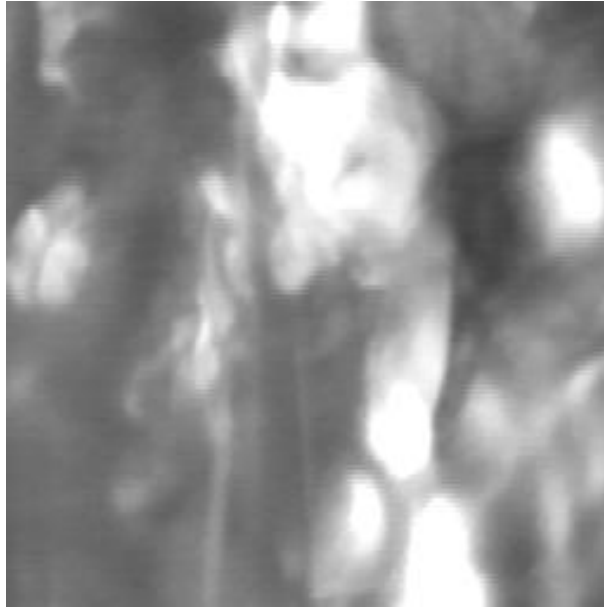


Turbulence in Optical Flow Fields



Kim Steenstrup Pedersen

Technical Report DIKU-TR-00/3
Department of Computer Science
University of Copenhagen
Universitetsparken 1
DK-2100 København Ø
DENMARK

30th January 2001

CR subject classification: I.2.10,J.2,I.4.8,I.4.9,G.1.6

Abstract

In this thesis I will examine the relations between turbulence as found in physical systems and turbulence in optic flow fields. I will record some image sequences of a theoretically well understood turbulent physical system and from these image sequences I will examine the relations between the optic flow of the image sequences and the physical parameters of the turbulent system. I will examine the image sequences at different scales using the theory of linear Gaussian scale-space, because linear Gaussian scale-space has proven to be a valuable tool in digital image processing and because the physics of turbulence can be explained through scaling properties of the turbulent flow. From these analyses I hope to be able to describe the theoretical relations between turbulence in optic flow fields and turbulence in physical systems.

I have obtained the permission to borrow and use equipment and experimental setup for the recording of the image sequences by Preben Alstrøm and Mogens Levinsen from the Niels Bohr Institute (NBI).

This abstract is a translated version of the description of the project (“Arbejdsbeskrivelse”), delivered to the administration of DIKU at the start of this project.

Contents

1	Preface	3
2	Introduction	4
3	Linear Gaussian Scale-Space	6
3.1	Theoretical Foundation of Linear Gaussian Scale-Space	7
3.1.1	The Diffusion Equation Formulation	8
3.1.2	The Axiomatic Formulation	10
3.2	Spatiotemporal Scale-Space Representations	12
3.3	Properties of Scale-Space	14
3.4	Implementation Issues	17
3.5	Summary	18
4	Optic Flow	20
4.1	The Background of Optic Flow	20
4.1.1	Categorisation of Optic Flow Methods	21
4.1.2	The Optic Flow Constraint Equation	22
4.1.3	The Aperture Problem	22
4.2	Optical Flow Fields Extracted From One Image	24
4.2.1	Implementation of Algorithm 4.1	27
4.2.2	Performance of Algorithm 4.1	27
4.3	Normal Flow Field Estimation Using Scale-Space	28
4.3.1	Scale Selection and Normal Flow Uncertainty Measurement	32
4.4	Optical Flow Field Estimation Using an Integration Scale-Space	32
4.4.1	Integration Scale-Space	33
4.4.2	The Complete Optic Flow Algorithm	38
4.4.3	Scale Selection and Uncertainty Measurement in Optic Flow	38
4.5	Singularities in Optic Flow Fields	41
4.6	Summary	44
5	Evaluation of Optic Flow Methods	45
5.1	Measuring Errors in Optical Flow Fields	45
5.2	Collection of Test Data	46
5.3	Results	47
5.4	Summary	60

6	Turbulence	66
6.1	The Navier-Stokes Equations	67
6.2	Scale-Space Energy Budget	70
6.3	Kolmogorov's 1941 Theory	72
6.4	Experimental Laws of Turbulence	75
6.5	Summary	77
7	Turbulence in Optical Flow Fields	79
7.1	The Faraday Experiment	79
7.2	Collection of Experimental Data	80
7.3	Results	82
7.4	Summary	91
8	Conclusion	93
A	Notation	96
B	Proofs	97
B.1	Proof of Theorem 4.1	97
B.2	Proof of Theorem 4.2	98
C	Examples of the Faraday Image Sequences	102
D	Implementation	105
D.1	Implementation of Linear Gaussian Scale-Space	105
D.1.1	LinearScaleSpace	105
D.1.2	CalcInvariance	114
D.1.3	The Kims.Speciale.Utils package	121
D.2	Optic Flow Implementations	134
D.2.1	Implementation of Algorithms	134
D.2.2	Functions for Measurements	141
D.2.3	Auxiliary Functions	145

Chapter 1

Preface

This work is the masters thesis of Kim Steenstrup Pedersen. The thesis was written in the period of November 1998 to August 1999, with a preliminary study throughout 1998. The work was supervised by associate professor Mads Nielsen, previously at Department of Computer Science, University of Copenhagen, now at IT Højskolen in Copenhagen.

I would like to thank associate research professor Preben Alstrøm and associate professor Mogens Levinsen, CATS, Niels Bohr Institute, for lending me equipment and advice on turbulence. Furthermore, I would like to thank Elisabeth Schroll for helping me proof-read the text of this thesis.

Chapter 2

Introduction

The phenomena of turbulence has fascinated man throughout time, because we encounter it in our everyday life. We see it in meteorological phenomena such as sky formations and storms, and in a pot of boiling water. Turbulent flow has been studied back to the sixteenth century, where the famous Italian artist and scientist Leonardo Da Vinci made presumably the first studies of turbulence, [20]. In our century Richardson has proposed that turbulence arises from the interactions of the flow at different scales and in 1941 Kolmogorov made a substantial contribution to our knowledge of turbulence with his K41 theory, [20]. This theory states a set of scaling laws of turbulence, which describes its qualitative behaviour through statistical statements. Turbulence has been studied by either simulation or evaluation of experimental data. Simulation of turbulence is done by using Computational Fluid Dynamics techniques (CFD). Evaluation of experimental data is done by methods such as Particle Imaging Velocimetry (PIV), Particle Tracking Velocimetry (PTV), pattern recognition methods, or optic flow techniques, [61].

In both physics and computer vision, scale plays an important role. In order to understand the phenomena of turbulence, one has to acknowledge the fact that turbulence is a multi-scale phenomenon. Turbulence exists and interacts at multiple scales and turbulence has been theoretically described through scaling laws. In computer vision the knowledge of the scale of phenomena has been used as a tool for analysis throughout the field, [26, 54, 28, 11, 12, 60, 30]. A theory called linear Gaussian scale-space has emerged, [26, 60, 30], which introduces a formalism for representing and analysing digital images at multiple scales.

A well founded way of inferring information about motion from digital images is the so-called optic flow methods, in which information about motion is extracted from the changing intensity patterns of image sequences. This type of methods has been used for investigation of fluid flow and turbulence in general, [42, 32, 41, 45, 61]. Furthermore, optic flow methods have been formulated using the concept of scale through linear Gaussian scale-space, [16, 17, 47, 44, 18].

My hypothesis is therefore:

It is possible to examine the qualitative behaviour of fluid flow using a properly chosen optical flow field model and especially to examine the scaling properties of turbulent flow using linear Gaussian scale-space.

The purpose of this thesis is therefore two fold. The first purpose is to examine the scaling properties of turbulence using linear Gaussian scale-space optic flow techniques. The second is to introduce a new algorithm for estimation of optical flow fields using linear Gaussian scale-space, which my supervisor Mads Nielsen inspired me to develop.

The algorithm I propose uses an integration scale-space over a given normal flow field and a polynomial expansion model of the sought optic flow as the basis for estimation of the optic flow field. I let the uncertainty of the underlying normal flow have an influence on the chosen direction of the optic flow. Furthermore, I propose a scale selection mechanism, which can be used in conjunction with this algorithm. I will implement and evaluate the performance of the proposed optic flow algorithm.

I will also describe two optic flow algorithms proposed by Maurizot et al. [41] and Florack et al. [16, 17, 44, 18]. I will implement both of these algorithms, but only evaluate the performance of the method proposed by Florack et al. [16, 17, 44, 18].

The evaluation of the performance of the different optic flow methods discussed in this thesis is done using the methods proposed by Barron et al. [7] in their survey on performance of different optic flow methods. The performance results in this thesis will be compared with what different authors have found for other optical flow techniques, [7, 57, 18].

In order to investigate the scaling properties of turbulent flow using linear Gaussian scale-space optic flow techniques, I have chosen to use the optic flow method, which will be proposed in the thesis. I use this method to investigate different experimental laws of turbulence.

The Structure of This Thesis

The reader of this thesis is assumed to have a reasonable knowledge of the basic theory of digital image processing. Furthermore, it is an advantage to be familiar with the basic concepts and language of physics. But in order to insure that the reader of this thesis is acquainted with the necessary theory of linear Gaussian scale-space I will in Chapter 3 introduce parts of this theory and notation used throughout this thesis. Furthermore, a short list of the mathematical notation used in this thesis can be found in Appendix A.

In Chapter 4 I introduce the concepts of optic flow methods. I describe two methods proposed by Maurizot et al. [41] and Florack et al. [16, 17, 44, 18]. I also propose a new method for optical flow field estimation and a corresponding scale selection mechanism. At the end of this chapter I will briefly discuss singularities of optical flow fields and introduce a method for detection of singular points in optical flow fields.

In Chapter 5 I evaluate the performance of the optic flow method proposed in this thesis and the normal flow method proposed by Florack et al. [16, 17, 44, 18].

An introduction to the basic theory of turbulence formulated in terms of linear Gaussian scale-space can be found in Chapter 6, where I also introduce a set of experimental laws, which I propose, can be evaluated using scale-space based optical flow techniques.

In Chapter 7 a presentation and discussion of the results of the examination of turbulence using optic flow methods can be found. Examples of the image sequences and corresponding flow fields used in this evaluation has been placed in Appendix C.

At the end of this thesis I will make an overall conclusion and discuss the results of this thesis.

All implementations described and used in this thesis are either done in the Java programming language or in the language of the Matlab mathematics system. Listings of the source code for all implementations described and used in this thesis can be found in Appendix D.

Chapter 3

Linear Gaussian Scale-Space

In the computer vision and digital image processing community an interest for different types of multi-scale image representations has been growing for some time, [26, 54, 28, 11, 12, 60, 30]. Multi-scale image representations can be used in the solution of various computer vision problems, from signal representation [26, 28, 11, 12, 60, 30, 31, 36] over feature detection [54, 33, 34, 35] and segmentation [48] to estimation of motion and analysis of stereo images [16, 17, 18, 45, 43].

The concept of scale is well known in physics, where scale is taken into account, when conducting a physical measurement of some property, and when describing physical properties theoretically. The physicist uses the scale most suitable for the analysis or measurement. That is, he does not use the number of atoms in an iron rod to account for its length, but uses instead a macroscopic scale metric like the SI length unit — meter. From a theoretical perspective both types of measurement are valid, but from a practical point of view, the use of the number of atoms in the iron rod as a measure of its length is not an obtainable measure. In physics we also encounter cases where a measurement is theoretically impossible at certain scales. Take for instance the measurement of the interaction of elemental particles, which can only be described at very small scales. In Chapter 6 I will make an account of the theory of turbulence, in which the concept of scale has great importance.

In computer vision and digital image processing the concept of scale can be used for the description of the fact that objects in images has an extension over a set of pixels and that different objects have different sizes. If we are looking for the contour of a car in an image, we should probably ignore details on the car, like the tyres and rear-view mirrors. Furthermore, there might be other things in the image like trees, a road, and a shrubbery that could disturb our finding of the cars contour. We therefore need to focus on objects of the same size as the whole car.

The capturing of digital images can be seen as a measurement with some device or filter. The measuring device or filter has a certain resolution at which the image is obtained and this resolution is determined by the size of the so-called aperture of the measuring device or filter.

A digital image incorporates two limits of the possible scales of objects within it. We talk about the inner scale, which normally is given by the pixel grid and is a lower limit on the scales of objects in the image¹. Each pixel represents a small region of intensity coming from the objects captured in the image. We cannot, without “guessing”, get more information out of

¹Later on in this chapter, I will explain that by using linear Gaussian scale-space on an image, we can in fact change the inner scale of that image.

the details of the image, than the pixel grid size allows. We can of course interpolate between the pixel intensity values to obtain inter-pixel values, but by doing so we introduce a model of the pixel intensity function and we thereby feed information into the image representation. The inner scale is defined as the intrinsic size of the aperture of the measuring device.

An image also incorporates an upper limit of the possible scales of objects within the image and we denote this limit the outer scale. The outer scale of an image is given by the size of the image plane or rather the size of the view port we have captured the image through, e.g. a camera lens. This means that we can only discern objects, which are completely within the image boundaries. For instance, we would not know whether we are looking at a small house or a big villa, by only looking at a small section of a brick of a house wall.

Lindeberg gives in his book on scale-space theory [34], an overview of different multi-scale representations used in digital image processing. He explains that back in 1971 Rosenfeld and Thurston [54] used edge detection filters of different size, hereby using larger regions for edge detection.

The same year, Klinger [28] introduced the quad-tree representation. This is, as the name implies, a tree representation where each leaf is a quadratic segment of the original image. Each leaf image segment is chosen in such a way that some measure of variation Σ in the image segment is smaller than some threshold α . The measure of variation Σ could for instance be the variance of the grey level values in the image segment.

Lindeberg continues with an account of the pyramid multi-scale representation. In 1981 Burt and Crowley [11, 12] proposed independently the idea of low-pass filtered image pyramids. A pyramid representation of an image is obtained by an iterative process of applying a linear smoothing filter followed by a down-sampling. By this process a pyramid of images can be build, where the base of the pyramid is the original image and on top of this image lies a set of smaller and smaller, smoothed and down-sampled versions of the original image. The linear low-pass filters proposed for smoothing is a local mean filter or a Gaussian filter. Band-pass pyramids or the so-called Laplacian pyramids can be implemented by constructing two low-pass pyramids with different band widths and then finding the difference between these two pyramids.

In the rest of this chapter I will be concerned with a multi-scale representation called linear Gaussian scale-space, [26, 60, 30]. The linear Gaussian scale-space theory gives us a representation that is a one-parameter family of scale functions. The scale function is called the aperture function, because it enables us to recapture the image and change the inner scale of the image. The aperture function is continuous in the scale parameter and we can therefore produce a continuous range of images at different scales. We call this range, the scale-space of the original image.

In this chapter I will introduce the linear Gaussian scale-space theory. In Sec. 3.1 I will state the theoretical foundation of linear Gaussian scale-space and in Sec. 3.2 I will introduce possible extension of linear Gaussian scale-space to include a temporal dimension, which could be time causal. After this I will explain a set of important properties of linear Gaussian scale-space in Sec. 3.3. This chapter ends with Sec. 3.4 in which a discussion of a possible implementation of a linear Gaussian scale-space image representation can be found.

3.1 Theoretical Foundation of Linear Gaussian Scale-Space

Linear Gaussian scale-space was, according to Weickert et al. [58], independently introduced by Iijima [26], Witkin [60] and Koenderink [30]. They based their derivation of a linear scale-

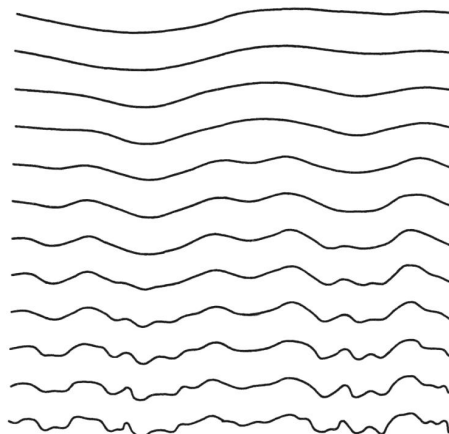


Figure 3.1: Samples from the scale-space of a 1D signal. The signal at the bottom of the drawing is the original signal at scale zero and the scale parameter increases from the bottom and up. The drawing is an adaption from Witkin [60].

space representation on an axiomatic formulation. Koenderink was inspired by the behaviour of physical heat diffusion governed by the heat diffusion equation, and he therefore stated a set of axioms, which leads to a scale-space representation based on the solution to the heat diffusion equation. The remarkable thing is that they all independently arrived at the same representation, namely to use Gaussian convolution as a generator for the scale-space representation.

The linear Gaussian scale-space representation is a continuous one parameter family of Gaussian smoothed versions of the raw image, where the continuous parameter corresponds to the scale. This results in a continuous range of smoothed images called the scale-space. In Fig. 3.1 and Fig. 3.2 examples of scale-spaces for a 1 dimensional signal and a 2 dimensional image is shown.

In the following sections I will outline the diffusion equation formulation and the axiomatic formulation of linear Gaussian scale-space.

3.1.1 The Diffusion Equation Formulation

Koenderink [30] stated a set of axioms or properties that he thought necessary of a scale-space representation. From these properties he derived the linear Gaussian scale-space as the solution of the heat diffusion equation.

The heat diffusion equation models the effect of heat diffusion in a material. Imagine a plate of metal in a thermally isolated space, i.e. heat cannot escape away from the metal plate. The plate is heated at its centre at some initial moment in time t_0 . When time evolves the heat will be diffused out into the rest of the plate and after some time has elapsed, depending on the size and heat conductivity properties of the metal plate, the heat is distributed evenly out into the plate. This behaviour is exactly what Koenderink [30] was looking for in a scale-space representation of images. The analogy to heat diffusion is that the intensity pattern of an image can be looked upon as a distribution of heat and when “time” evolves, the heat is redistributed evenly. The time parameter should now be looked upon as the scale parameter of the image.

The linear Gaussian scale-space representation can in the heat diffusion formulation be stated as the following definition:

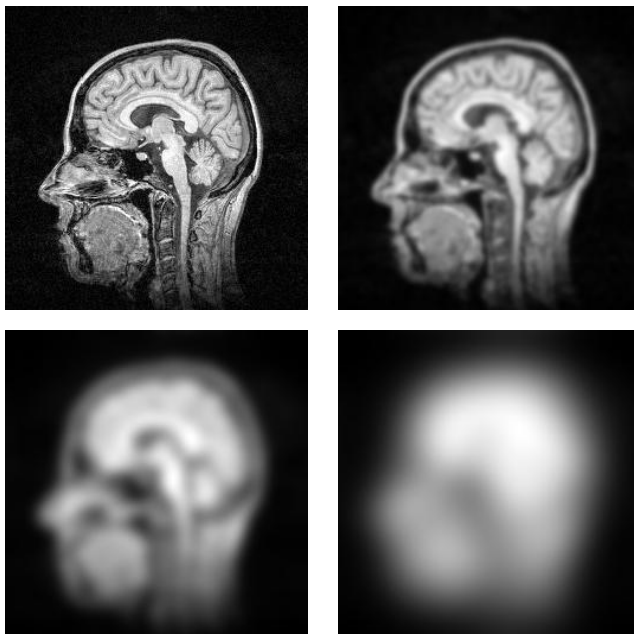


Figure 3.2: Samples of the scale-space of a 2-dimensional image. The original 256×256 pixel image is shown in the upper left corner and the scale parameter increases from left to right and down. The scale parameter σ take on the values 0, 2, 6 and 16 pixels.

Definition 3.1 (Linear Gaussian Scale-Space)

The linear Gaussian scale-space of the N -dimensional raw image $f(\vec{x}) : \mathbb{R}^N \mapsto \mathbb{R}$ is defined as the solution of the diffusion equation,

$$\partial_t L = \nabla^2 L$$

where $t = 2\sigma^2$ and σ is the scale parameter. The solution $L(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \mapsto \mathbb{R}$ of this differential equation with the boundary condition that $L(\vec{x}; 0) = f(\vec{x})$, is given by convolution of the raw image $f(\vec{x})$ with the N -dimensional Gauss function $G(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$,

$$L(\vec{x}; \sigma) = \int_{\vec{\xi} \in \mathbb{R}^N} f(\vec{\xi}) G(\vec{x} - \vec{\xi}; \sigma) d\vec{\xi} = f(\vec{x}) *_{\vec{x}} G(\vec{x}; \sigma) \tag{3.1}$$

where $f(\vec{x}) *_{\vec{x}} G(\vec{x}; \sigma)$ denotes convolution² of $f(\vec{x})$ and $G(\vec{x}; \sigma)$ in the variable \vec{x} .

The Gauss function is the so-called Green's³ function of the diffusion equation. The N -dimensional Gauss function $G(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is given by

$$G(\vec{x}; \sigma) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{\vec{x} \cdot \vec{x}}{2\sigma^2}\right)$$

where the standard deviation σ of the Gauss function is the scale parameter. Some authors choose to use the double variance $t = 2\sigma^2$ as the scale parameter.

²In this thesis I will some times just write $f(\vec{x}) * g(\vec{x})$ whenever it is obvious over which variable the convolution is done.

³A solution to a non-homogeneous differential equation can be expressed using the Green's function of the problem. See for instance [13] for an explanation of Green's functions.

Changing the scale parameter σ , effectively corresponds to changing the inner scale of the image. This could be seen as a change of resolution without a down-sampling. Furthermore, by choosing the boundary condition $L(\vec{x}; 0) = f(\vec{x})$ we have introduced a natural lower bound for the scale-space. When we increase the scale parameter σ , the original image becomes increasingly blurred and when σ goes toward infinity the image intensity values goes toward the mean value of the image intensity function.

In the following section I will explain some of the axioms used in the axiomatic deduction of the linear Gaussian scale-space representation.

3.1.2 The Axiomatic Formulation

Weickert et al. [58] give an overview of the axioms used by different authors in axiomatic deductions of the linear Gaussian scale-space representation.

The axiomatic deduction of the linear Gaussian scale-space representation can be stated as the search for the scale-space operator

$$\mathbf{T}_t : f(\vec{x}) \mapsto L(\vec{x}; t).$$

Here I list some of the most important axioms used by various authors in the deduction of linear Gaussian scale-space:

- **Convolution kernel:**

The operator \mathbf{T}_t should be translational invariant and a linear integral operator.

This axiom implies that \mathbf{T}_t is a convolution operator and it can for that reason be stated as

$$(\mathbf{T}_t f)(\vec{x}) = \int_{\vec{x}' \in \mathbb{R}^N} f(\vec{x}') k_t(\vec{x} - \vec{x}') d\vec{x}',$$

where $\{k_t : \mathbb{R}^N \mapsto \mathbb{R} | t \geq 0\}$ is a one-parameter family of convolution kernels.

- **Causality:**

Isophotes at spatial extremal points in scale-space should be convex in the direction of the increasing scale parameter.

This can for 1-dimensional signals be restated as the condition that no new extremal points should be created when the scale parameter increases. This can also be stated as the condition that no new zero crossings of an arbitrary order of derivatives are created when the scale parameter increases, [34]. This leads to the fact shown in Fig. 3.3 and in the words of Witkin [60] that scale-space paths of zero crossings are never closed from below.

This non-creation of extremal points is not the case for higher dimensionalities, where creation of extremal points is a generic event.

If this axiom is satisfied we are able to track features at coarse scale down through the scale-space. In the words of Koenderink [30] one could say that we are able to find the “cause” of the feature in scale-space.

- **Non-negativity:**

The scale-space kernel should be positive,

$$k_t(\vec{x}) \geq 0 \quad \forall \vec{x}, \forall t > 0.$$

This axiom secures that no new zero crossings will appear in scale-space, because this will require a negative kernel.

- **Semi-group property:**

Let $t_1 > 0$ and $t_2 > 0$, then the semi-group property of the Gaussian scale-space of a function $f(\vec{x})$ can be stated as

$$f(\vec{x}) *_x G(\vec{x}; t_1 + t_2) = (f(\vec{x}) *_x G(\vec{x}; t_1)) *_x G(\vec{x}; t_2).$$

A consequence of the semi-group property of the linear Gaussian scale-space is that all possible scales are handled in the same fashion. This means that the result of applying the scale-space operator T_t with scale $t > 0$ on an image, can be seen as a new image with t as its inner scale.

Furthermore, the semi-group property of the Gaussian scale-space allows us to implement the scale-space as an iterative process of smoothing of the raw image.

- **Normalisation:**

The normalisation condition can be stated as

$$\int_{\vec{x} \in \mathbb{R}^N} k_t(\vec{x}) d\vec{x} = 1.$$

This axiom implies that the scale-space representation is intensity average invariant,

$$\int_{\vec{x} \in \mathbb{R}^N} T_t f(\vec{x}) d\vec{x} = \int_{\vec{x} \in \mathbb{R}^N} f(\vec{x}) d\vec{x}.$$

- **Isometry Invariance:**

Let $\mathbf{R} \in \mathbb{R}^{N \times N}$ be a rotational transformation and $(\mathbf{R}f)(\vec{x}) = f(\mathbf{R}\vec{x})$, then isometry invariance implies that

$$\mathbf{T}_t(\mathbf{R}f) = \mathbf{R}(\mathbf{T}_t f) \quad \forall f, \forall t > 0.$$

This axiom could also be named rotational invariance.

In the case of the spatiotemporal scale-space representation used in this thesis, rotational invariance apply for rotations in space only or mirror projection in the temporal dimension only. That is, isometry invariance is violated when rotating space into time or vice versa.

- **Separability:**

The convolution kernel $k_t(\vec{x})$ should be separable in the components of $\vec{x} = (x_1, \dots, x_N)$. That is,

$$k_t(\vec{x}) = k_{1,t}(x_1) \cdots k_{N,t}(x_N).$$

This axiom could also be named projection invariance, because separability insures, in the 3-dimensional case, that the projection of a 3-dimensional scale-space onto a 2-dimensional scale-space along one of the 3-dimensional scale-space dimensions, corresponds to the two other dimensions of the 3-dimensional scale-space being equivalent with the 2-dimensional scale-space.

From an implementation point of view this axiom is valuable, because separability gives us the opportunity to construct the N dimensional convolution kernel as the product of a set of 1 dimensional convolution kernels.

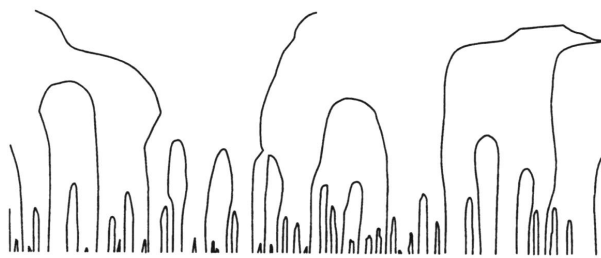


Figure 3.3: This schematic drawing of the zero crossings in scale-space of an arbitrary order of derivatives of a 1D signal, shows the fact that no new zero crossings are created when the scale parameter increases (the upwards direction on the drawing). It also shows the fact that annihilation of zero crossings or extremal points are possible. The drawing is adapted from Witkin [60].

- **Homogeneity and isotropy:**

All spatial positions and scale parameters values should be treated in the same manner.

- **Scale invariance:**

Let \mathbf{S} be a scaling operator defined by $(\mathbf{S}_\lambda f)(\vec{x}) = f(\lambda\vec{x})$. Then a $t'(\lambda, t)$ exists so that

$$S_\lambda T_{t'} = T_t S_\lambda.$$

Using a subset of these axioms it is possible to derive the linear Gaussian scale-space $L(\vec{x}; \sigma)$ of the image $f(\vec{x})$, as given by convolution with the Gauss function $G(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R} \mapsto \mathbb{R}$, where σ is the scale parameter. We can therefore define the following operator $T_\sigma : f(\vec{x}) \mapsto L(\vec{x}; \sigma)$ as the generator of the linear Gaussian scale-space,

$$T_\sigma f(\vec{x}) = f(\vec{x}) *_{\vec{x}} G(\vec{x}; \sigma) = L(\vec{x}; \sigma). \quad (3.2)$$

That is, the linear Gaussian scale-space representation is given by Eq. (3.1) and furthermore all of the axioms stated above can be shown to be properties of the linear Gaussian scale-space representation.

3.2 Spatiotemporal Scale-Space Representations

Koenderink [31] proposes a temporal causal scale-space representation, which handles the fact that time has a preferred direction and that we have a present moment and a past. Furthermore, we usually do not have access to the future.

Koenderink introduces a logarithmic scaling $s(t)$ of the temporal dimension,

$$s(t) = \ln \frac{t_0 - t}{\tau},$$

where $s(t)$ is a transformation of the time $t \in [-\infty, t_0]$ into the logarithmic domain of $s(t)$. The parameter t_0 is the present moment and τ is some arbitrary time constant. The function $s(t)$ prevents us from reaching the present and going into the future, because $s(t)$ is a logarithmic function.

A causal temporal scale-space representation can then be stated as the solution to the following temporal diffusion equation

$$\partial_{\sigma_t} L = L_{ss}, \quad (3.3)$$

where $L(t; \sigma_t) : \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is the temporal scale-space image at the temporal scale σ_t .

In the domain of $s(t)$ a solution $M(s; t_0, \sigma_t) : \mathbb{R} \times \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ to the temporal diffusion equation Eq. (3.3) is given by convolution of the original temporal signal $f(t) : \mathbb{R} \mapsto \mathbb{R}$ with a Gauss function $G(s; \sigma_t) : \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$

$$M(s; t_0, \sigma_t) = f(t_0 - \tau \exp(s)) *_s G(s; \sigma_t)$$

where the Gauss kernel is given by

$$G(s; \sigma_t) = \frac{1}{\sqrt{4\pi\sigma_t}} \exp\left(-\frac{s^2}{4\sigma_t}\right).$$

In the temporal domain t a solution $L(t; \delta, \sigma_t) : \mathbb{R} \times \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ to the temporal diffusion equation Eq. (3.3) changes into a convolution of the past with a kernel $Q(t; \delta, \sigma_t) : \mathbb{R} \times \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$,

$$L(t; \delta, \sigma_t) = \int_{-\infty}^{t_0} f(t') Q(t_0 - t'; \delta, \sigma_t) dt'$$

where $f(t)$ is the original temporal image, $\delta = t_0 - t$ is a temporal delay, σ_t is the temporal scale, and the convolution kernel is

$$Q(t; \delta, \sigma_t) = \frac{1}{\delta \sqrt{4\pi\sigma_t}} \left(\frac{t}{\delta}\right)^{-\frac{1}{4\sigma_t} \ln\left(\frac{t}{\delta}\right) - 1}.$$

The convolution kernel $Q(t; \delta, \sigma_t)$ is, in the words of Koenderink, to be seen as the “specious present” or the “blurred present” temporal operator.

Koenderink notes that temporal scale-space derivatives can be obtained by differentiation in the s domain and then transforming the result into the temporal domain. Furthermore, he notes that a spatiotemporal scale-space representation can be obtained by creating the spatial scale-space representation at all moments in time and then for each point in the original image create a temporal scale-space of the stack of spatial scale-space images. This corresponds to using the separability property of linear Gaussian scale-space and computing the product of the spatial scale-space with the temporal scale-space.

Lindeberg [37, 36] has investigated Koenderink’s ideas about a time causal scale-space representation and he proposes another scale-space representation with a causal time direction. Lindeberg proposes both a continuous and a discrete temporal scale-space representation. He states that in the case of continuous time, the only primitive convolution kernels are the truncated exponential kernels given by,

$$h(t; \mu) = \frac{1}{\mu} \exp\left(-\frac{t}{\mu}\right) \quad (t > 0),$$

where μ is the scale parameter. This continuous temporal scale-space representation is not a semi-group.

In the case of discrete time he proposes convolution with discrete Poisson kernels, which have the following filter coefficients

$$p(n; \mu) = \frac{\mu^n}{n!} \exp(-\mu)$$

where n is the n th discrete time moment. Lindeberg states that the discrete time scale-space representation forms a semi-group structure and that the discrete Poisson kernel case is the only, which forms a semi-group structure.

Lindeberg notes that a spatiotemporal scale-space representation can be obtained by using the separability axiom and computing the product of the spatial scale-space with the temporal scale-space, independent of whether the continuous or the discrete kernels are used.

In this thesis I am going to use a spatiotemporal scale-space representation based on the 3-dimensional linear Gaussian scale-space, where I will use a 3-dimensional spatiotemporal Gaussian convolution kernel, for which the temporal scale parameter is explicit. This means that I assume that I can look into the future and that I let both the past and the future have equal influence at a certain moment in time. By doing this I can use the linear Gaussian scale-space representation described in Definition 3.1 by using the spatiotemporal Gauss function $G(\vec{x}; \sigma_s, \sigma_t) : \mathbb{R}^{N+1} \times \mathbb{R}_+^2 \setminus \{0\} \mapsto \mathbb{R}$, which is given by

$$G(\vec{x}; \sigma_s, \sigma_t) = \frac{1}{(2\pi\sigma_t^2)^{1/2}(2\pi\sigma_s^2)^{N/2}} \exp\left(-\frac{x^i \cdot x^i}{2\sigma_s^2} - \frac{t^2}{2\sigma_t^2}\right),$$

where x^i is the spatial vector of $i = 1, \dots, N$ components, $t = x^0$ is the time dimension, and σ_s and σ_t are respectively, the spatial and the temporal scale parameter. I will sometimes use the shorthand, $\vec{\sigma} = (\sigma_s, \sigma_t)^T$, for the spatiotemporal scale parameters.

We can then define a spatiotemporal scale-space operator $T_{\vec{\sigma}} : f(\vec{x}) \mapsto L(\vec{x}; \vec{\sigma})$ at the spatiotemporal scale $\vec{\sigma} = (\sigma_s, \sigma_t)^T$ as

$$T_{\vec{\sigma}}f(\vec{x}) = f(\vec{x}) *_{\vec{x}} G(\vec{x}; \sigma_s, \sigma_t) = \int_{\vec{x}' \in \mathbb{R}^{N+1}} f(\vec{x}') G(\vec{x} - \vec{x}'; \sigma_s, \sigma_t) d\vec{x}' = L(\vec{x}; \vec{\sigma}). \quad (3.4)$$

In this thesis I choose to use the spatiotemporal linear Gaussian scale-space representation described above, because spatial and temporal scales are handled in the same manner, which simplifies the implementation and makes analysis of the representation fairly straight forward. Furthermore, the purpose of this thesis is not to make a real-time implementation of a scale-space representation, which makes the choice of non-causality acceptable, because we can assume that we have knowledge of both the past and the future.

3.3 Properties of Scale-Space

In this section I will state some important properties of the linear Gaussian scale-space representation. Furthermore, I will introduce some notation, which will be used throughout this thesis.

The following commutation relation exists for scale-space derivatives of a differentiable function $f(\vec{x})$

$$L_x = f_x(x) * G(x; \sigma) = f(x) * G_x(x; \sigma), \quad (3.5)$$

because it can be proven that differentiation commutes with convolution. Raw images are in general not differentiable, which invalidates the above stated commutation relation. But by applying the Gaussian scale-space kernel, images become differentiable, because convolution with the C^∞ differentiable Gaussian kernel makes the scale-space image C^∞ differentiable.

A result of the Gaussian convolution of the scale-space solution of Eq. (3.1) and the above stated commutation relation is that partial derivatives of a scale-space image can be computed

by partial differentiation of the Gauss function and then convolving the image with the result. This can be stated as the following definition:

Definition 3.2 (Differentiation in Scale-Space)

The n th order partial derivatives of the scale-space image $L(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ with respect to the i th component of \vec{x} is given by

$$\underbrace{\frac{\partial^n}{\partial x^i \partial x^j \dots \partial x^k}}_n L(\vec{x}; \sigma) = f(\vec{x}) *_{\vec{x}} \underbrace{\frac{\partial^n}{\partial x^i \partial x^j \dots \partial x^k}}_n G(\vec{x}; \sigma),$$

where $f(\vec{x}) : \mathbb{R}^N \mapsto \mathbb{R}$ is the raw image and $G(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is the Gauss function.

It can be shown that scale-space derivatives also satisfy the diffusion equation and for that reason also constitutes a scale-space representation. Note that the scale-space derivatives violates the non-negativity axiom, but it is restored in the scale-space of the derived image for scales higher than the inner scale of the derived image.

In the spatiotemporal frequency domain the spatiotemporal $(N + 1)$ -dimensional Gauss function is

$$\mathcal{F}\{G\}(\omega^t, \vec{\omega}; \sigma_s, \sigma_t) = \exp(-2(\sigma_t^2 |\omega^t|^2 + \sigma_s^2 |\vec{\omega}|^2)). \quad (3.6)$$

The partial derivatives of the $(N + 1)$ -dimensional Gauss function in the spatiotemporal domain with respect to components of \vec{x} can be obtained by usage of the general relation of the Fourier transform for n th order partial derivatives of a differentiable function

$$\mathcal{F}\left\{\underbrace{\frac{\partial^n}{\partial x^i \partial x^j \dots \partial x^k}}_n G(\vec{x}; \sigma_s, \sigma_t)\right\} = \underbrace{(j\omega^i)(j\omega^j) \dots (j\omega^k)}_n \mathcal{F}\{G\}(\vec{\omega}; \sigma_s, \sigma_t) \quad (3.7)$$

where $j = \sqrt{-1}$ and $\mathcal{F}\{\cdot\}$ denotes the Fourier transform. The N -dimensional Fourier transform of a function $f(\vec{x})$ is defined as

$$\mathcal{F}\{f(\vec{x})\} = \int_{\vec{x} \in \mathbb{R}^N} f(\vec{x}) e^{-i\vec{\omega} \cdot \vec{x}} d\vec{x}$$

and the inverse Fourier transform is

$$\mathcal{F}^{-1}\{f(\vec{\omega})\} = \frac{1}{2\pi} \int_{\vec{x} \in \mathbb{R}^N} f(\vec{\omega}) e^{i\vec{\omega} \cdot \vec{x}} d\vec{\omega}.$$

The results in Eq. (3.6) and Eq. (3.7) can be useful in an implementation of a linear Gaussian scale-space generator and I use these two equations in my implementation of such a generator, described in Sec. 3.4.

Gaussian derivatives can be stated as a Gaussian function multiplied by a Hermite polynomial, [34]. This relation can be stated as the following definition.

Definition 3.3 (Gaussian Derivatives)

The n th order derivative of a 1-dimensional Gauss function $G(x; \sigma) : \mathbb{R} \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is given by

$$\frac{d^n}{dx^n} G(x; \sigma) = \left(\frac{-1}{\sigma}\right)^n H_n\left(\frac{x}{\sigma}\right) G(x; \sigma),$$

where $H_n(x)$ is a 1-dimensional Hermite polynomial of order n .

The partial derivatives of an N -dimensional Gaussian is related to the N -dimensional Hermite polynomials as defined in the following definition, [16]:

Definition 3.4 (Partial Gaussian Derivatives)

The n th order partial derivatives of the N -dimensional Gauss function $G(x; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is given by

$$\frac{\partial^n}{\partial x^{i_1} \dots \partial x^{i_n}} G(\vec{x}; \sigma) = \left(\frac{-1}{\sigma}\right)^n \mathcal{H}_{i_1 \dots i_n} \left(\frac{\vec{x}}{\sigma}\right) G(\vec{x}; \sigma).$$

Here $\mathcal{H}_{i_1 \dots i_n}(\vec{x})$ denotes the N -dimensional Hermite polynomial of order n , which is related to the 1-dimensional Hermite polynomial $H_n(x)$ by

$$\mathcal{H}_{i_1 \dots i_n}(\vec{x}) = \prod_{j=1}^N H_{\alpha_j^{i_1 \dots i_n}}(x^j)$$

in which $\alpha_j^{i_1 \dots i_n}$ denotes the number of indices in i_1, \dots, i_n which is equal to j .

