Structure from Motion in nD Image Analysis

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 9 februari 2004 om 15:30 uur

door

Bernd RIEGER

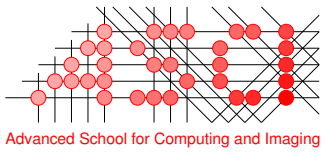
Diplom Physiker, Technische Universität München
geboren te Landsberg am Lech, Duitsland.

Dit proefschrift is goedgekeurd door de promotor:
Prof.dr.ir. L.J. van Vliet

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof.dr.ir. L.J. van Vliet	Delft University of Technology, promotor
Dr. P.W. Verbeek	Delft University of Technology
Prof.dr.ir. P. Kruit	Delft University of Technology
Prof.dr. J. Bigün	Hamlstad University, Sweden
Prof.dr.ir. M.A. Viergever	University Medical Center Utrecht
Prof.dr.ir. J. Westerweel	Delft University of Technology
Dr.ir. L. Dorst	University of Amsterdam
Prof.dr. I.T. Young	Delft University of Technology, reservelid

This work was partly supported by the Netherlands Organization for Scientific Research NWO, grant number 612-012-003.



This work was carried out in graduate school ASCI. ASCI dissertation series number 95.

ISBN: 90-75691-11-4

Nach dem Spiel ist vor dem Spiel.
Sepp Herberger, 1897-1977

Summary

Structure from Motion in nD Image Analysis

In this thesis we investigate the measurement of local properties in multi-dimensional grey-value images. Special attention is given to orientation representation and curvature estimation. Furthermore, the possibility to compute global shape properties from these local properties. We aim to derive sampling-error free measurements. A sampling-error free measurement of a properly sampled image yields exactly the same result as the sampled analog measurement (chapter 3).

We propose a new robust curvature estimation algorithm in nD grey-level images for lines (chapter 4) and surfaces (chapter 5). A better understanding and a generalization of the continuous orientation representation introduced by Knutsson (chapter 2) enables us to compute derivatives of the orientation field leading to the new curvature estimator. In our approach images are represented as multi-dimensional functions and we assume that locally only a single orientation is present. No parametric description of the objects is needed. The curvature can reliably be computed on ridges and in valleys, where classical isophote curvature fails. Isophote curvature cannot deal with areas of vanishing gradient as the curvature is normalized by the gradient magnitude. The local property curvature is used to estimate global properties as bending energy and Euler characteristic (chapter 4 and 5). The integration over a digital object surface is accomplished by a summation of the gradient magnitude over the whole image after soft-clipping.

These local and global measurements are applied to time series: (x, t)-images. Here orientation corresponds to velocity and curvature to acceleration. We will discuss several motion estimation algorithms (chapter 6) and show the connection between spatial and temporal sampling (chapter 3).

The developed algorithms are employed in three applications: 1) the characterization of the rising behavior of bread dough, 2) the flow field analysis of the motion of microspheres in nanoliter vials during evaporation of the liquid, where the liquid flow and the Brownian motion are discussed, 3) a biological application where fluorescent labeled chromatin is observed in a living cell with a confocal microscope. We propose a robust method to correct for the superimposed motion/drift of the cell on the chromatin during the acquisition, which enables a better study of the dynamics of the chromatin.

Contents

Chapter 1

Introduction

The interest to analyze and understand the things we observe is one of the driving forces behind science, philosophy and technological developments. The human visual system (HVS) was and is a source of inspiration in the fields of computer vision, image processing and image analysis. As of today, the HVS is unmatched in the recognition and the communication tasks of visual information. For measurement purposes the HVS is strong in qualitative tasks. However, the field of exact sciences relies to a high extent on objective, reproducible measurement values. Image processing and analysis provide a consistent theory to obtain "hard numbers" for visual phenomena, which allows for a scientific approach to describe visual information (images). In this thesis we will focus on measurements in images. The goal is to measure different properties of 2D and 3D time series as accurately and precisely as possible and to express these measurements quantitatively.

The desire to measure and provide a quantitative description for visual observations has had an impact on recent advances in biology and medicine, which lie in between the exact and the empirical sciences. This desire has been the largest in fields where visual inspection of temporal phenomena plays an important role through the advances in acquisition techniques (resolution and speed) and computer power to process the recorded data. In this respect, the developments of 3D plus time resolved imaging modalities such as confocal microscopes and MRI machines are important. The Netherlands Organization for Scientific Research NWO has made possible an integrated approach to study the dynamic structure of the living cell and of living tissue by bringing together researchers from different fields such as biology, medicine, chemistry, physics and computer science. It funded the project "4D imaging of living cells and tissues", in which three universities and four different fields participate. The present thesis is carried out within this project under the name "4D grey-level image processing".

1.1 Problem

Structure in spatio-temporal images is formed by motion, e.g. a moving point forms a spaghetti-like trajectory in space time, where the local slope of the spaghetti corresponds to the velocity at that point. Imagine it as a moving object leaving a trace in an image much like a charged particle going trough a bubble chamber in particle physics experiments in the last century. There are also other structure parameters in the spatio-temporal image of a moving point that correspond to motion properties, such as the curvature to the acceleration and the path length to the traveled distance. The shape of the generated structure depends on the initial shape at rest; a moving point will form a line, a moving line a plane-like structure and so on. The methods developed here cannot only be applied to space-time images but also to images where all coordinates are space-like. In principle they can be applied to any multi-valued (spectral channels) multi-dimensional data, but the interpretation of the outcome may not be clear. This thesis focuses on measurement techniques describing the structure in spatio-temporal images, where structure is composed of local and global characteristics.

Our goal is to find local measurements of properties such as orientation and curvature and their relation to global information, such that we obtain numeric descriptors for structure that are not necessarily confined to (x, t)-images.

To do so, different aspects of 4D image processing need to be addressed, such as digitization, restoration, low-level processing, motion analysis, measurement, etc.. A 4D (x, t) image is not the same as a 4D (x) image. Even as physicists treat space and time in the same manner, they make a fine distinction, giving time the opposite sign in the space time metric tensor of the Minkowski space. A direction in the time axis is introduced by the increase in entropy. In image processing, the fact that one image axis represents time, has certain consequences. To start with, most acquisition devices do not sample the time axis in the same way as the spatial axes, which can be acquired at different resolutions. Typically, the spatial domain is acquired with lots of sample points, pixels (PICture ELement) or voxels (VOlume ELement), whereas often only a few dozen time samples are available. This is especially the case for in vivo confocal microscopy in biology, MRI imaging in medicine and seismic imaging where two 3D data sets are considered a 4D representation. The amount of available time samples in standard video applications is much higher, as typical frame rates are range at approximately 25 Hz. Given the limited number of time samples, in our applications due to the acquisition constraints, one has actually more often to do with 3D plus time. The term 4D, suggesting an equal amount of available samples of all four dimensions, may be a little overdone here. Time also introduces causality. This means that given an initial shape, its movement cannot produce arbitrary shapes, as the object cannot move backwards in time, i.e. a moving point cannot form a closed ring in

any (x^i, t) projection. Or in other words, spaghetti cannot be laid into a pretzel shape. However, when we observe a Y-shape in (x, t), we see something that is dividing, whereas a Λ-shape in (x, t) represents two merging objects, given that the time axis is pointing upwards. Therefore, causality allows interpretation of structure in (x, t).

An important issue in time resolved applications is the estimation of velocity. One can think of the speed of a moving point, the velocity of a cloud of points, but also of the speed at which a plant grows, the speed of change in texture of a rising dough, etc.. The velocity in itself can be of interest, further properties can be derived from the analysis of the flow field of a region, which can give indications for special points, i.e. the sources/drains and the rotation center of the flow. Spatial velocity distributions can help to identify different parts of a scene that belong together, a process referred to as segmentation. In terms of N-body motion, different velocity aspects are interesting: the center of mass motion, movement of objects with respect to each other and motion analysis of the body itself. The body can be formed out of individual discrete objects or can be a continuous object.

The basis of measurements, characterization and interpretation should be a properly recorded image. Under recording we understand the process to capture an analog phenomenon and transform it into a digital representation, which is suitable for further processing. The distance and size of the digital samples have to be chosen such that reliable measurements can be made. Any analog, physical signal can be synthesized from a superposition of sinusoids of specific amplitude, phase and frequency. If there is a frequency above which the amplitude is zero, we call such a signal band-limited. For proper sampling, the sampling distance must be less than half the smallest occurring period of the sinusoids. This is known as the Nyquist/Shannon sampling theorem and allows reconstruction of the analog signal from the discrete samples. In many applications, the spatial sampling can be done properly, i.e. satisfying this criterion. For a time series the sampling criterion must also hold along the time dimension. Unfortunately, this is often not the case, especially in confocal microscopy. This leads to restrictions on the measurement side and gives rise to the correspondence problem, e.g. the difficulty to identify things that belong together. The nature of acquisition in time and space is very different as well. Optical acquisition of space variant data is ultimately done by a lens system, which imposes certain properties to the recorded signal, such as inevitable blurring by the point spread function and causes mutual influence of neighboring points. This is not the case in the acquisition of time samples. In chapter 3 we will discuss the entanglement of spatial and temporal sampling and present a method to sample time in the same thorough manner as space.

1.2 Approach

Analysis of structure in spatio-temporal images involves several aspects from causality over sampling to local and global measurements in images. The presented approach addresses the relevant sub-problems and motivates them. We address:

- the localization issue of image processing operators,
- grey-value versus binary methods,
- sampling in images sequences,
- the choice of a regularization function,
- local measurements, and
- the combination of these measurements with global knowledge to obtain shape descriptors.

Let us first start with a short note on scale. In cell biological, the resolution of the imaging device a (confocal) microscope *together* with the structure at hand (cell nucleus) dictate the minimal and maximal observable sizes. If you study, for example, the motion of chromatin, you know the size of the structures at hand, i.e. the biological knowledge provides this information. The image processing and analysis operators have to know these typical sizes involved, which are called *scales* in image processing. Rather than exploring the structure as a function of scale, we use knowledge from the application domain to process only a limited number of selected scales that contain relevant information. These operators are in general local operators, which means that they only "see" a certain neighborhood around a point in space and time. More mathematically, their support is bound. The operators will only give the desired responses if the scale is chosen wisely. The scales of interest are dictated by the structures to be studied. In a specific application of image processing and analysis, it is clear what you are looking at. Therefore the scales are defined by that knowledge.

A lot of research has been done to study the human visual system (HVS). This does not only apply to the physiological part, but also to "how we see". It turned out that the eye has separate cells that are sensitive to intensity changes on a large variety of scales and orientations. The eye uses all operators at the same time and the optimal scale is then determined [53, 114]. Having this result in mind, image processing techniques have been developed to emulate parts of this scale selection mechanism. Starting from all scales, the goal is to extract the scales of interest automatically. In our case this framework is not necessary due to the fact that in a specific application knowledge about the important scales is present. A general purpose approach similar to the flexibility of the HVS, incorporated by the

scale-space approach, is not needed. This does, however, not mean that rescaling the operator neighborhoods during the measurement is not desirable or promising. Images can degrade in quality over time, can be distorted by different noise sources which noticeably reduce operator performance. These different contributions, especially from noise, have an influence on the chosen scales of regularization methods applied before or as an integral part of the actual measurement operator. The combination of information from different scales is useful to estimate the local dimensionality and the hierarchy of objects. The estimation of depth by humans at one point is greatly influenced by the global surroundings [67]. Here we see that (limited) local knowledge alone, can lead to wrong conclusions about the observations/measurements. In image processing, localization of image operators may lead to ambiguity of the interpretation of their outcome. In chapter 2 an example for such an arising ambiguity is discussed in depth.

Traditionally, measurements are done after the input image is segmented by binarisation and labeling. Segmentation is a process to separate interesting objects from a background. An object can for instance be as complex as a tree or be its building blocks. Segmentation of a tree in a natural scene is difficult, however most applications have much easier objects which can often be classified by only one or a few local properties, e.g. intensity, texture, size, shape or color[1]. This information may be corrupted by noise, so that thresholding on the one parameter at hand fails, which makes segmentation the critical processing step in the majority of applications. If labeling is acquired, the image is binarised, i.e. assigning the value one to the foreground pixels (object) and the value zero to the background pixels. Measurements are derived from the labeled binary image, e.g. the area being just the sum of the foreground pixels of one object. The processing of binary images had a tremendous advantage in the days before the rise of cheap personal computers in the early 1980s. A binary image can be stored extremely memory-efficient by only coding its boundary. Algorithms working only at the boundary are still among the fastest. However binarisation has great intrinsic problems. Hardly any imaging devices does has a high enough resolution to resolve the quantized structure of nature and even more important most, of the time larger spatial scales are of interest, such that the final observations are continuous. For example a circle drawn with a pair of compasses is round to the eye. Acquiring an image of it and storing/processing it on a computer, quantizes the circle, assigning values to (nearly always) rectangular pixels on a rectangular grid. If the sampling of the grid satisfies the Nyquist criterion, reconstruction of the analog object from the sampled representation is possible and unique. If we now binaries the values of the image, the boundaries of the objects become jagged. The boundary is not even

[1]Color is not considered or referred to in this thesis as humans perceive it. It is solely considered as a function of the photon energy. Images may be multi channel, each channel indicating signal strength (grey-value).

clearly defined, as there is no natural way to split a curved line into square blocks. Furthermore, rotations and sub-pixel shifts of the object are not naturally included into binarisation and can cause an arbitrary error, which will hamper all subsequent measurements.

In general, the deviation that a measurement has from the true value is composed of two terms: the systematic and statistic (or stochastic) error, which are independent of each other. The systematical error is also called bias and determines the accuracy. It describes the deviation of the mean of many measurements from the truth. This error is caused by errors in the measurement procedure or neglecting side effects. Sometimes it is possible to correct this type of error afterwords if the problem and the measurement system are completely understood. The statistical error determines the precision and gives an indication for the deviation between one random measurement result and the mean of many. It characterizes the influence of random effects (noise) on the measurement procedure.

In our opinion, the more appropriate approach to image analysis is to avoid binarisation and to keep the grey-value information for the measurement. It is especially important to preserve the regions of intensity change as the information is essentially confined to these places. Van Vliet and Verbeek introduced a variety of measurements that can be obtained from the grey-volume of an object (sum of the samples), which is a sampling error-free measurement. This concept incorporates affine invariance and sub-pixel precision as long as proper sampling of the image is guaranteed. The sampling constraints along with measurement techniques based on the sum of samples are introduced in chapter 3. The paradigm within this framework is to perform measurements on grey-value images and to avoid aliasing of the image by using band-limited operations. Finally, a measurement value is accompanied by a confidence value that indicates the trustworthiness of the measurement value. The confidence is computed from the structure present in the image, taking into account the model assumptions of the measurement procedure. Nevertheless, noise reduction was and is necessary. This can be done without losing edge information by a regularization step, which can be adaptive smoothing or anisotropic diffusion.

We will follow our paradigm to keep the grey-value information for the measurements in 4D (or higher dimensional) image analysis. Memory constraints and computer power set us back 20 years. We face similar limitations with the computation of three- and four dimensional grey-value images as the pioneers in two dimensions with binary data.s The theories and algorithms we will present are first developed in analog space, keeping in mind the local nature of image operators, and are in the end digitized and applied to the digitized images.

Measurements in images requires the application of filters to the image. Gaussian filters are used throughout the thesis by the convolution of a Gaussian (or a

derivative of a Gaussian) with the image. The approach to convolve the image with a regularization function (the Gaussian) is a natural extension of the function theory given by the distribution theory. In this theory a framework is constructed for the handling of, for example, Dirac deltas, which are not functions in the traditional way, as a function value reaches infinity at one point. The choice for the Gaussian as the regularization function is based inter alia on the minimal space-frequency product of the Gaussian. Convolution with a regularization function allows further for a straight forward manner to compute derivatives of discrete images. The derivative of an image is just the convolution with the derivative of the regularization function.

In chapter 2 and 3 we specify the localization problem of image operators, sampling issues and the need to perform band-limited operations. Along these lines we present in chapters 4 and 5 a measurement algorithm for the curvature of lines and surfaces (local measurements) and discuss several global shape descriptors computed from these local curvatures. Different estimation methods of velocity in spatio-temporal images available in the literature are revised in chapter 6. They are used together with the previously developed shape descriptors in three applications.

ξυνον γαρ ἀρχη και περας ἐπι κυκλου περιφερεια.
Beginning and end coincide in the circumference of the circle.
Herakleitos, 536–470 (frag.103)

Chapter 2

A systematic approach to nD orientation representation[1]

We define simple neighborhoods in images as areas that are shift invariant in at least one direction and not shift invariant in at least one other direction. Such areas play a key role in the description of local structure. The aforementioned shift directions can be determined up to point inversion. Therefore a pair of opposite directions is designated by a single orientation.

The first order intensity variation is the gradient. A collection of local gradients is needed to compute a dominant orientation. The accompanying intensity variations and that of orthogonal directions can be used to describe lines, surfaces and edges as well as texture. A characterization of simple neighborhoods is by the dominant orientation [10, 60, 45, 11].

Orientation is direction up to point inversion, therefore leaving room for ambiguity in representation. Direction is described by the full angle, which is in 2D characterized by $\vartheta \in [0, 2\pi]$, and in general by $n-1$ angles in nD. Direction can also be represented by vectors, for example $v = (1, 2)$ or $w = (-1, -2)$. These vectors point in opposite directions but have the same orientation.

Representing orientation by vectors (direction information) leads to troublesome descriptions, in the sense that it is discontinuous. Representing a line in 2D by its angle with respect to a fixed coordinate axis and a plane in 3D by its normal vector is therefore not a suitable representation. In figure 2.1 a test and a real 2D image are shown on the left and the orientation fields on the right. We clearly see two jumps or discontinuities. They cannot be removed, for example by phase

[1]Part of this chapter has been published in B. Rieger and L.J. van Vliet, Representing orientation in N-dimensional spaces, CAIP'03, *10th International Conference on Computer Analysis of Images and Patterns (Groningen, The Netherlands)*, August 25-27, 2003 [104] and an extended version is accepted for publication in Image and Vision Computing [105].

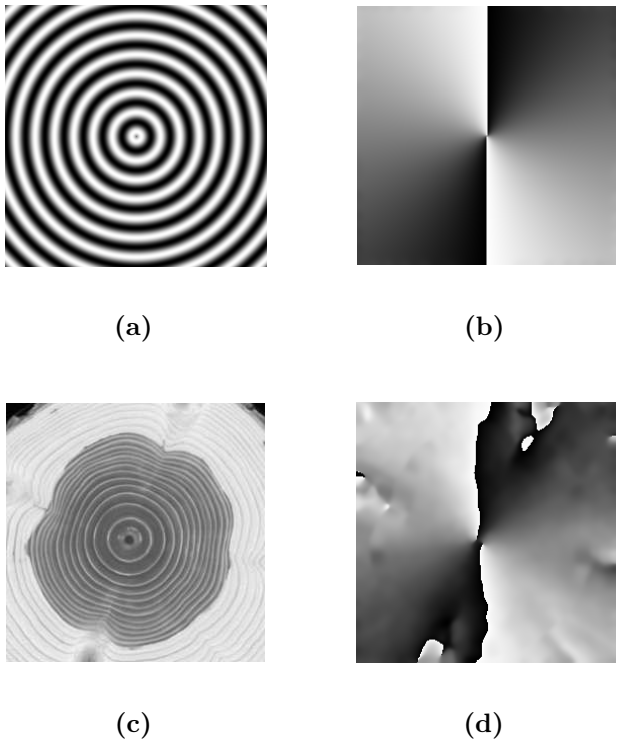


Figure 2.1: a) Image with concentric circles forming an oriented texture, b) the local orientation field $[-\frac{\pi}{2}, \frac{\pi}{2}[$ (dark,light) with a discontinuity mod π, c) a cross section through a CT image of a tree trunk and d) the orientation field.

unwrapping [137, 70, 29]. Phase unwrapping can only successfully be applied to discontinuities that form non-intersecting closed lines.

A consistent definition of direction is only possible in a global frame work, whereas most image operators are bound to a local neighborhood. The heart of the problem is sketched in figure 2.2. The support of the operator may have a size as indicated by the circle. The scale is local, whereas the structure has a global connection. The content of the upper and lower window are the same, although the outward pointing normal vector changes continuously along the line. So we are left with two identical windows and an estimated normal orientation with a discontinuity.

The gradient vectors in a local neighborhood need to be combined to obtain an estimate of the local orientation. A simple averaging of gradient vectors fails, on lines in 2D (planes in 3D etc.) because vectors from opposite sides of the line point in opposite directions and will cancel each other. Thus we need a suitable continuous representation of gradient vectors to average the structure inside a local window.

Furthermore, a discontinuous representation is very often not suitable for fur-

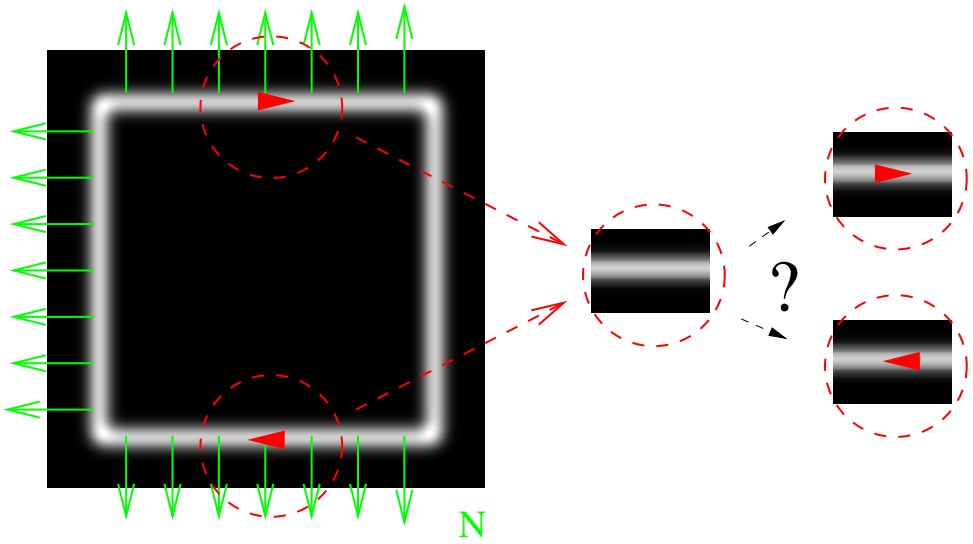


Figure 2.2: Local analysis and the problem of direction versus orientation information. The normal vector in the two dashed windows point in different direction whereas the content of the windows is the same. Global knowledge is needed to resolve direction information.

ther processing. Most image operators give incorrect response to apparent discontinuities. Therefore the approach should be as follows: obtain the gradient vectors, map them to a continuous representation, carry out the averaging (or apply another filter). The interpretation of the results of the filtering operation on the new representation is then - in general - not straightforward. A well-known tool to analyze local structure is the Gradient Structure Tensor $\bar{G}$ (GST) [10, 60, 56, 45, 71, 43, 11]. It is defined as

$$\overline{G} := \overline{\nabla I \nabla I^t} \text{ with } G := \nabla I \nabla I^t \,, \tag{2.1}$$

where I is a grey-value image and the over-lining stands for averaging the elements inside a local neighborhood.

A physical interpretation of the GST can be given in the terms of friction [123]. Imagine the grey-value surface of a 2D image as a washboard where the friction is proportional to absolute gradient strength. The GST is then a measure for the local mean squared friction $o^t \overline{G} o$ along orientation o when rubbing over the surface. An eigensystem analysis of the GST yields the orientation with the least and the highest friction; along and perpendicular to the ribs of the washboard.

The structure tensor also is similar to the covariance matrix $C = \langle X^2 \rangle - \langle X \rangle^2$. In statistical pattern recognition the covariance matrix is used to describe a set of points (here the points are generated by the endpoints of the gradient vectors) [57, 12]. The relation is given by

$$C = \overline{G} - \overline{\nabla I_i} \overline{\nabla I_j} \quad 1 \leq i, j \leq n \,. \tag{2.2}$$

The covariance matrix and the structure tensor are identical if the average (expec-

tation) per element is zero, $\langle X \rangle = 0$, i.e. on lines and planes. The orientation estimation of the GST is very robust to noise. An accurate estimate for the orientation can be obtained for signal-to-noise ratios close to 0dB (dependent on the window size) [61, 140].

The gradient structure tensor is also related to a well-known quantity in physics, the inertia tensor [111] by the following relation

$$\mathbf{J} = \mathrm{tr}\,(\mathbf{G})\mathbf{I} - \mathbf{G} \quad \text{or} \quad J_{ij} = \mathrm{tr}\,(G_{ij})\delta_{ij} - G_{ij}\,. \tag{2.3}$$

One way to see the connection is that the GST estimates the inertia of the gradient vector endpoints which are translated to the a common origin.

The gradient structure tensor overcomes the problem in averaging orientation by mapping the local gradient ∇I via a dyadic product to a continuous representation which allows filtering; averaging with a weight function. The outcome cannot be interpreted directly but first an eigenvalue analysis of $\overline{G}$ has to be done, where the ratios of the eigenvalues characterize local structure [56, 43], i.e. the local dimensionality. Due to the nonlinearity of the structure tensor, applying arbitrary linear filters to the tensor result may produce an unexpected outcome.

The gradient structure tensor clearly treats gradients (x) pointing in opposite direction ($-x$) equally with respect to direction *and* magnitude $G : \mathbb{R}^n \ni x \rightarrow xx^t \in \mathbb{R}^{n\times n}$. These two properties are necessary conditions for sensible averaging of the tensor elements. In other words, rotation of the image yields an equally rotated result of the tensor space. In other circumstances it may be desirable to preserve absolute differences in orientation in the mapping resolving the orientation problem. The GST has $n(n+1)/2$ independent components. Limiting the number of elements needed reduces the memory requirements for higher dimensional images. We will look for a minimal set of elements to describe local orientation satisfying three conditions that are outlined below.

2.1 Requirements of a continous distance preserving mapping

Knutsson proposed the following three properties for a continuous distance preserving representation of orientation: *Uniqueness, Uniform Stretch* and *Polar Separability* [64, 63]. Let be $x \in \mathbb{R}^n$:

- Uniqueness: Antipodal vectors should be mapped onto one point, this removes the phase jump, e.g. opposite gradient vectors are treated equally$M(x) = M(-x)$.

- Polar Separability: The norm of the mapped vector should be rotation invariant; information carried by the magnitude of the original vector does normally not depend on the angle. $\|M(x)\| = f(\|x\|)$, where f is an arbitrary function $f : \mathbb{R}^+ \to \mathbb{R}^+$.

- Uniform Stretch: The mapping should carry implicitly information about the distances in the original space that is rotation invariant and scales linearly with the angle between two hyper planes $\|\delta M(x)\| = c\|\delta x\|$ for $\|x\| = const.$

2.2 The mapping

A mapping that fulfills the above requirements is $M : \mathbb{R}^n \to \mathbb{R}^{n\times n}$

$$M(x) = \frac{xx^t}{\|x\|} \,. \tag{2.4}$$

It was introduced by Knutsson in this form in 1989 [64]. From the construction it is clear that M is symmetric and has only $\frac{n(n+1)}{2}$ independent components. The mapping is slightly different from the structure tensor G. The latter does not meet all of the above requirements. The uniqueness requirement is met by G, also the polar separability as $\|G(x)\| = \|x\|^2$, but the uniform stretch property is not met as $\|\delta G(x)\| = c\|x\|\|\delta x\|$. The structure tensor is therefore no distance preserving mapping.

If other than blurring filters are applied to G or M the outcome has not been studied. The uniform stretch property will allow us to compute curvature of lines and surfaces in nD by applying derivative filters to M and to interpret the outcome. This is the key to the new curvature estimation methods of chapter 4, eq.(4.7) and chapter 5, eq.(5.12).

The Knutsson mapping M is a scaled version of G. Where G has only one non-zero eigenvalue $\|x\|^2$ with eigenvector x, M has the same eigenvector to the scaled eigenvalue $\|x\|$.

For the norm in the uniformity and the polar separability Knutsson chose the Fröbenius norm

$$\|M\|^2 := \sum_{ij} m_{ij}^2 \,. \tag{2.5}$$

If the norm is rotation invariant, as given here by the polar separability requirement, then

$$\|M\|^2 = \mathrm{tr}\,(M^t M) = \sum_n \lambda_n^2 \tag{2.6}$$

where λ_n are the eigenvalues of M. Here the Fröbenius norm is equal to the often used spectral norm $\|M\|_{spec} := \sqrt{\text{largest eigenvalue of}(M^t M)}$ as M has only one nonzero eigenvalue $\|x\|$.

At this point we notice that we can further reduce the number of independent components of M; and G for that matter. The polar separability requirement guarantees for both mappings a rotation invariant norm and

$$\|M\|^2 = \lambda_{max}^2 = \lambda_1^2 = \mathrm{tr}\,(M)^2 = const.. \tag{2.7}$$

So there is another restriction imposed on the sum of the diagonal elements of M. Therefore from the n diagonal elements only $n-1$ linear combinations are necessary; the mappings M and G have $\frac{n(n+1)}{2} - 1$ independent components. This can be important as the fewer dimensions to process, the less memory is needed.

2.2.1 The mapping in 2D

In the following we identify $M(x) = \frac{x_i x_j}{\|x\|}$ as an ordered n-tuple. For the 2D case the mapping M reads $x \in \mathbb{R}^2$:

$$M(x) = \frac{1}{\|x\|}(x_1^2, x_1x_2, x_2x_1, x_2^2)\,, \tag{2.8}$$

or in polar coordinates $x_1 = r\cos\varphi, x_2 = r\sin\varphi$,

$$M(x) = r(\cos^2\varphi, \sin\varphi\cos\varphi, \cos\varphi\sin\varphi, \sin^2\varphi)\,. \tag{2.9}$$

From the above consideration we know that there are only $\frac{n(n+1)}{2} - 1 = 2$ independent components. Linear combinations of the diagonal elements yield only one component carrying information

$$\cos^2\varphi + \sin^2\varphi = 1 \tag{2.10}$$
$$\cos^2\varphi - \sin^2\varphi = \cos 2\varphi\,. \tag{2.11}$$

To scale all elements evenly we take twice the off-diagonal element m_{12}, $2\sin\varphi\cos\varphi = \sin 2\varphi$. Summing up we get a *reduced set*

$$M_{r2D}(x) = \frac{1}{\|x\|}(x_1^2 - x_2^2, 2x_1x_2) = r(\cos 2\varphi, \sin 2\varphi)\,. \tag{2.12}$$

The Knutsson mapping M, reduced to M_{r2D} in 2D, is equivalent to the well known *double angle method* [10, 56]: $r(\cos\varphi, \sin\varphi) \rightarrow r(\cos 2\varphi, \sin 2\varphi)$. Note that the double angle method cannot be generalized to higher dimensions in a straightforward manner. The functions $r(\cos 2\varphi, \sin 2\varphi)$ are the circular harmonics [17]. They form

a basis of the polynomials of second degree in 2D, so the reduced set M_{r2D} (2.12) is indeed minimal.

Already in 1987 Bigün and Granlund proposed an orientation operator for 2D images using these linear combinations of the components of the structure tensor to describe the orientation φ [10]

$$\tan 2\varphi = \frac{2\overline{\partial_x I \partial_y I}}{\overline{(\partial_x I)^2} - \overline{(\partial_y I)^2}}\,, \tag{2.13}$$

where the overhead bar denotes averaging in a local neighborhood.

In figure 2.3 we illustrate that the gradient $(\partial_x I, \partial_y I)$ is not continuous whereas the reduced set is, and therefore averaging of theses elements is allowed (i.e. it does not yield false responses to the apparent discontinuities).

