  $h$  and a fixed size  $w$  of the surface are used, where  $h$  is the distance between two surface points within every dimension and  $w$  is the number of surface points in all dimensions

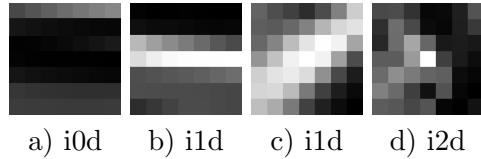


Figure 6.1: Discretized surface functions for two dimensional parameter space.

after discretization, e.g.  $h = 0.5$ ,  $w = 13$  yielded good results. The variances can only be estimated reliably if  $h$  is chosen between 0 and 1 and if bicubic interpolation of the surface function is used. Further preprocessing steps have been applied to the surface functions to obtain good results:

- (a) Since we expect the correct parameter set to be similar to the estimated one, only those values of  $S_{\mathbf{x}, \mathbf{u}, c}(\mathbf{p})$  with small arguments of  $\|\mathbf{p}\|_2$  can actually be considered as alternatives for the current parameter vector. Hence, the examination of the surface function is limited to the direct neighborhood of its origin.
- (b) Since the eigenvalues of the Hessian yield noisy curvature estimates, a robust curvature estimator is introduced. It averages  $n$  curvature values along the principal axis using the following filter mask:  $\frac{1}{n}(\underbrace{1 \dots 1}_n - 2n \underbrace{1 \dots 1}_n)$ .
- (c) To estimate the intrinsic dimension of the surface function only those parts of the image of  $S_{\mathbf{x}, \mathbf{u}, c}$  are relevant which are close to the maximum  $S_0$ , since only these locations denote possible alternatives for the current parameter vector.
- (d) Locations that are separated from the origin of the surface function by a local minimum are likely to belong to other local minima of the original energy and, thus, should not influence the intrinsic dimension of the surface function. Hence, all surface function values that are separated from the origin by a local minimum are set to 0. Such local minima can for example be found by means of a simple flood fill algorithm with starting point at the origin.

Typical discretized surface functions for a two-dimensional parameter space are shown in Figure 6.1.

## 6.4 Experiments and Results

### 6.4.1 Comparison to i2d Measures

To evaluate my results I first compare the quality of the surface measures used as situation measures with zero flow field to the previously known situation measures detecting

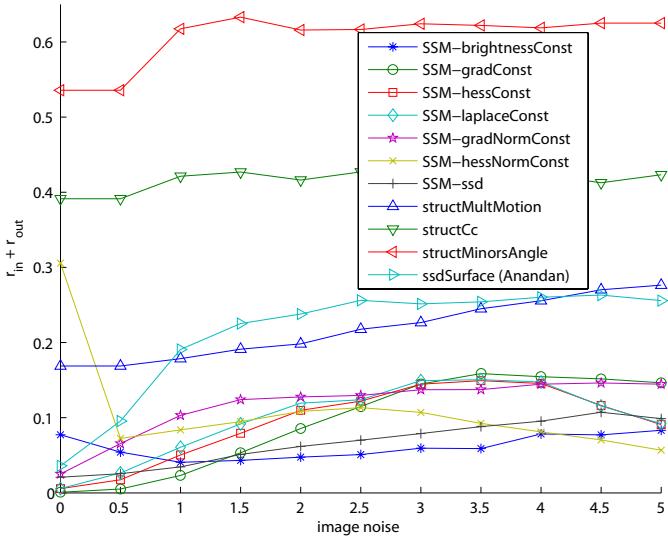


Figure 6.2: Comparison of surface situation measures (SSM-measures) based on different energy functions for the recognition of the i2d situation not due to occlusion to known situation measures for increasing noise levels.

i2d situations and show that all surface measures - independent of the underlying energy function - perform better than the best previously proposed measures and are robust to noise as well.

As test sequence I again use the synthetic sequence shown in Figure 5.2, as it contains every intrinsic dimension and their occlusion. In this way we can examine if the surface measures are able to recognize the situation of accurate flow estimability (intrinsic dimension two without outliers caused by occlusion).

To obtain numerical results I use the same error measure consisting of the sum of the errors within and outside the situation, which was presented in section 5.3. As no measures are known for the detection of i2d situations not due to occlusions the proposed surface measures are compared to the best known measures for the i2d situation: structMultMotion derived from [75], structCc [51], Anandan's measure [3] and structMinorsAngle [12]. Figure 6.2 shows the error measure plotted against an increasing noise level  $\sigma \in [0, 5]$  in the test sequence. The proposed surface situation measures are labeled by the prefix "SSM" and an abbreviation of the energy function they are based on.

We can see that the proposed surface measures generally perform better than the best previously proposed i2d measures for any underlying energy function  $c$ . All surface measures are robust to noise but depend on the robustness of the underlying energy function. The susceptibility to noise increases with the order of the derivatives in the energy function. However, the influence of noise on the surface measures is limited by

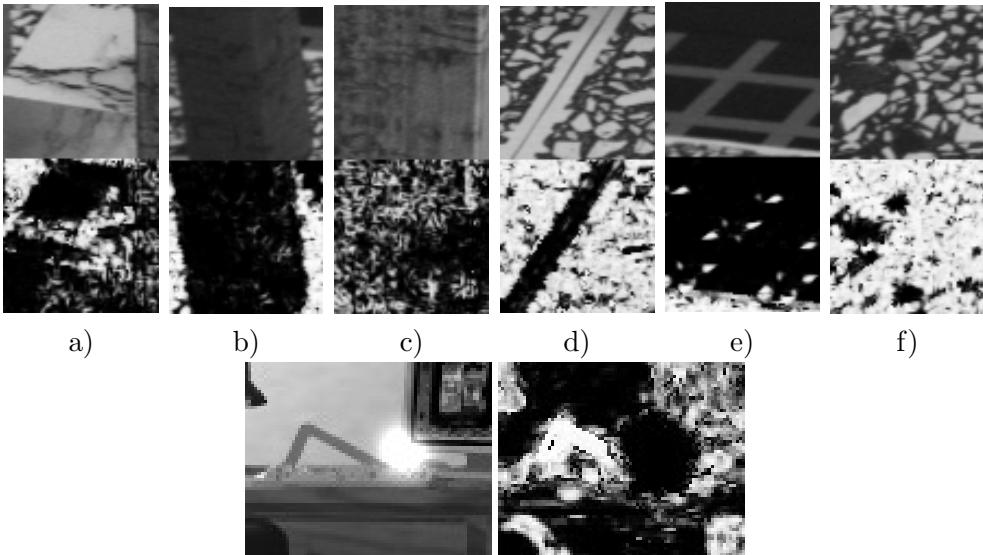


Figure 6.3: Top: Cropped Marble sequence regions with the result of the brightness constancy surface measure for the recognition of the accurate flow estimability situation (i2d not due to occlusions). a), b), c) Texture of blocks (good estimability/i0d), d) Diagonal table line (i1d) e) Flagstones in the background (i1d, good estimability at corners), f) Table (good estimability/i0d), Bottom: Office sequence with additional lens flare and result of the SSD constancy Surface Measure correctly identifying the occlusion.

the robust curvature estimation along the principal axes.

#### 6.4.2 Application to Other Test Sequences

For further validation of the surface situation measures I apply them to standard test sequences. As no ground truth concerning the accurate flow estimability situation is available for these sequences, only a visual evaluation is feasible.

Figure 6.3 a)-f) shows six different cropped regions of the Marble sequence, which has been used for the same purpose in the previous chapter, and the corresponding surface measure result based on the brightness constancy energy function.

In Figure 6.3 a), b) and c) we can see the application of the surface measure to different textures. In a) and b) the Marble blocks show only very little texture, which makes these regions unreliable for flow estimation. In contrast, most parts of the block texture in c) are classified as sufficient for a reliable flow computation. In d) and e) we can see examples of aperture problems (i1d). The diagonal line on the table as well as the edges of the flagstones in the background of the sequence are typical examples for this

situation. Both are recognized well by the surface measure. The corners of the flagstones are correctly recognized as regions, where the optical flow can be estimated reliably. The table region in f) is partially recognized as reliable and partially as i0d. This is due to the larger homogeneous regions in the table texture, as here the result depends on the size of the surface considered. If the whole surface function lies within the homogeneous region, the curvature along the main axis is 0 and, thus, the surface measure result as well.

To demonstrate that surface measures can also detect outliers (e.g. occlusions), I use the cropped Office sequence [73] with an additional lens flare occluding part of the background in Figure 6.3 (this kind of lens flare often poses problems e.g. in traffic scenes). The brightness constancy surface measure detects this region.

#### 6.4.3 Motion Inpainting Based on Situation Measures

To show an application for surface measures used as confidence measures based on the computed flow field I reconstruct optical flows. First, a surface confidence measure map is used to sparsify flow fields calculated on four ground truth sequences (Marble, Yosemite, Street and Office) by the three dimensional linear combined local global (CLG) method by Bruhn et al. [28] and by the local structure tensor method by Bigün [16], both described in Chapter 3. Then motion inpainting is applied to the sparsified displacement fields in order to reconstruct the flow at pixels with low surface measure values. I demonstrate that the angular error [10] is reduced significantly by means of motion inpainting. Table 6.1 shows the average angular error and standard deviation over ten frames for the sparsification and reconstruction of the flow field based on the best previously proposed situation measure (structMultMotion) compared to the new surface measures. For sparsification, the flow field density optimal for motion inpainting with respect to the angular error is chosen.

Concerning the quality of the proposed measures, we can draw several conclusions from the results presented in Table 6.1.

- The average angular error of the motion inpainting algorithm based on the surface measures is lower than the error we obtain based on the best previously proposed situation measure. Hence, using the surface measures we can make more reliable statements on the accuracy of the flow than by means of previous i2d situation measures.
- The average angular error after motion inpainting is lower than the original angular error for the CLG and the structure tensor method. Thus, I conclude that the remaining flow vectors after sparsification contain the most relevant information of the original flow field, and that most other information is dispensable, even obstructive, for the computation of a 100% dense flow field.

<b>CLG</b>	original	sparsified	density	inpainting	best previous
Marble	$3.88 \pm 3.39$	$3.59 \pm 3.03$	70.6	$3.87 \pm 3.38$	$3.88 \pm 3.39$
Yosemite	$4.13 \pm 3.36$	$2.78 \pm 2.24$	20.7	$3.85 \pm 3.00$	$4.13 \pm 3.36$
Street	$8.01 \pm 15.47$	$2.77 \pm 2.52$	11.5	$7.73 \pm 16.23$	$7.99 \pm 15.48$
Office	$3.74 \pm 3.93$	$3.25 \pm 4.80$	26.7	$3.59 \pm 3.93$	$3.62 \pm 3.91$
<b>ST</b>	original	sparsified	density	inpainting	best previous
Marble	$4.49 \pm 6.49$	$2.96 \pm 2.25$	42.3	$3.40 \pm 3.56$	$3.88 \pm 4.89$
Yosemite	$4.52 \pm 10.10$	$2.90 \pm 3.49$	37.5	$2.76 \pm 3.94$	$4.23 \pm 9.18$
Street	$5.97 \pm 16.92$	$2.07 \pm 5.61$	34.6	$4.95 \pm 13.23$	$5.69 \pm 16.47$
Office	$7.21 \pm 11.82$	$2.59 \pm 4.32$	5.1	$4.48 \pm 4.49$	$6.35 \pm 10.14$

Table 6.1: Angular error for four test sequences for the original field, the sparsified field with given density (percentage), the result of motion inpainting based on the best surface measure and the result of motion inpainting based on the previously best situation measure (structMultMotion), averaged over ten frames for the CLG and the structure tensor (ST) method; the density of the sparsified field was chosen as optimal for motion inpainting.

- The table also indicates the average angular error for the sparsification of the flow field by means of the surface measures. Here I chose the sparsification density which has been found optimal for motion inpainting. The sparsification error is lower than the motion inpainting error and can be achieved if a dense flow field is not required.
- For both the CLG and the structure tensor method the inpainting of the sparsified flow fields yields lower angular errors than the original methods for all test sequences. The results of the local structure tensor method after motion inpainting are even superior to the original and the inpainted global CLG method in all cases but one. Therefore, I conclude that – in contrast to the accepted opinion which favors global methods over local methods if dense flow fields are required – the filling-in effect of global methods is not necessarily beneficial for obtaining an accurate dense flow field. Instead, local and global methods alike can lead to better results if motion inpainting in combination with surface measures for sparsification is employed. Here, local methods often even seem preferable.

## 6.5 Summary and Conclusion

I have proposed surface measures, which can be employed either as confidence measures to analyze the accuracy of a given flow field or as situation measures to estimate the feasibility of accurate optical flow computation. They can be applied in case of an

arbitrary number of parameters. The proposed surface measures have proven robust to noise, yield better results than all previously proposed situation measures, also for real-world sequences, and contain the most relevant information for the reconstruction of the original flow field with even higher quality. Based on these measures locally or globally computed flow fields were sparsified and the missing flow vectors were filled in by a basic motion inpainting algorithm. Tests have been conducted using the CLG method and the structure tensor method on four standard test sequences. For the chosen test sequences I conclude that the application of a postprocessing method to sparsified flow fields calculated with local or global methods yields better results than can be achieved by exploiting the filling-in effect of global methods. Hence, in contrast to the accepted opinion, global methods are not always preferable to local methods if a dense flow field is required, because motion inpainting only based on reliable flow vectors can lead to superior results.

# Chapter 7

---

## Statistical Confidence Estimation

### 7.1 Introduction

Confidence measures evaluate the accuracy of a given flow vector and are, thus, indispensable to assess and increase the quality of optical flow fields. Using the information provided by confidence measures, the accuracy of the estimated flow field can be improved by integrating the confidence measure into the calculation method or by post-processing, e.g. removing and reconstructing incorrect flow vectors. In this chapter I propose two statistical approaches and extensions to confidence estimation for optical flow fields.

#### 7.1.1 Motivation

It is of utmost importance for any optical flow measurement technique to give a prediction of the quality and reliability of each individual flow vector. This was already asserted in 1994 in the landmark paper by Barron et al. [10], where the authors stated that “confidence measures are rarely addressed in literature” even though “they are crucial to the successful use of all [optical flow] techniques”.

Confidence measures map each individual flow vector to a value within the interval  $[0, 1]$ , where 0 stands for no confidence and 1 for high confidence. In contrast to situation measures they always use the optical flow field to assess its accuracy

$$\varphi : D \times I \times \mathbb{R}^d \rightarrow [0, 1]. \quad (7.1)$$

An example can be seen in Figure 7.1, where the confidence is continuously indicated between the colors red (which stands for confidence 0) and green (which stands for confidence 1).

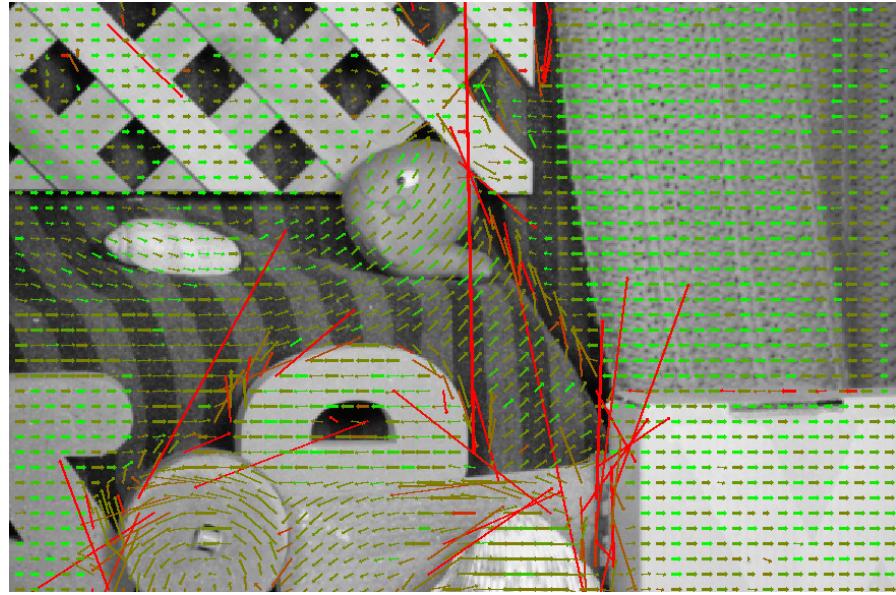


Figure 7.1: Color coded confidence computed for a structure tensor flow field and the Rubber Whale sequence. Green stands for high confidence and red for low confidence.

There are mainly four benefits of confidence measures:

- (a) unreliable flow vectors can be identified before they cause harm to subsequent processing steps,
- (b) corrupted optical flow regions can be identified and possibly recovered by model-based interpolation (also denoted as “inpainting”),
- (c) existing optical flow methods can be improved, e.g. by integrating the confidence measure into variational approaches,
- (d) fast, structurally simple optical flow methods in combination with a confidence measure can replace slow, complicated ones.

Yet, the confidence measures known today are inadequate for the assessment of the accuracy of optical flow fields due to the following reasons:

- (a) Many confidence measures are, in fact, situation measures as they infer confidence values based on the local structure of the image sequence only without taking into account the computed flow field.

- (b) Most confidence measures are directly derived from specific optical flow computation techniques and, thus, can only be applied to flow fields computed by this method. But if the same model for flow and confidence estimation is used the confidence measure only verifies the restrictions already imposed by the flow computation model. Thus, errors are often not detected as the flow obeys the model. Hence, I opt against using the same motion model for confidence estimation.
- (c) None of the proposed measures is statistically motivated despite the notion “confidence measure”.

In this chapter we, therefore, propose a statistical confidence measure, which is generally applicable independently of the flow computation method. An additional benefit of our method is its adaptability to application-specific data, i.e. it exploits the fact that typical flow fields can be very different for various applications.

### 7.1.2 Related Work

For optical flow estimators a thorough analysis of the errors in the estimated flow field is important. These errors have been analyzed by Fermüller et al. [40]. To predict errors without ground truth confidence measures are used. The number of previously proposed confidence measures for optical flow fields is limited. In addition to the comparison by Barron et al. [10], another comparison of different confidence measures was carried out by Bainbridge and Lane [7].

In the following I will present confidence measures that have been proposed in the literature so far. Confidence measures can be classified based on two aspects: first, I distinguish between confidence measures which derive their information from the image sequence and those, which derive it from the computed flow field. The second group is then subdivided into measures that depend on the flow computation method and those which are independent of it.

Most of the measures rely on the image sequence only and are, thus, naturally independent of the flow computation method. These are, in fact, situation measures, which were described, classified and compared in Chapter 5. For comparison, in this chapter I use the three measures based on the structure tensor,  $structCt$ ,  $structCs$  and  $structCc$  as well as the image gradient  $grad$ . All of these measures are examples of confidence measures which assess the reliability of a given flow vector exclusively based on the input image sequence.

The second group consists of measures taking into account the flow field, but these measures are derived from and are, thus, limited to special flow computation methods. An example is the confidence measure proposed by Bruhn and Weickert [26] for variational optical flow methods, which computes the inverse of the variational energy remaining after optimization. In this way, locations are identified, where the energy could not be minimized, e.g. in cases where the model assumption of the method is not valid, such

as in the vicinity of edges in case of homogeneous regularization. Hence, their approach assigns a low confidence value to these locations. So far, there are no measures which take into account the flow without being restricted to its computation method. Previous work on optical flow statistics and statistical models mainly focuses on the work by Roth and Black. In [85] they investigated the statistics of the horizontal and vertical velocities and found that the derivative statistics strongly resemble a Student t-distribution. They used these insights to formulate an optical flow prior for a Markov random field. In [96] Sun et al. learn statistical models of brightness constancy errors, high-order constancy assumptions such as gradient constancy and spatial properties of optical flow fields by means of random fields.

### 7.1.3 Contribution

Our contribution in this chapter are two statistical confidence measures. The first is based on linear subspace projections and mainly applicable to flow fields computed by local optical flow methods. It has been published in [65]. The second confidence measure is a purely statistical confidence measure, which is generally applicable to any kind of computed flow field. It is based on a hypothesis test and can be extended to a nonlinear method. Furthermore, it can be adapted to deal with sparse flow fields often occurring in applications such as traffic sequences. The linear confidence measure has been published in [66]. The measures are compared to previously used situation and confidence measures based on error quantile plots.

## 7.2 A Confidence Measure Based on Linear Subspace Projections

I first propose a new confidence measure that is adaptable to the current flow computation problem by means of unsupervised learning. In fact, the measure can be used for all optical flow fields that have been computed with no or minor smoothness assumptions. Even ground truth data, which is generally unavailable, is not necessary, as the model can be learned either from a set of ground truth flow fields or from a previously computed flow field. The linear subspace projection method has been applied to the estimation of optical flows before, directly by Black et al. in [23] and by means of Markov Random Fields by Roth and Black in [85]. In contrast to these approaches, where only spatial information is used, I extend the subspace method to include temporal information of the flow field and derive a new confidence measure. Since much of the information contained in a flow field is only obvious in the temporal domain, the inclusion of temporal information is indispensable.

The concept of our confidence measure is based on the idea of learning typical displacement vector constellations within a local neighborhood. The resulting model consists of

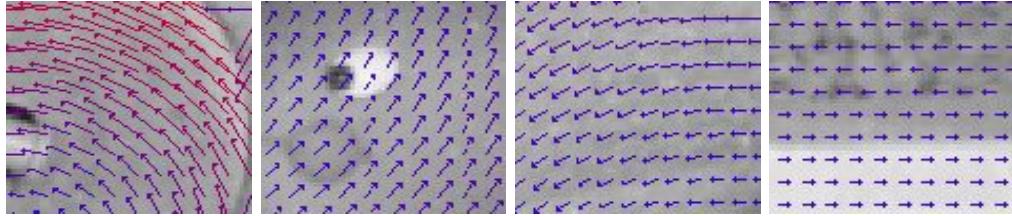


Figure 7.2: Examples of flow field patches from which the motion statistics are computed.

a set of basis flows, a linear subspace of the flow fields, that is sufficient to reconstruct 99% of the information contained in the flow fields. Displacement vectors that cannot be reconstructed by this model are considered unreliable. Hence, the reconstruction error is chosen as confidence measure. It performs better than previously proposed confidence measures and obtains a substantial gain of quality in several cases.

### 7.2.1 Training Data Selection

Instead of defining a motion model statistical methods are used to learn the motion model directly from sample motion data. To draw conclusions on the accuracy of a flow vector, the surrounding flow field patch of a predefined size (see Figure 7.2) is examined.

In the following a spatio-temporal flow field patch is defined. For  $n$  given image sequence locations  $(x_i, t_i) \in \Omega \times [0, T]$ , a finite time interval  $[t_i - \tau, t_i + \tau]$ ,  $t_i \geq \tau$  and a spatial neighborhood  $\omega(x_i, t_i) \subset \Omega$  of fixed size let

$$S_i : \omega(x_i, t_i) \times [t_i - \tau, t_i + \tau] \rightarrow \mathbb{R}^2, i \in \mathbb{N}_n, \quad (7.2)$$

denote  $n$  spatio-temporal flow field patch samples centered on  $(x_i, t_i)$ .

As each flow vector  $S_i$  consists of a horizontal and vertical flow component its size  $p$  corresponds to twice the number of flow field vectors contained in the spatio-temporal neighborhood,  $p := 2|\omega|(2\tau + 1)$ . Let  $\mathbf{s}_i = \text{vec}(S_i) \in \mathbb{R}^p$ ,  $i \in \mathbb{N}_n$ , denote the columnwise vectorization of the  $i$ -th sample flow field patch. The relation between indices in the original spatio-temporal sample flow field patch  $S_i$  and the vector  $\mathbf{s}_i$  can then be described by a mapping

$$q : \omega(x_i, t_i) \times [t_i - \tau, t_i + \tau] \times \{1, 2\} \rightarrow \mathbb{N}_p, \quad (7.3)$$

where 1 or 2 indicates the horizontal or vertical component of each flow vector, respectively.