The following list of lower order monomials and their relation to the 2-dimensional Gaussian and its derivatives can be constructed from Definition 3.4:

$$\begin{aligned} xG &= -\sigma^2 G_x, \\ yG &= -\sigma^2 G_y, \\ x^2G &= \sigma^4 G_{xx} + \sigma^2 G, \\ xyG &= \sigma^4 G_{xy}, \\ y^2G &= \sigma^4 G_{yy} + \sigma^2 G. \end{aligned} \tag{3.8}$$

I will use these monomials in the construction and proof of Algorithm 4.1 and Theorem 4.2.

In order to obtain dimensionless and scale invariant derivatives it is useful to normalise the partial derivatives and use the so-called natural coordinates, which is the coordinates scaled by σ , that is $\frac{x}{\sigma}$, [34]. The standard method of normalisation is based on dimensional analysis, [19], and can be stated as the following definition:

Definition 3.5 (Normalised scale-space derivatives)

Let $L(\vec{x}; \sigma) : \mathbb{R}^N \times \mathbb{R}_+ \mapsto \mathbb{R}$ be a scale-space image. Then the n th order normalised scale-space derivatives of this image is given by

$$\left(\frac{\partial^n L}{\partial x^{i^n}}\right)_{\text{norm}} = \sigma^n \frac{\partial^n L}{\partial x^{i^n}}.$$

Lindeberg [33, 35] has proposed another method in which the structure of the raw image controls the normalisation factor. Pedersen and Nielsen [50, 51] has augmented this method by introducing the fractal dimension of the underlying image as a control parameter of the normalisation factor. I have in this thesis, chosen to use the standard normalisation method instead of the more advanced normalisation procedures. But for future work, it could be interesting to examine the use of image structure as a control parameter for normalisation of derivatives in the context of optical flow field estimation.

In this thesis I will sometimes use index notation and Einstein's summation convention for indices when describing partial derivatives of functions, e.g. of scale-space images $L(\vec{x}; \sigma)$. In

index notation the expression, f_i , denotes the vector of the first order partial derivatives of the function $f(\vec{x}) : \mathbb{R}^N \mapsto \mathbb{R}$ with respect to all the components x^i , $i = 1, \dots, N$, of $\vec{x} \in \mathbb{R}^N$,

$$f_i = \frac{\partial f}{\partial x^i}.$$

In this notation the 2-dimensional spatial scale-space image gradient can be written as

$$L_i = \nabla L = \left(\frac{\partial L}{\partial x}, \frac{\partial L}{\partial y} \right)^T = (L_x, L_y)^T.$$

When more indices are introduced, these should be varied independently, which means for instance that L_{ij} is the Hessian matrix. In the 2-dimensional case the Hessian matrix is

$$L_{ij} = \begin{pmatrix} L_{xx} & L_{xy} \\ L_{yx} & L_{yy} \end{pmatrix}.$$

Einstein's summation convention for index notation states that whenever a product of partial derivatives includes the same index twice, the terms obtained by varying the index should be added together. This means for instance that the 3D Laplacian can be written as

$$L_{ii} = \nabla^2 L = \sum_{i=x,y,z} L_{ii} = L_{xx} + L_{yy} + L_{zz}.$$

Repeated indices over which we sum, is called contracted indices, while indices that only appear once is called free indices.

If a product of indexed partial derivatives includes n free indices, the product is called an n -tensor. The gradient L_i is an example of a 1-tensor and the Hessian matrix L_{ij} is an example of a 2-tensor. Hilbert has shown that some products of tensors where all indices are contracted are invariant of coordinate transformations from the Euclidean group, which includes rotations and translations. All tensor expressions, which is invariant of Euclidean coordinate transformations are called invariants. The Laplacian L_{ii} is an example of an invariant. Tensor products with free indices are called covariant, and they depend on the chosen coordinate system. The gradient L_i is an example of a covariant tensor.

3.4 Implementation Issues

There are several different ways in which one can implement the linear Gaussian scale-space representation. In his book on scale-space theory [34], Lindeberg discuss two methods of implementation of scale-space representations.

Lindeberg [34] proposes a method for implementation of the linear Gaussian scale-space representation, in which the original image is convolved with a set of discrete and truncated convolution kernels. He proposes to use modified Bessel functions as filter kernels. The other method mentioned by Lindeberg is to convolve the original image with Gauss kernels using the Fourier transform. In this thesis I have chosen this method of implementation, because it is, as will be explained below, fairly straight forward to analyse and implement.

In this method we use the fact that convolution of two functions can be done by multiplication in the Fourier domain, because of the following relation between convolution and the Fourier transform, [10],

$$f(x) * h(x) = \mathcal{F}^{-1} \{ \mathcal{F} \{ f \} \cdot \mathcal{F} \{ h \} \}.$$

As an implementation of the discrete Fourier transform, I will use the Fast Fourier Transform (FFT) algorithm⁴. This algorithm induces some constraints on the transformed signal. For discrete images these constraints is captured by the following definition:

Definition 3.6 (Discrete Images)

A discrete image $I(\vec{x}) : \mathbb{R}^N \mapsto \mathbb{R}$ is a band limited and periodic function.

By assuming that the image is periodic, we can bypass the problem of spectral leakage. Spectral leakage is what we call the errors introduced into a discrete Fourier transformed signal, which is truncated at positions other than a complete period in the spatial domain, [10].

I will assume that the discrete images, used together with the algorithm outlined here, conform to Definition 3.6.

The original image should be Fourier transformed using the FFT algorithm, then multiplied by the Gaussian kernel and at last it is inverse Fourier transformed again using the FFT algorithm. The Gaussian kernels is created in the Fourier domain by use of Eq. (3.6), taking care that the kernel should be periodic, because of the periodic constraint on the discrete Fourier transform described above.

An N dimensional scale-space representation can be obtained by using the separability axiom and convolving the image with one dimensional Gaussian kernels $G(x; \sigma)$ along each of the N dimensions.

A differentiated scale-space image can be obtained by using Definition 3.2 and therefore computing the derivatives of the Gaussian kernel. The Gaussian kernels can be differentiated by use of the relation described in Eq. (3.7). To obtain normalised Gaussian derivatives one should multiply the intensity values with σ^n , where n is the order of differentiation, as captured in Definition 3.5.

The implementation of this algorithm has a performance, which is good compared for instance to convolution in the spatial domain with discrete filters, because the performance is independent of the scale parameter. This is not the case for an algorithm using convolution in the spatial domain, because the size of the convolution kernel is dependent on the scale parameter. A very efficient implementation of the algorithm using convolution via the Fourier domain can be obtained by using specialised hardware, which implement parts of or the complete FFT algorithm.

The source code for my implementation of the linear Gaussian scale-space representation, using the above outlined algorithm, can be found in Appendix D.1.

3.5 Summary

In this chapter I have introduced parts of the theory of linear Gaussian scale-space and I have outlined a method for implementation of this type of scale-space.

The linear Gaussian scale-space representation can be deduced either as a solution to the heat diffusion equation as described in Sec. 3.1.1 or from the axioms described in Sec. 3.1.2. The linear Gaussian scale-space can be generated by convolution of the raw image with Gaussian kernels of different variance.

I have in Sec. 3.2 outlined different types of spatiotemporal scale-space representations of which I choose to use the linear Gaussian scale-space representation with an explicit temporal scale parameter.

⁴See either Numerical Recipes [53] or Brigham [10] for a description of the FFT algorithm.

Important properties described in Sec. 3.3 of the linear Gaussian scale-space representation is that images, which in general is non-differentiable, can be differentiated by partial derivation of the Gaussian kernel prior to the convolution. Furthermore, derivatives of the Gauss kernel can be obtained by using the Hermitian structure of the partial derivatives of the Gauss kernel.

I have chosen to use the Fourier domain implementation method described in Sec. 3.4, because this method is fairly simple to implement, the performance of the algorithm is independent of the scale parameter, and it is also fairly easy to control discretisation effects of this method.

Chapter 4

Optic Flow

In this chapter I will describe the problems associated with the estimation of optical flow fields from image sequences and digital images. Optical flow fields can be defined through the change in intensities of an image sequence, coming from the motion flow field, which can be defined as the projection of a real world 3D motion onto the 2D image plane. In Sec. 4.1 I will explain the fundamental concepts of optical flow fields and outline the concepts of different methods for estimation of these fields.

I will describe two methods proposed by different authors for estimation of optic flow. The first method, described in Sec. 4.2, was proposed by Maurizot et al. [41] and it enables use to extract a pseudo optical flow field from a *single* 2-dimensional image, under the assumption that the underlying physical flow is a 2-dimensional fluid flow with a suitable visual trace element. I have altered this algorithm by adding two linear Gaussian scale-spaces. The second method, which is described in Sec. 4.3, was proposed by Florack et al. [16, 17, 47, 44, 18] and it is a framework for estimation of optic flow from image sequences utilising linear Gaussian scale-space and a model of the flow field. Additional *gauge conditions*, as the authors call these, should be added in order to obtain optical flow fields. In this thesis I will add the so-called normal flow gauge condition in order to obtain the normal flow field of image sequences. The normal flow field is a vector field, which is normal to iso-intensity contours or, as some authors call these contours, isophotes.

In Sec. 4.4 I will propose an extension of the algorithm by Florack et al., in which I fit a model of the optical flow field to a given normal flow field. This is done by integration of the normal flow field under a Gaussian scale-space aperture function. In this algorithm I incorporate a measure of uncertainty of the normal flow field in such a way that points, where the normal flow is well estimated, determine the dominant direction of the optical flow field. I will also propose a method of selecting the most suitable scales for the integration of the normal flow field.

I will end this chapter, in Sec. 4.5, with a brief discussion of singularities in optical flow fields.

4.1 The Background of Optic Flow

Optic flow methods are widely used in computer vision for the analysis of motion in image sequences. The basic idea of optic flow methods is that we want to capture the motion of real world objects from their projection onto a sequence of 2D images. This is not a trivial problem.

In order to understand the problem we first need to define the 2-dimensional motion flow field:

Definition 4.1 (Motion Flow Fields)

The motion flow field is the 2-dimensional velocity field, resulting from the projection of the real world 3-dimensional object motion onto a 2-dimensional image plane.

Unfortunately, we cannot capture the projection of a 3-dimensional object motion onto a 2-dimensional image plane, because the only evidence of object motion in an image sequence, is the variation of intensities. The change (or lack of change) in the intensity at a point does not generally correspond to the real world motion at that point. As an example of this, you can imagine a perfect billiard ball rotating on a table and lit by an ambient light source. The ball is obviously moving, but when we capture an image sequence of the moving ball, we get the exact same intensity pattern in each image of the sequence. From only looking at the image intensities, we would therefore say that the billiard ball were laying still. Further examples that indicate that the motion flow field is not extractable from the intensity of image sequences are, motion induced by moving light sources and occlusion of objects in the sequence. The concept of optical flow fields has therefore been introduced as the velocity field extractable from intensity variations in image sequences. A more formal definition of this concept can be found in Definition 4.2.

There are different reasons why we should be interested in the extraction of motion of real world objects. Knowledge of object motion could be used for tracking objects [14, 32], solving time-to-contact and other robot navigation problems [3, 2, 6], and obtaining structure from motion [4, 5, 14]. Optic flow methods have also been used for fluid flow determination by different people, e.g. Maurizot et al. [42, 41] and Larsen [32], and in this thesis I will use optic flow to analyse physical scaling properties of fluid flow.

4.1.1 Categorisation of Optic Flow Methods

Different kind of methods for estimation of optical flow fields have been proposed. An overview of the different methods can be found in Barron et al. [7] and Larsen [32], and the different methods can be categorised into:

- Region based matching methods
- Frequency based methods
- Gradient based methods

For region based matching it is assumed that it is possible to match local image regions between different time frames. Using the spatial difference between the original region and the matched region it is possible to compute the velocity of that region.

Barron et al. [7] divide the frequency based methods into energy based and phase based methods. Energy based methods use the spatiotemporal energy of the image sequence for the estimation of the velocity field. Phase based methods use the phase behaviour of the band-pass filtered image sequence as a measure for the velocity field.

Gradient based methods utilise the spatiotemporal gradients of the image intensity for the estimation of the velocity field. In this thesis, I will focus on gradient based methods and discuss two existing gradient based methods and introduce a novel gradient based method.

4.1.2 The Optic Flow Constraint Equation

In 1981, Horn and Schunck [25, 24] proposed a gradient based optic flow method. They derived a constraint equation that relates the spatiotemporal change of the image intensity to the velocity of objects in the image. The constraint equation states that intensity should be preserved along the flow. This constraint equation is now known as either the motion constraint equation, the brightness constraint equation, or the optic flow constraint equation. I will now give a formal definition of optical flow fields and outline the derivation of the optic flow constraint equation in the 2-dimensional case.

Inspired by Nielsen and Olsen [45] I define spatiotemporal optical flow fields as:

Definition 4.2 (Spatiotemporal Optical Flow Fields)

The N -dimensional spatiotemporal optical flow field $v(\vec{x}, t) : \mathbb{R}^N \times \mathbb{R} \mapsto \mathbb{R}^N \times \mathbb{R}$ is any vector field preserving the intensity of the image sequence along the flow.

The above stated definition implies that the intensity $I(\vec{x}, t) : \mathbb{R}^2 \times \mathbb{R} \mapsto \mathbb{R}$ of a particular point $\vec{x}(t) = (x(t), y(t))^T$ carried along with the flow should be constant over time. This can, in the formalism of Horn and Schunck [25, 24], be stated as

$$\frac{D}{Dt}I(\vec{x}(t), t) \equiv 0.$$

Applying the common assumption that the temporal velocity component is constant, $v^t = 1$ and using the chain rule for differentiation, we get the optical flow constraint equation

$$\frac{\partial I}{\partial x}v^x + \frac{\partial I}{\partial y}v^y + \frac{\partial I}{\partial t} \equiv 0 \tag{4.1}$$

where v^x and v^y are the spatial velocity components. The optical flow constraint equation is often rewritten in this form

$$\nabla I \cdot \vec{v} = -I_t \tag{4.2}$$

where $\nabla I = (I_x, I_y)^T$ is the spatial image gradient, $\vec{v} = (v^x, v^y)^T$ is the velocity and I_t the temporal partial derivative of the intensity. This equation is the basis for most gradient based optic flow methods.

Florack et al. [16, 17, 47, 44, 18] have stated the optic flow constraint equation in the context of linear Gaussian scale-space as the preservation of the Lie derivative of the scale-space image $L(\vec{x}; \sigma) : \mathbb{R}^{(2+1)} \times \mathbb{R} \mapsto \mathbb{R}$ along the optical flow field $\vec{v}(\vec{x}) : \mathbb{R}^{(2+1)} \mapsto \mathbb{R}^{(2+1)}$. This can be stated as

$$\mathcal{L}_{\vec{v}}L = L_xv^x + L_yv^y + L_tv^t \equiv 0.$$

If we again use the assumption that the temporal velocity component is constant, $v^t = 1$, we then reach at the constraint equation defined in Eq. (4.1),

$$\mathcal{L}_{\vec{v}}L = L_xv^x + L_yv^y + L_t \equiv 0. \tag{4.3}$$

4.1.3 The Aperture Problem

The optic flow constraint equation Eq. (4.1) does not uniquely define the optical flow field, because it in the 2-dimensional case is one equation of two unknown variables. This raises the so-called *aperture problem*¹.

¹Note that there is a difference between the scale-space aperture function and the aperture problem of optic flow.

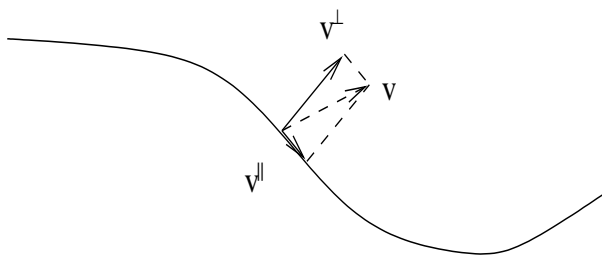


Figure 4.1: This is a schematic drawing of the velocity at some point on an isophote (symbolised by the fully drawn curve). It shows the normal v^\perp and tangential v^\parallel velocity components. The aperture problem prevents use from extracting the tangential velocity component, because no change can be seen along the isophote.

A more intuitive description of the problem is that from a measurement of the flow at a single point on a moving isophote, it is only possible to extract the velocity component perpendicular to the isophote. In other words it is not possible to locally extract the tangential movement along the isophote. The movement along the isophote cannot be extracted due to the fact that there is no visible change along the isophote. A schematic drawing of the problem can be viewed in Fig. 4.1. The velocity component perpendicular to the isophote constitutes the so-called normal velocity or *normal flow* at a given point on the isophote (see Fig. 4.1).

In order to estimate the optic flow we need to estimate both the normal and tangential components of the velocity. This can be done by complementing the optical flow constraint equation Eq. (4.2) with additional constraints.

Hildreth [23] lists different types of constraints that have been used in different optic flow methods. The constraints can be categorized into:

- Motion models
- Smoothness constraints

The most prominent motion model is the normal flow model, in which it is assumed that the optic flow consists of only normal flow. This assumption is enforced by introducing the so-called normal flow constraint, which will be discussed further in Sec. 4.3.

Hildreth [23] gives examples of motion models used in optic flow methods, in which either local constant velocity or rigid motion is assumed. The local constant velocity assumption constrains the motion to be purely translational. The rigid motion assumption constrains the types of valid motion to translation and rotation. That is, deformation of the moving objects are not allowed.

Florack et al. [16, 17, 47, 44, 18] have introduced a polynomial expansion of the velocity field as a model of the flow. In Sec. 4.3 I will describe this motion model and use this notion in an extension of the optic flow method proposed by Florack et al. [16, 17, 47, 44, 18].

Both Horn and Schunck [25, 24] and Hildreth [23] use a smoothness constraint in their algorithms. Horn and Schunck [25] look for the solution of the optic flow constraint equation Eq. (4.2), which ensures that the velocity field has minimal variation. Horn and Schunck construct the following minimisation problem using the Lagrange multiplier², α^2 , and solve the problem

²See for example [8] for a description of the method of Lagrange multipliers in constrained minimisation problems.

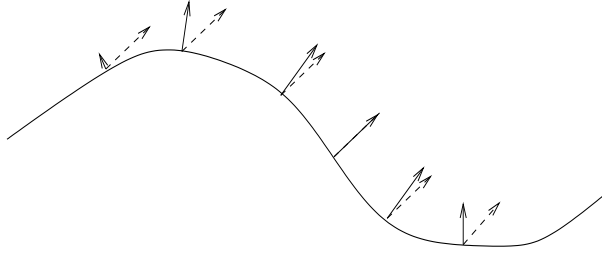


Figure 4.2: This schematic drawing shows the optic \vec{u} (dashed vectors) and the normal \vec{u}^\perp (fully drawn vectors) flow along an isophote. Hildreth [23] utilises the local integration window over a normal flow field to estimate the missing tangential component of the real optical flow field.

using variational methods [8, 56]:

$$E^2 = \int_{\vec{x} \in \mathbb{R}^N} (E_m^2 + \alpha^2 E_s^2) d\vec{x}$$

where

$$E_m = \nabla I \cdot \vec{u} + I_t,$$

and they use the following smoothness constraint E_s of the flow field \vec{u} ,

$$E_s^2 = \left(\frac{\partial u^x}{\partial x} \right)^2 + \left(\frac{\partial u^x}{\partial y} \right)^2 + \left(\frac{\partial u^y}{\partial x} \right)^2 + \left(\frac{\partial u^y}{\partial y} \right)^2.$$

Hildreth [23] states her smoothness constraint as the problem of finding the optical flow field \vec{u} under a local integration window, which varies smoothly. She utilises a local integration window over the normal flow field to estimate the missing tangential component of the real optical flow field. Hildreth assumes that the normal flow field \vec{u}^\perp is known and that it varies smoothly along isophotes, and that the orientation of the normal flow at two points on the isophote is different (see Fig. 4.2 for a schematic drawing of the optic and normal flow along an isophote). She seeks the vector field, which has a normal component corresponding to the known normal flow and which minimises

$$E^2 = \int \left| \frac{\partial \vec{u}(s)}{\partial s} \right|^2 ds$$

where $\vec{u}(s)$ is some parameterisation of the flow field along an isophote. This means that the variation of the flow should be minimised along the isophote.

4.2 Optical Flow Fields Extracted From One Image

In this section, I will look at an algorithm proposed by Maurizot et al. [41] for estimation of the optical flow field from a single 2-dimensional image of fluid flow. Maurizot et al. use this algorithm for the localisation and tracking of singular points in fluid flow. I have introduced the usage of two linear Gaussian scale-spaces into the algorithm and adapted it for my purposes. Furthermore, I have made an implementation of the modified algorithm as stated earlier. I include this algorithm in order to show an example of other algorithms used for the analysis of turbulent flow and as a curiosity, because of the fact that only a single image is in use.

Maurizot et al. assume that the motion of the underlying fluid flow is tangential to the isophotes of the image and therefore perpendicular to the image intensity gradient. This is only the case of image sequences, where the underlying motion flow field induces an intensity pattern, which is time-independent. That is, constant throughout the image sequence, like in the example of the spinning billiard ball from Sec. 4.1. This can in some situations be true for image sequences of turbulent flow, but is in general not a valid assumption.

Maurizot et al. use the assumption that the velocity $\vec{v}(\vec{x}) : \mathbb{R}^2 \mapsto \mathbb{R}^2$ at the point \vec{x} can be modelled by a linear motion model given by

$$\vec{v}(\vec{x}) = \mathbf{A}\vec{x} + \vec{b}$$

where $\vec{b} = (b_1, b_2)^T$ and

$$\mathbf{A} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix}.$$

They restate the optic flow constraint equation Eq. (4.2) according to the assumption that the flow is perpendicular to the image gradient,

$$\nabla I \cdot \vec{v} = \nabla I \cdot (\mathbf{A}\vec{x} + \vec{b}) = \mathbf{\Gamma}_{\vec{x}}^T \Theta = 0$$

where $\mathbf{\Gamma}_{\vec{x}} = (I_x, xI_x, yI_x, I_y, xI_y, yI_y)^T$ and $\Theta = (b_1, a_1, a_2, b_2, a_3, a_4)^T$ is the vector of model parameters.

Maurizot et al. estimate the model parameters Θ by a least squares minimisation under a local window function. The minimal parameters $\hat{\Theta}$ at the point \vec{x} are, under the assumption that $|\Theta| = 1$, therefore given by

$$\hat{\Theta} = \arg \min_{\Theta} \sum_{\vec{x} \in F} \Theta^T \mathbf{\Gamma}_{\vec{x}} \mathbf{\Gamma}_{\vec{x}}^T \Theta$$

where F is the set of all points in a local window of the image. The $\hat{\Theta}$ can be computed by finding the smallest eigenvalue and the corresponding eigenvector of the matrix

$$\mathbf{M}_F = \sum_{\vec{x} \in F} \mathbf{\Gamma}_{\vec{x}} \mathbf{\Gamma}_{\vec{x}}^T.$$

The unit eigenvector of \mathbf{M}_F corresponding to the smallest eigenvalue is the estimated model parameters $\hat{\Theta}$.

Maurizot et al. [41] use their algorithm for tracking singular points in flow fields and they have chosen to incorporate a local window resizing mechanism into their algorithm, in order to improve the accuracy of the found singular points. The window resizing mechanism changes the window size until at most one singular point is within the window. Furthermore, they add a, according to them, robust algorithm for estimation of the image gradients, which iteratively improves both the image gradient estimate and the motion model estimate, thereby reducing the model and measurement noise. In an earlier paper by Maurizot et al. [42] they propose a slightly different algorithm for localisation of singular points in fluid flow. In this algorithm they use Gaussian pyramids for the estimation of the local flow model, which is based on the idea of taking scale into account when estimating the optical flow field.

I choose to modify the algorithm proposed by Maurizot et al. [41] by computing the image gradients using spatial linear Gaussian scale-space derivatives. Furthermore, I choose to substitute the local unweighted integration window F with a Gauss weighted integration window,

by convolving the image gradients with the linear Gaussian scale-space aperture function. I thereby introduce an integration scale-space of the spatial image gradients. I ignore the resizing algorithm proposed by Maurizot et al. [41], because they only use it to improve their singularity tracking mechanism, which is not necessary in the framework of this thesis. A new resizing mechanism could be added by introduction of an automatic scale selection mechanism. In Sec. 4.3.1 I describe a possible scale selection method proposed by Niessen and Maas [47]. This method could be reformulated to match the algorithm by Maurizot et al. by using the Frobenius norm of \mathbf{M}_F as an uncertainty measure of the estimated flow and then selecting those scales for which the uncertainty measure is minimal.

Movement of the local integration window is not interesting when scale-space is introduced, because we compute the scale-space by convolution at every point in the image and therefore do not need to move our focus of attention. Furthermore, the reason that Maurizot et al. introduces this window movement mechanism is to save computations.

The modified algorithm can be stated as follows:

Algorithm 4.1 (Flow Field Estimation using a Single Image)

Under the assumption that the underlying real world flow field results in an optical flow field that can be modelled by the linear relation

$$\vec{v} = \mathbf{A}\vec{x} + \vec{b}$$

and which is tangential to the isophotes of the image, a non-temporal optic flow constraint equation can be constructed, which has the form

$$\nabla L \cdot \vec{v} = \nabla L \cdot (\mathbf{A}\vec{x} + \vec{b}) = \mathbf{\Gamma}_{\vec{x}}^T \Theta = 0,$$

where $\nabla L(\vec{x}; \sigma_s) = (L_x, L_y)^T$ is the scale-space image gradient at spatial scale σ_s ,

$$\mathbf{A} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix}$$

and $\vec{b} = (b_1, b_2)^T$ are the model parameters,

$$\mathbf{\Gamma}_{\vec{x}} = (L_x, xL_x, yL_x, L_y, xL_y, yL_y)^T,$$

and

$$\Theta = (b_1, a_1, a_2, b_2, a_3, a_4)^T.$$

The optical flow field can, under the assumption that $|\Theta| = 1$, be found by solving the following minimisation problem

$$\hat{\Theta} = \arg \min_{\Theta} \int_{\vec{x} \in \Omega} \Theta^T \mathbf{\Gamma}_{\vec{x}} \mathbf{\Gamma}_{\vec{x}}^T \Theta G(\vec{x}; \varpi) d\vec{x}$$

where ϖ is the spatial integration scale and $\Omega \subseteq \mathbb{R}^2$ is the set of image points. The solution is given by finding the unit eigenvector corresponding to the minimal eigenvalue of the matrix

$$\mathbf{M}_G(\vec{x}; \varpi) = (\mathbf{\Gamma}_{\vec{x}} \mathbf{\Gamma}_{\vec{x}}^T) *_{\vec{x}} G(\vec{x}; \varpi) = \tag{4.4}$$

$$\begin{pmatrix} \Lambda(x^2) & -\Lambda_x(x^2) & -\Lambda_y(x^2) & \Lambda(x,y) & -\Lambda_x(x,y) & -\Lambda_y(x,y) \\ -\Lambda_x(x^2) & \Lambda_{xx}(x^2) + \Lambda(x^2) & \Lambda_{xy}(x^2) & -\Lambda_x(x,y) & \Lambda_{xx}(x,y) + \Lambda(x,y) & \Lambda_{xy}(x,y) \\ -\Lambda_y(x^2) & \Lambda_{xy}(x^2) & \Lambda_{yy}(x^2) + \Lambda(x^2) & -\Lambda_y(x,y) & \Lambda_{xy}(x,y) & \Lambda_{yy}(x,y) + \Lambda(x,y) \\ \Lambda(x,y) & -\Lambda_x(x,y) & -\Lambda_y(x,y) & \Lambda(y^2) & -\Lambda_x(y^2) & -\Lambda_y(y^2) \\ -\Lambda_x(x,y) & \Lambda_{xx}(x,y) + \Lambda(x,y) & \Lambda_{xy}(x,y) & -\Lambda_x(y^2) & \Lambda_{xx}(y^2) + \Lambda(y^2) & \Lambda_{xy}(y^2) \\ -\Lambda_y(x,y) & \Lambda_{xy}(x,y) & \Lambda_{yy}(x,y) + \Lambda(x,y) & -\Lambda_y(y^2) & \Lambda_{xy}(y^2) & \Lambda_{yy}(y^2) + \Lambda(y^2) \end{pmatrix}$$

where $\Lambda^{(x^n, y^m)}(\vec{x}; \varpi) = (L_x^n L_y^m) *_{\vec{x}} G(\vec{x}; \varpi)$, $n, m = 0, 1, 2$, is a function of quadratic gradients at the local integration scale ϖ . It is assumed that the derivatives of $\Lambda^{(x^n, y^m)}(\vec{x}; \varpi)$ are all normalised using Definition 3.5 and that natural coordinates, $\frac{x}{\sigma}$, are used.

Proof: The construction of the matrix \mathbf{M}_G can be proved by noticing that

$$\begin{aligned} \mathbf{M}_G(\vec{x}; \varpi) &= (\mathbf{\Gamma}_{\vec{x}} \mathbf{\Gamma}_{\vec{x}}^T) *_{\vec{x}} G(\vec{x}; \varpi) \\ &= \begin{pmatrix} L_x^2 & xL_x^2 & yL_x^2 & L_xL_y & xL_xL_y & yL_xL_y \\ xL_x^2 & x^2L_x^2 & xyL_x^2 & xL_xL_y & x^2L_xL_y & xyL_xL_y \\ yL_x^2 & xyL_x^2 & y^2L_x^2 & yL_xL_y & xyL_xL_y & y^2L_xL_y \\ L_xL_y & xL_xL_y & yL_xL_y & L_y^2 & xL_y^2 & yL_y^2 \\ xL_xL_y & x^2L_xL_y & xyL_xL_y & xL_y^2 & x^2L_y^2 & xyL_y^2 \\ yL_xL_y & xyL_xL_y & y^2L_xL_y & yL_y^2 & xyL_y^2 & y^2L_y^2 \end{pmatrix} *_{\vec{x}} G(\vec{x}; \varpi). \end{aligned}$$

Each element of the matrix is then convolved with the Gauss function $G(\vec{x}; \varpi)$. We can then get to the partial derivatives of $\Lambda^{(x^n, y^m)}(\vec{x}; \varpi)$ of the matrix of Eq. (4.4) by use of Definition 3.4 and the monomials of Eq. (3.8) and their relation to the partial derivatives of the Gauss kernel.

□

Algorithm 4.1 utilises two scale-spaces; one scale for estimation of image gradients and another scale for solving the minimisation problem. I will use this trick again in Sec. 4.4, where I propose a novel algorithm for estimation of optic flow.

4.2.1 Implementation of Algorithm 4.1

The implementation of Algorithm 4.1 is fairly straightforward. The image gradients can be found by using some form of scale-space implementation and the same goes for the local integration scale-space. When finding the minimal eigenvalue of \mathbf{M}_G it can be used that Eq. (4.4) is symmetric³.

In my implementation of the algorithm I have used the scale-space implementation described in Sec. 3.4 and the eigenvalue finder available in Matlab. I fix the direction of the velocity field so it is always left hand oriented to the image gradient. I do this in order to remove the ambiguity coming from the fact that the flow direction is estimated up to a sign. The source code for my implementation of the algorithm can be found in Appendix D.2.1.

4.2.2 Performance of Algorithm 4.1

As stated earlier, I have included Algorithm 4.1, described above, to show an example of other algorithms used for the investigation of turbulence by optic flow methods. In order for the algorithm to work properly we have to assume that the flow is perpendicular to the image gradient. Under this assumption the algorithm enables use to attempt to estimate the optic flow from only a single image. Images of turbulence used with this algorithm should consist of whirl-like structures in order for the above stated assumption to be valid and for the algorithm to work properly. I will not evaluate the quality of the algorithm in lack of a suitable measure, but instead just show some examples of flow fields estimated using Algorithm 4.1, which can be found in Fig. 4.3. The method is fairly good at detecting whirl-like structures, as can be seen from this figure. But the overall validity of the flow field is rather questionable.

³See [22] for different methods of solving the symmetric eigenvalue problem

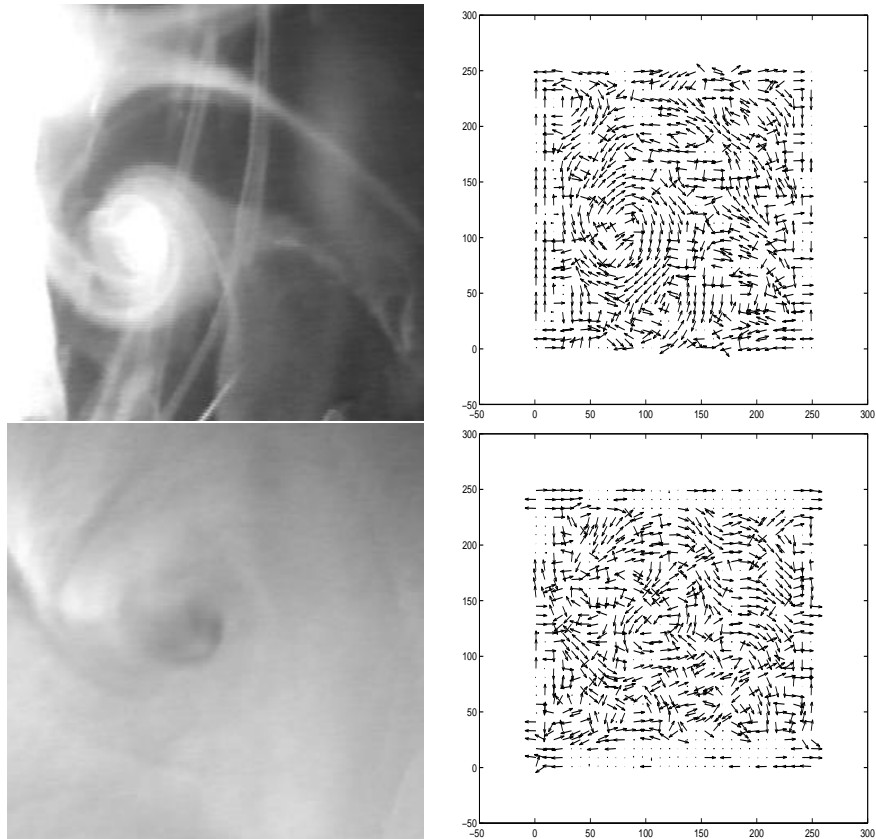


Figure 4.3: Examples of flow fields obtained by use of the single image algorithm, Algorithm 4.1. The images used are from the image sequences of turbulent flow described in Sec. 7.2. The middle images of the E3/serie1 and E1/serie2 sequences are used. The flow were computed for the following scales ($\sigma_s = 1.4$, $\varpi = 8.0$).

4.3 Normal Flow Field Estimation Using Scale-Space

In this section I will describe a method for normal flow and optic flow estimation, which takes the temporal changes into account. The method, which was proposed by Florack et al. [16, 17, 47, 44, 18], utilises a linear Gaussian scale-space and restates Horn and Schunck's [25] optic flow constraint equation under the scale-space aperture function.

Florack et al. [16, 17, 47, 44, 18] have investigated the problem of optic flow field estimation in the context of linear Gaussian scale-space theory. Florack et al. [17, 18] introduces the concept of measurement duality in conjunction with linear Gaussian scale-space. They propose that a duality exists between the real world scenes and the measuring devices or filters used for capturing the scenes. This means that we have to be aware of the relation between the actual scene and the measuring device. In the context of linear Gaussian scale-space this duality is recreated by the use of the scale-space aperture function, which can be seen as a measuring device. The raw image at scale zero can be seen as the original scene, which is captured by the scale-space aperture function. This idea is used in their derivation of an optic flow method using linear Gaussian scale-space.

The duality principle described above is very similar to the measurement duality of quantum mechanics, which states that we cannot conduct a measurement in the quantum mechanical

domain, without disturbing the system on which we are conducting the measurement. We therefore have to see the measurement as a part of the quantum mechanical system when making inference about the system. For a discussion about measurement duality and uncertainty in quantum mechanics see Bohr [9] or an introductory text book on quantum mechanics like Lévy-Leblond and Balibar [38].

Florack et al. [16, 17, 44, 18] state the optic flow constraint equation of Eq. (4.1) under the spatiotemporal scale-space aperture function $G(\vec{x}; \sigma_s, \sigma_t) : \mathbb{R}^{N+1} \times \mathbb{R}_+^2 \setminus \{0\} \mapsto \mathbb{R}$,

$$\int_{\vec{x} \in \Omega} (I_t + v^i I_i) G d\vec{x} \equiv 0,$$

where $I(\vec{x}(t), t) : \Omega \mapsto \mathbb{R}$ is the intensity function and $\Omega \in \mathbb{R}^{N+1}$ is the set of spatiotemporal points in the image sequence.

Florack et al. show that by partial integration and using the Hermitian structure of Gaussian derivatives defined in Definition 3.4 the optic flow constraint equation can be restated in terms of derivatives of the Gaussian aperture function $G(\vec{x}, t; \sigma_s, \sigma_t)$. We hereby get the aperture optic flow constraint equation,

$$\int_{\vec{x} \in \Omega} (G_t + v_i^i G + v^i G_i) I d\vec{x} = 0. \quad (4.5)$$

Florack et al. [16, 17, 18] introduce a model of the flow field at a spatiotemporal point $\vec{x} = (t, x^i)^T$, $i = 1, \dots, N$, by polynomial approximation of the field up to an order M . The model can be defined as:

Definition 4.3 (Mth Order Formal Expansion)

The formal expansion of order M of the vector field $v^i(\vec{x})$ at $\vec{x} = \vec{0}$, denoted $v_M^i(\vec{x})$, is the M th order polynomial

$$v_M^i(\vec{x}) = \tilde{v}^i + \tilde{v}_\alpha^i x^\alpha + \frac{1}{2!} \tilde{v}_{\alpha\beta}^i x^\alpha x^\beta + \dots + \frac{1}{M!} \underbrace{\tilde{v}_{\alpha\beta\dots\nu}^i}_M x^\alpha x^\beta \dots x^\nu$$

where x^α denotes the components of the spatiotemporal vector \vec{x} and \tilde{v}^i is the flow field coefficients approximating the partial derivatives of $v^i(\vec{x} = \vec{0})$ at the origo.

Using the M th order formal expansion of Definition 4.3, Florack et al. [16, 17, 18] propose to write the aperture optic flow constraint equation Eq. (4.5) in terms of an expansion of the flow vector field to the M th order.