2.2.2 The mapping in 3D

Again we want to reduce the dimensionality of the general mapping M. For the 3D case we apply the successful recipe from 2D; taking twice the off-diagonal elements and cyclic combinations of the diagonal elements

$$\frac{1}{\|x\|}(\underbrace{x_1^2 - x_2^2, 2x_1x_2}_{\text{again 2D}}, 2x_1x_3, 2x_2x_3, x_2^2 - x_3^2, x_3^2 - x_1^2)\,. \tag{2.14}$$

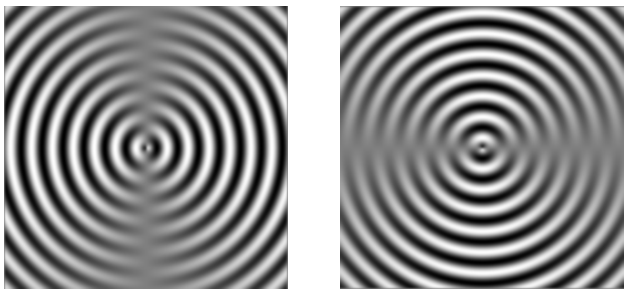
We have 6 components, from the above considerations we know that there are only $\frac{n(n+1)}{2} - 1 = 5$ independent components. Due to the restriction on the trace a combination of the cyclic diagonal elements m_1, m_5, m_6 is suitable to reduce the set by one. We choose $m_6 - m_5$

$$M_{r3D}(x) = \frac{1}{\|x\|}(x_1^2 - x_2^2, 2x_1x_2, 2x_1x_3, 2x_2x_3, \frac{1}{\sqrt{3}}(2x_3^2 - x_1^2 - x_2^2))\,. \tag{2.15}$$

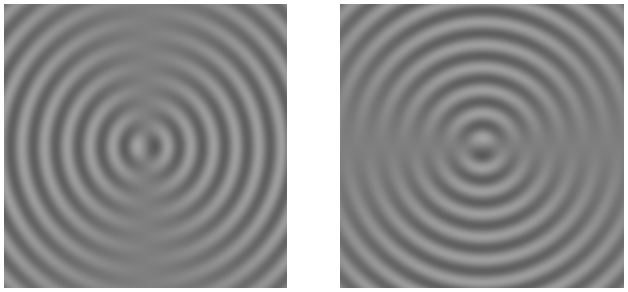
For a 2D subspace ($x_3 = 0$) this reads

$$M_{r3D}(x) = \frac{1}{\|x\|}(x_1^2 - x_2^2, 2x_1x_2, 0, 0, \frac{-\|x\|^2}{\sqrt{3}})\,, \tag{2.16}$$

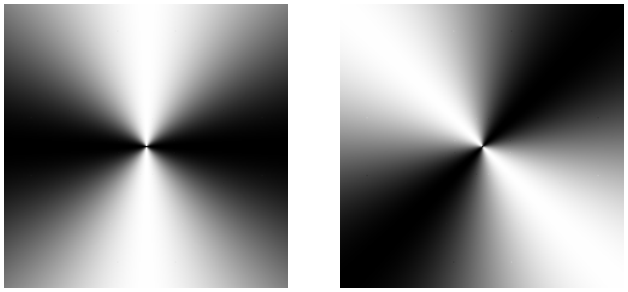
where we recognize the 2D mapping again. A 2D orientation half plane in 3D is mapped onto a 2D cone in 5D with opening angle $\alpha = 120°$ (slope of the cone $1/\sqrt{3}$ and $\tan 30° = 1/\sqrt{3}$) as depicted in figure 2.4.



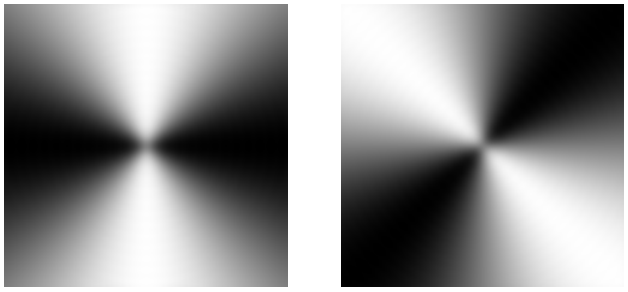
(a) $\partial_x I$ **(b)** $\partial_y I$



(c) $\overline{\partial_x I}$ **(d)** $\overline{\partial_y I}$



(e) $(\partial_x I)^2 - (\partial_y I)^2$ **(f)** $2\partial_x I \partial_y I$



(g) $\overline{(\partial_x I)^2} - \overline{(\partial_y I)^2}$ **(h)** $2\overline{\partial_x I \partial_y I}$

Figure 2.3: Illustration along with the mapping (2.13), a,b) the derivatives of the image in figure 2.1a), c,d) smoothing of the derivatives with a Gaussian filter, e,f) the reduced mapping M_{r2D}, g,h) smoothing of the reduced mapping with a Gaussian filter. For easier educational insight elements e)-h) are normalized by the gradient magnitude.

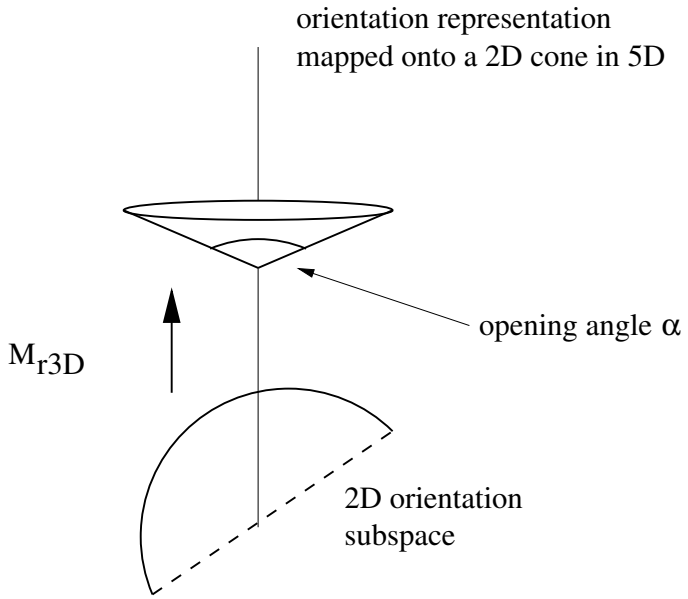


Figure 2.4: A 2D orientation half plane in 3D is mapped via M_{3rD} onto a 2D cone in 5D with opening angle $\alpha = 120°$ (slope of the cone $1/\sqrt{3}$ and $\tan 30° = 1/\sqrt{3}$).

Knutsson 1985:

In 1985 Knutsson introduced a mapping fulfilling the above requirements suitable for 3D [63]. This function $M_K : \mathbb{R}^3 \rightarrow \mathbb{R}^5$ is written in spherical polar coordinates $r \in \mathbb{R}^+, \phi \in [-\pi, \pi]$ and $\theta \in [0, \pi]$

$$\begin{aligned} M_K(r,\theta,\phi) \;\rightarrow\; & r(s,t,u,v,w), \\ & s = \sin^2\theta\, \cos 2\phi, \\ & t = \sin^2\theta\, \sin 2\phi, \\ & u = \sin 2\theta\, \cos\phi, \\ & v = \sin 2\theta\, \sin\phi, \\ & w = \sqrt{3}(\cos^2\theta - \frac{1}{3})\,. \end{aligned} \tag{2.17}$$

Knutsson gave no derivation for his mapping M_K and said it was heuristically derived. We can derive M_K in a systematical way from the general mapping M (2.4).

This function M_K is in fact not different from M. M_K is the reduced set M_{r3D}(2.15) in spherical polar coordinates: $x_1 = r\cos\phi\sin\theta$, $x_2 = r\sin\phi\sin\theta$, $x_3 = r\cos\theta$. The extension from 2D to 3D is now no longer heuristic, as we presented its derivation. Extension to higher dimensions has become straightforward.

Another way to M_K:

The extension to 3D (2.17) of the 2D solution, the double angle method (2.12), can also be derived by analogy transfer. The double angle method maps the tangent to the circular harmonics. Now let's have a look at the spherical harmonics [17]

$$Y_l^m(\theta,\phi) = \begin{cases} P_l^m(\cos\theta)\cos(m\phi) & \text{for} \quad m = 0,1,2,\ldots,l \\ P_l^{|m|}(\cos\theta)\sin(|m|\phi) & \text{for} \quad m = -1,\ldots,-l \end{cases}, \tag{2.18}$$

where P_l^m are the associated Legendre polynomials.

The spherical harmonics of second degree Y_2^m are (up to a scaling factor) identical to the components of the mapping M_K

$$\begin{aligned} s &= 1/3Y_2^2, \\ t &= 8Y_2^{-2}, \\ u &= 2/3Y_2^1, \\ v &= 4Y_2^{-1}, \\ w &= 2/\sqrt{3}Y_2^0\,. \end{aligned}$$

The spherical harmonics form a basis of the polynomials of second degree in 3D, so the set of components of M_{r3D} (2.15) is indeed minimal.

2.3 Properties of the mappings

In table 2.1 we summarize some properties of the mappings presented here.

	Dimension	Uniform Stretch	Polar Separability	Angle
G	n	–	$\|x\|^2$	–
M	n	$c=\sqrt{2}$	$\|x\|$	$\cos\alpha=\cos^2\psi$
M_{r2D}	2	$c=2$	$\|x\|$	$\cos\alpha=\cos 2\psi$
M_K	3	$c=2$	$\frac{2}{\sqrt{3}}\|x\|$	$\cos\alpha=\frac{3}{4}(\cos 2\psi+\frac{1}{3})$

Table 2.1: Summary of the properties of the different mappings.

The angle ψ of two vectors $x, y \in \mathbb{R}^n$ in the original space can be related to the angle α that the mapped vectors $M(x), M(y) \in \mathbb{R}^{n\times n}$ will form in the mapped space M. Knutsson has done this only for M_K [63, 64]. For the general mapping

M (2.4) the deduction is

$$\cos\psi = \frac{x \cdot y}{\|x\| \, \|y\|} \tag{2.19}$$

$$\cos\alpha = \frac{M(x) \cdot M(y)}{\|M(x)\| \, \|M(y)\|} = \frac{\sum_{ij} x_i x_j y_i y_j}{\|x\|^2 \, \|y\|^2} = \frac{(x \cdot y)^2}{\|x\|^2 \, \|y\|^2} \tag{2.20}$$

$$\Rightarrow \quad \cos\alpha = \cos^2\psi \, . \tag{2.21}$$

In figure 2.5 we plot the angle transfer functions for M, M_K, M_{r2D} between the angle of two vectors in one space and the angle of the vectors in the mapped space. We observe the following properties:

$$M : n\text{D} \rightarrow n^2\text{D} \qquad \psi \in [0, 180^\circ] \rightarrow \alpha \in [0, 90^\circ] \tag{2.22}$$

$$M_K : 3\text{D} \rightarrow 5\text{D} \qquad \psi \in [0, 180^\circ] \rightarrow \alpha \in [0, 120^\circ] \tag{2.23}$$

$$M_{r2D} : 2\text{D} \rightarrow 2\text{D} \qquad \psi \in [0, 180^\circ] \rightarrow \alpha \in [0, 180^\circ] \tag{2.24}$$

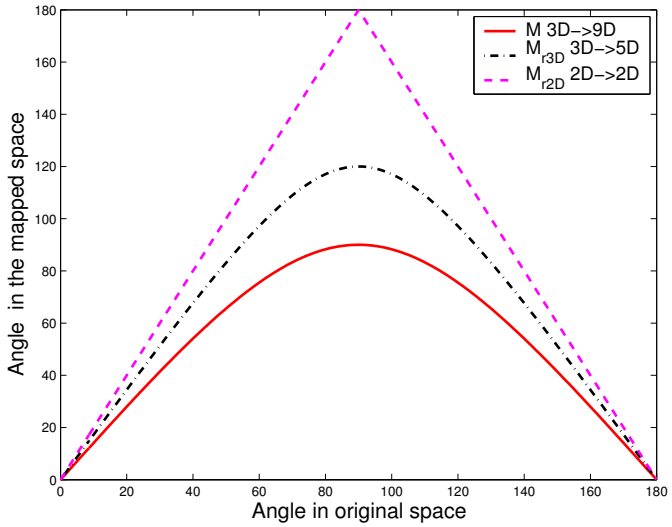


Figure 2.5: Transfer function of the angle depicting the relation between the angle in the original space and the corresponding angle after mapping the vectors. The maximal angle in the mapped space is the opening angle of the mapped cone of a 2D half plane, compare figure 2.4.

Both reduced mappings M_{r2D} and $M_{r3D} \equiv M_K$ have the same uniformity factor of 2, but the maximal angle in the mapped space is different. The 2D mapping maps the 2D orientation half plane onto the full 2D plane, which is a cone with opening angle 180°, whereas M_{r3D} maps the half plane to a cone with opening angle 120° (see also figure 2.4). Flatting of the 3D cone to a 2D plane is possible. The stretching is canceled by the polar separability factor $\frac{2}{\sqrt{3}}$ to one, thus flattening of M_{r3D} is identical to use M_{r2D} directly.

Furthermore, Knutsson pointed out that the sum of two non-zero mapped vectors (by M or M_K) is always greater than zero because the maximal angle they can make is smaller than 180°. He interpreted this as a consequence of the fact that there is no fully symmetric representation[2] of two 3D lines [63], but there is one of three 3D lines, i.e. three perpendicular lines. If we map three perpendicular lines

$$\begin{array}{lrcl} \text{z-axis:} & M_K(r,0,\phi) & = & r(0,0,0,0,2/\sqrt{3}) \\ \text{y-axis:} & M_K(r,\pi/2,\pi/2) & = & r(1,0,0,0,-1/\sqrt{3}) \\ \text{x-axis:} & M_K(r,\pi/2,0) & = & r(-1,0,0,0,-1/\sqrt{3}) \end{array}$$

then the sum of these three mapped vectors is zero. Here we see the possibility to generate, from two mapped vectors (by M_K) in 5D, a third vector that is perpendicular to the other two *and* that is in the image of M_K. To find a vector in 5D that is perpendicular to two other vectors is not difficult, it is, however, more complicated to find the one which lies in the subspace of the image of M_K. In 3D the outer product is used to obtain a perpendicular vector from two other, in 5D the sum of those three must be zero

$$\begin{array}{lrcll} \text{3D} & z & = x \times y & & (2.25) \\ \text{5D} & M_K(z) & = -[M_K(x) + M_K(y)]\,. & & (2.26) \end{array}$$

2.4 Conclusions

We have shown in which manner specific solutions (2D [10, 60] and 3D [63, 64]) for a continuous orientation representation are connected through a general framework for arbitrary dimensions. In a general manner we can reduce the dimensionality of the Knutsson mapping (2.4) and of the gradient structure tensor to a minimal set necessary to describe orientation in multi-dimensional images. The relation with the circular and spherical harmonics has become clear, as a minimal set of necessary components in 2D and 3D. Furthermore, the difference between the structure tensor and the Knutsson mapping are discussed, the first being not distance preserving which is important when applying other than averaging filters. The uniqueness and polar separability are properties of both mappings.

[2] Fully symmetric in the sense that reflections map the lines onto each other.

Chapter 3

Sampling

In this chapter we present the terminology and tools necessary for the discussion in the following chapters. At the basis of image processing and analysis stands the acquisition of images that are subsequently stored in a digital format. The step from the analog to digital domain has to be done with some thought. First we summarize some definitions and important relations that describe how sampling has to be done in order to allow reconstruction of the continuous data from the sampled data and, closely related, the conditions for performing reliable measurements. In the second part we discuss the relation between spatial and temporal sampling. For an overview of sampling (of space and time variant phenomena) see a textbook, such as [95] chapter 7, [55] chapter 4.6, [56] chapter 9 or [79] chapter 4.

- Fourier transform: The Fourier transform of a function $f \in L^2(\mathbb{R}^n), \xi \in \mathbb{R}^n$ is the isometric isomorphism[3] $\mathcal{F} : L^2(\mathbb{R}^n) \to L^2(\mathbb{R}^n)$

$$\mathcal{F}\{f\}(\xi) := \hat{f}(\xi) = (2\pi)^{-n/2} \int_{\mathbb{R}^n} f(x) e^{-ix\xi} \, dx \,, \tag{3.1}$$

  with ξ the angular frequency and the inverse given by

$$\mathcal{F}^{-1}\{f\}(x) := \check{f}(x) = (2\pi)^{-n/2} \int_{\mathbb{R}^n} f(\xi) e^{ix\xi} \, d\xi, \text{ with } \check{\hat{f}} = f \,. \tag{3.2}$$

  For periodic functions it is convenient to define the variable in Fourier space as $2\pi\nu = \xi$. Then ν is directly the conventional frequency. Furthermore, we have the property:

$$\widehat{f * g} = (2\pi)^{n/2} \hat{f} \hat{g} \,, \tag{3.3}$$

[3] Isometric isomorphism: bijective and preserves the norm. The actual definition of the Fourier transform is on $f \in L^1$ but can be extended to L^2 by Plancherel's theorem, see for the details [108] chapter 7.

with $*$ the convolution operator $(f * g)(x) := \int f(t)g(x-t)dt$. The Fourier transformation of a Gaussian is again a Gaussian

$$g(x) := (\sqrt{2\pi}\sigma)^{-n} e^{-\frac{x^2}{2\sigma^2}} \tag{3.4}$$

$$\hat{g}(\xi) = (2\pi)^{-n/2} e^{-\frac{1}{2}\sigma^2\xi^2} . \tag{3.5}$$

- Band-limit: A function f is called band-limited if $\hat{f}$ exists and

$$\hat{f}(\xi) = 0 \text{ for } |\xi_i| > b_i \quad i = 1 \dots n . \tag{3.6}$$

  The b_i are called the band-limit. If f is acquired by a lens system it is band-limited. In optics spatial data is observed through a lens. The finite aperture of this lens ensures band-limitation of the signal in the image plane. The band-limitation manifests itself through some blurring. If f is not band-limited, it can be made band-limited using a low-pass filter. Van Vliet [132], p. 28/29 suggests the use of a Gaussian with $\sigma = 0.9$, as it is a good approximation to the Point Spread Function (PSF) of a diffraction limited lens. The low-pass filter, however, must be applied in the analog domain (before sampling).

- Nyquist/Shannon sampling theorem: A function that is band-limited is uniquely determined by its samples, if the sampling frequency $\xi_{i,s}$ exceeds twice the highest occurring frequency b_i [93, 113, 41]

$$\xi_{i,s} > 2b_i \; i = 1 \dots n . \tag{3.7}$$

  For lens systems $>$ can be replaced by $\geq$ as the energy at the cut-off frequency is zero. Digital filtering (after sampling) cannot achieve band-limitation.

3.1 Reconstruction and measurement

3.1.1 Perfect reconstruction by ideal low-pass filtering

The multidimensional delta train is used to model the sampling process, and is defined by

$$\text{comb}_\Delta(x) := \sum_{m \in \mathbb{Z}} \prod_{i=1}^{n} \Delta_i \delta(x_i - \Delta_i m_i) , \tag{3.8}$$

where Δ_i is the sampling distance in the i-dimension. The Fourier transform of the comb function is again a comb function with spacing $\frac{2\pi}{\Delta}$. Sampling a function I is multiplication with the comb function

$$I[m] \overset{ADC}{\longleftrightarrow} \text{comb}_\Delta(x) I(x) . \tag{3.9}$$

In figure 3.1 the spectrum of a band-limited function I with band-limit b, the spectrum of the comb function and the sampling by the comb function are shown. The sampling frequency ξ_s does not meet (3.7), i.e. the signal is undersampled and all frequencies above ξ_r are aliased. The (unaliased part of the) signal $I(x)$ can be

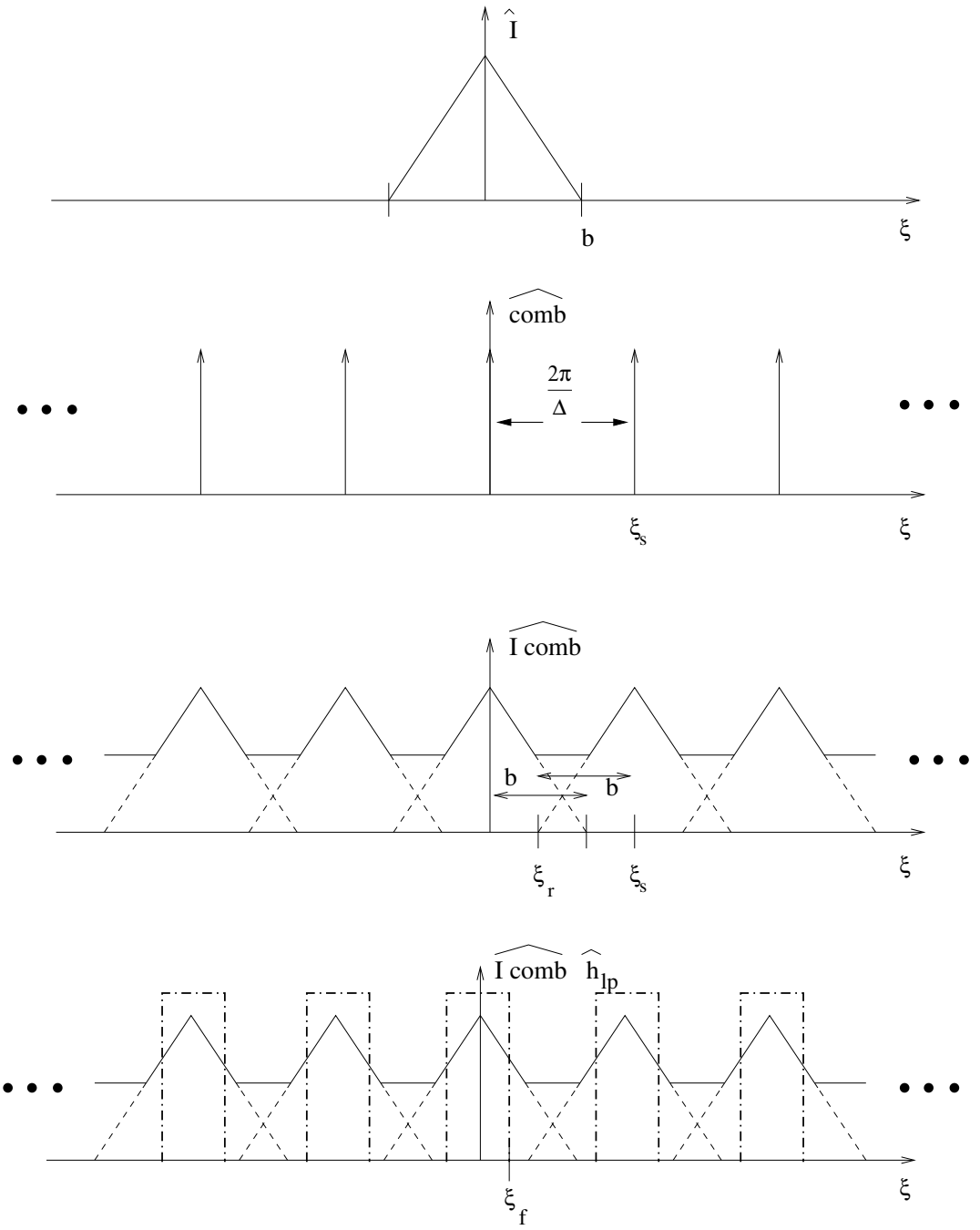


Figure 3.1: a) Spectrum of a band-limited signal I with band-limit b, b) Fourier-transform of the comb function eq.(3.8) c) the product of a) and b) resulting in an aliased signal, d) recovering the unaliased part of the spectrum with a low-pass filter h_{lp} with $\xi_f \leq \xi_s - b$.

retrieved from $\mathrm{comb}_\Delta(x)I(x)$ by ideal low-pass filtering. An ideal low-pass filter to achieve this is a block in Fourier domain with cut-off frequency ξ_f

$$\xi_f \leq \xi_s - b\,. \tag{3.10}$$

For critical sampling ($\frac{\xi_{s,crit}}{2b} = 1 + \varepsilon, \varepsilon \searrow 0$) $\xi_f = b$ recovers the whole signal. If $\xi_s < b$, the required cut-off frequency ξ_f would be negative indicating that $\hat{I}(0)$, the DC component, is aliased and therefore the whole signal. The inverse Fourier transformation of a block is the sinc function

$$\mathrm{sinc}(x) := \frac{\sin x}{x}\,. \tag{3.11}$$

Ideally a sinc function would be needed to interpolate the original from the samples. There are, however, two reasons this is not done in practice. First, the sketched sampling theory is only valid on the domain $\mathbb{R}^n$, whereas we are restricted to finite-length signals in practice. Second the sinc decays slowly towards zero in the spatial domain.

3.1.2 Finite number of sample points

In figure 3.2 we show the effects of a finite number of sample points on the Fourier transform of a sinc. It is no longer a true block function. At the folding frequency ($\xi = \pi$) there occur small oscillations, this is the so called Gibbs phenomenon [95] section 3.

3.1.3 Approximately band-limitation

The concept of approximate band-limitation is adopted to deal with signals f that are not truly, but for most practical reasons, band-limited. The approximated band-limit ξ_a can be defined as the value where $\hat{f}$ encloses 99% of the total volume or has dropped to 1% of the maximal signal value. We choose the latter definition. The reason is simple: Functions that decay as $1/\xi$ (e.g. step edges introduced by thresholding) accumulate volume very fast as we walk along the ξ-axis due the divergence at $\xi = 0$. Other definitions for the approximate band-limit may be reasonable.

For a Gaussian $g(x)$ the condition

$$\hat{g}(\xi) \leq 0.01 \max \hat{g}(\xi) \quad \forall\, |\xi| \geq \xi_a \tag{3.12}$$

is satisfied from

$$\xi_a = \frac{\sqrt{2 \ln 100}}{\sigma} \approx \frac{3}{\sigma} =: b_G\,. \tag{3.13}$$

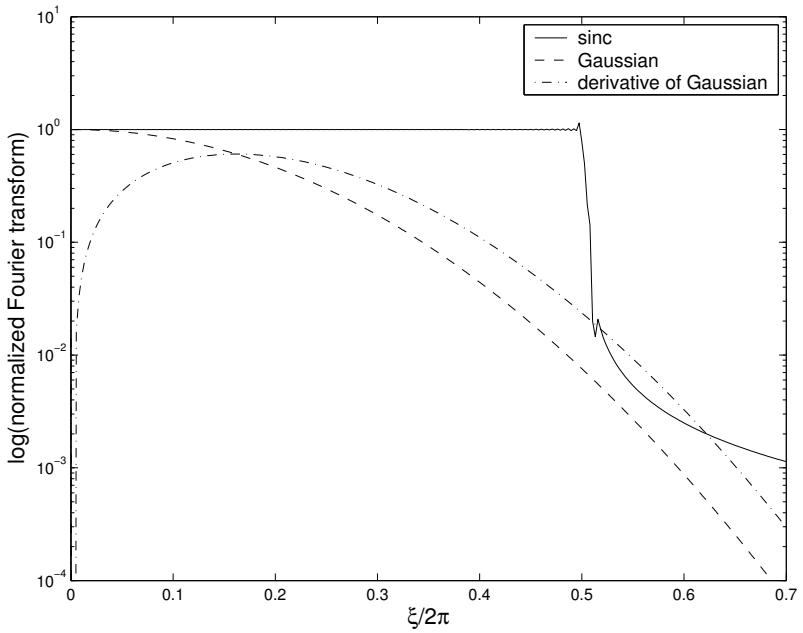


Figure 3.2: The normalized (to be one at $\xi = 0$) Fourier transforms of a sinc, a Gaussian and the first order derivative of a Gaussian all at $\sigma = 1$.

Sampling a Gaussian according to the Nyquist theorem (3.7) requires sampling $\frac{2\pi}{\Delta} = \xi_s > 2\xi_a$, therefore the sampling distance should be

$$\Delta \lesssim \sigma \,. \tag{3.14}$$

With standard sampling $\Delta = 1$ this means that a Gaussian should at least have $\sigma = 1$ to be approximately band-limited. Van Vliet argued, based on the PSF function of a microscope, that $\sigma \geq 0.9\Delta$ should be fulfilled as an appropriated sampling condition for band-limitation of a Gaussian [132] chapter 2.5.

For truly band-limited functions f the following relation holds for the band-limit b of the derivative

$$b(f) = b(Df) \,. \tag{3.15}$$

For approximate band-limited function this relation is not true in general, as $\widehat{Df} = i\xi\hat{f}$, but for a Gaussian the exponent $e^{-\frac{1}{2}\sigma^2\xi^2}$ decays stronger than any power of ξ. Therefore we will also use relation (3.15) for derivatives of approximate band-limited functions. The Fourier transform of the Gaussian and its first derivative, both with $\sigma = 1$ and standard sampling $\Delta = 1$, are shown in figure 3.2. We see that for $\frac{\xi}{2\pi} = 0.5$ (folding frequency) $\hat{g}$ is indeed about 1% of the maximal value. The derivative has a slightly higher approximate band-limit but we stick to eq.(3.15) for further use.

The integral of a Gaussian, the erf function

$$\mathrm{erf}(x) := \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \tag{3.16}$$

is thus also approximately band-limited. Filtering a step edge with a Gaussian results in a erf function. An erf function is a 'smooth step edge'.

3.1.4 Sum of samples

Before we start with the use of the sum of samples to obtain measurements, we take an excursion to the measurement theory in discretized images. This will motivate why in the remainder of the thesis the grey-volume is used as the integration measure to obtain global measurements.

Measurement theory in images

An image has discrete pixels whereas the real world has not. This is the heart of the problem. Our goal is to define a framework such that measurements on the digital image give the same result as the measurements of the analog object, .i.e. the framework must take care of the discrezation.

The first approaches to measure global object properties, such as volume, surface area and perimeter have been binary in nature. This means the object is recorded and separated from the background by thresholding of the data. On the binarised object a measurement is performed to obtain the desired information. Unfortunately, even binary methods that have no systematical error, can have a high statistical error. Different realizations (by shift and rotation) of the same original object on the image grid followed by binarisation do *not* yield the same result (except for integer shifts and few rotations). The goal therefore is to develop a set of rules such that the statistical error is zero (or at least small). The requirements of such a framework fulfills the class of sample-error free measurements, which includes the sum of samples.

Sum of samples

The sum of samples $\sum_m I[m]$ is a sampling-error free measure [122]. A sampling-error free measure is a measurement that performed on a sampled image gives the same result as its analog counterpart. The sum of samples is proportional to the integrated grey-value (*grey-volume*), if the sampling frequency ξ_s is higher than the

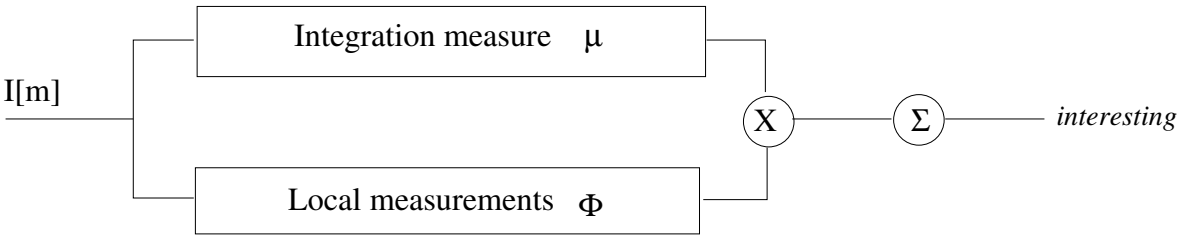


Figure 3.3: Measuring scheme for global measurements of an object. Combination of the integration measure μ with the point measurements is the input for the sum of samples. This sum must be direct proportional to the property to be measured.

highest frequency b of the analog signal [122] and [132] chapter 2.3

$$\begin{aligned}
\left(\prod_i^n \Delta_i\right) \sum_m I[m] &= \int_{\mathbb{R}^n} I(x)\mathrm{comb}_\Delta(x)\,dx && (3.17)\\
&= (2\pi)^{n/2}\hat{I}(\xi) * \widehat{\mathrm{comb}}_{2\pi/\Delta}\Big|_{\xi=0} && (3.18)\\
&= (2\pi)^{n/2}\hat{I}(0) && (3.19)\\
&= \int_{\mathbb{R}^n} I(x)\,dx, \quad \frac{2\pi}{\Delta_i} > b_i\,. && (3.20)
\end{aligned}$$

The sampling requirement is twice as loose as the Shannon/Nyquist sampling theorem (3.7).

3.1.5 From the sum of samples to global measurements

The grey-volume is a sampling-error free measure, and thus it is desirable to measure global object characteristics by grey-volume measurements. Now we will focus on how a measurement can be transferred to a grey-volume measurement. The big picture is emphasized, for a much more comprehensive discussion of the details see [122, 129, 128, 132, 135].

Assume we want to measure the global property *interesting* of an object (e.g. surface area), then the measurement scheme in figure 3.3 should be followed. We obtain

$$interesting = \sum \Phi\mu(I)\,, \tag{3.21}$$

where the filter Φ is a combination of local measurements $\Phi_1, \ldots, \Phi_m$. The product $\Phi\mu(I)$ must be approximately band-limited, and so must all local measurements Φ_i. This can require oversampling dependent on the operators μ and Φ [122]. Many measurements are done by applying (derivatives of) Gaussians, thus we need an approximately band-limited Gaussian. This can be realized as discussed above (3.1.3).

Let us, for the moment, assume $\Phi = 1$. The integration measure μ, a transformation of the input image, is dependent on the dimensionality of the measurement k to be done. In table 3.1 we summarize integration measures μ for different measurement dimensionalities k which are applicable to *closed filled* objects (objects which are enclosed by edges only) in nD images [132] part III. On the way we introduce erf-clipping. Erf-clipping (a form of soft-clipping) of an image between the limits $\ell \pm \frac{1}{2}h$ is defined as [132] chapter 2.6

$$I_{\text{clip}}(\ell, h) = \ell + \frac{h}{2}\text{erf}\left(\frac{\sqrt{\pi}}{h}(\ell - I)\right) . \tag{3.22}$$

Erf-clipping is approximately band-limited, i.e. at half the sampling frequency the spectrum of the erf-clipped signal is more than one order of magnitude lower than hard clipping or thresholding (and preserves the edge information) [132]. The erf-clipped result is in good approximation a Gaussian filtered step edge. The approximate band-limit is given by the corresponding Gaussian.