To obtain statistical information on the accuracy, a probabilistic motion model is learned from training data, which can be ground truth flow fields, synthetic flow fields, computed flow fields or every other flow field that is considered suitable. If motion estimation is performed for an application domain, where typical motion patterns are known a priori,

the training data should of course reflect this. Yet, in general there is no need for prior knowledge on the type of motion occurring in the scene as we still obtain results of high accuracy if no such prior knowledge is available. It is even possible to use the flow field for which we want to compute the confidence as training data, i.e. finding outliers in one single data set. This leads to a very general approach, which allows for the incorporation of different levels of prior knowledge.

### 7.2.2 Symmetrization

In order to avoid any directional bias in the following prediction methods, it is important to perform all possible rotations and reflections on the training data, including time reversal. This means that the training flow field patches are rotated several times, the vectors are reflected on the horizontal and vertical axis, and the temporal direction of the flow field patch sample is inverted. In this way we obtain, as desired, a mean vector  $\mathbf{m} = \vec{0}$ .

### 7.2.3 Learning the Motion Model

In order to learn the motion model, principal component analysis (PCA) described in Chapter 2.5.4 is used similar to [23].

By means of PCA (or, alternatively, robust PCA [33]) a new orthogonal basis system  $B = [\mathbf{b}_1, \dots, \mathbf{b}_p]$  can be computed, within which the original sample fields are decorrelated. Let the basis components be sorted according to decreasing eigenvalues. Then the first  $k \leq p$  eigenvectors with the largest eigenvalues contain most of the variance contained in the sample data, whereas the eigenvectors with small eigenvalues usually represent noise or errors in the sample flow fields and, thus, should be removed from the set. Thus, the first  $k$  basis components derived in this way span a linear  $k$ -dimensional subspace of the original sample data preserving most of the sample data information. Within this subspace the sample flow fields can be approximated by a linear combination of the first  $k$  principal components  $\mathbf{b}_j$ ,  $j \in \mathbb{N}_k$ , and the sample mean  $\mathbf{m} = \frac{1}{n} \sum_{j=1}^n \mathbf{s}_j$ :

$$\mathbf{s}_i = \sum_{j=1}^k \alpha_j \mathbf{b}_j + \mathbf{m} + \mathbf{e} , \quad (7.4)$$

where  $\mathbf{e} \in \mathbb{R}^p$  denotes the approximation error. In order to select the number of eigenvectors containing the fraction  $\delta \in (0, 1)$  of the information of the original dataset the value  $k$  is chosen based on the eigenvalues  $\lambda_i$  of the eigenvectors  $\mathbf{b}_i$ ,  $i \in \mathbb{N}_p$ , according to (2.30), such that

$$k := \operatorname{argmin}_{j \in \mathbb{N}_p} \frac{\sum_{i=1}^j \lambda_i}{\sum_{i=1}^p \lambda_i} \geq \delta . \quad (7.5)$$

The linear subspace, thus, restricts possible solutions of the flow estimation problem to the subspace of typical motion patterns statistically learned from sample data. Examples for such typical motion patterns are presented in Figure 7.3. Using temporal information the resulting eigenflows can represent complex temporal phenomena such as a direction change, a moving motion discontinuity or a moving divergence.

With the eigenvectors ("eigenflows") any vectorized displacement vector neighborhood  $\mathcal{N}_x$  centered on position  $x$  can now be approximately reconstructed by a linear combination of the  $k$  selected eigenflows using the reconstruction function  $r$

$$r(\mathcal{N}_x, k) = \sum_{i=1}^k \alpha_i \mathbf{b}_i + \mathbf{m} . \quad (7.6)$$

In order to obtain the coefficient vector  $\alpha$  containing the eigenflow coefficients  $\alpha_i$ , it is sufficient to project the sample neighborhood  $\mathcal{N}_x$  into the linear subspace spanned by the eigenflows using the transformation

$$\alpha = B^T (\mathcal{N}_x - \mathbf{m}) . \quad (7.7)$$

The linear combinations of the previously derived eigenflow vectors represent typical flow field neighborhood constellations. Depending on the training data the information contained in the learned model varies. If ground truth flow fields are used many sample sequences are necessary to include most of the possible flow constellations. However, as only very few sequences with ground truth exist the resulting eigenflows only represent an incomplete number of constellations. In contrast, it is possible to compute the flow for a given sequence and use exactly this computed flow as input for the unsupervised learning algorithm. In this way the resulting model will be well adapted to the current flow problem. However, if the flow computation method does not allow certain displacement vector constellations such as rotations the trained linear subspace will not be sufficient to represent these constellations either, as all training samples are derived from the computed flow field. In both cases, if we learn from insufficient ground truth flows or from incorrect, computed flow fields, the problem that correct flow constellations cannot be reconstructed from the eigenflows persists.

#### 7.2.4 A Confidence Measure from Eigenflows

To evaluate the confidence of a given flow vector its validity within its spatio-temporal context has to be considered, that is within its neighborhood  $\mathcal{N}_x$  of flow vectors. Given a number of  $k$  model parameters, e.g. eigenflows, a confidence measure can be derived based on the assumption that displacement vectors are the more reliable within their neighborhood the better these flow vector constellations can be reconstructed from the eigenflows.

The accuracy of a computed flow vector can in general be assessed based on a chosen

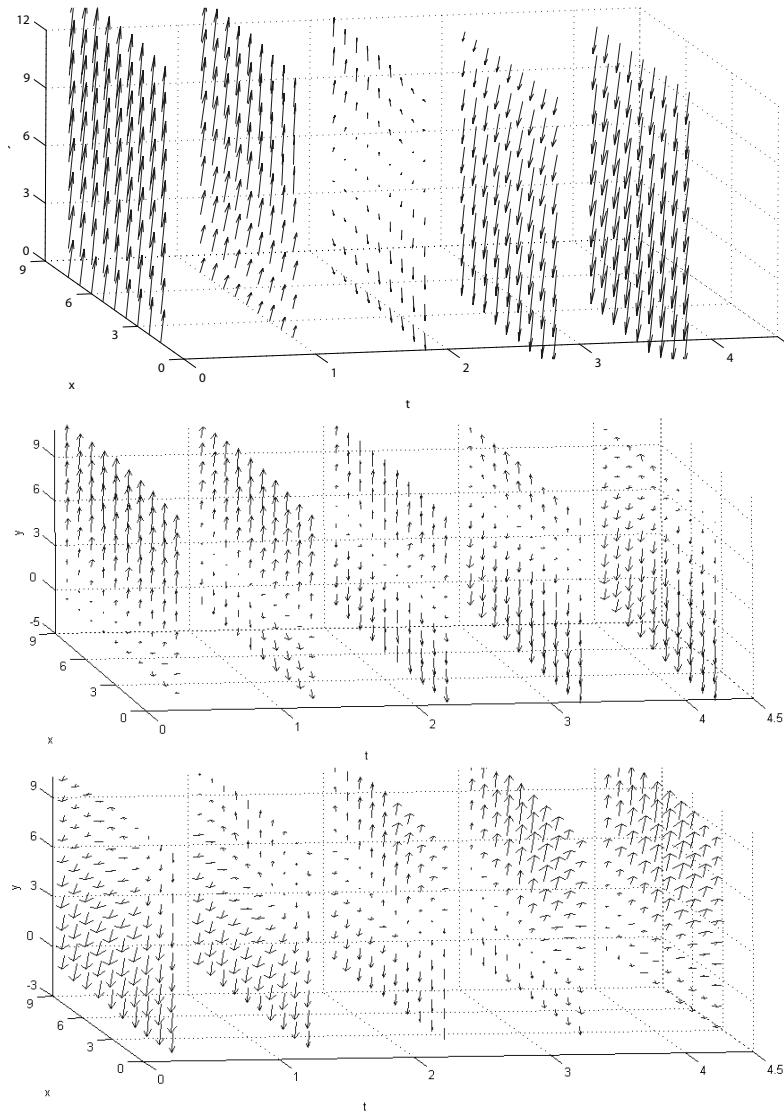


Figure 7.3: Examples for eigenflows calculated from computed flow fields using spatial and temporal information. The inclusion of temporal information allows for the representation of complex temporal phenomena such as a flow direction change (top), a moving motion discontinuity (center) and a moving divergence (bottom).

error measure  $E$  (see section 4.1.2). Hence, the normalized reconstruction error of the flow vector will serve as confidence measure:

$$\varphi(\mathbf{x}, \mathbf{u}) = 1 - \frac{E(\mathbf{u}, r(\mathcal{N}_{\mathbf{x}}, k))}{\max(E)}. \quad (7.8)$$

The size of the neighborhood  $\mathcal{N}_{\mathbf{x}}$  of course has to be the same as for the eigenflows. Our proposed method may fail on rare occasions of untypical, but correct flows encountered in the image data. These are singular events, which in case of underrepresentation in the training data may not be adequately incorporated into our basic PCA framework. A range of more refined algorithms has been developed in the field of statistical learning. Some of these might solve the problem of underrepresentation, such as multiclass PCA [77] or partial least squares regression.

## 7.3 A Statistical Confidence Measure

The second confidence measure I propose is a purely statistical measure based on a learned model of typical flow field patches. The model consists of the first and second order moment of the flow field patch distribution obtained from training data. To assess the accuracy of a given flow vector a test statistic is formulated together with a hypothesis test (see section 2.2.2). Since the true distribution of the test statistic is unknown, an empirical distribution is estimated from training data. The p-value, the minimum significance level for which the current hypothesis is rejected (see section 2.2.2), expresses the confidence associated with the current vector.

To evaluate the proposed method the confidence is used to indicate the order of sparsification of the flow field. In each sparsification step the average error of the remaining flow field is computed and compared to the optimal value and other confidence measures. The results show that the proposed statistical method yields lower errors than common confidence measures for almost all test sequences and optical flow methods. Furthermore, I show that it can be extended into a nonlinear estimator, which further increases its accuracy, and that it can be modified in order to handle sparse flow fields.

### 7.3.1 Hypothesis Testing

For the statistical confidence measure I use the same training data selection process as for the previous confidence measure including symmetrization (see section 7.2.1 and 7.2.2). To obtain the motion model, the first and second order moments of the flow field distribution, the empirical mean  $\mathbf{m}$  and the covariance matrix  $\mathbf{C}$ , are computed from the training data set, which contains the vectorized flow field patches in its columns.

To assess the reliability of a given flow vector based on its neighborhood the following hypothesis is tested

$H_0$ : “The central flow vector of a given flow field patch follows the underlying conditional distribution given the remaining flow vectors of the patch.”.

Let  $D := \Omega \times [0, T]$  again denote the spatio-temporal image domain and  $V : D \rightarrow \mathbb{R}^p$  a  $p$ -dimensional real valued random variable describing possible vectorized flow field patches. Testing the confidence of the central vector of a regarded flow patch boils down to specifying the conditional pdf of the central vector given the remainders of the flow patch, and comparing the candidate flow vector against this prediction, considering a metric induced by the conditional pdf.

For a given image sequence location  $(x, y, t) \in D$  let  $\mathbf{v} \in \mathbb{R}^p$  correspond to the vectorized flow field patch centered on this location, and let

$$(i, j), i < j, \quad (7.9)$$

denote the line indices of  $\mathbf{v}$  corresponding to the horizontal and vertical flow vector component of the central vector of the original patch. We partition  $\mathbf{v}$  into two disjoint vectors, the central flow vector  $\mathbf{v}_a$ , and the “remainders”  $\mathbf{v}_b$  of the regarded flow patch:

$$\begin{aligned} \mathbf{v}_a &= (\mathbf{v}_i, \mathbf{v}_j)^T, \\ \mathbf{v}_b &= (\mathbf{v}_1, \dots, \mathbf{v}_{i-1}, \mathbf{v}_{i+1}, \dots, \mathbf{v}_{j-1}, \mathbf{v}_{j+1}, \dots, \mathbf{v}_p)^T. \end{aligned} \quad (7.10)$$

The mean vector and covariance matrix  $\mathbf{C}$  are partitioned accordingly:

$$\mathbf{m} = \begin{pmatrix} \mathbf{m}_a \\ \mathbf{m}_b \end{pmatrix} \quad \mathbf{C} = \begin{pmatrix} \mathbf{C}_{aa} & \mathbf{C}_{ab} \\ \mathbf{C}_{ba} & \mathbf{C}_{bb} \end{pmatrix}.$$

The basic idea is now to predict the central vector of a flow field patch from its neighboring vectors and to evaluate the difference between the predicted vector and the actually measured vector in a hypothesis test. As shown in Chapter 2.3, the best linear unbiased estimator (BLUE) of the central vector and its prediction error correspond to the first and second order moments of the conditional pdf, which are given by

$$\hat{\mathbf{v}}_a = \mathbf{m}_a + \mathbf{C}_{ab}\mathbf{C}_{bb}^{-1}(\mathbf{v}_b - \mathbf{m}_b) = \mathbf{m}_{a|b}, \quad (7.11)$$

$$\text{Var}(\mathbf{v}_a - \hat{\mathbf{v}}_a) = \mathbf{C}_{aa} - \mathbf{C}_{ab}\mathbf{C}_{bb}^{-1}\mathbf{C}_{ba} = \mathbf{C}_{a|b}. \quad (7.12)$$

I stress that these first and second order moments of the conditional pdf are valid independent of the assumption of a normal distribution. The covariance matrix does not imply Gaussianity nor does it imply elliptical shape of the pdf. A covariance matrix is next to the mean an important characteristic of any distribution.

To derive the test statistic let

$$\begin{aligned} d_M : \mathbb{R}^p &\rightarrow \mathbb{R}_0^+ \\ d_M(\mathbf{v}) &= (\mathbf{v}_a - \mathbf{m}_{a|b})^T \mathbf{C}_{a|b}^{-1} (\mathbf{v}_a - \mathbf{m}_{a|b}) \end{aligned} \quad (7.13)$$

denote the squared Mahalanobis distance between  $\mathbf{v}_a$  and the mean vector  $\mathbf{m}_{a|b}$  given the covariance matrix  $\mathbf{C}_{a|b}$ . Even though we do not know the true conditional distribution, the squared Mahalanobis distance is chosen as test statistic for the following reasons:

- Since according to Chapter 2.3 the best linear unbiased estimator for the central vector  $\mathbf{v}_a$  given the remaining flow vectors in  $\mathbf{v}_b$  corresponds to the conditional mean  $\mathbf{m}_{a|b}$  of the learned distribution with covariance matrix  $\mathbf{C}_{a|b}$ , the Mahalanobis distance can be understood as the weighted distance between the central vector of the patch and its prediction  $\mathbf{m}_{a|b}$  from the surrounding field.
- The Mahalanobis distance is the optimal test statistic in case of a normally distributed conditional pdf of the central flow vector. This does not imply that the image data or the flow data are assumed to be normally distributed as well.

To carry out a hypothesis test (significance test), we have to determine quantiles of the distribution of the test statistic for the case that the null hypothesis to be tested is known to be true. To this end, the empirical cumulative distribution function  $G : \mathbb{R}_+ \rightarrow [0, 1]$  of the test statistic is computed from training data. We obtain the empirical quantile function

$$G^{-1} : [0, 1] \rightarrow \mathbb{R}_+ \quad (7.14)$$

$$G^{-1}(q) = \inf\{x \in \mathbb{R} \mid G(x) \geq q\} . \quad (7.15)$$

To, finally, examine the validity of  $H_0$  a hypothesis test is applied

$$\phi_\alpha : \mathbb{R}^p \rightarrow \{0, 1\} \quad (7.16)$$

$$\phi_\alpha(\mathbf{v}) = \begin{cases} 0, & \text{if } d_M(\mathbf{v}) \leq G^{-1}(1 - \alpha) \\ 1, & \text{otherwise} \end{cases} \quad (7.17)$$

where  $\phi_\alpha(\mathbf{v}) = 1$  indicates the rejection of the hypothesis  $H_0$ . Based on this hypothesis test we would obtain a binary confidence measure instead of a continuous mapping to the interval  $[0, 1]$ . Furthermore, it would be inconvenient to recompute the confidence measure each time the significance level  $\alpha$  is modified. Therefore, I propose to use the concept of *p-values*, which was introduced by Fisher [41] and defined in section 2.2.2. A p-value function  $\Pi$  maps each sample vector to the minimum significance level  $\alpha$  for which the hypothesis would still be rejected, i.e.

$$\Pi : \mathbb{R}^p \rightarrow [0, 1] \quad (7.18)$$

$$\begin{aligned} \Pi(\mathbf{v}) &= \inf\{\alpha \in [0, 1] \mid \phi_\alpha(\mathbf{v}) = 1\} \\ &= \inf\{\alpha \in [0, 1] \mid d_M(\mathbf{v}) > G^{-1}(1 - \alpha)\} . \end{aligned}$$

Hence, we finally obtain the following confidence measure

$$\varphi : \mathbb{R}^p \rightarrow [0, 1] \quad (7.19)$$

$$\varphi(\mathbf{v}) = \Pi(\mathbf{v}) = \inf\{\alpha \in [0, 1] \mid d_M(\mathbf{v}) > G^{-1}(1 - \alpha)\} .$$

## 7.4 Applicability of the Test

One issue we have to cope with is the applicability of the proposed hypothesis test. In case of inconsistent flow field patches, where the central vector cannot be predicted reliably from the surrounding vectors, the result of the hypothesis test is unreliable. Reliability is only given for typical surrounding flow field patches. Hence, we can only compare the central vector to the prediction by the surrounding flow vectors and the learned model, if the surrounding vectors follow the model.

In order to make the results independent of the average flow vector length of sample and test patch the whole patch is normalized by dividing by its  $l_2$ -norm. To detect locations, where the confidence measure is unreliable, I propose a second hypothesis test, which examines the hypothesis

$H_1$ : “*The flow vectors of the current flow field patch, which surround the central vector, follow the underlying distribution.*”.

As reasoned before, I again choose the Mahalanobis distance as test statistic. The covariance matrix and mean vector can simply be computed by marginalizing the original flow field patch distribution over the variables indicated by  $i$  and  $j$  in (7.9). Since the underlying distribution of the surrounding flow vectors is again unknown, I then compute their empirical cumulative distribution function from sample data and obtain p-values in exactly the same way as for the test of  $H_0$ . The result of the hypothesis test for  $H_1$  now yields information on the applicability of the hypothesis test for  $H_0$ . In case of a low p-value for the  $H_1$  test, the result of the  $H_0$  test is not reliable.

In this way, we finally arrive at a two-stage method: For any given flow field we estimate the cumulative distribution function of the central vector given the surrounding vectors of the patch, and we estimate the cumulative distribution function of the surrounding flow vectors themselves. Then the  $H_1$  test yields results on the applicability of the confidence test by judging the consistency of the surrounding flow vectors only. In a second step the  $H_0$  test can then be applied in locations identified as reliable by the  $H_1$  test in order to obtain statements on the reliability of the central flow vector given the surrounding ones.

## 7.5 Application to Sparse Vector Fields

Sparse vector fields often occur in applications. Since the proposed confidence measure is based on a distribution over the flow vectors of a complete patch the approach has to be modified in order to allow for the confidence computation of sparse vector fields. To compute the confidence at the current location, let  $k$  refer to the number of flow vectors existing in the current patch, and let  $\mathbf{w} \in \{0, 1\}^p$  indicate if the flow vector belonging to the current index is set in the current patch. In case it is set this is denoted by  $\mathbf{w}_i = 1$ ,

otherwise the entry is set to 0.

We project the learned distribution described by its first and second order moments  $\mathbf{m}$  and  $\mathbf{C}$  into the subspace corresponding to the vectors existing in the current flow field patch. This is done by multiplying  $\mathbf{C}$  and  $\mathbf{m}$  by a matrix  $P \in \mathbb{N}^{k \times p}$ , which is obtained from an identity matrix by removing every line with index  $i$  for which  $\mathbf{w}_i = 0$ :

$$\mathbf{C}' = P\mathbf{C}P^T \quad (7.20)$$

$$\mathbf{m}' = P\mathbf{m}. \quad (7.21)$$

The resulting distribution is described by its moments  $\mathbf{C}'$  and  $\mathbf{m}'$ . Again, we condition on the central vector  $\mathbf{v} \in \mathbb{R}^{2k}$  as described in equation (7.11) and obtain the moments  $\mathbf{C}'_{a|b}$  and  $\mathbf{m}'_{a|b}$  of the projected, conditional distribution. Based on this distribution, we can then compute the Mahalanobis distance for the central vector of the sparse flow field patch just as in the case of a dense flow field. To obtain p-values we would have to estimate the distribution of the Mahalanobis distance function  $d'_M(\mathbf{v})$ , the test statistic, for the current population of the flow field patch. Since there are  $(\frac{p}{2})^2$  possible populations of the flow field patch, just as many different distributions over the test statistic would have to be estimated. This is computationally infeasible. Hence, I propose to simply use the inverse of the Mahalanobis distance as confidence function in case of sparse vector fields

$$\varphi(\mathbf{v}) = \frac{1}{1 + d'_M(\mathbf{v})}. \quad (7.22)$$

## 7.6 A Nonlinear Extension

The previously proposed confidence measure relies on a linear prediction of the central vector of the flow field patch from its surrounding flow vectors. In order to obtain results of higher accuracy, the vector  $\mathbf{v}$  is extended by nonlinear, polynomial combinations of flow vector components. This way, even more relations between the central flow vector and those in the surrounding patch can be represented by  $\mathbf{v}$ .

To limit the number of possible flow vector combinations I restrict this approach to polynomials of degree three. In this way higher powers of flow vector components are possible without too much complexity and without losing the sign (as could be the case for degree two). To identify the most meaningful combinations of flow vectors the normalized cross covariance  $Z$  is used. For two random variables,  $X$  and  $Y$ ,  $Z$  is defined as follows:

$$Z(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X)} \sqrt{\text{Var}(Y)}}. \quad (7.23)$$

We estimate the normalized cross covariance matrix  $\mathbf{Z}$  of the central vector of the patch and all degree three polynomials computed from its surrounding vectors based on sample data taken from the Yosemite, Marble, Street, Office and Rubber Whale sequences.

		1		
	3	2		
			3	
			2	
	1			1
	1	1		
			2	
	1	1		
			1	
	3			
		1		
		2		
			2	
			1	
	1	2		
			1	
		1		
		2		
			2	
	1	1	1	
			2	
	1	1		
		1		
		2		
			1	
				1
				2
				1

Figure 7.4: Polynomials containing nonlinear combinations of the flow vector components in a  $3 \times 3$  neighborhood, which are used to extend the input vector  $\mathbf{v}$ . The rectangles represent the  $3 \times 3$  neighborhood, and the numbers the power of the components in the polynomial. The first and second row describe the polynomials chosen to estimate the horizontal flow vector component, the third and fourth row those chosen to estimate the vertical flow vector component. As the horizontal central flow vector is best described by horizontal flow components and the vertical central flow by vertical neighborhood components, only the corresponding dimension (horizontal in the first and second row, vertical and the third and fourth row) is indicated by the rectangles, as no interrelations between both dimensions exist in the polynomials.

Then the polynomials with the highest normalized cross covariance are chosen for each of the two components, because they describe the central vector best. The computation of the normalized cross covariance matrix clearly shows that relations between the horizontal central vector are strongest with the horizontal components of its neighbors, that means that horizontal motion is best described by nonlinear combinations of horizontal neighboring flow vector components. The equivalent holds true for vertical motion. In case a  $3 \times 3$  neighborhood is chosen, the polynomials in Figure 7.4 have been identified as having the most significant relation with the central vector. For testing the nonlinear confidence measure, I altogether chose 20 polynomials for each flow vector component.

## 7.7 Results

As there are several test sequences with ground truth data and numerous optical flow computation methods with different parameters each, it is impossible to present an ex-

tensive comparison between the proposed and previously known confidence measures. Hence, I will present results for a selection of typically used real and artificial sequences and flow computation methods. Here, the Yosemite, the Marble and the Rubber Whale sequence (from the Middlebury database [8]) are used. As optical flow computation methods the local structure tensor method [16], the non-linear 2D multiresolution combined local global method (CLG) [28] as well as the methods proposed by Nir [78] and Farnebäck [37] are employed. To quantify the error  $e(\mathbf{x}) \in \mathbb{R}_0^+$  of a given flow vector at image sequence location  $\mathbf{x} \in D$  the endpoint error and the angular error (see section 4.1.2) are used.