This means that if we, in the $(2 + 1)$ -dimensional spatiotemporal case, choose to use a zeroth order approximation of the flow field, $v_0^i = \tilde{v}^i$, and use the assumption that $v^t = 1$, the aperture optic flow constraint equation, by substitution of v_0^i into Eq. (4.5) and using the fact that $\tilde{v}_i^i = 0$ because the model is of the zeroth order, can be restated as the classical formulation of the optic flow constraint equation by Horn and Schunck [25]

$$L_t + \tilde{v}^x L_x + \tilde{v}^y L_y = 0, \quad (4.6)$$

where $L(\vec{x}; \sigma_s, \sigma_t) : \mathbb{R}^N \times \mathbb{R}_+^2 \mapsto \mathbb{R}$ is the spatiotemporal scale-space image at spatial scale σ_s and temporal scale σ_t . The model coefficients \tilde{v}^x and \tilde{v}^y are the unknown flow components.

If we use higher order models, we get more complex optic flow constraint equations. The first order model of the $(2 + 1)$ -dimensional spatiotemporal flow is given by,

$$\begin{aligned} v_1^x &= \tilde{v}^x + x\tilde{v}_x^x + y\tilde{v}_y^x + t\tilde{v}_t^x \\ v_1^y &= \tilde{v}^y + x\tilde{v}_x^y + y\tilde{v}_y^y + t\tilde{v}_t^y. \end{aligned} \quad (4.7)$$

In the rest of this thesis I will use the shorthand *first order model* to denote the first order 2-dimensional spatiotemporal model of Eq. (4.7).

By substituting v_1^i into Eq. (4.5) and using Definition 3.4 it can be shown by partial integration that we can write the aperture optic flow constraint equation as,

$$L_t + \tilde{v}^x L_x + \tilde{v}^y L_y + \tilde{v}_t^x L_{xt} + \tilde{v}_t^y L_{yt} + \tilde{v}_x^x L_{xx} + (\tilde{v}_y^x + \tilde{v}_x^y) L_{xy} + \tilde{v}_y^y L_{yy} = 0. \quad (4.8)$$

As stated earlier in Sec. 4.1.3, additional constraints, besides the aperture optic flow constraint equation Eq. (4.5), are needed to estimate either the normal flow or the full optic flow. Florack et al. [16, 17, 44, 18] call these constraints for gauge conditions. In the rest of this section I will only be concerned with the estimation of normal flow fields, and will for that reason explain how to introduce a normal flow constraint or gauge condition into the formalism of Florack et al. [16, 17, 47, 44, 18].

The classical normal flow constraint can in terms of scale-space images be stated as the following constraint equation,

$$\nabla L \cdot \hat{v} = v^x L_y - v^y L_x \equiv 0 \quad (4.9)$$

where $\hat{v} = (-v^y, v^x)^T$. By substitution of the zeroth order flow model, $v_0^i = \tilde{v}^i$, into Eq. (4.9) we get the zeroth order normal flow constraint equation,

$$\tilde{v}^x L_y - \tilde{v}^y L_x = 0. \quad (4.10)$$

We can now formulate an algorithm for estimation of the normal flow using a zeroth order model by combining Eq. (4.6) and Eq. (4.10):

Algorithm 4.2 (Scale-Space Zeroth Order Normal Flow Estimation)

The zeroth order normal flow field of the scale-space image $L(\vec{x}; \sigma_s, \sigma_t)$ at the spatiotemporal scale $\vec{\sigma} = (\sigma_s, \sigma_t)$ constrained by Eq. (4.6) and Eq. (4.10) can be calculated by solving the following system of linear equations

$$\mathbf{A}_{\vec{\sigma}} \tilde{v} = \vec{b}$$

where $\tilde{v} = (\tilde{v}^x, \tilde{v}^y)^T$ is the vector of estimated flow components,

$$\mathbf{A}_{\vec{\sigma}} = \begin{pmatrix} L_x & L_y \\ L_y & -L_x \end{pmatrix}, \quad (4.11)$$

and $\vec{b} = (-L_t, 0)^T$. All scale-space derivatives should be normalised using Definition 3.5 and natural coordinates, $\frac{x}{\sigma}$.

Florack et al. [18] formulate a normal flow constraint for the first order spatiotemporal flow model given by Eq. (4.7) as

$$0 = \tilde{v}^x L_y + \tilde{v}_t^x L_{yt} + \tilde{v}_x^x L_{xy} + \tilde{v}_y^x L_{yy} - \tilde{v}^y L_x - \tilde{v}_t^y L_{xt} - \tilde{v}_x^y L_{xx} - \tilde{v}_y^y L_{xy}. \quad (4.12)$$

This equation can be proved by substituting Eq. (4.7) into Eq. (4.9) and using Definition 3.4.

The first order spatiotemporal model give us Eq. (4.8) and Eq. (4.12), which are two equations of six unknown variables. In order to solve this problem and compute the first order normal flow, Florack et al. [18] find the partial derivatives with respect to x , y , and t of the two equations Eq. (4.8) and Eq. (4.12), thereby getting the algorithm described below. It is important that all derivatives are dimensionless because of the Hermitian structure of Gaussian derivatives stated in Definition 3.4 and therefore all derivatives should be normalised as defined in Definition 3.5.

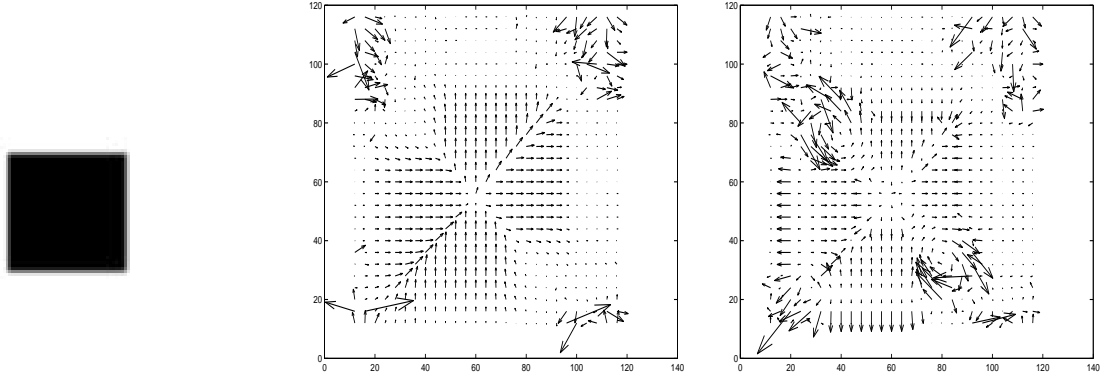


Figure 4.4: Examples of zeroth and first order normal flow of a synthetic image sequence of a moving square, estimated using Algorithm 4.2 and Algorithm 4.3. On the left is shown the middle image frame of the image sequence, in the middle the obtained zeroth order flow at this frame, and on the right the corresponding first order flow. The correct flow direction is 45° and 1.8 pixels per frame. The scales used are $\sigma_s = 2.0$ and $\sigma_t = 2.0$. Only every fourth flow vector is shown and they are scaled by a factor of four.

Algorithm 4.3 (Scale-Space First Order Normal Flow Estimation)

The first order normal flow field of the scale-space image $L(\vec{x}; \sigma_s, \sigma_t)$ at the spatiotemporal scale $\vec{\sigma} = (\sigma_s, \sigma_t)$ constrained by Eq. (4.8) and Eq. (4.12) can be calculated by solving the following system of linear equations

$$\mathbf{A}_{\vec{\sigma}} \tilde{\mathbf{v}} = \vec{\mathbf{b}}$$

where $\tilde{\mathbf{v}} = (\tilde{v}^x, \tilde{v}^y, \tilde{v}_t^x, \tilde{v}_t^y, \tilde{v}_x^x, \tilde{v}_x^y, \tilde{v}_y^x, \tilde{v}_y^y)^T$ is the vector of estimated flow components and its derivatives,

$$\mathbf{A}_{\vec{\sigma}} = \begin{pmatrix} L_x & L_y & L_{xt} & L_{yt} & L_{xx} & L_{xy} & L_{xy} & L_{yy} \\ L_{xt} & L_{yt} & L_x + L_{xtt} & L_y + L_{ytt} & L_{xxt} & L_{xyt} & L_{xyt} & L_{yyt} \\ L_{xx} & L_{xy} & L_{xxt} & L_{xyt} & L_x + L_{xxx} & L_y + L_{xxy} & L_{xxy} & L_{xyy} \\ L_{xy} & L_{yy} & L_{xyt} & L_{yyt} & L_{xxy} & L_{xyy} & L_x + L_{xyy} & L_y + L_{yyy} \\ L_y & -L_x & L_{yt} & -L_{xt} & L_{xy} & -L_{xx} & L_{yy} & -L_{xy} \\ L_{yt} & -L_{xt} & L_y + L_{ytt} & -L_x - L_{xtt} & L_{xyt} & -L_{xxt} & L_{yyt} & -L_{xyt} \\ L_{xy} & -L_{xx} & L_{xyt} & -L_{xxt} & L_y + L_{xxy} & -L_x - L_{xxx} & L_{xyy} & -L_{xxy} \\ L_{yy} & -L_{xy} & L_{yyt} & -L_{xyt} & L_{xyy} & -L_{xxy} & L_y + L_{yyy} & -L_x - L_{xyy} \end{pmatrix}, \quad (4.13)$$

and $\vec{\mathbf{b}} = (-L_t, -L_{tt}, -L_{xt}, -L_{yt}, 0, 0, 0, 0)^T$. All scale-space derivatives should be normalised using Definition 3.5 and natural coordinates, $\frac{x}{\sigma}$.

Examples of zeroth and first order normal flow fields of a synthetic image sequence obtained by using Algorithm 4.2 and Algorithm 4.3 can be found in Fig. 4.4. In Chapter 5 I will evaluate the performance of this algorithm and compare it to other normal flow methods.

Implementation of Algorithm 4.2 and Algorithm 4.3

The two algorithms can be implemented using some kind of linear Gaussian scale-space implementation. I have implemented the two algorithms using the scale-space implementation described in Sec. 3.4.

The linear system of equations of each of the two algorithms can be solved using for instance LU decomposition⁴. In my implementation of the two algorithms I use the linear equation solver of Matlab. The implementation of these algorithms can be found in Appendix D.2.1.

4.3.1 Scale Selection and Normal Flow Uncertainty Measurement

Niessen and Maas [47] and Nielsen et al. [43] propose a measure of the uncertainty of the estimated flow field. This uncertainty measure is used as a scale selection mechanism, which selects scales on the basis of the uncertainties in the measurements of the image gradients. The reason for introducing a scale selection mechanism is that we are interested in finding the scale for which the flow estimation is optimal at a certain image point. They propose to minimise the Frobenius norm of $\mathbf{A}_{\vec{\sigma}}^{-1}$ from Eq. (4.11) or Eq. (4.13), depending on the chosen model, in the spatiotemporal scale-space $\vec{\sigma} = (\sigma_s, \sigma_t)$ at each pixel in the image. The Frobenius norm is defined as:

Definition 4.4 (The Frobenius Norm)

The Frobenius norm of a matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$ is given by, [22],

$$\|\mathbf{B}\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |b_{ij}|^2},$$

where b_{ij} is the (i, j) 'th element in \mathbf{B} .

In order to get a good estimate of the flow field it is necessary to ensure that $\mathbf{A}_{\vec{\sigma}}$ is well-conditioned. This can be done by minimisation of the Frobenius norm of $\mathbf{A}_{\vec{\sigma}}^{-1}$, because of the relation between the Frobenius norm and the singular values $\lambda_i(\mathbf{A}_{\vec{\sigma}}^{-1})$ of this matrix

$$\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2 = \sum_i \lambda_i^2(\mathbf{A}_{\vec{\sigma}}^{-1}) = \sum_i \frac{1}{\lambda_i^2(\mathbf{A}_{\vec{\sigma}})}. \quad (4.14)$$

That is, the Frobenius norm of a matrix is the sum of the singular values of that matrix. From Eq. (4.14) we get that $\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ should be small in order for $\mathbf{A}_{\vec{\sigma}}$ to be well-conditioned, because $\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ is large when the singular values of $\mathbf{A}_{\vec{\sigma}}$ is small and when the singular values is large $\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ becomes small. Examples of the Frobenius norm of $\mathbf{A}_{\vec{\sigma}}^{-1}$ of image sequences can be found in Fig. 4.5.

An automatic scale selection mechanism can be stated as the selection, at a given point in space-time, of the spatiotemporal scale $\vec{\sigma}$, for which $\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ is minimal.

We can also use $\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ as a measure of the quality of the estimated flow at a given point in space-time and then disregard bad measurements. I will use this principle in Sec. 4.4 to obtain an optic flow method, which extracts the optic flow from a normal flow and incorporates the flow uncertainty of the normal flow.

4.4 Optical Flow Field Estimation Using an Integration Scale-Space

In this section I will extend the ideas of Florack et al. [16, 17, 47, 44, 18], described in the previous Sec. 4.3, by introducing an integration scale-space of the normal flow. By using this

⁴See either Numerical Recipes [53] or Golub and van Loan [22] for an explanation of this method.

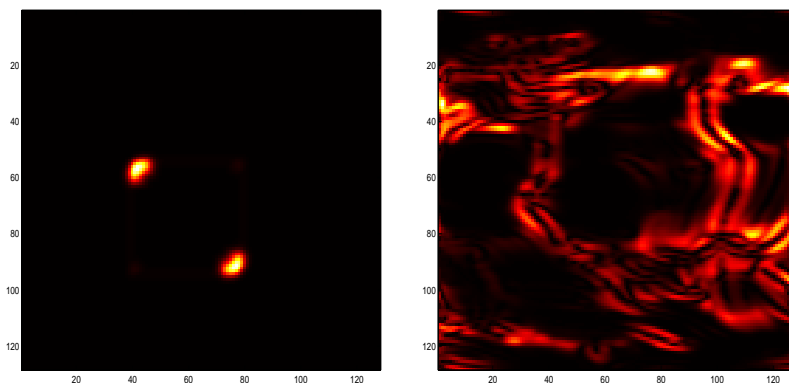


Figure 4.5: This figure shows $1/\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$ of the first order normal flow of the square image sequence of Fig. 4.4 on the left and the translating trees sequence (described further in Chapter 5) on the right. Light pixels indicates high values of $1/\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2$. The scales used are $\sigma_s = 2.0$ and $\sigma_t = 2.0$.

integration scale-space I intend to model the tangential component of the optical flow field. The use of an integration scale-space is inspired by Horn and Schunck [25, 24] and Hildreth [23], who use integration windows for estimation of optic flow.

I first introduce the integration scale-space algorithm in Sec. 4.4.1. In Sec. 4.4.2 I state the complete algorithm for estimation of the optical flow field using an integration scale-space. In Sec. 4.4.3 I will discuss different measures of the quality of the estimated optic flow and I propose a scale selection mechanism for the optic flow.

4.4.1 Integration Scale-Space

In order to model the tangential flow I need to introduce a model of the optic flow. I use the assumption that the optic flow around a point \vec{x}_0 can be modelled by the M th order expansion of Definition 4.3.

I propose that by using variational methods [56, 8], it is possible to estimate the model parameters by minimising an energy functional $E(\vec{v})$ of the normal flow field \vec{u} and the sought optical flow field \vec{v} under a local window function,

$$\vec{v} = \arg \min_{\vec{v}} E(\vec{v}). \quad (4.15)$$

I choose to introduce an integration scale-space and use the aperture function of linear Gaussian scale-space as the local integration window function. A possible choice of the energy functional $E(\vec{v})$, which describes the difference between the normal flow \vec{u} and the optic flow \vec{v} is

$$E(\vec{v}) = \int_{\vec{x} \in \mathbb{R}^2} w(\vec{x}) \|\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon} - \vec{u}\|_2^2 G(\vec{x}_0 - \vec{x}; \varpi) d\vec{x} \quad (4.16)$$

where

$$\vec{\Upsilon}(\vec{x}) = \frac{\vec{u}}{\|\vec{u}\|_2}$$

is the normalised direction of the normal flow field \vec{u} , \vec{v} is the sought optic flow, and $G(\vec{x}; \varpi) : \mathbb{R}^N \times \mathbb{R}_+ \setminus \{0\} \mapsto \mathbb{R}$ is the Gaussian aperture function at scale ϖ .

In order to handle singularities of the underlying normal flow field, we need a measure of uncertainties in the estimated normal flow. As mentioned in Sec. 4.3.1 Niessen and Maas [47]

and Nielsen et al. [43] suggest the usage of the Frobenius norm of the system of linear equations Eq. (4.11) or Eq. (4.13) as a measure of quality of the normal flow estimated by the method described in Sec. 4.3. Using the argumentation of Niessen and Maas [47] and Nielsen et al. [43] I choose to use the following weighting function in the energy functional $E(\vec{v})$ of Eq. (4.16),

$$w(\vec{x}) = \frac{1}{\|\mathbf{A}_{\vec{\sigma}}^{-1}\|_F^2}.$$

By introducing the weight function $w(\vec{x})$ into the energy functional $E(\vec{v})$ we suppress contributions coming from numerically uncertain normal flow estimates.

When we minimise the energy functional $E(\vec{v})$ of Eq. (4.16), we seek the optic flow \vec{v} , which has a normal component \vec{v}^\perp that is as close as possible to the normal flow \vec{u} . In the ideal case we have that $\vec{v}^\perp = \vec{u}$, but in a realistic image sequence this will in general not be true. The integration scale-space allows us to collect and weight information about the correspondence between the optic flow \vec{v} and the normal flow \vec{u} in a neighbourhood with a size proportional to the integration scale ϖ . Note that in the case where we use a zeroth order model for the normal flow \vec{u} , $\vec{\Upsilon}$ reduces to be the unit gradient direction.

A solution \vec{v} to the minimisation problem of Eq. (4.15) using the energy functional $E(\vec{v})$ of Eq. (4.16) can be found by solving the following equation

$$\frac{\partial E}{\partial v^i} = 0.$$

That is, we need to find zero crossings of the partial derivatives of $E(\vec{v})$ with respect to the components v^i of \vec{v} .

We could come up with other energy functionals than $E(\vec{v})$ proposed in Eq. (4.16), for instance we could add the constraint that the first order derivatives of the flow should be smooth, which would lead to the following energy functional $E'(\vec{v})$,

$$E'(\vec{v}) = \int_{\vec{x} \in \mathbb{R}^2} \left(\|\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon} - \vec{u}\|_2^2 + \left\| \frac{\partial(\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon})}{\partial x^i} - \frac{\partial \vec{u}}{\partial x^i} \right\|_2^2 \right) G(\vec{x}_0 - \vec{x}; \varpi) d\vec{x}. \quad (4.17)$$

A necessary condition that a solution to the minimisation problem of Eq. (4.15) using the energy functional $E'(\vec{v})$ of Eq. (4.17) exists, is that the Euler-Lagrange differential equation, [8], is satisfied

$$\frac{\partial}{\partial v^i} \|\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon} - \vec{u}\|_2^2 - \frac{\partial}{\partial x^i} \left\| \frac{\partial(\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon})}{\partial x^i} - \frac{\partial \vec{u}}{\partial x^i} \right\|_2^2 = 0.$$

A solution to the minimisation problem of Eq. (4.15) using the energy functional $E'(\vec{v})$ of Eq. (4.17) requires the computation of second order partial derivatives of the flow, which demands the use of a second or higher order model of the flow. This comes from the fact that $E'(\vec{v})$ is a function of first order derivatives of \vec{v} and the solution \vec{v} can be found by solving

$$\frac{\partial E'}{\partial v^i} = 0.$$

In this thesis I will only analyse the zeroth and first order flow models, which limit my analysis to the energy functional of Eq. (4.16).

In the following theorems I will give solutions to the minimisation problem of Eq. (4.15) using zeroth and first order models and the energy functional of Eq. (4.16).

Theorem 4.1 (Zeroth Order Model)

Assuming that a normal flow \vec{u} is given and that the optical flow field can be modelled by a zeroth order model, $v^i = \tilde{v}^i$, the solution of the minimisation problem of Eq. (4.15) using the energy functional $E(\vec{v})$ of Eq. (4.16) can be stated as the solution of the following linear system of equations

$$\mathbf{\Lambda}_0 \vec{v}_0 = \vec{\Omega}_0 \quad (4.18)$$

Here $\vec{v}_0 = (\tilde{v}^x, \tilde{v}^y)^T$ is the two unknown local optical flow field model parameters and

$$\mathbf{\Lambda}_0 = \begin{pmatrix} \Lambda^{x,x} & \Lambda^{x,y} \\ \Lambda^{x,y} & \Lambda^{y,y} \end{pmatrix}$$

and

$$\vec{\Omega}_0 = \begin{pmatrix} \Omega^x \\ \Omega^y \end{pmatrix}.$$

Where

$$\Lambda^{i,j}(\vec{x}; \sigma_s, \sigma_t, \varpi) \equiv (w(\vec{x}) \Upsilon^i(\vec{x}; \sigma_s, \sigma_t) \Upsilon^j(\vec{x}; \sigma_s, \sigma_t)) *_{\vec{x}} G(\vec{x}; \varpi) \quad (4.19)$$

and

$$\Omega^k(\vec{x}; \sigma_s, \sigma_t, \varpi) \equiv (w(\vec{x}) u^k(\vec{x}; \sigma_s, \sigma_t)) *_{\vec{x}} G(\vec{x}; \varpi) \quad (4.20)$$

are a scale-space functions of the normalised normal flow components, $\Upsilon^i = u^i / \|\vec{u}\|_2$, the normal flow $\vec{u} = (u^x, u^y)^T$, and the weighting function $w(\vec{x})$. The index symbols can take on the following values $i, j, k \in \{x, y\}$.

Proof: The proof of this theorem, which is quite lengthy, can be found in Appendix B.1. \square

Theorem 4.2 (First Order Model)

Assume that the normal flow \vec{u} is known and that the optical flow field can be modelled by a first order model,

$$v^x = \tilde{v}^x + x \tilde{v}_x^x + y \tilde{v}_y^x \quad (4.21)$$

$$v^y = \tilde{v}^y + x \tilde{v}_x^y + y \tilde{v}_y^y \quad (4.22)$$

where \tilde{v}_j^i are the model parameters. Then the solution to the minimisation problem of Eq. (4.15) using the energy functional $E(\vec{v})$ of Eq. (4.16) can be stated as the solution of the following linear system of equations

$$\mathbf{\Lambda}_1 \vec{v}_1 = \vec{\Omega}_1 \quad (4.23)$$

where $\vec{v}_1 = (\tilde{v}^x, \tilde{v}^y, \tilde{v}_x^x, \tilde{v}_x^y, \tilde{v}_y^x, \tilde{v}_y^y)^T$ is a vector of the six unknown flow field model parameters and

$$\mathbf{\Lambda}_1 = \begin{pmatrix} \Lambda^{x,x} & \Lambda^{x,y} & -\Lambda_x^{x,x} & -\Lambda_x^{x,y} & -\Lambda_y^{x,x} & -\Lambda_y^{x,y} \\ \Lambda^{x,y} & \Lambda^{y,y} & -\Lambda_x^{x,y} & -\Lambda_x^{y,y} & -\Lambda_y^{x,y} & -\Lambda_y^{y,y} \\ -\Lambda_x^{x,x} & -\Lambda_x^{x,y} & \Lambda_{xx}^{x,x} + \Lambda^{x,x} & \Lambda_{xx}^{x,y} + \Lambda^{x,y} & \Lambda_{xy}^{x,x} & \Lambda_{xy}^{x,y} \\ -\Lambda_x^{x,y} & -\Lambda_x^{y,y} & \Lambda_{xx}^{x,y} + \Lambda^{x,y} & \Lambda_{xx}^{y,y} + \Lambda^{y,y} & \Lambda_{xy}^{x,y} & \Lambda_{xy}^{y,y} \\ -\Lambda_y^{x,x} & -\Lambda_y^{x,y} & \Lambda_{xy}^{x,x} & \Lambda_{xy}^{x,y} & \Lambda_{yy}^{x,x} + \Lambda^{x,x} & \Lambda_{yy}^{x,y} + \Lambda^{x,y} \\ -\Lambda_y^{x,y} & -\Lambda_y^{y,y} & \Lambda_{xy}^{x,y} & \Lambda_{xy}^{y,y} & \Lambda_{yy}^{x,y} + \Lambda^{x,y} & \Lambda_{yy}^{y,y} + \Lambda^{y,y} \end{pmatrix} \quad (4.24)$$

and

$$\vec{\Omega}_1 = \begin{pmatrix} \Omega^x \\ \Omega^y \\ -\Omega_x^x \\ -\Omega_x^y \\ -\Omega_y^x \\ -\Omega_y^y \end{pmatrix}. \quad (4.25)$$

Here we again use the abbreviations from Eq. (4.19) and Eq. (4.19). We use natural coordinates, $x = \frac{\xi}{\sigma}$, and all derivatives of $\Lambda^{i,j}(\vec{x}; \vec{\sigma}, \varpi)$ and $\Omega^k(\vec{x}; \vec{\sigma}, \varpi)$ should be normalised using Definition 3.5.

Proof: The proof of this theorem, which is quite lengthy, can be found in Appendix B.2. \square

The determinant of $\mathbf{\Lambda}_0$ from Eq. (4.18) is almost always nonzero

$$\det |\mathbf{\Lambda}_0| = \Lambda^{x,x} \Lambda^{y,y} - \Lambda^{x,y} \Lambda^{x,y} \neq 0$$

and is therefore almost always nonsingular. This is also valid for the matrix $\mathbf{\Lambda}_1$ from Eq. (4.23).

The method proposed in Theorem 4.1 and Theorem 4.2 is not bound to the normal flow method proposed by Florack et al. [16, 17, 47, 44, 18] (Algorithm 4.2 and Algorithm 4.3) and in general the normal flow field used in Theorem 4.1 and Theorem 4.2 can be obtained by other methods, but in this thesis I will use the Florack et al. method.

In the two theorems above I have used the normal flow field and its derivatives as the basis of the estimation of the optical flow field. Furthermore, when using Algorithm 4.2 or Algorithm 4.3 the two theorems use two different scale-spaces; a scale-space for the computation of the image derivatives used for the estimation of the normal flow field and an integration scale-space of the normal flow field used for the computation of the optical flow field. The process sequence of Algorithm 4.2 and Algorithm 4.3 and the two theorems can be seen in Fig. 4.6.

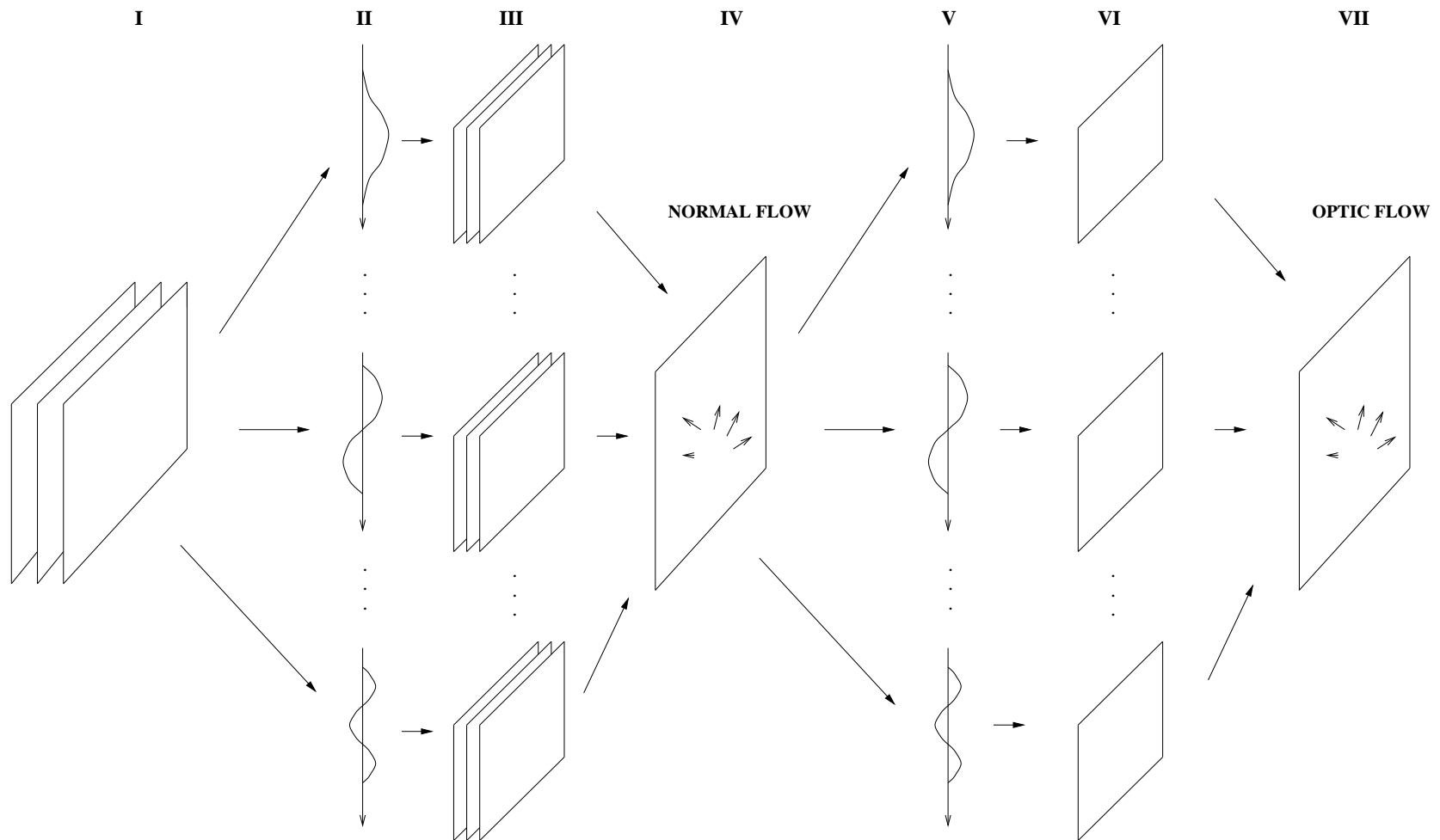


Figure 4.6: This drawing shows the structure of the optical flow estimation process using Algorithm 4.2 or Algorithm 4.3 and Theorem 4.1 or Theorem 4.2. Part I of the drawing shows the raw image sequence. Part II shows a schematic drawing of the scale-space filters used on the image sequence and part III is the spatiotemporal scale-spaces resulting from the filtering. Part IV shows the normal flow estimated for one image in the time sequence of the spatiotemporal scale-spaces of part III using either Algorithm 4.2 or Algorithm 4.3. The normal flow from part IV is then passed to part V, which is the integration scale-space filtering of functions of the normal flow. This filtering results in the spatial integration scale-spaces of part VI. Part VII is the optical flow estimated via Theorem 4.1 or Theorem 4.2 using the scale-spaces of part VI.

4.4.2 The Complete Optic Flow Algorithm

I can now state a complete algorithm for optical flow field estimation using an integration scale-space over a normal flow field and the corresponding uncertainty of the normal flow. The normal flow can be obtained using either Algorithm 4.2 or Algorithm 4.3, but other normal flow methods delivering an uncertainty measure of the flow could be used:

Algorithm 4.4 (Optic Flow Estimation Using an Integration Scale-Space)

Assume that a normal flow field \vec{u} of some order and a corresponding measure of uncertainty $w(\vec{x})$ in this normal flow is known. A zeroth or first order optical flow field can then be obtained by using either Theorem 4.1 or Theorem 4.2.

An example of using Algorithm 4.4 on a synthetic image sequence can be found in Fig. 4.7.

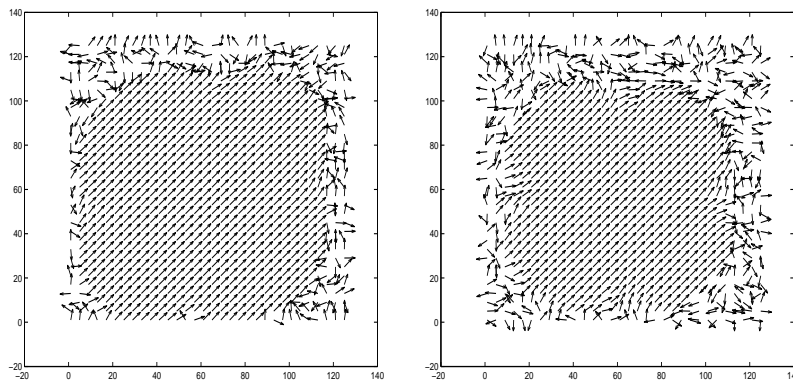


Figure 4.7: Examples of zeroth (left) and first (right) order optical flow fields obtained by using Algorithm 4.4. The analysed image sequence is the square sequence introduced in Fig. 4.4. Both flow fields were estimated using the zeroth order normal flow obtained by using Algorithm 4.2, and both flow fields were normalised to unit length. Only vectors at every fourth point are shown. The scales used are $\sigma_s = 2.0$, $\sigma_t = 2.0$, and $\varpi = 4.0$.

Implementation of Algorithm 4.4

In order to implement this algorithm we have to develop a 2-dimensional spatial scale-space representation, which should be used for the scale-space of the normal flow field. The system of linear equations of the algorithm can be solved by using for instance LU decomposition.

In my implementation of this algorithm I choose to use the method of scale-space implementation described in Sec. 3.4. Because my implementation of Algorithm 4.4 has been done in Matlab I have, for this algorithm, used a Matlab scale-space implementation written by Jon Sparring at DIKU, which uses the method described in Sec. 3.4. The source code for my implementation of Algorithm 4.4 can be found in Appendix D.2.1.

4.4.3 Scale Selection and Uncertainty Measurement in Optic Flow

As previously noted, structure of image sequences exists on different scales. We do not take this fact into account by using only a single scale in Algorithm 4.4. We can handle this fact by introducing a scale selection mechanism like the one discussed in Sec. 4.3.1.

As stated in Sec. 4.3.1, Niessen and Maas [47] and Nielsen et al. [43] propose to use the Frobenius norm of the matrix from the system of linear equations, used for the estimation of the

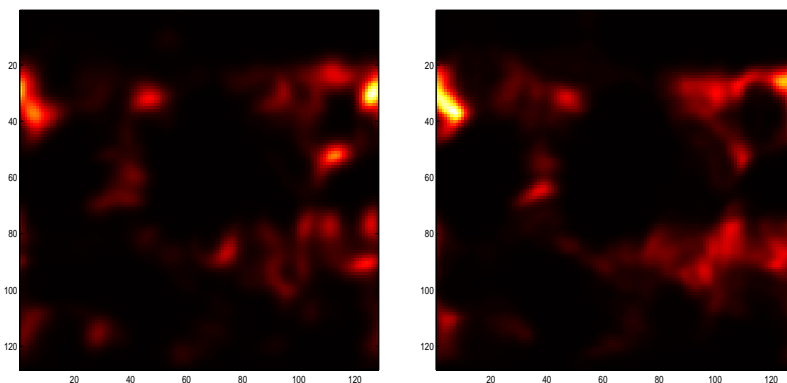


Figure 4.8: The inverse uncertainty measure for the translating trees sequence (described further in Chapter 5). The figure show $1/U_0$ (left) and $1/U_1$ (right) for the zeroth and first order optic flow computed from zeroth order normal flow using the scales $\sigma_s = 2.0$, $\sigma_t = 2.0$, and $\varpi = 4.0$. Light pixels indicates certain flow estimates and dark indicates uncertain flow estimates.

flow, as a measure of the uncertainty of the estimated flow. They use this uncertainty measure as a scale selection mechanism in conjunction with flow field estimation.

Inspired by this idea I choose to use the Frobenius norm of the inverse of either of the matrices Λ_0 or Λ_1 of Eq. (4.18) and Eq. (4.23) as a measure of the error of the optical flow field estimated using Algorithm 4.4. That is, I will use the following uncertainty measure for respectively the zeroth and first order optic flow,

$$U_0(\vec{x}; \sigma_s, \sigma_t, \varpi) = \|\Lambda_0^{-1}\|_F^2$$

and

$$U_1(\vec{x}; \sigma_s, \sigma_t, \varpi) = \|\Lambda_1^{-1}\|_F^2.$$

As stated in Sec. 4.3.1, this measure accounts for singularities in Λ_0^{-1} and Λ_1^{-1} . In Fig. 4.8 an example of this uncertainty measure for a synthetic image sequence can be found.

Using this uncertainty measure it is possible to implement a scale-selection mechanism, by choosing the scale at each pixel for which, depending on the chosen model, the uncertainty measure $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$ is minimal. But this uncertainty measure does not have an upper bound, and in the worst case a scale selection method based only on this measure could always choose the highest possible scale. The inclusion of an upper bound for the uncertainty measure would presumably remove this problem. Nielsen et al. [43] and Weber and Malik [57] handle this problem by constructing an error measure, which is a function of $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$ and a model error measure. Inspired by this I will therefore introduce the concept of a model error and use this error in the scale selection mechanism. A model error can be obtained by evaluating the energy functional $E(\vec{v})$ of Eq. (4.16) by entering the estimated optical flow field $\vec{v}(\vec{x}; \sigma_s, \sigma_t, \varpi)$ into $E(\vec{v})$. The model error is therefore given by the energy functional,

$$E(\vec{v}) = \int_{\vec{x} \in \mathbb{R}^2} w(\vec{x}) \|\vec{\Upsilon} \vec{v} \cdot \vec{\Upsilon} - \vec{u}\|_2^2 G(\vec{x}_0 - \vec{x}; \varpi) d\vec{x}.$$

The model error $E(\vec{v})$ is large whenever the average normal component of the optical flow field \vec{v} is far from the correct normal flow \vec{u} . In Fig. 4.9 an example of the model error for a real image sequence can be found.

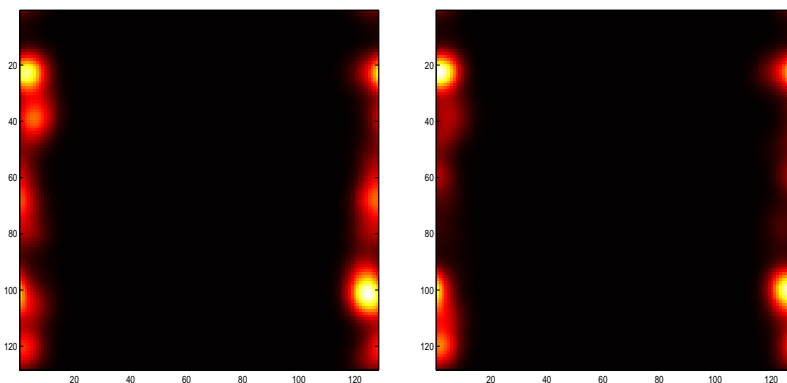


Figure 4.9: The model error $E(\vec{v})$ for the translating trees sequence (described further in Chapter 5). I show $E(\vec{v})$ for the zeroth (left) and first (right) order optic flow obtained by using Algorithm 4.4 at the scales $\sigma_s = 2.0$, $\sigma_t = 2.0$, and $\varpi = 4.0$. Light pixels indicates certain flow estimates and dark indicates uncertain flow estimates.

In order to combine both the measure of singularity, i.e. the uncertainty $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$, and the model error $E(\vec{v})$, I propose the following error measure

$$M(\vec{v}; \sigma_s, \sigma_t, \varpi) = E(\vec{v}) \cdot U(\vec{x}; \sigma_s, \sigma_t, \varpi).$$

This error measure is large whenever either of $E(\vec{v})$ or $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$ is large. If either of the two measures is small and the other one is large then the error measure $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$ is large and $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$ is small whenever both $E(\vec{v})$ and $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$ is small.