μ	k	Property
I_{clip}	n	Volume
$\|\nabla I_{\text{clip}}\|$	$n-1$	Surface ($n \geq 3$)/Perimeter ($n = 2$)
$\partial_{gg} I_{\text{clip}}$	$n-2$	Length ($n = 3$)
$\partial_g^{(n-1)} I_{\text{clip}}$	1	Length

Table 3.1: Integration measures μ that produce a grey-volume that is proportional to the specified property in nD dimensional images I with property dimensionality k [132]. Where I_{clip} is defined by eq.(3.22) and ∂_g is the derivative in the local gradient direction.

Example: $\mu = \|\nabla I_{\text{clip}}\|,\ \Phi = 1$

The gradient magnitude is found to be the integration measure for surface area [132]. Here we discuss why this transformation is suitable to measure surface area. However, we leave out details of the edge localization by Gaussian filters. For the gradient magnitude standard sampling is sufficient [122].

Assume, a single band-limited solid object O is embedded in an image I by an iso-surface at level ℓ. The integration measure μ must produce an image which integrated volume = *const.* × surface area. Thus each cross section e (the edge) perpendicular to the iso-surface ∂O should contribute an equal amount. First we apply erf-clipping, then the cross section is a clipped edge, $e(r) = h - \int_0^r p(\tilde{r})\, d\tilde{r}$ with profile distribution $p(\tilde{r})$ satisfying $\int_0^\infty p(\tilde{r})\, d\tilde{r} = h$. Perpendicular to the iso-surface we have

$$\|\nabla I_{\text{clip}}\| = \|\frac{d}{dr} I_{\text{clip}}\| . \tag{3.23}$$

Note that the integral $\int \|\frac{d}{dr} I_{\text{clip}}\|\, dr = \int \|\frac{d}{dr} e(r)\|\, dr = h$. Finally, volume integration of $\|\nabla I_{\text{clip}}\|$ accumulates the contribution of all cross sections and hence $\frac{1}{h}\int \|\nabla I_{\text{clip}}\|\, dV$ yields the surface area of the embedded object. Thus integration over an iso-surface element dA is accomplished by integration of the gradient magnitude after clipping over the entire image volume

$$\int_{\partial O} dA = \frac{1}{h}\int_I \|\nabla I_{\text{clip}}\|\, dV\,. \tag{3.24}$$

The volume integral can be replaced by a sum of the sampled image iff $\|\nabla I_{\text{clip}}\|$ is approximately band-limited. The derivatives are implemented by convolutions with derivatives of Gaussians. Here we see again that we require a band-limited Gaussian to transfer an analog theory to the discrete domain without sampling-errors. The approximate band-limitation is no fixed criterion, but as experiments by van Vliet [135] show, for $\sigma \geq 0.9\Delta$ there is no statistical error, which justifies the chosen restriction.

Example: $\mu = \|\nabla I_{\mathbf{clip}}\|,\ \Phi = \kappa_1^2 + \kappa_2^2$

For the bending energy of an object embedded in a 3D image, we have to set $\Phi = \kappa_1^2 + \kappa_2^2$, where κ_i are the principal curvatures of the object. Together with $\mu = \|\nabla I_{\text{clip}}\|$ this yields

$$\begin{aligned}\int \kappa_1^2 + \kappa_2^2\, dA \;\;\leftrightarrow\;\; & \int (\kappa_1^2 + \kappa_2^2)\|\nabla I_{\text{clip}}\| dV \\ & = \sum \text{comb}_\Delta(\kappa_1^2 + \kappa_2^2)\|\nabla I_{\text{clip}}\|\,.\end{aligned} \tag{3.25}$$

The curvatures κ_i have to be computed along the lines laid out before on the grey-value information of the image. The sum of samples only corresponds to the physical volume of the object if the recording does not contain specular reflection or other disturbances that do not reflect the physical object height. The erf-clipping takes care of this in most cases.

For hollow (or ridge like) objects this approach does not hold, they must first be filled, see chapter 5.1.4.

3.2 (x,t)-sampling

The sampling of space-time dependent data is important for many tasks ranging from reconstruction of the continuous (x,t)-signal from its (x,t)-samples to characterization, recognition and measurement of the true or apparent motion. The choice of the temporal sampling frequency, given the image content, i.e. the velocity of

the fastest, important object, is closely related to the problem to establishing a relation between the observations of an object in a time series as will be discussed later. Acquisition of time data differs from the acquisition of spatial data. In the time domain there is no inherent blurring by a lens system: a shutter of a camera is opened and then closed, defining the exposure time. Time band-limitation is introduced by spatial band-limitation as we will discuss in the following.

3.2.1 Introducing time band-limitation by spatial band-limitation

In practice the question often arises, given a spatial sampling, how fast do we have to sample in time to measure a certain motion without aliasing? It can also be put the other way around: given a limited time sampling by the acquisition aperture what is the maximal observable velocity? The sampling issues in space and time, however, cannot be seen separately from each other. The image obtained by the spatial acquisition by a lens system is band-limited by the cut-off frequency of the lens. Spatial band-limitation, luckily, induces temporal band-limitation as can be seen from figure 3.4. If we assume uniform rectilinear motion, i.e. $\ddot{x} = 0$, then the velocity v is

$$v = \cot\varphi = \frac{\Delta x}{\Delta t}\,. \tag{3.26}$$

The Fourier transform of an uniform rectilinear motion is a straight line in Fourier space. The truncation in the reciprocal spatial axis k induces therefore truncation of the reciprocal temporal axis ω as can be seen from figure 3.4. For linear motion $I(x,t) = I(x - vt), I(x,0) = I(x)$ the spectrum lies on a 90° rotated line

$$\begin{aligned}
\mathcal{F}\{I\}(k,\omega) &= \frac{1}{(2\pi)^2}\int dtdx\, I(x-vt)e^{-i(kx+\omega t)} && (3.27)\\
&= \frac{1}{(2\pi)^2}\int dtd\tilde{x}\, I(\tilde{x})e^{-i(k\tilde{x})}e^{-it(kv+\omega)} && (3.28)\\
&= \frac{1}{\sqrt{2\pi}}\hat{I}(k)\delta(kv+\omega)\,. && (3.29)
\end{aligned}$$

Here we did not apply the conventional space-time metric $(-1,1,1,1)$, but treated all components in the Fourier transform the same.

With the spatial band-limits b_i and velocity components v_i in dimension i, we obtain for the time band-limit b_t from figure 3.4

$$b_t = \max_i(b_i v_i)\ i = 1,\ldots n\,. \tag{3.30}$$

This implies that given a spatial band-limit, the time band-limit is proportional to the velocity. Eq.(3.30) gives a useful relation between the space and time band-limit/sampling. It is now clear that both cannot be addressed separately.

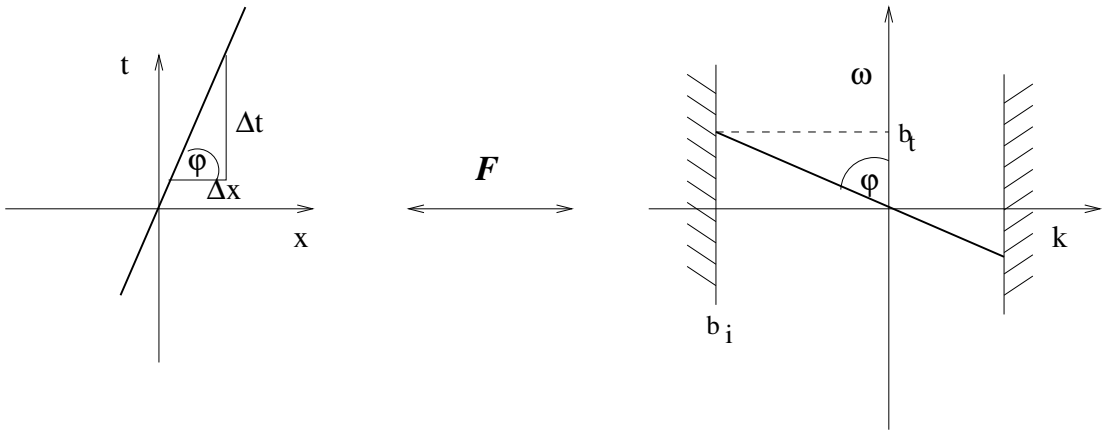


Figure 3.4: Sampling of linear motion and the relation between temporal and spatial band-limit. For linear motion $I(x,t) = I(x - vt)$ the spectrum lies on a 90° rotated line $\mathcal{F}\{I\}(k,\omega) = \hat{I}(k)\delta(kv + \omega)$. Here we only show one object with $I(x,0) = \delta(x)$ for simplicity.

Now assume constant acceleration as $I(x,t) = I(x - at^2)$. Fourier analysis yields $\hat{I}(k,\omega) = \hat{I}(k)\frac{1}{\sqrt{2\pi}}\int_{v=0}^{v_{max}} dv\,\delta(kv + \omega)$, so we see that the temporal band-limit is just given by the maximum velocity in the case of constant acceleration.

Here we want to raise a practical interesting question. What can be said about the spatial and temporal sampling if only the sampled time series is observed and the physical sampling distances $\Delta_x = \frac{2\pi}{\xi_i}$ and $\Delta_t = \frac{2\pi}{\xi_t}$ could not be recorded or are lost? This is for example the case if a time series is inspected on the screen from one time frame to the next. Suppose the series is sampled with sampling frequency ξ_s according to the Nyquist theorem, where b is the band-limit, then

$$\xi_s > 2b \;\leftrightarrow\; \frac{\xi_s}{2b} = 1 + \varepsilon,\ \varepsilon > 0\,, \tag{3.31}$$

where critical sampling corresponds to $\varepsilon \searrow 0$. Let Δ_x and Δ_t be the spatial and temporal sampling interval, then substituting eq.(3.31) in eq.(3.30) we obtain for the velocity in these sampling units

$$v = \frac{b_t}{b_x} \tag{3.32}$$

$$= \frac{(1+\varepsilon_x)}{(1+\varepsilon_t)}\frac{\Delta_x}{\Delta_t}\,. \tag{3.33}$$

From this we can immediately draw the conclusion that critical sampling ($\varepsilon \searrow 0$) implies $v < 1\Delta_x/\Delta_t = 1$pixel/frame for the objects in a time series. If we observe velocities $v > 1\Delta_x/\Delta_t$ and the series should be properly sampled then this implies $\varepsilon_x > \varepsilon_t$, i.e. we need spatial oversampling to a higher degree than temporal sampling.

Note: Observation of (apparent) velocities that are smaller than one pixel per frame in a sampled image sequence does not guarantee proper sampling, as temporal aliasing could already have corrupted the signal.

Measurement of velocity/acceleration is performed by applying a (combination of) first/second order Gaussian derivatives. The Gaussian and its derivatives are low-pass filters, all having the same approximated band-limit $b_G = 3/\sigma$. The Nyquist sampling theorem requires sampling as

$$\xi_{i,s} > 2b_i \; i = 1 \ldots n \,. \tag{3.7}$$

A possibility to recover the unaliased part of an aliased signal is to use a filter ξ_f

$$\xi_f \leq \xi_s - b, \text{ with } b < \xi_s < 2b \tag{3.10}$$

before sampling, where b is the band-limit. For a temporally aliased signal with temporal band-limit b_t and temporal sampling frequency $\xi_{s,t}$ this reads for a temporal Gaussian filter $\xi_f = 3/\sigma_t$

$$\begin{aligned} \xi_f &\leq \xi_{s,t} - b_t & (3.34)\\ 3/\sigma_t &\leq \xi_{s,t} - \max(v_i b_i) & (3.35)\\ \sigma_t &\geq \frac{3}{\xi_{s,t} - \max(v_i b_i)} \,. & (3.36) \end{aligned}$$

In practice the spatial band-limit is often known and an assumption of the maximal velocity (based on the underlying physics) can be made, then the temporal band-limit can be calculated from $b_t = \max(v_i b_i)$. Eq.(3.36) provides a helpful restriction on the choice of σ_t.

For a line spectrum, e.g. the two dimensional spectrum of uniform rectilinear motion shown in figure 3.4, the restriction on the recovery of the unaliased part can even be loosened to the following. All but one axis can be sampled with $\xi_s < b$. The DC component $\vec{0}$ must be preserved, but the $(\omega, 0)$ or $(0, k)$ axis can be partly aliased. The restriction on the temporal $\xi_{f,t}$ and spatial filter $\xi_{f,x}$ can be taken from figure 3.5:

$$\xi_{f,t}\xi_{f,x} \leq \min(\xi_{s,t}, \xi_{s,t}/v) \tag{3.37}$$

where without loss of generality the time axis is undersampled with $\xi_{s,t} < b_t$. The area of the rectangle spanned by $(\xi_{f,t}, \xi_{f,x}), (-\xi_{f,t}, \xi_{f,x})(-\xi_{f,t}, -\xi_{f,x})(\xi_{f,t}, -\xi_{f,x})$ must be confined in the diamond $(\xi_{s,t}/v, 0), (0, \xi_{s,t}), (-\xi_{s,t}/v, 0), (0, -\xi_{s,t})$.

3.2.2 The correspondence problem

The *correspondence problem* refers to all sorts of identifying and following tasks of individual objects in time. The following thought experiment explains the problem.

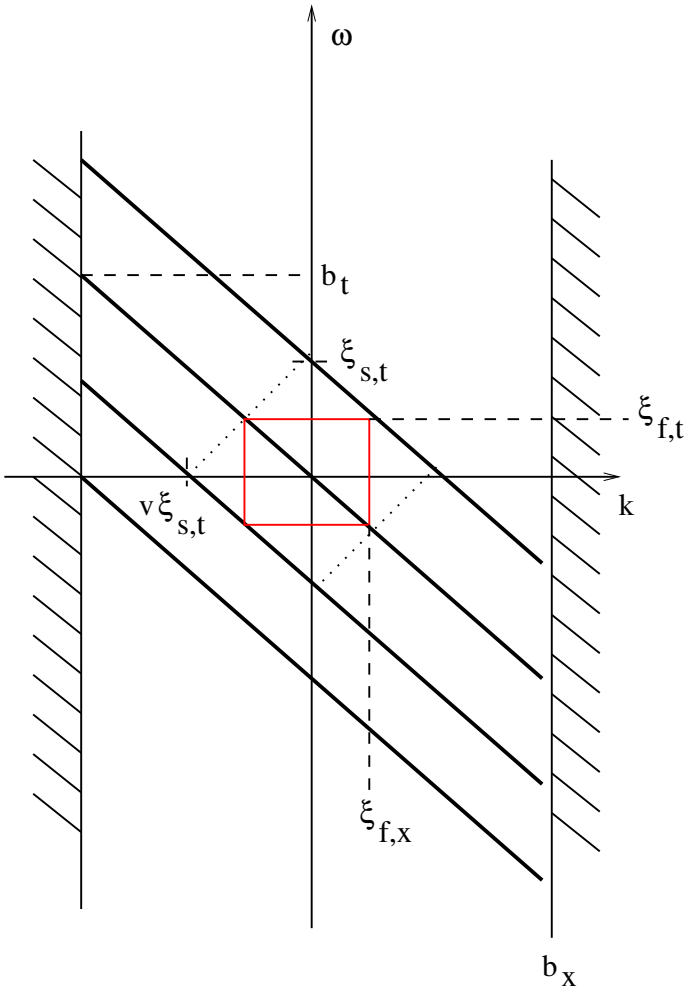


Figure 3.5: Temporal undersampling by $\xi_{s,t} < b_t$ of a time series with line spectrum. Part of the signal can still be recovered if a spatio-temporal filter with $\xi_{f,t}, \xi_{f,x}$ is applied which footprint lies within the diamond spanned by $(\xi_{s,t}/v, 0), (0, \xi_{s,t}), (-\xi_{s,t}/v, 0), (0, -\xi_{s,t})$. One possible filter is indicated here by the rectangle in the middle.

Imagine that you have twenty glass marbles placed on a table, take a picture and let them roll off the table on the ground. Now try to place the marbles back on the table using the photo in exactly the same position. Even if all marbles are different it will take some time to do it. However, some marbles are not distinguishable at all by sight and/or touch. In that case the task is not solvable. This problem is encountered in image processing of time dependent data (here we usually have more than only two observations). The marbles are the equivalent of bright blob-like spots in living-cell confocal microscopy. The different blobs may be characterized by their peak intensity and shape [8]. This tracking by fitting certain shape parameters is computationally complex. The blobs may change shape during motion and different blobs can even merge to one or blobs may split. Taking care of theses exceptions makes tracking for these image types a difficult task.

Finding correspondences is easier as long as all objects are connected via a path in the (x,t)-image. For critical sampling the maximal velocity is $v < 1\Delta_x/\Delta_t = 1\text{pixel/frame}$ and thus the objects are connected, however, objects may still touch and cross each other. Perfect correspondence without the need to model each blob is only given if the (characteristic) spatial distance λ_s between two blobs is larger than the shift between two time frames

$$\lambda_s > v\Delta_t\,. \tag{3.38}$$

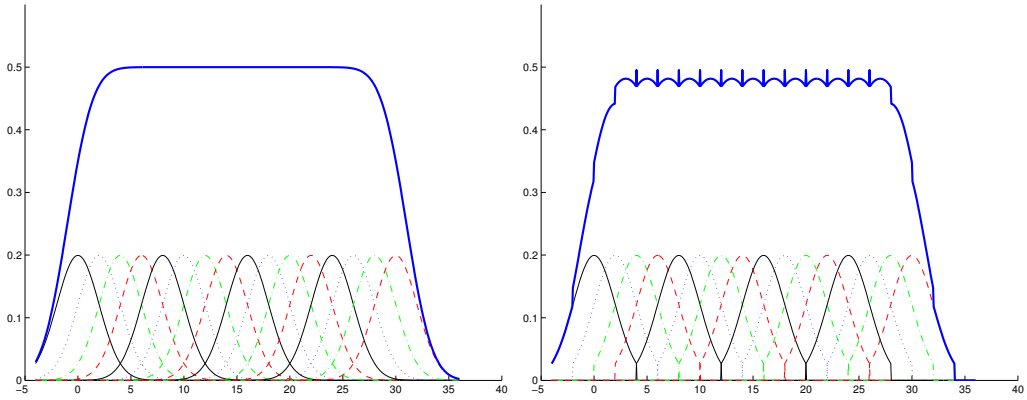
If the trajectories in the (x,t)-image are not connected, a pre-processing step can sometimes connect the trajectory of one object. Therefore the time axis has to be resampled (inserting time frames which allow for a connected trajectory). In chapter 8 we will use adaptive filtering to compensate for varying intensity along a trajectory. This 'dotted' trajectory is, however, not caused by temporal undersampling but by objects moving in and out of focus. A morphological approach presented by Okker [94] to fill in missing parts of static structures could be used for this problem as well. If the measurement of velocities with Gaussian filters is not possible due to temporal undersampling, then tracking and computing the velocities from the center of mass positions is more feasible.

3.2.3 Time band-limitation by acquisition

As discussed, spatial band-limitation induces temporal band-limitation (a restriction on the maximal apparent velocity). Standard temporal acquisition does not introduce a cut-off frequency analog to an optical lens system ($2NA\lambda$). In the spatial domain the Point Spread Function (PSF) blurs a point source and thereby creates an overlap of intensities from different locations. The question rises whether the same can be incorporated into a camera system for temporal acquisition? This can be answered affirmatively, but not with *one* camera. As a sampled time point should have an overlap with its neighbors, more cameras have to be used. In the following, we formulate a concept idea to construct such an acquisition device with minimal loss of light intensity [134].

The intensity during the exposure time should be blurred by a (temporal) PSF. The PSF can be very well approximated by a Gaussian [132] p.28/29. If we truncate the Gaussian at 2σ then four cameras are required to produce 2σ overlap. In figure 3.6 we show four reoccurring Gaussians with peak-to-peak distance σ. In a) ideal Gaussians are plotted, and in b) the effects of the truncation are shown. The sum of the Gaussians is constant as can be seen from the following consideration. One Gaussian is described by

$$g_i(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t-i\sigma)^2}{2\sigma^2}},\ i \in \mathbb{N} \tag{3.39}$$



(a) **(b)**

Figure 3.6: Four reoccurring Gaussians with $\sigma = 2$ at sampling distance 1σ and their sum, a) ideal, b) after truncation of the Gaussians at 2σ.

and the sum is (neglecting border effects)

$$s(t) = \sum_i g_i(t) \tag{3.40}$$

$$= \sum_i g_0(t \pm i\sigma) \approx \frac{1}{\sigma}\,. \tag{3.41}$$

The last line is the formula for numerical integration for the normalized Gaussian without the step size $\Delta = \sigma$

$$1 = \int g(t)dt \approx \sum_j g(t \pm j\Delta)\Delta\,. \tag{3.42}$$

From here we see that it is theoretically possible to distribute an incoming signal to four Gaussian distributions without losing signal in the acquisition if we choose $\sigma = 1$.

Experimental setup

Here we formulate an idea for an experimental setup which has four cameras and can acquire time signals with a Gaussian distributed intensity, time delayed by one σ with minimal loss of photons. A beamsplitter setup with Gaussian diffusers (for example by optical active crystals) as in figure 3.7a) is possible but then each camera only receives about 1/8 of the incoming intensity. An advantage is, however, that this setup is truly time overlapping. We propose a different setup with three mirrors as shown in figure 3.7b). The idea is to realize the Gaussian intensity profiles by dithering.

Mirrors that can be switched fast, reflect light in time intervals Δ_d to the different cameras according to the overlap at that time point. Where Δ_d is limited by the switching time of the mirrors. Form figure 3.7b) we see how the incoming light

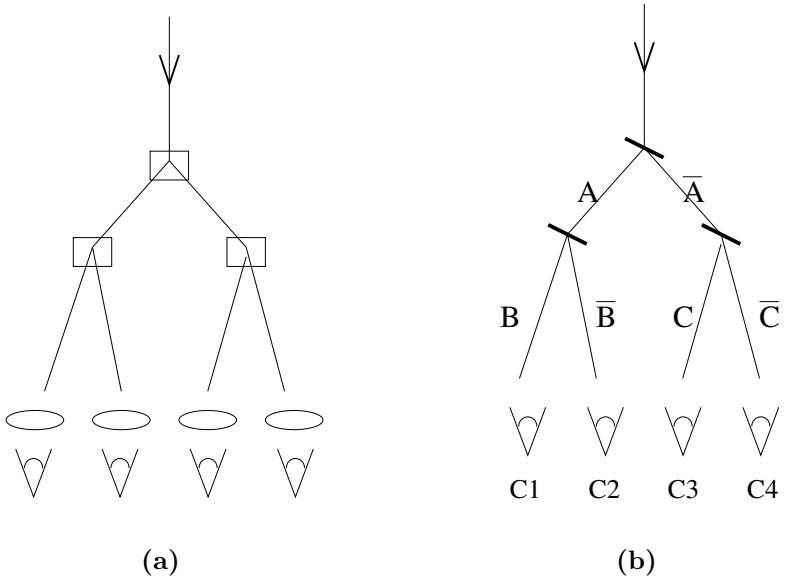


Figure 3.7: a) Beamsplitter setup with Gaussian diffusers at each splitting, b) Setup with switchable mirrors, for example a Digital Mirror Device from Texas Instruments can be used here. They have about 70% light efficiency (including area fill factor) and can switch a whole page at about 10 kHz and single mirrors at 66 kHz[119].

is divided over the four cameras C1-C4

$$\mathrm{C1} = AB \qquad \mathrm{C3} = \bar{A}C \tag{3.43}$$
$$\mathrm{C2} = A\bar{B} \qquad \mathrm{C4} = \bar{A}\bar{C}\,. \tag{3.44}$$

We truncate the Gaussian $g_i, i = 1 \ldots 4$ at 2σ and make them for each of the four cameras periodic with period 4σ. So, one Gaussian represents one camera $g_i(x) = g_i(x + 4\sigma)$. The total amount of incoming light must be distributed at t_0 as

$$1 = \sum_{i=1}^{4} g_i(t_0) \tag{3.45}$$
$$= g_1(t_0) + g_1(t_0 - \sigma) + g_1(t_0 - 2\sigma) + g_1(t_0 + \sigma)\,. \tag{3.46}$$

The ratio of switching of the mirrors is

$$B(t): \quad \frac{g_1(t)}{g_2(t)} \tag{3.47}$$

$$C(t): \quad \frac{g_3(t)}{g_4(t)} \tag{3.48}$$

$$A(t): \quad \frac{g_1(t)+g_2(t)}{g_3(t)+g_4(t)}\,. \tag{3.49}$$

An optimal dithering pattern still has to be found for the set of g_i according to the ratios given above.

Chapter 4

Curvature of n-dimensional space curves[4]

Local curvature represents an important shape parameter of space curves which are well described by differential geometry. Curvature is the change of orientation, for which we have developed a continuous representation in chapter 2. Here we present an estimator for local curvature of space curves embedded in nD grey-value images. The space curves are implicitly represented by grey-level isophotes (level-sets). There is neither a segmentation of the curve needed nor a parametric model assumed. The method exploits the differential structure of images. Our estimator works on the orientation field of the space curve. This orientation field and a description of local structure is obtained by the Gradient Structure Tensor (GST). The computed orientation field has discontinuities; walking around a closed contour yields two such discontinuities in orientation, see figure 2.1b). This field is mapped via the Knutsson mapping eq.(2.4) to a continuous representation; from a nD vector to a symmetric n^2D tensor field. The curvature of a space curve, a coordinate invariant property, is computed in this tensor field representation. An extensive evaluation shows that our curvature estimation is unbiased even in the presence of noise, independent of the scale of the object and furthermore the relative error stays small.

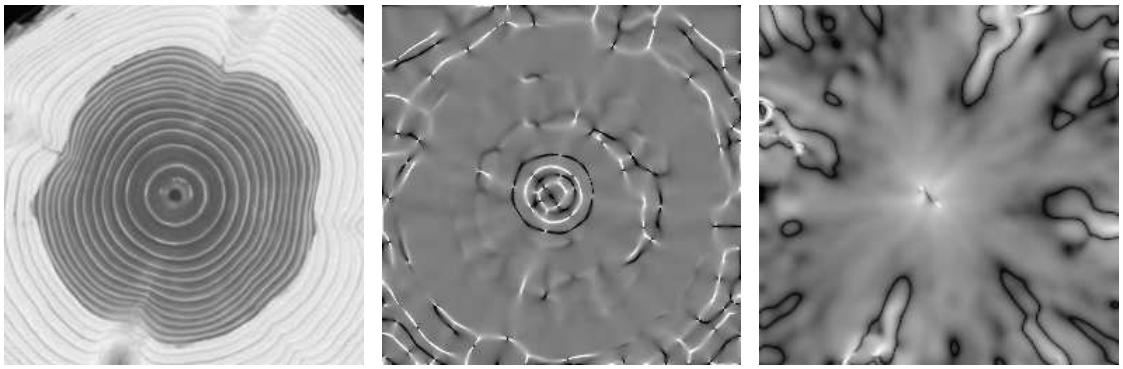
Curvature in 2D images has been studied extensively , both in segmented and in grey-level images [43, 90, 56, 147, 146]. Curvature is the first order shape descriptor of an object and therefore an important feature. In 2D it totally determines the shape of a curve. Isophote curvature [122, 16, 135] can successfully be applied to edges in 2D and 3D grey-value images, but it fails when applied to lines (space

[4]The main content of this chapter has been published in: B. Rieger and L.J. van Vliet, Curvature of n-dimensional space curves in grey-value images, *IEEE Transactions on Image Processing*, 11(7):738-745, 2002 [103]. New material in sections 4.2.5, 4.3 and 4.4.

curves) or line patterns [140]. Isophote curvature in 2D is computed as [132, 56]

$$\kappa = \frac{-(I_{xx}I_y^2 - 2I_xI_yI_{xy} + I_{yy}I_x^2)}{(I_x^2 + I_y^2)^{3/2}}\,. \tag{4.1}$$

A lower index denotes a partial derivative with respect to that coordinate. For regions where the denominator (essentially the gradient magnitude) is zero this formula does not hold. This is the case on ridges and in valleys. Furthermore, the curvature will have a different sign on both sides of the slope. This problem becomes even more apparent if we deal with a line pattern, the scale at which the isophote curvature is computed cannot be increased for noise suppression without smoothing out the pattern we are interested in. Let us consider, for the sake of simplicity, a 2D example. In figure 4.1 the isophote curvature and the new proposed algorithm are applied to a natural line pattern image. For this example in 2D the curvature could also have been computed differently than by our proposed method, see for example [91, 140, 42, 5, 126].



(a) **(b)** **(c)**

Figure 4.1: a) Image of growing rings of a tree, b) isophote curvature with $\sigma = 5$, range [-1, 1], c) new curvature estimator as in eq. 4.7 $\sigma_g = 1, \sigma_T = 5$, log stretch.

In our approach we transform the grey-value image into an orientation map to overcome the problems associated with isophote curvature. From the orientation map the curvature can be derived after solving the discontinuity problem [43]. In 2D the use of the double angle method is well-known [43, 63], but in 3D it remained an obstacle that prevented the computation of curvature in 3D. Therefore traditional 3D methods are applied to segmented images, or even on curves represented by ordered points, which enables one to fit a parametric model to the curve [15, 81, 22]. These methods rely heavily on the preceding segmentation, labeling and orderings steps, which may fail due to noise or the presence of a bundle of space curves comparable to in a lock of hair or the wood growing rings in figure 4.1.

Isophote curvature in 3D has been studied by van Vliet and Verbeek [135], but the isophote curvature fails on lines and plates as illustrated above. A scale-space related segmentation free method for curvature has been presented by Salden et al. [110]. They do not consider the localization problem of the derivative operator and assume global direction information. Bakker et al. have introduced a segmentation free method in 3D to compute curvature of lines or line bundles based on a local parabolic model of the GST [5, 4]. We will compare our results to this approach in section 4.4.

In 2D, the curvature κ of a curve in every point describes the shape of this curve completely. In 3D, a second parameter, the torsion τ, is needed to give a full description, in 4D another and so forth. These parameters totally determine the shape of a space curve but do not tell anything about its position. This makes these parameters well suited as curve descriptors. The curvature κ is a first order feature of a nD curve, describing the second order change. The curvature of a space curve $k : \mathbb{R} \to \mathbb{R}^n$ is given by (see appendix B)

$$\kappa = \|k''\|\,, \tag{4.2}$$

where $'$ is the derivative with respect to the arclength. In words, eq.(4.2) states that the curvature is the magnitude of the second order change of the curve along the curve. Along the curve means in tangent orientation. See appendix B for an introduction to the mathematics of differential geometry that describe space curves or the textbooks [28, 116, 39].

From eq.(4.2) we see that the curvature is always greater or equal to zero. Indeed for space curves it does not make sense to speak of a signed curvature in a coordinate independent description. In contrary to closed surfaces there is no border separating two distinct parts of space. By choosing an origin one can speak of signed curvature also for space curves.

4.1 Curvature of space curves in grey-value images

A local orthonormal basis $\{v_i\}$ attached to the space curve can be obtained from the grey-level images itself by local orientation analysis.

4.1.1 The orientation field; a local orthonormal basis

In order to obtain the orientation field along a space curve embedded in a nD image we use the GST [60, 10, 64, 61, 151, 56]. See appendix A for more information about the GST. For any image I we can compute it as:

$$\bar{G} := \overline{\nabla I \nabla I^t}\,. \tag{4.3}$$

The overhead bar denotes smoothing which is done per element $\bar{G}_{ij}$, where each element is a nD image. The GST can be expanded in terms of the eigenvalues λ_i and eigenvectors u_i as

$$\bar{G} = \sum_{i}^{n} \lambda_i u_i u_i^t, \text{ with } \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \,. \tag{4.4}$$

The eigenvectors of the GST contain information about the local structure in the image. We can compute the largest λ_1 and smallest eigenvalue λ_n and the associated eigenvector for any dimension of the image by using the power method [44]. For the smallest eigenvalue $\bar{G}$ has to be inverted, which becomes time consuming for large n. In the 2D or 3D case analytic solutions are possible and much faster [42]. For line-like structures the tangent orientation is given by the "smallest" eigenvector u_n. A normalized line detector is the ratio $\frac{\lambda_{n-1}-\lambda_n}{\lambda_{n-1}+\lambda_n}$ [42, 61].

All derivatives are implemented as convolutions with Gaussian derivatives [83]. The scale σ_g denotes the resolution at which the Gaussian derivatives are computed. The size of the tensor smoothing σ_T of $\bar{G}$ defines how local the image structure is computed. The local set of eigenvectors $\{u_i\}$ of the GST consists of the same set of vectors $\{v_i\}$ (up to sign) as the local orthonormal basis. The ordering, however, is different.