The proposed approaches are compared to several of the situation and confidence measures described in previous sections. These are the three measures examining the intrinsic dimension of the image sequence by Haussecker and Spies [51] ( $strCt$ ,  $strCs$ ,  $strCc$ ), the inverse of the energy of the global flow computation method by Bruhn et al. [26] ( $inverseEnergy$ ), and the image gradient measure ( $grad$ ), which is approximated by central differences. Note that the inverse of the energy measure is only applicable for variational approaches and has, thus, not been applied to the flow fields computed by methods other than CLG. The Yosemite flow field by Nir et al. [78] was obtained directly from the authors. Hence, no variational energy is available for the computation of the inverse energy confidence measure.

In the following, the three approaches proposed in this chapter will be abbreviated by  $pcaRecon$  for the measure based on linear subspace projections,  $pVal$  for the linear statistical confidence measure and  $pValNonlin$  standing for its nonlinear extension.

In order to numerically compare different measures I follow the comparison method suggested by Bruhn et al. in [26] called “sparsification”, which is based on quantile plots. To this end, a specific fraction of the flow vectors (indicated on the horizontal axis in the following figures) is removed from the flow field in the order of increasing confidence and compute the average error of the remaining flow field. Hence, removing fraction 0 means that all flow vectors are taken into account, so the value corresponds to the average error over all flow vectors. Removing fraction 1 indicates that all flow vectors have been removed from the flow field yielding average error 0.

For some confidence measures, the average error even increases after removing a certain fraction of the flow field. This is the case if flow vectors with errors below the average error are removed instead of those with the highest errors. As a benchmark, I also calculate an “optimal confidence”,  $c_{opt}$ , which reproduces the correct rank order of the flow vectors in terms of the chosen error measure  $e$  and, thus, indicates the optimal order for the sparsification of the flow field:

$$c_{opt}(\mathbf{x}) = 1 - \frac{e(\mathbf{x})}{\max\{e(\mathbf{y}) | \mathbf{y} \in D\}} . \quad (7.24)$$

For the experiments the patch size was not optimized but kept constant at  $3 \times 3 \times 1$  for all test sequences, where  $3 \times 3$  stands for the spatial and 1 for the temporal dimension.

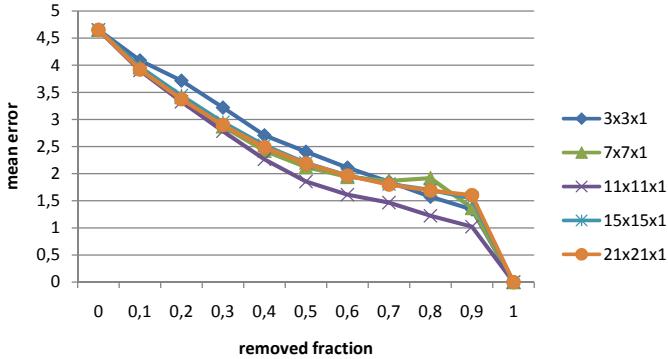


Figure 7.5: Remaining mean error for given fraction of removed flow vectors based on different patch sizes for the proposed confidence measure (Farnebäck method on Rubber Whale sequence, trained on ground truth data). The results show that the patch size chosen for the confidence measures is rather negligible.

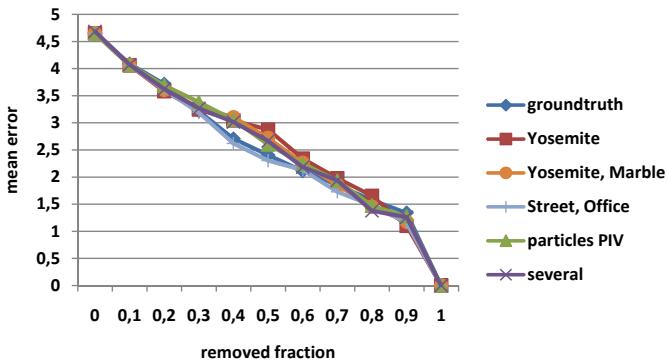


Figure 7.6: Remaining mean error based on different training sequences for the proposed confidence measure (Farnebäck method on Rubber Whale sequence for  $3 \times 3 \times 1$  patch size). The results show that the methods are hardly sensitive to the choice of training data.

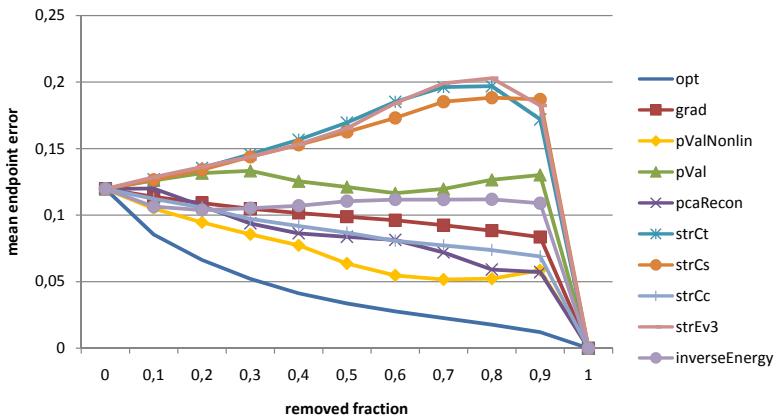
The influence of this parameter is rather negligible as shown in Figure 7.5. Figure 7.6 shows that the performance of the confidence measures is also mostly independent of the training data. In case that ground truth or similar training data is used the performance is improved, but even particle sequence data yields results close to ground truth data.

Quantile plots of the average flow field error for the state-of-the-art flow computation method by Nir et al. [78], Farnebäck et al. [37], the nonlinear 2D CLG method [28] and the structure tensor method [16] have been computed for the Rubber Whale sequence proposed in [8] as well as for the standard Yosemite and Marble test sequences. Selected results are shown in Figure 7.7.

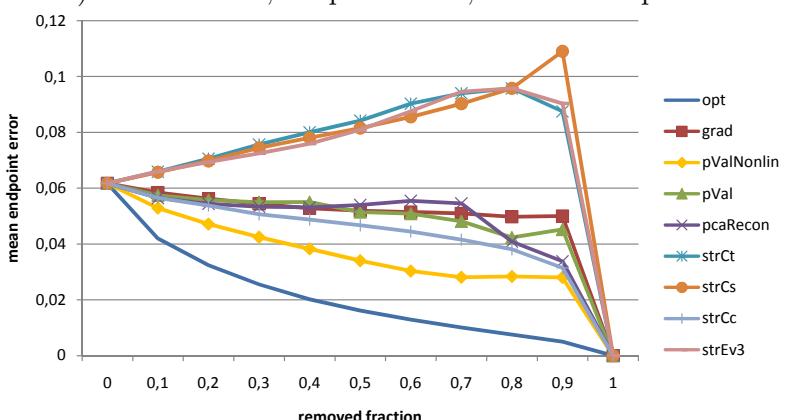
For nearly all test examples the results indicate that the remaining average error for almost all fractions of removed flow vectors is lowest for the proposed confidence measures. As confidence measures are applied to remove the flow vectors with the highest errors only, the course of the curves is most important for small fractions of removed flow vectors and can in practice be neglected for larger fractions. The results indicate that at least one of the proposed confidence measures outperforms the previously employed measures for locally and globally computed optical flow fields on all test sequences.

When looking at the result plots in Figure 7.7 it becomes apparent that the original and the nonlinear version of the proposed statistical measure do not perform equally well. In case the endpoint error is used as error measure the nonlinear version yields mostly better results. In case the results are based on the angular error the original linear confidence measure often performs better. The reason for this could be that the nonlinear polynomials increase vector components of large magnitudes. These usually also yield larger endpoint errors, since this error measure is absolute, which means that it depends on the ground truth length.

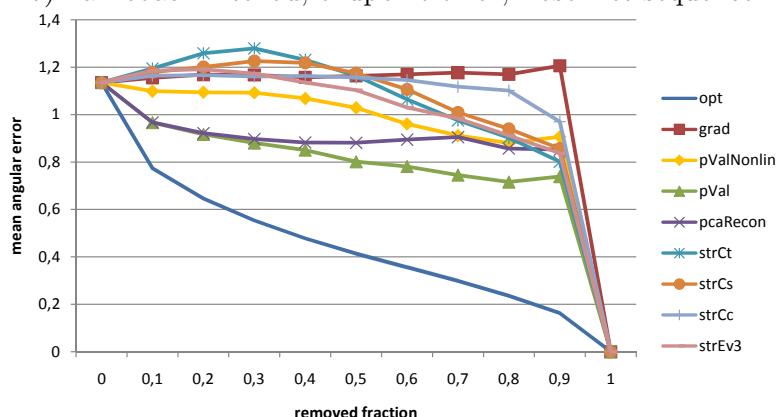
It should also be noted that for a flow field density of 90% the average error of the local structure tensor method is already lower than that of the CLG flow fields for 100% density on the Marble and Yosemite test sequences. If the CLG flow field is sparsified to 90% as well, the error is approximately equal to that of the structure tensor method for the Yosemite sequence and only half of that of the CLG method at 90% for the Marble sequence. Yet, the structure tensor approach only needs a fraction of the computation time of the CLG method and is much simpler to implement. Hence, for the local structure tensor method in two out of three cases I was able to obtain a flow field of 90% density of a quality level equal or better than that of the CLG method by means of the proposed linear confidence measure, which clearly shows the benefit of the suggested approaches.



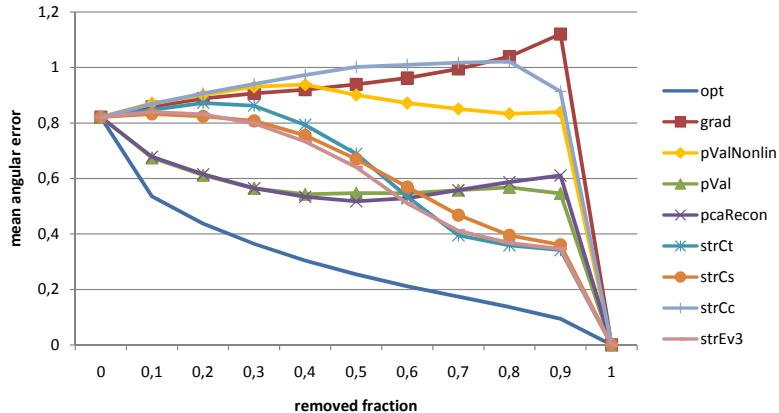
a) CLG method, endpoint error, Yosemite sequence



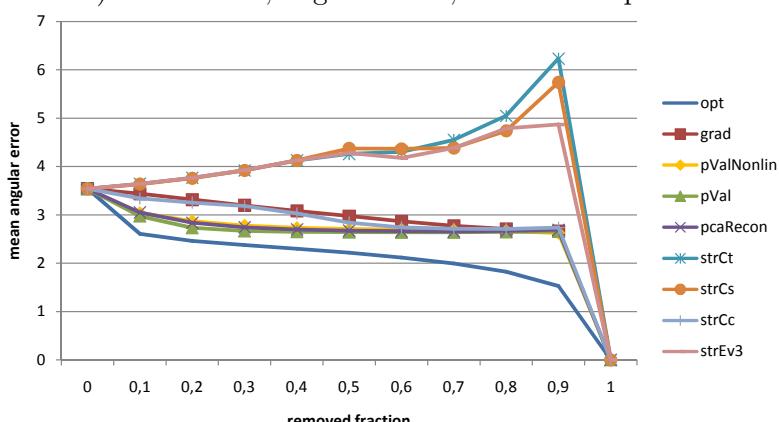
b) Farnebäck method, endpoint error, Yosemite sequence



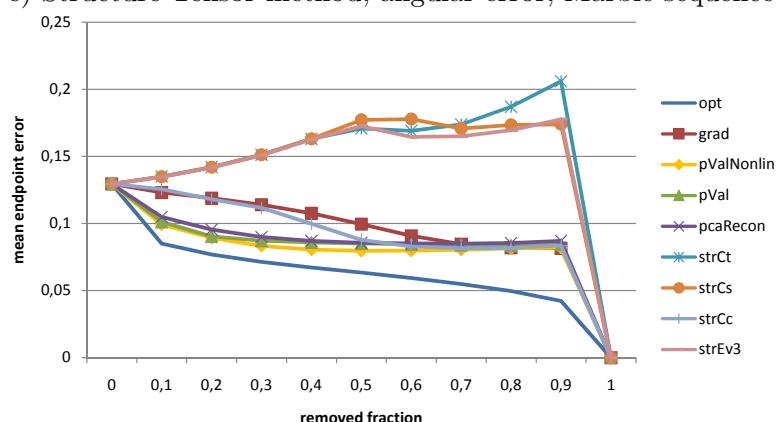
c) Farnebäck method, angular error, Yosemite sequence



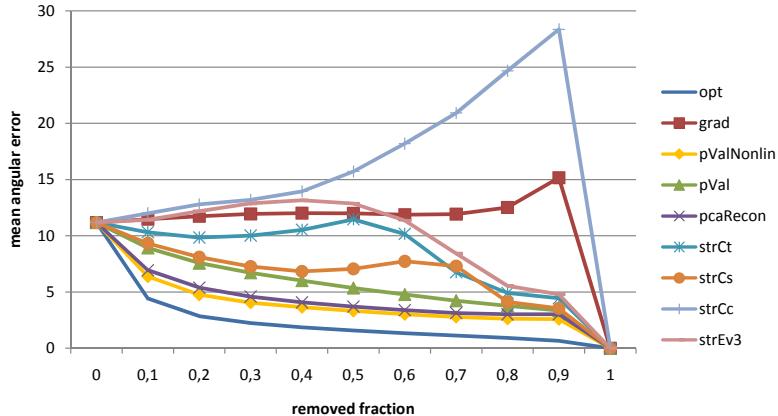
d) Nir method, angular error, Yosemite sequence



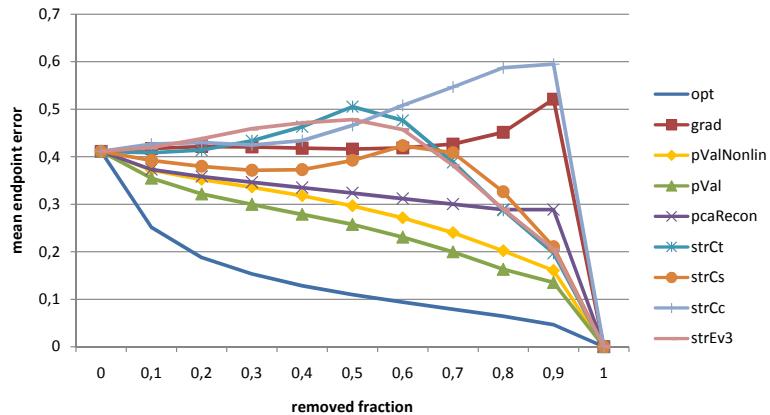
e) Structure Tensor method, angular error, Marble sequence



f) Structure Tensor method, endpoint error, Marble sequence



g) Structure Tensor method, angular error, Rubber Whale sequence



h) Horn-Schunck method, endpoint error, Rubber Whale sequence

Figure 7.7: Average error quantile plots based on different optical flow methods and error measures for the comparison of previous confidence measures (*strCt*, *strCs*, *strCc*, *grad*, *inverseEnergy*) to the proposed methods (*pcaRecon*, *pVal*, *pValNonlin*) and the optimal confidence defined in (7.24) (*optConf*). The horizontal axis indicates the fraction of removed flow vectors, the vertical axis the mean error of the remaining flow field.

To graphically compare confidence measure results I use the structure tensor flow field computed on the Rubber Whale test sequence based on the angular error as example, as here the difference between the proposed confidence measure and the previously used ones is most eminent. As the scale of confidence measures is not unique I again only compare the order of removal of the flow vectors based on increasing confidence. Hence, each flow vector is assigned the time step of its removal from the field. The resulting orders for three of the confidence measures is shown in Figure 7.8.

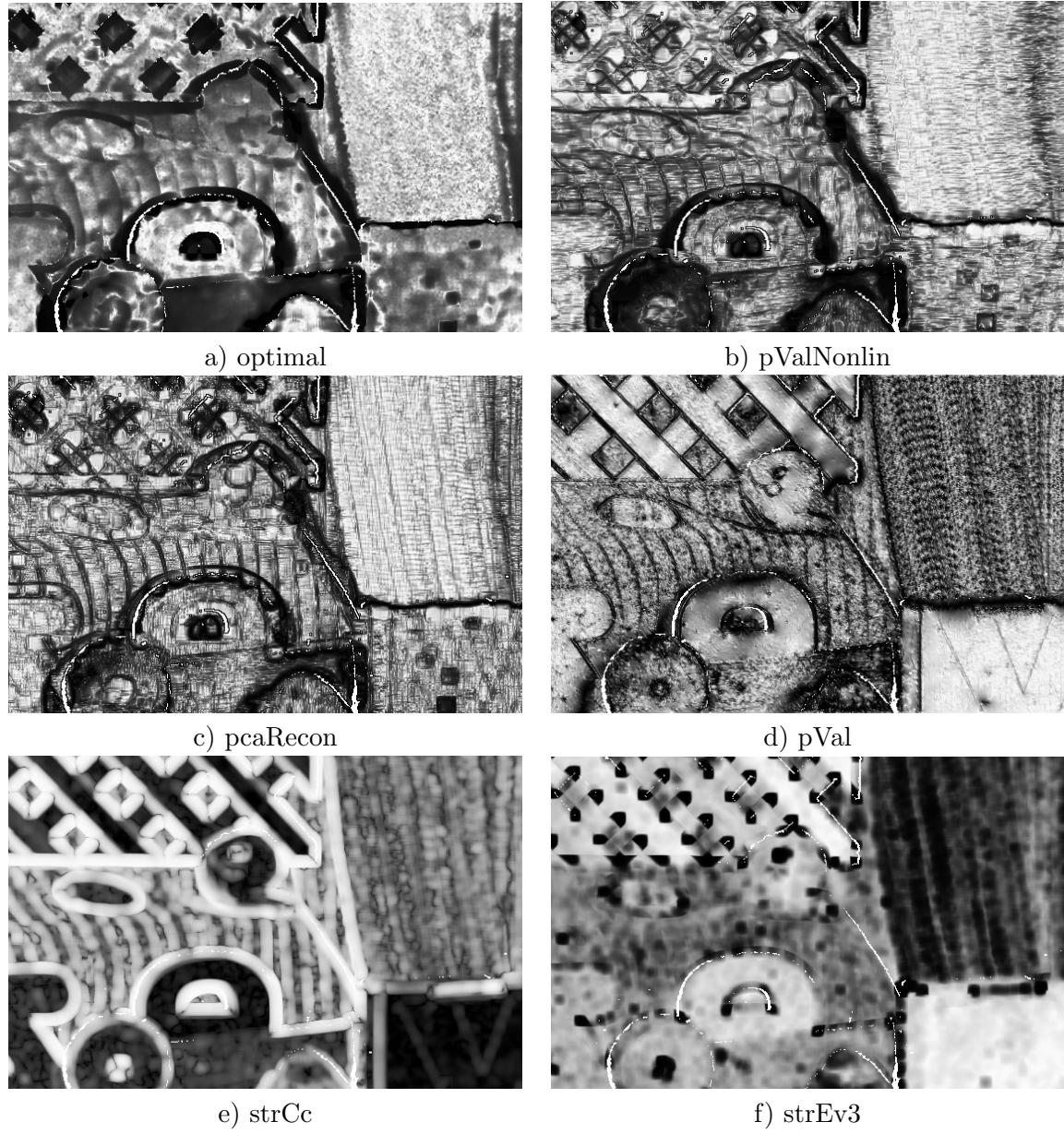


Figure 7.8: Sparsification order of flow vectors based on increasing confidence value for the structure tensor flow field on the Rubber Whale sequence based on the angular error. The proposed confidence measures (pValNonlin, pVal, pcaRecon) are closest to the optimal confidence.

To examine the results of the second hypothesis test examining  $H_1$ , it is applied to three sample flow fields: the Nir flow field on the Yosemite sequence, the CLG flow field on the Marble sequence and the structure tensor flow field on the Rubber Whale sequence. What we expect is that flow edges and regions with large, inconsistent motion vector fields are detected as unreliable inputs for confidence estimation and, thus, for the  $H_0$  test. The results are shown in Figure 7.9 and confirm our expectations. Hence, the applicability test is suitable to detect inconsistent flow field patches, which prevent reliable confidence estimates.

Finally, I show results for the sparse variant of the pVal confidence measure. Figure 7.10 shows a car sequence with sparse flow field and the color coded result of the confidence measure. Even though there is no ground truth available and, thus, no numerical results, the results show that improbable flow vectors are marked by low confidence values.

## 7.8 Summary and Conclusion

In this chapter I have proposed three confidence measures. All three are based on the learned first and second order moments of the flow field patch distribution. The first measure assigns confidence values based on the distance of the original vector and its projection into a linear subspace computed by principal component analysis. The second measure carries out a statistical hypothesis test and uses p-values in order to assign confidence values. The third measure is, in fact, an extension of the second measure, which integrates nonlinear interrelations between different flow vector components by means of polynomials. In combination with the hypothesis test I also proposed to examine the confidence measure's applicability. To this end I formulated an additional hypothesis test, which assesses the consistency of flow field patches. The results show that especially flow edges and regions with inconsistent flow are detected as unreliable for confidence estimation. In this way, I finally obtained a method consisting of two sequential hypothesis tests: the first test,  $H_1$ , estimates the applicability of the following confidence test,  $H_0$ , which is only reliable in case  $H_1$  yields high results.

All measures and their extensions are generally applicable to arbitrarily computed optical flow fields. As the measures are based on the computation of motion statistics from sample data, they are to the best of my knowledge the first confidence measures for optical flows, for which the notion “confidence measure” is in fact justified in a statistical sense. Slight changes in the algorithm also allow for an application to non-dense flows, which often occur, e.g. in traffic sequences. Results for locally and globally computed flow fields on ground truth test sequences based on different error measures show the superiority of the suggested method compared to previously employed confidence measures.