The proposed error measure can be used for scale selection, by choosing the set of scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$ at an image point that minimises $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$. This is formulated in the following algorithm:

Algorithm 4.5 (Optic Flow Estimation Using Scale Selection)

Assume that $\vec{v}(\vec{x}; \sigma_s, \sigma_t, \varpi)$ is an optical flow field obtained by Algorithm 4.4, $U(\vec{x}; \sigma_s, \sigma_t, \varpi)$ is the corresponding uncertainty measure, and $E(\vec{v})$ is the corresponding model error given by Eq. (4.16). An improved optical flow field $\vec{v}'(\vec{x}; \sigma'_s, \sigma'_t, \varpi')$ can then be obtained from $\vec{v}(\vec{x}; \sigma_s, \sigma_t, \varpi)$ by minimising the error measure $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$ over the spatiotemporal integration scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$,

$$\vec{\sigma}' = \arg \min_{\vec{\sigma}} M(\vec{v}; \sigma_s, \sigma_t, \varpi) \tag{4.26}$$

where

$$M(\vec{v}; \sigma_s, \sigma_t, \varpi) = E(\vec{v}) \cdot U(\vec{x}; \sigma_s, \sigma_t, \varpi). \tag{4.27}$$

In Algorithm 4.5 we have to search for minima in a 3-dimensional space of optic flow fields at spatiotemporal and integration scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$, which can be a computationally demanding job. But if we give up precision of the minima scales $\vec{\sigma}$, we can simply sample the space of scales and thereby turn the problem into a discrete search problem.

Other scale selection algorithms are possible using the error measure $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$ of Eq. (4.27). We could for instance choose to only select optic flow fields based on the minimisation of $M(\vec{v}; \sigma_s, \sigma_t, \varpi)$ over the integration scale ϖ . This would make the selection mechanism independent of the underlying normal flow. The normal flow could then be obtained by whatever method we wished, e.g. by scale selection as discussed in Sec. 4.3.1.

Implementation of Algorithm 4.5

In this algorithm we need to solve the 3-dimensional minimisation problem of Eq. (4.26). This can be implemented by sampling the spatiotemporal integration scale-space of optical flow fields determined by Algorithm 4.4. We hereby get a set of samples in $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$ of optical flow fields $\vec{v}(\vec{x}; \vec{\sigma})$ and a set of corresponding uncertainty measures $U(\vec{x}; \vec{\sigma})$ and model errors $E(\vec{v})$. We can then solve the minimisation problem of Eq. (4.26) at each pixel position \vec{x} by choosing the flow vector $\vec{v}(\vec{x}; \vec{\sigma})$ at the scales $\vec{\sigma}$, for which $M(\vec{v}; \vec{\sigma})$ is minimal among all the samples at that position.

4.5 Singularities in Optic Flow Fields

Nielsen and Olsen [45] have investigated the structure of optical flow fields especially normal flow fields, in the context of scale-space and catastrophe theory, [21]. They use the result by Arnol'd, which states that the structure of vector fields can be described through the singularities of the field. A point of singularity in an optical flow field is in general any point where the flow is vanishing.

When using a first order linear model of the optical flow field, as is done in Algorithm 4.3 and Algorithm 4.4, we have the opportunity to classify the possible singularities of the flow field through the eigenvalues of the linear model. Maurizot et al. [42] use this approach in their examination of singularities in turbulent flow.

Recall that the first order linear flow model used for the estimation of the optical flow field in Algorithm 4.4 can be written as

$$\vec{v} = \mathbf{A}\vec{x} + \vec{b}$$

where $\vec{b} = (\tilde{v}^x, \tilde{v}^y)^T$, $\vec{x} = (x, y)^T$, and

$$\mathbf{A} = \begin{pmatrix} \tilde{v}_x^x & \tilde{v}_y^x \\ \tilde{v}_x^y & \tilde{v}_y^y \end{pmatrix}.$$

It is only the matrix \mathbf{A} , which determines the flow around singular points, because $\vec{b} \equiv 0$ at singular points. In order to classify the flow fields around singularities it is valuable to transform the model matrix \mathbf{A} into its Jordan canonical form. The Jordan canonical form is described below:

Theorem 4.3 (Jordan Canonical Form)

Let $\mathbf{M} \in \mathbb{C}^{n \times n}$, then a nonsingular transformation matrix \mathbf{S} exists such that

$$\mathbf{S}\mathbf{M}\mathbf{S}^{-1} = \text{diag}(\mathbf{J}_1, \dots, \mathbf{J}_t)$$

which is called the Jordan Canonical Form, [21, 22], where

$$\mathbf{J}_i = \begin{pmatrix} \lambda_i & 1 & & \dots & 0 \\ 0 & \lambda_i & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \dots & & 0 & \lambda_i \end{pmatrix}$$

and $\mathbf{J}_i \in \mathbb{C}^{d_i \times d_i}$ is called a Jordan block. Here λ_i denotes the i th eigenvalue of \mathbf{M} and d_i is the degree of degeneracy of the i th eigenvalue. Furthermore $n = d_1 + \dots + d_t$.

Proof: I will not prove this theorem, but instead refer to [22]. □

The six types of possible flow fields around singularities is listed in Fig. 4.10. Furthermore, these types of singular points can be classified as stable (attractors), or unstable (repellers) depending on the sign of the eigenvalues. A node is said to be stable if the real part of the eigenvalues is negative, $\lambda_1 < \lambda_2 < 0$. A node is said to be unstable if the real part of the eigenvalues is positive, $\lambda_1 > \lambda_2 > 0$. The saddle node is called bistable, because the two eigenvalues have opposite signs. Nielsen and Olsen [45] group the center node, and spiral node together in what they call a whirl type node.

In order to find the singular points of the estimated optical flow field, Maurizot et al. [42, 41] propose that one simply uses the estimated local flow model and finds the points where the estimated velocity is zero or a zero crossing in the velocity components has occurred. That is,

$$\vec{v} = \mathbf{A}\vec{x} + \vec{b} = 0 \Rightarrow \vec{x} = -\mathbf{A}^{-1}\vec{b}.$$

This method demands a very good estimate of the model parameters. Maurizot et al. [41] tries to solve this problem in their algorithm described in Sec. 4.2, by positioning and resizing their local integration window F so that the window is centered at the singular point and no other singular points is present inside the window.

Kalitzin et al. [27] have proposed to use a variety of topological numbers as a means of analysing and classifying singularities of scalar images. Inspired by this idea, I will use the winding number of the vector field as a method of locating the singular points of the estimated optical flow field. Singularities in a vector field are given at points where the winding number is nonzero, [1]. The winding number can be defined as, [1]:

Definition 4.5 (Winding Number of Vector Fields)

Assume that $C(s) : I \mapsto \mathbb{R}^2$ is a closed regular⁵ parametric curve over the parameter interval $s \in I$ that surrounds a point $(x_0, y_0) \in \mathbb{R}^2$ in the vector field $\vec{v}(x, y) : \mathbb{R}^2 \mapsto \mathbb{R}^2$. Assume also that $C(s)$ does not run through any singular points of the field. Further assume that $\theta(x, y) \in [0; 2\pi]$ is the angle of the orientation of the vector field at the point (x, y) . The winding number η of the vector field $\vec{v}(x_0, y_0)$ at the point (x_0, y_0) is then given by

$$\eta = \oint_{s \in I} \frac{\partial \theta(C(s))}{\partial s} ds$$

where $\theta(x, y) = \arctan \frac{v^y}{v^x}$.

The following theorem insure us that at least one singular point exists within $C(s)$ when the winding number is nonzero.

Theorem 4.4 (Sum of Winding Numbers of Singular Points)

The winding number of the closed regular parametric curve $C(s) : I \mapsto \mathbb{R}^2$ equals the sum of the winding numbers of the singular points lying inside the curve.

Proof: The proof of this theorem can be found in the book by Arnol'd, [1]. □

⁵A parametric curve $\vec{r}(t)$ is regular if it is a C^1 differentiable function and $\vec{r}'_i(t) \neq \vec{0}, \forall t \in I$, where $\vec{0}$ is the zero vector.

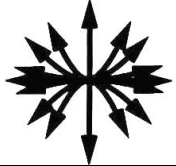
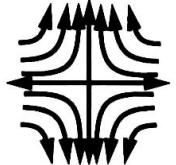
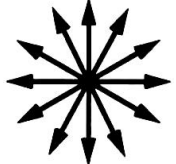
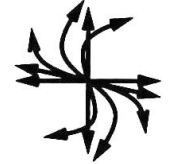
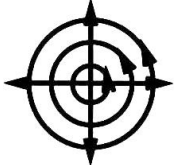

Eigenvalues	Jordan canonical form	Type	Schematic Drawing
$\lambda_1, \lambda_2 \in \mathbb{R}, \lambda_1 \neq \lambda_2$ and with same signs	$\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$	node	
$\lambda_1, \lambda_2 \in \mathbb{R}, \lambda_1 \neq \lambda_2$ and with opposite signs	$\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$	saddle node	
$\lambda_1, \lambda_2 \in \mathbb{R}$ and $\lambda_1 = \lambda_2$	$\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_1 \end{pmatrix}$	star node	
	$\begin{pmatrix} \lambda_1 & 1 \\ 0 & \lambda_1 \end{pmatrix}$	improper node	
$\lambda_1, \lambda_2 \in \mathbb{C}, \lambda_1 = \alpha + i\beta$ and $\lambda_2 = \alpha - i\beta$	$\begin{pmatrix} i\beta & 0 \\ 0 & -i\beta \end{pmatrix}$	center node	
	$\begin{pmatrix} \alpha + i\beta & 0 \\ 0 & \alpha - i\beta \end{pmatrix}$	spiral node	

Figure 4.10: The types of possible 2-dimensional vector fields around singular points of an analytical vector field. This table is a modified reproduction of a table from [42].

In the rest of this thesis I will only be concerned with the number of singularities in flow fields without making a classification of these, even though the winding number could be used as a classifier, [1]. An investigation into the structure of singularities of fluid flow in conjunction with the theory of turbulence, as done among others by Zhong et al. [61], could be of interest especially in the context of linear Gaussian scale-space, but I choose not to investigate this further, because of the time frame and limitations on this thesis.

4.6 Summary

In this chapter I presented the background theory of optic flow field estimation. An optic flow constraint equation was used to obtain a relation between the intensity surface and the underlying motion field.

In Sec. 4.2 I presented a method for extraction of the flow field from one image. This method assumed that the underlying physical flow is a fluid flow in which movement is tangential to the isophote contours of the image. I have modified the algorithm by introducing a scale-space for computation of image derivatives and an integration scale-space.

I proceeded in Sec. 4.3 with the introduction of a scale-space algorithm for normal flow field estimation.

In Sec. 4.4 I introduced a new method for estimation of optic flow fields using a local integration aperture function over a normal flow field. I incorporated an uncertainty measure of the used normal flow into this algorithm. Furthermore, I discussed methods for selecting the scales that are optimal for estimation of the optic flow and I proposed a scale selection mechanism based on these ideas.

In the last section of this chapter, Sec. 4.5, I described possible singularities of the optic flow vector field and a method for detection of these.

Chapter 5

Evaluation of Optic Flow Methods

In this chapter I will evaluate the quality of the optic flow produced by Algorithm 4.2, Algorithm 4.3, Algorithm 4.4, and Algorithm 4.5 presented in Chapter 4.

5.1 Measuring Errors in Optical Flow Fields

In order to evaluate the quality of the flow field produced by different algorithms, we need a quality measure. Barron et al. [7] have made an investigation of a large set of different optic flow methods. As a measure of the quality of the estimated flow field, they use angular errors between a known correct flow field and the estimated flow field. For optical flow fields the following definition of angular error can be used:

Definition 5.1 (Angular Error of Optic Flow)

The angular error of the estimated optical flow field $\vec{u} = (u^x, u^y, 1)^T$, according to the correct flow field $\vec{v} = (v^x, v^y, 1)^T$, is given by

$$\theta_{err} = \arccos \left(\frac{\vec{v}}{\|\vec{v}\|} \cdot \frac{\vec{u}}{\|\vec{u}\|} \right).$$

The components v^x and v^y of the velocity vector $\vec{v} = (v^x, v^y, 1)^T$ have dimension of length and the third component has dimension of time. This definition of angular error both accounts for the actual angular difference and the difference of the length of the velocity vectors.

The definition of angular errors of optic flow clearly does not work for normal flow. Barron et al. [7] therefore use the below stated definition when measuring angular errors of normal flow:

Definition 5.2 (Angular Error of Normal Flow)

The angular error of the estimated normal flow field $\vec{u} \in \mathbb{R}^2$, according to the correct flow field $\vec{v} \in \mathbb{R}^2$, is given by

$$\theta_{err} = \arcsin \left(\frac{\vec{v}}{\|\vec{v}\|} \cdot \frac{\vec{u}}{\|\vec{u}\|} \right),$$

where $\vec{v} = (v^x, v^y, 1)^T$ and $\vec{u} = \left(\frac{u^x}{\|\vec{u}\|}, \frac{u^y}{\|\vec{u}\|}, -\|\vec{u}\| \right)^T$.

In this definition it is used that ideally normal flow components u^i should lie on the plane, which has the flow vector \vec{v} as its normal vector. The angular error of the normal flow \vec{u} is the angle

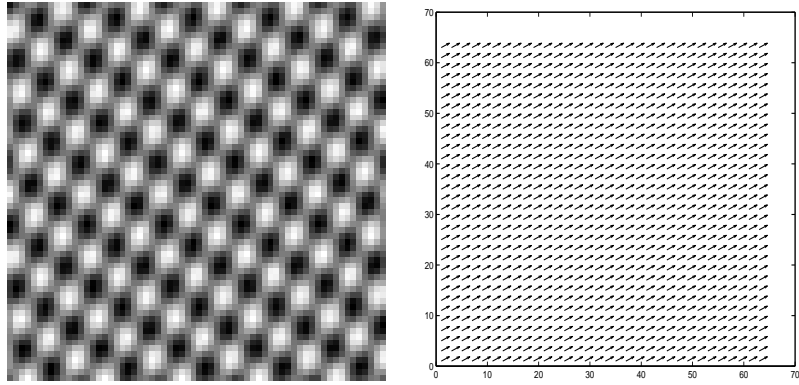


Figure 5.1: This is an image from the sinusoid sequence and the corresponding correct velocity field. Only half of all velocity vectors are plotted. The length of the vectors corresponds to the actual length.

between the normal flow vector and this plane. This definition also handles both the difference in the angle and in the length of the vectors.

I will use these two definitions in my investigation of the algorithms described in this thesis. Two Matlab functions for calculating the angular errors of a flow field, using the above stated definitions, can be found in Appendix D.2.2.

5.2 Collection of Test Data

Barron et al. [7] use a set of image sequences with known velocity fields. I have chosen to use four image sequences from the paper by Barron et al., which I have obtained together with the corresponding correct velocity fields from the Internet¹. These sequences will in this thesis be called the sinusoid, translating trees, diverging trees, and Yosemite sequences. The image sequences that I have chosen to use are all of synthetic motions. The description below, of the test sequences, is based on the paper by Barron et al. [7]:

- **Sinusoid sequence:** This image sequence is a translating pattern of sinusoidal plane-waves given by

$$\sin(\vec{k}_1 \cdot \vec{x} + \omega_1 t) + \sin(\vec{k}_2 \cdot \vec{x} + \omega_2 t)$$

with a wave length of 6 pixels, an orientation of 54° and -27° , and angular velocities of 1.63 and 1.02 pixels/frames. The resulting pattern is translated at each point with a velocity vector $\vec{x} = (1.585, 0.863)^T$. The image sequence consists of 16 images, each having 64×64 pixels. The pattern and the correct velocity field can be viewed in Fig. 5.1.

- **Translating trees sequence:** This sequence is created by simulating a translational camera motion along the x-axis of an image of some trees, which is mapped onto a tilted plane. The velocities varies between 1.76 to 2.21 pixels/frames from the left side of the image to the right side. The image sequence consists of 16 images of 128×128 pixels each. An example from the tree image sequence and the correct velocity field can be viewed in Fig. 5.2.

¹Via anonymous FTP from the following address: <ftp.csd.uwo.ca/pub/vision/TESTDATA/>.

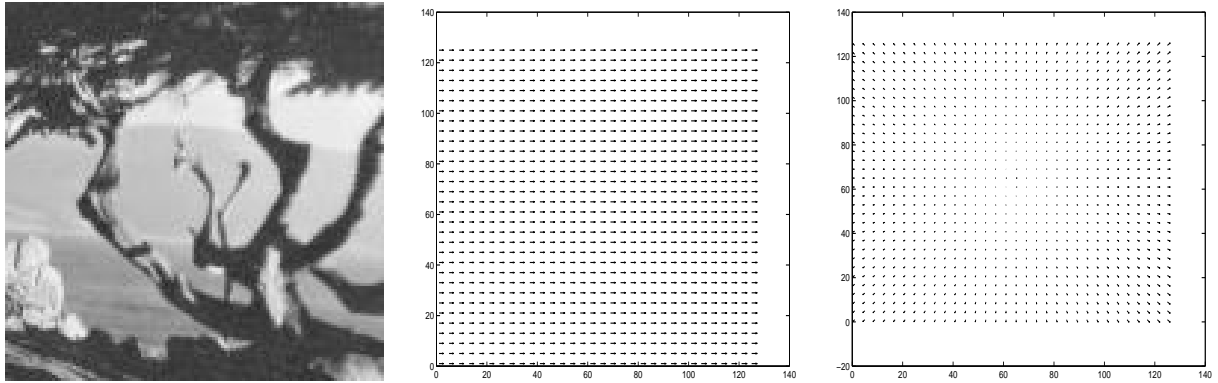


Figure 5.2: This is an image from the translating and diverging trees sequences and the corresponding correct velocity fields. Velocity vectors are only plotted for every fourth point of the field and the length of the vectors corresponds to their actual length.

- **Diverging trees sequence:** This sequence is created by simulating a camera motion along the optical axis and with focus of expansion in the center of the image. The tree image is mapped onto a tilted plane. The velocities varies from the left side of the image to the right side by 1.21 pixels/frame to 1.65 pixels/frame. The image sequence consists of 16 images of 128×128 pixels each. An example from the tree image sequence and the correct velocity field can be viewed in Fig. 5.2.
- **Yosemite sequence:** This sequence is a simulated flight through a valley. The sequence consists of both translational and divergent motion. The velocities varies from 1 pixel/frame to 4 pixels/frame. The image sequence consists of 8 images of each 256×256 pixels. An image from the flight sequence and the correct velocity field can be found in Fig. 5.3.

The results discussed in the next section were all calculated using the middle frame of the sequences. This is done in order to reduce the noise induced by the rim of the image sequence into the scale-space image sequence calculated in the algorithms. This noise comes from the fact that image sequences in general are not spatiotemporal periodic as required by the FFT algorithm. In the FFT algorithm it is assumed that the transformed function is periodic and that a whole period is presented to the algorithm, [10]. The lack of periodicity introduces sharp jumps in intensity values at the rim of the image, which leads to high frequencies noise at the spatiotemporal boundary of the image sequence. In order to remove this noise from the angular error measurements I only include pixels that has a position, which is two times the scale lengths away from the edges of the image.

5.3 Results

In this section I discuss the performance of Algorithm 4.2, Algorithm 4.3, Algorithm 4.4, and Algorithm 4.5 by evaluating the mean angular error of the flow fields produced by these algorithms for the image sequences described above. All angular errors and standard deviations are measured in units of degrees ($[\circ]$).

I have estimated the angular error, defined in Definition 5.2, of normal flow fields produced by using the zeroth order model of Algorithm 4.2 and the first order model of Algorithm 4.3 on

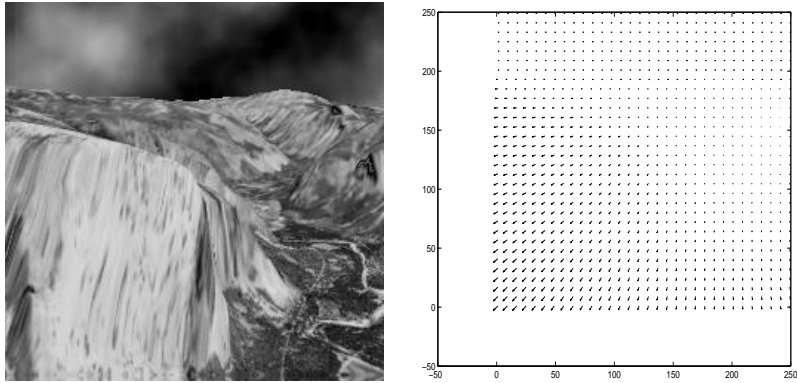


Figure 5.3: This is an image from the Yosemite sequence and the corresponding correct velocity field. Velocity vectors are only plotted for every eighth point of the field and the length of the vectors corresponds to their actual length.

Scale (σ_s, σ_t)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
(1.0, 1.0)	0.00	0.32	14.74	14.10
(2.0, 1.0)	-0.02	0.68	8.18	8.72
(4.0, 1.0)	-0.21	10.17	0.57	2.99
(1.0, 2.0)	0.15	1.70	3.05	8.74
(2.0, 2.0)	0.11	2.38	11.37	10.76
(4.0, 2.0)	-2.05	18.37	0.91	8.33

Table 5.1: Mean angular errors and standard deviations of the zeroth ($N = 0$) and first ($N = 1$) order normal flow of the sinusoid sequence estimated using Algorithm 4.2 and Algorithm 4.3. The normal flow has been estimated for different configurations of scales (σ_s, σ_t) .

the four image sequences. The results can be found in Table 5.1 – Table 5.4. Examples of the normal flow fields of the four image sequences can be found in Fig. 5.4.

From these tables it can be seen that the zeroth order normal flow model of Algorithm 4.2 in general performs better on translational motion than on divergent motion. For the first order model of Algorithm 4.3 it seems to be the other way around. The reason for this can be found in the fact that the zeroth order model only models translational motion contrary to the first order model, which also allows for divergent motion. It can also be seen that in general the best results for both the zeroth and first order normal flow are obtained at fine spatiotemporal scales, which is an expected property of both Algorithm 4.2 and Algorithm 4.3, because of the scale-space blurring of the image sequence.

In Table 5.1 and Table 5.2 it can be observed that for the sinusoid and translating trees sequence, the zeroth order normal flow estimates have better angular errors than the first order normal flow estimates. This phenomenon has also been found by Florack et al. [18] for optical flow fields estimated using fixed scales. They note that the accuracy of higher order derivatives, which is needed in the first order model, is weak for fine scales.

The angular errors of the diverging trees sequence found in Table 5.3, also seems to indicate that the zeroth order normal flow estimates have a better angular error than the flow estimated

Scale (σ_s, σ_t)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
(1.0, 1.0)	-0.23	4.20	5.45	17.80
(2.0, 1.0)	-0.13	2.57	4.60	17.38
(4.0, 1.0)	-0.20	3.49	3.96	17.71
(8.0, 1.0)	-0.12	1.76	5.77	14.68
(1.0, 2.0)	-0.11	2.92	4.62	18.11
(2.0, 2.0)	-0.01	1.96	4.48	18.20
(4.0, 2.0)	-0.06	3.22	5.08	18.12
(8.0, 2.0)	-0.04	2.02	5.98	16.56

Table 5.2: Mean angular errors and standard deviations of the zeroth ($N = 0$) and first ($N = 1$) order normal flow of the translating trees sequence estimated using Algorithm 4.2 and Algorithm 4.3. The normal flow has been estimated for different configurations of scales (σ_s, σ_t) .

Scale (σ_s, σ_t)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
(1.0, 1.0)	-0.54	4.72	3.01	12.60
(2.0, 1.0)	-0.39	4.47	2.32	12.49
(4.0, 1.0)	-1.19	7.42	1.85	11.77
(8.0, 1.0)	-4.06	10.95	2.80	9.20
(1.0, 2.0)	-0.15	3.57	3.11	13.04
(2.0, 2.0)	-0.19	3.66	2.70	12.36
(4.0, 2.0)	-0.97	6.87	2.07	11.54
(8.0, 2.0)	-3.93	10.84	2.92	9.27

Table 5.3: Mean angular errors and standard deviations of the zeroth ($N = 0$) and first ($N = 1$) order normal flow of the diverging trees sequence estimated using Algorithm 4.2 and Algorithm 4.3. The normal flow has been estimated for different configurations of scales (σ_s, σ_t) .

Scale (σ_s, σ_t)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
(1.0, 1.0)	-7.73	16.14	-4.53	20.52
(2.0, 1.0)	-7.32	15.18	-3.65	19.72
(4.0, 1.0)	-7.40	15.46	-3.73	20.01
(8.0, 1.0)	-8.17	16.35	-4.50	21.00
(1.0, 2.0)	-1.96	18.61	-0.84	23.01
(2.0, 2.0)	-0.53	17.73	1.20	22.23
(4.0, 2.0)	0.63	17.72	3.16	22.55
(8.0, 2.0)	0.44	18.35	4.62	23.21

Table 5.4: Mean angular errors and standard deviations of the zeroth ($N = 0$) and first ($N = 1$) order normal flow of the Yosemite sequence estimated using Algorithm 4.2 and Algorithm 4.3. The normal flow has been estimated for different configurations of scales (σ_s, σ_t) .

via the first order model. It would have been fair to assume that the first order model would perform better than the zeroth order model on a divergent flow field, because this model allows for divergent flow, which the zeroth order model does not. The reason for this discrepancy can again be found in the lack of accuracy in the higher order derivatives needed in the first order normal flow model.

The Yosemite sequence consist of both translational and divergent motion and the flow field of the sequence is quite complex. From Table 5.4 it can be seen that both Algorithm 4.2 and Algorithm 4.3 handles this sequence well for certain choices of scales. We again see that the zeroth order model perform better than the first order model.

In Table 5.5 I list angular errors of the normal flow of the divergent trees sequence produced by other methods. Unfortunately, I have only been able to find reports of the performance of other normal flow methods for the divergent trees sequence. By comparing the results in Table 5.3 with Table 5.5 it can be seen that both the zeroth and first order model perform fairly well on the divergent trees sequence compared to other normal flow methods. Note that none of the methods in Table 5.5 includes 100 % density of the flow, and taking this into account makes the normal flow produced by Algorithm 4.2 and Algorithm 4.3 good.

In order to evaluate the performance of both the zeroth and first order optic flow model of Algorithm 4.4 I have computed the zeroth and first order optical flow fields of the four image sequences. I used both the zeroth and first order normal flow produced by Algorithm 4.2 and Algorithm 4.3 as input for the optic flow algorithm. The angular errors of the optical flow fields were calculated by using Definition 5.1 and can be found in Table 5.6 – Table 5.9. Examples of the optical flow fields of the four image sequences can be found in Fig. 5.5.

From the results in Table 5.6 – Table 5.7 it can be seen that Algorithm 4.4 handles translational motion equally well, for both the zeroth order ($M = 0$) and first order ($M = 1$) model. The results based on the first order normal flow ($N = 1$) is very disappointing and the reason for this is that the error, introduced by the higher order derivatives used for the computation of the normal flow, are propagated into and amplified in the approximated optic flow.

In Table 5.8 – Table 5.9 we see a similar pattern. The errors of the results based on the first order ($N = 1$) normal flow are very high. Results based on zeroth order ($N = 0$) normal flow are much better and it is interesting to note that first order ($M = 1$) optic flow results are

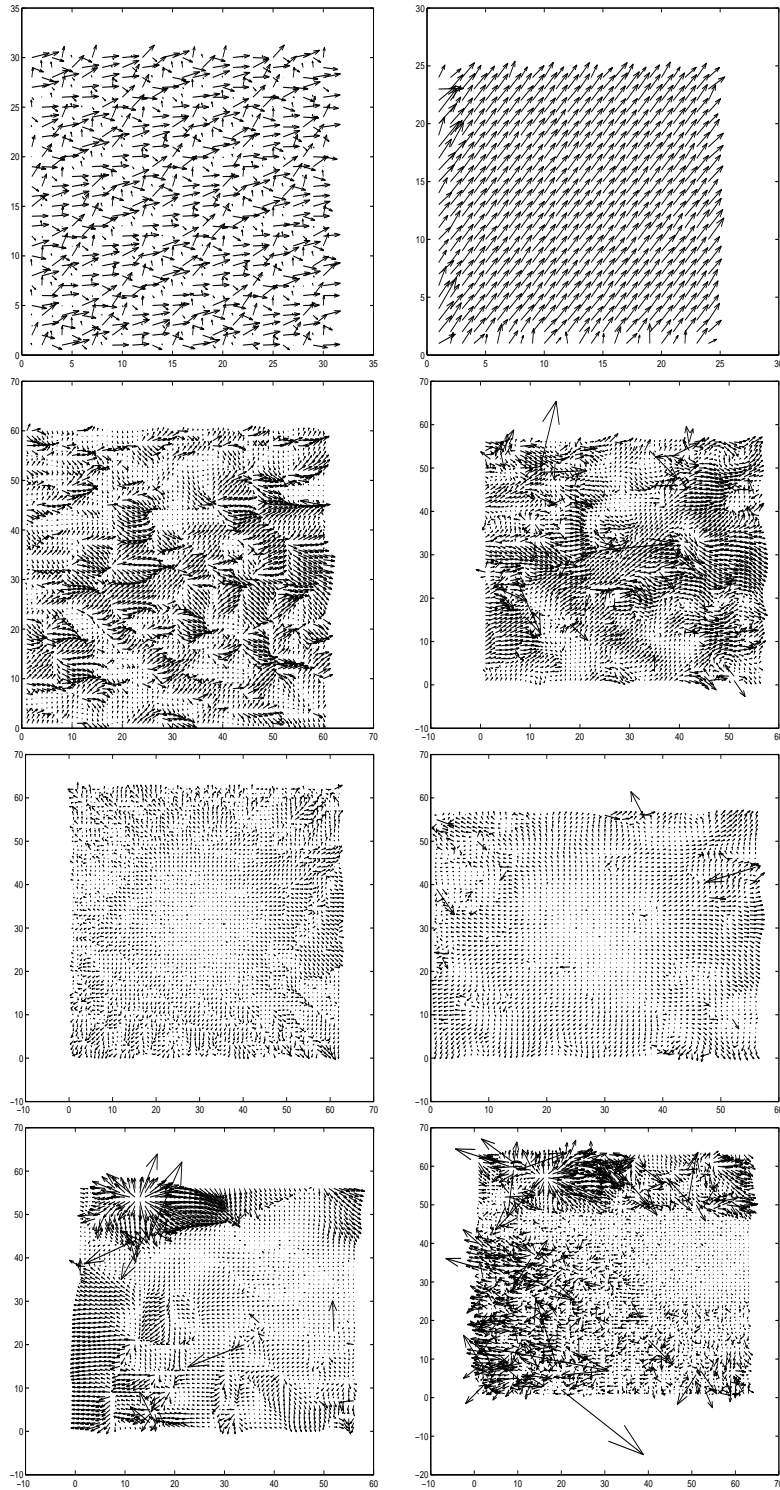


Figure 5.4: Examples of zeroth (left) and first (right) order normal flow for the four image sequences. The used scales are for the Sinusoid sequence $\vec{\sigma} = (1.0, 1.0)^T$ and $\vec{\sigma} = (4.0, 1.0)^T$, the translating sequence $\vec{\sigma} = (2.0, 2.0)^T$ and $\vec{\sigma} = (4.0, 1.0)^T$, the diverging trees sequence $\vec{\sigma} = (1.0, 2.0)^T$ and $\vec{\sigma} = (4.0, 1.0)^T$, and for the Yosemite sequence $\vec{\sigma} = (8.0, 2.0)^T$ and $\vec{\sigma} = (1.0, 2.0)^T$. Only optic flow vectors which are more than two spatial scale σ_s lengths away from the image edges are shown.

Techniques	Mean error [°]	Standard Deviation [°]	Density
Lucas & Kanade (LS) ($\lambda_1 \geq 1.0$)	1.0	0.83	36.0 %
Lucas & Kanade (LS) ($\lambda_1 \geq 5.0$)	0.86	0.70	46.0 %
Lucas & Kanade (Raw) ($\ \nabla L\ \geq 5.0$)	0.77	0.85	53.5 %
Heeger	1.92	3.18	25.8 %
Waxman et al. ($\sigma_f = 2.0$)	8.26	11.16	8.8 %
Fleet & Jepson ($\tau = 1.25$)	-0.04	0.78	61.0 %
Fleet & Jepson ($\tau = 2.5$)	-0.11	1.30	77.3 %

Table 5.5: Mean angular error of the normal flow fields produced by other methods for the divergent trees sequence. The density column show how much of the complete flow field is included in the computation of the angular error. The results are taken from Barron et al. [7]. An explanation of each of the methods not discussed in this thesis, can be found in Barron et al. [7].

better than the zeroth order ($M = 0$) results. This is not surprising because these two sequences consists of divergent motion and we would expect the first order model to be better than the zeroth order model for this type of motion.

From Table 5.6 – Table 5.9 it can be seen that Algorithm 4.4 encounter problems when the integration scale ϖ is smaller than or close to the inner spatial scale σ_s of the normal flow. This is not surprising, because this situation corresponds to using a vanishing integration window size.

In Table 5.12 I list mean angular errors for other optic flow methods. Comparing the results for the zeroth order normal flow model ($N = 0$) in Table 5.6 with Table 5.12 it can be seen that for properly chosen scales both the zeroth and the first order model used in Algorithm 4.4 outperform nearly all other optic flow techniques on the sinusoid sequence.

In Table 5.7 it can be observed that for zeroth order normal flow ($N = 0$) and properly chosen scales both the zeroth and first order model of Algorithm 4.4 handles translational motion of complex image structures well. The results based on the first order normal flow ($N = 1$) are much worse, which, as explained earlier, can be attributed to the noise introduced by the higher order derivatives needed in this model. Observe that for both normal flow models the choice of fine spatial scales leads to a drastic reduction of the mean angular error and the standard deviation of the angular error when the local integration scale ϖ increases. Not surprisingly it can also be seen that the spatiotemporal scales (σ_s, σ_t) used for the estimation of the normal flow also affects the angular error. For the translating trees sequence the best combination of mean angular error and standard deviation for the zeroth and first order optic flow based on zeroth order normal flow is found at the scales $\sigma_s \in [2.0, 4.0]$ and $\sigma_t = 2.0$, which is also the set of scales for which the zeroth order normal flow algorithm, Algorithm 4.2, performs best, as can be seen in Table 5.2. We can also see that the first order optic flow model perform a little better than the zeroth order model and that at coarse spatiotemporal scales (σ_s, σ_t) the optic flow produced by Algorithm 4.4 results in large angular errors.

From Table 5.7 we can also see that we encounter an angular error minimum when the integration scale ϖ increases and we find it at $\sigma_s = 4.0$, $\sigma_t = 2.0$, and approximately $\varpi = 8.0$.

For properly chosen combinations of spatiotemporal and integration scales ($\sigma_s, \sigma_t, \varpi$) it can be seen by comparing the zeroth order ($N = 0$) normal flow results in Table 5.7 with Table 5.12 and Table 5.13 that Algorithm 4.4 outperforms nearly all other optical flow techniques for the

Scale ($\sigma_s, \sigma_t, \varpi$)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
M=0, (1, 1, 1)	0.09	0.06	34.87	8.35
M=0, (1, 1, 2)	0.06	0.04	34.38	4.44
M=0, (1, 1, 4)	0.06	0.04	34.00	1.36
M=0, (1, 1, 8)	0.08	0.04	33.88	0.19
M=1, (1, 1, 1)	0.09	0.06	33.76	8.13
M=1, (1, 1, 2)	0.06	0.04	34.69	4.30
M=1, (1, 1, 4)	0.06	0.04	34.11	1.36
M=1, (1, 1, 8)	0.08	0.04	33.88	0.18
M=0, (2, 1, 1)	0.16	0.14	9.14	3.57
M=0, (2, 1, 2)	0.09	0.05	7.34	0.52
M=0, (2, 1, 4)	0.06	0.04	7.25	0.11
M=0, (2, 1, 8)	0.12	0.07	7.22	0.03
M=1, (2, 1, 1)	0.14	0.12	11.37	4.39
M=1, (2, 1, 2)	0.09	0.05	7.36	0.53
M=1, (2, 1, 4)	0.06	0.04	7.25	0.11
M=1, (2, 1, 8)	0.12	0.07	7.22	0.03
M=0, (1, 2, 1)	0.84	0.49	9.46	5.30
M=0, (1, 2, 2)	0.67	0.36	7.65	1.01
M=0, (1, 2, 4)	0.58	0.28	7.60	0.23
M=0, (1, 2, 8)	0.45	0.20	7.64	0.10
M=1, (1, 2, 1)	0.85	0.52	12.53	6.69
M=1, (1, 2, 2)	0.67	0.36	7.70	1.01
M=1, (1, 2, 4)	0.58	0.28	7.60	0.23
M=1, (1, 2, 8)	0.45	0.20	7.64	0.10
M=0, (2, 2, 1)	0.95	0.52	13.46	3.86
M=0, (2, 2, 2)	0.73	0.37	12.45	1.59
M=0, (2, 2, 4)	0.61	0.28	12.40	0.75
M=0, (2, 2, 8)	0.43	0.23	12.52	0.36
M=1, (2, 2, 1)	0.90	0.50	13.76	4.57
M=1, (2, 2, 2)	0.73	0.37	12.40	1.55
M=1, (2, 2, 4)	0.61	0.28	12.39	0.75
M=1, (2, 2, 8)	0.42	0.22	12.52	0.36

Table 5.6: Mean angular errors and standard deviations of the optic flow of the sinusoid sequence estimated using Algorithm 4.4 for different configurations of scales ($\sigma_s, \sigma_t, \varpi$) and choices of normal flow N and optic flow M order. Only pixels which is two integration scale ϖ lengths away from the image edges are included in the computation of the mean angular error.

Scale ($\sigma_s, \sigma_t, \varpi$)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
M=0, (1, 2, 1)	1.67	5.83	28.73	17.82
M=0, (1, 2, 2)	0.99	4.50	22.73	13.27
M=0, (1, 2, 4)	0.75	3.20	16.14	9.09
M=0, (1, 2, 8)	0.47	0.92	11.30	5.80
M=1, (1, 2, 1)	3.61	11.65	35.48	22.90
M=1, (1, 2, 2)	0.87	1.76	26.82	15.24
M=1, (1, 2, 4)	0.71	2.02	19.86	11.28
M=1, (1, 2, 8)	0.43	0.99	12.01	5.61
M=0, (2, 2, 1)	0.98	3.20	32.12	19.66
M=0, (2, 2, 2)	0.66	2.51	24.62	14.01
M=0, (2, 2, 4)	0.42	0.87	16.81	9.15
M=0, (2, 2, 8)	0.31	0.31	11.89	6.49
M=1, (2, 2, 1)	1.20	3.23	38.43	22.86
M=1, (2, 2, 2)	0.70	1.44	30.10	16.85
M=1, (2, 2, 4)	0.52	1.23	20.79	11.33
M=1, (2, 2, 8)	0.28	0.56	12.92	6.25
M=0, (4, 2, 1)	1.06	3.28	34.92	21.18
M=0, (4, 2, 2)	0.69	1.53	27.25	15.73
M=0, (4, 2, 4)	0.43	0.34	17.99	9.83
M=0, (4, 2, 8)	0.31	0.19	11.76	6.29
M=1, (4, 2, 1)	1.32	4.27	39.81	24.20
M=1, (4, 2, 2)	0.75	1.63	32.14	17.75
M=1, (4, 2, 4)	0.45	0.46	22.43	12.34
M=1, (4, 2, 8)	0.25	0.25	13.32	5.98
M=0, (8, 2, 1)	2.03	5.08	34.93	21.72
M=0, (8, 2, 2)	1.37	2.61	28.39	18.24
M=0, (8, 2, 4)	0.83	0.91	20.23	13.03
M=0, (8, 2, 8)	0.54	0.34	15.15	7.30
M=1, (8, 2, 1)	3.54	8.80	38.95	24.46
M=1, (8, 2, 2)	1.88	3.53	31.96	21.28
M=1, (8, 2, 4)	1.06	1.49	22.08	13.97
M=1, (8, 2, 8)	0.54	0.45	14.69	7.06

Table 5.7: Mean angular errors and standard deviations of the optic flow for the translating trees sequence estimated using Algorithm 4.4 for different configurations of scales ($\sigma_s, \sigma_t, \varpi$) and choices of normal flow N and optic flow M order. Only pixels which is two integration scale ϖ lengths away from the image edges are included in the computation of the mean angular error.

translating trees sequence. The results based on the first order ($N = 1$) normal flow model are very disappointing.