4.1.2 Discontinuity of the orientation field

Unfortunately the calculated orientation field u_n contains a discontinuity mod π [64], i.e. the direction of the line is undefined. Computation of partial spatial derivatives of the orientation field are not possible without some preparation. In general, a mapping to a higher dimensional space is needed to solve the discontinuity problem. For example in 2D, the phase jump can be resolved by doubling the angle of the gradient vector [43, 63]. In higher dimensions the Knutsson mapping

$$M(x) = \frac{xx^t}{\|x\|} \,. \tag{2.4}$$

can be used as described in chapter 2. The Knutsson mapping removes the phase jump by the property $M(x) = M(-x)$. Vectors that are pointing in the opposite directions are mapped onto the same point. Furthermore, M is distance preserving and relates the magnitude of a variational input vector linearly with the variation of a mapped vector by $\|\delta M(x)\| = \sqrt{2}\|\delta x\|$. This is called the *uniform stretch property*.

4.1.3 Curvature in nD grey-value images

Our goal is to compute the curvature via eq.(4.2) $\kappa = \|k''\|$. We start with the tangent orientation $u_n \equiv T \equiv k'$ obtained with the GST. Here, we already have the first derivative. The discontinuity problem (discussed earlier) prevents direct computation of the derivative of T along u_n. This is solved by mapping the tangent orientation T via the quadratic mapping M to a continuous representation. The elements of $M(T)$ form a new n^2D vector w. The ordering of the elements in this vector w does not influence our curvature estimation, because we only evaluate a norm as shown in eq.(4.7) which is independent of a permutation of the elements of w.

Now, we calculate the derivative of w in the direction of the tangent T, which is again a n^2D vector [17]

$$\frac{\partial w}{\partial T} = \mathbf{DM} \cdot T \,, \tag{4.5}$$

where DM is the $n^2 \times n$ functional matrix

$$\mathbf{DM} = \begin{pmatrix} \frac{\partial w_1}{\partial x_1} & \frac{\partial w_1}{\partial x_2} & \cdots & \frac{\partial w_1}{\partial x_n} \\ \vdots & & & \vdots \\ \frac{\partial w_{n^2}}{\partial x_1} & \frac{\partial w_{n^2}}{\partial x_2} & \cdots & \frac{\partial w_{n^2}}{\partial x_n} \end{pmatrix} . \tag{4.6}$$

From the uniform stretch requirement of M, $\|\delta M(x)\| = c\|\delta x\|$, (see chapter 2.1) we know how to scale the norm of a variation vector δx. Thus, starting from eq.(4.2) and the mapping $M(T)$ we obtain a new expression for the curvature

$$\kappa = \frac{1}{\sqrt{2}} \left\| \frac{\partial M(T)}{\partial T} \right\| . \tag{4.7}$$

This reads explicitly as

$$\kappa = \frac{1}{\sqrt{2}} \left(\sum_{i,j}^{n} \left[\sum_{k=1}^{n} \frac{\partial M_{ij}}{\partial x^k} T^k \right]^2 \right)^{1/2} . \tag{4.8}$$

This formula can directly be applied to grey-value images.

We are aware of the fact that we cannot take it for granted that eq.(4.7) represents the curvature of the original curve. First, we apply a non-linear mapping M to the tangent orientation T and then in the mapped space a derivative and a projection are done. The reason why eq.(4.7) is indeed another form of eq.(4.2), is the uniform stretch requirement imposed on the mapping M, that ensures that there is a fixed relation between the norms of the mapped and the original vector. For a formal proof that eq.(4.7) equals eq.(4.2) see appendix E.

In chapter 5 the same procedure will be used to compute the curvature on surfaces which will lead to eq.(5.12).

4.1.4 The method in 2D

Let us assume that the 2D tangent orientation is given $T = \begin{pmatrix} f(x,y) \\ g(x,y) \end{pmatrix}$, where $f, g \in \mathcal{C}^1 : \mathbb{R}^2 \to \mathbb{R}$ are arbitrary functions. The mapped vector w is

$$w = \frac{1}{\sqrt{f^2+g^2}} \begin{pmatrix} f^2 \\ fg \\ gf \\ g^2 \end{pmatrix} . \tag{4.9}$$

Having the vector w explicitly, we construct the functional matrix, compute the projection onto the tangent direction (4.5) and then we can calculated the curvature using eq.(4.7)

$$\kappa = \left| \frac{gf(\partial_x f) - f^2(\partial_x g) + g^2(\partial_y f) - fg(\partial_y g)}{f^2+g^2} \right| . \tag{4.10}$$

If we have the 2D orientation field $\phi(x,y)$ given, then the tangent is

$$T = \begin{pmatrix} -\sin\phi(x,y) \\ \cos\phi(x,y) \end{pmatrix} . \tag{4.11}$$

Filling in this tangent in eq.(4.10) and simplifying the expression with the help of the trigonometric relations, we get

$$\kappa = \left| \sin\phi \frac{\partial\phi}{\partial x} - \cos\phi \frac{\partial\phi}{\partial y} \right| = \left| \frac{\partial\phi}{\partial c} \right| , \tag{4.12}$$

where c is a local coordinate along the contour (level curve).

At this point we are able to state that our method of calculating the curvature via the mapping in a higher dimensional space is consistent with the standard definition of curvature in 2D. Also the isophote curvature formula in 2D can be obtained by eq.(4.10) with the gradient being $g = (I_x, I_y)$, the contour $c = (-I_y, I_x)$ and therefore the isophote tangent

$$T = \frac{c}{\|c\|} = \frac{(-I_y, I_x)}{\sqrt{I_x^2 + I_y^2}} . \tag{4.13}$$

The indices being partial derivatives. Filling in this tangent in eq.(4.10), we get the isophote curvature formula [135, 16, 43]

$$\Rightarrow \kappa = \left| \frac{-(I_{xx} I_y^2 - 2 I_x I_y I_{xy} + I_{yy} I_x^2)}{(I_x^2 + I_y^2)^{3/2}} \right| . \tag{4.14}$$

4.2 Evaluation of the algorithm

To assess the performance of the proposed estimator we will apply it to synthetic test images of various scales and hampered by noise.

4.2.1 Test images

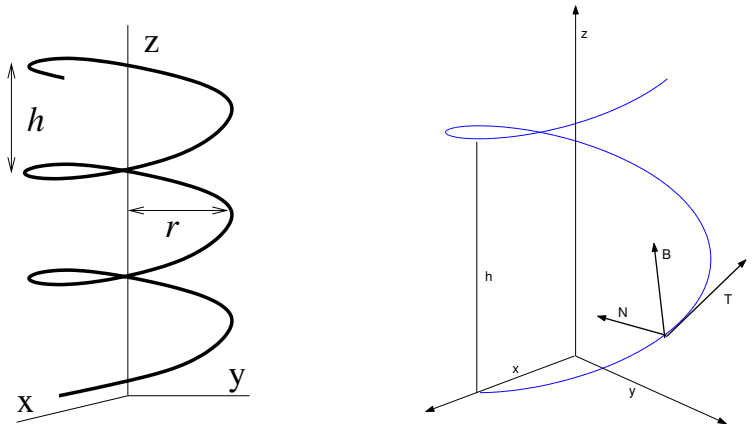
Definition: Under a space curve in a grey-value image we understand a line of constant grey-value, i.e. an isophote (level-set). It should be noted that such isophotes are implicitly represented by the voxel-values of a properly sampled band-limited image.

To evaluate the algorithm on space curves, some proper test images are needed, that reflect this definition. A general method of creating smooth and approximately band-limited space curves in images is described in the following. One has to consider that a suitable test image must be band-limited before sampling, otherwise one might encounter problems due to aliasing [135]. We do this by computing the distance from every voxel in the image to the mathematical function and assign it to the voxel. In the next step, the image is multiplied by itself to create a steeper slope; subsequently, an erfclip operation [136] is applied to produce a smooth space curve, embedded in an image.

In 2D we use a simple ring as a test image. In 3D, as a first step we create a torus of arbitrary orientation. In a second step we study the simplest possible test object in 3D not only having curvature. The object with constant curvature and torsion is the circular helix eq.(4.15). So we finally get a smooth representation of a torus/helix. This guarantees a sub-pixel precision and approximately band-limitation. The image consists of an isophote (same grey level) center line and isocylinders around it. In figure 4.2 the center line of a grey-level helix is shown, where the line indicates constant grey-value. The vectors given in the figure represent the local orthonormal basis. In figure 4.3 an isosurface plot of the smooth string shaped into a spaghetti is shown. The isocap through the spaghetti shows the isocylinder around the imaginary center line. Furthermore, the cross-section through the object is shown. In figure 4.4 a 3D grey-value torus is shown with contour planes to indicate the grey-level.

4.2.2 Ring

To start off in 2D we generate smooth rings with different radii and signal strength $S = 1$. To measure the robustness of the estimator we add different levels of Gaussian noise. We use the definition SNR$= 20 \log \frac{S}{\sigma_n}$, where σ_n^2 is the variance of the Gaussian noise. The results are depicted in figure 4.5. The different noise



(a) (b)

Figure 4.2: a) Sketch of a circular helix with parameters r, h, b) Center line of a grey-level helix with local coordinate system, the tangent T , normal N and binormal vector B.

levels are always calculated at the same radii, but slightly shifted in the figure for a better display. The error bars indicate the standard deviation over 40 runs. A tensor smoothing $\sigma_T = 5$ and a gradient smoothing $\sigma_g = 1$ was used. For SNR=10 dB the average relative error is smaller than 10%. The estimator is unbiased.

4.2.3 Torus

In figure 4.6 we show three torii with respectively 20,10 and 3dB noise and again we use $\sigma_T = 5$ and $\sigma_g = 1$. The performance of the estimator is tested over 20 runs for different noise levels, see figure 4.7. The error bars indicate the standard deviation. The different noise levels are again shifted in the figure for better display. The error bars include even for the high noise level (3 dB) the true value.

4.2.4 Circular helix

The curvature

$$\kappa = \frac{r}{r^2 + h^2} \tag{4.15}$$

of a circular helix given by the parametrization $(r\cos t, r\sin t, ht)$ depends on the two parameters r and h which scale the helix. See appendix B for a deduction of eq.(4.15). For increasing size of the helix radius the curvature first rises, being at its maximum at $r = h$ and then decreases (figure 4.8). In order to make a scale

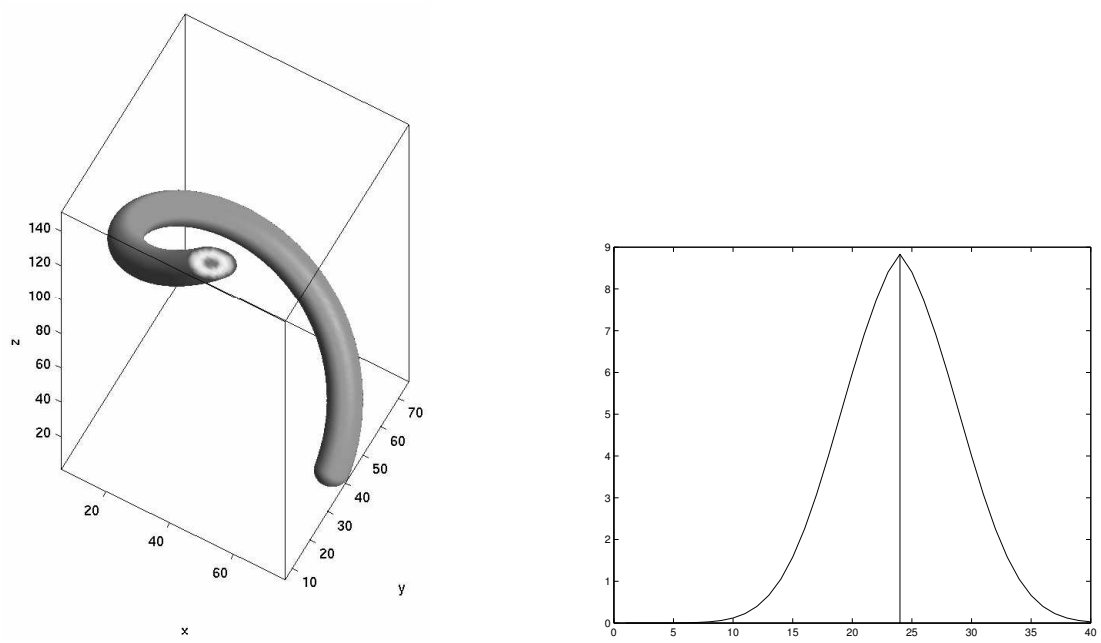


Figure 4.3: a) Isosurface plot of a grey-level helix, b) a cross-section through the test object.

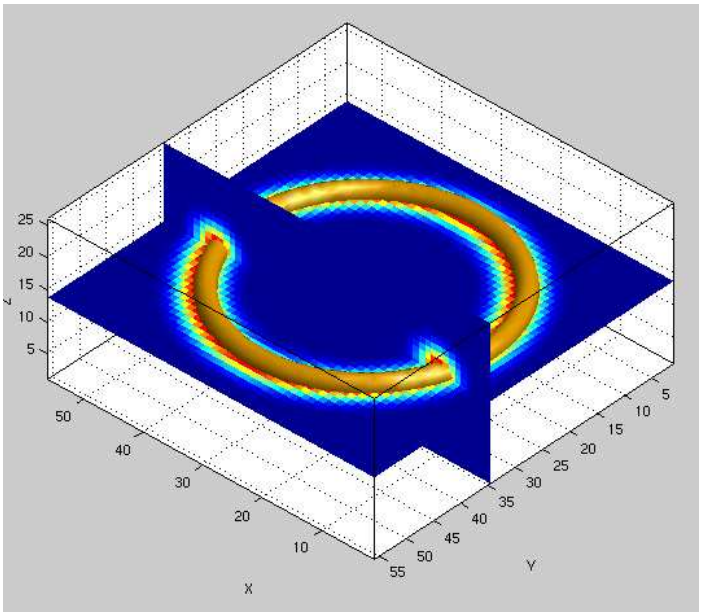


Figure 4.4: Surface plot with contour planes of a grey-level 3D torus.

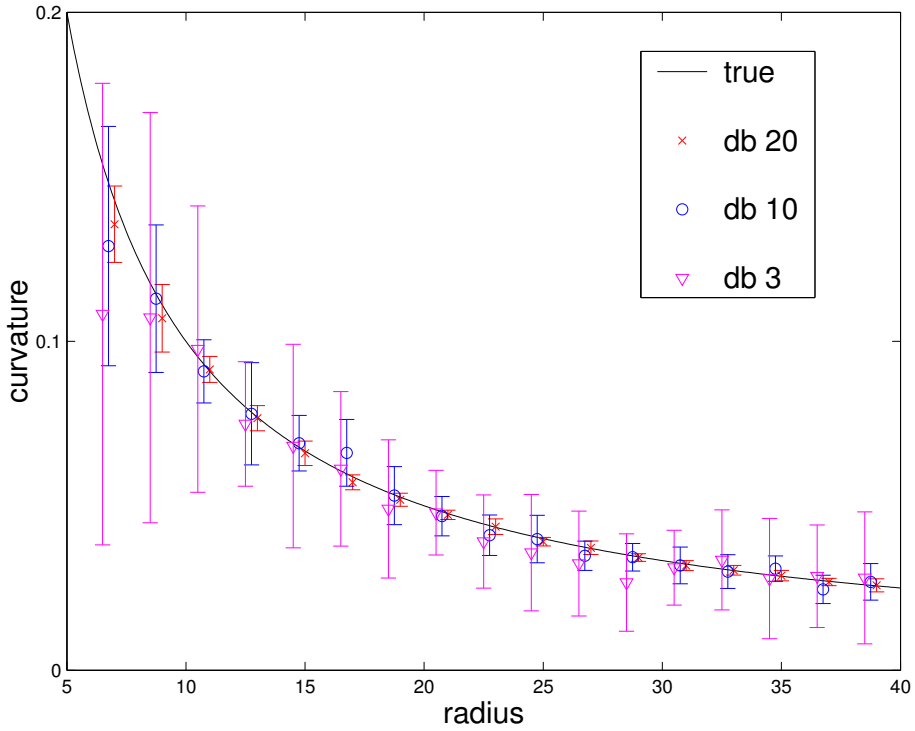
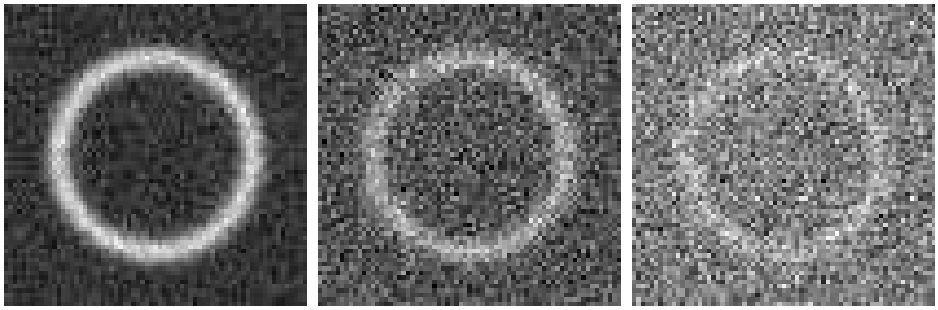


Figure 4.5: Average curvature estimation on a 2D ring for different noise levels over 40 runs with $\sigma_g = 1$ and $\sigma_T = 5$.



(a) **(b)** **(c)**

Figure 4.6: Cross section through a 3D torus with different noise levels. a) 20dB, b) 10dB, c) 3dB.

invariant statement about the performance of our algorithm we sample the scale space (r, h), generate the according test images and compute the curvature of the center line. Therefore we rewrite eq.(4.15) to

$$\kappa = \frac{\frac{r}{h^2}}{1 + \frac{r^2}{h^2}} \text{ and } \kappa r = \frac{x^2}{1 + x^2}, \; x = \frac{r}{h}\,. \tag{4.16}$$

Now, we deal with dimensionless quantities x and κr, which are suitable to show

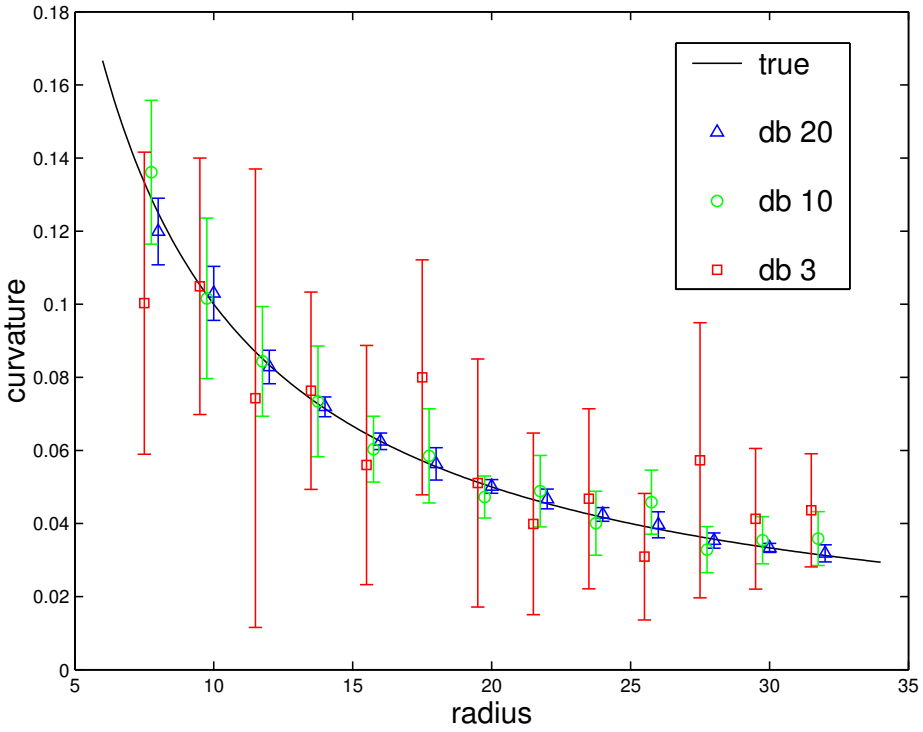


Figure 4.7: Average curvature estimation on a torus for different noise levels over 20 runs with $\sigma_g = 1$ and $\sigma_T = 5$.

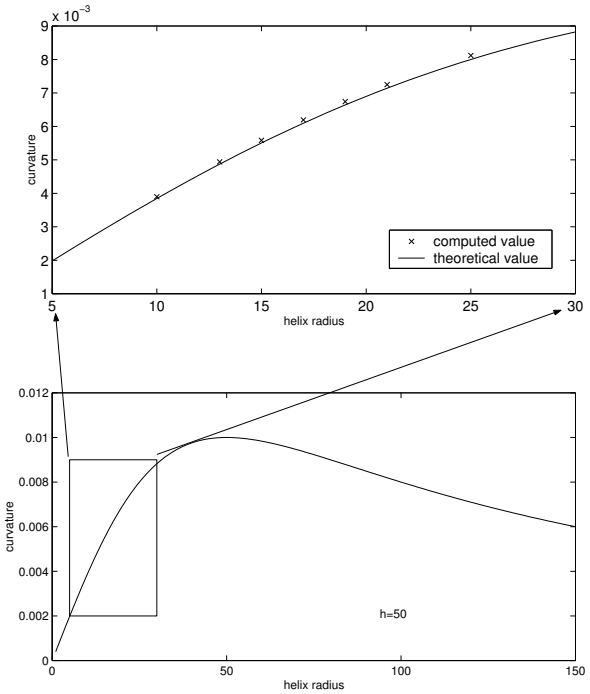


Figure 4.8: Change of the curvature with the radius r of a helix.

that our estimation works fine over a wide range of scales. In figure 4.9 we plot the theoretical prediction and our calculations, in which the different symbols indicate helices having either the same radius or pitch. Because our estimation stays so close to the true value over a wide range of scales we conclude that for all values in between our sampled (r, h) points the estimation works as well. The relative error of the estimation that is shown in figure 4.9 is smaller than 3% over a range from $r/h \in [0.1, 6.0]$. For the bulk of estimations the error is even smaller than 1%, the exceptions are helices with small radii $r \approx 10$ and pitch $h \approx 15 - 20$. For these helices the tensor smoothing $\sigma_T = 4$ is too large, different parts of the helix fall in one smoothing window. The curvature values were computed for the whole image and extracted with sub-pixel precision by cubic interpolation from the mathematical centerline.

Influence of noise

• Noise for different helix scales
In the same scale invariant manner as for figure 4.9 we investigate the performance under noisy circumstances. In figure 4.10 the results are shown for 13, 19 and 25 dB, where the error bars indicate the standard deviation for 20 runs. The estimation is unbiased, since the error bars always intersect the true curvature.

• Studies along a cross-section of the helix profile
For one helix ($r = 10, h = 20$) we add noise (SNR=19dB) to the image and plot the computed curvature (mean over 20 runs) and the standard deviation in figure 4.11 along a cross-section of the helix (see figure 4.3). Again we see that the mean remains around the true value, and the variation stays approximately constant. It should be pointed out that we see here not only the isophote line at the point where the mathematical helix would lie but a cross-section through the 15 pixel diameter of the helix. Even away from the center line we compute nearly the helix curvature (value for this helix $\kappa = 10/(10^2 + 20^2)$, dashed line). Due to the regularization effect of the GST, we can estimate the true curvature even if we are not at the exact position. If we choose a small tensor smoothing σ_T, then the relative error at the exact point becomes smaller but for the surrounding values it becomes larger.

Influence of the Gaussian derivative in the mapped space

For a limited number of helices ($h = 50, r = 10, \ldots, 25$) we investigate the dependency of the relative error of the curvature on σ_K, the standard deviation of the Gaussian derivative in the mapped space eq.(4.5) (compare figure 4.12). Here we observe an overall σ_K^2 dependency in the relative error.

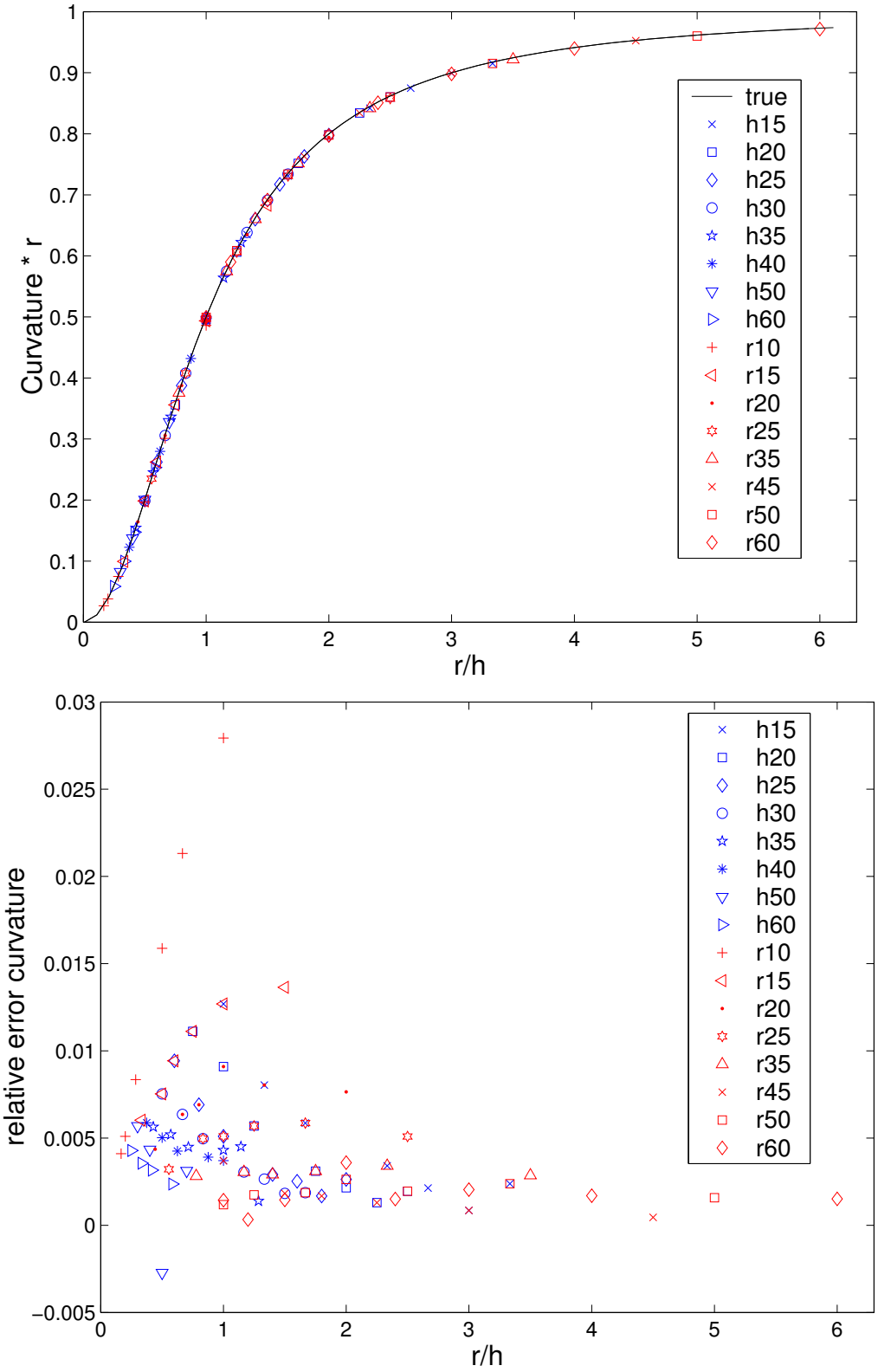


Figure 4.9: Curvature estimation of different helices and their relative error in the noise free case, scale invariant with $\sigma_g = 1, \sigma_T = 4$.

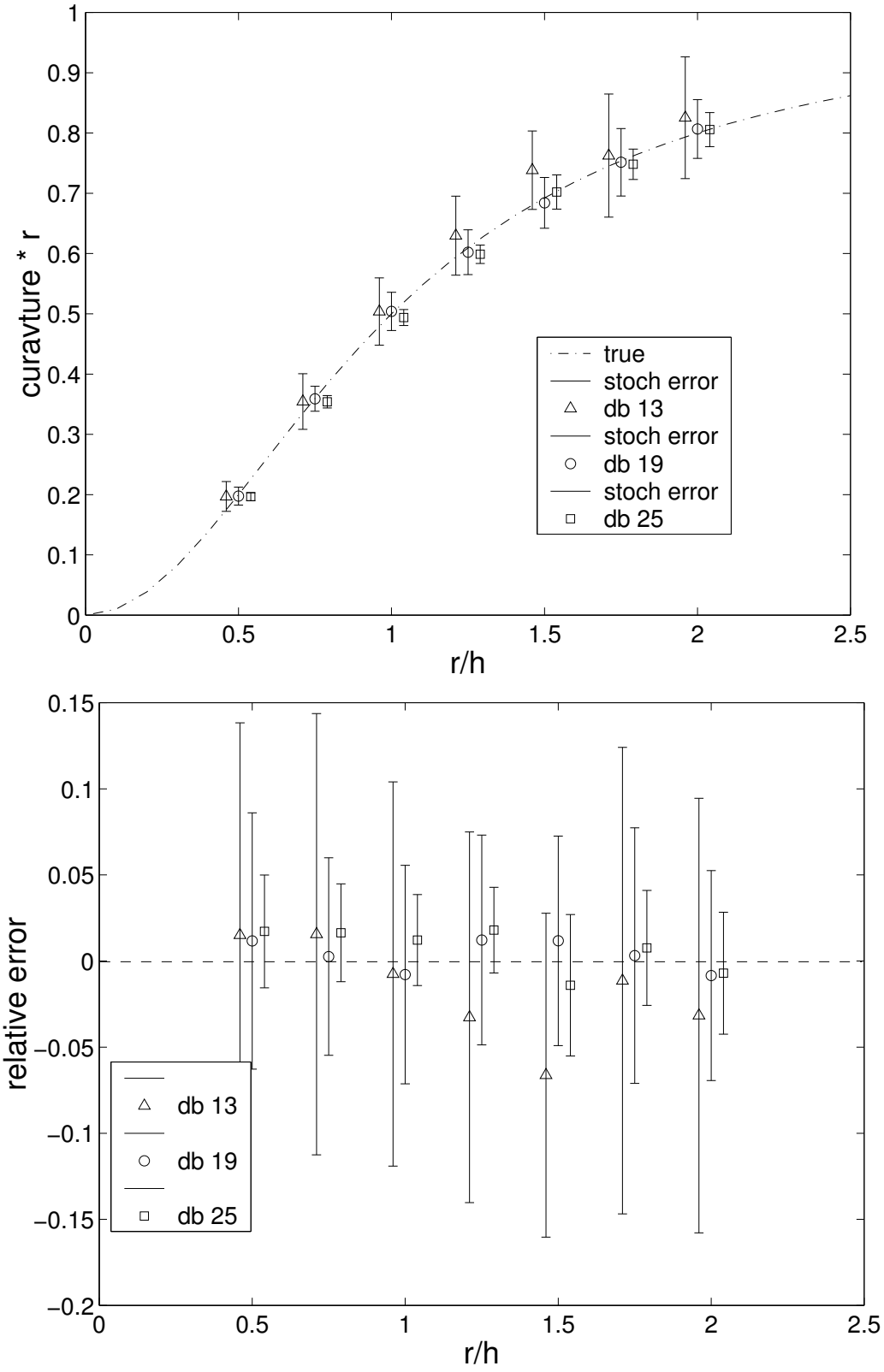


Figure 4.10: Curvature estimation and relative error with added noise of 13,19 and 25dB, scale invariant with $\sigma_g = 1, \sigma_T = 5$.

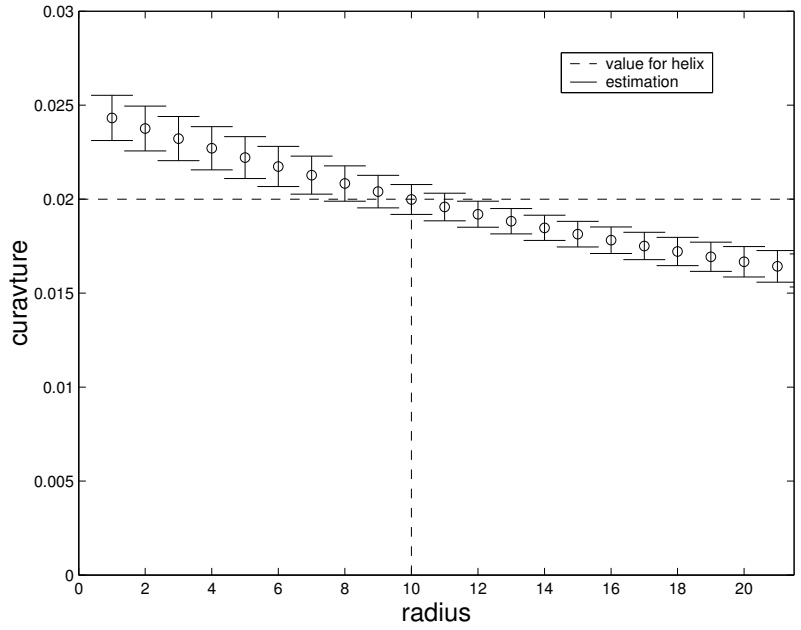


Figure 4.11: Curvature along a cross-section of a helix with $r = 10, h = 20$ with 19dB added Gaussian noise.