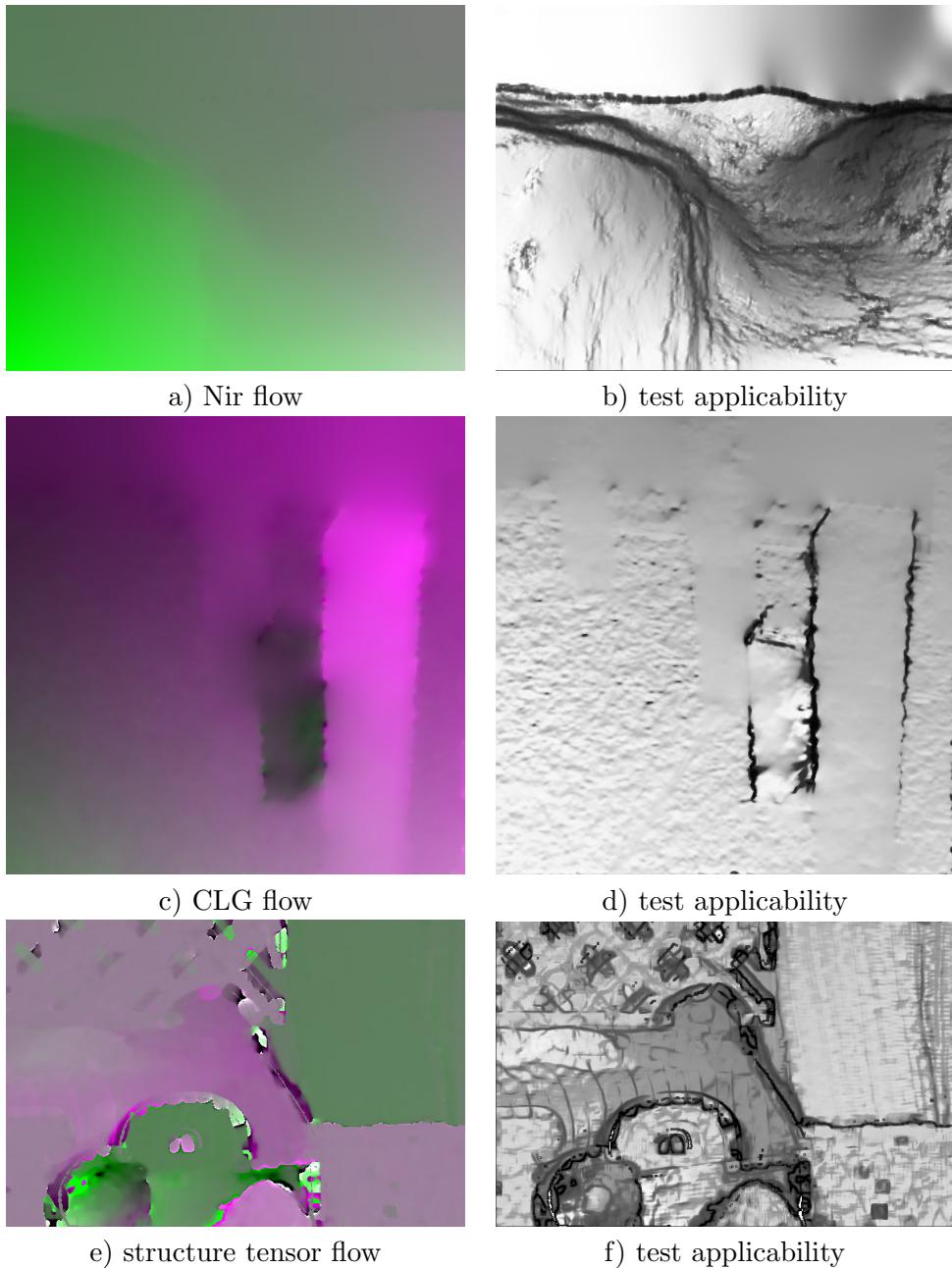
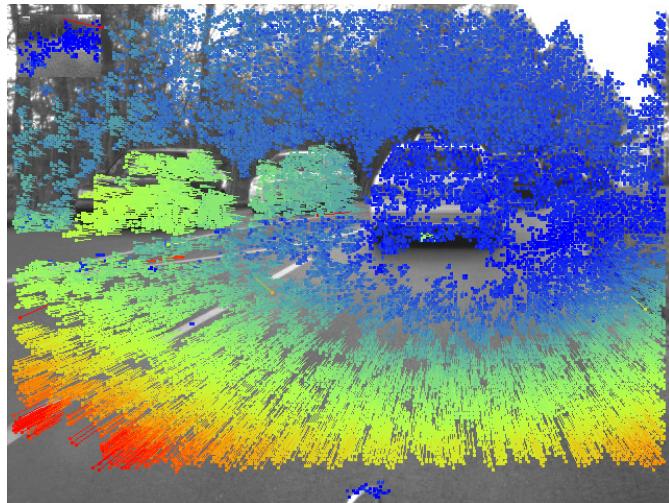
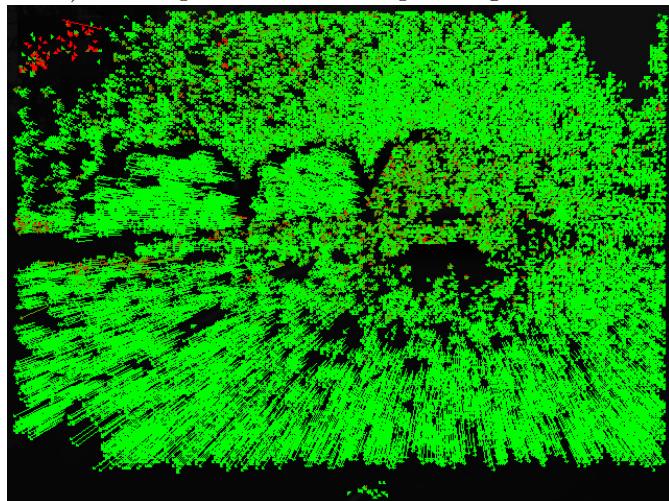


Figure 7.9: Results of  $H_1$  hypothesis test, which examines the applicability of the  $H_0$  test. Especially flow edges and difficult flow regions are detected. In these cases the result of the  $H_0$  test, and, thus, the computed confidence, is unreliable.



a) Car sequence and corresponding flow field



b) Result of sparse confidence measure

Figure 7.10: Result of the pVal confidence measure applied to a sparse flow field. a) Original car sequence with computed flow, b) Confidence measure result (green: high confidence, red: low confidence).

# Chapter 8

---

## A Model Based Optical Flow Algorithm

### 8.1 Introduction

Every confidence measure builds on some idea on what a correct optical flow field should look like. Therefore, confidence measures and optical flow estimators are highly related, and most confidence measures already contain constraints for a new optical flow computation method. In this chapter I employ the original idea used for the subspace projection based confidence measure in Chapter 7 to propose a new optical flow estimation method. The confidence measure is based on the idea that computed flow field patches can be expressed as a linear combination of typical, learned basis flows. Following the same line of thought the coefficients of these basis flows can be estimated based on the brightness constancy constraint. In this way, we obtain a highly accurate optical flow estimation method, which can be easily implemented and parallelized.

#### 8.1.1 Motivation

Optical flow refers to the displacement field between subsequent frames of an image sequence. Methods for the computation of the optical flow are usually tradeoffs between speed, accuracy and implementation effort. Local methods such as the Lucas/Kanade approach [70] or the structure tensor approach by Bigün [16] are fast and easy to implement but not very accurate. Global methods such as the method by Horn and Schunck [53], Bruhn et al. [28], Brox et al. [25], Papenberg et al. [81] and Nir et al. [78] are much more accurate and can even be applied in realtime by means of multigrid methods [27], yet with considerable implementation effort. Farnebäck [36] proposed a local method which is accurate and fast, but also rather complex to implement due to its rule-based image segmentation scheme. Hence, in this chapter I propose to extend the

method by Black et al. [23] in order to obtain a local optical flow method, which ranges among the most accurate methods today and at the same time is fast, yet simple to implement. Furthermore, the suggested method relies on natural motion statistics and is, thus, adaptable to specific motion patterns occurring e.g. in fluid dynamics or driver assistance systems. Finally, the original subspace projection confidence measure (Chapter 7) is directly inherent to this flow computation method and can easily be applied yielding considerable improvements of the flow field.

### 8.1.2 Related Work

There is a number of local methods for optical flow computation today. Because the optical flow problem is underdetermined, all these methods involve additional assumptions on the structure of the motion, i.e. they are based on a model of the flow. Variational methods incorporate such models e.g. in regularization terms [28, 81, 25]. Local methods explicitly model assumptions on the flow vectors within spatio-temporal neighborhoods. Lucas/Kanade [70] and Bigün [16] assume that the velocity is constant for a local neighborhood centered on the current pixel. This leads to an overdetermined system of equations, which can be solved by the least squares (section 2.6) or total least squares method. A more robust method for solving the overdetermined system, the least median of squares approach, has been proposed by Bab-Hadiashar and Suter [6]. Yet, the model of piecewise constant motion is not adequate for most image sequences. Hence, other methods assume more general models like constant or affine models [58, 19, 36], local planar models [18] or physics based models [46, 50]. An overview can be found in [94].

Flow fields based on such models are often more accurate than those based on assumptions of constant flow. However, there are situations where more complex models would be necessary to compute accurate flow fields, e.g. situations with motion discontinuities and transparent motion, which have been addressed by Barth et al. [13]. General affine models have been integrated into a variational framework by Nir et al. [78]. In case of even more complex situations learning motion models from given sample motion data is a method to obtain superior results. Roth and Black [85] employ a general learning based approach using fields of experts, which is integrated into a global optical flow method. Here, learning is not adapted to special image sequences. In contrast, Black et al. [23] as well as Yacoob and Davis [100] integrate adapted, learning based models into a local optical flow method. To learn these models, principal component analysis (PCA) is used. This leads to a nonlinear energy functional, which is linearized and minimized by means of a coarse-to-fine strategy and coordinate descent. However, the models employed are either purely spatial [23] or purely temporal [100].

Our approach differs in five main aspects from these two methods:

- (a) Instead of formulating a non-linear energy functional I obtain an overdetermined system of equations, which can be solved by established least squares methods

instead of performing gradient descent,

- (b) Spatio-temporal instead of purely spatial or temporal motion models are employed, which can represent complex motion patterns over time,
- (c) I show highly accurate results comparable to Farnebäck's, but with much less effort, for test sequences typically used in optical flow computation,
- (d) By means of a model-based confidence measure directly inherent to the flow computation method the resulting flow field can be sparsified yielding significantly lower angular errors. Via interpolation or inpainting after sparsification it is also possible to reconstruct a dense flow field with lower angular error [62].
- (e) I additionally integrate the learned motion model into a global optical flow approach, which could be understood as a learning based extension of the method by Nir et al. [78].

Based on the simple structure tensor method, we end up with a simple, fast, parallelizable and accurate optical flow method, which is able to incorporate learned prior knowledge on special types of motion patterns and, thus, can be adapted to all kinds of motion estimation problems.

### 8.1.3 Contribution

In this chapter I use the original idea of the subspace projection confidence measure to formulate a new optical flow estimator. Following Nir [78] the optical flow vectors are not estimated explicitly. Instead, a parameter vector is estimated which contains the coefficients of the principal components spanning the learned subspace of typical motion constellations in flow field neighborhoods. Thus, every flow vector is expressed as a linear combination of learned basis flow field patches. In this way, spatio-temporal motion models learned from sample data can be employed to constrain the optical flow field. Using the brightness constancy constraint equation (3.1), we end up with an overdetermined system of equations, which can be solved efficiently. Part of this work has been submitted [64].

## 8.2 Parameter Estimation

I again use the learned motion model proposed in Chapter 7 consisting of the mean vector  $\mathbf{m}$  and the principal components  $\mathbf{b}_j, j \in \mathbb{N}_k$ , in (7.4) learned from sample data. Instead of estimating the optical flow itself we want to estimate the corresponding coefficients  $\alpha_j, j \in \mathbb{N}_k$ , of the principal components, which implicitly define the displacement field for the neighborhood of the current pixel. In this way, the resulting optical flow is restricted to the learned subspace spanned by the  $k$  principal components. In the sense

of Nir et al. [78] one could speak of an over-parameterized model, since we estimate  $k$  coefficients to obtain a two-dimensional flow vector, which is chosen from the center of the flow field patch defined by the corresponding parameters.

To estimate the coefficients  $\alpha_j, j \in \mathbb{N}_k$ , we can solve an overdetermined system of equations. For a given flow vector and its spatio-temporal neighborhood I make two assumptions:

- (a) The flow field patch can be represented as a linear combination of principal components,
- (b) Each of the flow vectors within the patch fulfills the brightness constancy constraint equation (3.1) as explained in Chapter 3.

For a given pixel position  $(\mathbf{x}, t) \in D$  let  $I_x, I_y, I_t \in \mathbb{R}^{\frac{p}{2}}, p = 2|\omega|(2\tau + 1)$  denote the vectorized image derivatives with respect to  $x, y$  and  $t$  within the spatio-temporal flow field patch  $\omega(\mathbf{x}) \times [t - \tau, t + \tau]$  centered on  $(\mathbf{x}, t)$ . Then the assumptions (a) and (b) can be combined by substituting  $\mathbf{u}$  in (3.1) by the linear combination of basis flows. For each  $\mathbf{x}' \in \omega(\mathbf{x}), t' \in [t - \tau, t + \tau]$  we, thus, obtain one equation of the following form:

$$\begin{pmatrix} \sum_{j=1}^k \alpha_j \mathbf{b}_{jr} + \mathbf{m}_r \\ \sum_{j=1}^k \alpha_j \mathbf{b}_{js} + \mathbf{m}_s \end{pmatrix}^T \cdot \begin{pmatrix} I_{xr} \\ I_{yr} \end{pmatrix} = -I_{tr} \quad (8.1)$$

where  $r := q(\mathbf{x}', t', 1), s := q(\mathbf{x}', t', 2)$  (see 7.3) .

We can rewrite these  $\frac{p}{2}$  equations in matrix formulation:

$$L \alpha = -\mathbf{d}, \quad (8.2)$$

$$L \in \mathbb{R}^{\frac{p}{2} \times k}, \mathbf{d} \in \mathbb{R}^{\frac{p}{2}}, \alpha = (\alpha_1, \dots, \alpha_k)^T .$$

with  $L$  and  $\mathbf{d}$  defined as follows. Under the assumption that the first  $\frac{p}{2}$  entries of the principal components  $\mathbf{b}_i, i \in \mathbb{N}_k$ , correspond to the horizontal components of each flow vector and the remaining entries to the vertical components, the  $j$ -th column of  $L$  denoted by  $l_j$  and the vector  $d$  are given by

$$l_j := \begin{pmatrix} I_{x1}\mathbf{b}_{j1} + I_{y1}\mathbf{b}_{j(\frac{p}{2}+1)} \\ \vdots \\ I_{x\frac{p}{2}}\mathbf{b}_{j\frac{p}{2}} + I_{y\frac{p}{2}}\mathbf{b}_{jp} \end{pmatrix}, j \in \mathbb{N}_k, \quad (8.3)$$

$$\mathbf{d} := \begin{pmatrix} I_{x1}\mathbf{m}_1 + I_{y1}\mathbf{m}_{\frac{p}{2}+1} + I_{t1} \\ \vdots \\ I_{x\frac{p}{2}}\mathbf{m}_{\frac{p}{2}} + I_{y\frac{p}{2}}\mathbf{m}_p + I_{t\frac{p}{2}} \end{pmatrix}. \quad (8.4)$$

Many methods are available to solve this overdetermined system of equations for the vector  $\alpha$ . One way is to use the simple and efficient least squares method (see section 2.6)

$$\alpha = -(L^T L)^{-1} L^T \mathbf{d} , \quad (8.5)$$

which already yields results of high accuracy. As the integration area of this method is large for spatially and/or temporally large principal components the accuracy in the vicinity of motion boundaries may be lower. Yet, to handle outliers, the original least squares method is inappropriate.

In such cases, I propose to use the more robust least median of squares approach by Rousseeuw [86, 87] applied to optical flow estimation by Bab-Hadiashar and Suter [6]. Here, the basic idea is to randomly and repeatedly choose a subset of equations from the original system that contains as many equations as unknowns and can, thus, be solved precisely. For each solution obtained from a random subset of equations the residual of the remaining equations is computed and, finally, the solution with the lowest residual median is chosen.

Independent of the method used to solve the overdetermined system of equations the resulting parameter vector  $\alpha$  finally represents the optical flow within a spatio-temporal neighborhood, from which the central vector is chosen as displacement vector for the current image sequence location.

## 8.3 Confidence Estimation

Since the principal components have already been computed for the flow estimation method, they can be employed to estimate the confidence of the computed flow vectors using the linear subspace projection confidence measure described in Chapter 7.2.4. Based on this confidence measure vectors with high errors can be removed from the flow field and, if a dense flow field is necessary, reconstructed afterwards as demonstrated in chapter 9.

## 8.4 Integration of the Model into a Global Optical Flow Method

In 2005 Bruhn et al. [28] proposed to introduce the original structure tensor method into a global framework and came up with the very well-known CLG method. Since I proposed an extension to the structure tensor by introducing learned motion models I will show that the proposed structure tensor extension can also be introduced into a global framework, thus extending the original CLG method to incorporate motion

models and to estimate parameter vectors as done by Nir [78]. Let

$$\tilde{\alpha} = (\alpha_1, \dots, \alpha_k, 1)^T, \quad (8.6)$$

$$\tilde{L} = (L \mathbf{d}), \text{ (see 8.3),} \quad (8.7)$$

$$J = \tilde{L} \tilde{L}^T. \quad (8.8)$$

Then the following energy is minimized, which is similar to that proposed by Bruhn et al. [28]. In contrast to their method, I apply the regularizer to the parameters  $\alpha$  as done by Nir et al. [78] instead of to the flow vectors themselves.

$$E(\alpha) = \int_D \psi_1(\alpha^T J \alpha) + \lambda \psi_2(\|\nabla \alpha\|^2) dx dy dt, \quad \lambda \in \mathbb{R}_+. \quad (8.9)$$

Here  $\psi_1$  and  $\psi_2$  stand for outlier functions, which allow for nonlinearities in the flow field, e.g. at motion boundaries. I use an outlier function proposed by Charbonnier et al. [31]:

$$\psi(s^2) = 2\beta_i^2 \sqrt{1 + \frac{s^2}{\beta_i^2}}, i \in \{1, 2\}. \quad (8.10)$$

Via calculus of variations (see Chapter 2.1), this leads to  $k$  Euler-Lagrange equations, for  $r \in \mathbb{N}_k$ :

$$\psi'_1(\alpha^T J \alpha) \left( \sum_{j=1}^k J_{r,j} \alpha_j + J_{r,k+1} \right) - \lambda \operatorname{div}(\psi'_2(\|\nabla \alpha\|^2)) = 0. \quad (8.11)$$

The solution of this nonlinear system of equations corresponds to a minimum of the original energy functional. In order to compute derivatives the central differences scheme is used for spatial derivatives, and the forward differences scheme for temporal derivatives. To improve the computation of image derivatives the scene is first smoothed by means of a Gaussian filter. The importance of this step for more accurate results is discussed in [28]. For optimization the multiresolution scheme with warping proposed by Brox et al. [25] is used. This method has been formulated for flow fields, not for parameter maps as in this case. The scheme can be adapted to parameter maps by making two changes: First, to scale parameter maps between different pyramid levels, we cannot scale the parameters themselves but we need to compute the flow corresponding to the parameters, then scale the flow and transform the scaled flow back to the parameter space. Let  $B := (\mathbf{b}_1, \dots, \mathbf{b}_n)$ , then the following function  $f$  scales the flow given by the parameter vector  $\alpha$  by a factor  $s$

$$f : \mathbb{R}^k \times \mathbb{R} \rightarrow \mathbb{R}^k \quad (8.12)$$

$$f(\alpha, s) = B^T(s(B\alpha + \mathbf{m}) - \mathbf{m}). \quad (8.13)$$

Second, the initial guess for each level is not zero but the parameters,  $\alpha_0$ , which correspond to a zero flow field at this level:

$$\alpha_0 = -B^T \mathbf{m}. \quad (8.14)$$

With these two changes, the multiresolution scheme with warping can easily be applied to parameter maps, as well. For each new pyramid level the current complete solution in the parameter space,  $\mathbf{p}_0$ , is scaled to the current level by means of the function  $f$ . Then the image is warped by the flow corresponding to the scaled parameters. With  $\alpha_0$  as initial guess the optical flow problem is solved for the warped sequence in the parameter space on the current level of the pyramid. Finally, the solution is scaled to the original size by the function  $f$  and added to the previous solution  $\mathbf{p}_0$ . Then the procedure is repeated for the next level. More detailed information on pyramids with warping is given in [81].

## 8.5 Results

In this section I present results on the accuracy, efficiency and adaptability of the proposed optical flow method. For the implementation of the proposed local method the filters optimized for optical flow by Scharr [89] are used to estimate derivatives. Usually, I use size 7x7x7 if the length of the image sequence permits it. Furthermore, all sequences are presmoothed by a Gaussian filter with spatial  $\sigma = 0.8$ . For the computation of the principal components 5000 samples were randomly selected from the training sequences.

### 8.5.1 Accuracy

In order to evaluate the accuracy of the proposed optical flow method several experiments were conducted on ground truth data. Here, the Yosemite, the Marble and the Rubber Whale sequence from the Middlebury dataset [8] as well as the Street and Office sequence [73] shown in Figure 8.1 were used.

#### Quality of the Estimator

A comparison of the angular error and standard deviation to previously proposed local and global optical flow methods for the Yosemite sequence is shown in Table 8.1. Among them can be found the method by Roth and Black [85] who also obtained very good results by learning statistical motion models from sample data and integrating this model into a variational approach. Yet, their algorithm is rather complex, and for the Yosemite sequence they trained their model on ground truth data. Hence, the proposed method is preferable due to accuracy, speed and straightforwardness. In fact, the results obtained are as accurate as the relatively involved method by Farnebäck [36], yet with a much lower standard deviation of 1.45 compared to 2.57. It is even more accurate than many global methods such as the combined local global method by Bruhn et al. [28]. Yet, in contrast to most other methods, it is also simple to implement and adaptable to different types of motion.

This approach was, furthermore, applied to the Marble and Rubber Whale sequence.



Figure 8.1: One frame of the Yosemite, the Marble, the Rubber Whale, the Street and the Office sequence.

Comparison to Other Approaches		
Method	Ang. Err.	Std
Black & Anandan [22]	4.46	4.21
Bruhn et al. (2d CLG linear) [28]	2.64	2.27
Black & Jepson [21]	2.29	2.25
Ju et al. [58]	2.16	2.0
Bab-Hadiashar & Suter [6]	1.97	1.96
Bruhn et al. (2d CLG non-linear) [28]	1.79	2.34
<b>our method (trained on other gt)</b>	<b>1.53</b>	<b>1.69</b>
Roth & Black [85] (trained on gt)	1.47	1.54
Bruhn et al. (3d CLG non-linear) [28]	1.46	1.50
<b>our method (trained on HS)</b>	<b>1.45</b>	<b>1.47</b>
Farnebäck [36]	1.40	2.57
<b>our method (trained on gt)</b>	<b>1.35</b>	<b>1.45</b>
Farnebäck [37]	1.14	2.14
Papenberg et al. [81]	0.99	1.17
Nir et al. [78]	0.85	1.18

Table 8.1: Comparison of the proposed model based local motion estimator to angular error and standard deviation obtained by previously proposed local and global methods for the Yosemite sequence without clouds.

To obtain the results in Tables 8.2 and 8.3 spatial model sizes  $\omega$  between  $3 \times 3$  and  $21 \times 21$ , temporal model sizes  $\tau$  between 0 and 3 and numbers of principal components between 2 and 10 were tested. Table 8.4 shows the parameters used (spatial model size, temporal model size, number of principal components) in order to obtain the results in Tables 8.2 and 8.3. The values indicate that large model sizes but only between 5 and 10 or even less principal components and, thus, coefficient parameters  $\alpha_j$  are necessary to obtain good results.

### Evaluation of the Confidence Measure

In the context of Tables 8.2 and 8.3, I will also refer to the effects of the confidence measure on the angular error of the remaining flow field. The column titled “density” indicates the density of the remaining flow field, which is obtained by removing the flow vectors with lowest confidence values. The results in the table indicate that the high accuracy of the method is generally increased by the application of the confidence measure. Its computation is very simple, since the principal components have been computed before the actual motion estimation.

An important issue to investigate is the dependency of the accuracy of the proposed

Results				
Sample data	Density (%)	Yosemite	Marble	Rubber Whale
Ground truth	100	$1.35 \pm 1.45$	$2.06 \pm 3.64$	$7.87 \pm 16.12$
Ground truth	90	$1.26 \pm 1.30$	$1.60 \pm 2.78$	$5.30 \pm 10.49$
Ground truth	80	$1.14 \pm 1.22$	$1.38 \pm 2.52$	$4.45 \pm 9.55$
Ground truth	70	$1.06 \pm 1.22$	$1.27 \pm 2.36$	$4.20 \pm 9.83$
Computed	100	$1.45 \pm 1.47$	$2.32 \pm 3.95$	$7.87 \pm 16.14$
Computed	90	$1.32 \pm 1.21$	$1.68 \pm 2.60$	$5.31 \pm 10.52$
Computed	80	$1.19 \pm 1.05$	$1.30 \pm 1.85$	$4.44 \pm 9.52$
Computed	70	$1.11 \pm 0.97$	$1.10 \pm 1.45$	$4.20 \pm 9.82$
Other GT	100	$1.53 \pm 1.69$	$2.55 \pm 4.25$	$7.85 \pm 15.95$
Other GT	90	$1.37 \pm 1.43$	$1.87 \pm 2.73$	$5.24 \pm 10.43$
Other GT	80	$1.24 \pm 1.37$	$1.49 \pm 2.05$	$4.36 \pm 9.45$
Other GT	70	$1.15 \pm 1.38$	$1.27 \pm 1.65$	$4.12 \pm 9.75$

Table 8.2: Angular error and standard deviation for different sample data selections (ground truth flow of the same sequence, Horn-Schunck flow field for the same sequence, ground truth flows of other sequences) and densities after sparsification based on the confidence measure in [65] for the Yosemite, Marble and Rubber Whale sequence.