The angular errors of the optic flow produced by Algorithm 4.4 for the divergent trees sequence can be found in Table 5.8. We see that the first order ($M = 1$) optic flow model performs better than the zeroth order ($M = 0$) model. The results based on the first order ($N = 1$) normal flow are close to and in some cases better than the results based on the zeroth order normal flow. This is as we would have expected, because divergent motion is better modeled by first order models.

From Table 5.3 it can be seen that the best zeroth and first order normal flow fields are obtained at the range of scales $\sigma_s \in [1.0, 4.0]$ and $\sigma_t = 2.0$ and we should therefore expect to get reasonable estimates of the optic flow for combinations of these scales. This also seems to be the case, because these combinations of spatiotemporal scales give the best combinations of both mean angular error and standard deviation for the estimated optical flow fields, except for first order ($M = 1$) optic flow results based on first order ($N = 1$) normal flow which seems to perform best at $\sigma_s = 8$, $\sigma_t = 2$, and $\varpi = 8$.

If we compare the angular errors of the optic flow of the diverging trees sequence with the angular errors for optic flow found by other methods, Table 5.12 and Table 5.14, we see that Algorithm 4.4 compares well to other methods for results based both on zeroth and first order normal flow.

The Yosemite sequence consists of a mixture of translational and divergent motion. The mean and standard deviation of the angular errors of this sequence can be found in Table 5.9 and an example of the estimated flow field can be found in Fig. 5.5. Again we see that the first order optic flow model performs slightly better than the zeroth order model, which is not surprising taking the previous results on translational and divergent motion into account.

Comparing the results in Table 5.9 with the results in Table 5.12 and Table 5.15 it can be seen that Algorithm 4.4 performs a little bit worse than other methods on the Yosemite sequence.

In order to evaluate the performance of the scale selection algorithm, Algorithm 4.5, I have used scale-spaces of zeroth and first order normal flow and zeroth and first order optic flow for the four image sequences. The mean angular errors and the standard deviation of the angular errors for the sequences can be viewed in Table 5.10. The flow fields obtained by using Algorithm 4.5 can be viewed in Fig. 5.6.

From Table 5.10 it can be seen that Algorithm 4.5 handles divergent motion and translational motion well in the case of results based on zeroth ($N = 0$) order normal flow. In the case of results based on first ($N = 1$) order normal flow we get, as we would expect from the previous results, very poor results except in the case of the diverging trees sequence. We see that in all cases the first ($M = 1$) order optic flow model is an improvement compared to the zeroth ($M = 0$) order model as we would expect.

The angular errors of the optic flow produced by Algorithm 4.5 based on zeroth order normal flow compares very well with other methods as listed in Table 5.12. In the case of the results without full density produced by other techniques, Table 5.13 – Table 5.15, my method seems inferior, but remember that the density of my results are 100 %.

In order to evaluate the impact of introducing the proposed normal flow uncertainty measure $w(\vec{x}) = \|\mathbf{A}_{\vec{x}}^{-1}\|_F^{-2}$ as a weight in Algorithm 4.4 I have calculated optical flow fields using this algorithm without the uncertainty measure $w(\vec{x})$. This is done by setting the uncertainty measure equal to one at all pixels. The mean angular error and standard deviation of the flow fields of the translating trees, diverging trees, and the Yosemite sequences can be found in Table 5.11.

Scale ($\sigma_s, \sigma_t, \varpi$)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
M=0, (1, 2, 1)	2.86	6.09	14.48	11.02
M=0, (1, 2, 2)	2.43	3.42	10.58	7.14
M=0, (1, 2, 4)	3.19	3.04	8.99	5.39
M=0, (1, 2, 8)	3.83	2.28	6.93	4.21
M=1, (1, 2, 1)	5.46	13.28	20.70	17.20
M=1, (1, 2, 2)	1.17	1.74	11.82	8.24
M=1, (1, 2, 4)	1.16	2.36	8.90	6.08
M=1, (1, 2, 8)	1.12	1.51	6.82	4.07
M=0, (2, 2, 1)	3.06	3.25	15.72	11.90
M=0, (2, 2, 2)	3.07	2.86	11.66	8.03
M=0, (2, 2, 4)	3.41	2.64	8.95	5.47
M=0, (2, 2, 8)	4.12	2.44	6.44	3.58
M=1, (2, 2, 1)	3.13	4.24	20.39	16.15
M=1, (2, 2, 2)	2.31	2.46	13.38	9.47
M=1, (2, 2, 4)	1.49	1.88	9.38	6.61
M=1, (2, 2, 8)	0.91	0.93	6.44	3.61
M=0, (4, 2, 1)	6.98	6.82	16.02	12.34
M=0, (4, 2, 2)	5.87	4.78	12.14	8.81
M=0, (4, 2, 4)	4.95	3.23	8.46	5.68
M=0, (4, 2, 8)	5.19	3.10	5.75	3.21
M=1, (4, 2, 1)	8.44	9.18	19.44	15.24
M=1, (4, 2, 2)	6.41	5.93	13.52	10.00
M=1, (4, 2, 4)	4.18	2.93	8.98	6.82
M=1, (4, 2, 8)	1.49	0.88	5.03	3.20
M=0, (8, 2, 1)	15.23	12.36	17.75	12.30
M=0, (8, 2, 2)	13.36	9.81	13.93	9.37
M=0, (8, 2, 4)	10.70	6.90	9.03	5.83
M=0, (8, 2, 8)	9.45	4.52	5.83	3.13
M=1, (8, 2, 1)	18.06	16.12	19.64	15.88
M=1, (8, 2, 2)	13.97	11.42	14.56	10.68
M=1, (8, 2, 4)	11.06	7.26	10.17	6.72
M=1, (8, 2, 8)	6.21	3.44	4.63	2.98

Table 5.8: Mean angular errors and standard deviations of the optic flow for the diverging trees sequence estimated using Algorithm 4.4 for different configurations of scales ($\sigma_s, \sigma_t, \varpi$) and choices of normal flow N and optic flow M order. Only pixels which is two integration scale ϖ lengths away from the image edges are included in the computation of the mean angular error.

Scale ($\sigma_s, \sigma_t, \varpi$)	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
M=0, (1, 2, 1)	24.73	22.72	34.27	21.78
M=0, (1, 2, 2)	20.15	17.37	29.74	17.88
M=0, (1, 2, 4)	17.37	14.16	26.36	14.51
M=0, (1, 2, 8)	17.33	12.43	25.04	11.42
M=1, (1, 2, 1)	29.83	27.54	39.03	25.15
M=1, (1, 2, 2)	21.83	19.00	31.26	18.77
M=1, (1, 2, 4)	17.25	13.58	26.75	14.66
M=1, (1, 2, 8)	15.20	10.45	24.08	11.33
M=0, (2, 2, 1)	22.25	20.73	35.37	22.56
M=0, (2, 2, 2)	19.24	16.68	30.77	18.54
M=0, (2, 2, 4)	17.16	13.89	26.91	14.65
M=0, (2, 2, 8)	16.92	11.32	24.97	11.13
M=1, (2, 2, 1)	23.18	22.43	40.06	25.81
M=1, (2, 2, 2)	19.84	17.77	33.05	19.84
M=1, (2, 2, 4)	16.94	13.45	27.84	15.30
M=1, (2, 2, 8)	15.33	10.19	24.61	11.40
M=0, (4, 2, 1)	22.38	19.17	36.72	22.50
M=0, (4, 2, 2)	20.51	16.68	32.82	18.70
M=0, (4, 2, 4)	18.82	14.37	28.82	14.90
M=0, (4, 2, 8)	18.42	11.95	26.46	11.60
M=1, (4, 2, 1)	24.36	22.13	42.07	25.77
M=1, (4, 2, 2)	21.18	17.88	35.94	20.29
M=1, (4, 2, 4)	18.71	14.14	30.48	15.44
M=1, (4, 2, 8)	17.30	10.69	26.31	11.38
M=0, (8, 2, 1)	25.12	20.84	39.50	22.59
M=0, (8, 2, 2)	23.96	19.75	36.80	19.62
M=0, (8, 2, 4)	22.39	17.83	33.54	16.86
M=0, (8, 2, 8)	21.37	15.69	30.43	13.39
M=1, (8, 2, 1)	29.46	25.50	44.87	26.68
M=1, (8, 2, 2)	25.40	21.53	39.80	21.67
M=1, (8, 2, 4)	22.92	18.43	36.07	17.56
M=1, (8, 2, 8)	20.98	13.61	32.21	13.35

Table 5.9: Mean angular errors and standard deviations of the optic flow for the Yosemite sequence estimated using Algorithm 4.4 for different configurations of scales ($\sigma_s, \sigma_t, \varpi$) and choices of normal flow N and optic flow M order. Only pixels which is two integration scale ϖ lengths away from the image edges are included in the computation of the mean angular error.

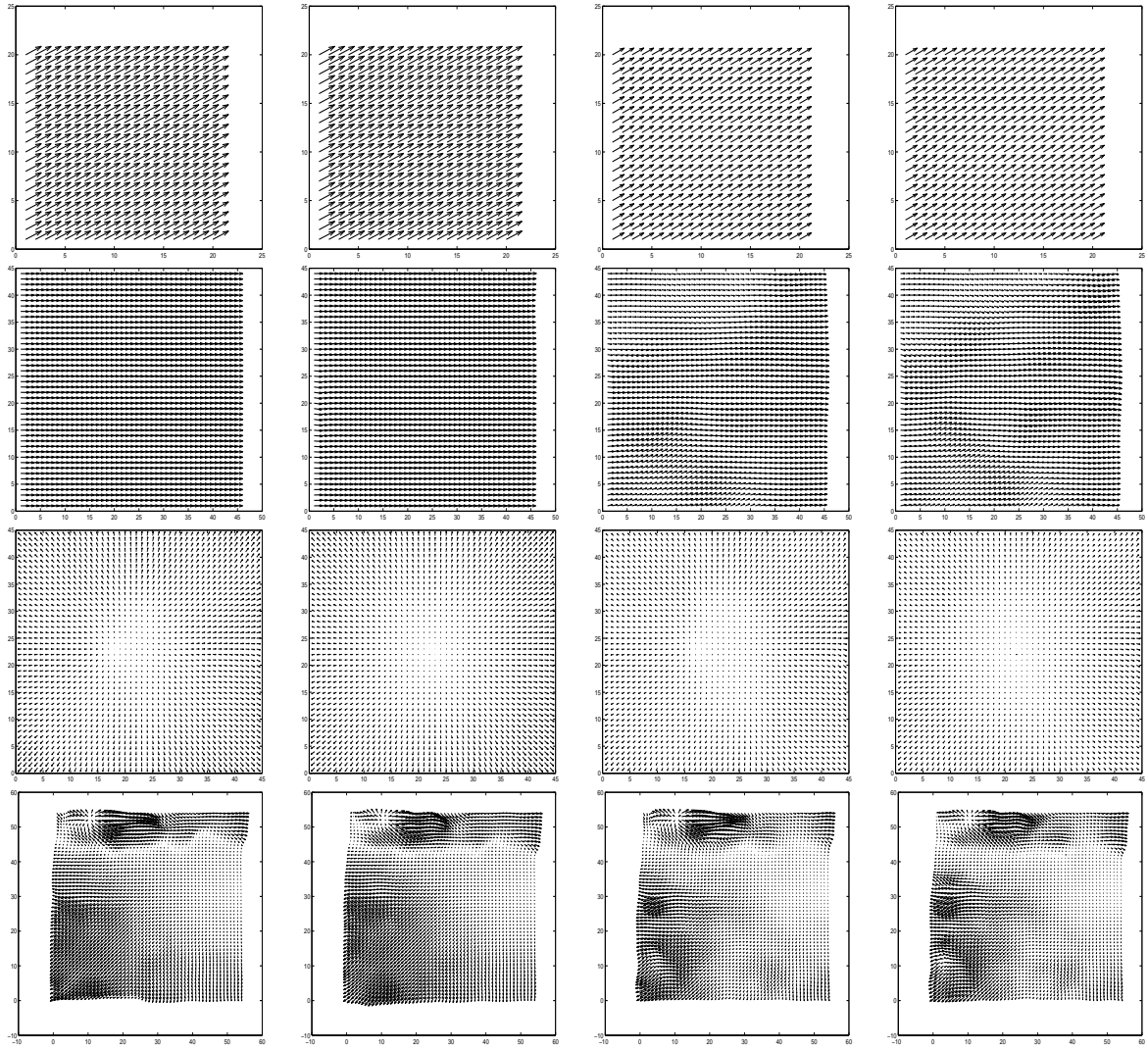


Figure 5.5: Examples of zeroth (first column) and first (second column) order optic flow based on zeroth order normal flow and zeroth (third column) and first (fourth column) order optic flow based on first order normal flow for the four image sequences. The scales used are for the Sinusoid sequence $\vec{\sigma} = (2.0, 1.0, 4.0)^T$, for the translating and diverging trees sequences $\vec{\sigma} = (2.0, 2.0, 8.0)^T$, and for the Yosemite sequence $\vec{\sigma} = (2.0, 2.0, 8.0)^T$. Only optic flow vectors are shown, which are more than two times the scale away from the image edges.

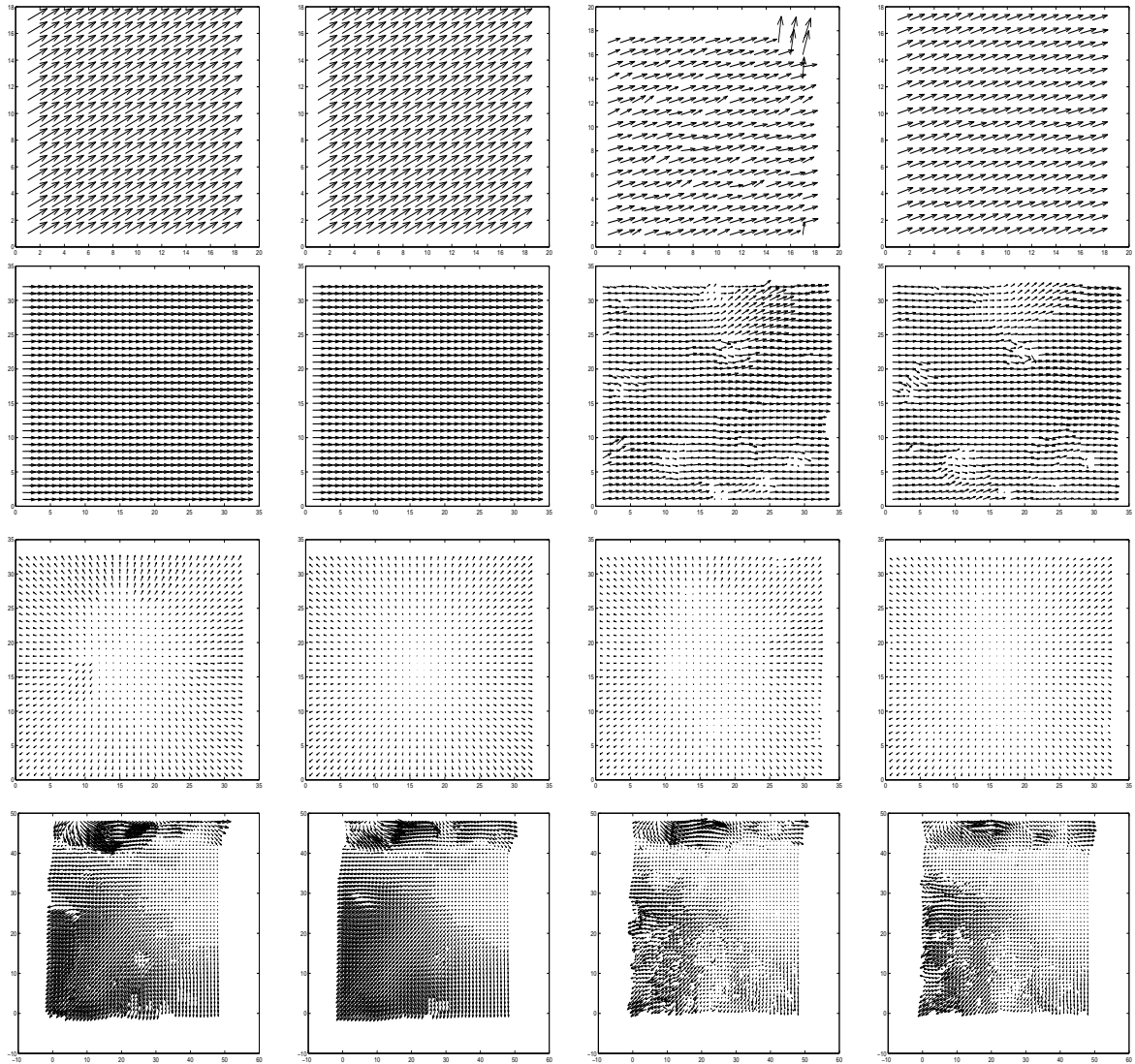


Figure 5.6: Zeroth (first column) and first (second column) order scale selected optic flow based on zeroth order normal flow and zeroth (third column) and first (fourth column) order scale selected optic flow based on first order normal flow for the four image sequences. The scale selected optic flow has been computed using Algorithm 4.5. A description of the used spatiotemporal integration scales and angular errors of the flow can be found in Table 5.10.

Image sequence	$N = 0$		$N = 1$	
	Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
Sinusoid, M=0	0.07	0.05	13.96	2.71
Sinusoid, M=1	0.06	0.01	12.62	0.74
Translating trees, M=0	0.34	0.23	13.32	9.01
Translating trees, M=1	0.15	0.11	13.08	9.25
Diverging trees, M=0	5.02	3.86	5.10	3.13
Diverging trees, M=1	0.80	0.47	4.63	2.61
Yosemite, M=0	11.50	15.66	24.94	13.31
Yosemite, M=1	8.51	12.57	23.26	11.99

Table 5.10: Mean angular errors and standard deviations of the scale selected zeroth ($M = 0$) and first ($M = 1$) order optic flow for the four image sequences estimated using Algorithm 4.4 and scale selected using Algorithm 4.5. I have used both the zeroth ($N = 0$) and first ($N = 1$) order normal flow computed by Algorithm 4.2 and Algorithm 4.3. I used the following scale samples $(\sigma_s, \sigma_t, \varpi)$; $\sigma_s \in \{1.0, 1.414, 2.0, 2.828, 4.0, 5.656, 8.0\}$, $\sigma_t \in \{1.0, 2.0\}$, and $\varpi \in \{1.0, 2.0, 4.0, 8.0\}$. Only pixels which are more than two times the sum of the largest spatial and integration scales $2 \times (\sigma_s + \varpi)$ away from the image edges were included in the computation of the mean angular error.

Comparing the results in this table with the results in Table 5.7 – Table 5.9 it can be seen that in general the results without the uncertainty weight are worse for fine scales as we would expect. But at coarse scales it seems that the results produced without the uncertainty measure are better than those with this measure, except for the Yosemite sequence where in general all results with this measure are better than those without. The reason for this behaviour could be that at coarse scales errors at individual pixels does not have the same degrading impact as for finer scales, due to the fact that contributions to the flow at one pixel are collected in a smooth fashion over a larger area. This shows that the chosen normal flow uncertainty measure might not be the optimal solution. We could come up with much more sophisticated solutions like for instance an uncertainty measure based on the stability of the image geometry, but in the rest of this thesis I will settle with the already discussed choice of normal flow uncertainty measure because it at fine scales produce better results than without this measure.

5.4 Summary

In this chapter I have evaluated the quality of the flow fields produced by Algorithm 4.2, Algorithm 4.3, Algorithm 4.4, and Algorithm 4.5 presented in Chapter 4.

Algorithm 4.2 and Algorithm 4.3 both handle translational and divergent motion well. The first order normal flow produced by Algorithm 4.3 has larger errors due to the fact that the higher order derivatives needed in this computation are more error prone than lower order derivatives and are as well poorly estimated at fine scales.

Algorithm 4.4 seems to handle both translational and divergent motion reasonably well for both the zeroth and first order optic flow model. The optic flow produced by this algorithm is of course dependent on the quality of the underlying normal flow field, which can be seen in the large errors for the optic flow results based on first order normal flow.

The scale selection algorithm, Algorithm 4.5, also seems to handle both translational and divergent motion well for both the zeroth and first order optic flow model. The scale selection

Sequence	Scale ($\sigma_s, \sigma_t, \varpi$)	$N = 0$		$N = 1$	
		Mean error [°]	Standard deviation [°]	Mean error [°]	Standard deviation [°]
Trans. trees	M=0,(2.0, 2.0, 1.0)	1.07	2.85	34.33	22.48
	M=0,(2.0, 2.0, 2.0)	0.70	1.08	27.10	17.42
	M=0,(2.0, 2.0, 4.0)	0.43	0.44	18.96	11.18
	M=0,(2.0, 2.0, 8.0)	0.24	0.18	12.57	6.10
	M=1,(2.0, 2.0, 1.0)	2.50	8.37	40.17	24.17
	M=1,(2.0, 2.0, 2.0)	0.79	1.22	31.77	18.80
	M=1,(2.0, 2.0, 4.0)	0.46	0.52	22.31	12.22
	M=1,(2.0, 2.0, 8.0)	0.22	0.23	14.02	6.54
Div. trees	M=0,(2.0, 2.0, 1.0)	3.83	5.15	16.98	12.83
	M=0,(2.0, 2.0, 2.0)	3.10	3.36	12.39	8.97
	M=0,(2.0, 2.0, 4.0)	2.20	1.59	8.45	5.51
	M=0,(2.0, 2.0, 8.0)	2.05	1.24	5.24	3.15
	M=1,(2.0, 2.0, 1.0)	7.66	14.10	20.19	15.20
	M=1,(2.0, 2.0, 2.0)	3.07	4.12	14.27	10.12
	M=1,(2.0, 2.0, 4.0)	1.66	1.57	9.42	6.15
	M=1,(2.0, 2.0, 8.0)	0.90	0.65	5.81	2.97
Yosemite	M=0,(2.0, 2.0, 1.0)	23.81	22.93	37.05	24.82
	M=0,(2.0, 2.0, 2.0)	21.46	20.09	32.37	21.55
	M=0,(2.0, 2.0, 4.0)	18.86	16.60	27.56	17.67
	M=0,(2.0, 2.0, 8.0)	17.40	13.68	24.82	14.05
	M=1,(2.0, 2.0, 1.0)	29.36	29.80	41.56	27.53
	M=1,(2.0, 2.0, 2.0)	21.89	20.89	34.66	22.34
	M=1,(2.0, 2.0, 4.0)	19.12	17.10	28.50	17.54
	M=1,(2.0, 2.0, 8.0)	16.92	12.95	24.78	13.59

Table 5.11: Mean angular error and standard deviation of optical flow fields obtained by using Algorithm 4.4 without the uncertainty measure proposed in this thesis.

algorithm is dependent on the quality of the used optic flow.

Furthermore, it can be concluded that using the proposed uncertainty measure of the underlying normal flow as a weight in the computation of optic flow seems in general to be a reasonable choice. It would be more satisfiable to come up with a measure which are founded more directly on the geometry of the image sequence contrary to numerical stability, but in the lack of this measure we must accept the already proposed uncertainty measure.

From the results presented in this chapter the overall conclusion must be that in general the combination of zeroth order normal flow and zeroth and first order optic flow gives the best results. The reason for this is that we in the first order normal flow model need to compute higher order derivatives of the image sequence, which does not give accurate results.

Sequence	Techniques	Mean error [°]	Standard Deviation [°]
Sinusoid	Horn & Schunck (original)	4.19	0.50
	Horn & Schunck (modified)	2.55	0.59
	Lucas & Kanade (no thresholding)	2.47	0.16
	Uras et al. (no thresholding)	2.59	0.71
	Nagel	2.55	0.93
	Singh ($n = 2, w = 2, N = 2$)	2.24	0.02
	Fleet & Jepson ($\tau = 1.25$)	0.03	0.01
Translating trees	Horn & Schunck (original)	38.72	27.67
	Horn & Schunck (modified)	2.02	2.27
	Uras et al. (no thresholding)	0.62	0.52
	Nagel	2.44	3.06
	Anandan	4.54	3.10
	Singh ($n = 2, w = 2, N = 2$)	1.64	2.44
	Singh ($n = 2, w = 2, N = 4$)	1.25	3.29
	Florack et al. ($M = 0, \sigma_s = 2.0, \sigma_t = 2.0$)	1.60	8.68
	Florack et al. ($M = 1, \sigma_s = 4.0, \sigma_t = 2.0$)	0.71	5.73
Florack et al. ($M = 1$, scale selection)	0.49	1.92	
Diverging trees	Horn & Schunck (original)	12.02	11.72
	Horn & Schunck (modified)	2.55	3.67
	Uras et al. (no thresholding)	4.64	3.48
	Nagel	2.94	3.23
	Anandan	7.65	4.96
	Singh ($n = 2, w = 2, N = 2$)	17.66	14.25
	Singh ($n = 2, w = 2, N = 4$)	8.60	4.78
	Florack et al. ($M = 0, \sigma_s = 2.0, \sigma_t = 2.0$)	3.88	10.65
	Florack et al. ($M = 1, \sigma_s = 4.0, \sigma_t = 2.0$)	2.65	8.63
Florack et al. ($M = 1$, scale selection)	1.15	3.32	
Yosemite	Horn & Schunck (original)	32.43	30.28
	Horn & Schunck (modified)	11.26	16.41
	Uras et al. (no thresholding)	10.44	15.00
	Nagel	11.71	10.59
	Anandan	15.84	13.46
	Singh ($n = 2, w = 2, N = 2$)	18.24	17.02
	Singh ($n = 2, w = 2, N = 4$)	13.16	12.07

Table 5.12: This table includes mean angular errors and standard deviations for other optical flow techniques. The results for the different techniques are obtained from Barron et al. [7] and Florack et al. [18]. Furthermore, only results where flow is known for the complete set of image points, i.e. 100 % density, are included here. An explanation of each of the methods not discussed in this thesis, can be found in the following papers [7, 18].

Techniques	Mean error [°]	Standard Deviation [°]	Density
Horn & Schunck (original, $\nabla L \geq 5.0$)	32.66	24.50	55.9 %
Horn & Schunck (modified, $\nabla L \geq 5.0$)	1.89	2.40	53.2 %
Lucas & Kanade ($\lambda_2 \geq 1.0$)	0.66	0.67	39.8 %
Lucas & Kanade ($\lambda_2 \geq 5.0$)	0.56	0.58	13.1 %
Uras et al. ($\det(H) \geq 1.0$)	0.46	0.35	41.8 %
Nagel ($\ \nabla\ _2 \geq 5.0$)	2.24	3.31	53.2 %
Singh (step 1, $n = 2, w = 2, \lambda_1 \leq 5.0, (N = 4)$)	0.72	0.75	41.4 %
Heeger (level 1)	4.53	2.41	57.8 %
Fleet & Jepson ($\tau = 2.5$)	0.32	0.38	74.5 %
Fleet & Jepson ($\tau = 1.25$)	0.23	0.19	49.7 %
Fleet & Jepson ($\tau = 1.0$)	0.25	0.21	26.8 %
Weber & Malik	0.49	0.35	96.8 %
Florack et al. ($M = 0, \sigma_s = 4, \sigma_t = 2$)	0.33	0.43	60 %
Florack et al. ($M = 1$, scale selection)	0.16	0.18	60 %
Florack et al. ($M = 0$, scale selection)	0.29	0.33	40 %
Florack et al. ($M = 1$, scale selection)	0.14	0.13	40 %

Table 5.13: Results for other techniques for the translating trees sequence are presented here. They are obtained from Barron et al. [7], Weber and Malik [57], and Florack et al. [18]. An explanation of each of the methods not discussed in this thesis, can be found in the following papers [7, 57, 18].

Techniques	Mean error [°]	Standard Deviation [°]	Density
Horn & Schunck (original, $\nabla L \geq 5.0$)	8.93	7.79	59.8 %
Horn & Schunck (modified, $\nabla L \geq 5.0$)	2.50	3.89	32.9 %
Lucas & Kanade ($\lambda_2 \geq 1.0$)	1.94	2.06	48.2 %
Lucas & Kanade ($\lambda_2 \geq 5.0$)	1.65	1.48	24.3 %
Uras et al. ($\det(H) \geq 1.0$)	3.83	2.19	60.2 %
Nagel ($\ \nabla\ _2 \geq 5.0$)	3.21	3.43	53.5 %
Singh (step 1, $n = 2, w = 2, \lambda_1 \leq 5.0, (N = 4)$)	7.09	6.59	3.9 %
Heeger (level 1)	4.95	3.09	73.8 %
Fleet & Jepson ($\tau = 2.5$)	0.99	0.78	61.0 %
Fleet & Jepson ($\tau = 1.25$)	0.80	0.73	46.5 %
Fleet & Jepson ($\tau = 1.0$)	0.73	0.46	28.2 %
Weber & Malik	3.18	2.50	88.6 %
Florack et al. ($M = 0, \sigma_s = 4, \sigma_t = 2$)	2.66	2.68	60 %
Florack et al. ($M = 1$, scale selection)	0.79	1.13	60 %
Florack et al. ($M = 0$, scale selection)	2.04	1.81	40 %
Florack et al. ($M = 1$, scale selection)	0.43	0.40	40 %

Table 5.14: Results for other techniques for the diverging trees sequence are presented here. They are obtained from Barron et al. [7], Weber and Malik [57], and Florack et al. [18]. An explanation of each of the methods not discussed in this thesis, can be found in the following papers [7, 57, 18].

Techniques	Mean error [°]	Standard Deviation [°]	Density
Horn & Schunck (original, $\nabla L \geq 5.0$)	25.41	28.14	59.6 %
Horn & Schunck (modified, $\nabla L \geq 5.0$)	5.48	10.41	32.9 %
Lucas & Kanade ($\lambda_2 \geq 1.0$)	4.10	9.58	35.1 %
Lucas & Kanade ($\lambda_2 \geq 5.0$)	3.05	7.31	8.7 %
Uras et al. ($\det(H) \geq 1.0$)	6.73	16.01	14.7 %
Nagel ($\ \nabla\ _2 \geq 5.0$)	6.03	11.04	32.9 %
Singh (step 1, $n = 2, w = 2, \lambda_1 \leq 5.0, (N = 4)$)	16.29	25.70	2.2 %
Heeger (level 1)	10.51	12.11	15.2 %
Fleet & Jepson ($\tau = 2.5$)	4.29	11.24	34.1 %
Fleet & Jepson ($\tau = 1.25$)	4.95	12.39	30.6 %
Weber & Malik	4.31	8.66	64.2 %

Table 5.15: Results for other techniques for the Yosemite sequence are presented here. They are obtained from Barron et al. [7] and Weber and Malik [57]. An explanation of each of the methods not discussed in this thesis, can be found in the following papers [7, 57].

Chapter 6

Turbulence

In our every day life, turbulence can be seen in a great variety of phenomena and has, for that reason, probably always fascinated mankind. We know that as early as around year 1500, Leonardo Da Vinci made presumably the first scientific studies of the phenomenology of turbulence, [20]. Turbulence can be found in both fluids and gases and the phenomenon can be observed at a variety of scales. At geophysical scales turbulence can be found in the form of meteorological phenomena such as the transport of water vapour and air through the atmosphere or the flow of water in the oceans. At much smaller scales turbulence can be found in the water flow of streams or in a pot of boiling water.

The theoretical foundation of fluid dynamics were laid with the invention of the Navier-Stokes equations in 1823, which encompasses the dynamics of fluids and gases. In 1922 Richardson proposed that turbulence is created by large scale velocity fluctuations, which is propagated onto smaller scales. This idea is the basis of the so-called *Richardson cascade model*. In 1941 Kolmogorov stated a set of hypotheses about the qualitative behaviour of turbulence and derived a set of laws from these hypotheses, but a deduction of these laws directly from the Navier-Stokes equations is yet to be found, and furthermore, a full theoretical understanding of the phenomena of turbulence has not yet been established.

Turbulence in fluid dynamical systems has been investigated either analytically, by simulation, or empirically. Analytical investigations concerns the analysis of the Navier-Stokes equations, which will be described in Sec. 6.1. Computational fluid dynamics¹ is used for the simulation of fluid dynamic systems. The Navier-Stokes equations are evaluated using for instance finite element methods to solve the partial differential equations of the Navier-Stokes equations. Empirical investigations of fluid flow, involves experimental setups and the analysis of the obtained data. Examples of experimental methods are the Particle Imagery Velocimetry (PIV) methods, where the velocity field of a turbulent flow is estimated by tracking particle motion over time.

In this chapter I will present relevant parts of the theory of turbulence in dynamical systems of fluids and gases, and I will try to relate it to the theory of linear Gaussian scale-space as described in Chapter 3. Furthermore, I will discuss experimental laws, which could be validated using optic flow methods. All theory of turbulence presented in this chapter is, when nothing else is stated, based on the book on turbulence by Frisch [20], Schröder's PhD thesis [55] and a masters thesis by Nielsen [46].

¹See [59] for an introduction to the various methods of computational fluid dynamics.

6.1 The Navier-Stokes Equations

As stated above fluid dynamics includes both the dynamics of fluids and gases. Fluid dynamics is governed by the laws of conservation of mass, conservation of energy, and Newton's second law

$$\vec{F} = m\vec{a}$$

where \vec{F} is the force, m the mass and \vec{a} the acceleration.

Turbulent flow obey the above stated laws and can be modeled by the Navier-Stokes equation and the continuity equation², which are the corner stones of fluid dynamics. These two equations can for an incompressible fluid be stated as

$$\begin{aligned}\partial_t \vec{v} + (\vec{v} \cdot \nabla) \vec{v} &= -\nabla p + \nu \nabla^2 \vec{v} + \vec{f} \\ \nabla \cdot \vec{v} &= 0\end{aligned}\tag{6.1}$$

where \vec{v} is the velocity of the fluid, p is the pressure exerted on the fluid, ν is the kinematic viscosity of the fluid, and \vec{f} is an external driving force. The Navier-Stokes equation and the continuity equation have to be supplemented with initial and boundary conditions, such as the behaviour of the fluid flow near the walls of the confinement basin of the fluid. In order to obtain the Navier-Stokes equation and the continuity equation for compressible fluids, the change of fluid density ρ should be taken into account in each term of the equations.

If the Navier-Stokes equation is put into dimensionless form, where velocities are measured in units of the characteristic velocity V and lengths are measured in units of the characteristic length scale³ L , then the flow only has one control parameter, [20]. This control parameter is called the Reynolds number:

Theorem 6.1 (The Reynolds Number)

The Reynolds number R is the only control parameter of an incompressible flow, if the Navier-Stokes equation Eq. (6.1) with no external driving force is put on a dimensionless form. The Reynolds number is then defined as

$$R \equiv \frac{LV}{\nu},$$

where L and V respectively is the characteristic length scale and velocity of the flow and ν is the kinematic viscosity of the fluid.

Proof: We need to put the Navier-Stokes equation with no external driving force into a dimen-

²In this thesis I will sometimes use the abbreviation *the Navier-Stokes equations* instead of writing *the Navier-Stokes equation and the continuity equation*.

³Here characteristic length scale means the most prevailing scale of the flow and the same goes for the characteristic velocity.

sionless form. To do this we need the following definitions

$$\begin{aligned}
x' &\equiv \frac{x}{L} \\
y' &\equiv \frac{y}{L} \\
z' &\equiv \frac{z}{L} \\
\vec{v}' &\equiv \frac{\vec{v}}{V} \\
p' &\equiv \frac{p}{\rho V^2} \\
\nabla' &= \left(\frac{\partial}{\partial x'}, \frac{\partial}{\partial y'}, \frac{\partial}{\partial z'} \right)^T \equiv L \nabla \\
t' &\equiv t \frac{V}{L}
\end{aligned}$$

where ρ is the density of the flow. The Navier-Stokes equation with no external driving force for compressible fluids can now be written as

$$\rho \frac{\partial(V \vec{v}')}{\partial(\frac{L}{V} t')} + \rho \left(V \vec{v}' \cdot \frac{1}{L} \nabla' \right) V \vec{v}' = -\frac{1}{L} \nabla'(\rho V^2 p') + \frac{\eta}{L^2} \nabla'^2 V \vec{v}'$$

where η is the dynamic viscosity. Because we are only interested in incompressible fluids, we can assume that the density ρ is constant. By multiplying both sides of the equation with $L/(\rho V^2)$ we get

$$\frac{\partial \vec{v}'}{\partial t'} + (\vec{v}' \cdot \nabla') \vec{v}' = -\nabla' p' + \frac{\eta}{\rho L V} \nabla'^2 \vec{v}'.$$

Because the kinematic viscosity is given by $\nu = \eta/\rho$, we get

$$\frac{\partial \vec{v}'}{\partial t'} + (\vec{v}' \cdot \nabla') \vec{v}' = -\nabla' p' + \frac{1}{R} \nabla'^2 \vec{v}'.$$

□

The Reynolds number describes the degree of irregularity of the flow. It is small for laminar flow⁴ and turbulence is introduced into the flow at high Reynolds numbers, approximately $R > 30$. Turbulence at very high Reynolds numbers is also called fully developed turbulence.

When conducting experiments on fully developed turbulence another definition of the Reynolds number is often used. It is called the Taylor-scale Reynolds number R_λ and is defined as

$$R_\lambda \equiv \frac{v_{rms} \lambda}{\nu},$$

where ν is the viscosity and $v_{rms}^2 = \overline{(v_{\ell_0}^i)^2}$ is the root mean square velocity fluctuation $v_{\ell_0}^i$ of one of the velocity components v^i at the largest scale⁵ ℓ_0 . The Taylor-scale λ of the experiment is defined as

$$\frac{1}{\lambda^2} \equiv \frac{\overline{(v_i^i)^2}}{v_{rms}^2},$$

where $\overline{(v_i^i)^2}$ denotes the second order statistical moment⁶ of the partial derivative v_i^i .

⁴Smooth and regular flow, without turbulence.

⁵In Sec. 6.3 I will introduce the fact that turbulence only exists on a finite range of scales called the inertial range. The largest scale of this range is called the integral scale and is proportional to the size of the flow basin.

⁶A definition of the second order moment can be found below.