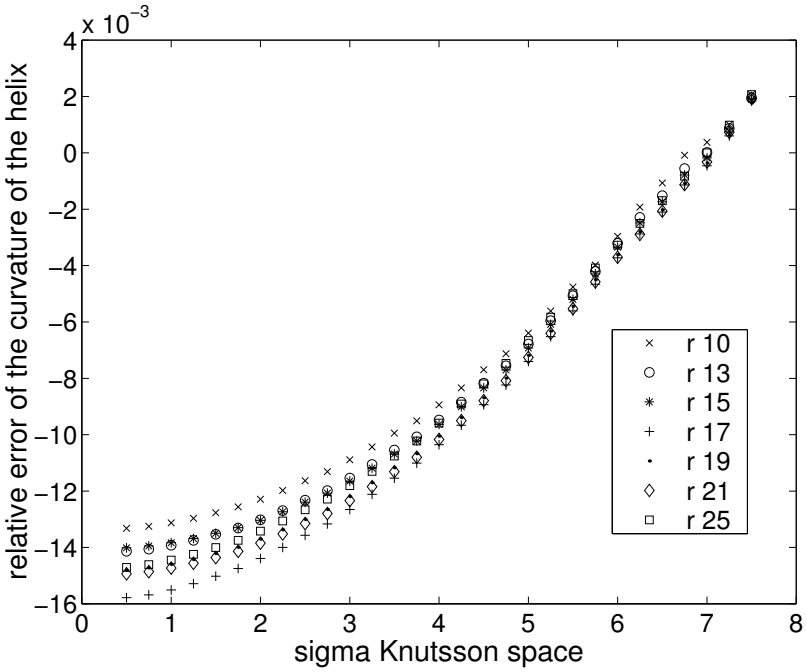


Figure 4.12: Influence of the derivative filter size σ_K in the Knutsson space on the relative error for different radii on noise free data.

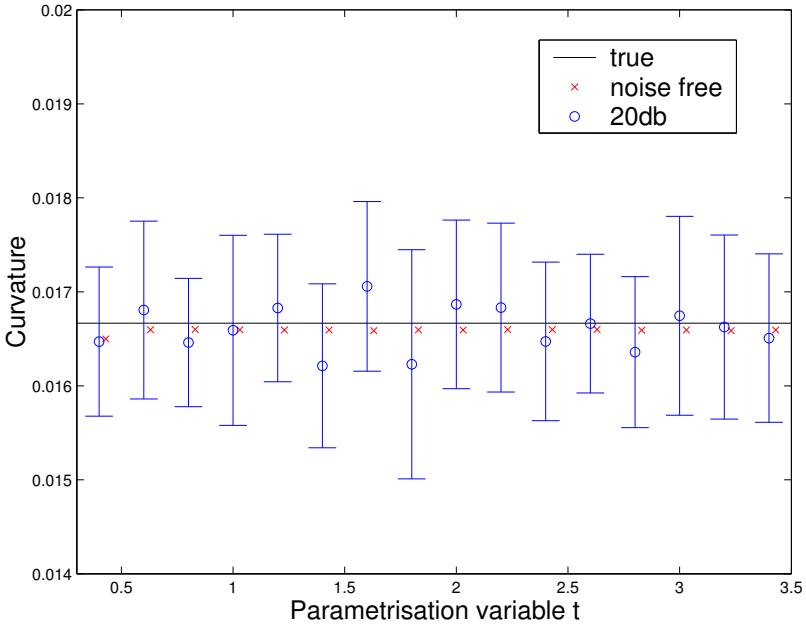


Figure 4.13: Curvature along a circular helix with $r = 30, h = 30$ averaged over 20 runs with $\sigma_g = 1$ and $\sigma_T = 5$.

4.2.5 Curvature along a curve

In this section we will study the estimation of curvature along a space curve. The circular and elliptic helix will be studied in the noise free case and hampered by noise of respectively 20,10 and 3 dB. We keep $\sigma_g = 1$ and $\sigma_T = 5$ and average the results over 20 noisy realizations. In figure 4.13 the curvature along a circular helix $(r\cos t, r\sin t, ht)$ with $r = 30, h = 30$ is shown. The curvature is constant along the curve. The x-axis is the running parameter t, where $t \in [0, 2\pi]$ for a full turn of the helix. As $ds/dt = const.$ the sample points are equally spaced along the curve. We retrieve the sub-pixel values by cubic interpolation of the curvature image. For the noise free case we observe an excellent estimation with a relative error of about 0.5%. For a noise level of 20dB the mean stays relative close to the true value and only the spread becomes larger. The elliptic helix given by the parametrization $(a\cos t, b\sin t, ht)$ has a non-constant curvature along the curve, compare eq.(B.31). In figure 4.14 we show the estimated curvature value and the relative error along an elliptic helix $(a = 20, b = 30, h = 40)$ in the noise free case and for 20 dB Gaussian noise. The relative error is less than 3% in the noise free case and for 20 dB the mean remains stable and again only the standard deviation increases. The sampled points are no longer equally spaced along the curve as $ds/dt \neq const.$ for an elliptic helix. For very low signal to noise ratios of 10 and 3 dB our algorithm still produces reasonable results as seen in figure 4.15. For 10 dB the estimation is still good as the average estimation is very close to the true value. For 3 dB however even the mean is not longer very well estimated. The error bars still intersect the true value

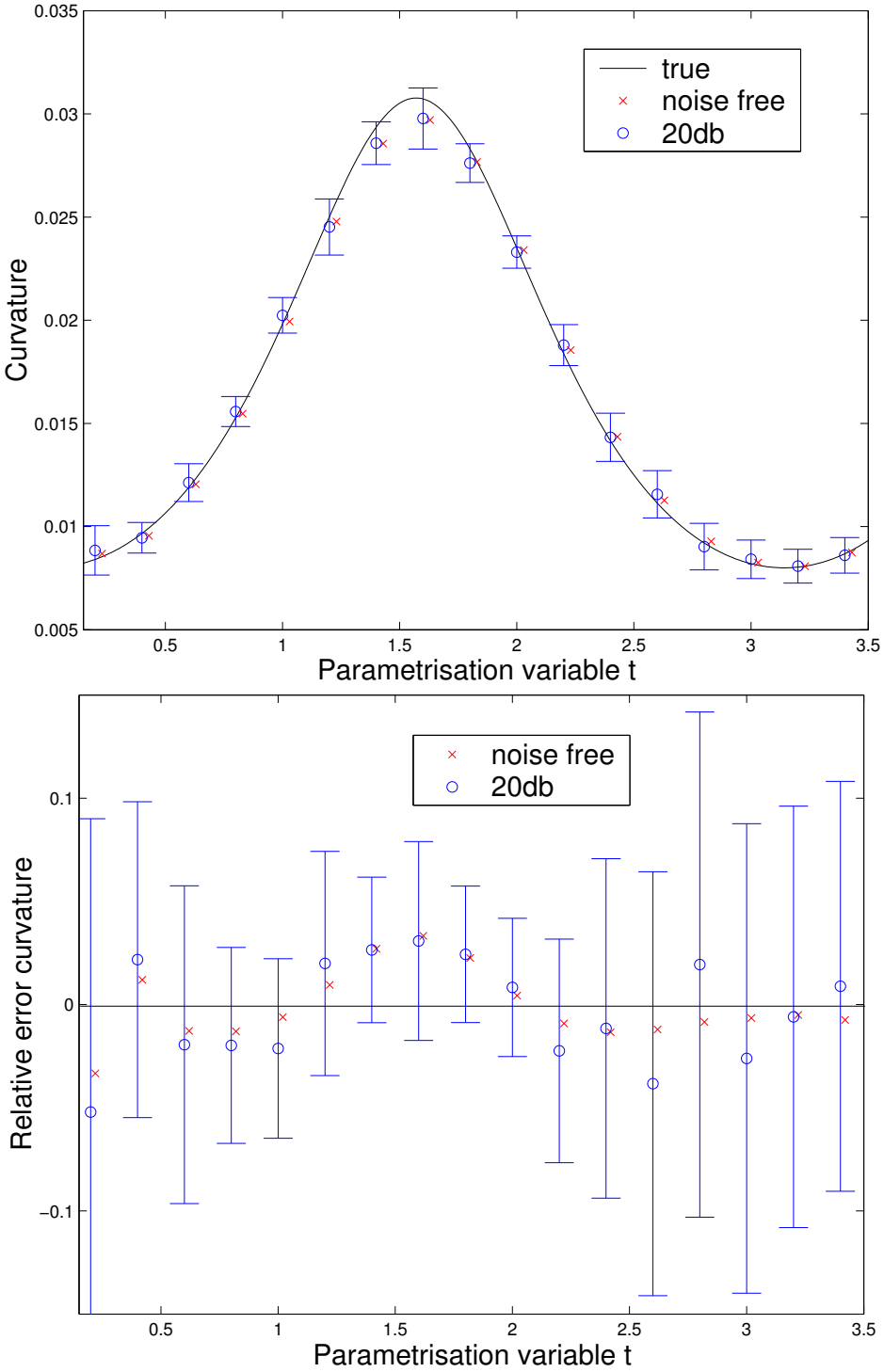


Figure 4.14: Curvature along an elliptic helix with $a = 20, b = 40, h = 30$ averaged over 20 runs with $\sigma_g = 1$ and $\sigma_T = 5$ for the noise free case and 20dB Gaussian noise.

most of the time.

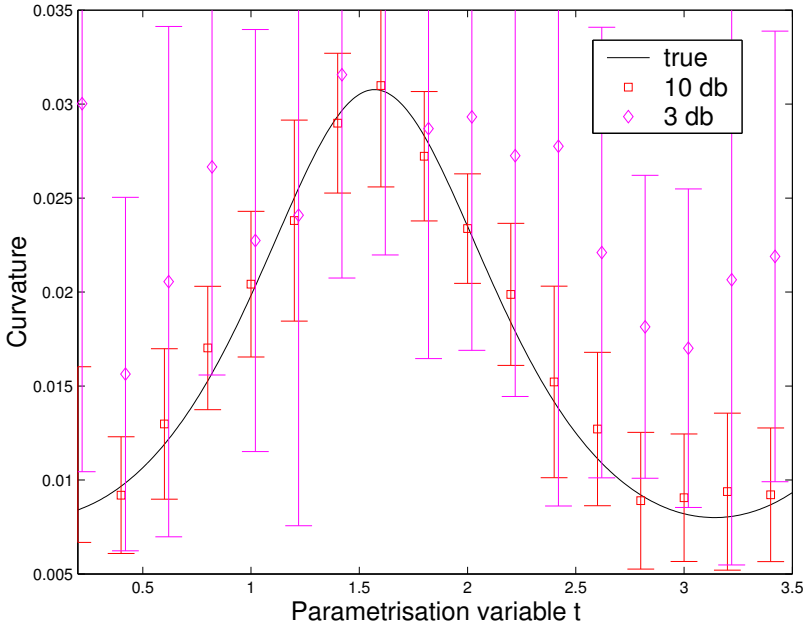


Figure 4.15: Curvature estimation along an elliptic helix with $a = 20, b = 40, h = 30$. averaged over 20 runs with $\sigma_g = 1$ and $\sigma_T = 5$ for 10dB and 3dB Gaussian noise.

4.3 Path length and bending energy

The bending energy of a thin rod, which is equivalent to a space curve, is defined as

$$BE = \int \kappa^2 ds \,, \tag{4.17}$$

if we neglect the material constants as the elastic (Young) modulus [118]. A closed line which forms a circle minimizes the bending energy as it has the minimal edge length for a given enclosed area and therefore the minimal average curvature. Bending energy can be seen as the roughness of a surface and can be used to characterize the shape of objects [30, 148].

Furthermore, we need the possibility to integrate the curvature along the line. Van Vliet proposed an sampling-error free method to measure 3D length for cylinders, see [132] chapter 8.2 and compare the summary in chapter 3.1. The first observation is that the grey-volume of an object is a sampling-error free measurement [122]. The idea is that an applied operator is dominant over the edge form and that the result has to be constant in one direction, then the remaining degree of freedom is proportional to the energy/volume of the response. This volume can be summed sampling-error free as it is a grey-volume, so we have to get a relation as grey-volume = *const.* $\times$ edge length. Van Vliet proposed the Second Derivative

in Gradient Direction (SDGD or ∂_{gg}) as an appropriated operator and $\sum \partial_{gg}$ as a measure of the length. Assume we have a cylinder with radius R, edge blurring σ and constant height of one in the middle, then the edge profile in cylindrical coordinates is

$$B(r,\varphi) = -\frac{1}{\sqrt{2\pi}\sigma}\int_r^\infty d\xi\, e^{-\left(\frac{\xi-R}{\sqrt{2}\sigma}\right)^2}. \tag{4.18}$$

If we neglect the smoothing effect of ∂_{gg} on the image, then the second derivative in radial direction is ∂_{rr} and the integral over the radial component is

$$\int_0^\infty dr\, r\frac{r-R}{\sqrt{2\pi}\sigma^3}e^{-\left(\frac{r-R}{\sqrt{2}\sigma}\right)^2} = \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{R}{\sqrt{2}\sigma}\right)\right) \tag{4.19}$$

$$\approx 1 \text{ for } R > 2\sqrt{2}\sigma\,, \tag{4.20}$$

where erf is the error function $\operatorname{erf}(x) := \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$ [17]. The sum of the ∂_{gg} is in good approximation independent of the radius R of the cylinder and the edge-slope parameter σ. However, our test images as in figure 4.3 do not fulfill the condition $R > 2\sqrt{2}\sigma$. The test-object represents a cylinder with radius $R = 0$. The physical equivalent of this test object, if acquired by an imaging system, would a point source. The observed signal width would be entirely due to the point spread function of the imaging system. In most real life situation there are no point sources and therefore the requirement $R > 2\sqrt{2}\sigma$ is often met.

We investigate two implementations of ∂_{gg}. First the classical expression [132] p.230

$$\partial_{gg} = \frac{g^t\mathbf{H}g}{\|g\|^2} = \frac{I_{xx}I_x^2 + 2I_{xy}I_xI_y + I_{yy}I_y^2}{I_x^2+I_y^2}\,, \tag{4.21}$$

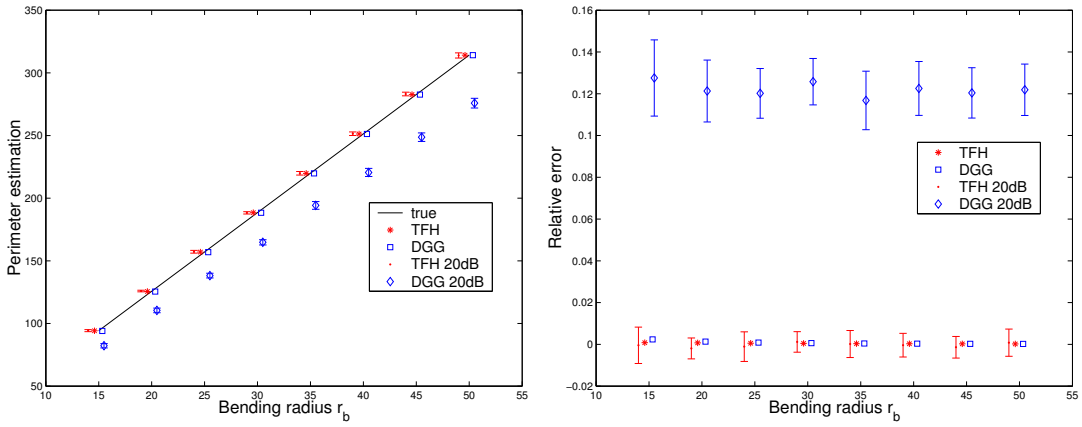
given for 2D, but this can easily be extended to nD. This method will be referred to as DGG. Second, a regularized formulation

$$\partial_{gg} = G^t\mathbf{H}G\,, \tag{4.22}$$

where $\mathbf{H} = I_{ij}, i,j \in 1,\dots n$ is the Hessian matrix , $g = (I_x, I_y)$ the gradient vector and G the gradient orientation computed with the gradient structure tensor (GST), see appendix A. The advantages of the regularization effect of the GST is that eq.(4.22) holds also on ridges and valleys where the gradient magnitude vanishes, we will refer to this method as the Tensor Framed Hessian (TFH). Furthermore, the disentanglement of the second derivatives $(\mathbf{H}, \sigma_g)$ and the orientation estimation step (G, σ_T) provides better noise suppression.

In figure 4.16 we investigate the behavior of the sum of the SDGD for bent cylinders ($R = 5, \sigma = 1$) with different bending radii $r_b \in [15,\dots 50]$ in the noise

free case and hampered with 20dB Gaussian noise. The experimental results show that eq.(4.19) also holds for bent structures. The TFH is outperforming the DGG in the presence of noise considerably by an order of magnitude. The relative error of the TFH is less than 1% in the noise free case and with 20dB added Gaussian noise.



(a) **(b)**

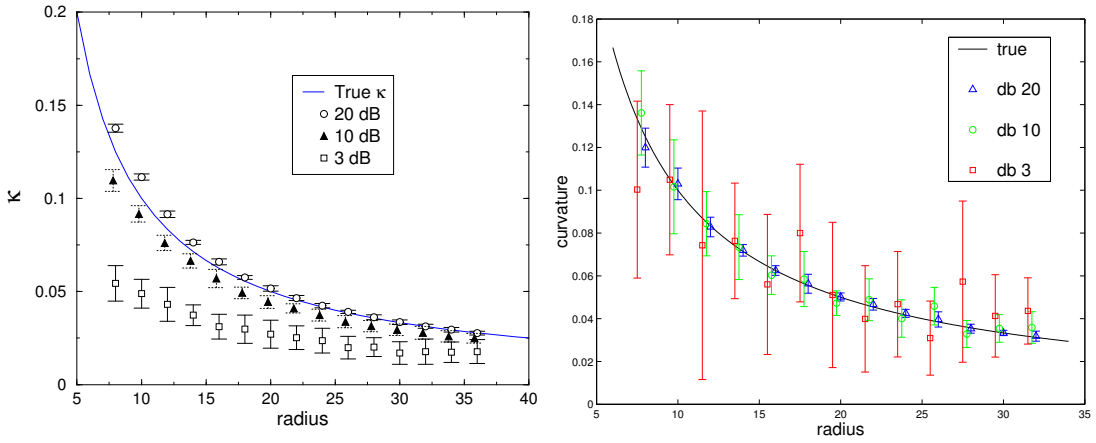
Figure 4.16: a) Perimeter/length estimation of a bend cylinder ($R = 5, \sigma = 1$). TFH: $\sigma_g = 1, \sigma_T = 3$, DGG $\sigma = 1$. The results are slightly shifted for better display, b) Relative error.

4.4 Comparison with the GST in parabolic coordinates

Bakker et al. presented in [5] a method to determine curvature of lines and line patterns or bundles in 3D. It can be also applied to 3D surfaces [4]. It is an extended method that was introduced by Verbeek et al. for the 2D case [140, 126]. In this method a local coordinate transformation is applied to the image such that the GST becomes translation invariant. Then the GST is calculated in this transformed coordinate system. The transformation has curvature as a free parameter that will adopt the curvature of the underlying pattern by minimizing the gradient energy along the "straightened" pattern. This yields an expression for the eigenvalues of the GST with one (in 2D) and two (in 3D) free parameters. Form the optimal transformation parameters an estimate for the local curvature can be obtained.

Three different types of artificial test images have been investigated in [5] and [4] p.61-79. A torus, a helix and an 2D ellipse have been studied. In section 4.2.3 we already discussed the estimation behavior of our new estimator on torii for

different radii and noise levels as shown in figure 4.7. Compared to the results of the method by Bakker [4] p.69 figure 4.5, we see that our method performs with a smaller bias but larger statistical error for noise levels of 20 and 10 dB, compare figure 4.17. With increasing noise a bias term is introduced in the parametric



(a) **(b)**

Figure 4.17: Curvature estimation on different torii for different noise levels. a) Reprint of Bakker [4] p.69 figure 4.5, b) here presented algorithm figure 4.7.

GST whereas our method remains unbiased as the mean value is relatively stable and the error-bars always intersect the true curvature value. The parametric GST gives large underestimations, this structural problem of the parameter estimation is explained in [4] p.79, but no solution to resolve this underestimation is given. In their estimation of the curvature, terms influenced by the noise do not cancel as they appear quadratic in the denominator and only linear in the numerator. In our approach to obtain the orientation via the GST we use a regularization method that averages out noise terms. In addition, noisy orientations are canceled in the curvature estimation as it is a difference of orientations. The same problem persists with the two other test objects. Our algorithm does not show this behavior as seen in figure 4.10. We did not use the flat ellipse as a test object, but however an elliptic helix, which also has a changing curvature along the curve. Again we observe even for high noise corruption a relative stable mean value as shown in figure 4.15.

We conclude that our algorithm performs at least as good as the approach of the parametric GST in the noise free case. For low signal-to-noise ratio our algorithm remains unbiased. The statistical error of our method is, however, larger than of the parametric GST approach.

4.5 Comparison with a segmentation based approach

Coeurjolly et al. [22] presented a purely discrete algorithm to compute curvature in images based on discrete osculating circles as an extension to the method by Worring and Smeulders [147].

The estimation error of Coeurjolly et al. is dependent on the resolution of the grid. Our algorithm is independent of this quantity as long as the object is not undersampled. Their estimation error on a 2D noise free disk is about 2.5% for a circle with radius 50 and about 1 % for a circle with radius 100 pixels (corresponding to grid step size of 1/50 and 1/100 resp.). On noise free 2D rings our method performs with a relative error of about 0.5% for radii greater than 25 pixels, see figure 4.18. For smaller radii the estimation is less reliable due to the influence of opposing ring edges in the smoothing of the gradient structure tensor elements for $\sigma_T = 5$, but still more accurate than the discrete approach. In the case of noisy

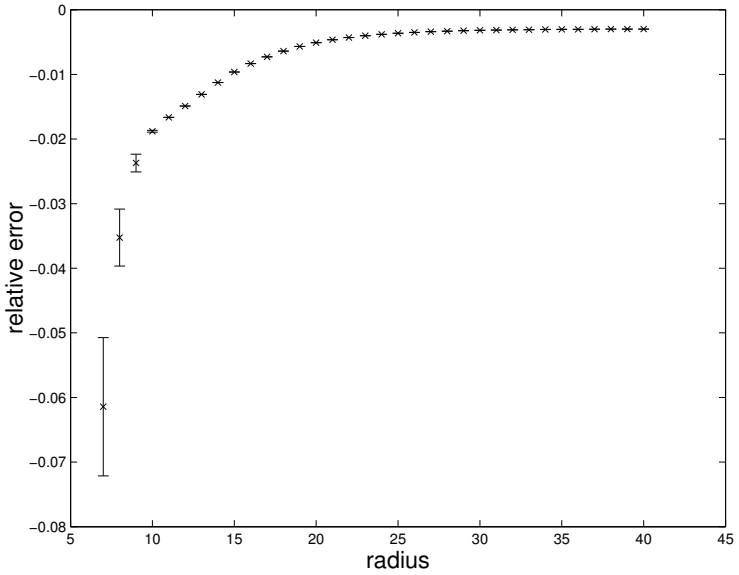


Figure 4.18: Relative error of the curvature estimation of a 2D ring as a function of the radius.

images the segmentation algorithm plays a key role for the quality of the curvature estimation, this is not the case for our proposed method.

Measuring the curvature along the boundary of a circle (40 pixel radius) Coeurjolly et al. get a relative error of about ±15% due to quantization. Our method gives only a relative error of ±1.2% if the values are retrieved from the curvature image at integer positions. However, we are allowed to interpolate the curvature image and retrieve curvature estimations with sub-pixel precision at the true center position of the ring. Then the relative error is $< 10^{-2}$%, compare figure 4.19.

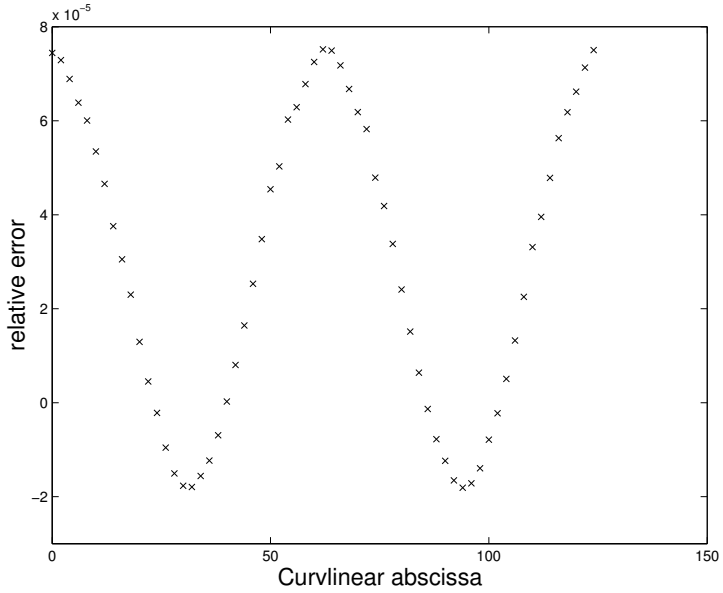


Figure 4.19: Relative error of the curvature estimation along half a 2D ring contour with r=40.

4.6 Application

In fluorescent confocal microscopy, especially in biological life time applications, 3D time series are acquired [8]. Typically these images contain moving bright spots. Here it could be of interest to compute the acceleration of the spots. The acceleration is related to the curvature of the spatio-temporal space curve formed by the moving spots. The acceleration a can be computed from the 4D image as follows (appendix B and [39])

$$\left(\frac{ds}{dt}\right)^2 = \|\dot{k}\|^2 = |v|^2 \tag{4.23}$$

$$a = |\dot{v}|T + |v|^2\kappa(s)N\,. \tag{4.24}$$

This is the classical expression of the tangential and the normal components of the acceleration.

The velocity v can be computed by e.g. optical flow or via the gradient structure tensor. See chapter 5 for a comprehensive introduction to gradient based velocity estimation. In a spatio-temporal 4D image (x, y, z, t) the orientation of the space curve, formed by a moving spot, is a measure for the velocity of the spot [59, 55]. It should be noted that the eigenvalue analysis of the gradient structure tensor which is used to measure the 4D orientation does not give direction information. We can, however, retrieve the velocity vector if we shift the discontinuity of the orientation field to the time dimension. This is reasonable as we know there is a causal connection between the time frames, i.e. $\Delta u_t > 0$. We can retrieve the

components of the velocity vector as follows:

$$v_i = \frac{\partial u_i}{\partial t} \approx \frac{\Delta u_i}{\Delta u_t} \; 1 \leq i \leq 3\,, \tag{4.25}$$

where u_i are the eigenvectors of the gradient structure tensor.

See also section 9 for an application to estimate curvature of bright spots in 4D (x, y, z, t) biological data.

4.7 Conclusions

We have shown that the curvature of space curves embedded in nD grey-value images can be estimated using the formulas given by differential geometry adjusted to the higher dimensional space mapped by the Knutsson mapping. Our new estimation formula eq.(4.7) reduces in 2D to the known expression, which clearly indicates that our work is consistent with older work. Furthermore, the estimation is unbiased, which even holds in the presence of noise. The approach holds for constant and variable curvature values along lines. The curvature calculation is independent of the scale of the objects as shown by our computations. Our grey-value based approach is clearly superior to a discrete curvature estimation and to the parabolic GST approaches as the signal to noise ratio becomes lower.

Chapter 5

Curvature and local shape descriptors of surfaces[5]

Curvatures of surfaces are the key to compute shape descriptors and to classify different classes of surfaces. We have developed a method to estimate principal curvatures of surfaces that are implicitly represented by grey-level isophotes (level-sets). Surfaces are embedded in the image by a grey-level difference with respect to their surroundings. Our method works directly on the grey-value information of the image, neither a segmentation is needed to detect the surface nor a parametric fit is done at any time during the analysis. The method exploits the differential structure of images.

The isophote curvature can successfully be applied to edges in 3D grey-value images [122, 47, 16, 120, 135, 80], but it fails when applied to planar structures and shells (hollow objects) [140]. First we give an illustrative example of this fact. In figure 5.1 we calculate the isophote curvature by the Hessian approach, an estimation via direct first and second derivatives as presented by Thirion and Gourdon [120] and by our new algorithm. The artificial test image consists of three ellipsoid shells, where the larger contains the smaller. Imagine it like the Russian doll, babushka, which contains a smaller versions of itself if you open it. Classically, the Hessian matrix $\mathbf{H}$ is rotated to be aligned with the normal of the surface

$$\mathbf{H}' = \mathbf{R}\mathbf{H}\mathbf{R}^T = \left(\begin{array}{c|c} I_{norm} & \\ \hline & \mathbf{H}'_{tang} \end{array}\right) . \tag{5.1}$$

Then the surface subspace Hessian $\mathbf{H}'_{tang}$ is diagonalized and the principal curva-

[5]Different parts of this chapter have been published in B. Rieger, F.J. Timmermans, L.J. van Vliet and P.W. Verbeek, Curvature estimation of surfaces in 3D grey-value images, *ICPR'02, Proc. 16th Int. Conf. on Pattern Recognition*, August 11-15 2002 [101] and B. Rieger, L.J. van Vliet and P.W. Verbeek, Estimation of curvature based shape properties of surfaces in 3D grey-value images, *SCIA'03, Proc. of the 13th Scandinavian Conf. on Image Analysis (Göteborg, Sweden)*, June 29 -July 2 2003 [106] . The main part has been accepted for publication in *IEEE Transactions on Pattern Analysis and Machine Intelligence* [102].

tures are computed from the eigenvalues [2, 132] chapter 5.4

$$\kappa_{1,2} = \frac{-\lambda_{1,2}}{\|\nabla I\|} \,. \tag{5.2}$$

Furthermore, for patterns of ridges, a robust estimation is not possible as noise reduction involves larger smoothing, but this also smooths out the pattern. The problems associated with isophote curvature can be overcome if we transform the grey-value image into an orientation map (normal vector field up to sign) from which the curvatures can be derived after solving the discontinuity problem. In 2D the use of the double angle method is well-known [43, 63], but in 3D this problem remained an obstacle that prevented the computation of curvature. We solve this problem by mapping the orientation field to a closed representation via the Knutsson mapping eq.(2.4). This representation is suitable for further processing and enables us to compute the principal curvatures.

In the early 1990s the first segmentation free estimation algorithms for 3D iso-surface curvature computations in grey-level images were presented. Bårman presented the use of quadrature filters on a vector image to estimate orientation and curvature [2]. Monga et al. [85] made use of a 4D description for 3D surfaces by Monge patches [116]. Van Vliet presented isophote curvature computation by a rotated Hessian [135]. Later other authors presented different methods which are purely based on first and second derivatives of the image [47, 120, 54, 110]. These methods have in common that only one scale is involved in the computation of the derivatives for the orientation, the change of the orientation and the noise suppression. Furthermore, if normalization by the gradient magnitude is required, this implies automatic failure on ridges and valleys.