Results			
Sample data	Density (%)	Street	Office
Other GT	100	$4.99 \pm 13.72$	$3.83 \pm 4.98$
Other GT	90	$3.65 \pm 8.38$	$3.35 \pm 3.85$
Other GT	80	$3.04 \pm 6.05$	$3.01 \pm 3.38$
Other GT	70	$2.44 \pm 4.52$	$2.75 \pm 3.25$

Table 8.3: Angular error and standard deviation for ground truth sample data taken from other sequences, and densities after sparsification based on the confidence measure in [65] for the Street and Office sequence.

method on the choice of parameters for a) the sample data for the computation of the motion statistics PCA model, b) the size of the model and c) the number of principal components.

### Dependency on Sample Data

For the investigation of the dependency on the sample data I used sample data taken from

Parameters			
Sequence	$\omega$	$\tau$	k
Yosemite (ground truth)	$21 \times 21$	3	7
Yosemite (computed)	$19 \times 19$	2	6
Yosemite (other)	$19 \times 19$	3	10
Marble (ground truth)	$21 \times 21$	3	9
Marble (computed)	$21 \times 21$	5	7
Marble (other)	$19 \times 19$	7	6
Rubber Whale (ground truth)	$19 \times 19$	0	2
Rubber Whale (computed)	$19 \times 19$	0	2
Rubber Whale (other)	$19 \times 19$	1	2
Street (other)	$19 \times 19$	3	2
Office (other)	$21 \times 21$	1	5

Table 8.4: Model parameters (spatial model size  $\omega$ , temporal model size  $\tau$ , number of principal components  $k$ ) used to obtain the results in Tables 8.2 and 8.3. Note that for the Rubber Whale sequence the ground truth flow only contains a single frame and, thus, limits  $\tau$  to 0 in the training process.

- (a) the specific ground truth flow field,
- (b) a flow field computed by the Horn-Schunck method [53],
- (c) ground truth flow fields of other sequences.

For the computation of the Horn-Schunck flow fields used for (b) a multiresolution approach was employed yielding angular errors of  $2.21 \pm 2.83$  on the Yosemite sequence,  $4.90 \pm 2.74$  on the Marble sequence and  $12.05 \pm 16.25$  on the Rubber Whale sequence. For (c) the other ground truth sequences are the Marble, Street and Office sequence in case we want to estimate the optical flow of the Yosemite sequence, and the Yosemite, Street and Office sequence in case we want to estimate the optical flow of the Marble sequence, and from all other four sequences if we want to estimate the flow of the Rubber Whale sequence. These training sequences have been chosen because their ground truth flow fields are readily available.

Table 8.2 compares the angular error and standard deviation obtained for three test sequences for the different flow field densities, if the PCA model is trained on ground truth data, a computed Horn-Schunck flow field or other ground truth sequences. From the results in Table 8.2 we can draw the conclusion that best results are usually obtained from accurate prior knowledge as expected. However, even in the absence of such knowledge learning from optical flow estimates (e.g. Horn & Schunck) yields results of almost equal quality. Even in the case of learning from generic motion fields still competitive

results can be obtained. For the Street and Office sequences the choice of training data is limited to ground truth sequences taken from other scenes in Table 8.3.

### Dependency on Model Size

Concerning the sensitivity of the method with respect to different model sizes and numbers of principal components Tables 8.5 and 8.6 show the dependency of the method on the chosen size  $(\omega, \tau)$  of the PCA model based on the Yosemite sequence for ground truth sample data and based on the Marble sequence for Horn-Schunck data, respectively, for 7 principal components. The results suggest that larger model sizes yield lower angular errors.

Dependency on model size			
$\omega \setminus \tau$	0	1	3
5 × 5	7.12 ± 12.76	4.72 ± 7.62	3.01 ± 3.55
9 × 9	3.93 ± 6.46	2.69 ± 3.49	2.12 ± 2.27
15 × 15	2.39 ± 3.01	1.81 ± 1.97	1.50 ± 1.70
21 × 21	1.79 ± 1.87	1.66 ± 1.57	1.35 ± 1.45

Table 8.5: Angular error and standard deviation for different spatio-temporal sizes  $(\omega, \tau)$  for the Yosemite sequence trained on ground truth data using 7 principal components.

Dependency on model size			
$\omega \setminus \tau$	0	1	3
5 × 5	9.74 ± 14.10	6.68 ± 10.29	6.49 ± 9.76
9 × 9	6.61 ± 9.60	5.05 ± 8.11	3.42 ± 6.52
15 × 15	5.01 ± 5.85	3.24 ± 4.78	2.54 ± 5.12
21 × 21	4.10 ± 3.78	2.90 ± 3.24	2.42 ± 3.97

Table 8.6: Angular error and standard deviation for different spatio-temporal sizes  $(\omega, \tau)$  for the Marble sequence trained on Horn-Schunck data using 7 principal components.

### Dependency on Numbers of Principal Components

Tables 8.7 and 8.8 show the results for different numbers of principal components for the Yosemite sequence trained on ground truth data and for the Marble sequence trained on Horn-Schunck data for the model size of  $\omega = 21 \times 21, \tau = 3$ . The error values suggest that this number does not have much influence on the accuracy of the flow field if at least

five or six components are chosen. This shows that most of the variance for the occurring motion patterns is already contained in the first five or six principal components.

Dependency on principal components			
k	ang. err.	k	ang. err.
2	$1.93 \pm 2.07$	7	$1.35 \pm 1.45$
3	$1.86 \pm 2.06$	8	$1.36 \pm 1.40$
4	$1.53 \pm 1.53$	9	$1.35 \pm 1.41$
5	$1.44 \pm 1.56$	10	$1.36 \pm 1.42$
6	$1.40 \pm 1.53$		

Table 8.7: Angular error and standard deviation for different numbers of principal components for the Yosemite sequence trained on ground truth using a spatio-temporal model size of  $\omega = 21 \times 21, \tau = 3$ .

Dependency on principal components			
k	ang. err.	k	ang. err.
2	$3.28 \pm 4.99$	7	$2.32 \pm 3.95$
3	$3.32 \pm 4.95$	8	$2.34 \pm 3.99$
4	$3.13 \pm 4.57$	9	$2.35 \pm 4.06$
5	$2.95 \pm 4.34$	10	$2.35 \pm 4.04$
6	$2.37 \pm 3.99$		

Table 8.8: Angular error and standard deviation for different numbers of principal components for the Marble sequence trained on the Horn-Schunck result using a spatio-temporal model size of  $\omega = 21 \times 21, \tau = 2$ .

Hence, I have shown that neither the choice of (reasonably general) sample data nor higher numbers of principal components have much influence on the results of the proposed optical flow method. In contrast, the spatio-temporal model size is important to obtain high accuracy.

## Graphical Results

The flow fields for all test sequences with 100 % density computed on ground truth data are shown in Figures 8.2 and 8.3. Apparently the accurate results of the method are especially due to the high accuracy of angular errors close to 0 for large regions without motion boundaries, e.g. the valley of the Yosemite sequence and the table of the Marble sequence. In case of motion boundaries the accuracy of the flow field is lower, especially for higher numbers of principal components.

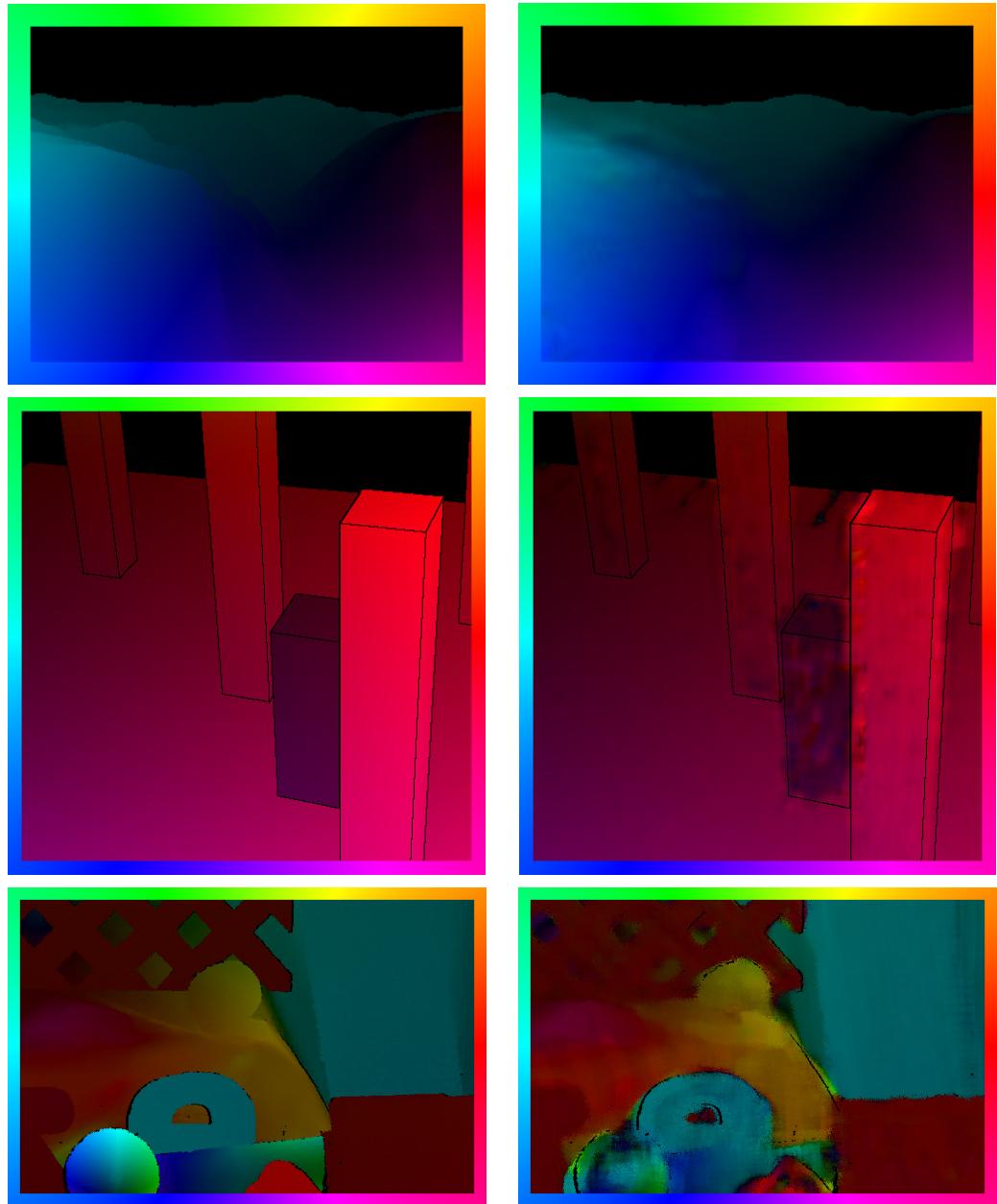


Figure 8.2: HSV-coded ground truth flow fields (left) and result of the proposed estimator (right) for the Yosemite, the Marble and the Rubber Whale sequence based on motion statistics obtained from ground truth data.

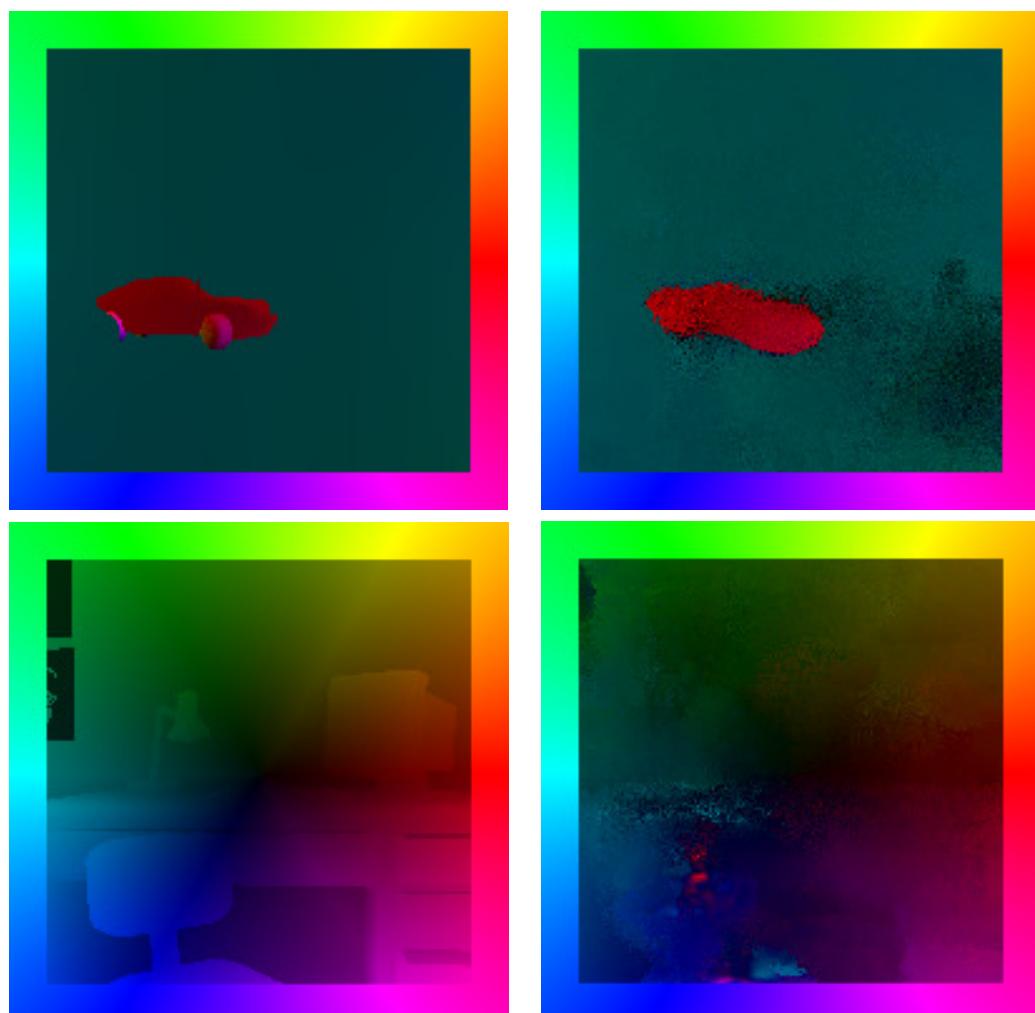


Figure 8.3: HSV-coded ground truth flow fields (left) and result of the proposed estimator (right) for the Street and the Office sequence based on motion statistics obtained from generic ground truth data of other scenes.

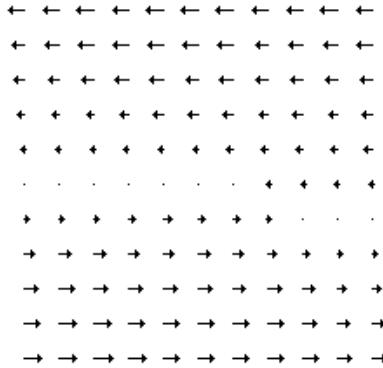


Figure 8.4: Example for a principal component representing a horizontal motion boundary. Note the slight, non-sharp transition between the speeds on both sides of the boundary.

The reason for this lies in the fact that motion boundaries are not accurately learned by the PCA model if smooth flow field patches and motion boundaries are combined in the training data and if the location of the motion boundary varies within the flow field patch. Principal components representing edges then typically show a slight, non-sharp transition between the speeds on different sides of the motion boundary. An example for such a principal component is shown in Figure 8.4.

If such principal components are contained in the set of eigenflows they are predominant in case of motion boundaries due to high coefficients. Yet, since the learned motion boundaries are not sharp the resulting computed motion boundary is not sharp, either. Hence, the method can become inaccurate at motion boundaries. In case that such principal components are not included in the set of eigenflows the least median of squares approach usually leads to sharp motion boundaries, since the pixels on one side of the motion boundary are considered as outliers.

Another reason for inaccurate estimates at motion boundaries are the derivative filters. They are spatially and temporally large and, thus, yield inaccurate image derivatives near motion boundaries, which in turn lead to inaccurate values in the linear system of equations (8.3).

Figure 8.5 illustrates the different behavior of the proposed method near motion boundaries for different numbers of eigenvectors compared to the original structure tensor method. In all cases the least median of squares method has been used to solve the system of equations. From Figure 8.5 we can deduce the following: Since the third and fourth principal component contains edge structures these components are mainly used to represent the edge in case of four principal components (8.5 b). Hence, in the center the values for the third and fourth parameters are predominant. As the learned motion boundaries are not perfectly sharp the transition is also visible in the resulting computed

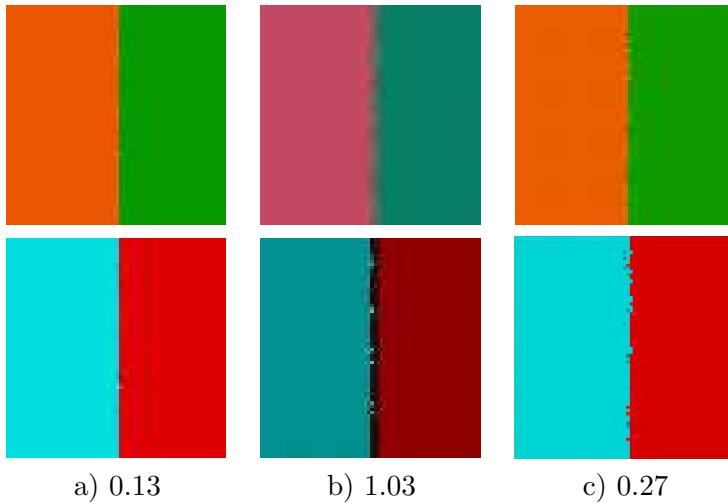


Figure 8.5: Estimation result with least median of squares method based on (a) two, (b) four eigenvectors and (c) without learned eigenvector model. Top: estimated parameters, Bottom: flow indicated by the parameters above. The values below correspond to the average angular error of the estimated flow field.

flow and parameter field. In case only two principal components are used for the flow estimation (8.5 a) the edge cannot be represented by the model, and hence the least median of squares method considers pixels across the motion boundary as outliers and ensures a correct edge. On the other hand, in case no motion model is used (8.5 c), we end up with the original structure tensor approach combined with the least median of squares method. The resulting motion field and the higher angular error shows that the edge is not as clear as in the case that motion models are used (8.5 a).

A drawback of the proposed method is that it has difficulties with very large displacements due to the linearization of the brightness constancy equation. This is a common problem for many optical flow methods relying on the linearized expression. Usually the use of a multiscale pyramid-type based approach helps to solve this problem.

The rather deceiving results for the office sequence, whose ground truth flow field only consists of a divergence, is probably due to the limitations of the training data, which was taken from various sequences without divergences. Therefore, the largest errors appear at the center of the divergence of the flow field.

### 8.5.2 Efficiency

The proposed local optical flow method can be implemented efficiently due to several reasons. First, the method only takes into account a limited local image region in order to estimate the displacement vector for each pixel. Hence, it takes only limited space and

can be easily parallelized. Second, the computation of the PCA model can be carried out once before the estimation of the optical flow and can be used for all kinds of sequences and the confidence estimation later on.

I will now analyze the speed of the suggested method more precisely. To compute the structure tensor we need to calculate several scalar products, which can be obtained by means of convolutions in order to avoid recalculations. Let

$$\mathbf{v}_i := \mathbf{b}_{i(1:\frac{p}{2})} \quad (8.15)$$

$$\mathbf{w}_i := \mathbf{b}_{i(\frac{p}{2}+1:p)} \quad (8.16)$$

$$\mathbf{m}_a := \mathbf{m}_{(1:\frac{p}{2})} \quad (8.17)$$

$$\mathbf{m}_b := \mathbf{m}_{(\frac{p}{2}+1:p)} \quad (8.18)$$

for  $i \in \mathbb{N}_k$  denote the horizontal and vertical component of the  $i$ -th eigenflow and the mean vector. Let  $J = (L, \mathbf{d})^T(L, \mathbf{d})$  denote the structure tensor containing  $(k+1) \times (k+1)$  entries. Then each entry  $J_{ij}$  of the structure tensor corresponds to a scalar product of the following form (as  $J$  is symmetric only the upper triangle and diagonal is indicated):

$$1 \leq i \leq j \leq k : \\ J_{ij} = \langle \mathbf{v}_i * I_x + \mathbf{w}_i * I_y, \mathbf{v}_j * I_x + \mathbf{w}_j * I_y \rangle$$

$$1 \leq i \leq k, j = k+1 : \\ J_{ij} = \langle \mathbf{v}_i * I_x + \mathbf{w}_i * I_y, I_t + \mathbf{m}_a * I_x + \mathbf{m}_b * I_y \rangle$$

$$i = j = k+1 : \\ J_{ij} = \langle I_t + \mathbf{m}_a * I_x + \mathbf{m}_b * I_y, I_t + \mathbf{m}_a * I_x + \mathbf{m}_b * I_y \rangle.$$

In the first case  $[1 \leq i \leq j \leq k] \quad 3\frac{k}{2}(k+1)$  convolutions of eigenvectors with image derivatives are necessary. In the second case  $[1 \leq i \leq k, j = k+1] \quad 6k$  convolutions are necessary, and in the third case  $[i = j = k+1] \quad 9$  convolutions are necessary. So, all in all, this amounts to  $\frac{3}{2}k^2 + \frac{15}{2}k + 9$  convolutions to compute the structure tensor for all image locations.

This number can be reduced to  $\frac{3}{2}k^2 + \frac{7}{2}k + 1$  convolutions if the training data is symmetrized in such a way that the mean vector computed by PCA equals  $\vec{0}$  (see section 7.2.2). For comparison, the original structure tensor method by Bigün for  $k$  parameters yields  $\frac{1}{2}k^2 + k + 1$  convolutions.

After computing the structure tensor the proposed method as well as the original method by Bigün simply amounts to solving a  $(k+1) \times (k+1)$  system of equations.

Figure 8.6 shows the computation times per pixel on a 2.4 GHz machine for increasing integration area sizes and numbers of eigenflows computed on the Yosemite sequence

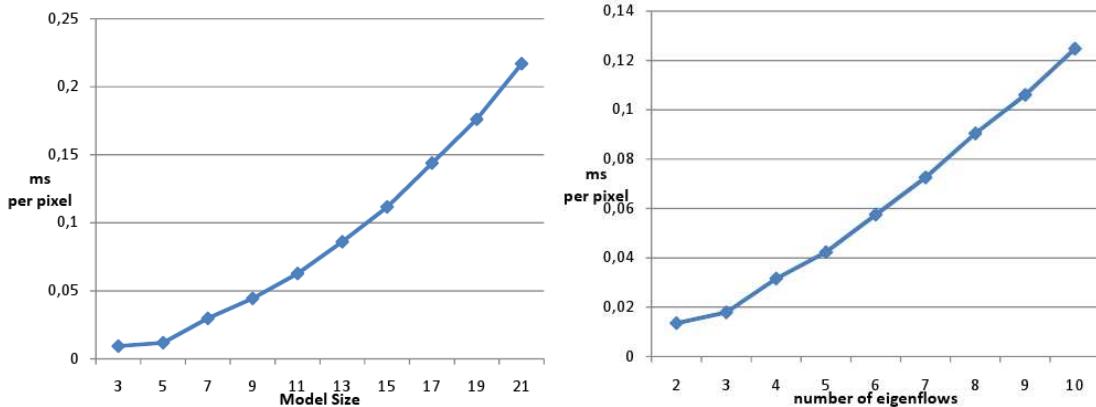


Figure 8.6: Computation times per pixel for increasing integration area sizes and increasing numbers of eigenflows.

containing 15 frames based on the basic least squares method. We can conclude that for small integration areas and numbers of eigenflows the method works near-realtime without further tuning. For larger sizes and numbers of eigenflows the computation time increases approximately quadratically. Furthermore, the convolutions carried out to compute the structure tensor images take most of the computation time, the application of the model to the sequence is very fast. For comparison: the previously proposed method “Real-Time Optic Flow Computation with Variational Methods” by Bruhn et al. [27] takes 0.0094 ms per pixel on a 3.06 GHz machine. To speed up the approach, the method can be parallelized or implemented on GPUs similarly as described by Strzodka and Garbe [95].