Statistical tools are very useful in the description of turbulence, because of the non-deterministic nature of turbulence. We cannot repeat an experiment involving turbulence and expect to arrive at the same quantitative results each time we perform the experiment. The qualitative behaviour of the experiment will on the other hand be the same. The qualitative properties of turbulence is best described by the usage of statistical tools. I will for this reason introduce some useful statistical concepts:

Definition 6.1 (The m th Order Statistical Moment)

The m th order moment of a random function $f(\vec{x})$ is defined as

$$\overline{f^m} \equiv \int_{-\infty}^{\infty} f^m p(f) df$$

where $p(f)$ is the probability density function of $f(\vec{x})$.

The first order moment \overline{f} is the mean value of the random function and is, in the case of spatial random functions, called the ensemble average in order to distinguish it from time averages. The second order moment $\overline{f^2}$ is called the variance, if the random function f is centred, meaning $\overline{f} = 0$.

I will now define the p th order longitudinal structure function which is of much importance, because a great deal of the laws of turbulence can be stated in terms of this function.

Definition 6.2 (The p th Order Longitudinal Structure Function)

The p th order longitudinal structure function $S_p(\vec{x}, \vec{\ell})$ is defined as

$$S_p(\vec{x}, \vec{\ell}) \equiv \overline{[\delta v_{\parallel}(\vec{x}, \vec{\ell})]^p}$$

where $\delta v_{\parallel}(\vec{x}, \vec{\ell})$ is the longitudinal velocity increment given by

$$\delta v_{\parallel}(\vec{x}, \vec{\ell}) \equiv [\vec{v}(\vec{x} + \vec{\ell}) - \vec{v}(\vec{x})] \cdot \frac{\vec{\ell}}{\ell}$$

and $\ell = \|\vec{\ell}\|$ (this notation will be used throughout this chapter whenever convenient).

The longitudinal velocity increment $\delta v_{\parallel}(\vec{x}, \vec{\ell})$ is the velocity increment between to points \vec{x} and $\vec{x} + \vec{\ell}$, projected onto the direction of the separation vector $\vec{\ell}$.

The following two definitions are taken from a paper by Mandelbrot and van Ness [40]. The first definition is necessary in order to define the important concept of self-similarity of random functions.

Definition 6.3 (Equality in Law)

Two random functions $X(t)$ and $Y(t)$ is said to be equal in law (\triangleq), if they have the same joint distribution function, and we write

$$X(t) \triangleq Y(t).$$

Frisch [20] notes that equality in law implies that the random functions $X(t)$ and $Y(t)$ have the same statistical moments.

Definition 6.4 (Self-Similarity)

The increment $\delta X(t, \ell) = X(t + \ell) - X(t)$ of a random function $X(t)$ is said to be self-similar with parameter h if for any $\lambda \in \mathbb{R}_+$ and any t

$$\delta X(t, \lambda \ell) \triangleq \lambda^h \delta X(t, \ell).$$

A self-similar random function has the same statistical properties for all scales. Informally one could say that the function “looks the same” at all scales. This behaviour is especially seen in fractal sets.

I will now define the fractal dimension of a function and I have chosen to use the Hausdorff dimension, as described in [49], as a definition of the fractal dimension⁷. In order to state this definition I need the following definition of the diameter of a set:

Definition 6.5 (The Diameter of a Set)

Let A be a set of points in an N -dimensional metric space. The diameter of A is then defined as the largest distance between two points in the closure of A , $x, y \in \bar{A} = A \cup \delta A$ (where δA is the set of limit points of A). That is the diameter $|A|$ of A is given by,

$$|A| = \sup_{x, y \in \bar{A}} |x - y|.$$

Definition 6.6 (The Hausdorff Dimension)

Let A be a set of points in an N -dimensional metric space and let S be a countable collection of subsets of A for which the diameter $\epsilon_i = |S_i|$ of the subset $S_i \in S$, $i = 1, \dots, m$, is confined by

$$0 < \epsilon_i \leq \delta.$$

Furthermore, let S cover A , i.e. $A \subset \bigcup_i S_i$. Define the d -dimensional quality $\Gamma_d(\delta)$ of the Hausdorff measure as

$$\Gamma_d(\delta) \equiv \inf_{S \in M} \sum_{i=1}^m \epsilon_i^d,$$

where M is the set of all possible collections S . The d -dimensional Hausdorff measure is then defined as

$$\Gamma_d \equiv \lim_{\delta \rightarrow 0} \Gamma_d(\delta).$$

It can be shown that a critical value D_H of d exists, for which the Hausdorff measure $\Gamma_d = +\infty$ if $d < D_H$ and $\Gamma_d = 0$ if $d > D_H$. When $d = D_H$ the Hausdorff measure Γ_d is either zero, $+\infty$, or a positive finite number. This critical value D_H is called the Hausdorff dimension.

Whenever the set is a geometric form like a point, line, or a plane, the Hausdorff dimension is an integer and corresponds to the topological dimension of the set. Furthermore, the Hausdorff dimension is always equal to or smaller than the topological dimension N of the set, $D_H \leq N$.

6.2 Scale-Space Energy Budget

In his book on turbulence, Frisch [20] derives a scale-by-scale energy budget of turbulence using ideal low-pass and high-pass filtered functions, which removes all Fourier coefficients above or

⁷Note that a variety of definitions exist for the fractal dimension.

below a certain wave number. Frisch remarks that other filters than the ideal filter can be used. In this section, I will restate his derivation using the Gaussian low-pass filters of linear Gaussian scale-space and thereby introduce different important physical concepts in the context of linear Gaussian scale-space.

Let us define the inner product of two square integrable functions $f(\vec{x}), g(\vec{x}) \in L^2(\mathbb{R}^N)$,

$$\langle f, g \rangle \equiv \int_{\vec{x} \in \mathbb{R}^N} f(\vec{x})g(\vec{x}) d\vec{x}. \quad (6.2)$$

In order to derive the energy budget it is necessary to first define a scale-space velocity $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) : \mathbb{R}^{(N+1)} \times \mathbb{R}_+^2 \setminus \{0\} \mapsto \mathbb{R}^{(N+1)}$ as applying the scale-space operator $T_{\vec{\sigma}} : \vec{v}(\vec{x}) \mapsto \vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$, defined in Eq. (3.4), at the spatiotemporal scale $\vec{\sigma} = (\sigma_s, \sigma_t)^T$ to the velocity vector $\vec{v}(\vec{x})$,

$$\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) \equiv T_{\vec{\sigma}}\vec{v}(\vec{x}) = \int_{\vec{x}' \in \mathbb{R}^{N+1}} \vec{v}(\vec{x}')G(\vec{x} - \vec{x}'; \sigma_s, \sigma_t) d\vec{x}'.$$

Furthermore, I am going to enforce a periodic boundary condition on the solution $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$ of the Navier-Stokes equations, which can be stated as

$$\vec{v}_{\vec{\sigma}}(\vec{x} + \vec{n}L; \vec{\sigma}) = \vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) \quad (6.3)$$

where $\vec{x} \in \mathbb{R}^{(N+1)}$ is the position and $\vec{n} \in \mathbb{Z}^{(N+1)}$ is a vector of the number of period lengths $L \in \mathbb{R}_+$ in each dimension. This is a fairly strong condition, but if we are interested in the unbounded space \mathbb{R}^3 we can generalise by letting $L \rightarrow \infty$.

We are now ready to state the energy budget of turbulence in the context of linear Gaussian scale-space.

Proposition 6.1 (Scale-Space Energy Budget of Turbulence)

Using the periodic boundary condition Eq. (6.3) on the Navier-Stokes equations, the energy budget of turbulence in linear Gaussian scale-space is given by the scale-space energy budget equation,

$$\partial_t \varepsilon_{\vec{\sigma}} + \Pi_{\vec{\sigma}} = -2\nu\Omega_{\vec{\sigma}} + \mathcal{F}_{\vec{\sigma}},$$

where the cumulative energy is

$$\varepsilon_{\vec{\sigma}} = \frac{1}{2} \overline{\langle |\vec{v}_{\vec{\sigma}}|^2 \rangle},$$

the cumulative enstrophy is

$$\Omega_{\vec{\sigma}} = \frac{1}{2} \overline{\langle |\vec{\omega}_{\vec{\sigma}}|^2 \rangle},$$

the cumulative energy injection is

$$\mathcal{F}_{\vec{\sigma}} = \overline{\langle \vec{v}_{\vec{\sigma}}, \vec{f}_{\vec{\sigma}} \rangle}$$

and the energy flux through the spatiotemporal scale $\vec{\sigma}$

$$\Pi_{\vec{\sigma}} = \overline{\langle \vec{v}_{\vec{\sigma}}, T_{\vec{\sigma}}[(\vec{v} \cdot \nabla)\vec{v}] \rangle}.$$

Here $\vec{\omega}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) = \nabla \times \vec{v}_{\vec{\sigma}}$ is the vorticity of the velocity field $\vec{v}_{\vec{\sigma}}$ at spatiotemporal scale $\vec{\sigma}$.

Proof: To prove this proposition we start off by applying the scale-space operator $T_{\vec{\sigma}}$ to the Navier-Stokes equation Eq. (6.1) and we obtain the following by using the fact that the scale-space operator commutes with differentiation, Eq. (3.5),

$$\partial_t \vec{v}_{\vec{\sigma}} + T_{\vec{\sigma}} [(\vec{v} \cdot \nabla) \vec{v}] = -\nabla p_{\vec{\sigma}} + \nu \nabla^2 \vec{v}_{\vec{\sigma}} + \vec{f}_{\vec{\sigma}}.$$

We take the inner product of these equations with the scale-space velocity $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$ and get

$$\partial_t \left\langle \frac{|\vec{v}_{\vec{\sigma}}|^2}{2} \right\rangle + \langle \vec{v}_{\vec{\sigma}}, T_{\vec{\sigma}} [(\vec{v} \cdot \nabla) \vec{v}] \rangle = -\langle \vec{v}_{\vec{\sigma}}, \nabla p_{\vec{\sigma}} \rangle + \nu \langle \vec{v}_{\vec{\sigma}}, \nabla^2 \vec{v}_{\vec{\sigma}} \rangle + \langle \vec{v}_{\vec{\sigma}}, \vec{f}_{\vec{\sigma}} \rangle.$$

We now calculate the ensemble average of this equation, because we are interested in the statistical properties as explained earlier,

$$\partial_t \left\langle \frac{|\vec{v}_{\vec{\sigma}}|^2}{2} \right\rangle + \overline{\langle \vec{v}_{\vec{\sigma}}, T_{\vec{\sigma}} [(\vec{v} \cdot \nabla) \vec{v}] \rangle} = -\overline{\langle \vec{v}_{\vec{\sigma}}, \nabla p_{\vec{\sigma}} \rangle} + \nu \overline{\langle \vec{v}_{\vec{\sigma}}, \nabla^2 \vec{v}_{\vec{\sigma}} \rangle} + \overline{\langle \vec{v}_{\vec{\sigma}}, \vec{f}_{\vec{\sigma}} \rangle}. \quad (6.4)$$

Frisch [20] proves that if we use a periodic boundary condition on the Navier-Stokes equations, we can, from properties of the ensemble average of periodic functions and the continuity equation, get the relations $\overline{\langle \vec{v}_{\vec{\sigma}}, \nabla p_{\vec{\sigma}} \rangle} = 0$ and $\overline{\langle \vec{v}_{\vec{\sigma}}, \nabla^2 \vec{v}_{\vec{\sigma}} \rangle} = -\overline{\langle |\vec{\omega}_{\vec{\sigma}}|^2 \rangle}$. Using these relations we can then rearrange equation Eq. (6.4) and obtain

$$\partial_t \left\langle \frac{|\vec{v}_{\vec{\sigma}}|^2}{2} \right\rangle + \overline{\langle \vec{v}_{\vec{\sigma}}, T_{\vec{\sigma}} [(\vec{v} \cdot \nabla) \vec{v}] \rangle} = -\nu \overline{\langle |\vec{\omega}_{\vec{\sigma}}|^2 \rangle} + \overline{\langle \vec{v}_{\vec{\sigma}}, \vec{f}_{\vec{\sigma}} \rangle}$$

and we have therefore reached

$$\partial_t \varepsilon_{\vec{\sigma}} + \Pi_{\vec{\sigma}} = -2\nu \Omega_{\vec{\sigma}} + \mathcal{F}_{\vec{\sigma}}.$$

□

The cumulative energy $\varepsilon_{\vec{\sigma}}$ is the cumulated mean energy of the flow at all spatiotemporal scales up to $\vec{\sigma}$. The cumulative enstrophy $\Omega_{\vec{\sigma}}$ is the cumulated mean energy of the vorticity or rotation $\vec{\omega}_{\vec{\sigma}} = \nabla \times \vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$ of the flow at all spatiotemporal scales up to $\vec{\sigma}$. The cumulative energy injection $\mathcal{F}_{\vec{\sigma}}$ is the cumulated mean energy injected into the flow at all spatiotemporal scales up to $\vec{\sigma}$.

The energy flux $\Pi_{\vec{\sigma}}$ through scale $\vec{\sigma}$ characterises the energy flux from higher scales to lower scales at the current scale. It can be shown (see Frisch [20]) that the energy flux $\Pi_{\vec{\sigma}}$ for homogeneous and isotropic turbulence can be expressed in terms of the third order longitudinal structure function $S_3(\ell)$, but I will not use this fact in this thesis and will therefore not comment further on this.

6.3 Kolmogorov's 1941 Theory

In this section I will describe some important theoretical results of turbulence and especially parts of the theory presented by Kolmogorov⁸ in 1941.

The course of turbulence has to be sought in the interaction of the fluid flow on different scales. Richardson proposed that fully developed turbulence consists of velocity fluctuations at different scales. These fluctuations are called eddies in the literature on turbulence.

⁸This theory is often called the K41 theory.

Eddies are introduced at large scales by external forces and they are turned into heat at small scales, where dissipation dominates. This phenomenon is called the Richardson cascade and a schematic drawing of this can be found in Fig. 6.1. The largest size or scale ℓ_0 of eddies in a turbulent flow is called the integral scale and is obviously bounded upwards by the scale of the confinement basin of the flow. The smallest scale η is called the Kolmogorov dissipation scale and is defined as

$$\eta \equiv \left(\frac{\nu^3}{\varepsilon} \right)^{1/4}$$

where ν is the viscosity of the fluid and ε is the mean energy dissipation per unit mass. In the language of linear Gaussian scale-space the integral scale ℓ_0 corresponds to the outer scale and the dissipation scale η corresponds to the inner scale. The range of scales or the scale-space between the dissipation scale η and the integral scale ℓ_0 is called the inertial range of the turbulent flow.

If we use linear Gaussian scale-space as the scaling mechanism, we can write the velocity of the flow at the scale $\vec{\sigma} = (\sigma_s, \sigma_t)^T$ as the scale-space velocity $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$. The size of eddies ℓ and the spatial scale σ_s at which they occur are in the 1-dimensional case interchangeable, i.e. we can write $\ell = \sigma_s$.

The space filling or size ℓ of eddies at a certain scale in scale-space decreases with the scale parameter. If we introduce a discrete numbering, ($n = 0, 1, 2, \dots$), of generations of scales in the inertial range, then the space filling ℓ_n of eddies decreases as

$$\ell_n = \ell_0 r^n$$

where $0 < r < 1$ is a constant usually chosen to be $r = 1/2$. In the K41 theory, it is assumed that the set of children or sub-eddies are taking up as much space as their parent eddy. This phenomenon can be viewed schematically in Fig. 6.1. Under the assumption of equal space filling at all scales, the number of eddies per unit volume should grow with n as r^{-3n} .

In his K41 theory, Kolmogorov made a set of hypotheses about turbulent flow. Frisch [20] has restated these hypotheses and I use his formulation.

Hypothesis 6.1 (Restoration of Symmetries)

In the limit of infinite Reynolds number, all the possible symmetries of the Navier-Stokes equation, usually broken by the mechanisms producing the turbulent flow, are restored in a statistical sense at small scales and away from boundaries.

In this hypothesis the statement *symmetries of the Navier-Stokes equation* designates the symmetry group G of the Navier-Stokes equations, [20]. This means that for any solution \vec{v} of the Navier-Stokes equations and all $g \in G$ the function $g\vec{v}$ is also a solution to the Navier-Stokes equations. The known symmetries of the Navier-Stokes equations are: Space and time translation, Galilean transformation, parity, rotation and scaling.

Hypothesis 6.2 (Self-Similarity of Turbulence)

In the limit of infinite Reynolds number, the turbulent flow is self-similar, as defined in Definition 6.4, at small scales, i.e. it possesses a unique scaling exponent $h \in \mathbb{R}$, such that

$$\delta\vec{v}(\vec{x}, \lambda\vec{\ell}) \triangleq \lambda^h \delta\vec{v}(\vec{x}, \vec{\ell}) \quad \forall \lambda \in \mathbb{R}_+.$$

Under this hypothesis we assume that the turbulent flow has the same statistical properties for all scales in the inertial range and that a unique scaling exponent h exists.

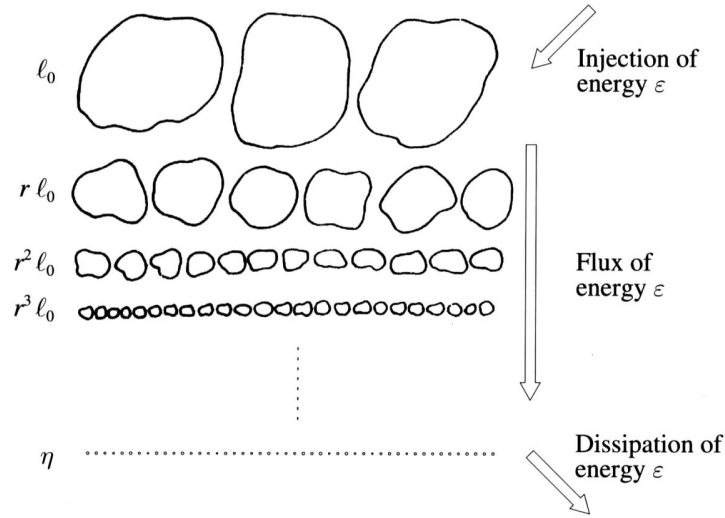


Figure 6.1: A schematic drawing of the Richardson cascade. It is shown that energy is injected into the fluid at the integral scale ℓ_0 . The energy is then transported down through eddies of increasingly smaller scales $\ell_k = r^k \ell_0$, $r \in]0, 1[$, until ℓ_k reaches the inner dissipation scale η where the energy is converted to heat. This figure is a reproduction of a drawing found in [20].

Hypothesis 6.3 (Finite Mean Dissipation Rate)

In the limit of infinite Reynolds number, the turbulent flow has a finite non-vanishing mean rate of energy dissipation ϵ per unit mass at small scales and away from boundaries.

This hypothesis is important for the derivation of the four-fifths law, which is stated below.

The main result of the K41 theory is the four-fifths law, which states the third order longitudinal structure function of the velocity difference in terms of the distance ℓ and the energy dissipation ϵ , [20].

Proposition 6.2 (Four-Fifths Law)

In the limit of infinite Reynolds number, the third order longitudinal structure function of homogeneous isotropic turbulence evaluated for increments ℓ within the inertial range, is given in terms of the mean energy dissipation ϵ per unit mass, assuming that Hypothesis 6.3 is valid, by

$$S_3(\vec{x}, \vec{\ell}) = -\frac{4}{5}\epsilon\ell$$

where $\ell = \|\vec{\ell}\|$.

I will not prove this proposition but instead refer to the book by Frisch [20], for a thorough explanation.

We can, in the proposition described above, drop the dependency on the position \vec{x} and the direction of the increment $\vec{\ell}$ and just write $S_3(\ell) = -\frac{4}{5}\epsilon\ell$, because we assume that the turbulence is homogeneous and isotropic. I will now state the more general Kolmogorov-Obukhov law:

Proposition 6.3 (The Kolmogorov-Obukhov Law)

In the limit of infinite Reynolds number and assuming homogeneous and isotropic turbulence, the p 'th order longitudinal structure function evaluated for increments ℓ within the inertial

range, is given in terms of the mean energy dissipation ε per unit mass and the increment ℓ ,

$$S_p(\ell) = C_p \varepsilon^{p/3} \ell^{p/3}$$

where C_p is a dimensionless factor.

This proposition can be proved by using the four-fifths law and Hypothesis 6.1 – Hypothesis 6.3 to get the scaling exponent $h = 1/3$ and then use dimensional analysis to see that $(\varepsilon \ell)^{p/3}$ has the same dimensions as S_p .

It is easily seen that the four-fifths law is a special case of the Kolmogorov-Obukhov law, with $C_3 = -4/5$. The factor C_3 of the four-fifths law is universal, i.e. it does not depend on the measured flow, but this is not necessarily true for C_p in general, $p \neq 3$, [20]. In the next section I will describe two experimental laws of turbulence, which also are special cases of the Kolmogorov-Obukhov law.

Fully developed turbulence is actually not self-similar at all points in space, but exhibit high activity at a fraction of the space confining the flow and no activity at the rest of the confinement space. A turbulent flow exhibiting this behaviour, is said to be intermittent. A consequence of this is that eddies only occupy a fraction of the parent eddy space, instead of the whole space as assumed in the K41 cascade model. For this reason the K41 cascade model does not handle the phenomenon of intermittency. Other cascade models has been developed in which intermittency is handled (See [20, 46] for an description of different cascade models). The β -model proposed by Frisch [20] is an example of a cascade model, which handles intermittency. This model states that the space occupied by active eddies is decreased by a factor β , $0 < \beta < 1$. The fraction p_ℓ of space occupied by an eddy with size $\ell = \ell_0 r^n$ decreases as a power of ℓ ,

$$p_\ell = \beta^{\frac{\ln(\ell/\ell_0)}{\ln r}} = \left(\frac{\ell}{\ell_0}\right)^{N-D_H}, \ell \rightarrow 0 \quad (6.5)$$

where $N - D_H \equiv \ln \beta / \ln r$, N is the topological dimension of the space, and D_H can be interpreted as the fractal dimension of the space filled by eddies, as defined in Definition 6.6.

6.4 Experimental Laws of Turbulence

In this section I will describe three experimental laws of turbulence and attempt to develop new experiments to verify these laws. The proposed experiments all use scale-space optical flow fields, which could be obtained by using the algorithms described in Chapter 4. I will look at the so-called 2/3-law and its dual, the 5/3-law. Furthermore, I will propose an experiment for examining the space filling of eddies.

The 2/3-law is an example of a special case of the Kolmogorov-Obukhov law described in Proposition 6.3. This law is an experimental law, which states the following relation for the second order structure function $S_2(\ell)$,

$$S_2(\ell) = C \varepsilon^{2/3} \ell^{2/3}, \quad (6.6)$$

where ε is the mean energy dissipation per unit mass.

The 2/3-law can be restated in terms of the spatial energy spectrum $E(k)$ in the spatial frequency domain, and one then gets the experimental 5/3-law. Frisch [20] defines the energy

spectrum⁹ $E(k)$ using continuous ideal low-pass filtered velocities. In correspondence with Frisch [20], I define the energy spectrum of the scale-space velocity field $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$ as

$$E(k) \equiv \frac{d\mathcal{F}\{\varepsilon_{\vec{\sigma}}\}}{dk},$$

where $\mathcal{F}\{\varepsilon_{\vec{\sigma}}\}$ is the spatial Fourier transformation of the cumulative scale-space energy $\varepsilon_{\vec{\sigma}}$ described in Proposition 6.1,

$$\varepsilon_{\vec{\sigma}} = \frac{1}{2} \overline{|\vec{v}_{\vec{\sigma}}|^2}.$$

The 5/3-law can then be stated as

$$E(k) = C_{\text{Kol}} \varepsilon^{2/3} k^{-5/3} \quad (6.7)$$

where k is the spatial frequency and C_{Kol} is a dimensionless constant called the Kolmogorov constant. This law can, according to Frisch [20], be proved using the relation for random functions, which states that if the energy spectrum of the function is a power law,

$$E(k) \propto k^{-n},$$

then the second order structure function is also a power law,

$$S_2(\ell) \propto \ell^{n-1}.$$

If we use the β -model instead of the K41 cascade model, the above stated power laws will change exponents. When introducing the β -model, the proportionality for the p th order longitudinal structure function of the Kolmogorov-Obukhov law is changed to

$$S_p(\ell) \propto v_0^p \left(\frac{\ell}{\ell_0}\right)^{\zeta_p}$$

where $v_0 = v_{rms}$ is the root mean square velocity fluctuation and

$$\zeta_p = \frac{p}{3} + (N - D_H) \left(1 - \frac{p}{3}\right). \quad (6.8)$$

From this it is easily seen that the 2/3-law changes to

$$S_2(\ell) \propto \ell^{2/3+(N-D_H)/3}, \quad (6.9)$$

and the 5/3 scaling law of the energy spectrum therefore must change to

$$E(k) \propto k^{-5/3+(N-D_H)/3}. \quad (6.10)$$

If we assume that the turbulence in our experimental setup is homogeneous and isotropic we can conduct measurements in order to verify the 2/3-law of Eq. (6.9). Schröder [55] notes that the second order longitudinal structure function $S_2(\ell)$ can be written as,

$$S_2(\ell) = \overline{\left[\left(\vec{v}_{\vec{\sigma}}(\vec{x} + \vec{\ell}; \vec{\sigma}) - \vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) \right) \cdot \vec{\ell}/\ell \right]^2} = 2 \left[\overline{\left| \vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) \cdot \vec{\ell}/\ell \right|^2} - \Gamma(\ell; \vec{\sigma}) \right]. \quad (6.11)$$

⁹Also known as the power spectrum or the power spectral density.

We can for that reason find the second order longitudinal structure function of an optic flow field at a certain scale $\vec{\sigma}$ by calculating the mean square length of the velocity vectors and the correlation function $\Gamma(\ell; \vec{\sigma})$ of velocities projected onto the direction of the increment vector $\vec{\ell}/\ell$. The correlation function of the centred velocity field, $\overline{\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})} = 0$, is given by

$$\Gamma(\ell; \vec{\sigma}) = \overline{[\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma}) \cdot \vec{\ell}/\ell] [\vec{v}_{\vec{\sigma}}(\vec{x} + \vec{\ell}; \vec{\sigma}) \cdot \vec{\ell}/\ell]}. \quad (6.12)$$

I choose to vary the increment lengths ℓ and keep the directions fixed to the horizontal and vertical directions. I can then compute Eq. (6.11) of the velocity increments for each pixel, thereby getting an estimate of $S_2(\ell)$.

In order to verify the 5/3-law of Eq. (6.10) we need to compute the spatial energy spectrum from the spatial velocity field. In the case of an incompressible isotropic flow we can, according to Frisch [20], use the Wiener-Khintchine formula to calculate the spatial energy spectrum $E(k)$ as the Fourier transformation of the correlation function $\Gamma(\ell; \vec{\sigma})$, given by Eq. (6.12), of the velocity $\vec{v}_{\vec{\sigma}}(\vec{x}; \vec{\sigma})$ projected onto the increment direction $\vec{\ell}/\ell$,

$$E(k) = \int_0^{\infty} \Gamma(\ell; \vec{\sigma}) \exp(-ik \cdot \ell) d\ell. \quad (6.13)$$

Nielsen and Olsen [45] proposes another experiment using optic flow data, which is an investigation of change in the space filling of eddies in scale-space. In Sec. 6.3 I have introduced Richardson's cascades and the reduction of space filling incorporated in the β -model. Nielsen and Olsen [45] assumes that eddies corresponds to singularities of the whirl type in the normal flow field and that the space filling of the corresponding eddies is given by their area defined by the squared spatial scale σ_s at which they appear. We are therefore able to estimate the space filling of eddies at separate scales by counting the singularities of the flow field.

In this thesis I will assume that eddies correspond to all possible types of singularities. I use the winding number of the flow field as defined in Definition 4.5 as a method for detection of singular points in order to count singularities in the flow field at different scales. I then conjecture that we can compute the scaling exponent of the fraction of space p_ℓ in which active eddies reside, by restating Eq. (6.5) in terms of the number of singularities $N_S(\sigma_s)$ at the spatial scale σ_s ,

$$p_\ell = \sigma_s^2 N_S(\sigma_s) \propto \sigma_s^{2-D_H}. \quad (6.14)$$

In this thesis I concentrate on studying turbulent motion as projected onto a 2 dimensional image plane, which only allows me to study the scaling exponent of the fraction of area which active eddies occupy. The topological dimension N of Eq. (6.5) therefore reduces to $N = 2$ in Eq. (6.14).

I conjecture that the number of singularities at different scales should behave according to the above stated law, Eq. (6.14).

6.5 Summary

In this chapter I have explained that fluid dynamics is centered around the Navier-Stokes equations. I have explained that turbulence obey certain scaling laws in a statistical sense. The ensemble average of velocity increments of turbulent flow changes proportionally with the length of the increment separation vector. Furthermore, the size of eddies at certain scales can be modelled by the so-called cascade models, of which I have introduced the K41 model and the

β -model. I have also derived a scale-space energy budget by using linear Gaussian scale-space theory. I ended this chapter by proposing different measurements of the scaling properties, which is possible using optic flow methods and the linear Gaussian scale-space representation.

Chapter 7

Turbulence in Optical Flow Fields

In this chapter I will examine the possible usage of optic flow methods in empirical studies of turbulence. I have performed the so-called Faraday experiment, which is discussed in Sec. 7.1, and thereby obtained a set of image sequences, which is described in Sec. 7.2.

I have estimated the zeroth order normal flow fields and zeroth order optical flow fields of these image sequences using Algorithm 4.2 and Algorithm 4.4. I choose to use these algorithms for estimation of the optical flow fields in the light of the conclusion of Chapter 5 and because no assumptions has been made about the physical nature of the flow, except for the choice of flow model order. By doing so, I insure that I do not introduce a bias into the results toward a certain physical behaviour.

I will analyse the data sets in order to see if evidence of the experimental laws discussed in Sec. 6.4 can be found and the results of this investigation are presented in Sec. 7.3.

7.1 The Faraday Experiment

The Faraday experiment is designed for the investigation of turbulence in fluids, [55]. The setup of the Faraday experiment is that a small dish with a diameter of approximately 10 cm and a hight of about 2 cm is mounted upon a vibration exciter or oscillator, which is able to oscillate the dish horizontally (see drawing of the experimental setup in Fig. 7.1). A fluid is poured into the dish such that the surface of the fluid is approximately 1 cm above the bottom of the dish. By varying the oscillation frequency and amplitude of the exciter it is possible to induce and control the degree of turbulence in the flow of the fluid. Fluids with different viscosity could be used in this experiment, thereby changing another of the control parameters of the flow and indirectly changing the Reynolds number, as described by Theorem 6.1, of the flow.

The standard method of observing turbulence in the Faraday experiment is to add a trace element to the fluid. The trace element visualises the turbulent flow, as long as the trace element itself does not interfere and change the flow in an approximative sense. Schröder [55] uses mushroom spores as the trace element and also dye or tiny plastic particles has been used as trace element by other authors. Schröder uses a CCD¹ video camera and a VHS video recorder to record the movement of the particles. The particles are lit by a small directed spot light. Schröder digitize the video and tracks the particles by using a PTV² tracking program, which initially selects four arbitrary particles that is tracked into the next video frame by searching

¹Charge-Coupled Device.

²Particle Tracking Velocimetry.

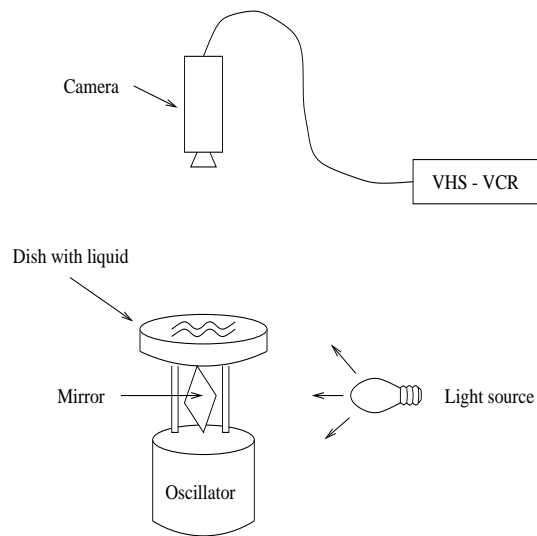


Figure 7.1: Schematic drawing of the experimental setup of the Faraday experiment. A flow is induced into the fluid by the oscillator or vibration exciter. The fluid is lit from below by a spot light through a tilted mirror and up through the dish. The movement is captured by a gray value CCD camera and recorded on a VHS video recorder.

small boxes around the particles. The program can hereby deliver a set of particle traces from the digital video sequence.

In this thesis I have chosen to conduct the Faraday experiment as described by Schröder [55], but instead of tracking particles I have chosen to investigate the mixing of two fluids. I choose to fix the viscosity and use a mixture of water and milk, henceforth known as the water-milk mixture. The milk functions as a trace element and when it mixes with the water it visualizes the turbulent flow of the mixing of the two fluids. Another benefit of using milk as a trace element is that by doing so I am able to track a region of the flow and to determine the singularity structure of the flow, contrary to tracking the motion patterns of particles. I hereby get another situation of measurement than that described by Schröder [55].

I use a gray value CCD video camera and, due to the absence of a fast frame grabber, a VHS video recorder. Selected parts of the video tape have been digitized frame by frame using a frame grabber. The water-milk mixture is lit from below by a small directed spot light. The spot light is positioned at the side of the dish and it projects the light onto a mirror that reflects the light up through the bottom of the dish (see drawing of the experimental setup in Fig. 7.1). The mirror is tilted in order to reflect the light coming from the side of the dish. Furthermore, the mirror is not mounted on the exciter in order to prevent oscillation of the mirror.

7.2 Collection of Experimental Data

As explained in the previous section I have recorded a video film of the turbulence exhibited in the mixing of two fluids. The video film is digitized, in such a way that I have a separate image for each video frame. Better representations of image sequences exist, but the choice of representation is not important in the context of this thesis.

The digital camera used for recording the image sequences delivered 25 frames per second and had a resolution of 512×512 pixels. This is also the resolution at which the images of the

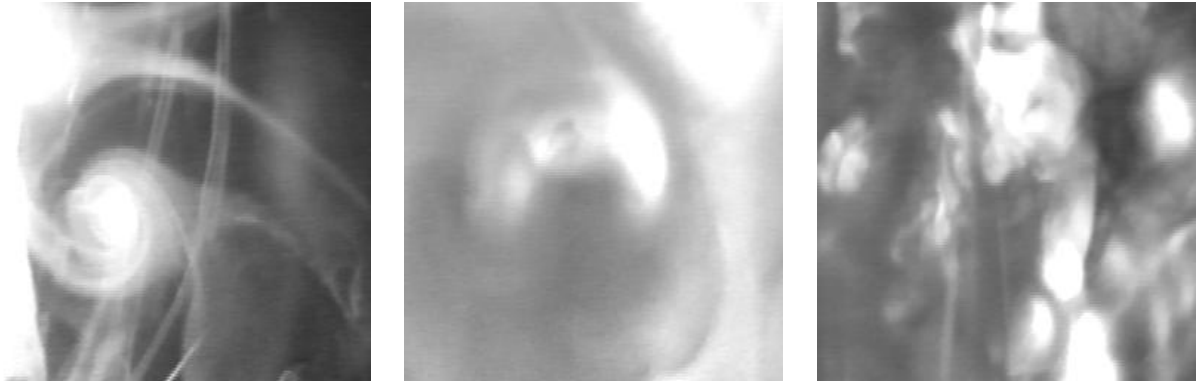


Figure 7.2: Examples of frame number 7 of image sequences E3/serie1, E4/serie2, and E5/serie1 obtained from the Faraday experiment.

sequence were grabbed. Only movement in approximately half of the dish of the experimental setup was captured by the camera. For that reason, regions captured by the camera have an area of approximately 5×5 cm, which gives an inter-pixel distance of 0.01 cm. Regions of the obtained images were selected and cropped into images of 256×256 pixels, thereby reducing the area of interest to approximately 2.5×2.5 cm. This reduction was done in order to make sure that only the water-milk mixture were within the image and to reduce the amount of computations needed per image. Furthermore, each image sequence consists of 16 frames and because I use a 25 Hz camera there is $1/25$ seconds between each frame.

Examples of frames from some of the image sequences can be found in Fig. 7.2 and examples of frames from all of the image sequences can be found in Appendix C.

The oscillation frequencies of the exciter used for the different image sequences were in the interval of 6 Hz to 20 Hz, but the knowledge of the actual frequency will not be used in this evaluation of turbulence in optical flow fields.

Some of the image sequences exhibits temporal noise, like periodic surface reflections of light. Because it is periodic, we could try to remove the noise by temporal low-pass filtering with for instance a Gauss filter, but this is actually the same as increasing the inner temporal scale of the image sequence. This benefit arise from the semi-group property of linear Gaussian scale-space, as described in Sec. 3.1.2. I therefore choose to select, by visual inspection, an approximately lower bound on the temporal inner scales of the image sequences. The names of all the used image sequences and the chosen lower bounds can be found in Table 7.1.

For the investigation of turbulence I have also used an image sequence of smoke in a pigsty. The smoke is introduced into the pigsty and then dispersed by the draught in the staples. The image sequence is captured by illuminating the smoke by a laser sweeping through the staples and then recording it with a video camera. The pigsty sequence has previously been used by Nielsen and Olsen [45] in their investigation of the structure of singularities in optic flow. This sequence consists of 16 images of each 256×256 pixels. Sample frames from the pigsty sequence can be viewed in Fig. 7.3.

Image sequence	σ_t
E1/serie2	1.0
E3/serie1	2.0
E3/serie3	2.0
E3/serie4	2.0
E3-2/serie2	1.0
E4/serie2	1.0
E5/serie1	1.0
E5/serie2	1.0
Vhs/vhs7	0.5
Vhs/vhs8	0.5

Table 7.1: Names of all the Faraday image sequences and corresponding lower bounds on the temporal scales of image sequences under which they exhibit temporal periodic noise. The reason for the abrupt change in the numbering of the image sequences, is that I have selected these from a much larger set of data.

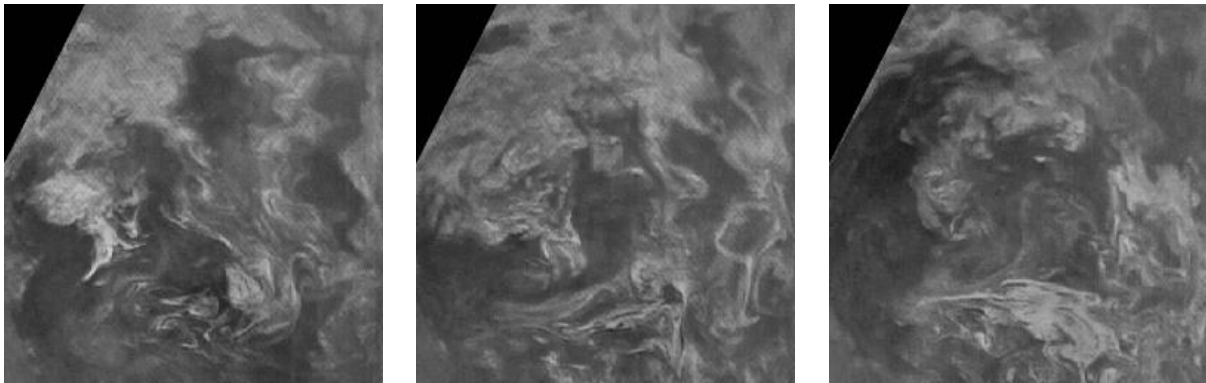


Figure 7.3: The first, eighth, and the sixteenth images from the pigsty sequence. The pigsty sequence depicts smoke/air transport in a pigsty.