First we briefly introduce the mathematics needed to compute curvatures on 2D surfaces. For a comprehensive discussion on the differential calculus on surfaces see for example [39, 28, 116, 84]. In appendix C the mathematics of curvatures of surfaces are summarized. In addition, higher dimensional surfaces are discussed. The curvature κ at a point p in a tangent direction T on the surface is defined as the magnitude of the change of the surface normal N in a tangent direction T

$$\kappa_T(p) = \|\nabla_T N\| \,. \tag{5.3}$$

There exist two mutual orthogonal tangent direction T_1, T_2, for which the curvature is extremal. They are called principal directions, which associated curvatures κ_1, κ_2. Two classical measures of curvature in a point are

$$\text{Gaussian curvature} \quad K := \kappa_1 \kappa_2 \tag{5.4}$$

$$\text{mean curvature} \quad H := \frac{\kappa_1 + \kappa_2}{2} \,. \tag{5.5}$$

From Gauss Theorema Egregium we know that the Gaussian curvature is an isometric invariant [28], i.e. if a surface is bent without stretching K remains the

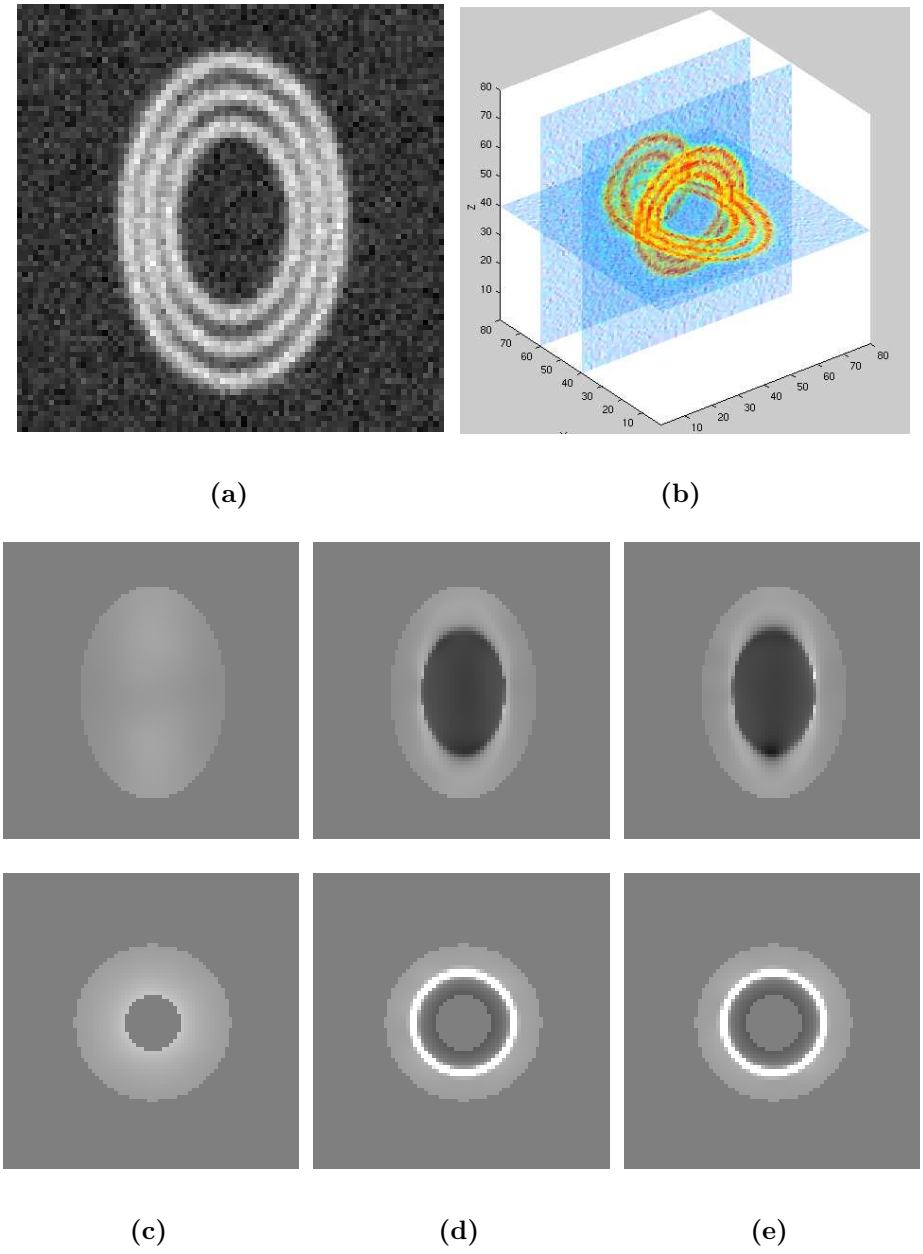


Figure 5.1: a,b) Stack of ellipsoid surfaces (like the Russian nesting doll Babushka) with 20dB Gaussian noise in an image with $80 \times 80 \times 80$ voxels, c) Our new curvature estimation as in eq.(5.12), d) Isophote curvature via eq.(5.2), e) Curvature estimation as in [120] eq.(12-14), display range [-0.2, 0.2], $\sigma_g = 1, \sigma_T = 4$. All cross sections of the estimation are not in the symmetry planes of the test image but are shown 10 pixels off.

H \ K	> 0	0	< 0
> 0	Peak	Ridge	Saddle ridge
0	-	Flat	Minimal surface
< 0	Pit	Valley	Saddle valley

κ_1 \ κ_2	> 0	0	< 0
> 0	Peak	Ridge	Saddle
0	Ridge	Flat	Valley
< 0	Saddle	Valley	Pit

Table 5.1: Principal curvatures and their connection to qualitative surface properties.

same, although the principal curvatures and the mean curvature change. In table 5.1 the shape of a surface is connected to the curvatures.

A shape descriptor that can relate the shape of a 2D surface to its principal curvatures by only one number is the shape index introduced by Koenderink [68]

$$s = \frac{2}{\pi} \arctan \frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1}, \ \kappa_2 \geq \kappa_1 \,, \tag{5.6}$$

if $\kappa_1, \kappa_2 \neq 0$.

In order to build (local) shape descriptors for objects based on these curvatures we must be able to integrate the curvatures over the whole object surface. To do this we introduce a sampling-error free surface area estimator based on the sum of the samples in a local image. The local image is transformed into an image whose sum is proportional to the surface of the embedded object. With this technique at hand we can now estimate the Euler characteristic and bending energy. In images that contain texture in the form of planar structures, the local bending energy, $\kappa_1^2 + \kappa_2^2$ [135], is a useful characteristic (local deformation energy) and can be computed per point with sub-pixel precision. It can be integrated over the object surface

$$E_B = \int \kappa_1^2 + \kappa_2^2 \, dA, \tag{5.7}$$

into a scale invariant shape descriptor. A sphere minimizes bending energy for a given closed surface. Therefore it can be seen as the roughness of a surface and can be used to characterize biological objects [30, 148]. The Euler characteristic is (by the Poincaré-Hopf index theorem or the Gauss-Bonnet theorem)

$$\chi = \frac{1}{2\pi} \int \kappa_1 \kappa_2 \, dA \,. \tag{5.8}$$

describes the global topology of a closed surface by an integer number. It describes the topology of the surface, and is in general the alternating sum of Betti numbers, which reads in 3D for polygons $\chi = n_{\text{vertices}} - n_{\text{edges}} + n_{\text{faces}}$. The Euler characteristic χ as a function of scale is closely related to the morphological granulometry [112, 13, 78]. The latter is a volume weighted distribution, whereas χ counts the number of objects minus the number of handles (or tunnels) visible at a certain scale.

5.1 Curvature of surfaces in grey-value images

In order compute the principal curvatures and shape descriptors of iso-surfaces embedded in grey-level images we need 1) To find the vector field normal N and the principal directions T_1 and T_2, 2) solve the discontinuity problem of N and compute $\|\nabla_T N\|$ and 3) be able to integrate over weighted iso-surfaces to form shape descriptors.

5.1.1 The gradient structure tensor and the principal directions

The gradient structure tensor (GST) is a tool to analyze local structure in images [10, 60, 151]. It is defined as

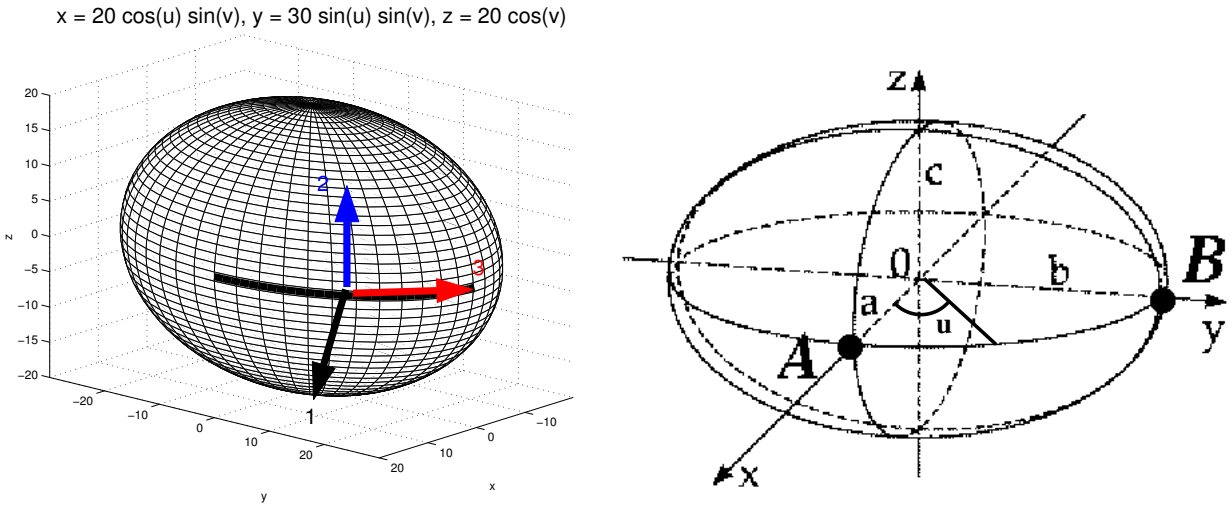
$$G := \nabla I \nabla I^t,\ \overline{G} := \overline{\nabla I \nabla I^t}\,, \tag{5.9}$$

where I is a grey-value image and the overhead bar $\overline{(\cdot)}$ stands for averaging the elements over a local neighborhood σ_T. The gradient structure tensor borrows its descriptive power from the analogy to a well-known quantity in physics, the inertia tensor. See also appendix A for a review of the properties of the GST.

Two scales are involved. The gradient vector ∇I is computed by convolutions with Gaussian derivatives at the scale σ_g. The size of the Gaussian weighted tensor smoothing σ_T defines the neighborhood in which the image structure is computed. The latter can be used for noise suppression without hampering the signal strength. An eigenvalue analysis of $\overline{G}$ is a standard approach to classify local structure in images. In the case of planar structures the eigenvalues will have the following ordering: $\lambda_1 \gg \lambda_2 > \lambda_3$. The eigenvector v_1 (corresponding to the largest eigenvalue λ_1) is aligned with the surface normal. The estimated orientation v_1 for symmetric neighborhoods is unbiased, as gradient contributions from one side of the symmetry axis (or symmetry plane) are balanced by their mirrored counterpart. For asymmetric neighborhoods a small bias is introduced. The remaining two eigenvectors v_2, v_3, lie in the principal directions of the surface. Summarizing we find, compare figure 5.2a) :

$$v_1 \leftrightarrow N,\ v_{2,3} \leftrightarrow T_{1,2}\,. \tag{5.10}$$

A drawback of the GST is that the set $\{v_i\}$ only contains orientation information. Therefore we have to deal with a discontinuous representation as orientation is direction up to point inversion.



(a) **(b)**

Figure 5.2: a) The principal directions on a ellipsoid $v_1 \leftrightarrow N, v_2 \leftrightarrow T_1, v_3 \leftrightarrow T_2$. The line is the parametrization region $v = \pi/2, u \in [0, \pi/2]$. κ_2 is the curvature along the line and κ_1 perpendicular to it (in the direction of v_2. b) A sketch of an ellipsoid with half axes a, b, c.

5.1.2 Estimation of the principal curvatures

The principal directions T_1, T_2 and the surface normal N can be found using the GST as described above. Now we want to compute the principal curvatures differentiating the normal with respect to the principal directions, from eq.(5.3) and eq.(5.10) we obtain

$$|\kappa_{1,2}| = \|\nabla_{T_{1,2}} N\| \leftrightarrow \|\nabla_{v_{2,3}} v_1\| \,. \tag{5.11}$$

Unfortunately the orientation field v_1 contains a discontinuity. Computation of partial derivatives of orientation data is not trivial as explained in chapter 2 and 4. The idea is to transform v_1 into a continuous representation by a mapping M for which $\|\delta M(v)\| = K\|\delta v\|$ holds. A suitable mapping was introduced by Knutsson $M(v) = \frac{vv^T}{\|v\|}$, with $K = \sqrt{2}$ for $\|v\| = const.$ [63]. The norm of the mapped derivative $M(v)$ is linearly related to the norm of the derivative of $\mathbf{v}$. Therefore we propose a new expression, together with (5.10), that is suitable to compute curvature of surfaces in images

$$|\kappa_{1,2}| = \frac{1}{\sqrt{2}} \|\nabla_{v_{2,3}} M(v_1)\| \,. \tag{5.12}$$

For the computation we write down the formula in index notation. Let the elements of $M(v_1) = \frac{v_1 v_1^t}{\|v_1\|}$ be M_{ij} and x^i a coordinate with $1 \leq i, j \leq 3$. Then we have

$$|\kappa_{1,2}| = \frac{1}{\sqrt{2}} \| \sum_{k=1}^{3} \frac{\partial M_{ij}}{\partial x^k} v_{2,3}^k \| \,, \tag{5.13}$$

where the norm of M is defined as the Fröbius norm $\|M\|^2 := \sum M_{ij}^2$. The ordering of the elements M_{ij} does not influence the outcome as only the norm is evaluated. For a formal proof that eq.(5.12) equals eq.(5.3) see appendix E.

The application of eq.(5.12) to hyper-surfaces with codimension 1 (i.e. the surface is $n-1$D) in nD images is straightforward as all involved algorithms, i.e. GST and Knutsson mapping, are applicable to nD images. Then there are $n-1$ principal curvatures and directions which are again mutual orthogonal (if the curvatures are non-degenerated). See also appendix C for a discussion of high-dimensional surfaces.

The sign of the curvature, i.e. if we deal with a elliptic or hyperbolic situation is lost by the outlined algorithm. The sign can be retrieved even from planar structures by the scheme described in 5.1.4 .

5.1.3 Surface area estimation

In this section we sketch how to compute surface area from the grey-level volume of an object. We will focus on the main idea and leave out details of the edge localization by Gaussian filters, for a much more comprehensive discussion see [129, 132] and the summary in chapter 3.1.

Assume a band-limited solid object embedded in an image I by an iso-surface at level ℓ. The surface area of the object $\int_S dA$ can be computed by a volume integral $\int_I \Phi[I(x)]\, dV$ after a suitable transformation of the image. The transformation Φ must produce an image whose integrated volume = *const.* $\times$ surface area. Thus each cross section $e(r)$ perpendicular to the iso-surface S should contribute an equal amount. First we apply erf-clipping (a form of soft-clipping) to the image between the limits $\ell \pm \frac{1}{2}h$. The cross section is a clipped edge, $e(r) = h - \int_0^r p(\tilde{r})\, d\tilde{r}$ with profile distribution $p(\tilde{r})$ satisfying $\int_0^\infty p(\tilde{r})\, d\tilde{r} = h$. Along the iso-surface $\|\nabla I_{\text{clip}}\| = \|\frac{d}{dr} I_{\text{clip}}\|$. Note that the integral $\int \|\frac{d}{dr} I_{\text{clip}}\|\, dr = \int \|\frac{d}{dr} e(r)\|\, dr = h$. Finally, volume integration of $\|\nabla I_{\text{clip}}\|$ accumulates the contribution of all cross sections and hence $\frac{1}{h} \int \|\nabla I_{\text{clip}}\|\, dV$ yields the surface area of the embedded object. Thus integration over an iso-surface element dA is accomplished by integration of the gradient magnitude after clipping over the entire image volume

$$\int_S dA = \frac{1}{h} \int_I \|\nabla I_{\text{clip}}\|\, dV \,. \tag{3.24}$$

The volume integral can be replaced by a sum of the sampled image iff $\|\nabla I_{\text{clip}}\|$ is approximately bandlimited and sampled at half the Nyquist rate [122, 129]. Since $\|\nabla I\| \geq 0$ noise contributions will not compensate each other, but sum up over the entire image. A mask image can be used to only integrate over the region of interest. The mask can be computed from the gradient magnitude with a larger filter and then performing an iso-data threshold operation. Another possibility is to apply a non-linear smoothing operation (e.g. anisotropic diffusion along the contour $\frac{dI}{dt} = -I_{\mathbf{TT}}$ [139]) for preprocessing followed by an erf-clipping operation [129]. The sampling-error free measurement of the surface area is only possible if the object is solid.

5.1.4 Solid objects from shells and the sign of curvature

To reconstruct a solid object form a shell, the shell is first preliminary filled by the grey-weighted distance transform [109, 127]. Now we can distinguish interior from exterior parts of the object. This yields a sign which we can add to the estimated orientation field and obtain the normal vector field. Together with the original object shells in the input image we can create a gradient vector image from which we can obtain solid objects by an advanced integration technique [125]. The sign of the surface principal curvatures for shell like object in grey-value images is obtained by investigating the value of the second derivative along the tangent direction $T^t HT$, where H is the Hessian matrix of the image, so a maximum/minimum corresponds to a negative/positive value.

5.2 Evaluation of the algorithm

5.2.1 Test images

We created band-limited grey-value test images containing shells of constant thickness of spheres and ellipsoids, see figure 5.3. A band-limited ellipsoid shell is produced by computing the squared gradient magnitude of a band-limited filled ellipsoid. Isophote curvature estimation $\frac{I_{\mathbf{TT}}}{\|\nabla I\|}$ applied to these images will fail as the gradient vanishes on ridges and valleys. The relative edge localization error of the gradient in 3D is about $-\left(\frac{\sigma}{R}\right)^2$ for constant curvature [130], where R is the bending radius and σ the standard deviation of the derivative filter. Therefore the peak position of $\|\nabla I\|^2$ will be shifted, but this is negligable for $R > 10$. We must satisfy $\sigma \geq 0.9\sqrt{2} \approx 1.3$ to ensure band-limitation [130].

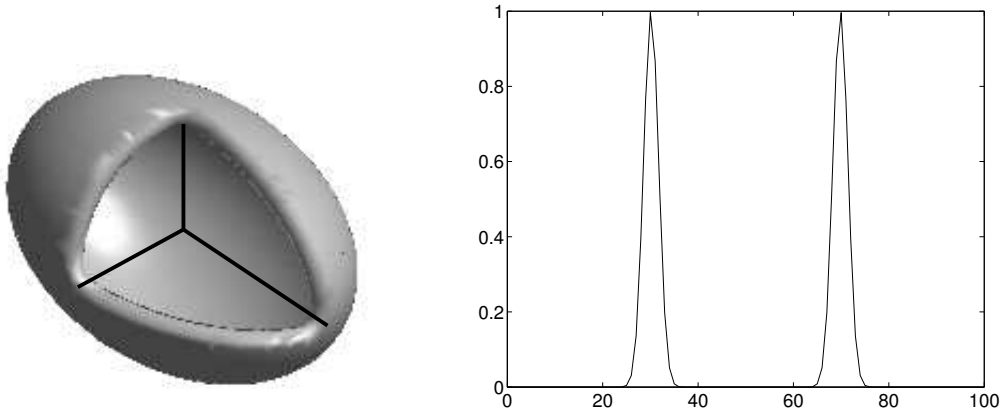


Figure 5.3: A test object; hollow ellipsoid with iso-intensity lines and profile plot.

5.2.2 Curvature

In this section we investigate the performance of our curvature estimator on different ellipsoids with different levels of added Gaussian noise. An ellipsoid with half axes a, b, c (figure 5.2b)) given by the equation

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1\,, \tag{5.14}$$

is parameterized by $(a\cos u \sin v, b \sin u \sin v, c \cos v), u \in [0, 2\pi], v \in [0, \pi]$. Due to symmetry it is sufficient to evaluate our algorithm in the xy-plane ($v = \pi/2$) in the first quadrant ($u \in [0, \pi/2]$), compare figure 5.2a) the black line. The principal curvatures can be computed analytically from the parametrization, see eq.(C.17) The first principal direction lies perpendicular to the xy-plane, whereas the second lies in it. The curvatures are computed for the whole image via eq.(5.12) and then evaluated at the mathematical surface position. We can retrieve the values with sub-pixel precision as our test image is sampled correctly. Curvatures will be denoted as 0.2(5). The first value is the actual curvature, the second between the brackets is the radius of the corresponding oscillating circle, i.e. $1/\kappa$.

In figure 5.4 the estimated and true principal curvatures are plotted for three different ellipsoids with half axes $a = c$ and $b/a = 1.5$. The estimation is averaged over 20 sub-pixels shifts. The relative estimation error is about 1% at the blunt side ($u = 0, v = \pi/2$) and increases with increasing curvature to about 7% for $\kappa = 0.075(13.\bar{3})$ at $u = \pi/2$. For curvatures smaller than 0.04(25) our estimation has very little bias ($< 2\%$). The small underestimation of the true curvature as observed in figure 5.6b) for higher curvatures can be understood as follows. The derivatives are implemented as convolutions with Gaussians derivatives. For reasons of band-limitation one should always choose $\sigma > 0.9$ [132]. Recall that the curvatures are computed as the derivative of the (mapped) surface normal in the direction of the principals (eq.(5.12)), then the average over a surface neighborhood includes

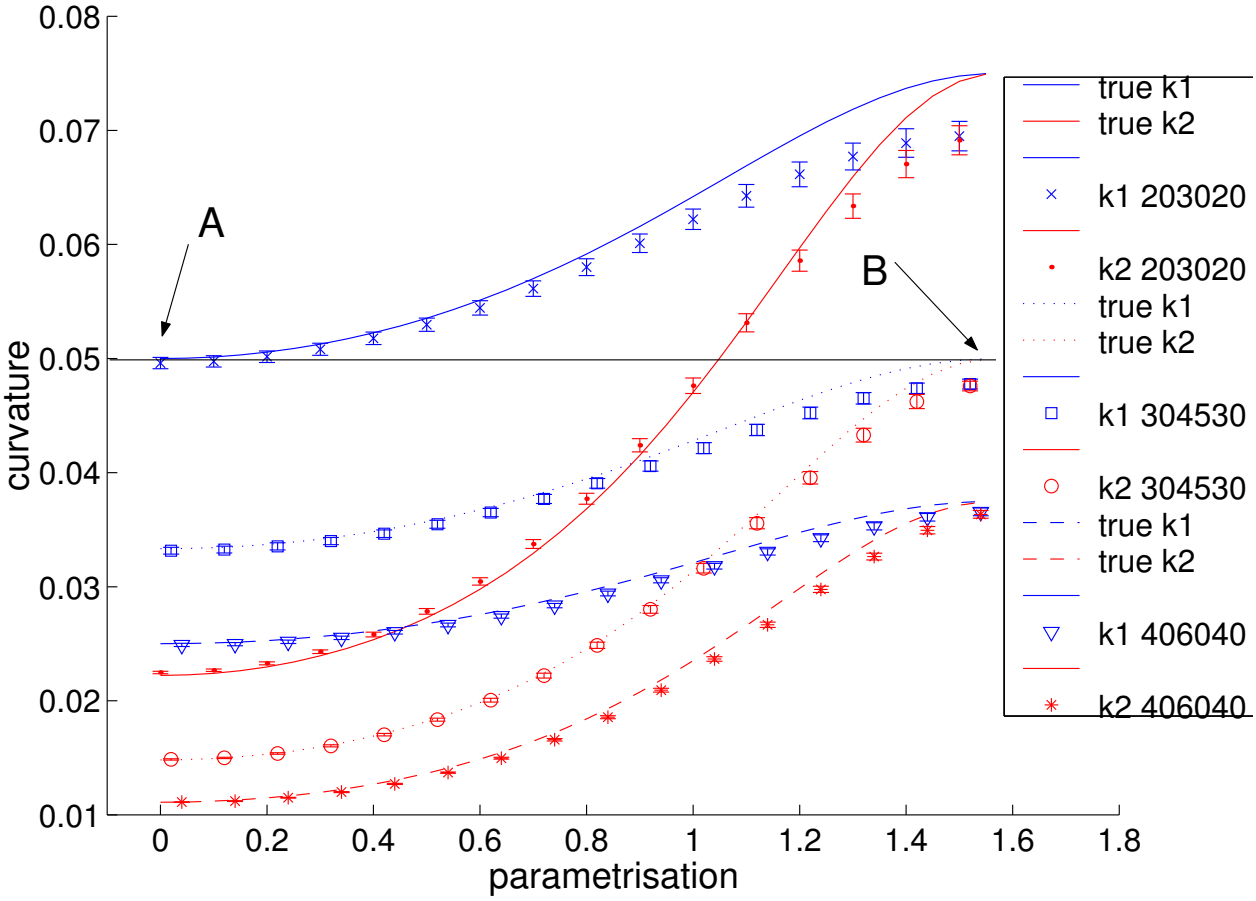


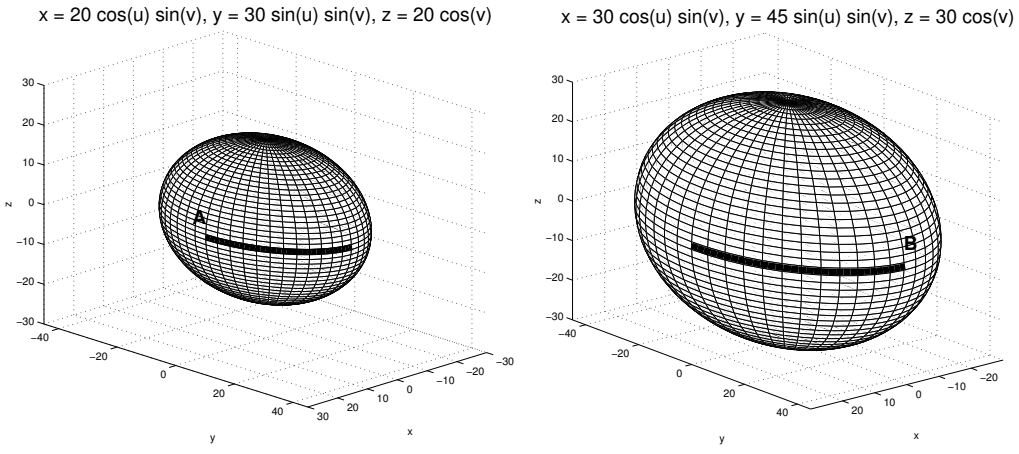
Figure 5.4: Curvature estimation on 3 different ellipsoids averaged over 20 subpixel shifts. Top $a = c = 20, b = 30$ middle $a = c = 30, b = 45$, bottom $a = c = 40, b = 60$. The points `A` and `B` lie on two different ellipsoids with the same curvature $\kappa_1 = 0.05(20)$. The estimation error, however, is different, see the text for a detailed explanation.

surfaces normals that are not perpendicular to the direction of the derivation, thus have a smaller component than the normal at the point under consideration. Therefore the more the local neighborhood is curved the larger the error. This reasoning is also valid in the Knutsson representation where the derivative is taken, as the Knutsson mapping preserves the local geometry (uniform stretch requirement).

In figure 5.5a) and b) two points (A,B) are indicated that lie on two different ellipsoids with the same curvature $\kappa_1 = 0.05(20)$. In A the estimation is excellent ($\epsilon < 1\%$), although the local geometry is asymmetric ($\kappa_1 \neq \kappa_2$), whereas at the umblic point B (curvatures in all directions are equal) we observe a larger error ($\epsilon \approx 4\%$). The reason for this observation is that at A $\kappa_1 = const.$ and κ_2 changes slowly, whereas at B $\kappa_1 = \kappa_2$ but both are changing rapidly. Hence the estimation error is not only dependent on the value of the curvature, it also depends on the change of the curvatures in the neighborhood, i.e. the local geometry.

Influence of the scale of the filters: $\sigma_g, \sigma_T, \sigma_k$

There are three Gaussian kernel sizes involved in the computation: first the derivative kernel σ_g to compute the gradient ∇I, second the smoothing kernel σ_T of the GST and finally the derivative kernel σ_k applied to the mapped normal field (eq.(5.12)). We do not investigate the dependency on the size of σ_g as the smallest reasonable size, $\sigma_g \approx 1$, gives the best gradient [43]. The influence of σ_k is shown in figure 5.6a). At symmetric neighborhoods ($\kappa \neq 0, \dot{\kappa} = 0$) the size of σ_T has no influence on the quality of the orientation estimation of the GST due to cancellation of equally strong gradients lying at opposite sides of the true gradient. The estima-



(a) **(b)**

Figure 5.5: a) Ellipsoid with $a = c = 20, b = 30$, b) ellipsoid with $a = c = 30, b = 45$, $\kappa_1(\mathtt{A}) = \kappa_1(\mathtt{B}) = \kappa_2(\mathtt{B}) \neq \kappa_2(\mathtt{A})$.

tion will be biased if the curvature is not constant ($\dot{\kappa} \neq 0$) in the kernel window. The estimated gradient orientation is overestimated in this case, as it is the mean gradient within the area given by σ_T. Secondly, as the filter size increases on top of the ellipsoid surfaces patches are averaged that do belong to different sides (of the yz-plane). Nevertheless the relative error in figure 5.6b) is much smaller for the same size of σ_T as for σ_k in figure 5.6a). This is due to the fact that curvature is a difference of orientations, biased orientation terms (dependent on σ_T and $\dot{\kappa}$) cancel to a limited degree. The actual choice of the size of σ_T for generic neighborhoods is a trade off between noise suppression and orientation estimation precision.

Performance in the presence of noise

We add different levels of Gaussian noise to the test images to measure the robustness of the estimator . We use the definition SNR$= 20 \log \frac{S}{\sigma_n}$, where σ_n^2 is the variance of the Gaussian noise and S the maximal signal strength. In figures 5.6c,d) the results are shown for two different ellipsoids. The error bars indicate the standard deviation over 40 runs. The estimation is consistent as the mean stays around the true value and the error-bars intersect the true curvature. With increasing noise level only the error-bars become larger, the mean remains stabile. For a discussion on noise robustness for traditional methods see for example [51].

5.2.3 Bending energy, Euler characteristic and surface area

In this section we test the performance of our estimator to compute (local) surface shape descriptors based on the principal curvatures. We investigate the behavior for a torus and for ellipsoids as a function of scale and deformation. In all experiments the results of the true, the noise free and two noise level (20, 40dB) are plotted. The noise runs are averaged over 20 runs and the noise free over 20 sub-pixel shifts. For all computation we keep the following kernel sizes $\sigma_g = \sigma_k = 1, \sigma_T = 2$ fixed.

The sign of the curvature is needed to compute the Euler characterization. If only closed shells are present in the image then by filling the objects (compare 5.1.4) and assuming that objects are white, it is clear what is inside and outside and a sign can be given to the curvature. The sign cannot be given by local information only.