### 8.5.3 Adaptability

The algorithm is adaptable to all kinds of scenes where typical, complex motion patterns need to be computed such as in fluid dynamics or driver assistance systems in vehicles. Especially in such cases it is valuable to learn motion statistics from sample data. If no such prior knowledge on the type of motion is available the algorithm achieves results of already high accuracy. The accuracy is increased in case such prior knowledge is in fact available. Figure 8.7 shows spatial principal components computed on particle image velocimetry (PIV) test data, on the Yosemite sequence and on a motion boundary. The examples show that for very different kinds of flow fields we obtain very different principal components. In this way, the algorithm can be adapted to special applications. The proposed method can easily be extended to three dimensional optical flow problems and brightness changes [50].

Figure 8.8 shows results for PIV data from the PIV challenge [79]. As the displace-

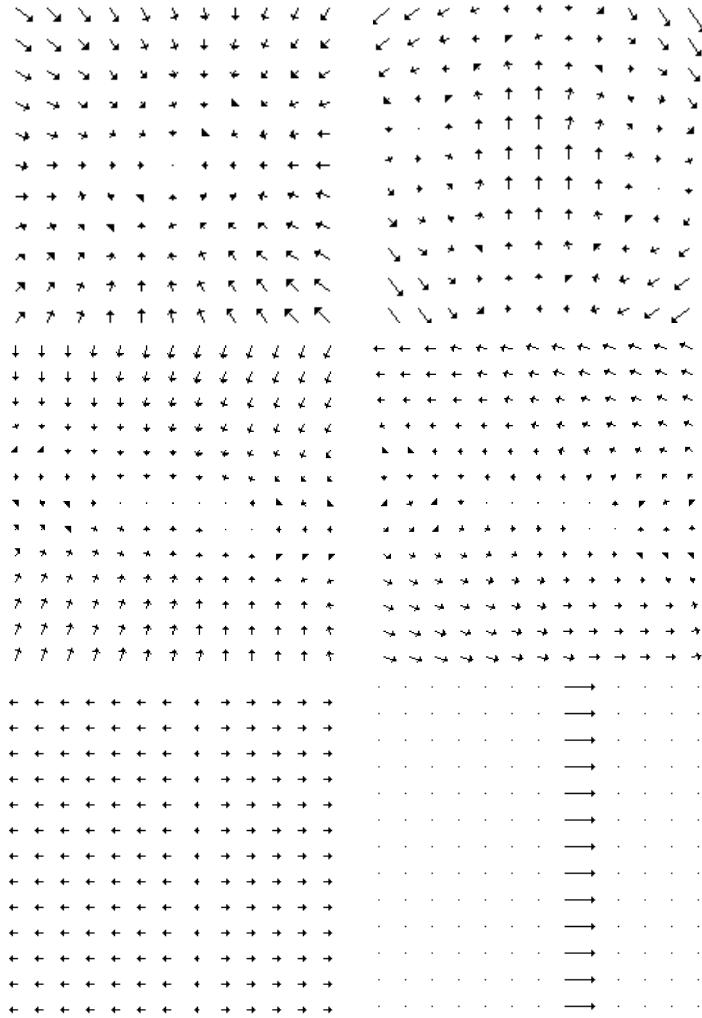


Figure 8.7: Examples of spatial principal components computed on PIV data (top), the Yosemite sequence (center) and on a motion boundary (bottom).

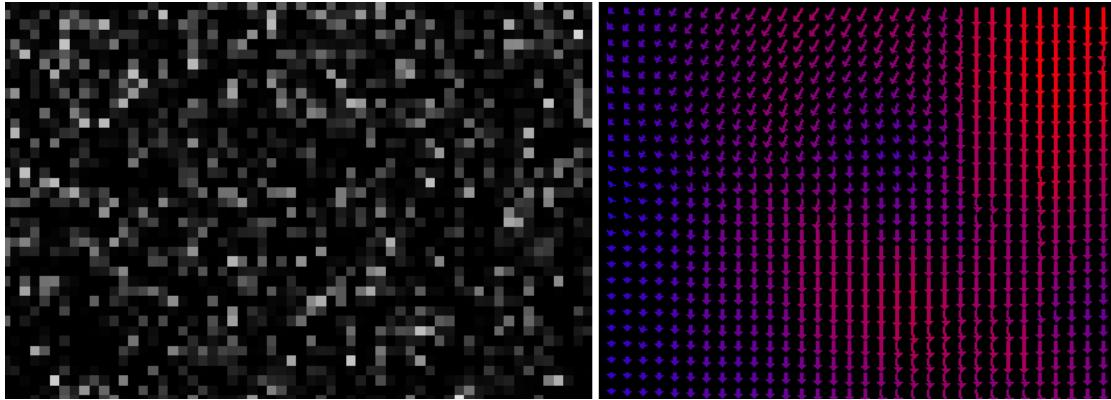


Figure 8.8: Example sequence and computed flow field for PIV data.

ments are extremely large in the upper right corner of the image, the image derivatives for the small particles are incorrect leading to incorrect results due to the linearized brightness constancy constraint. Hence, the resolution of the image sequence is reduced by a factor of 4. To compute image derivatives, the  $7 \times 7 \times 7$  filters proposed by Scharr [89] are employed. Six eigenflows sized  $15 \times 15 \times 1$  are used to estimate the flow, and the sequence is preblurred by a Gaussian filter of spatial standard deviation 0.8. In this way root mean squared (RMS) errors of 0.13 could be obtained, which lie within the range of typical PIV methods applied in the PIV challenge [93]. Figure 8.8 shows an example for such a sequence and computed flow field.

#### 8.5.4 Results for the Global Approach

To demonstrate the global approach it is applied to the Yosemite and the Rubber Whale sequence. For the Yosemite sequence the Gaussian filter with  $\sigma = 0.7$  is used. Furthermore, a  $13 \times 13 \times 1$  model is used to estimate 6 model parameters. The smoothing parameter was set to  $\lambda = 500$ . For the data term function  $\psi_1 \beta_1 = 10$  is chosen, and for the regularizer  $\beta_2 = 2$ . 200 linear iterations and 5 nonlinear steps on each of the 5 pyramid levels were used. I employed warping as described above, the central difference scheme for spatial discretization and forward differences for temporal discretization. The resulting flow field and the corresponding angular error can be found in Figure 8.9. An average angular error of  $1.77 \pm 1.61$  could be obtained.

For the second test sequence, the Rubber Whale sequence, the following parameters were used: Gaussian  $\sigma = 1.0$ , model size  $5 \times 5 \times 1$ , 10 model parameters,  $\lambda = 500$ ,  $\beta_1 = 70$  and  $\beta_2 = 0.1$ . The flow field and angular error are depicted in Figure 8.9. The average angular error for this sequence is  $9.13 \pm 17.43$ .

The problem with the global approach is that the high memory complexity does not

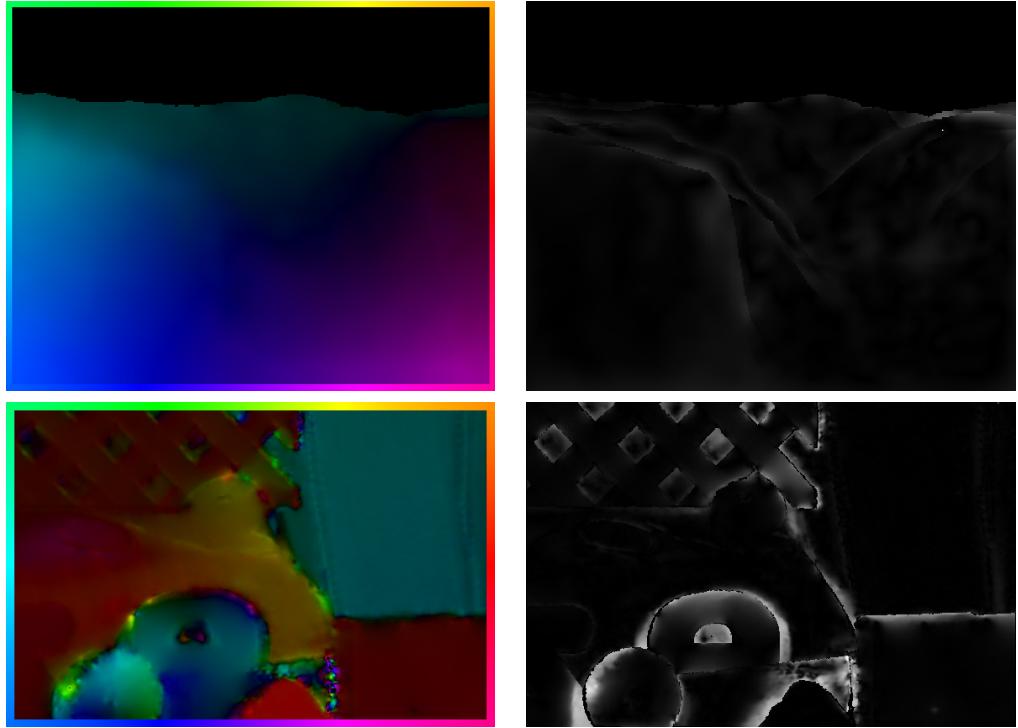


Figure 8.9: Flow and angular error computed on the Yosemite sequence (top) and on the Rubber Whale sequence (bottom) by means of the global approach described in section 8.4.

allow for large model sizes, especially not temporal ones. Hence, the results had to be limited to rather small, solely spatial optical flow models. Better results could probably be obtained with larger model sizes.

## 8.6 Summary and Conclusion

I have presented a novel, local approach to optical flow estimation, which essentially extends the basic structure tensor method by incorporating prior knowledge by means of learning motion models. The proposed method yields results of high quality in case of small or intermediate displacements. For large displacements the linearization of the brightness constancy constraint equation is no longer a valid approximation. Here, a multiscale approach can be employed to yield good results. For the Yosemite sequence, among the local methods the proposed approach obtains errors comparable to Farnebäck [36] with much lower standard deviation. Among global methods it is more accurate than the non-linear 3d CLG method by Bruhn et al. [28] or the statistics based method by

Roth and Black [85]. Furthermore, the proposed method is not only accurate, but also simple to implement, easily parallelizable and adaptable to special motion patterns in case such prior knowledge is available. Hence, especially if implementation or computation time is sparse (as is often the case in industry applications) or in case typical motion patterns exist in the scene, the proposed algorithm is a good choice. I have demonstrated that the method is robust with respect to the selection of sufficiently general sample data for the motion statistics and other parameters as well. Besides, the suggested local approach can be integrated into a global optical flow method, which is based on learned motion models.



# Chapter 9

---

## The Restoration of Optical Flow Fields

### 9.1 Introduction

The previous chapters were dedicated to the analysis of computed optical flow fields. Situation measures can be used to make statements on the estimability of the optical flow vector based on the complexity of the image sequence. In contrast, confidence measures analyze the accuracy of an already computed flow field. Based on such indicators, incorrect vectors can be removed from given flow fields. In this chapter I propose methods for the restoration of sparsified fields in order to obtain dense fields with lower average error. In the results section I will show how flow fields can be automatically refined based on the combination of the nonlinear statistical confidence measure suggested in Chapter 7 and an image based inpainting approach proposed in this chapter.

#### 9.1.1 Motivation

Many methods have been proposed to estimate motion in image sequences. Yet, in difficult situations such as in case of multiple motions, aperture problems or at occlusion boundaries incorrect optical flow estimates often occur. These incorrect flow vectors can be detected and removed from the flow field e.g. by means of confidence measures [26, 66]. But since many applications demand a dense flow field it would be beneficial to reconstruct the missing vectors based on information from the surrounding flow field. A similar task has been addressed in the field of image reconstruction, where it was called “inpainting”.

The reconstruction of optical flow fields can be accomplished by a simple extension of these inpainting functionals for images, e.g. TV-inpainting on two dimensional vector fields. However, these methods sometimes fail in situations where the course of the

motion boundary is unclear, e.g. if round motion boundaries or junctions occur. Since image edges often correspond to motion edges the information drawn from the image sequence can be important for the reconstruction, especially in such cases where the damaged vector field does not contain enough information to uniquely determine the course of motion boundaries.

Hence, in the special case of optical flow, the image sequence provides a source of information in addition to the corrupted vector field, which can be used to guide the reconstruction process in ambiguous cases. So far, optical flow fields have sometimes been used for the reconstruction of images, e.g. in video completion – this time I use the image to reconstruct the optical flow field. The resulting functional is nonlinear and can be minimized by means of the finite element method. I compare the results to diffusion based and TV inpainting methods.

### 9.1.2 Related Work

For the reconstruction of images, inpainting is a widely used technique. The reconstruction of corrupted images was first proposed by Masnou and Morel [72] and named “disocclusion”. The term “inpainting” was brought up by Bertalmio et al. in [15]. It refers to the art of restoring damaged paintings or, in case of digital images, to the reconstruction of blank image domains based on image information outside the domain. The classical inpainting problem can be formulated as follows.

Given an image  $I_0 : D \rightarrow \mathbb{R}$  and an inpainting domain  $G \subset \Omega$ , one asks for a restored image intensity  $I : D \rightarrow \mathbb{R}$ , such that  $I|_{D \setminus G} = I_0$  and  $I|_G$  is a suitable and regular extension of the image intensity  $I_0$  outside  $G$ . The simplest inpainting model is based on the construction of  $I$  on  $G$  with boundary data  $I = I_0$  on  $\partial G$ . This model is equivalent to the minimization of

$$E_L(I) = \frac{1}{2} \int_G \|\nabla I\|^2 \, d\mathbf{x} \quad (9.1)$$

for given boundary data. The resulting intensity function  $I$  is smooth – even analytic – inside  $G$  but does not continue any edge type singularity of  $I_0$  prominent at the boundary  $\partial G$ . To resolve this shortcoming TV-type inpainting models have been proposed [29]. They are based on the functional

$$E_{\text{TV}}(I) = \frac{1}{2} \int_G \|\nabla I\| \, d\mathbf{x}, \quad (9.2)$$

which allows for steep transitions on some edge contour. The resulting image intensity is a BV function and, thus, characterized by jumps along rectifiable edge contours.

In [9] Ballester et al. proposed a variational approach based on the continuation of isophote lines. A variational approach based on level set perimeter and mean curvature was presented by Ambrosio and Masnou in [2]. Other approaches have been proposed for image inpainting, e.g. curvature-driven diffusion inpainting suggested by Chan and Shen [30], or the restoration of motion fields for video reconstruction.

### 9.1.3 Contribution

In this chapter I address the restoration problem for locally corrupted optical flow fields. Often in practical applications, the flow field based on image derivatives may be corrupted locally, while the image data is still available. This available information has not been exploited previously for optical flow restoration. Thus, a novel anisotropic BV-type variational approach is proposed, where the anisotropy takes into account edge information of the underlying image sequence. To identify unreliable flow vectors a confidence measure is used. This measure is taken into account as a weight in the functional. The method will be validated on test data and on real world motion sequences with given ground truth. Part of this work has been submitted [14].

Let  $\varphi(\mathbf{x}) : \mathbb{R}^3 \rightarrow [0, 1]$  denote the confidence function which indicates the regions to be reconstructed. Let, furthermore,  $\theta$  stand for the threshold applied to  $\varphi$  in order to identify the regions to be reconstructed, and let  $H$  stand for the Heaviside function.

## 9.2 Diffusion Based Motion Inpainting

A fast and simple way to inpaint a given motion field  $\mathbf{u}$  is to smoothly reconstruct it by minimizing the gradient within the corrupted region. This can be achieved by minimizing the following energy functional in a variational approach [62]

$$\min \int_{\Omega, t} (\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x}))^2 H(\varphi(\mathbf{x}) - \theta) + \lambda \|\nabla \mathbf{u}(\mathbf{x})\|^2 (1 - H(\varphi(\mathbf{x}) - \theta)) d\mathbf{x} .$$

To minimize the energy the calculus of variations is used (see Chapter 2.1). As the region where  $\varphi(\mathbf{x}) > \theta$  is preserved, we only have to compute the minimum for the remaining set of pixels called  $G$ , that is the minimum of the function

$$\phi(\mathbf{u}) = \int_G L(\mathbf{u}) d\mathbf{x} = \int_G \|\nabla \mathbf{u}(\mathbf{x})\|^2 d\mathbf{x} = \int_G \mathbf{u}_{1x}^2 + \mathbf{u}_{1y}^2 + \mathbf{u}_{2x}^2 + \mathbf{u}_{2y}^2 d\mathbf{x} . \quad (9.3)$$

The Euler Lagrange equations for  $i \in \{1, 2\}$  are obtained according to Chapter 2.1 in the following way

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{u}_i} - \frac{\partial}{\partial x} \frac{\partial L}{\partial \mathbf{u}_{ix}} - \frac{\partial}{\partial y} \frac{\partial L}{\partial \mathbf{u}_{iy}} &= 0 \Leftrightarrow \\ -2\mathbf{u}_{ixx} - 2\mathbf{u}_{iyy} &= 0 \Leftrightarrow \\ \mathbf{u}_{ixx} + \mathbf{u}_{iyy} &= 0. \end{aligned}$$

Hence, the minimization of the energy in (9.3) in a variational approach leads to the set of linear partial differential equations, the Euler-Lagrange equations,

$$\begin{cases} \Delta \mathbf{u}_i(\mathbf{x}) = \mathbf{u}_{ixx} + \mathbf{u}_{iyy} = 0, & \text{if } \varphi(\mathbf{x}) < \theta, \\ \mathbf{u}(\mathbf{x}) = \mathbf{u}_0(\mathbf{x}), & \text{otherwise} . \end{cases} \quad (9.4)$$

These equations are discretized using a finite differences scheme. The second derivatives are discretized using a four point Laplace stencil. In this way, we obtain a large linear system of equations, which can be solved using standard methods such as the conjugate gradient method or the Gauss-Seidel method with successive overrelaxation (SOR). Yet, as only the gradient of the motion field is minimized within the corrupted regions, diffusion based motion inpainting is not suitable to continue motion edges into the region to be reconstructed.

### 9.3 TV Motion Inpainting

As edges are not taken into account by the diffusion based approach, the total variation based denoising functional by Rudin, Osher and Fatemi [88] can be adopted to the inpainting of motion fields. The total variation of a motion field is denoted by

$$\begin{aligned} TV(\mathbf{u}) &= \\ &\int_G \|\nabla \mathbf{u}(\mathbf{x})\| d\mathbf{x} = \\ &\int_G \sqrt{\mathbf{u}_{1x}^2(\mathbf{x}) + \mathbf{u}_{2x}^2(\mathbf{x}) + \mathbf{u}_{1y}^2(\mathbf{x}) + \mathbf{u}_{2y}^2(\mathbf{x})} d\mathbf{x} . \end{aligned}$$

The space of functions with bounded variations (BV space) is defined as

$$BV(\Omega) = \{f | f \in L^1(\Omega) \text{ and } TV(f) < \infty\} . \quad (9.5)$$

The BV space is a Banach space together with the BV norm

$$\|f\|_{BV} = \|f\|_{L^1} + TV(f) . \quad (9.6)$$

It allows jumps while having sufficient control over arbitrary oscillations. Hence, this space has often been chosen for the representation of images containing edges. As motion fields contain edges at every occlusion boundary, they can be represented as two-dimensional functions in BV space. I propose to minimize the following energy functional

$$\min \int_G (\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x}))^2 H(\varphi(\mathbf{x}) - \theta) + \lambda \underbrace{\|\nabla \mathbf{u}(\mathbf{x})\|(1 - H(\varphi(\mathbf{x}) - \theta))}_{TV(\mathbf{u}(\mathbf{x}))|_{\{\mathbf{x} | \varphi(\mathbf{x}) < \theta\}}} d\mathbf{x} .$$

Let

$$\phi(\mathbf{u}) = \int_G L(\mathbf{u}) d\mathbf{x} = \int_G \|\nabla \mathbf{u}(\mathbf{x})\| d\mathbf{x} = \int_G \sqrt{\mathbf{u}_{1x}^2 + \mathbf{u}_{1y}^2 + \mathbf{u}_{2x}^2 + \mathbf{u}_{2y}^2} d\mathbf{x} . \quad (9.7)$$

For simplicity in writing let

$$a := \mathbf{u}_{1x}^2 + \mathbf{u}_{1y}^2 + \mathbf{u}_{2x}^2 + \mathbf{u}_{2y}^2 . \quad (9.8)$$

The corresponding Euler-Lagrange equations for  $i \in \{1, 2\}$  are obtained in the following way

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{u}_i} - \frac{\partial}{\partial x} \frac{\partial L}{\partial \mathbf{u}_{ix}} - \frac{\partial}{\partial y} \frac{\partial L}{\partial \mathbf{u}_{iy}} &= 0 \Leftrightarrow \\ \frac{-\mathbf{u}_{ixx}\sqrt{a} + \frac{\mathbf{u}_{ix}^2 \mathbf{u}_{ixx}}{\sqrt{a}}}{a} + \frac{-\mathbf{u}_{iyy}\sqrt{a} + \frac{\mathbf{u}_{iy}^2 \mathbf{u}_{iyy}}{\sqrt{a}}}{a} &= 0 \Leftrightarrow \\ \frac{-(\mathbf{u}_{ixx} + \mathbf{u}_{iyy})a + \mathbf{u}_{ix}^2 \mathbf{u}_{ixx} + \mathbf{u}_{iy}^2 \mathbf{u}_{iyy}}{\sqrt{a}^3} &= 0. \end{aligned} \quad (9.9)$$

Or equivalently, the minimization of the energy in (9.7) leads to the following Euler-Lagrange equations, a set of non-linear partial differential equations

$$\begin{cases} \operatorname{div} \left( \frac{\nabla \mathbf{u}_i(\mathbf{x})}{\|\nabla \mathbf{u}(\mathbf{x})\|} \right) = 0, & \text{if } \varphi(\mathbf{x}) < \theta \\ \mathbf{u}(\mathbf{x}) = \mathbf{u}_0(\mathbf{x}), & \text{otherwise .} \end{cases} \quad (9.10)$$

To solve these equations they are discretized by common finite differences schemes. The Euler-Lagrange equation can be understood as the gradient in the functional space. Hence, we can write for an artificially introduced time step  $t$

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla L \quad (9.11)$$

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{dt} = -\nabla L \quad (9.12)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n - dt \nabla L. \quad (9.13)$$

Thus, to solve the system of nonlinear partial differential equations we can use the gradient descent method.