7.3 Results

I will analyse 2-dimensional normal and optical flow fields in order to investigate whether evidence of the laws discussed in Sec. 6.4 can be found in these flow fields. With the results of Chapter 5 in mind I choose to base my investigation on zeroth order normal flow fields produced by Algorithm 4.2 and zeroth order optical flow fields produced by Algorithm 4.4, as well as optical flow fields produced by the scale selection algorithm Algorithm 4.5.

Note that the spatiotemporal and integration scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$ used for the estimation of normal and optical flow fields, are chosen in order to obtain the best results of all the measurements described below. Furthermore, the same configuration of scales are used in the different measurements, where appropriate, in order to be able to compare the results for a given image sequence.

I assume that the underlying turbulent flow of the image sequences described in the previous section is homogeneous and isotropic. This is a fair assumption, because the water-milk sequences are obtained at approximately the center of the dish and are for that reason away

from the barrier of the flow. The pigsty sequence is also obtained in a region away from the barrier of the flow and can for that reason also be assumed to be homogeneous and isotropic. By assuming homogeneity and isotropy I am able to investigate the Kolmogorov-Obukhov law described in Proposition 6.3, especially the scaling properties of the second order longitudinal structure function $S_2(\ell)$.

I also assume that the scaling properties of turbulence in the experimental setups of the Faraday experiment and the pigsty experiment can be modelled by the β -model discussed in Sec. 6.3. For that reason the three experimental laws described in Sec. 6.4 all predicts the fractal dimension D_H of the space filling of eddies in the flow. When verifying these experimental laws empirically on a turbulent flow, we would then expect the fractal dimension predicted by the three laws to be approximately equal. We can therefore compare the results obtained from different measurements designed to verify separately each of the three experimental laws, by comparing the predicted fractal dimension D_H . As stated earlier I will, in this thesis, analyse 2-dimensional normal and optical flow fields and we would therefore expect to obtain estimates of the fractal dimension D_H of eddies in the range $D_H \in [1; 2]$. It is not a trivial task to predict the actual values of the fractal dimension D_H , but a reasonable assumption is that the flow could be modelled by classical Brownian motions, which in the 2-dimensional case has a fractal dimension of $D_H = 1.5$. This assumption has also been used among others by Mandelbrot [40, 39] in his investigations of the fractal sets in nature. Classical Brownian motions has also been proposed as models for the intensity patterns of images of natural scenes, [52, 15, 29].

In order to verify the 2/3-law, Eq. (6.9), I have computed the second order longitudinal structure function $S_2(\ell)$ of optical flow fields produced by scale selection, by using Eq. (6.11) for velocity increments of increasing separation size ℓ in the directions of the two coordinate axes. When computing the second order longitudinal structure function $S_2(\ell)$ it is only reasonable to use optical flow fields. If we choose normal flow for this computation, we would have to handle the fact that we only have knowledge of one component of the true flow field. We could try to bypass this problem by using increment directions along the direction of the normal flow field.

According to Eq. (6.8) we would expect that the scaling exponent $\gamma_{2/3}$ of the second order structure function $S_2(\ell) \propto \ell^{\gamma_{2/3}}$ of the 2-dimensional optical flow fields would be in the interval $\gamma_{2/3} \in [2/3, 1]$. We can estimate the scaling exponent $\gamma_{2/3}$ by finding the slope of the log-log graph of $S_2(\ell)$. The fractal dimension can, by use of Eq. (6.8), be obtained from the scaling exponent $\gamma_{2/3}$ of the second order longitudinal structure function $S_2(\ell)$ by

$$D_H = -3\gamma_{2/3} + 4. \quad (7.1)$$

The flow of the image sequences all have an inner scale equal to the spatiotemporal and integration scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$ on which the flow field is estimated. This means that when calculating the second order structure function $S_2(\ell)$ we should in general only use longitudinal velocity increments $\delta v_{\parallel}(\ell)$, which has a separation larger than the inner integration scale ϖ of the flow, $\ell > \varpi$. In this investigation of scaling properties in turbulence I have chosen to include increments below the inner scale in order to visualise the inner scale.

The integral scale of the turbulent flow is in general not equal to the outer scale of the scale-space in which the optical flow field is estimated, which means that there also is an upper limit ℓ_0 on the size of the separation ℓ . We cannot predict this upper limit ℓ_0 without an analysis of the flow, but it is, in the case of the Faraday image sequences, obviously smaller than or equal to the outer scale of the optical flow field scale-space.

The results of calculating the scaling exponent of the second order longitudinal structure function $S_2(\ell)$ for all the image sequences can be found in Table 7.2 and log-log plots of $S_2(\ell)$

Sequence	$\gamma_{2/3}$	D_H
E1/serie2	0.73	1.80
E3/serie1	0.90	1.31
E3/serie3	0.64	2.09
E3/serie4	0.47	2.59
E3-2/serie2	0.69	1.92
E4/serie2	1.06	0.83
E5/serie1	0.83	1.52
E5/serie2	0.95	1.14
Vhs/vhs7	0.94	1.19
Vhs/vhs8	0.83	1.51
Pigsty	0.84	1.49

Table 7.2: The scaling exponent $\gamma_{2/3}$ of the second order longitudinal structure function $S_2(\ell)$ of scale selected zeroth order optical flow fields. For all image sequences I used the increments $\ell \in [15, 19]$, measured in pixels, for estimation of the slope. The fractal dimension D_H of eddies were computed using Eq. (7.1).

can be found in Fig. 7.4. In the log-log plots we can clearly see the integral scale, which manifests itself in a sharp cut-off at large ℓ values. The inner scale is visible for some of the image sequences as a curve with a small slope at small increments $\ell < \varpi$, which change drastic above the integration scale, $\ell > \varpi$. In the estimation of $\gamma_{2/3}$ I used ℓ values between these visible inner and outer scales. The results in Table 7.2 show that the slopes $\gamma_{2/3}$ is generally estimated to be with in the theoretical range $\gamma_{2/3} \in [2/3, 1]$. But in the case of sequences E3/serie3, E3/serie4, and E4/serie2 the estimated slope is clearly out of range. The scaling exponents $\gamma_{2/3}$ of all the sequences vary in the full range of expected $\gamma_{2/3}$ values, but some of the results seems to correspond to the assumption that the flow can be modelled by Brownian motion, because they are close to $D_H \approx 1.5$.

In order to verify the 5/3-law of Eq. (6.10) I have computed the energy spectrum $E(\omega_\ell)$ of optical flow fields for all of the image sequences by using Eq. (6.13). I have calculated the optic flow at fixed scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$ and then computed the energy spectrum by varying the size of the increments ℓ in the directions of the two coordinate axes.

According to Eq. (6.10) we would expect that the scaling exponent $\gamma_{5/3}$ of the energy spectrum $E(\omega_\ell) \propto \ell^{\gamma_{5/3}}$ of the 2-dimensional flow fields, should be in the interval $\gamma_{5/3} \in [-5/3, -4/3]$. We can estimate the scaling exponent $\gamma_{5/3}$ by finding the slope of the log-log graph of $E(\omega_\ell)$. The fractal dimension can, by use of Eq. (6.10), then be obtained from the scaling exponent $\gamma_{5/3}$ of the energy spectrum $E(\omega_\ell)$ by

$$D_H = -3\gamma_{5/3} - 3. \quad (7.2)$$

A Matlab function for computing the energy spectrum can be found in Appendix D.2.2.

The results of the measurement of the energy spectrum can be viewed in Table 7.3 and a log-log plot of the energy spectrum $E(\omega_\ell)$ can be found in Fig. 7.5. As in the 2/3-law measurements of Fig. 7.4, we can in the log-log plot of the energy spectrum $|E(\omega_\ell)|^2$ observe the inner scale as a flat curve at small values of $\omega_\ell < \varpi$. The outer scale or integral scale is also visible for some of the image sequences. It can be seen that the range of reasonable values of $|E(\omega_\ell)|^2$ is larger than the range of reasonable values of $S_2(\ell)$ in Fig. 7.4. Frisch [20] notes that this behaviour is also seen in other more conventional measurements of the second order longitudinal structure

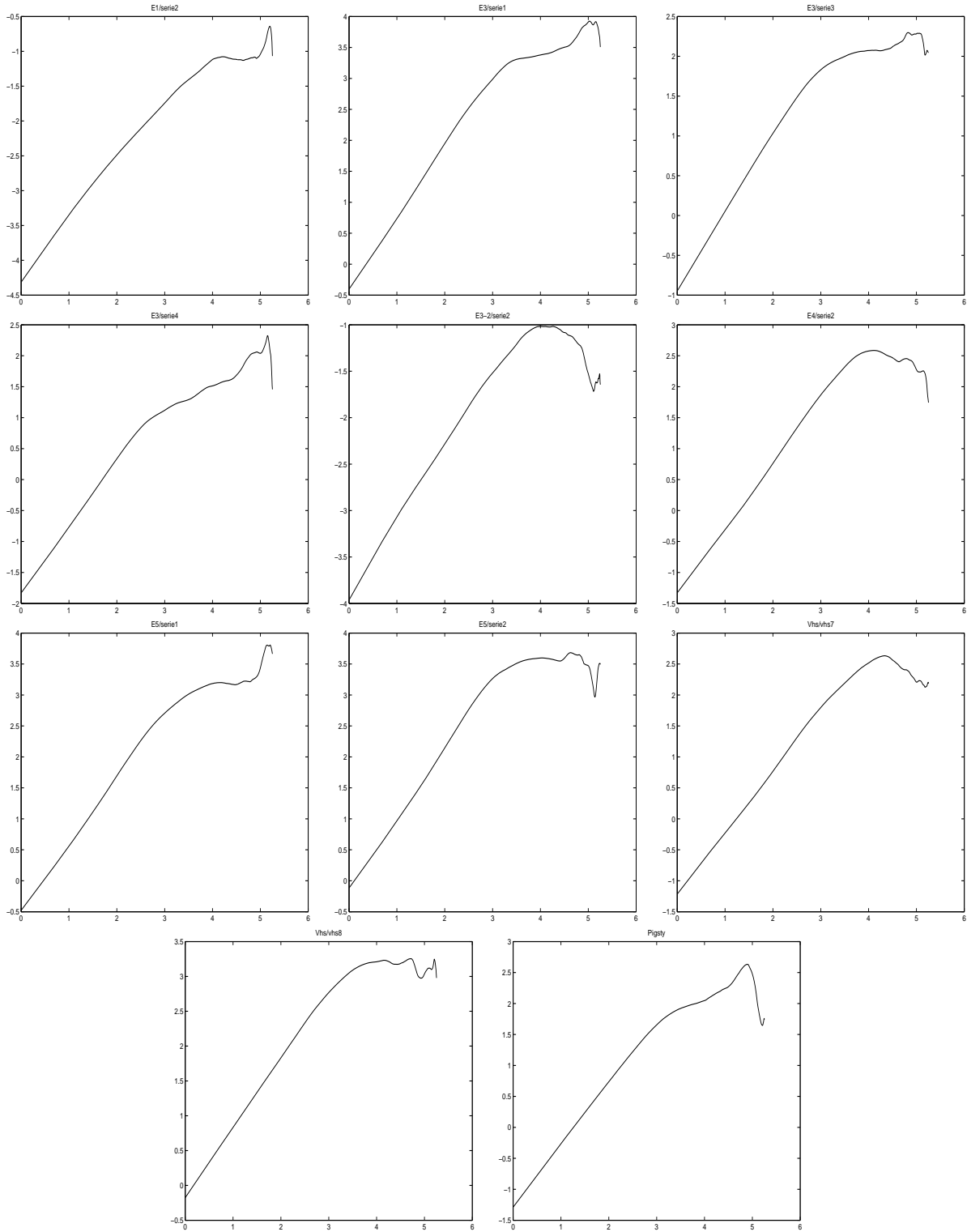


Figure 7.4: Scaling of the second order longitudinal structure function $S_2(\ell)$ of scale selected zeroth order optical flow fields based on zeroth order normal flow. The graphs show the scaling of $\log(S_2(\ell))$ as a function of $\log(\ell)$. The graphs are ordered such that the graph in the upper left corner corresponds to the E1/serie2 image sequence, and the graphs going from left to right and down correspond to the ordering of the image sequences in Table 7.2.

Sequence	$\gamma_{5/3}$	D_H
E1/serie2	-1.04	0.12
E3/serie1	-1.12	0.36
E3/serie3	-1.09	0.26
E3/serie4	-1.15	0.44
E3-2/serie2	-1.06	0.18
E4/serie2	-1.02	0.06
E5/serie1	-1.02	0.07
E5/serie2	-1.06	0.19
Vhs/vhs7	-1.04	0.12
Vhs/vhs8	-1.10	0.30
Pigsty	-1.05	0.14

Table 7.3: Scaling exponent $\gamma_{5/3}$ of the energy spectrum $E(\omega_\ell)$ of scale selected zeroth order optical flow fields. The values $\omega_\ell \in [8, 33]$ were used for estimation of the slope $\gamma_{5/3}$ and the fractal dimension D_H were computed using Eq. (7.2).

function $S_2(\ell)$ and energy spectrum $E(\omega_\ell)$. The results found in Table 7.3 are very disappointing because they are all underestimated by far. Frisch [20] notes that using the Fourier transform of the spatial autocorrelation, Eq. (6.13), as a means of estimating the power spectrum of a turbulent flow is very unstable in the case of a flow which can be said to have stationary³ increments but without being stationary it self. In that case, only the second order longitudinal structure function is valid, which by comparing the results in Table 7.3 with those found in Table 7.2 might lead us to the conclusion that the used image sequences fall into this category and has in fact stationary increments but without being stationary them self.

I have also tried to compute the energy spectrum by varying the spatial scale of normal flow instead of varying the size of the increments ω_ℓ . For each image sequence I have computed the normal flow at a set of different scales and then calculated the energy spectrum of the normal flow at each scale. I used the spatial scale σ_s as the increment size, $\omega_\ell = \sigma_s$. I choose to use the zeroth order normal flow produced by Algorithm 4.2.

The results of the measurements of the scale-space energy spectrum $E(\sigma_s)$ can be found in Table 7.4 and log-log plots of $|E(\sigma_s)|^2$ can be found in Fig. 7.6. From the results in Table 7.4 and Fig. 7.6 we can conclude that this method of measuring the energy spectrum is rather sensitive to noise, because the most of the estimated slopes are out of the range expected for this experiment. This also strengthen the conclusion made above that the investigated flows are not stationary.

As stated in Chapter 6, I conjecture that through Eq. (6.14) we can estimate the scaling properties, $p_\ell = \sigma_s^2 N_S(\sigma_s)$, of space filling of eddies in a flow field, by counting the singularities of that flow field at different spatial scales σ_s . I count singularities by finding all pixels in the flow field at which the winding number, as defined in Definition 4.5, is zero.

Inspired by Nielsen and Olsen [45] I have chosen to use the normal flow fields in the measurement of space filling of eddies, because the normal flow is more robust than the optic flow at higher scales.

According to Eq. (6.14) we would expect that the scaling exponent γ of the space filling p_ℓ

³Stationary in the statistical sense.

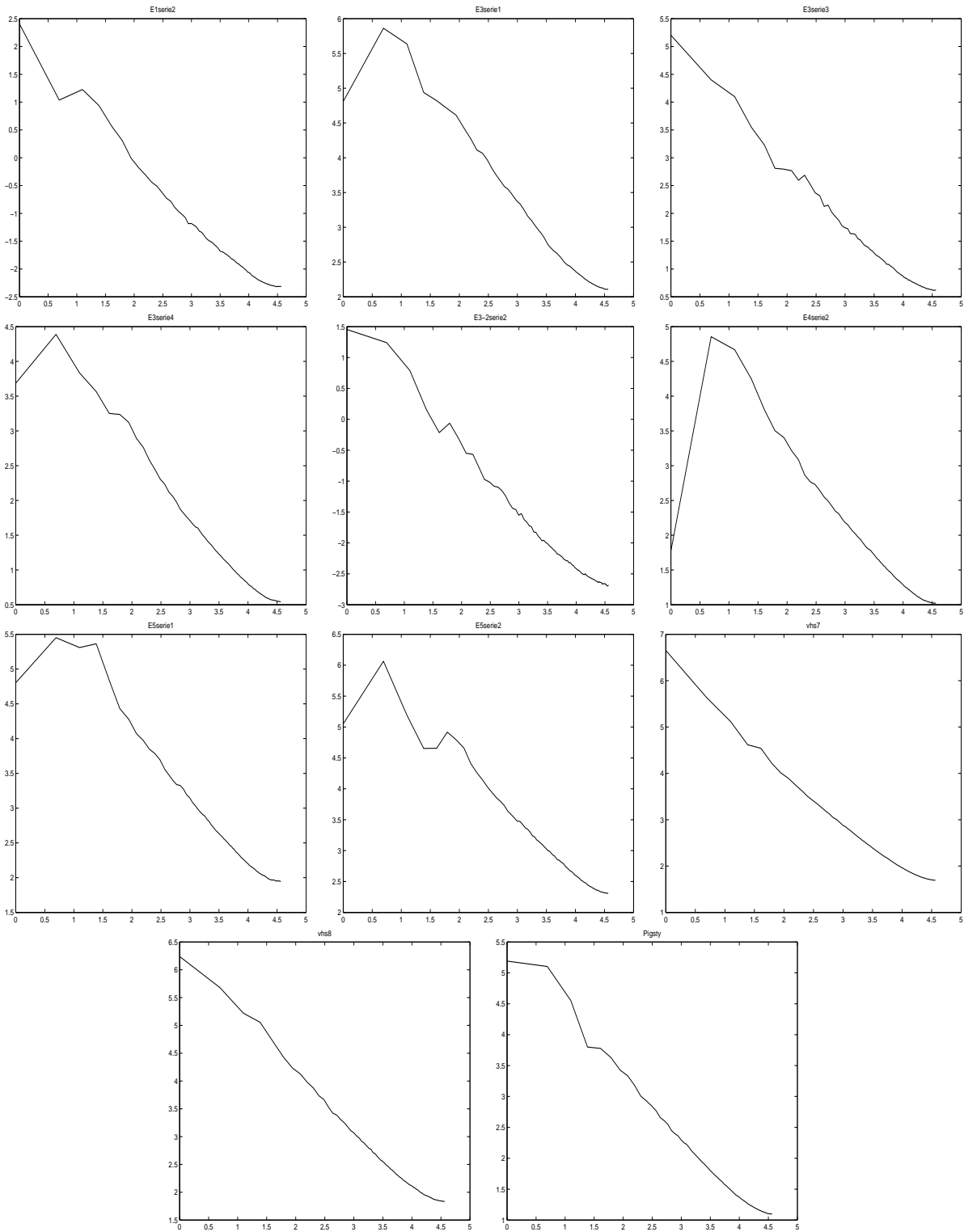


Figure 7.5: Energy spectrum $E(\omega_\ell)$ of optical flow fields. The figure shows graphs of $\log(|E|^2)$ as a function of $\log(\omega_\ell)$. The graphs are ordered, such that they correspond to the ordering of the image sequences in Table 7.3, with the graph corresponding to the E1/serie2 sequence in the upper left corner.

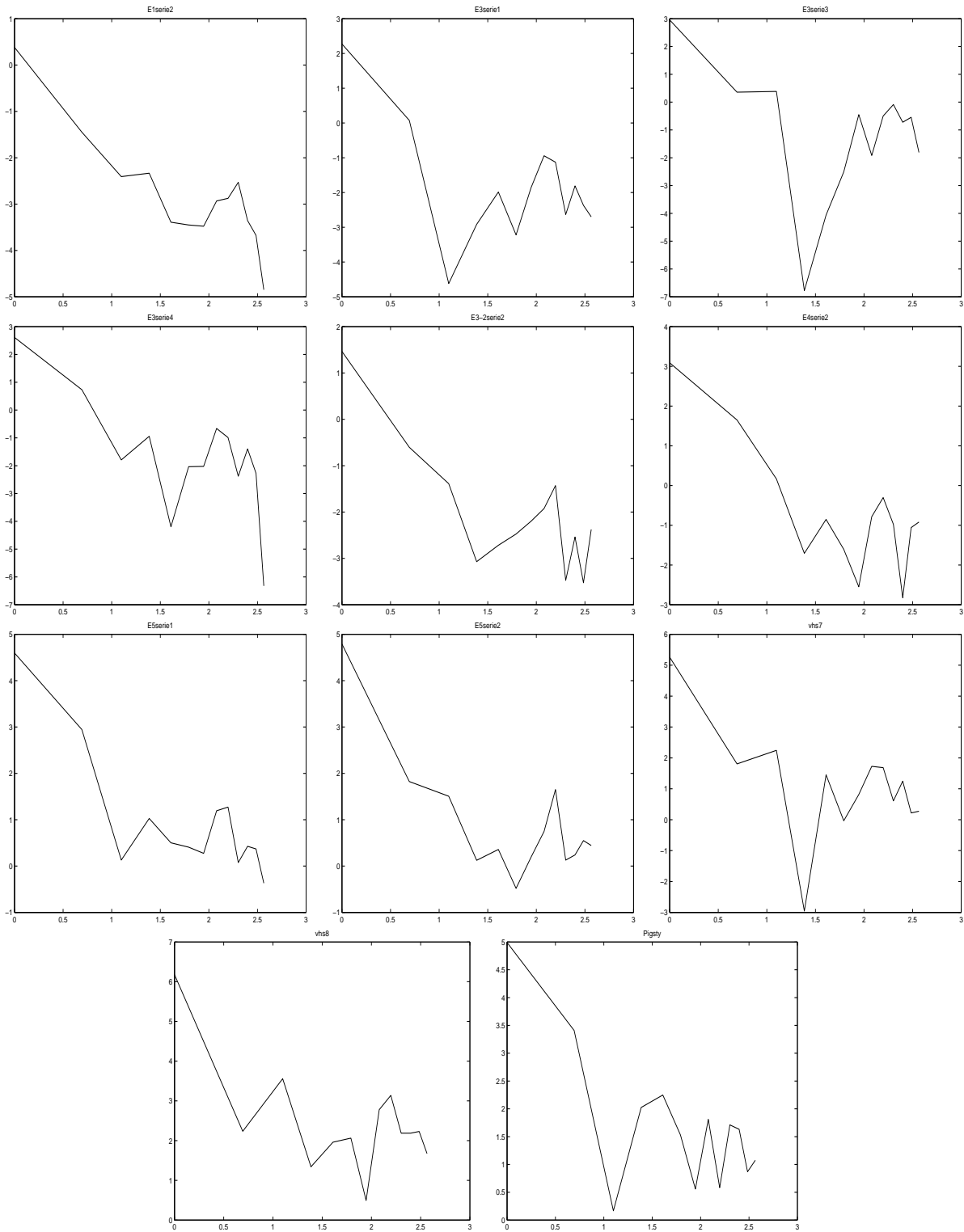


Figure 7.6: The scale-space energy spectrum $E(\sigma_s)$ of zeroth order normal flow fields. The figure shows graphs of $\log(|E|^2)$ as a function of $\log(\sigma_s)$. The graphs are ordered, such that they correspond to the ordering of the image sequences in Table 7.4 with the E1/serie2 sequence at the upper left corner.

Sequence	$\gamma_{5/3}$	D_H
E1/serie2	-1.44	1.31
E3/serie1	-1.13	0.39
E3/serie3	-0.92	-0.25
E3/serie4	-1.96	2.87
E3-2/serie2	-1.40	1.19
E4/serie2	-1.64	1.91
E5/serie1	-1.45	1.34
E5/serie2	-1.31	0.93
Vhs/vhs7	-1.21	0.62
Vhs/vhs8	-1.09	0.26
Pigsty	-1.22	0.67

Table 7.4: Scale-space energy scaling using zeroth order normal flow fields. Scales used are $\sigma_s \in [1, 8]$ and $\sigma_t = 2.0$ for every sequence. The spatial scales σ_s are exponentially growing with 13 samples, which is the natural choice for linear Gaussian scale-space. The fractal dimension were computed using Eq. (7.2). The three middle frames of the image sequences were used for the estimation of the scaling exponent $\gamma_{5/3}$.

of eddies would be in the interval $\gamma \in [0, 1]$. From Eq. (6.14) we see that the fractal dimension D_H can be computed from the scaling exponent γ , through

$$D_H = 2 - \gamma. \quad (7.3)$$

We can estimate the scaling exponent γ by finding the slope of the log-log graph of p_ℓ . The results of the measurement of space filling $p_\ell = \sigma_s^2 N_S(\sigma_s)$ can be found in Table 7.5 and a log-log plot of the space filling p_ℓ can be found in Fig. 7.7. From Table 7.5 it can be seen that the computed scaling exponents γ are fairly reasonable, because most of the results are within the expected range, expect for the exponent γ of the E3-2/serie2, E4/serie2, and E5/serie1 sequences. From the log-log plots it can be seen that the results for some of the sequences are not that accurate, because the value of p_ℓ varies abruptly with the scale σ_s . Furthermore, the scaling exponents γ for all the image sequences vary in the middle part of the expected range, $\gamma \in [0, 1]$, which indicates a fractal dimension of about $D_H \approx 1.5$.

From the results found in Table 7.2 – Table 7.4 it can be seen that the energy scaling exponent $\gamma_{5/3}$ is generally poorly estimated compared to the scaling exponent $\gamma_{2/3}$ of the second order longitudinal structure function. The results of the measurement of space filling in Table 7.5 seems in general to be better than both energy and structure function measurements.

Some of the image sequences, like E3/serie3, E3-2/serie2, and E4/serie2 seem to result in bad measurement independent of the type of measurement, which can be explained by either image noise or simply that the image sequences used does not include enough evidence of turbulence. That is, it might have been a mistake to choose these image sequences for the measurements discussed here.

As stated earlier we would expect the fractal dimension D_H estimated using the different types of measurements to be approximately equal, under the assumption that the β -model is valid for this type of experiment. This does not seem to be the case, because the fractal dimension deviate for all image sequences for the different types of measurements. This indicates that the β -model might not be a correctly chosen model for the mixing of two fluids in the Faraday experimental setup. One could also conjecture that the method used in this thesis for

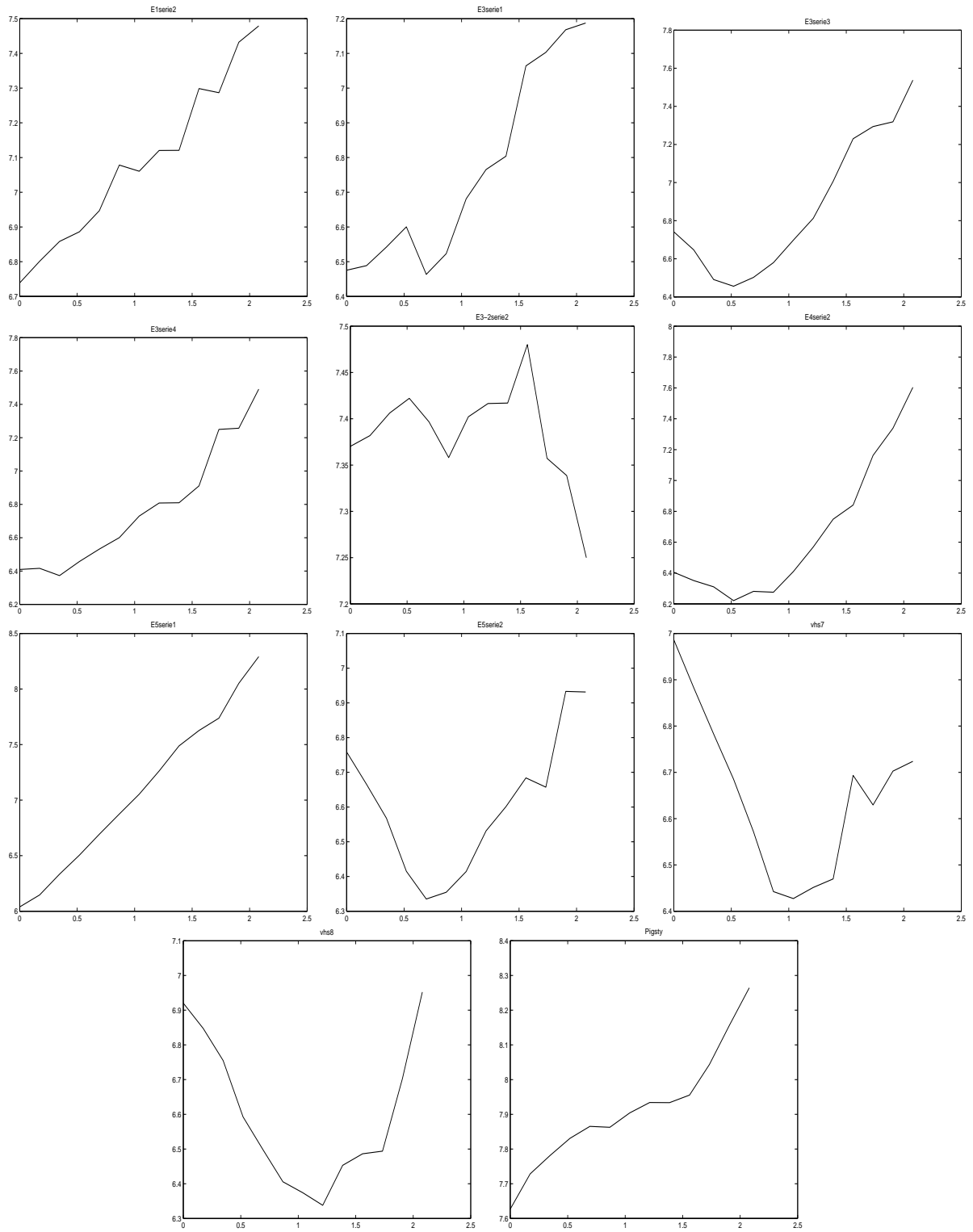


Figure 7.7: Space filling of eddies in zeroth order normal flow as a function of the spatial scale σ_s . The figure shows plots of $\log(\sigma_s^2 N_s)$ as a function of $\log(\sigma_s)$. The graphs are ordered, such that they correspond to the ordering of the image sequences in Table 7.5, where the E1/serie2 sequence can be found in the upper left corner.

Sequence	γ	D_H
E1/serie2	0.35	1.65
E3/serie1	0.57	1.43
E3/serie3	0.73	1.27
E3/serie4	0.61	1.39
E3-2/serie2	-0.03	2.03
E4/serie2	1.09	0.91
E5/serie1	1.08	0.92
E5/serie2	0.46	1.54
Vhs/vhs7	0.32	1.68
Vhs/vhs8	0.63	1.37
Pigsty	0.24	1.76

Table 7.5: Scaling of the space filling of eddies in zeroth order normal flow. The spatial scales used are $\sigma_s \in [1.0, 8.0]$ and the temporal scale used is $\sigma_t = 2.0$. The spatial scales σ_s are exponentially growing with 13 samples. The fractal dimension D_H were computed using Eq. (7.3). Three middle frames were used in the estimation of the scaling exponents.

investigation of turbulence is not valid, but I believe that evidence of the contrary has been given here.

The lack of validity of the β -model could be caused by the fact that the image sequences includes the mixing of the water and milk at different depths of the dish. This is a quite complex phenomenon, because we actually see the interaction of the turbulence at different scales at the same time.

Looking at the graphs of Fig. 7.4 – Fig. 7.7 we see that scaling laws clearly exists for the second order longitudinal structure function $S_2(\ell)$, the energy spectrum $E(\omega_\ell)$, and the space filling of eddies. We find large ranges in these graphs where a linear relation exists for the log-log plots, which indicates that the method used in this thesis might be a reasonable choice for an investigation of turbulence.

It would be interesting to investigate the validity of other types of cascade models using optic flow and linear Gaussian scale-space. Furthermore, the version of the Faraday experiment used in this thesis could be altered in order to get a setup, which is easier to analyse. We could for instance use milk and another visible trace element like dye or very small coloured particles, thereby reduce the effect of measuring the mixing of the fluids at different depths.

7.4 Summary

I have in this chapter examined the correspondence of the scaling properties of normal and optic flow and the scaling properties of the underlying physical fluid flow as introduced in Chapter 6. In order to obtain the necessary data needed for this evaluation I have performed the so-called Faraday experiment and thereby captured image sequences of the mixing of water and milk.

In this chapter I measured the second order longitudinal structure function and the energy spectrum by using scale selected zeroth order optical flow fields. Furthermore, I measured the energy spectrum by using a scale-space of zeroth order normal flow fields. I also measured the space filling of eddies in turbulence by counting singular points in a scale-space of normal flow fields.

I assumed that the β -model was a valid model for the turbulence of the mixing of water and milk and for that reason the different types of measurements predicts a fractal dimension of the space filling of eddies.

A comparison of the results of the different measurements for each image sequence shows that the fractal dimension estimated by the different measurements does not correspond. This indicates that the β -model is not valid for the turbulence exhibited in the mixing of water and milk. But scaling laws seem to exist for the second order longitudinal structure function, the energy spectrum, and the space filling of eddies, which indicates that it seems promising to investigate turbulence by using optical flow fields and the linear Gaussian scale-space representation.

Chapter 8

Conclusion

In this thesis I have analysed the usage of linear Gaussian scale-space theory and optic flow methods in the investigation of turbulence. For that reason I have introduced the theories of linear Gaussian scale-space, optic flow methods, and turbulence and tried to relate these.

I have discussed and implemented two existing methods for extraction of the optical flow field, proposed by Maurizot et al. [41] and Florack et al. [16, 17, 44, 18].

The method proposed by Maurizot et al. [41] and described in Sec. 4.2 allows us to extract the optic flow from a single image. This is possible under the assumption that the sought flow is perpendicular to the image gradient. I have altered the algorithm by introducing scale-space image derivatives and an integration scale-space of these derivatives. I included this method in order to show other methods used in conjunction with the examination of turbulence, but also because it is a peculiar method, and for that reason I wished to examine its functionality.

The method proposed by Florack et al. [16, 17, 44, 18] is a framework of optic flow methods utilising linear Gaussian scale-space and different so-called gauge conditions, which constraint the possible flow fields. In this method it is assumed that the flow can be modelled by a polynomial expansion of a chosen order. I have described the normal flow gauge condition that allows us to compute the normal flow using the method proposed by Florack et al. [16, 17, 44, 18].

In this thesis I have also proposed a new method for extraction of optic flow using linear Gaussian scale-space, which was described in Sec. 4.4. It is assumed that the optical flow field can be modelled by a polynomial expansion of some order and that the underlying normal flow field is known. The flow model parameters of the polynomial expansion are estimated by minimising the difference between the normal flow and the normal component of the optic flow. A so-called integration scale-space is used as an integration window function under which the flow model parameters are computed. The uncertainty of the underlying normal flow field is used as a weighting function under the integration scale-space, in order to let points with well estimated normal flow dominate the direction of the optic flow. The normal flow field used with this method can be obtained by any method as long as an uncertainty measure of the normal flow is computed.

I have used the normal flow, computed by using the method by Florack et al. [16, 17, 44, 18], in conjunction with the optic flow method proposed in this thesis. I used a spatiotemporal scale-space representation for this purpose. I have therefore in Sec. 4.4.3 proposed a joint scale selection mechanism, based on ideas by Niessen and Maas [47], Nielsen et al. [43], and Weber and Malik [57]. I constructed an error measure function consisting of the uncertainty measure and the model error of the estimated optic flow. The scale selection mechanism selects the flow components at the set of spatiotemporal integration scales for which the error measure is

minimal.

In Chapter 5 I evaluated the performance of the normal flow method by Florack et al. [16, 17, 44, 18] and the optic flow method proposed in this thesis.

I have evaluated the normal flow method by Florack et al. using the zeroth and first order normal flow model, and I find that this method compare well to other methods. The zeroth order normal flow model seems to perform better than the first order normal flow model for both translational and divergent motion. The reason for this discrepancy can be found in the lack of accuracy of the higher order image derivatives used in the first order model.

I have also evaluated the zeroth and first order model optic flow produced by the optic flow method proposed in this thesis, using both the zeroth and first order normal flow produced by the Florack et al. [16, 17, 44, 18] method. I have compared the results with other optic flow methods. The main conclusion of the evaluation of the optic flow method is that it in general perform very well compared to other methods. The choice of method for normal flow computation is crucial, which can be seen in the very poor results based on first order normal flow.

In Chapter 6 I presented selected parts of the theory of turbulence and proposed a set of experimental laws that can be validated using optic flow methods and the linear Gaussian scale-space representation.

In Chapter 7 I described the Faraday experiment, which I have conducted in order to obtain a real data set of turbulent flow. I have used this data set as the basis for my validation of experimental laws of turbulence using optic flow methods in conjunction with linear Gaussian scale-space. I used both normal and optic flow computed by using the Florack et al. [16, 17, 44, 18] method and the method proposed in this thesis. These methods do not make any assumptions about the nature of the underlying flow, except that the flow can be modelled locally by a polynomial expansion of a chosen order, and for that reason does not introduce a bias toward a certain physical behaviour.

By using optic flow methods we analyse a section of the motion induced by the turbulence, instead of, for instance, tracking particles using PIV¹ or PTV² methods [55, 61]. For that reason optic flow methods allow us to examine the mixing of fluids.

The results of Chapter 7 show that scaling laws can be found in image sequences of turbulence in the mixing of two fluids by using optic flow methods in conjunction with the linear Gaussian scale-space image representation. The results also show that these scaling laws cannot be modelled by the β -model. A further investigation of the proposed optic flow method with a changed experimental setup would be valuable, because the proposed experimental setup seem to be quite complex and for that reason hard to analyse.

The main conclusion of this thesis is therefore:

It seems promising to investigate turbulent flow and especially mixing of two fluids by using optic flow methods in conjunction with linear Gaussian scale-space. In linear Gaussian scale-space we have a natural framework for the examination of scaling laws of turbulence.

Further investigations are needed in order to fully understand the usage of scale-space optic flow methods as a tool for the investigation of turbulence. A simpler experiment with known scaling properties is needed in order to evaluate the precision of the method. It would also be

¹Particle Imaging Velocimetry.

²Particle Tracking Velocimetry.

interesting to investigate the validity of other cascade models, than the β -model, for the mixing of two fluids.

An analysis of singularity structures of scale-space optic flow in conjunction with studies of turbulence would also be of interest.

Furthermore, the optic flow method proposed in this thesis leave room for further developments, like for instance using the first order derivatives of the flow field as a further constraint on the minimisation of the difference between the normal component of the sought optic flow and the known normal flow.