Ellipsoids, scaled and deformed

The true surface area and bending energy are integrated numerically from the analytically computed curvatures eq.(C.17) with MATHEMATICA [99]. The Euler characteristic is for any closed surface without holes $\chi = 2$. The results of our

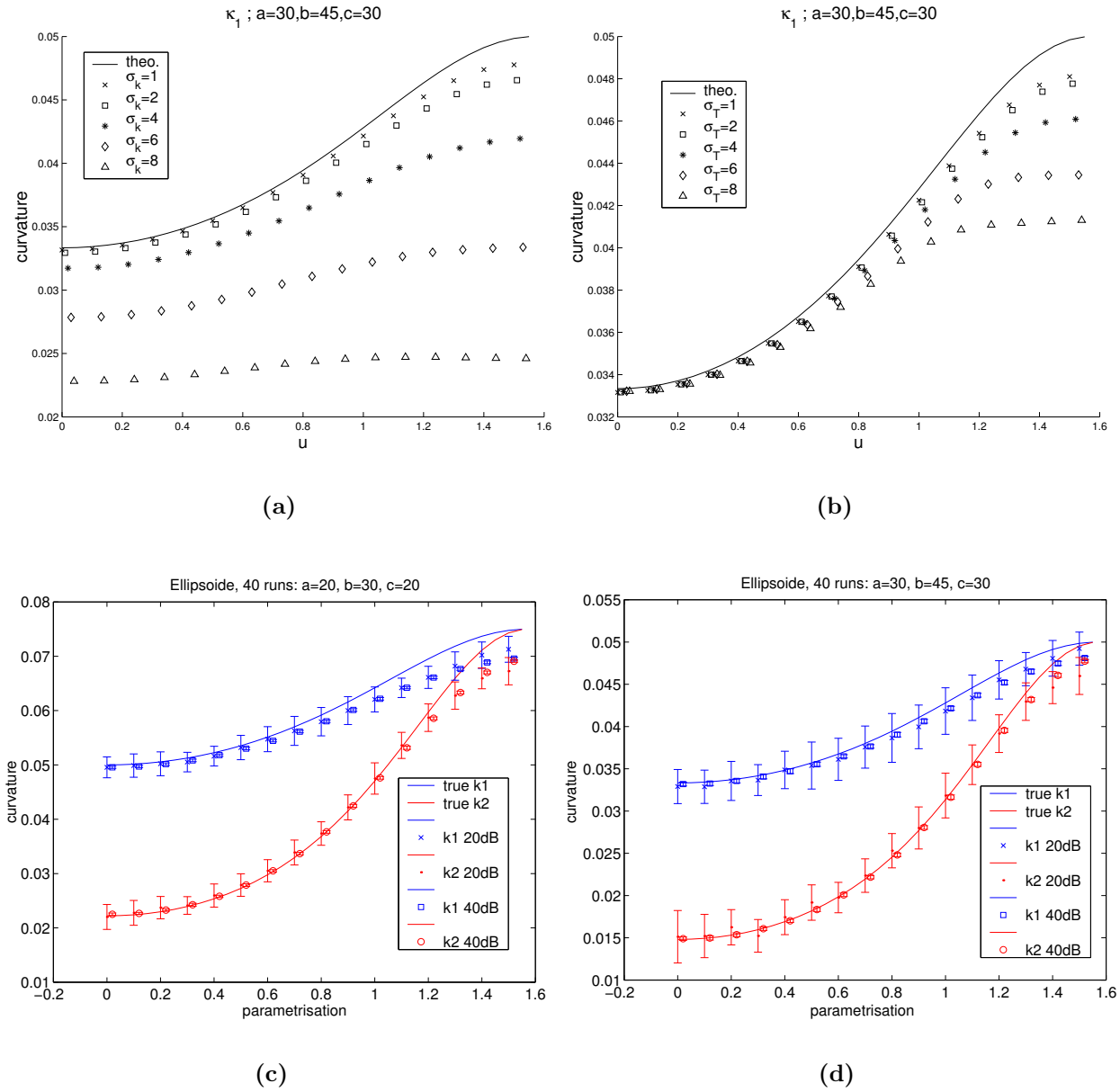
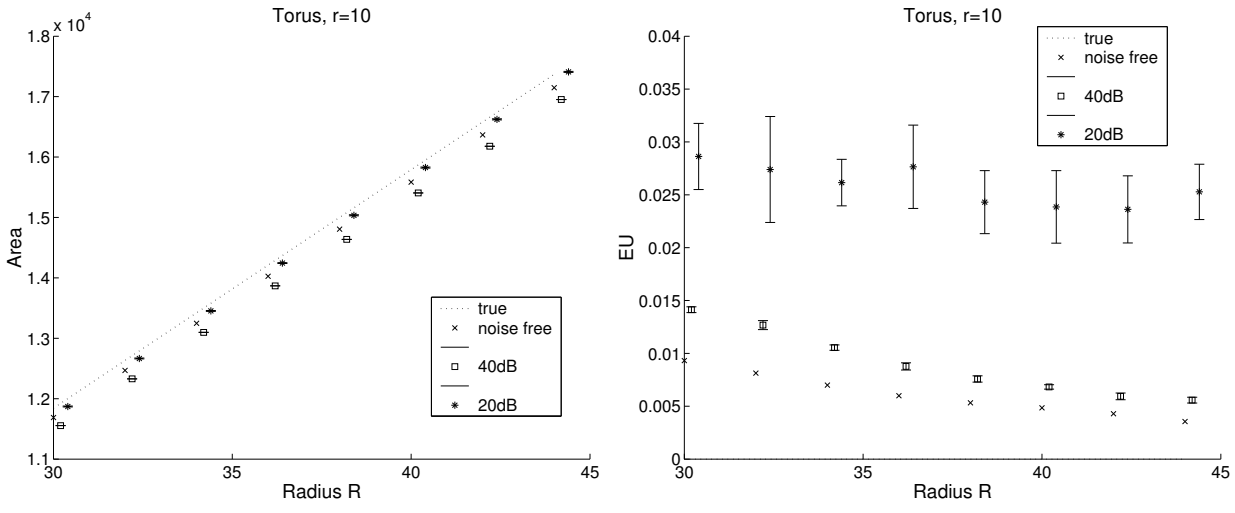


Figure 5.6: a) Influence of σ_k on κ_1, with $\sigma_g = 1, \sigma_T = 2$, b) Influence of σ_T on κ_1, with $\sigma_g = 1, \sigma_k = 1$, c,d) Principal curvatures $\kappa_{1,2}$ in xy-plane along the parametrization averaged over 40 runs for different noise levels c) $a = c = 20, b = 30$, d) $a = c = 30, b = 45$; $\sigma_g = 1, \sigma_T = 2, \sigma_k = 1$.

computations are shown in figure 5.8a) for a scaled ellipsoid and in figure 5.8b) for a deformed one. The deformation of an ellipsoid is done by slowly deforming a sphere ($r = 12$) into a elongated cigar like object ($a = c = 12, b = 38$); whereas the scaled ellipsoid is blown up from $a = c = 12, b = 18$ to $a = c = 36, b = 54$ with constant $b/a = 1.5$.

The area estimation is critical to all other estimations. For the noise free case the estimation is excellent, for small curvatures the estimation error is smaller than 0.5%. Even in the present of noise the area estimation is fine. The bending energy and the Euler characteristic are scale invariant properties. In figure 5.8a) we see that the run of the curve for the noise free and for 40dB indeed approach a constant value. For 20dB the curvature estimation for larger ellipsoids seems to be influenced by the noise in a scale variant manner, as the area estimation remains fine. As higher curvature are more biased (section 5.2.2) we expect the estimation to perform better for larger ellipsoids. The quality of the estimation of the bending energy and the Euler characterization on the other hand decrease with increasing ratio b/a (elongation). The curvature increases in some areas during deformation such that the bending energy increases. These larger values will have a larger negative bias as explained in 5.2.2.



(a) **(b)**

Figure 5.7: a) Surface area and b) Euler characterization estimation at different noise levels averaged over 20 runs for scaled torus, $\sigma_g = 1, \sigma_T = 2, \sigma_k = 1$.

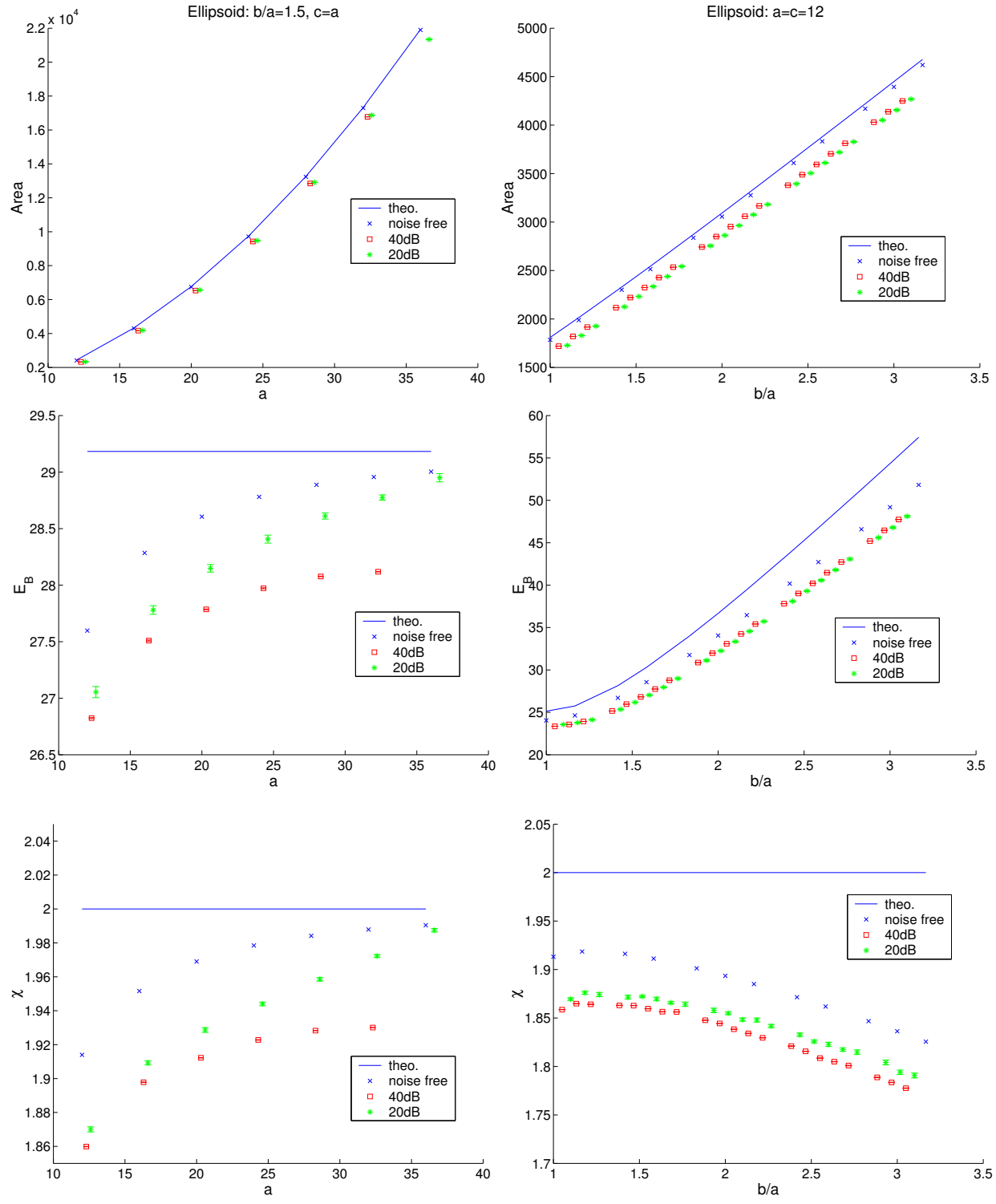


Figure 5.8: Surface area, bending energy and Euler characterization estimation at different noise levels averaged over 20 runs for a) scaled ellipsoid, b) deformed ellipsoid, $\sigma_g = 1, \sigma_T = 2, \sigma_k = 1$.

Torus

Finally we want to test the algorithm on an object that has an elliptic and hyperbolic part. For the spherical torus

$$((R + r\cos v)\cos u, (R + r\cos v)\sin u, r\sin v) \tag{C.12}$$

we obtain $K < 0$ for $v \in]\frac{\pi}{2}, \frac{3\pi}{2}[$ and $K > 0$ for $v \in]-\frac{\pi}{2}, \frac{\pi}{2}[$. The surface area of the torus is $4\pi^2 Rr$. The Euler characteristic is $\chi = 0$, as it is a sphere containing a handle. In figure 5.7a) the area estimation is shown, again it performs excellent in the noise free case and good for added noise. The same can be said for the Euler characteristic estimation in figure 5.7b). The measured points for 40dB lie a little beneath the noise free runs, whereas the 20dB runs lie above for the area estimation figure 5.7a). The masking of the integration measure $\|\nabla I\|$, done to avoid summing up components outside the surface, is not independent of the noise level, for lower noise levels a broader masking is permitted as for higher SNR this must be smaller. This dependency can however be avoided by using the more advanced masking method described above.

5.3 Application and comparison with existing work

We compare our method with the classical Hessian based isophote curvature and a method presented by Thirion and Gourdon [120]. They compute the curvatures directly from the first and second derivatives of the image. For further work on 3D curvature see for example [96, 85, 47, 54, 110]. In the Hessian approach the Hessian matrix is rotated to be aligned with the surface normal. Then the surface subspace Hessian is diagonalized and the curvatures are computed as these eigenvalues divided by the gradient magnitude [2, 135].

5.3.1 Torus test image

In figures 5.9a-e) we show different vertical slices through a synthetic 3D torus shell and the calculated mean curvatures $H = \frac{1}{2}(\kappa_1^2 + \kappa_2^2)$. In figure 5.9b) the estimation by our algorithm eq.(5.12) is shown, in c) the magnitude of the mean curvature by Thirion et al. [120], d) the mean curvature and in e) the numerator of the mean curvature by Thirion et al. [120]. Comparing figures 5.9c),d) and e) we see that the numerator is smooth, but that at the peak position of the ridge profile the normalization fails as the gradient magnitude is (nearly) zero. These positions display white in figure 5.9c),d). Note the sign change in figure 5.9c),d) on different sides of the ridge. In a coordinate independent description (arbitrary choice of the origin) the sign of curvature is not meaningful for ridges as the sign

cannot be defined consistently from local information only. For edges this is trivial as by convention the inside of objects is white.

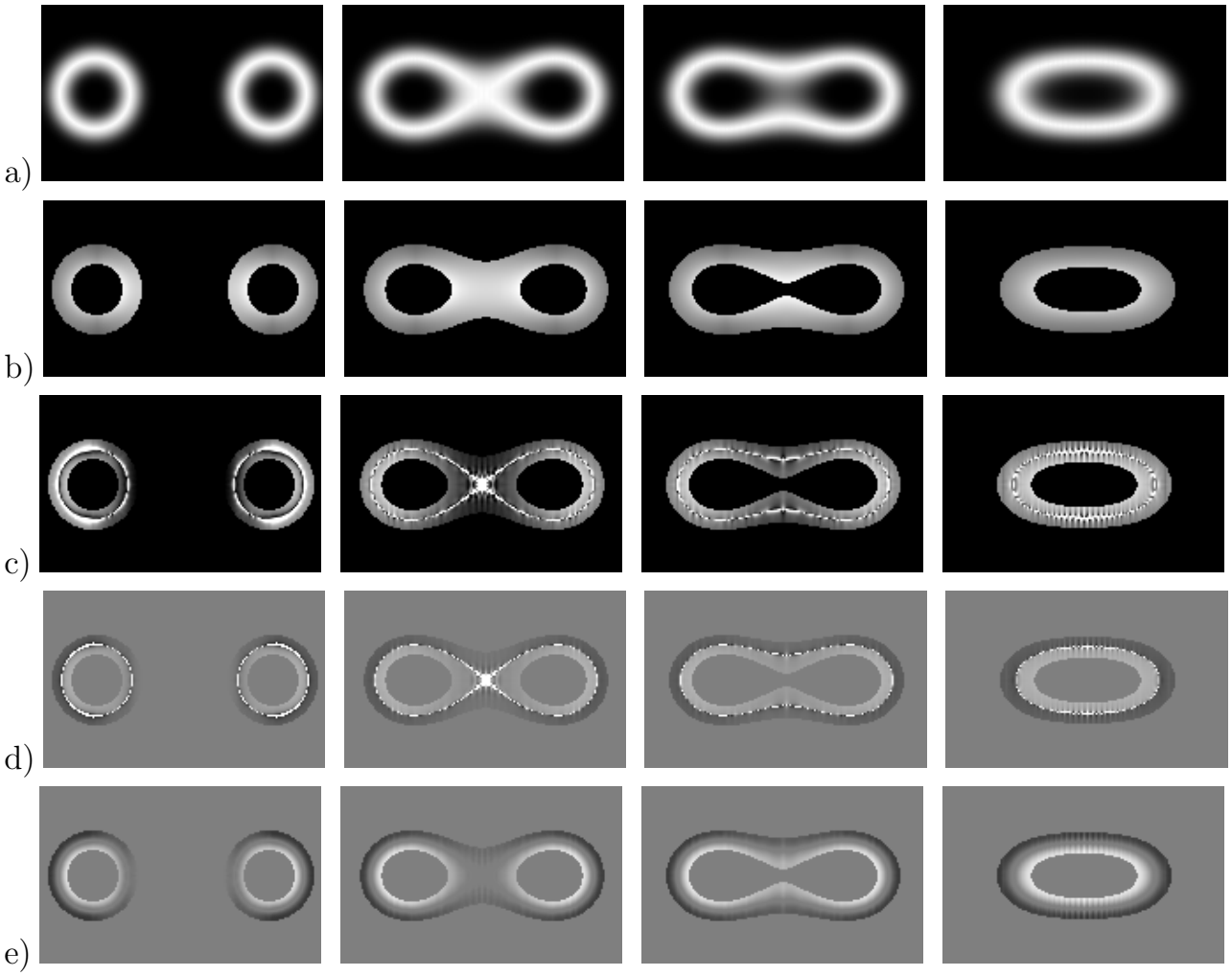


Figure 5.9: a) Vertical slices through a 3D torus shell with radii $R = 50$ and $r = 15$ and Gaussian ridge profile, b) mean curvature by eq.(5.12), $\sigma_g = 1, \sigma_T = 3, \sigma_k = 1$, c) magnitude of the mean curvature by Thirion et al. [120], d) mean curvature by Thirion et al. [120], e) numerator of the mean curvature by Thirion et al. [120] $\sigma = 1$.

5.3.2 Duplex board paper

In figure 5.10a) we show part of a pre-processed SEM scan of duplex board paper, from which for example milk cartons are made of. The image size is $3072 \times 768 \times 102$ voxels with a physical voxel size of $0.7 \times 0.7 \times 5\mu$m. The image was provided by StoraEnso Research, Sweden. We compute the curvature and bending energy via eq.(5.12) and the isophote curvature, see figure 5.10b) and c). The bending energy is only displayed for regions where fiber is present. With our algorithm the deformed parts are directly highlighted as bright parts, whereas the isophote curvature has difficulties dealing with the ridge-like cross-section of the image structure. The different paper fibers can be segmented [1] and then the bending energy can be integrated over each fiber. The deviation of their bending energy from a cylinder ($BE_{cyl} = l/r^2$) provides a measure for the deformation of the fiber. This contains more shape information than a pure fiber curl measure [1], where just the distance between start and end-point is compared to the middle line length. The deviation from a cylinder includes deformation of the surface and is therefore a 2D measure. Whereas the end-point-distance of the middle line is an intrinsic 1D measure. A 1D measure neglects squeezing (which can be wanted).

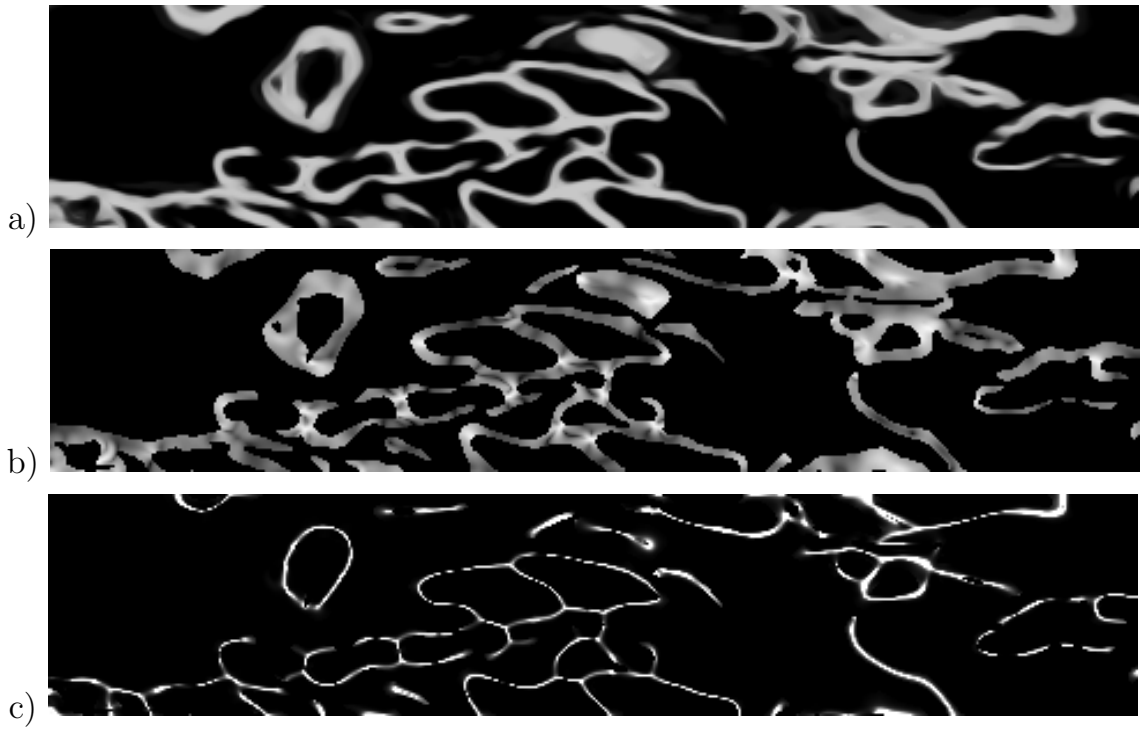


Figure 5.10: A $33 \times 67\mu$m region of a SEM paper scan at a depth of 73μm a) adaptive filtered input, b) bending energy via (5.12), log stretch, $\sigma_g = 1, \sigma_T = 3$ c) bending energy via the isophote curvature, range [0,0.5], $\sigma = 3$.

5.3.3 Seismic data

Curvature (and derived properties) can be used to interpret seismic data [107]. A typical seismic image is shown in figure 5.11a), which shows a strong layer structure and each layer represents a reflector. Tectonic forces can cause fractures in the subsurface rock leading to a vertical displacement of rock layers on both sides of the fracture. These fractures are called *faults*. The detection of these faults is of interest to the oil industry. Hand picking these faults is time consuming and can be difficult in 3D data. Automatic detection of strong faults with a nearly vertical displacement is possible and works reliable [4]. The detection of non-straight and weak faults is difficult. Curvature can help to find these weak faults.

Ordinarily, a layer/reflector of interest is segmented by hand and then curvatures are computed in a 3×3 or 5×5 neighborhood on the labeled 2D surface [107]. Our method can compute the curvatures of all (iso) surfaces in one image without segmentation. In figure 5.11b,c) we show the principal curvatures computed with eq.(5.12). Here again, isophote curvature by rotation of the Hessian or by Thirion's algorithm does not work due to the layer structure, compare figure 5.11d,e). The two step approach to compute the local structure at scale σ_T and the gradients at σ_g, σ_k shows it strength especially for this pattern of surfaces.

In general, seismic data is very large (on the order of thousands of pixels in each direction). Therefore, memory and computation time considerations are important. We have implemented our algorithm in the MATLAB toolbox DIPimage [76]. For an image of size $400 \times 400 \times 130$ the computation takes about 2 GB of memory and 15 minutes of CPU time on a Linux PC with an AMD 2000 processor. Thirions algorithm [120] needs about 1.3 GB of memory and 9 minutes of processor time.

5.4 Conclusions

We have demonstrated that our approach to estimate curvature and (local) shape descriptors on surfaces based on the differential structure of images is working excellent. It avoids problems associated with classical approaches such as matched filtering and polynomial fitting. It succeeds where standard isophote curvature estimation methods fail. Furthermore, we can conclude that the GST is able to estimate the principal directions on surfaces, as the computed curvatures in these directions are estimated accurately. Our new curvature estimation formula (5.12) is a consequence of the formulas of differential geometry taking into account the local structure of operators in grey-level images, e.g. circumventing the problem of non-unique orientation representation by using the Knutsson mapping.

In any application where patterns of ridge-like structures are present a robust estimation is not possible by the isophote curvature. First, on the ridges and val-

leys the gradient magnitude vanishes which results in a very high overestimation. Second, efficient noise reduction is not possible as it requires large smoothing kernels which smooth out the pattern of interest, such that the signal vanishes. Our algorithm can reliably estimate curvature in these cases.

The reason for these properties is that our algorithm uses a two step approach, first an orientation estimation process in a local neighborhood described by σ_T and then the derivative of the orientation field yields the curvature. In our algorithm σ_g, σ_k are the smoothing parameters for the first and second order derivative and correspond to the smoothing parameter of the isophote curvature. For $\sigma_T \to 0$ the GST returns the input vector as only eigenvector and our proposed method is equal to the isophote curvature. More explicitly, the added value of our estimation algorithm is a robust orientation estimate avoiding gradient magnitude normalization through the use of a local neighborhood σ_T.

The error of the computed curvature is dependent on the size of the curvature and its change in the local neighborhood. Nevertheless, the error stays small and the estimation is robust in the presence of noise. Surface area estimation and integration of functions over these embedded surfaces can be performed sampling-error free by relation (3.24). The area estimation has no bias and is independent of scale and local geometry. From these two ideas curvature based shape descriptors, the bending energy and Euler characterization, can be computed. The latter as a function of scale is closely related to the morphological granulometry [112, 13, 78]. Although a granulometry is a volume weighted distribution and the first only counts the number of objects visible at a certain scale. The estimation of these descriptors is consistent, robust and independent of the scale of the objects. It performs excellent for isotropic objects and small curvature ($\epsilon < 0.5\%$ for $\kappa < 0.03$). For highly asymmetric shapes and high curvatures the error stays small ($\epsilon < 7\%$ for $\kappa < 0.08$).

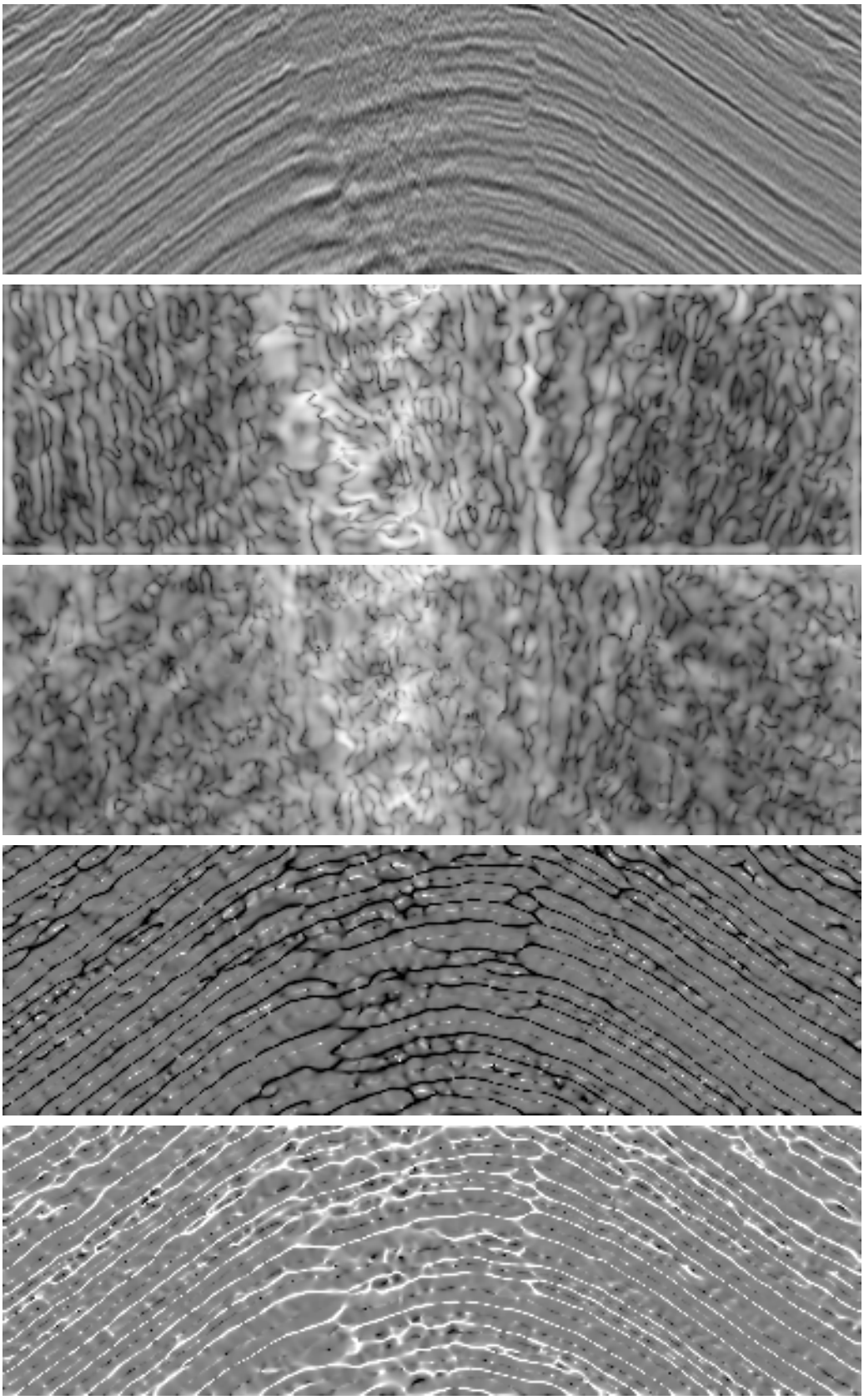


Figure 5.11: a) Slice of seismic image, b,c) principal curvatures by eq.(5.12), log stretch, d,e) principal curvatures by Thirion [120], range $[-1, 1]$.

Chapter 6

Motion in spatio-temporal images

6.1 Estimation of motion

Here we present and discuss techniques to measure velocities of moving objects or image parts in image sequences. A sequence can be a collection of 2D images taken with a standard photo camera at the speed that you can press the button or 4D imaging of a beating heart by MRI. We will focus the discussion on optic(al) flow via differential techniques and a tensor approach. We do not include a discussion on apparent motion, i.e. motion that is captured by projection. For example a standard camera records a 2D image of the 3D world. We assume that our recordings have the full dimensionality of the problem, therefore observed motion is true motion. For a comprehensive introduction of the different techniques see the textbooks [55, 115] and an overview article by Barron et al. [7] for a quantitative discussion of several methods.

6.1.1 Displacement from two successive images

If only two time frames are considered to estimated the velocity/shift, two methods come to mind: feature or region-based matching and a grey-value approach by first order Taylor expansion. Feature matching is essentially a two step process, where first, robust and descriptive features are computed and then corresponding features are matched. Different methods for matching are used, such as normalized cross-correlation and minimizing a distance measure. In industrial applications where there is a controlled scene and only a few feature present, this approach is often used. However, feature matching is not feasible for general scenes. Zero-crossings of grey-value functions have been proposed as a feature [82, 150] and found widespread use [50]. But it turned out that a multi-scale zero-crossing approach cannot describe even some special but simple structures correctly [24]. The local phase was used as a feature by Fleet et al. to compute the velocity [37]. From the conceptual point of

view feature matching is discrete in time. Furthermore, it has to balance the need for long time intervals for accurate displacement estimation and short time intervals for successfully finding corresponding features. In this two step category also falls color histogram tracking of people in natural scenes [92]. Tracking of segmented particles is successfully used in the study of turbulent gas and liquid flows. A controlled experimental setup enables a high contrast of the inserted marker particles, which are tracked in time. The amount of recorded data is generally very large so segmentation can reduce the amount of information drastically (several GByte). After segmentation the data is reduced to the point coordinates and can be handled efficiently. Recently Farnebäck presented a two frame estimation method based on the iterative region-based matching of polygons [35].

If the objects can be segmented from the background before the motion is estimated Diehl et al. [27] presented a fast and efficient method. The motion is modeled globally for a segment of the image and a few parameters describe the motion. The parameters are fitted via an iterative scheme. This method is more robust than local optical flow, but the image has to be segmented before the motion is estimated.

First order Taylor expansion

A grey-value approach to estimate the shift if only two images should be used is by first order Taylor expansion. This was presented by Luengo et al. for global shifts in images [58, 77, 133], but is also usable to obtain a local shift estimation by windowing. It turns out to be very similar to the optical flow equation solved with the constraint of constant motion in a surrounding neighborhood as presented by Lucas and Kanade [75], therefore it will be presented here. Let us start by investigating the Taylor expansion up to order r of a 2D image $I(x, y)$ around a point (x_0, y_0)

$$I(x,y) = I(x_0,y_0) + \sum_{i=1}^{r}\sum_{j=0}^{i}\binom{i}{j}\partial_x^j\partial_y^{i-j}I\Big|_{x_0,y_0}(x-x_0)^j(y-y_0)^{i-j} + \epsilon\,. \tag{6.1}$$

This reads up to first order in vector notation

$$I(x) = I(x_0) + \delta x \cdot \nabla I + \epsilon\,. \tag{6.2}$$

where ϵ is the error in the estimation and the shift $\delta x = (x - x_0, y - y_0)$. From now on x, x_0 will be considered vectors. Now we want to estimate the unknown shift between I_1 and I_2

$$I_1(x + \delta x) = I_2(x)\,. \tag{6.3}$$

Substituting the Taylor expansion for I_1 we obtain

$$I_1(x + \delta x) = I_1(x) + \delta x \cdot \nabla I_1 + \epsilon = I_2(x)\,. \tag{6.4}$$

From solely this equation it is not possible to obtain the shift vector unambiguously. This problem will be encountered again with the flow constraint eq.(6.27) and is known as the aperture problem. Only the normal component of the shift is given by eq.(6.4). Luengo et al. chose to use further the constraint of a local constant velocity, i.e. using more points in a small neighborhood to over-determine eq.(6.4). The error is minimized via the least square method in an area V

$$\delta x = \arg\ \min \frac{1}{V} \int\limits_V \left(I_2(x) - I_1(x) - \delta x \cdot \nabla I_1\right)^2 . \tag{6.5}$$

Setting the partial derivative with respect to δx to zero yields

$$\begin{aligned} 0 &= (I_2 - I_1 - \delta x \cdot \nabla I_1)\nabla I_1 && (6.6)\\ \delta x &= (\nabla I_1 \nabla I_1^t)^{-1}[(I_2 - I_1)\nabla I_1] . && (6.7) \end{aligned}$$

The integration over the volume V will be incorporated by a smoothing of the term $\nabla I_1 \nabla I_1^t$ by a Gaussian with standard deviation σ_T. This smoothing will be denoted by overlining be term. The derivatives ∇I are Gaussian derivatives with a standard deviation σ_g. In eq.(6.7) all terms must have the same magnitude to compare them, so also the first term in eq.(6.7) must be smoothed and we obtain finally the average shift in an area defined by σ_T

$$\Rightarrow \overline{\delta x} = (\overline{\nabla I_1 \nabla I_1^t})^{-1}[\overline{(I_2 - I_1)\nabla I_1}] . \tag{6.8}$$

For global shift estimation ($V = I$) this method is biased toward smaller magnitudes as M. Bezuijen pointed out during this bachelor project in our group [9]. Assume we have the true shift δu then by inserting the Taylor expansion for I_2 from eq.(6.4) in eq.(6.8) we obtain

$$\begin{aligned} \overline{\delta x} &= \overline{(\nabla I_1 \nabla I_1^t)}^{-1} \overline{(\delta u \nabla I + \epsilon)\nabla I_1} && (6.9)\\ &= \overline{\delta u} + \overline{(\nabla I_1 \nabla I_1^t)}^{-1} \overline{\epsilon \nabla I_1} . && (6.10) \end{aligned}$$

Let us now investigate the sign of the $\overline{\epsilon \nabla I}$, as $\overline{(\nabla I \nabla I^t)}$ is semi-positive definite [42] appendix B. We limit us to 1D and let N be the number of pixels of I

$$\begin{aligned} \overline{\epsilon \frac{dI}{dx}} &= \frac{1}{N} \sum_I \sum_{i=2}^{\infty} \frac{1}{i!} \delta u^i \frac{d^i I}{dx^i} \frac{dI}{dx} && (6.11)\\ &= \frac{1}{N} \sum_I \sum_{i=1}^{\infty} \frac{1}{(2i+1)!} \delta u^{2i+1} \frac{d^{2i+1} I}{dx^{2i+1}} \frac{dI}{dx} + && (6.12)\\ &\quad \frac{1}{N} \sum_I \sum_{i=1}^{\infty} \frac{1}{(2i)!} \delta u^{2i} \frac{d^{2i} I}{dx^{2i}} \frac{dI}{dx} && (6.13) \end{aligned}$$

The even terms of δu (6.13) vanish as $(f_x, f_{xx})_{L^2} = (i\xi\hat{f}, -\xi^2\hat{f})_{L^2} = 0$[6]. Odd powers of δu (6.12) preserve the sign and the inner product of the first derivative with an odd derivative is

$$(f_x, f_{odd})_{L^2} = \int (-i\xi\hat{f})(-(i\xi)^{2k+1}\hat{f}) \quad k \in \mathbb{N} \tag{6.14}$$

$$= i^{2k} \underbrace{\int \hat{f}^2 \xi^{2k}}_{>0} \quad k = 2, 3, \ldots . \tag{6.15}$$

The sign of the odd terms is alternating, neglecting 5^{th} order and higher we obtain

$$\text{sgn}\left(\overline{\epsilon\frac{dI}{dx}}\right) = \begin{cases} + & \text{for} \quad \delta u < 0 \\ - & \text{for} \quad \delta u > 0 \end{cases}, \tag{6.16}$$

i.e. the magnitude is always underestimated. This is true, as long as the integration interval is symmetric around the origin. This may not be the case for local shift estimation in an area defined by σ_T.