## 9.4 Image Guided Motion Inpainting

The TV motion inpainting approach is able to reconstruct flow field edges. However, the precise course of these edges is often unclear for larger destroyed regions. Consider for example the edge of a circle. Such problems could be handled by the integration of information from the image sequence such as the gradient. I propose to minimize the following variational approach

$$\phi(\mathbf{u}) = \min \int_G (\mathbf{u}(\mathbf{x}) - \mathbf{u}_0(\mathbf{x}))^2 H(\varphi(\mathbf{x}) - \theta) + \lambda \beta(\nabla I(\mathbf{x}), D\mathbf{u}(\mathbf{x}))(1 - H(\varphi(\mathbf{x}) - \theta)) d\mathbf{x}$$

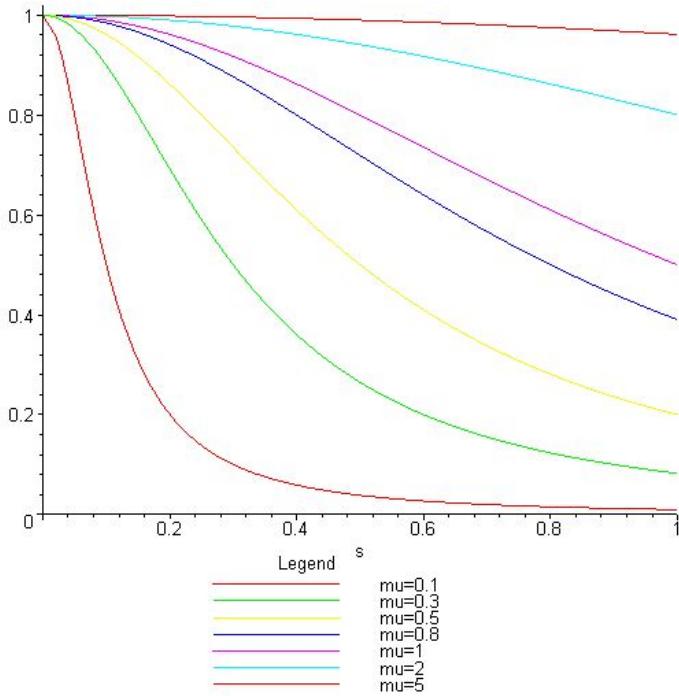


Figure 9.1: Examples for the mapping  $g$ , which controls the influence of the image gradient for different parameter values for  $\mu$ .

where  $D\mathbf{u}$  stands for the Jacobian matrix of  $\mathbf{u}$  and

$$\begin{aligned}
 g(s) &= \frac{1}{1 + \frac{s^2}{\mu^2}}, \\
 \beta(\nabla I(\mathbf{x}), D\mathbf{u}(\mathbf{x})) &= g(\nabla I(\mathbf{x}))|D\mathbf{u}(\mathbf{x})| + (1 - g(\nabla I(\mathbf{x})))\gamma(\nabla I(\mathbf{x}), D\mathbf{u}(\mathbf{x})), \\
 \gamma(\nabla I(\mathbf{x}), D\mathbf{u}(\mathbf{x})) &= \sqrt{\sum_i \nu^2 (\mathbf{n} \cdot \nabla \mathbf{u}_i(\mathbf{x}))^2 + (\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i(\mathbf{x}))^2}, \\
 \mathbf{n} &= \left( \frac{\nabla I(\mathbf{x})}{|\nabla I(\mathbf{x})|} \right).
 \end{aligned}$$

Figure 9.1 shows examples for the mapping  $g$ .

The idea of the reconstruction term behind this formulation is the following: In case the image gradient is low and, thus,  $g$  yields a value close to 1, we want to reconstruct the missing flow vectors isotropically by minimizing the Frobenius norm  $|D\mathbf{u}|$  of the flow field gradient

$$|D\mathbf{u}| = \sqrt{|\nabla \mathbf{u}_1|^2 + |\nabla \mathbf{u}_2|^2}. \quad (9.14)$$

In case the image gradient is high and, thus,  $g$  yields a value close to 0, we want to control the flow field's orientation along the image gradient. Hence, the gradient  $\nabla \mathbf{u}_i, i \in \{1, 2\}$ , of the flow field components is separated into two parts by means of the scalar product with the normalized image gradient  $\mathbf{n}$  and with the orthogonal to this direction,  $\mathbf{n}^\perp$ , respectively. The smaller the parameter  $\nu$  is chosen the less expensive is it to assign the largest part of the flow gradient component to the first term, that means that the flow gradient is oriented along the image gradient. In this way, the anisotropy of the restoration process is controlled by the parameter  $\nu$ .

The value  $\mu$  controls the strength of the image gradient necessary to influence the motion reconstruction process. It can approximately be understood as the weak “threshold” between high and low image gradients, so that for higher values the image gradient is taken into account.

Hence, locally minimizing the prior  $\beta$  will favor sharp motion edges aligned with edges in the underlying image. Apart from edges a usual TV prior is applied to the motion field. In particular for larger destroyed regions this leads to an effective image based guidance in the reconstruction of motion edges.

For  $\nu$  values close to 1 there is no preference for any orientation of a motion edge and we obtain the classical *TV* type inpainting model on motion fields. This will be proven in the following.

Due to orthonormality we know

$$\mathbf{n}_1^2 + \mathbf{n}_2^2 = \mathbf{n}_1^{\perp 2} + \mathbf{n}_2^{\perp 2} = 1, \quad (9.15)$$

$$(\mathbf{n}_1^\perp = -\mathbf{n}_2 \wedge \mathbf{n}_2^\perp = \mathbf{n}_1) \vee (\mathbf{n}_1^\perp = \mathbf{n}_2 \wedge \mathbf{n}_2^\perp = -\mathbf{n}_1). \quad (9.16)$$

Therefore, we obtain

$$\begin{aligned} \gamma(\nabla I, D\mathbf{u}) &= \\ &\sqrt{\sum_i (\mathbf{n} \cdot \nabla \mathbf{u}_i)^2 + (\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i)^2} = \\ &\sqrt{\sum_i (\mathbf{n}_1^2 + \mathbf{n}_1^{\perp 2}) \nabla \mathbf{u}_{ix}^2 + (\mathbf{n}_2^2 + \mathbf{n}_2^{\perp 2}) \nabla \mathbf{u}_{iy}^2 + 2(\mathbf{n}_1 \mathbf{n}_2 + \mathbf{n}_1^\perp \mathbf{n}_2^\perp) \nabla \mathbf{u}_{ix} \nabla \mathbf{u}_{iy}} = \\ &\sqrt{\sum_i \nabla \mathbf{u}_{ix}^2 + \nabla \mathbf{u}_{iy}^2} = |D\mathbf{u}|, \end{aligned} \quad (9.17)$$

and, thus,  $\beta(\nabla I, D\mathbf{u}) = |D\mathbf{u}|$ .

The functional derivative of the energy for  $i \in \{1, 2\}$  is computed as follows

$$\begin{aligned} & \langle \partial_{\mathbf{u}_i} \gamma(\nabla I, D\mathbf{u}), \epsilon \rangle = \\ & \frac{1}{2\gamma(\nabla I, D\mathbf{u})} \left( \nu^2 (\mathbf{n} \cdot \nabla \mathbf{u}_i) (\mathbf{n} \cdot \nabla \epsilon) + 2(\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i) (\mathbf{n}^\perp \cdot \nabla \epsilon) \right) = \\ & \frac{1}{2\gamma(\nabla I, D\mathbf{u})} \left( \nu^2 (\mathbf{n} \cdot \nabla \mathbf{u}_i) \mathbf{n} + (\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i) \mathbf{n}^\perp \right) \nabla \epsilon, \end{aligned}$$

$$\begin{aligned} & \langle \partial_{\mathbf{u}_i} \beta(\nabla I, D\mathbf{u}), \epsilon \rangle = \\ & g(|\nabla I|) \frac{\nabla \mathbf{u}_i}{|D\mathbf{u}|} \cdot \nabla \epsilon + (1 - g(|\nabla I|)) \langle \partial_{\mathbf{u}_i} \gamma(\nabla I, D\mathbf{u}), \epsilon \rangle = \\ & \left( g(|\nabla I|) \frac{\nabla \mathbf{u}_i}{|D\mathbf{u}|} + \frac{1 - g(|\nabla I|)}{\gamma(\nabla I, D\mathbf{u})} \left( \nu^2 (\mathbf{n} \cdot \nabla \mathbf{u}_i) \mathbf{n} + (\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i) \mathbf{n}^\perp \right) \right) \cdot \nabla \epsilon, \end{aligned}$$

$$\begin{aligned} & \langle \partial_{\mathbf{u}_i} \phi(\mathbf{u}), \epsilon \rangle = \\ & \int_{\Omega, t} H(\varphi - \theta)(\mathbf{u}_i - \mathbf{u}_{i_0}) \epsilon + \\ & \nu \left( g(|\nabla I|) \frac{\nabla \mathbf{u}_i}{|D\mathbf{u}|} + \frac{1 - g(|\nabla I|)}{\gamma(\nabla I, D\mathbf{u})} \left( \nu^2 (\mathbf{n} \cdot \nabla \mathbf{u}_i) \mathbf{n} + (\mathbf{n}^\perp \cdot \nabla \mathbf{u}_i) \mathbf{n}^\perp \right) \right) \cdot \nabla \epsilon \, d\mathbf{x}. \end{aligned} \tag{9.18}$$

This weak formulation of the Euler-Lagrange equation can be used for the minimization of the energy by means of the finite element method and gradient descent with Armijo step size control [5] to speed up the process.

## 9.5 Experiments and Results

### 9.5.1 Reconstruction of Artificial Motion Fields

To illustrate the image guided motion inpainting method it is applied to the reconstruction of a corrupted rectangular and circular motion field. To this end, Figure 9.2 shows the color coded ground truth flow field on the left hand side (a), the red shape indicating the region to be reconstructed in the second image (b), the initialization of the image guided motion inpainting algorithm in the third image (c) and the result of the algorithm on the right hand side (d). The final results show that the reconstruction process was successful in retrieving the motion boundary along the edge of the circle. The following set of parameters:  $\lambda = 1$ ,  $\mu = 50$  and  $\nu = 0.1$  was used. For the circle 6200 iteration steps were necessary, for the rectangle 12200.

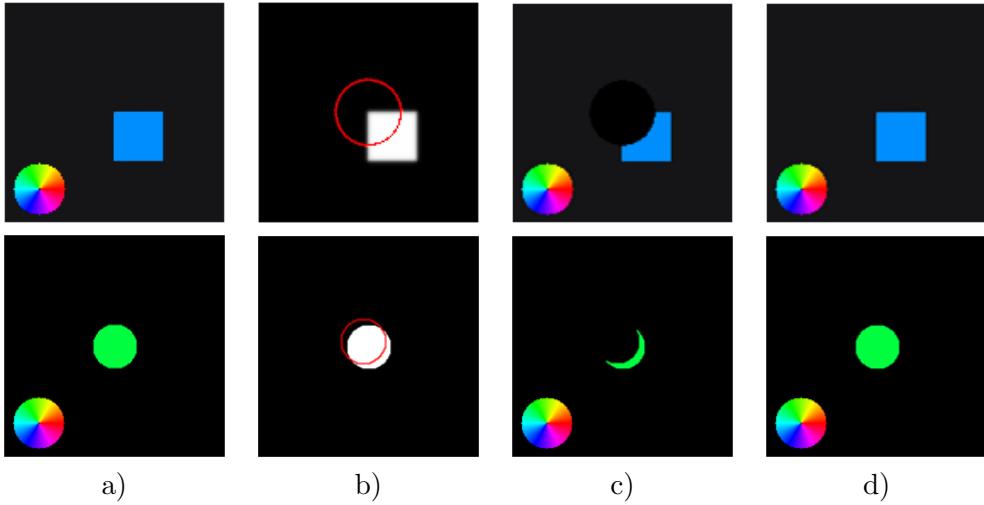


Figure 9.2: a) Ground truth flow field, b) Underlying image and corruption indicated by the red shape, c) Corrupted flow field which is the initialization of the image guided motion inpainting algorithm, d) Restored flow field.

### 9.5.2 Reconstruction of Real World Motion Fields

After reconstructing artificial motion fields I now turn to real world examples and reconstruct the motion field of a sequence taken from the Middlebury dataset [8]. Special attention will be turned to the effect of the parameters  $\nu$  and  $\mu$  on the reconstruction result. Figure 9.3 shows the Rubber Whale sequence with its corrupted regions marked by red shapes (a), the ground truth flow field (b), the result of the image guided reconstruction algorithm (c) and the angular error (d). The following set of parameters was used:  $\lambda = 1$ ,  $\mu = 1$  and  $\nu = 0.1$ .

To investigate the effect of the parameter  $\nu$  let us take a closer look at two different regions in the scene: the upper left corner of the turning wheel on the left hand side and the flap of the box on the right hand side. At the upper side of the wheel the image contrast is low and, thus, makes reconstruction along image edges difficult. Hence, the sensitivity of the method concerning the image gradient should be high and the method's inclination to follow image edges should be large as well, which would result in small values for  $\mu$  and  $\nu$ .

At the flap of the box we have the opposite problem. The image contrast is large, but the motion boundary does, in fact, not follow the strong but the weaker edge above. Hence, the inclination of the method to follow image edges should be reduced, which would result in a higher value for  $\nu$ .

The effect of different parameter constellations for both regions is shown in Figure 9.4. The results demonstrate that for low  $\nu$  values the wheel can be reconstructed quite well,

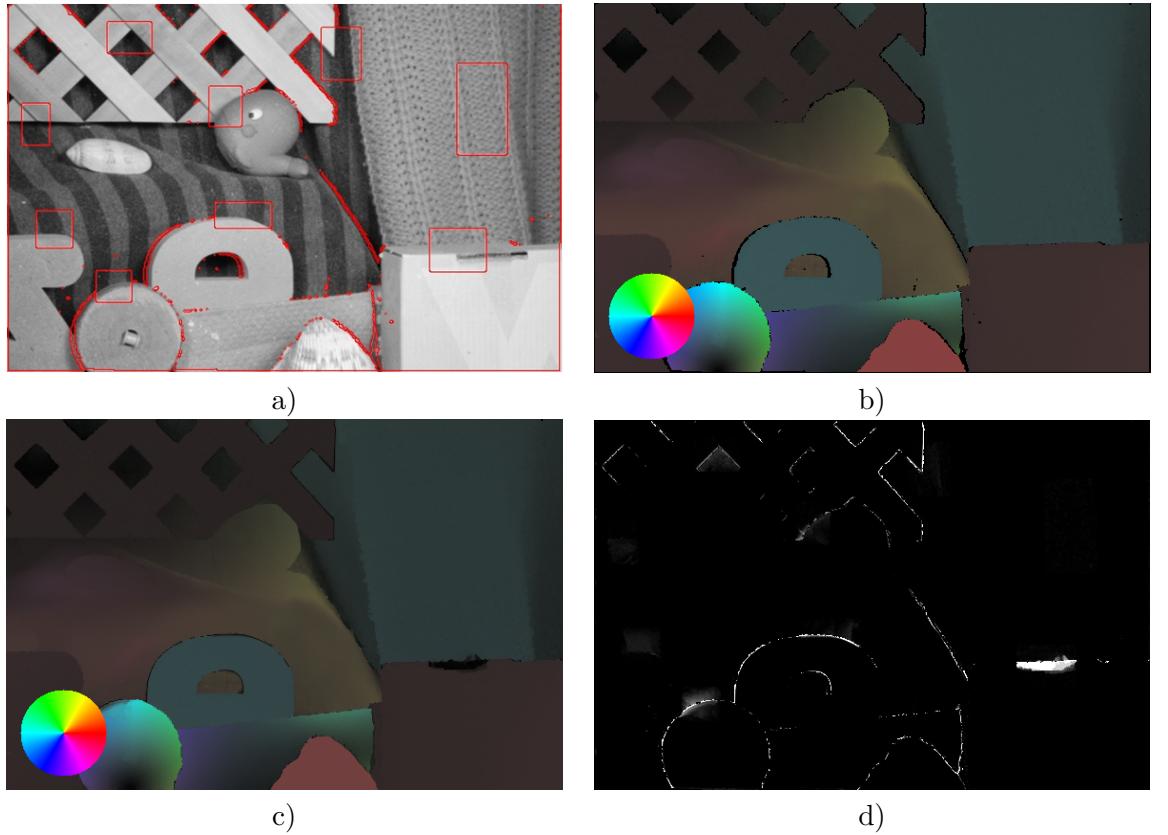


Figure 9.3: a) Original Rubber Whale frame, b) Ground truth flow field, c) Restored flow field, d) Angular error.

but the motion field also follows the sharp edge of the box flap and yields errors in that part of the sequence. In contrast, for high  $\nu$  values the box flap can be reconstructed well, but the wheel is reconstructed by a straight edge, which does not follow the contour of the wheel.

### 9.5.3 Comparison to Diffusion and TV Inpainting

To finally compare the image guided motion inpainting algorithm to the diffusion and TV inpainting method, I apply them to the corrupted Marble sequence. Figure 9.5 shows the original corrupted sequence and the results of the diffusion based, the TV-based and the image based motion inpainting methods.

The results demonstrate that the diffusion based motion inpainting is not able to reconstruct flow edges. In contrast, by means of TV motion inpainting flow edges can be reconstructed. However, the lower right corner of the central marble block cannot

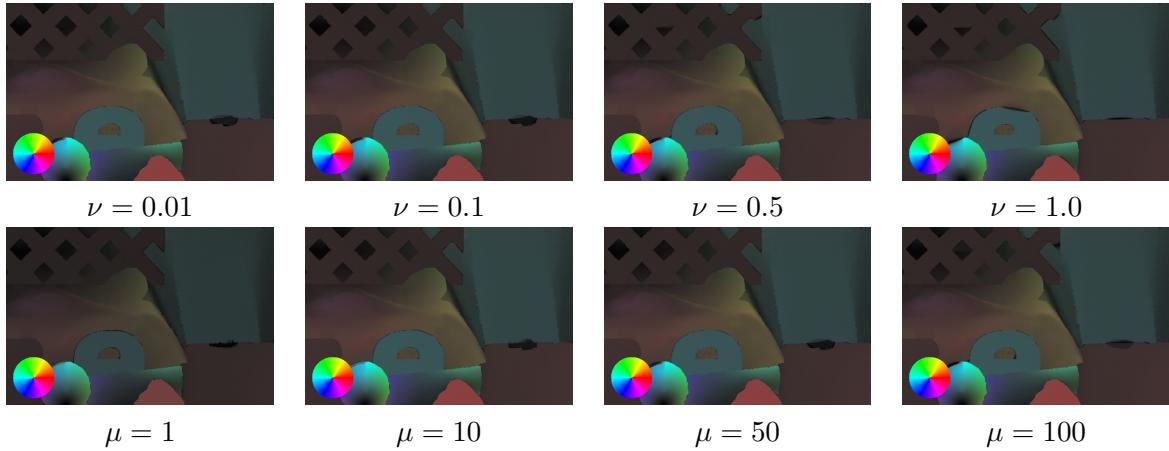


Figure 9.4: Upper row: results for different values of  $\nu$  for  $\mu = 50$ , lower row: results for different values of  $\mu$  for  $\nu = 0.1$

be reconstructed without information drawn from the original image, because the exact course of the edges near the junction is unclear. Image based motion inpainting uses the image gradient information to correctly reconstruct the motion boundary of the central marble block as well. Here the following set of parameters was used:  $\lambda = 1$ ,  $\mu = 50$  and  $\nu = 0.1$ .

#### 9.5.4 Reconstruction Based on Confidence Measures

Finally, the most effective confidence measure, the nonlinear statistical confidence measure, and the image guided motion inpainting approach are combined to improve given optical flow fields. The results of the nonlinear confidence measure (pValNonlin) are used as confidence function  $\varphi$ , which indicates the reliability of the current flow vector. Figure 9.6 shows the original flow fields and their reconstruction. For the Rubber Whale sequence and the structure tensor flow field a threshold of  $\theta = 0.03$  was applied to the pValNonlin confidence measure shown in a). By means of image based inpainting I was able to reduce the angular error from  $11.18 \pm 23.32$  to  $8.31 \pm 16.18$ .

For the Marble sequence the flow field was computed by Farnebäck's method. In this case, a threshold of  $\theta = 0.19$  was applied to the pValNonlin confidence measure. The image guided inpainting approach then reduced the angular error from  $2.13 \pm 3.21$  to  $1.93 \pm 2.54$ . Thus, I have shown how optical flow fields can be refined automatically to obtain lower average errors.

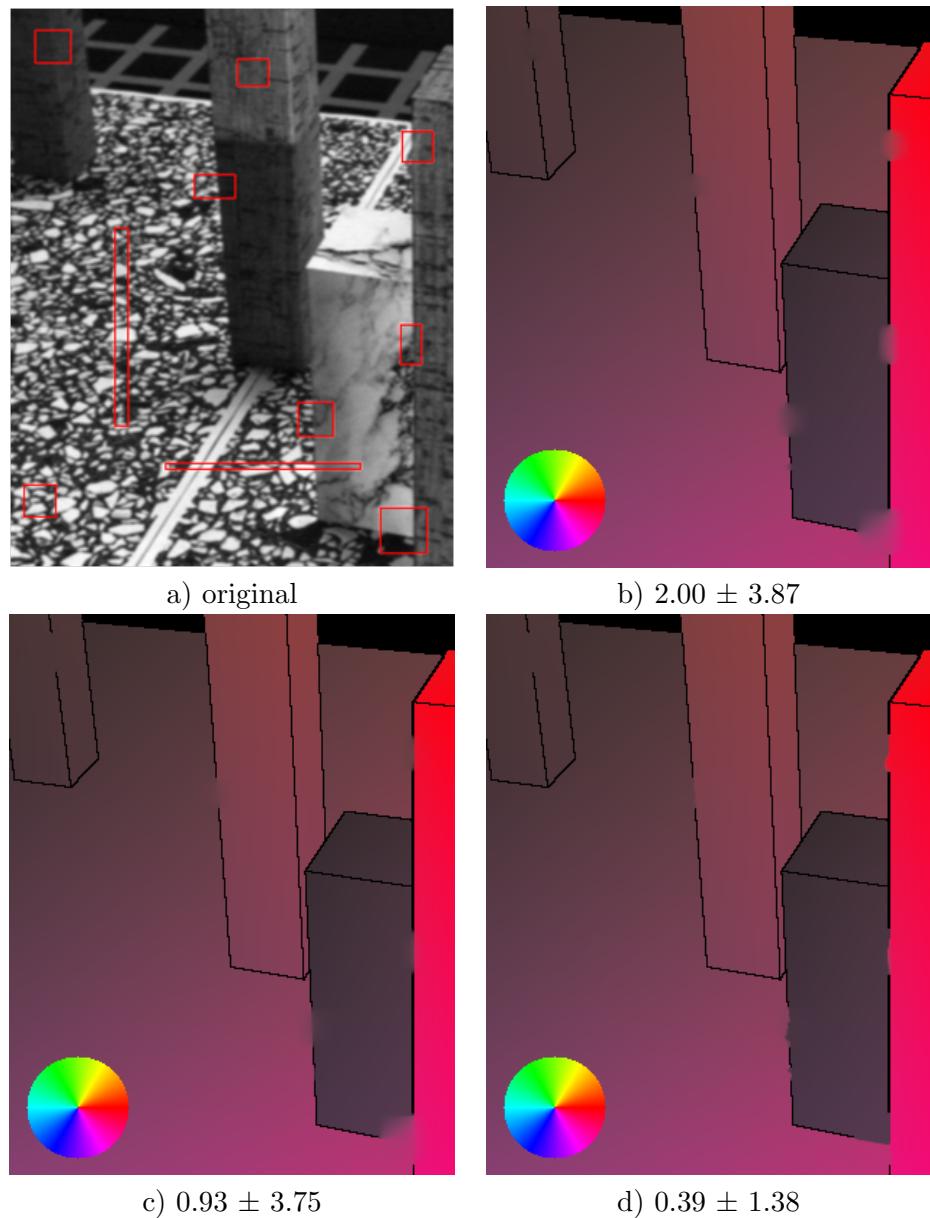


Figure 9.5: Comparison of the proposed image guided inpainting algorithm to diffusion and TV inpainting; the numbers indicate the average angular error within the corrupted regions after reconstruction; a) Original Marble sequence with corruptions indicated by red rectangles, b) Reconstruction result of diffusion based motion inpainting, c) Reconstruction result of TV based motion inpainting, d) Reconstruction result of image guided motion inpainting.