Appendix A

Notation

\mathbb{N}	The set of positive integer numbers.
\mathbb{Z}	The set of integer numbers.
\mathbb{R}	The set of real numbers.
\mathbb{R}_+	The set of real positive numbers.
$\mathbb{R}_+ \setminus \{0\}$	The set of real positive numbers except zero.
\mathbb{C}	The set of complex numbers.
C^n	The set of the n times differentiable functions.
$L^2(\mathbb{R}^N)$	The set of square integrable functions on \mathbb{R}^N .
$G(\vec{x}; \sigma)$	The Gauss function.
x^i	The i th component of the vector \vec{x} .
f_i	The partial derivative $\partial f(\vec{x}) / \partial x^i$.
$\mathcal{F}\{f(\vec{x})\}$	The Fourier transform of $f(\vec{x})$.
$\mathcal{F}^{-1}\{f(\vec{\omega})\}$	The inverse Fourier transform of $f(\vec{\omega})$.
$f(\vec{x}) *_{\vec{x}} g(\vec{x})$	Convolution of f with g in \vec{x} .
$I(\vec{x}), f(\vec{x})$	The raw original image.
$L(\vec{x}; \sigma_s)$	Spatial scale-space image at spatial scale σ_s .
$L(\vec{x}; \sigma_s, \sigma_t)$	Spatiotemporal scale-space image at scales (σ_s, σ_t) .
$\vec{v}(\vec{x}; \sigma_s, \sigma_t, \varpi)$	Spatiotemporal integration scale-space velocity at scales $(\sigma_s, \sigma_t, \varpi)$.
$\vec{v}(\vec{x}; \vec{\sigma})$	Spatiotemporal integration scale-space velocity at scales $\vec{\sigma} = (\sigma_s, \sigma_t, \varpi)^T$.
$\nabla f = (f_x, f_y)^T$	The gradient of $f(x, y)$.
$\text{diag}(\lambda_1, \dots, \lambda_n)$	A diagonal matrix with the diagonal elements $\lambda_1, \dots, \lambda_n$.
$\langle f(\vec{x}), h(\vec{x}) \rangle$	The inner product of the functions $f(\vec{x})$ and $h(\vec{x})$.
$\overline{v^m}$	The m th order statistical moment of the random variable v .
$\mathcal{L}_{\vec{v}} f(\vec{x})$	The Lie derivative of $f(\vec{x})$ along the vector field \vec{v} .
δ_{ij}	The Kronecker delta function, which is $\delta_{ij} = 1$, when $i = j$ and $\delta_{ij} = 0$ otherwise.
$\vec{0}$	The zero vector.

Appendix B

Proofs

The two proofs of Theorem 4.1 and Theorem 4.2 can be found in this appendix.

B.1 Proof of Theorem 4.1

Proof: To prove this theorem we need to find a solution to the minimisation problem of Eq. (4.15) using the energy functional $E(\vec{v})$ of Eq. (4.16). This can be done by solving these two partial differential equations of the energy functional $E(\vec{v})$

$$\frac{\partial E}{\partial \tilde{v}^x} = 0, \quad \frac{\partial E}{\partial \tilde{v}^y} = 0 \quad (\text{B.1})$$

for the two unknown local flow field model parameters $(\tilde{v}^x, \tilde{v}^y)^T$. We therefor have to find the partial derivatives, $\frac{\partial E}{\partial \tilde{v}^i}$.

I now introduce the substitution

$$q = \vec{v} \cdot \vec{\Upsilon} = \tilde{v}^x \Upsilon^x + \tilde{v}^y \Upsilon^y \quad (\text{B.2})$$

into $E(\vec{v})$

$$E(\vec{v}) = \int_{\vec{x} \in \mathbb{R}^2} w(\vec{x}) \|\vec{\Upsilon}q - \vec{u}\|_2^2 G(\vec{x}_0 - \vec{x}; \varpi) d\vec{x}. \quad (\text{B.3})$$

We can then find the partial derivatives of E with respect to \tilde{v}^i by use of the chain rule

$$\frac{\partial E}{\partial \tilde{v}^i} = \frac{\partial q}{\partial \tilde{v}^i} \frac{\partial E}{\partial q}.$$

The partial derivatives of q from Eq. (B.2) with respect to \tilde{v}^i are

$$\frac{\partial q}{\partial \tilde{v}^x} = \Upsilon^x, \quad \frac{\partial q}{\partial \tilde{v}^y} = \Upsilon^y \quad (\text{B.4})$$

and the partial derivatives of $E(\vec{v})$ of Eq. (B.3) with respect to q are, after interchange of the derivatives and the integrals,

$$\frac{\partial E}{\partial q} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) (2\Upsilon^x (\Upsilon^x q - u^x) + 2\Upsilon^y (\Upsilon^y q - u^y)) G(\vec{x}_0 - \vec{x}; \varpi) dx dy. \quad (\text{B.5})$$

By substitution of q into this equation we get

$$\frac{\partial E}{\partial q} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) \left((\Upsilon^{x^2} + \Upsilon^{y^2}) (\Upsilon^x \tilde{v}^x + \Upsilon^y \tilde{v}^y) - \Upsilon^x u^x - \Upsilon^y u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy.$$

Notice that $\Upsilon^{x^2} + \Upsilon^{y^2} = 1$ and $\Upsilon^x u^x + \Upsilon^y u^y = \|\vec{u}\|$, due to the fact that $\vec{\Upsilon}$ is the normalised normal flow vector. We then obtain the partial derivatives of E with respect to \tilde{v}^i by multiplication of $\frac{\partial E}{\partial q}$ with the two partial derivatives of Eq. (B.4)

$$\frac{\partial E}{\partial \tilde{v}^x} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) \left(\Upsilon^{x^2} \tilde{v}^x + \Upsilon^x \Upsilon^y \tilde{v}^y - u^x \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy \quad (\text{B.6})$$

and

$$\frac{\partial E}{\partial \tilde{v}^y} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) \left(\Upsilon^x \Upsilon^y \tilde{v}^x + \Upsilon^{y^2} \tilde{v}^y - u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy. \quad (\text{B.7})$$

If we split up these two integrals using the addition rule for integrals we can restate Eq. (B.6) and Eq. (B.7) as a series of convolutions in \vec{x}

$$\frac{\partial E}{\partial \tilde{v}^x} = 2 \left(\left(w(\vec{x}) \Upsilon^{x^2} \right) * G(\vec{x}; \varpi) \tilde{v}^x + \left(w(\vec{x}) \Upsilon^x \Upsilon^y \right) * G(\vec{x}; \varpi) \tilde{v}^y - \left(w(\vec{x}) u^x \right) * G(\vec{x}; \varpi) \right)$$

and

$$\frac{\partial E}{\partial \tilde{v}^y} = 2 \left(\left(w(\vec{x}) \Upsilon^x \Upsilon^y \right) * G(\vec{x}; \varpi) \tilde{v}^x + \left(w(\vec{x}) \Upsilon^{y^2} \right) * G(\vec{x}; \varpi) \tilde{v}^y - \left(w(\vec{x}) u^y \right) * G(\vec{x}; \varpi) \right).$$

I now use the substitutions

$$\Lambda^{i,j}(\vec{x}; \sigma_s, \sigma_t, \varpi) \equiv \left(w(\vec{x}) \Upsilon^i(\vec{x}; \sigma_s, \sigma_t) \Upsilon^j(\vec{x}; \sigma_s, \sigma_t) \right) *_{\vec{x}} G(\vec{x}; \varpi) \quad (\text{B.8})$$

and

$$\Omega^k(\vec{x}; \sigma_s, \sigma_t, \varpi) \equiv \left(w(\vec{x}) u^k(\vec{x}; \sigma_s, \sigma_t) \right) *_{\vec{x}} G(\vec{x}; \varpi) \quad (\text{B.9})$$

to rewrite the partial derivatives of E and get

$$\frac{\partial E}{\partial \tilde{v}^x} = 2 \left(\Lambda^{x,x} \tilde{v}^x + \Lambda^{x,y} \tilde{v}^y - \Omega^x \right) \quad (\text{B.10})$$

and

$$\frac{\partial E}{\partial \tilde{v}^y} = 2 \left(\Lambda^{x,y} \tilde{v}^x + \Lambda^{y,y} \tilde{v}^y - \Omega^y \right). \quad (\text{B.11})$$

The solution to the partial differential equations in Eq. (B.1) can now be stated as the matrix equation of Eq. (4.18). \square

B.2 Proof of Theorem 4.2

Proof: To prove this theorem we need to find a solution to the minimisation problem of Eq. (4.15) using the energy functional $E(\vec{v})$ of Eq. (4.16) and this can be done by solving the six equations of partial derivatives of the energy functional $E(\vec{v})$

$$\frac{\partial E}{\partial \tilde{v}_j^i} = 0, \quad (\text{B.12})$$

with respect to the six unknown optical flow field model parameters

$$\vec{v}_1 = (\tilde{v}^x, \tilde{v}^y, \tilde{v}_x^x, \tilde{v}_x^y, \tilde{v}_y^x, \tilde{v}_y^y)^T.$$

I introduce the substitution

$$q = \vec{v} \cdot \vec{\Upsilon} = (\tilde{v}^x + \tilde{v}_x^x x + \tilde{v}_y^x y) \Upsilon^x + (\tilde{v}^y + \tilde{v}_x^y x + \tilde{v}_y^y y) \Upsilon^y \quad (\text{B.13})$$

into $E(\vec{v})$

$$E(\vec{v}) = \int_{\vec{x} \in \mathbb{R}^2} w(\vec{x}) \|\vec{\Upsilon} q - \vec{u}\|_2^2 G(\vec{x}_0 - \vec{x}; \varpi) d\vec{x}.$$

We can then find the partial derivatives of $E(\vec{v})$ with respect to the model parameters \tilde{v}_j^i by use of the chain rule

$$\frac{\partial E}{\partial \tilde{v}_j^i} = \frac{\partial q}{\partial \tilde{v}_j^i} \frac{\partial E}{\partial q}$$

Substituting q into the partial derivative of E with respect to q from Eq. (B.5) we get

$$\begin{aligned} \frac{\partial E}{\partial q} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) & \left((\Upsilon^{x^2} + \Upsilon^{y^2}) (\Upsilon^x \tilde{v}^x + \Upsilon^y \tilde{v}^y + \Upsilon^{xx} \tilde{v}_x^x + \Upsilon^{xy} \tilde{v}_x^y + \Upsilon^{yx} \tilde{v}_y^x + \Upsilon^{yy} \tilde{v}_y^y) \right. \\ & \left. - \Upsilon^x u^x - \Upsilon^y u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy. \end{aligned}$$

Notice again that $\Upsilon^{x^2} + \Upsilon^{y^2} = 1$ and $\Upsilon^x u^x + \Upsilon^y u^y = \|\vec{u}\|$. The partial derivatives of q from Eq. (B.13) with respect to the model parameters \tilde{v}_j^i are

$$\begin{aligned} \frac{\partial q}{\partial \tilde{v}^x} &= \Upsilon^x, & \frac{\partial q}{\partial \tilde{v}^y} &= \Upsilon^y \\ \frac{\partial q}{\partial \tilde{v}_x^x} &= \Upsilon^{xx}, & \frac{\partial q}{\partial \tilde{v}_x^y} &= \Upsilon^{yx} \\ \frac{\partial q}{\partial \tilde{v}_y^x} &= \Upsilon^{xy}, & \frac{\partial q}{\partial \tilde{v}_y^y} &= \Upsilon^{yy}. \end{aligned}$$

The partial derivatives of $E(\vec{v})$ with respect to the six model parameters \tilde{v}_j^i can be obtained by multiplication under the integral with the partial derivatives of q , which gives us

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}^x} &= 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) \left(\Upsilon^{x^2} \tilde{v}^x + \Upsilon^x \Upsilon^y \tilde{v}^y + \Upsilon^{x^2} x \tilde{v}_x^x \right. \\ & \quad \left. + \Upsilon^x \Upsilon^y x \tilde{v}_x^y + \Upsilon^{x^2} y \tilde{v}_y^x \right. \\ & \quad \left. + \Upsilon^x \Upsilon^y y \tilde{v}_y^y - u^x \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy, \quad (\text{B.14}) \end{aligned}$$

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}^y} &= 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) \left(\Upsilon^x \Upsilon^y \tilde{v}^x + \Upsilon^{y^2} \tilde{v}^y + \Upsilon^x \Upsilon^y x \tilde{v}_x^x \right. \\ & \quad \left. + \Upsilon^{y^2} x \tilde{v}_x^y + \Upsilon^x \Upsilon^y y \tilde{v}_y^x \right. \\ & \quad \left. + \Upsilon^{y^2} y \tilde{v}_y^y - u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy, \quad (\text{B.15}) \end{aligned}$$

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}_x^x} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) & \left(\Upsilon^{x^2} x \tilde{v}^x + \Upsilon^x \Upsilon^y x \tilde{v}^y + \Upsilon^{x^2} x^2 \tilde{v}_x^x \right. \\ & + \Upsilon^x \Upsilon^y x^2 \tilde{v}_x^y + \Upsilon^{x^2} x y \tilde{v}_y^x \\ & \left. + \Upsilon^x \Upsilon^y x y \tilde{v}_y^y - x u^x \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy, \end{aligned} \quad (\text{B.16})$$

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}_x^y} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) & \left(\Upsilon^x \Upsilon^y x \tilde{v}^x + \Upsilon^{y^2} x \tilde{v}^y + \Upsilon^x \Upsilon^y x^2 \tilde{v}_x^x \right. \\ & + \Upsilon^{y^2} x^2 \tilde{v}_x^y + \Upsilon^x \Upsilon^y x y \tilde{v}_y^x \\ & \left. + \Upsilon^{y^2} x y \tilde{v}_y^y - x u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy, \end{aligned} \quad (\text{B.17})$$

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}_y^x} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) & \left(\Upsilon^{x^2} y \tilde{v}^x + \Upsilon^x \Upsilon^y y \tilde{v}^y + \Upsilon^{x^2} x y \tilde{v}_x^x \right. \\ & + \Upsilon^x \Upsilon^y x y \tilde{v}_x^y + \Upsilon^{x^2} y^2 \tilde{v}_y^x \\ & \left. + \Upsilon^x \Upsilon^y y^2 \tilde{v}_y^y - y u^x \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy, \end{aligned} \quad (\text{B.18})$$

and

$$\begin{aligned} \frac{\partial E}{\partial \tilde{v}_y^y} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(\vec{x}) & \left(\Upsilon^x \Upsilon^y y \tilde{v}^x + \Upsilon^{y^2} y \tilde{v}^y + \Upsilon^x \Upsilon^y x y \tilde{v}_x^x \right. \\ & + \Upsilon^{y^2} x y \tilde{v}_x^y + \Upsilon^x \Upsilon^y y^2 \tilde{v}_y^x \\ & \left. + \Upsilon^{y^2} y^2 \tilde{v}_y^y - y u^y \right) G(\vec{x}_0 - \vec{x}; \varpi) dx dy. \end{aligned} \quad (\text{B.19})$$

We now use the relation described in Definition 3.4 between partial derivatives of the Gauss function and Hermite polynomials, and use the monomials of Eq. (3.8). By using the addition rule for integrals, the substitutions Eq. (B.8) and Eq. (B.9), the monomials listed in Eq. (3.8), introducing tensor notation for the derivatives, assuming the use of natural coordinates, and that the partial derivatives of $\Lambda^{i,j}(\vec{x}; \sigma_s, \sigma_t, \varpi)$ and $\Omega^k(\vec{x}; \sigma_s, \sigma_t, \varpi)$ are normalised by the use of Definition 3.5, the partial derivatives of E with respect to the six model parameters can then be rewritten on the following form

$$\frac{\partial E}{\partial \tilde{v}^x} = 2 \left(\Lambda^{x,x} \tilde{v}^x + \Lambda^{x,y} \tilde{v}^y - \Lambda_x^{x,x} \tilde{v}_x^x - \Lambda_x^{x,y} \tilde{v}_x^y - \Lambda_y^{x,x} \tilde{v}_y^x - \Lambda_y^{x,y} \tilde{v}_y^y - \Omega^x \right), \quad (\text{B.20})$$

$$\frac{\partial E}{\partial \tilde{v}^y} = 2 \left(\Lambda^{x,y} \tilde{v}^x + \Lambda^{y,y} \tilde{v}^y - \Lambda_x^{x,y} \tilde{v}_x^x - \Lambda_x^{y,y} \tilde{v}_x^y - \Lambda_y^{x,y} \tilde{v}_y^x - \Lambda_y^{y,y} \tilde{v}_y^y - \Omega^y \right), \quad (\text{B.21})$$

$$\frac{\partial E}{\partial \tilde{v}_x^x} = 2 \left(-\Lambda_x^{x,x} \tilde{v}^x - \Lambda_x^{x,y} \tilde{v}^y + (\Lambda_{xx}^{x,x} + \Lambda^{x,x}) \tilde{v}_x^x + (\Lambda_{xx}^{x,y} + \Lambda^{x,y}) \tilde{v}_x^y + \Lambda_{xy}^{x,x} \tilde{v}_y^x + \Lambda_{xy}^{x,y} \tilde{v}_y^y + \Omega_x^x \right), \quad (\text{B.22})$$

$$\frac{\partial E}{\partial \tilde{v}_x^y} = 2 \left(-\Lambda_x^{x,y} \tilde{v}^x - \Lambda_x^{y,y} \tilde{v}^y + (\Lambda_{xx}^{x,y} + \Lambda^{x,y}) \tilde{v}_x^x + (\Lambda_{xx}^{y,y} + \Lambda^{y,y}) \tilde{v}_x^y + \Lambda_{xy}^{x,y} \tilde{v}_y^x + \Lambda_{xy}^{y,y} \tilde{v}_y^y + \Omega_x^y \right), \quad (\text{B.23})$$

$$\frac{\partial E}{\partial \tilde{v}_y^x} = 2 \left(-\Lambda_y^{x,x} \tilde{v}^x - \Lambda_y^{x,y} \tilde{v}^y + \Lambda_{xy}^{x,x} \tilde{v}_x^x + \Lambda_{xy}^{x,y} \tilde{v}_x^y + (\Lambda_{yy}^{x,x} + \Lambda^{x,x}) \tilde{v}_y^x + (\Lambda_{yy}^{x,y} + \Lambda^{x,y}) \tilde{v}_y^y + \Omega_y^x \right), \quad (\text{B.24})$$

and

$$\frac{\partial E}{\partial \tilde{v}_y^y} = 2 \left(-\Lambda_y^{x,y} \tilde{v}^x - \Lambda_y^{y,y} \tilde{v}^y + \Lambda_{xy}^{x,y} \tilde{v}_x^x + \Lambda_{xy}^{y,y} \tilde{v}_x^y + (\Lambda_{yy}^{x,y} + \Lambda^{x,y}) \tilde{v}_y^x + (\Lambda_{yy}^{y,y} + \Lambda^{y,y}) \tilde{v}_y^y + \Omega_y^y \right). \quad (\text{B.25})$$

Substituting these partial derivatives into Eq. (B.12) and reordering the equations, we get the linear system of equations found in Eq. (4.23). \square

Appendix C

Examples of the Faraday Image Sequences

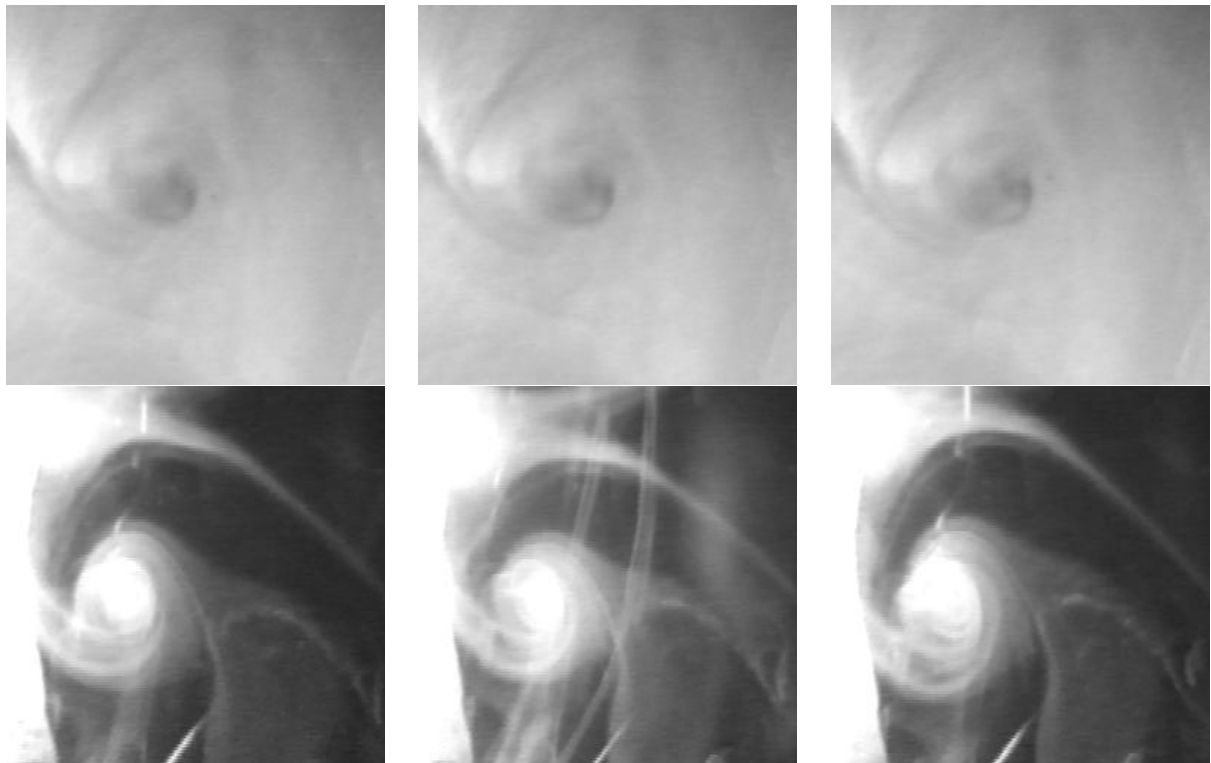


Figure C.1: Examples of frames from the E1/serie2 and E3/serie1 sequences of the Faraday data set. The first, eight, and the sixteenth frames from the sequences are shown.

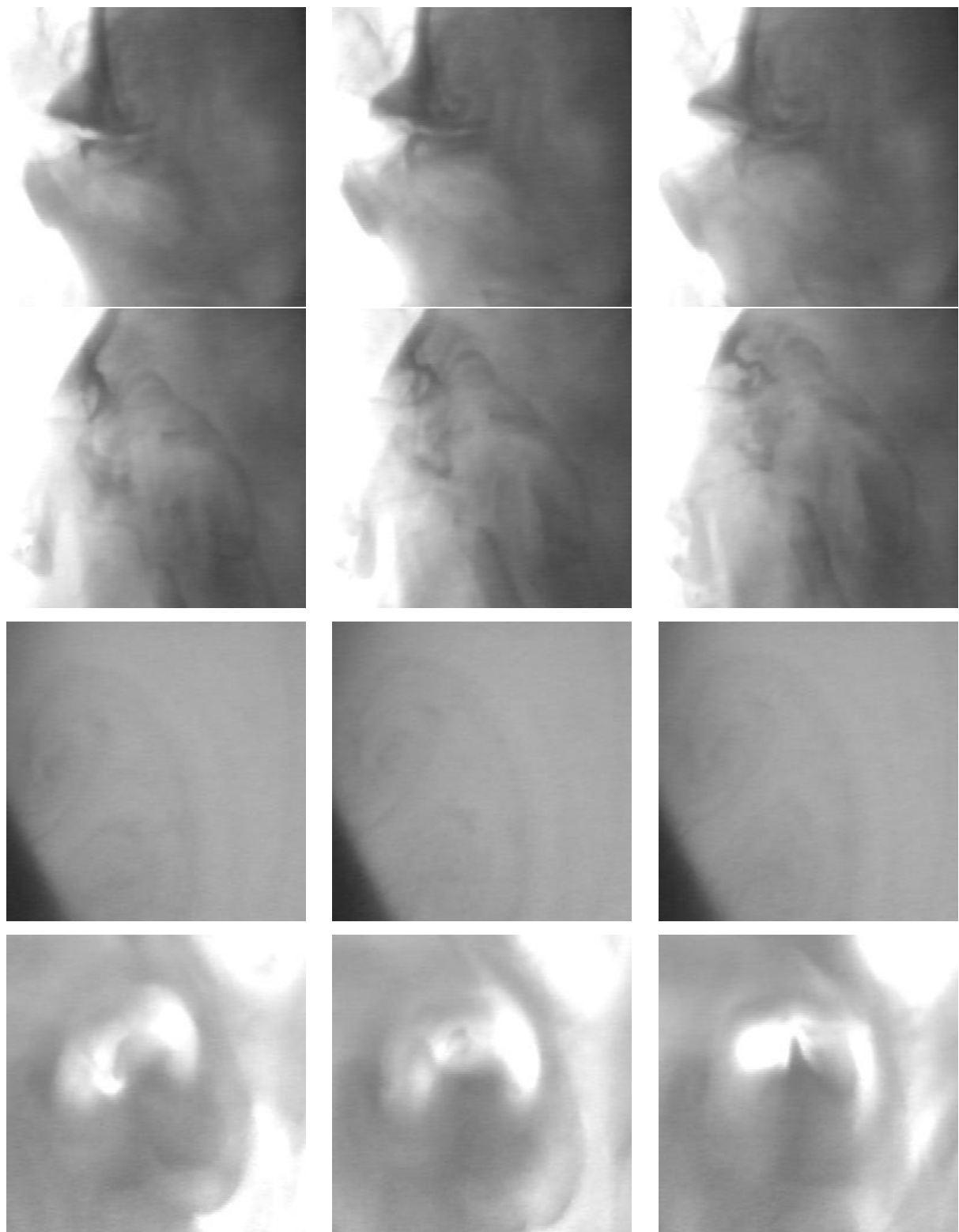


Figure C.2: Examples of frames from E3/serie3, E3/serie4, E3-2/serie2, and E4/serie2 sequences of the Faraday data set. The first, eight, and the sixteenth images from the sequences are shown.

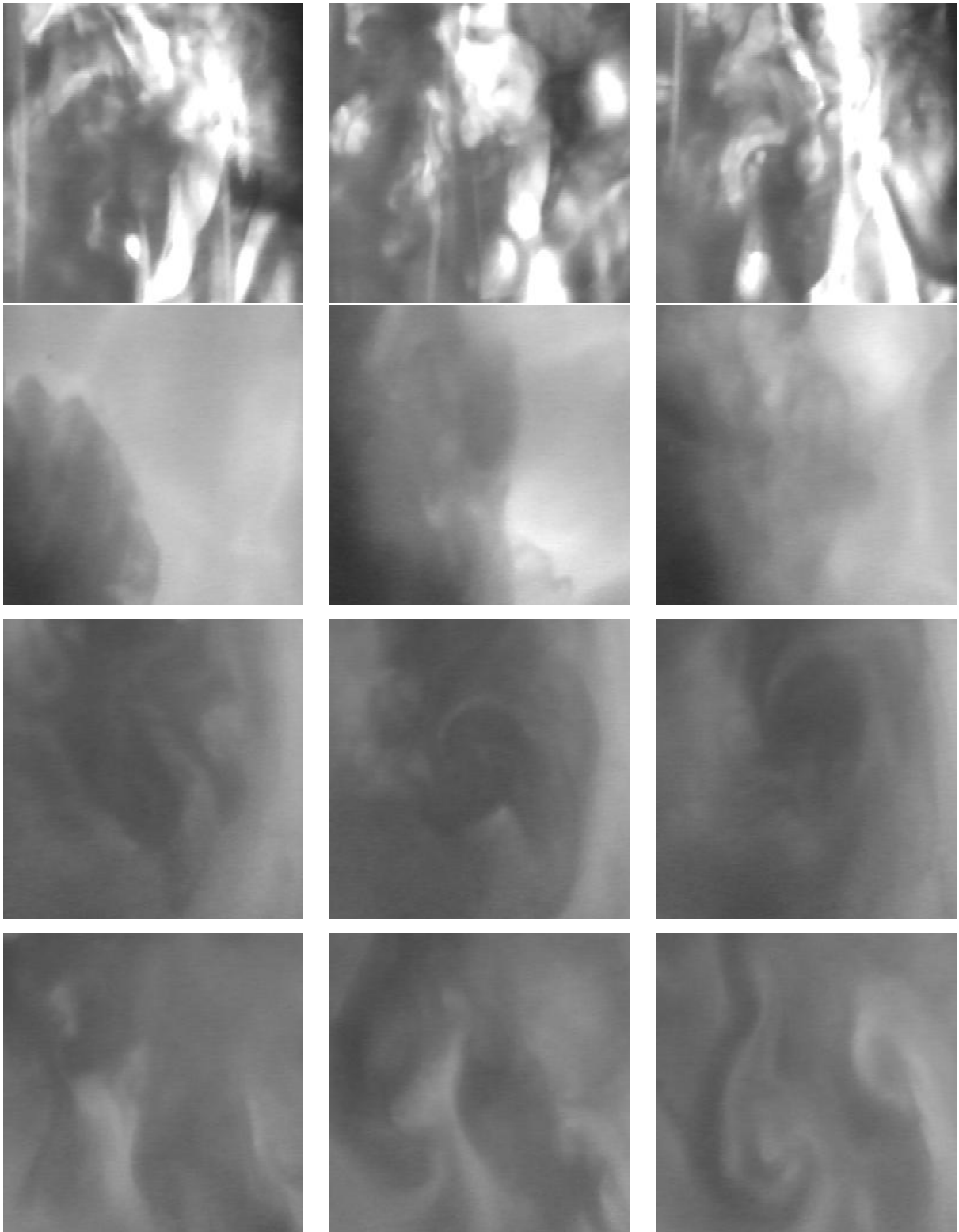


Figure C.3: Examples of frames from E5/serie1, E5/serie2, Vhs/vhs7, and Vhs/vhs8 sequences of the Faraday data set. The first, eight, and the sixteenth images from the sequences are shown.

Bibliography

- [1] V. I. Arnol'd. *Ordinary Differential Equations*. Springer-Verlag, 1992.
- [2] J. Arnsfang. Notes on local determination of smooth optic flow and translational property of first order optic flow. Technical Report 88/1, Department of Computer Science, University of Copenhagen, 1988.
- [3] J. Arnsfang. Optic acceleration. In *Second International Conference on Computer Vision*, pages 364–373. IEEE Computer Society, December 1988.
- [4] J. Arnsfang. Moving towards the horizon of a planar curve. In *Proceedings: Workshop on Visual Motion*, pages 54–59. IEEE Computer Society, March 1989.
- [5] J. Arnsfang. On the use of time varying shading and surface rim irradiance. *Pattern Recognition Letters*, 12:203–210, April 1991.
- [6] J. Arnsfang. Estimators of time to contact. In *SCIA'93: Proceedings of The 8th Scandinavian Conference on Image Analysis*, volume 1, pages 175–182, 1993.
- [7] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.
- [8] M. Bertero and P. Boccacci. *Introduction to Inverse Problems in Imaging*. IOP Publishing Ltd., 1998.
- [9] N. Bohr. *Atomteori og Naturbeskrivelse*. Københavns Universitet, November 1929. Festskrift i anledning af Universitetets aarsfest.
- [10] E. Oran Brigham. *The Fast Fourier Transform and its Applications*. Prentice Hall, 1988.
- [11] P. J. Burt. Fast filter transforms for image processing. *Computer Vision, Graphics, and Image Processing*, 16:20–51, 1981.
- [12] J. L. Crowley. *A Representation for Visual Information*. PhD thesis, Carnegie-Mellon University, Robotics Institute, Pittsburgh, Pennsylvania, 1981.
- [13] C. H. Jr. Edwards and D. E. Penney. *Elementary Differential Equations with Boundary Value Problems*. Prentice-Hall International, 3rd edition, 1993.
- [14] O. Faugeras. *Three-Dimensional Computer Vision. A Geometric Viewpoint*. The MIT Press, 1993.
- [15] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of Optical Society of America*, 4(12):2379–2394, December 1987.

- [16] L. Florack and M. Nielsen. The intrinsic structure of the optic flow field. Technical Report 07/94 - R033, INRIA, 2004 Route des Lucioles, B.P. 93, 06902 Sophia-Antipolis CEDEX, France, July 1994. http://www.ercim.org/publication/technical_reports/.
- [17] L. Florack, W. Niessen, and M. Nielsen. The intrinsic structure of optic flow incorporating measurement duality. Technical Report DIKU Nr. 96/15, ISSN 0107-8283, Department of Computer Science, University of Copenhagen, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2100 København Ø, May 1996.
- [18] L. Florack, W. Niessen, and M. Nielsen. The intrinsic structure of optic flow incorporating measurement duality. *International Journal of Computer Vision*, 27(3):263–286, May 1998.
- [19] L. M. J. Florack, B. M. ter Haar Romeny, J. J. Koenderink, and M. A. Viergever. Linear scalespace. *Journal of Mathematical Imaging and Vision*, 4(4):325–351, 1994.
- [20] Uriel Frisch. *Turbulence: The Legacy of A. N. Kolmogorov*. Cambridge University Press, 1995.
- [21] R. Gilmore. *Catastrophe Theory for Scientists and Engineers*. Dover, 1981.
- [22] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.
- [23] E. C. Hildreth. Computations underlying the measurement of visual motion. *Artificial Intelligence*, 23(3):309–354, August 1984.
- [24] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, 1986.
- [25] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, August 1981.
- [26] T. Iijima. Basic theory on normalization of a pattern. *Bulletin of Electrical Laboratory*, 26:368–388, 1962. In Japanese.
- [27] S. N. Kalitzin, B. M. ter Haar Romeny, A. H. Salden and P. FM Nacken, and M. A. Viergever. Topological numbers and singularities in scalar images: Scale-space evolution properties. *Journal of Mathematical Imaging and Vision*, 9:253 – 269, 1998.
- [28] A. Klinger. Pattern and search statistics. In J. S. Rustagi, editor, *Optimizing Methods in Statistics*. Academic Press, New York, 1971.
- [29] D. C. Knill, D. Field, and D. Kersten. Human discrimination of fractal images. *Journal of Optical Society of America*, 7(6):1113–1123, June 1990.
- [30] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.
- [31] J. J. Koenderink. Scale-time. *Biological Cybernetics*, 58:159–162, 1988.
- [32] R. Larsen. *Estimation of Visual Motion in Image Sequences*. PhD thesis, IMM-DTU, 1994.
- [33] T. Lindeberg. Scale selection for differential operators. Technical Report ISRN KTH/NA/P-9403-SE, Dept. of Numerical Analysis and Computing Science, Royal Institute of Technology, January 1994.

- [34] T. Lindeberg. *Scale-Space Theory in Computer Vision*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1994.
- [35] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. Technical Report ISRN KTH/NA/P-96/06-SE, Dept. of Numerical Analysis and Computing Science, Royal Institute of Technology, June 1996.
- [36] T. Lindeberg. Linear spatio-temporal scale-space. In B. ter Haar Romeny, L. Florack, J. Koenderink, and M. Viergever, editors, *Scale-Space Theory in Computer Vision: First International Conference, Scale-Space'97, Proceedings*, Lecture Notes in Computer Science, pages 113–127. Springer Verlag, 1997.
- [37] T. Lindeberg and D. Fagerström. Scale-space with casual time direction. In B. Buxton and R. Cipolla, editors, *Computer Vision – ECCV'96*, Lecture Notes in Computer Science, pages 229–240. 4th European Conference on Computer Vision Cambridge, Springer Verlag, 1996.
- [38] J.-M. Lévy-Leblond and F. Balibar. *Quantics: Rudiments of Quantum Physics*. North-Holland, Elsevier Science Publishers, 1990.
- [39] B. B. Mandelbrot. *The fractal geometry of nature*. W. H. Freeman and company, San Francisco, 1982.
- [40] B. B. Mandelbrot and J. W. van Ness. Fractional brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, October 1968.
- [41] M. Maurizot, P. Bouthemy, and B. Delyon. 2d fluid motion analysis from a single image: Application to experimental visualizations of fluid flow. Technical Report 1155, IRISA France, <http://www.irisa.fr>, January 1998.
- [42] M. Maurizot, P. Bouthemy, B. Delyon, A. Iouditski, and J. Odobez. Locating singular points and characterizing deformable flow fields in an image sequence. Technical Report 891, IRISA France, <http://www.irisa.fr>, December 1994.
- [43] M. Nielsen, R. Maas, W. J. Niessen, L. L. M. J. Florack, and B. M. ter Haar Romeny. Binocular stereo from grey-scale images. *Journal of Mathematical Imaging and Vision*, 10(2):103–122, March 1999.
- [44] M. Nielsen, W. Niessen, R. Maas, L. Florack, and B. ter Haar Romeny. On the duality of scalar and density flows. In B. ter Haar Romeny, L. Florack, J. Koenderink, and M. Viergever, editors, *Scale-Space Theory in Computer Vision*, pages 65–76. First International Conference, Scale-Space'97, Springer, July 1997.
- [45] M. Nielsen and O. F. Olsen. The structure of the optic flow field. In *Computer Vision – ECCV'98, Proceedings*, Lecture Notes in Computer Science. Springer Verlag, 1998.
- [46] M. B. Nielsen. Cascade models of fully developed turbulence. Master's thesis, Niels Bohr Institute, University of Copenhagen, 1996.

- [47] W. Niessen and R. Maas. Optic flow and stereo. In J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors, *Gaussian Scale-Space Theory*, chapter 3, pages 31–42. Kluwer Academic Publishers, 1997.
- [48] O. F. Olsen. Multi-scale segmentation of grey-scale images. Technical Report DIKU Nr. 96/30, ISSN 0107-8283, Department of Computer Science, University of Copenhagen, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2100 København Ø, 1996.
- [49] E. Ott. *Chaos in dynamical systems*. Cambridge University Press, 1993.
- [50] K. S. Pedersen and M. Nielsen. The hausdorff dimension and scale-space normalisation of natural images. In *Proceedings of Scale-Space Theory in Computer Vision: Scale-Space'99*, Lecture Notes in Computer Science. Springer Verlag, 1999.
- [51] Kim S. Pedersen and Mads Nielsen. The hausdorff dimension and scale-space normalisation of natural images. *Journal of Visual Communication and Image Representation*, 11(2):266 – 277, 2000. Special issue on scale-space'99.
- [52] A. P. Pentland. Fractal-based description of natural scenes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-6(6):661–674, November 1984.
- [53] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in Fortran: The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1994.
- [54] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Trans. Computers*, 20(5):562–569, 1971.
- [55] E. Schröder. *Diffusion and Weak Turbulence in Capillary Waves and in the Complex Ginzburg-Landau Equation*. PhD thesis, Niels Bohr Institute, University of Copenhagen, 1997.
- [56] M. Struwe. *Variational Methods: Applications to Nonlinear Partial Differential Equations and Hamiltonian Systems*, volume 34 of *A Series of Modern Surveys in Mathematics*. Springer Verlag, 2nd edition, 1996.
- [57] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14:67–81, 1995.
- [58] J. Weickert, S. Ishikawa, and A. Imiya. On the history of gaussian scale-space axiomatics. In J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors, *Gaussian Scale-Space Theory*, Computational Imaging and Vision, chapter 4. Kluwer Academic Publishers, 1997.
- [59] J. F. Wendt. *Computational Fluid Dynamics: An Introduction*. Springer Verlag, 2nd edition, 1996.
- [60] A. P. Witkin. Scale space filtering. In *Proc. of the Eighth International Joint Conference on Artificial Intelligence*, volume 2, pages 1019–1023, Karlsruhe, Germany, 1983.
- [61] J. Zhong, T. S. Huang, and R. J. Adrian. Extracting 3d vortices in turbulent fluid flow. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20(2):193–199, February 1998.