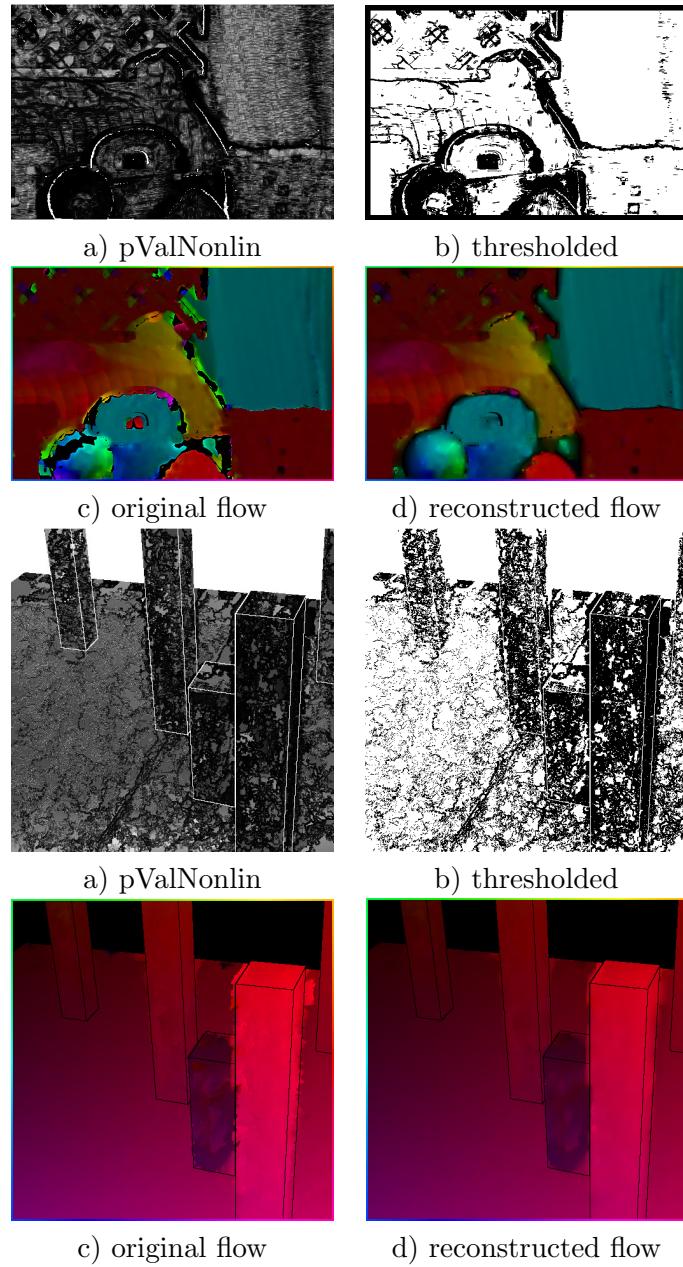


Figure 9.6: Reconstruction of the Rubber Whale and Marble flow computed by the structure tensor method and Farnebäck's method respectively. a) Result of the nonlinear confidence measure (pValNonlin), b) Thresholded confidence, c) Original flow field (cropped to maximum flow length 4), d) Reconstruction of the original flow field after image guided motion inpainting.

## **9.6 Summary and Conclusion**

Given an image sequence and an extracted underlying motion field together with a local measure of confidence for the motion estimation I have proposed a variational approach for the restoration of the motion field. This restoration is vital for a number of applications requiring dense motion fields. Based on a confidence measure regions of corrupted motion can be detected. The underlying image data is still available and reliable. I make use of this information to improve the restoration of the motion field. The approach is based on an anisotropic TV-type functional, where the anisotropy takes into account edge information extracted from the underlying image data. The approach has been applied to test data and to two different real world optical flow problems. The results are compared to diffusion based vector field inpainting and TV-type inpainting. I demonstrate that inpainting guided by the underlying intensity data outperforms purely flow driven approaches. I consider this as a feasibility study for the coupling of motion field and image sequence data in variational inpainting approaches.

# Chapter 10

---

## Conclusions and Perspectives

### 10.1 Summary and Conclusion

In this thesis I have addressed the analysis and restoration of arbitrarily computed optical flow fields.

Statements on the accuracy of flow vectors always require a given error definition. Therefore, chapter 4 was dedicated to the analysis of known error measures for optical flow fields. Due to shortcomings of previous methods I have derived a joint distribution of optical flow estimates, ground truth flow vectors and gray value neighborhoods, and obtained interesting statements from this distribution by means of marginalization and conditioning. For example, I observed that all estimators except the Nir method have difficulties with large displacements, and I obtained principal components showing typical gray value structures in case of very high or very low endpoint errors. I also proposed a statistically motivated scalar indicator suited for the ranking of different estimators. I applied the evaluation method to five known optical flow estimators. Based on this statistical evaluation and systematic analysis, I hope to further improve optical flow estimators and their analysis methods.

Quality estimates for optical flow vectors can be obtained by means of confidence measures. In Chapter 5 I composed the most important previously used confidence measures and classified them according to the intrinsic dimension of the image sequence they examine. In the end I was able to assign the most effective measure to each intrinsic dimension with respect to noise robustness. However, the measures only yielded acceptable results on artificial sequences, not on real-world sequences. Furthermore, they derive statements on the feasibility of accurate flow computation from the intrinsic dimension of the image sequence only. Thus, I found that none of these measures qualifies as confidence measure, since they do not even consider the computed flow field. We, therefore, suggested to distinguish between situation and confidence measures.

To account for this shortcoming, in Chapter 6 I proposed to instead analyze the intrinsic

dimension of the energy surface after optimization in the flow computation process. The resulting measure also detects outliers and can either be employed as confidence measure in case the energy surface is obtained from a computed flow field or as a situation measure in case the energy surface is based on a zero flow field. In the latter case, the measure examines the feasibility of an accurate optical flow computation and yielded noise resistant detection results far above previously used situation measures for artificial and real-world sequences. The additional application of a simple motion inpainting algorithm led to impressive results, as angular error reductions of up to 38% were feasible by confidence estimation and subsequent motion restoration. Furthermore, I concluded for the employed test sequences that the application of the suggested postprocessing method to sparsified flow fields calculated with local or global methods yielded better results than could be achieved by exploiting the filling-in effect of the original global methods. Hence, in contrast to the accepted opinion, global methods are not always preferable to local methods if a dense flow field is required, because motion inpainting only based on a set of reliable flow vectors led to superior results.

The confidence measures presented in Chapter 7 contribute to the automatic quality evaluation of optical flow fields. They are statistically motivated, since they are based on probability distributions, which are learned from sample data. One of the three measures relies on principal component analysis models, the other two are formulated as best linear unbiased estimators based on linear or nonlinear data vectors. In this way, I was able to define confidence measures, which take into account the flow field, can be applied to arbitrarily computed motion fields, and justifiably bear the notion “confidence measure” in a statistical sense. Slight changes in the algorithm even allow for applications to sparse flow fields, which often occur e.g. in traffic scenes. Results for locally and globally computed flow fields on ground truth test sequences based on different error measures showed the superiority of the proposed method compared to previously employed confidence measures. Error reductions of up to 50% for the whole flow field were feasible by removing only 10% of the flow vectors indicated by the proposed nonlinear confidence measure.

As every confidence measure contains some knowledge on correct flow fields, it usually contains the basic idea for a new optical flow estimator as well. Based on the linear subspace projection measure proposed in Chapter 7, I suggested a novel, local approach to optical flow estimation in Chapter 8, which essentially extends the basic structure tensor method by incorporating prior knowledge by means of learned motion models. Advantages of this approach are despite its high accuracy that it is simple to implement, easily parallelizable and adaptable to specific types of motion in special applications. On the Yosemite sequence, it yields results comparable to Farnebäck’s method, but with much less implementation effort, and obtains angular errors below the 3D combined local global (CLG) method by Bruhn et al. [28] and the learning based approach by Roth and Black [85].

The last Chapter 9 was dedicated to the restoration of optical flow fields. Methods from

image inpainting have been transferred to the reconstruction of motion fields. A simple approach is the diffusion based inpainting method, which is not able to continue motion boundaries. To solve this problem the total variation based inpainting approach was employed. However, there are situations, where the course of a motion boundary cannot be unambiguously deduced from the preserved neighboring motion field. To this end, the gradient of the image sequence was used to guide the reconstruction process. The image based motion inpainting approach allows for the definition of the minimum image edge strength and the strength of the anisotropy. The results showed that by means of image guided inpainting motion boundaries, which could not be retrieved by TV-inpainting, were correctly restored. Finally, the combination of the nonlinear statistical confidence measure and the image guided inpainting method yielded an automatic optical flow field refinement routine, which significantly reduced the angular error of the test flow fields. Our results indicate that by identifying a set of reliable flow vectors followed by subsequent mathematical motion inpainting approaches flow fields of lower angular errors can be obtained than by combining all model constraints in one functional.

## 10.2 Future Research

For future research, it would be interesting to further improve the proposed confidence measures. To this end, other motion models or different learning methods could be employed, e.g. advanced machine learning techniques. In this way, the motion models could also account for rare flow field constellations, which are underrepresented in the training data, and for improved representations of motion boundaries. Furthermore, it could be rewarding to directly learn a confidence estimator based on sample data without having to define or learn motion models at all.

Concerning the reconstruction of optical flow fields, a topic for future research would be the refinement of the image guided inpainting approach. Robustness and reliability of the approach might be improved based on a fully joint approach, where the motion field and the image sequence are jointly restored. A restoration in space time would be promising as well. Furthermore, as shown in the results section of Chapter 9 one set of parameters is sometimes not suitable for the whole optical flow field. Hence, it would be beneficial to adapt these parameters to the requirements of the current situation.

Furthermore, intrinsic dimension information of the energy surface could be used in optical flow estimators to define new regularization methods. E.g. in case of flat energy surfaces, which correspond to intrinsic dimension zero, a homogeneous regularization model would be suitable as many different displacement vectors lead to similar energies, which indicates low certainty. In contrast, for energy surfaces of intrinsic dimension one, smoothing of the flow field should only be applied in the direction along the aperture problem. In case of intrinsic dimension two, which indicates energy surfaces having a unique minimum, the computed flow vectors should have larger influence on their neigh-

bors due to high certainty. Hence, regularization models or strengths could be adapted to the intrinsic dimension of the energy surface.

Another rewarding issue would be a more thorough investigation of the model based global optical flow method proposed in Chapter 8. Here, due to memory complexity merely small patch sizes could be tested yielding only intermediate results. As already seen in the local approach, larger patch sizes might significantly reduce the flow field error.

In Chapter 5 we have seen that most of the situation measures yielding good or intermediate results for artificial test sequences fail for real-world applications. Hence, more sophisticated measures that are especially noise resistant would be another topic for future research.

# Bibliography

- [1] L. Alvarez, R. Deriche, and J. Papadopoulo, T. Sanchez. Symmetrical dense optical flow estimation with occlusion detection. In *European Conference on Computer Vision (ECCV)*, pages 721–735, 2002.
- [2] L. Ambrosio and S. Masnou. A direct variational approach to a problem arising in image reconstruction. *Interfaces and Free Boundaries*, 5:63–81, 2003.
- [3] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–319, 1989.
- [4] B. Andres, C. Konermann, D. Konermann, U. Köthe, F. Hamprecht, and C. Garbe. On errors-in-variables regression with arbitrary covariance and its application to optical flow estimation. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–6, 2008.
- [5] L. Armijo. Minimization of functions having Lipschitz continuous first derivatives. *Pacific Journal of Mathematics*, 16(1):1–3, 1966.
- [6] A. Bab-Hadiashar and D. Suter. Robust optic flow computation. *International Journal of Computer Vision (IJCV)*, 29(1):59–77, 1998.
- [7] R.G. Bainbridge-Smith, A. Lane. Measuring confidence in optical flow estimation. *IEEE Electronics Letters*, 32(10):882–884, 1996.
- [8] S. Baker, S. Roth, D. Scharstein, M. Black, J. Lewis, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.
- [9] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10(8):1200–1211, 2001.

## Bibliography

---

- [10] J. L. Barron, D. J. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.
- [11] E. Barth. Bewegung als Intrinsische Geometrie von Bildfolgen. In *Proceedings of the German Association for Pattern Recognition (DAGM)*, 1999.
- [12] E. Barth. The minors of the structure tensor. In *Proceedings of the German Association for Pattern Recognition (DAGM)*, 2000.
- [13] Erhard Barth, Ingo Stuke, Til Aach, and Cicero Mota. Spatio-temporal motion estimation for transparency and occlusions. In *Proceedings of the International Conference on Image Processing (ICIP)*, volume 3, pages 69–72, 2003.
- [14] B. Berkels, C. Konermann, C. Garbe, and M. Rumpf. Reconstructing optical flow fields by motion inpainting. In *Proceedings of the Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, 2009.
- [15] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press/Addison-Wesley Publishing Co., 2000.
- [16] J. Bigün, G.H. Granlund, and J.Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991.
- [17] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- [18] M. Black and A. Jepson. Estimating multiple independent motions in segmented images using parametric models with local deformations. *IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, 1994.
- [19] M. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 1995.
- [20] Michael J. Black, David J. Fleet, and Yaser Yacoob. Robustly estimating changes in image appearance. *Computer Vision and Image Understanding*, 78:8–31, 2000.
- [21] Michael J. Black and Allan D. Jepson. Estimating optical flow in segmented images using variable-order parametric models with local deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):972–986, 1996.

- [22] M.J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, January 1996.
- [23] M.J. Black, Y. Yacoob, A. Jepson, and D. Fleet. Learning parameterized models of image motion. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997.
- [24] M. Bleyer, M. Gelautz, and C. Rhemann. Segmentation-based motion with occlusions using graph-cut optimization. In *Symposium of the German Association for Pattern Recognition (DAGM)*, 2006.
- [25] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision, Proceedings*, pages 25–36, 2004.
- [26] A. Bruhn and J. Weickert. *A Confidence Measure for Variational Optic Flow Methods*, pages 283–298. Springer Netherlands, 2006.
- [27] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr. Real-time optic flow computation with variational methods. *IEEE Transactions in Image Processing*, 14(5):608–615, 2005.
- [28] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.
- [29] Tony F. Chan and Jianhong Shen. Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math*, 62:1019–1043, 2001.
- [30] Tony F. Chan and Jianhong Shen. Non-texture inpainting by curvature-driven diffusions. *J. Visual Comm. Image Rep*, 12:436–449, 2001.
- [31] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of the International Conference on Image Processing*, pages 168–172. IEEE Computer Society, 1994.
- [32] Pierre Comon. Independent component analysis, a new concept? *Signal Processing*, 36:287–314, 1994.
- [33] F. De la Torre and M. J. Black. A framework for robust subspace learning. *International Journal on Computer Vision*, 54(1-3):117–142, 2003.
- [34] Andrew P. Duchon, William H. Warren, and Leslie Pack Kaelbling. Ecological robotics. *Adaptive Behavior*, 6, 1994.

## Bibliography

---

- [35] G. Farnebäck. [http://lsvn.lysator.liu.se/svnroot/\linebreakspatial\\\_domain\\\_toolbox/trunk](http://lsvn.lysator.liu.se/svnroot/\linebreakspatial\_domain\_toolbox/trunk).
- [36] G. Farnebäck. Fast and accurate motion estimation using orientation tensors and parametric motion models. In *International Conference on Pattern Recognition, Proceedings*, volume 1, pages 135–139, Barcelona, Spain, September 2000.
- [37] G. Farnebäck. Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. In *International Conference on Computer Vision, Proceedings*, volume I, pages 171–177, Vancouver, Canada, July 2001.
- [38] M. Felsberg, S. Kalkan, and N. Krüger. Continuous dimensionality characterization of image structures. *Journal of Image and Vision Computing*, 2008.
- [39] C. Fennema and W. Thompson. Velocity determination in scenes containing several moving objects. *Computer Graphics and Image Processing*, 9:301–315, 1979.
- [40] C. Fermüller, D. Shulman, and Y. Aloimonos. The statistics of optical flow. *Journal of Computer Vision and Image Understanding*, 82(1):1–32, 2001.
- [41] R.A. Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, 1925.
- [42] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.
- [43] B. Furht, J. Greenberg, and R. Westwater. *Motion Estimation Algorithms for Video Compression*. Springer-Verlag GmbH, 1996.
- [44] B. Galvin, B. McCane, K. Novins, D. Mason, and S. Mills. Recovering motion fields: An analysis of eight optical flow algorithms. In *Proceedings of the 1998 British Machine Vision Conference*, 1998.
- [45] Christoph S. Garbe, Uwe Schimpf, and Bernd Jähne. A surface renewal model to analyze infrared image sequences of the ocean surface for the study of air-sea heat and gas exchange. *Journal of Geophysical Research*, 109:1–18, 2004.
- [46] Christoph S. Garbe, Hagen Spies, and Bernd Jähne. Estimation of surface flow and net heat flux from infrared image sequences. *Journal of Mathematical Imaging and Vision*, 19(3):159–174, 2003.
- [47] A. Giachetti, M. Campani, and V. Torre. The use of optical flow for road navigation. *Transactions on Robotics and Automation*, 14:34–48, 1998.

- [48] D. Goel and Tsuhan Chen. Real-time pedestrian detection using eigenflow. In *Proceedings of the International Conference on Image Processing (ICIP)*, volume 3, pages 229–232, 2007.
- [49] N. Hata, A. Nabavi, W. Wells, S. Warfield, R. Kikinis, P. Black, and F. Jolesz. Three-dimensional optical flow method for measurement of volumetric brain deformation from intraoperative magnetic resonance images. *Journal of Computer Assisted Tomography*, 24:531–538, 2000.
- [50] H. Haussecker and D. Fleet. Computing optical flow with physical models of brightness variation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 23(6):661–673, 2001.
- [51] H. Haussecker and H. Spies. Motion. In B. Jähne, H. Haussecker, and P. Geissler, editors, *Handbook of Computer Vision and Applications*, volume 2, chapter 13, pages 336–338. Academic Press, 1999.
- [52] D. Heeger. Model for the extraction of image flow. *Journal of the Optical Society of America*, 4(8):1455–1471, 1987.
- [53] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.
- [54] M. Irani and P. Anandan. Robust multi-sensor image alignment. In *Proceedings of the International Conference on Computer Vision*, pages 959–966, 1998.
- [55] Scholz J., Wiersbinski T., Ruhnau P., Kondermann D., Garbe C.S., Hain R., and Beushausen V. Double-pulse planar-lif investigations using fluorescence motion analysis for mixture formation investigation. *Experiments in Fluids*, 45(4):583–593, 2008.
- [56] B. Jähne. *Digital Image Processing*. Springer Verlag, 2002.
- [57] I. T. Jolliffe. *Principal Component Analysis*. Springer, 1986.
- [58] S. X. Ju, M. J. Black, and A. D. Jepson. Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996.
- [59] S. Kalkan, D. Calow, M. Felsberg, F. Wörgötter, M. Lappe, and N. Krüger. Optic flow statistics and intrinsic dimensionality. In *booktitle = Proc. of Brain Inspired Cognitive Systems*, 2004.
- [60] S. Kalkan, D. Calow, and Wörgötter. Local image structures and optic flow estimation. In *Network: Computation in Neural Systems*, 2005.

## Bibliography

---

- [61] K.R. Koch. *Parameterschätzung und Hypothesentests in linearen Modellen*. Ferd. Dümmlers Verlag Bonn, 2004.
- [62] C. Kondermann, D. Kondermann, and C. Garbe. Postprocessing of optical flows via surface measures and motion inpainting. In *Pattern Recognition*, volume 5096 of *LNCS*, pages 355–364. Springer, 2008.
- [63] C. Kondermann, D. Kondermann, and C. Garbe. The evaluation of optical flow estimators. *submitted to Computer Vision and Image Understanding*, 2009.
- [64] C. Kondermann, D. Kondermann, and C. Garbe. Local optical flow estimation based on learned motion models. *submitted to Transactions in Image Processing*, 2009.
- [65] C. Kondermann, D. Kondermann, B. Jähne, and C. Garbe. An adaptive confidence measure for optical flows based on linear subspace projections. In *Pattern Recognition*, volume 4713 of *LNCS*, pages 132–141. Springer, 2007.
- [66] C. Kondermann, R. Mester, and C. Garbe. A statistical confidence measure for optical flows. In *Proceedings of the European Conference of Computer Vision, ECCV*, pages 290–301, 2008.
- [67] B. Kröse, A. Dev, and F. Groen. Heading direction of a mobile robot from the optical flow. *Image and Vision Computing*, 18:415–424, 2000.
- [68] S. Lee, S. Park, N. Cho, Y. Kanatsugn, and J. Park. Occlusion detection and stereo matching in a stochastic method. In *Proceedings of the International Conference on Image Processing (ICIP)*, volume 1, pages 377–380, 2003.
- [69] K. Lim, M. Chong, and A. Das. A new MRF model for robust estimate of occlusion and motion vector fields. In *Proceedings of the International Conference on Image Processing (ICIP)*, volume 2, page 843, 1997.
- [70] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision (DARPA). In *Proceedings of the 1981 DARPA Image Understanding Workshop*, pages 121–130, 1981.
- [71] A. Malla and R. Green. Real-time adaptation to unconstrained lighting for occlusion detection. In *Image and Vision Computing New Zealand*, 2005.
- [72] S. Masnou and J. Morel. Level lines based disocclusion. In *Proceedings of the ICIP 1998*, volume 3, pages 259 – 263, 1998.
- [73] B. McCane, K. Novins, D. Crannitch, and B. Galvin. On benchmarking optical flow. *Computer Vision and Image Understanding*, 84(1):126–143, 2001.

- [74] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: A synopsis of current problems and methods. *International Journal of Computer Vision (IJCV)*, 19(1):2955, 1996.
- [75] C. Mota, I. Stuke, and E. Barth. Analytical solutions for multiple motions. In *Proceedings of the International Conference on Image Processing (ICIP)*, 2001.
- [76] F. Nejadasl, B. Gorte, and S. Hoogendoorn. Optical flow based vehicle tracking strengthened by statistical decisions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61:159–169, 2006.
- [77] C. Nieuwenhuis and M. Yan. Knowledge based image enhancement using neural networks. In *International Conference on Pattern Recognition, Proceedings*, pages 814–817, 2006.
- [78] Tal Nir, Alfred M. Bruckstein, and Ron Kimmel. Over-parameterized variational optical flow. *International Journal of Computer Vision*, 76(2):205–216, June 2006.
- [79] Nishio S. Kobayashi T. Saga T. Takehara K. Okamoto, K. Evaluation of the 3d-PIV standard images (PIV-STD Project). *Journal of Visualization*, 3-2:115–124, 2000.
- [80] M. Otte and H. Nagel. Optical flow estimation: advances and comparisons. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 51–60, 1994.
- [81] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision*, 67(2):141–158, 2006.
- [82] E. Parzen. On the estimation of probability density functions. *Annual Mathematical Statistics*, 33:1065–1076, 1962.
- [83] M. Raffel, C. Willert, and J. Kompenhans. Postprocessing of PIV data. In *Particle Image Velocimetry*, chapter 6. Springer, 1998.
- [84] A. Rosenberg and M. Werman. Representing local motion as a probability distribution matrix applied to object tracking. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 654–659, 1997.
- [85] S. Roth and M.J. Black. On the spatial statistics of optical flow. In *International Conference on Computer Vision, Proceedings*, volume 1, pages 42–49, 2005.
- [86] P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, (79):871–880, 1984.

## Bibliography

---

- [87] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. John Wiley, 1987.
- [88] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [89] H. Scharr. Optimal filters for extended optical flow. In *Complex Motion, Lecture Notes in Computer Science*, volume 3417. Springer, 2004.
- [90] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656, 1948.
- [91] A. Singh. Motion-compensated enhancement of medical image sequences. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 1660, pages 288–298, 1992.
- [92] H. Spies and C. Garbe. Dense parameter fields from total least squares. In *Proceedings of the German Association for Pattern Recognition (DAGM)*, 2002.
- [93] M. Stanislas, Okamoto K., C. Kaehler, and J. Westerveel. Main results of the second international PIV challenge. volume 39, pages 170–191, 2005.
- [94] C. Stiller and J. Konrad. Estimating motion in image sequences. *IEEE Signal Processing Magazine*, 16(4):70–91, 1999.
- [95] R. Strzodka and C. Garbe. Real-time motion estimation and visualization on graphics cards. In *Proceedings of the conference on visualization*, pages 545–552, 2004.
- [96] D. Sun, S. Roth, J.P. Lewis, and M.J. Black. Learning optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008.
- [97] S. Uras, F. Girosi, A. Verri, and V. Torre. A computational approach to motion perception. *Journal of Biological Cybernetics*, 60:79–97, 1988.
- [98] A. Waxman, J. Wu, and F. Bergholm. Convected activation profiles and receptive fields for real time measurement of short range visual motion. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 717–723, 1988.
- [99] 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.
- [100] Y. Yacoob and L. Davis. Learned temporal models of image motion. In *International Conference on Computer Vision, Proceedings*, 1998.

---

*Bibliography*

- [101] C. Zetzsche and E. Barth. Fundamental limits of linear filters in the visual processing of two dimensional signals. *Vision Research*, 30(7):1111–1117, 1990.
- [102] C. Zitnick and T. Kanade. A cooperative algorithm for stereo matching and occlusion detection. *IEEE Journal of Pattern Analysis and Machine Intelligence (PAMI)*, 22(7):675–684, 2000.