Cross correlation

Cross-correlation is another popular method to estimate the shift between two images. It is based on the shift property of the Fourier transformation. Let I_2 be the shifted version of I_1

$$I_1(x + \delta x) = I_2(x)\,, \tag{6.3}$$

then their Fourier transforms are related as

$$\hat{I}_1(\xi) = \hat{I}_2(\xi) e^{-i\xi\,\delta x} \tag{6.17}$$

and

$$\frac{\hat{I}_1^* \hat{I}_2}{|\hat{I}_1||\hat{I}_2|} = e^{+i\xi\,\delta x} \tag{6.18}$$

yields spatially a delta peak at the shift location δx. Equation (6.18) is spatially the cross-correlation between I_1 and I_2 with subtracted means. This method is standard only usable for integer pixel shifts, but can be extended to sub-pixel accuracy up to 0.1 pixel [142].

In the study of flow phenomena a technique called Particle Image Velocimetry (PIV) is used. It is mainly based on cross-correlation, although also tracking is used in the field [141, 142, 143]. A sheet-of-light is shine on the flow, in which small tracer particles where introduced. They reflect the light, which is recorded with

[6] $(f, g)_{L^2} := \int_{\mathbb{R}^n} f g^*$

a high speed camera. Each image (which can be up to 1024×1024) is divided in smaller subregions, and then in corresponding windows cross-correlation is used to estimate the shift in this window. A typical window, as a rule of thumb, should contain about 7 particles to estimate the velocity accurately [141].

6.1.2 Motion as orientation in (x, t)-space

For the moment we will only consider a spot as our moving object. Let the spot be located in an nD spatial image. Then the moving spot is located in a $(n+1)$D spatio-temporal image. In the terms of special relativity, the spot is an *event* or *world point* (x, t). The propagation of the spot described by the curve $(x(t), t)$ is then called a *world line* [111]. A moving spot in 2D will form a string of spaghetti in the spatio-temporal 3D image. For 1D movement as shown in figure 6.1 the

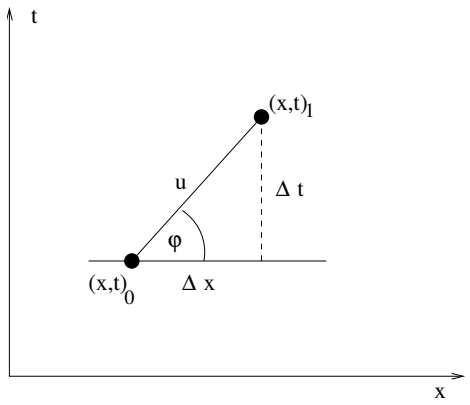


Figure 6.1: Movement of a spot in 1D

velocity is

$$v = \frac{\partial x}{\partial t} \approx \frac{\Delta x}{\Delta t} = \cot \varphi, \ \varphi \in [0, 180^\circ] \,. \tag{6.19}$$

An angle of $\varphi = 90^0$ corresponds to zero velocity and an angle of $\varphi = 0^\circ$ to infinite velocity. We see the similarity with a Minkowski diagram, where an angle of 45° corresponds to $v = c$.

From this it is clear that motion appears as orientation in space-time images. Conceptually different from the two time frame approach is the fact that the velocity can be estimated directly as orientation instead of a discrete displacement. This point of view also treats motion as continuous, which in turn enables us to develop algorithms that can be checked analytically before quantization. Furthermore, there are a few advantages coming along with the use of more than only two images as a more robust and sub-pixel estimation of the velocity. Fourier domain analysis is possible from; (x, t) to (k, ω).

There are four main approaches to obtain the velocity from the orientation in

a spatio-temporal image: the gradient structure tensor, steered quadrature filters, Fourier based and differential methods.

Gradient structure tensor

Directly from figure 6.1 we see that the gradient structure tensor (GST) is well suited to estimated the velocity. The GST is a generic tool to analyze local dimensionality (compare appendix A)

$$\bar{G} := \overline{\partial_\alpha \partial_\beta I}, \quad \alpha, \beta = 1, \dots n+1\,. \tag{6.20}$$

A moving point forms a line and a moving line a plane. The GST can detect these structures and estimated their orientation. The detection is based on the ratio of the eigenvalues, which is for lines $\frac{\lambda_n - \lambda_{n+1}}{\lambda_n + \lambda_{n+1}}$ as described in appendix A. The orientation of the structure is returned and from it the velocity can be computed in 1D as eq.(6.19). Now we make the step from 1D to nD movement. The spatio-temporal image will be $(n+1)$D and an eigenvalue analysis of the GST will provide the smallest eigenvector u along the world line (figure 6.1). We can retrieve the components of the velocity vector as follows:

$$v_i = \frac{\partial u_i}{\partial t} \approx \frac{\Delta u_i}{\Delta u_t} \; 1 \le i \le n\,. \tag{6.21}$$

For 2D motion the eigenvector u is given by two angles in spherical polar coordinates $\varphi \in [-\pi, \pi]$ and $\vartheta \in [0, \pi/2]$[7]

$$v = \begin{pmatrix} \frac{\cos\varphi \sin\vartheta}{\cos\vartheta} \\ \frac{\sin\varphi \sin\vartheta}{\cos\vartheta} \end{pmatrix}. \tag{6.22}$$

Even if the GST (or in fact any local dimensionality analysis) gives only orientation information (see chapter 2 and appendix A) we can retrieve the sign of the velocity. Here we assume the last component (time) to be positive, this introduces global knowledge and allows us to make the step from orientation to direction. So we can solve the discontinuity in an appropriated manner, i.e. do a point reflection of all vectors u which have negative u_t

$$\tilde{u}(x,t) = \begin{cases} -u & u_t < 0 \\ u & u_t \ge 0 \end{cases}. \tag{6.23}$$

One of the first to use the structure tensor approach - not only for velocity estimation - were Bigün et al. [11].

The acceleration in a (x,t)-image is the curvature of the trajectory

$$a = |\dot{v}|T + |v|^2 \kappa N\,. \tag{4.24}$$

The curvature can be estimated as explained in chapter 4.

[7] The fact that only orientation information is present is expressed by restricting $\vartheta \in [0, \pi/2]$.

Orientation selective quadrature filters

The main point here is to apply a set of velocity (orientation) selective filters to a (x,t)-image and compare the filtered results. This approach is capable to distinguish different orientations in a pattern that locally contains more than one orientation. This is in contrast to the differential approaches discussed later. However, there are a few conceptual problems involved in sampling of the orientation [42] chapter 3, which becomes even more complex in four or higher dimensions.

An approach by Knutsson [62] and later Bårman [3] that is especially suited is the use of orientation selective quadrature filters. A quadrature filter is formed by the filter and its Hilbert transform, resulting in a phase invariant filter. The set of filters should be similar to each other, and polar separable. Knutsson introduced a set of directional quadrature filters [62]. In 3D Knutsson showed that 6 or 10 quadrature filters can be symmetrically placed in Fourier space [64]. Up to recently the Hilbert transform was only well-defined for 1D, Felsberg and Sommer [36], Larkin et al. [70] and Verbeek [125] proposed independently a multi-dimensional Hilbert transform. Knutsson recently presented filters called 'loglets' that combine the multi-dimensional Hilbert transform with quadrature filters and introduce the concept of a generalized phase [65].

Energy/Fourier domain techniques

Imaging an image sequence $I(x,t)$ where all objects are moving in the same direction with velocity v, which is described by

$$I(x,t) = I(x - vt)\,. \tag{6.24}$$

This is the general solution of the dispersion-free wave equation $\Delta\Psi - \frac{1}{v}\partial_t^2\Psi = 0$ and therefore the Fourier transform of eq.(6.24) is

$$\hat{I}(k,\omega) = (2\pi)^{-n/2}\hat{I}(k)\delta(kv + \omega)\,. \tag{6.25}$$

The velocity is then computed by the slope of the (hyper-) plane in which the spectrum lies, which is given by the argument of the delta function. The slope cannot be determined if the spectrum lies on a line (for spatially oriented patterns). Only the velocity in the normal direction can be estimated here. This is known as the *aperture problem* and it will show up again with the differential methods.

Differential methods

This set of techniques computes the velocity from spatio-temporal derivatives. Assume that the intensity is conserved

$$\frac{dI(x,t)}{dt} = 0\,, \tag{6.26}$$

then we obtain the flow equation

$$\nabla I \cdot v + \partial_t I = 0\,, \tag{6.27}$$

where $\nabla = \partial_i, i = 1 \ldots n$ only operates on the spatial components. It is also known as the gradient constraint equation [7] or brightness change constraint equation [55] chapter 4. Eq.(6.26) is equivalent to eq.(6.3) when only two images are considered.

In eq.(6.26) we made the implicit assumption that the velocity is constant, i.e. we start with a function $I(x(t),t)$. Our estimated velocity will only be correct given by eq.(6.27) if the velocity is constant. If we start also with a velocity dependency $I(x(t),\dot{x}(t),t)$ then we get

$$\frac{dI}{dt} = \partial_t I + \nabla I v + \nabla I a\,, \tag{6.28}$$

from which we cannot extract the velocity estimation by a straightforward closed form solution.

However, even with a constant velocity the flow eq.(6.27) is not sufficient to retrieve all components of the velocity vector v. The flow equation for nD motion is a scalar equation and contains n unknown vector components. In effect, the scalar product $\nabla I \cdot v = \|\nabla I\| v_\perp$ and therefore we can only retrieve the motion normal to the edge. This is called the *aperture problem*. We encountered this before in the shift estimation for two images and also in the Fourier based approach. See for example a paper by Florack et al. [38], which uses measurement duality into forming a generalized flow constraint equation by including preprocessing filters.

In the following we will discuss different techniques to compute the full velocity vector. They all impose another constraint on the image sequence $I(x,t)$. Possibilities are: to assume a global smoothness constraint, use several points in a small neighborhood and assume that $v = const.$ in this neighborhood or perform a local modeling of the spatial structure, i.e. put a constraint on the second derivative $\frac{d\nabla I}{dt} = 0$.

- Hessian or differential geometry modeling
  Nagel [86] and Tretiak [121] were the first in using second order derivatives to compute the optical flow. If we restrict the intensity in such a way that

there should be no first order deformations, e.g. rotations and dilations, this leads to

$$\frac{d\nabla I}{dt} = 0 \tag{6.29}$$

$$\Rightarrow \partial_{ij} I \cdot v + \partial_i \partial_t I = 0\,. \tag{6.30}$$

This is a stronger requirement than eq.(6.27) and is (in most cases) capable of computing all velocity components. However, second order derivatives are quite noise sensitive and need more regularization to compensate high frequency noise. Therefore, second order approaches are generally assumed to be less accurate than first order ones.

- Global smoothness constraint
  This method is based on a paper of Horn and Schunck [52]. They combine the gradient constraint from eq.(6.27) with a global smoothness constraint on all components of the velocity with smoothness parameter λ

$$\min_v \int dx\,(\nabla I \cdot v + \partial_t I)^2 + \lambda^2 (\sum_i \|\nabla v_i\|^2)\,. \tag{6.31}$$

  The last term is equal to $\mathrm{tr}\,[\nabla v^t \nabla v]$, where the nabla of a vector results in a $n \times n$ tensor and is defined as $\nabla v := \partial_i v_j$. Nagel [86, 87] uses, like Horn and Schunck, a global constraint but on the orientation

$$\min_v \int dx\,(\nabla I \cdot v + \partial_t I)^2 + \alpha^2 \mathrm{tr}\,[\nabla v^t (B^t \mathbf{W} B)^{-1} \nabla v]\,, \tag{6.32}$$

  where the matrix $(B^t \mathbf{W} B)$ is defined in [87]. $(B^t \mathbf{W} B)^{-1}$ is essentially the gradient structure tensor, although the connection is not mentioned there. This is similar to flow computations via anisotropic diffusion by Weickert [138].

- Weighted least-square fit of local first-order constraints
  This constraint was proposed in the computation of optic flow by Lucas and Kanade [75]. The velocity is assumed to be constant in a small neighborhood $V = (\Delta x, \Delta t)$, then we can use a least square estimation

$$\min_v \frac{1}{V} \int_V (\nabla I v + \partial_t I)^2\,. \tag{6.33}$$

  Minimizing the above equation by setting the partial derivative with respect to v to zero yields

$$\frac{\partial \epsilon}{\partial v} = \nabla I \nabla I^t v + \nabla I \partial_t I = 0 \tag{6.34}$$

$$v = -(\nabla I \nabla I^t)^{-1} \nabla I \partial_t I\,. \tag{6.35}$$

The integration over the volume is done by smoothing with a Gaussian with σ_T over the tensor elements $\nabla I \nabla I^t$ before inverting it. The spatial derivatives have incorporated a smoothing of size σ_g and the time derivative a smoothing of size σ_t. The same magnitude of all terms in eq.(6.35) is ensured by smoothing $\nabla I \partial_t I$ with σ_T

$$\overline{v} = -(\overline{\nabla I \nabla I^t})^{-1} \overline{\nabla I \partial_t I}\,. \tag{6.36}$$

The first term in the equation is the formula for the GST, but it only contains spatial derivatives (and smoothing in the time direction) and is not to be confused with eq.(6.20). This formula is very similar to eq.(6.8) for the shift equation from two images, the difference is replaced by the time derivative.

This method delivers accurate results even if the grey-value structure cannot be described by a first-order Taylor expansion as long as the velocity is constant. There is no systematic error [55] chapter 6.4. This is in contrast to the shift estimation from two images only by first order Taylor estimation as discussed before in eq.(6.8). The difference is, here we have the temporal derivative $\partial_t I$ in contrast to the finite difference $I_2 - I_1$ for the Taylor approximation. The Taylor expansion incorporated in eq.(6.27) is not an approximation of the structure as in eq.(6.4) but of the velocity field.

Suppose we have the following constant motion field: $I(x,t) = I(x - ut)$. From the flow equation (6.27) we get $\nabla I \cdot u = -\partial_t I$ and substituting this in eq. (6.36) we get

$$\begin{aligned} \bar{v} &= -(\overline{\nabla I \nabla I^t})^{-1} \overline{\nabla I \partial_t I} && (6.37)\\ &= -(\overline{\nabla I \nabla I^t})^{-1} \overline{-u \nabla I \nabla I^t} && (6.38)\\ &= \bar{u}\,. && (6.39) \end{aligned}$$

This holds as long as the inverse $\overline{(\nabla I \nabla I^t)}^{-1}$ exists and the velocity is constant in the smoothing window. For noise hampered optic flow the estimate is biased towards lower values [55].

Furthermore, we address two other important points: density and confidence measure. Density is the fraction of the image where the optic flow is computed. For some regions this is not possible without regularization, as the gradient based methods rely on gradient information. Imagine a moving square, only at the borders gradient information is available, and only there the optic flow is computed. The statement of the density is always needed when comparing two techniques. The standard error measure is the average (per time step over the whole image) angle between the theoretical and measured flow in degrees. A confidence measure should also always be calculated for the whole spatio-temporal image. The differential methods rely on a local 1D image structure, if this is not the case (crossing patterns, merging points) the velocity cannot be computed reliably.

Orientation estimation in 4D data

Time sequences of 3D images yield huge amounts of data to process and store, even if there are only a dozen time frames as in common biological applications. A typical *isotropic* 4D images acquired in a confocal time series of a cell is of the size $256 \times 256 \times 100 \times 20$, (x, y, z, t). Given the data is recorded with 8 bits resolution the memory needed to store the image is 125MB. For further computation of gradients at least a resolution of 32 bits is needed (single float). Therefore the gradient alone would need 2GB.

Here we discuss how the orientation (velocity) can be computed in a GST or Lucas and Kanade implementation memory efficient. A state of the art i386 architecture computer can only handle up to 3-4GB of memory. The GST $\bar{G} = \overline{\partial_\alpha I \partial_\beta I},\ \alpha, \beta = 1 \ldots 4$ has 10 independent components from which the eigenvectors and values can be calculated. For the computation of the individual components of the GST only two gradient components are needed at a given time, further the result is smoothed to achieve averaging over a neighborhood, scale σ_T (typical size $\sigma_T = (4, 4, 4, 1)$). It is not necessary to store the elements of the GST $G_{\alpha\beta}$ at the original resolution, but it is sufficient to store a sub-sampled version by the factor of σ_T without losing information [132]. This yields a major reduction in memory use, needing only 104MB instead of 4.9GB to store the GST elements.

For linear structures (a moving point) the smallest eigenvector corresponds to the orientation, see appendix A. We can compute it numerically by inverting the GST and then use the power method to compute the largest eigenvalue of the inverted GST [44] chapter 7. The maximal memory usage is:
$\max(4 * vol(I), 29 * vol(I)/vol(\sigma_T))$, where we make explicit use of the symmetry of the GST in the implementation.

A full eigenvalue analysis of this symmetric problem can be done numerically using the cyclic Jacobi method [44] chapter 8. The eigenvalues are computed by stepwise rotation of the initial matrix to bring it on diagonal form. Here we did not code the involved matrix computations explicitly as being (anti)symmetric. Therefore the memory requirements are 66 copies of the sub-sampled image. This is not problematic as the computation of the structure tensor elements at full resolution is the memory critical task. The procedure does not sort the eigenvalues, but this can be done afterwords quite fast.

6.1.3 Conclusions

In the remainder of this thesis we will use the assumption of local constant velocity and least squares fitting as proposed by Lucas and Kanade [75] or the GST. The volume of constant intensity and velocity, defined by σ_T, is often easy to choose.

If that fails one could use automatic scale selection by Lindeberg [74]. Niessen [89] treats the scale selection problem especially for the problem of optic flow. As a confidence measure for a local 1D neighborhood we will use the GST eigenvalues λ_i of the $(n+1)$D spatio-temporal image (see appendix A)

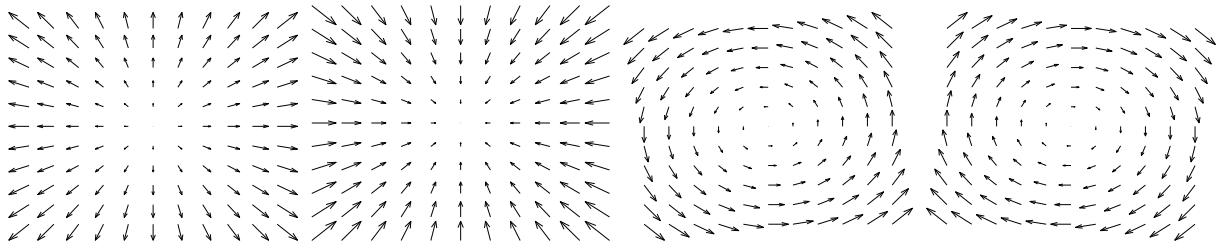
$$c_{1D} = \frac{\lambda_n - \lambda_{n+1}}{\lambda_n + \lambda_{n+1}}, \quad \lambda_i \geq \lambda_{i+1}\,. \tag{6.40}$$

Global approaches rely too strongly on the fit of the global constraint, i.e. if we assume a smooth model by choosing a high smoothness parameter then the evaluation of eq.(6.31) or (6.32) on the data also yields that.

The detection of 2D flow discontinuities has been discussed by Zetzsche and Barth [152] with the help of 3D curvature operators. The fastest algorithm at the moment is by Farnebäck [34]. Flow estimation of fluid can be done by using more apriori knowledge by incorporating the appropriated boundary conditions. Nakajima et al. used the continuity equation and the Navier-Stokes equation to obtain the additional constraint to solve the flow equation (6.27) [88].

6.2 Global motion analysis using differential operators on the flow field

Once the flow field is calculated by one of the methods described above, qualitative and quantitative information can be extracted. Apart from the local velocity (direction and magnitude), a vector field - static or time dependent - can have qualitative global patterns. Next to simple translation there are four basic types of variations as shown in figures 6.2a-d). They are source, drain, left and right turning rotation. These measures describe non-local properties of a vector field. Mathematically a source/drain is called a positive/negative *divergence* and a left/right



(a) **(b)** **(c)** **(d)**

Figure 6.2: Four basic types of a vector field. Central fields: a) Source, b) Drain; Rotation: c) left/positive turning, d) right/negative turning

turning a positive/negative *curl* or *rotation*. Let v be a $\mathcal{C}^1$ vector field over $\mathbb{R}^3$, then

$$\text{div } v = \nabla \cdot v = \sum_i \partial_i v^i \,, \tag{6.41}$$

$$\text{rot } v = \nabla \times v = \sum_{jk} \varepsilon_{ijk} \partial_j v^k \,, \tag{6.42}$$

with ε_{ijk} the Levi-Civita permutation tensor [17]

$$\varepsilon_{ijk} := \begin{cases} 0 & \text{for } i = j, i = k, j = k \\ +1 & \text{for cyclic/even permutation of } ijk \\ -1 & \text{for anti-cyclic/odd permutation of } ijk \end{cases} . \tag{6.43}$$

The global descriptors, rotation and divergence, are computed locally by derivative operators. A geometric interpretation of the divergence is given by the Gauss theorem. It states that the flow through a closed surface S is the integrated divergence of the enclosed volume V

$$\oint_S v \cdot \hat{n} \, dS = \int_V \text{div } v \, dV \,. \tag{6.44}$$

For the rotation or curl Stoke's theorem gives that the flow through a surface S is the integral of the field along (ds) the border of the surface ∂S

$$\int_S \text{rot } v \cdot \hat{n} \, dS = \oint_{\partial S} v \cdot \hat{t} \, ds \,. \tag{6.45}$$

The normal unit vector is $\hat{n}$ and the tangent vector $\hat{t}$. A vector field can only be written as a derivative of a potential function if its rotation vanishes [17]. A field with rot $v = 0$ is often called irrotational. In particular, all gradient fields are irrotational. The quantity rotation is proportional to the angular velocity.

In terms of image processing all these measures have to be evaluated at a certain scale, given by the smoothing of the partial derivatives, defining the meaning of global. A multi-scale approach can be used to find regions of different qualities, as vortexes in a turbulent flow. A vortex is a critical point in the vector field (the middle point of the rotation or divergence).

Let us investigate the filter response of the divergence and rotation operators on a 2D flow field. In figure 6.3a) we show 50 randomly placed Gaussian blobs which move away from the coordinate axes with different velocities ($v_{max} \leq 1 \frac{\text{pixel}}{\text{frame}}$, thus satisfying the sampling criterion eq.(3.7), (3.33)). Here we can conclude from $v_{max} \leq 1 \frac{\text{pixel}}{\text{frame}}$ that the signal is truly aliasing free, as we constructed the image sequence. The size of the image is $(x, y, t) = 128 \times 128 \times 50$, where $t = 15$ is shown. In figure 6.4b) 50 counter-clockwise rotating blobs with constant angular velocity

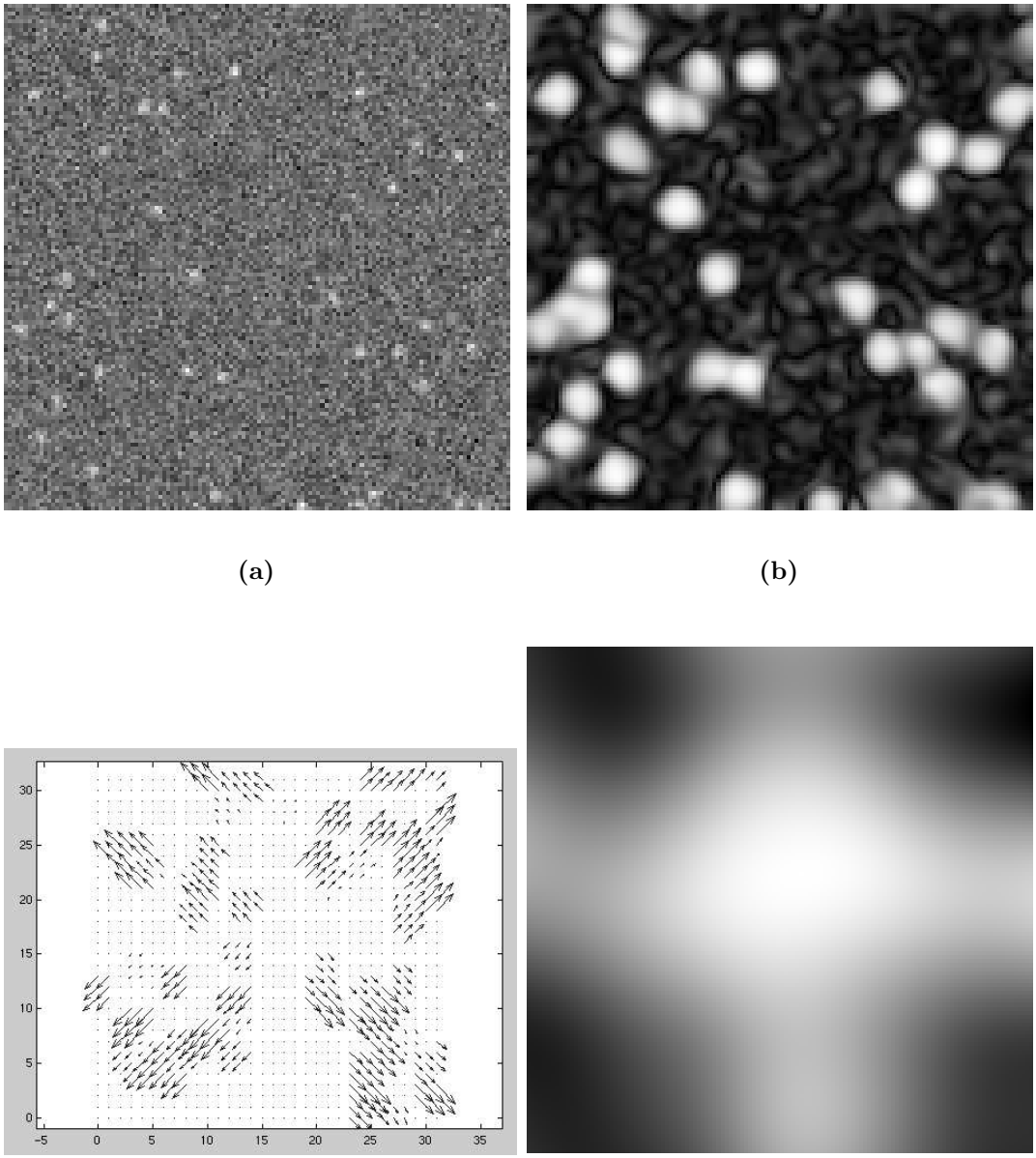
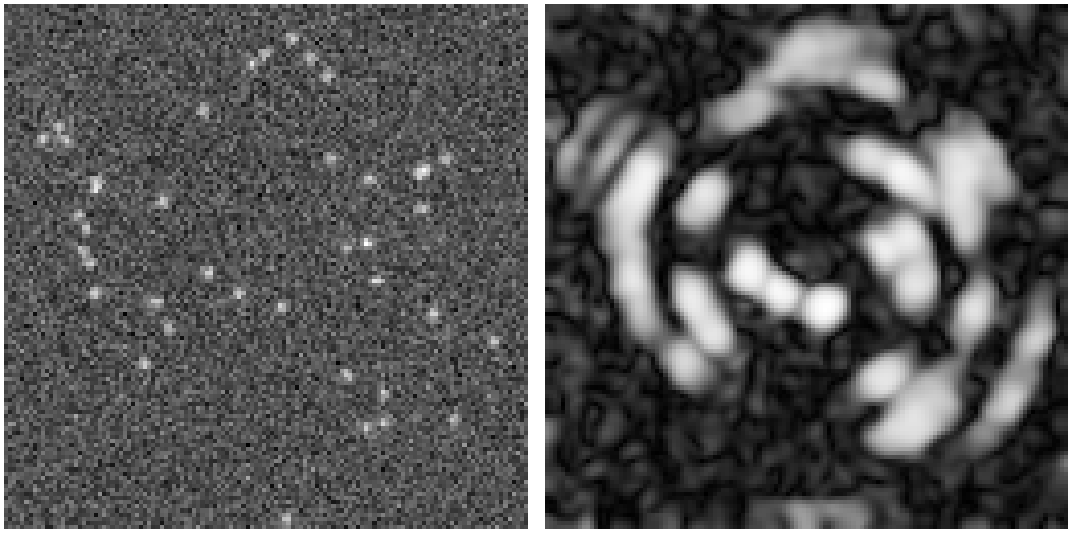
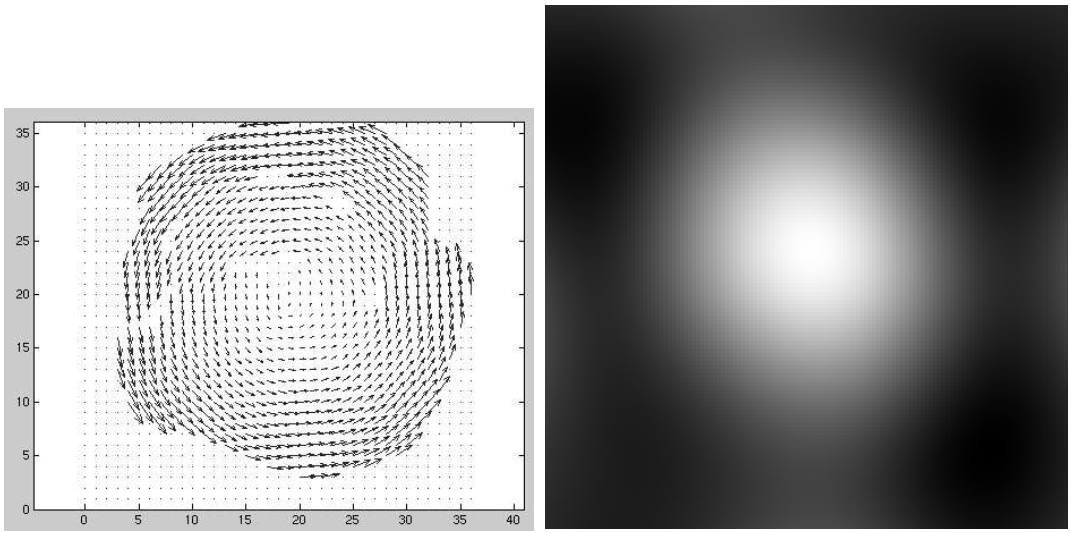


Figure 6.3: a) Random placed blobs with 20dB noise in a image of size $(x, y, t) = (128, 128, 50)$, b) confidence measure c_{1D} eq.(6.40), c) flow field at $c_{1D} > 0.7$ with $\sigma_g = 1, \sigma_T = 2$, d) divergence with $\sigma = 20$. All displays linear stretched.



(a) **(b)**



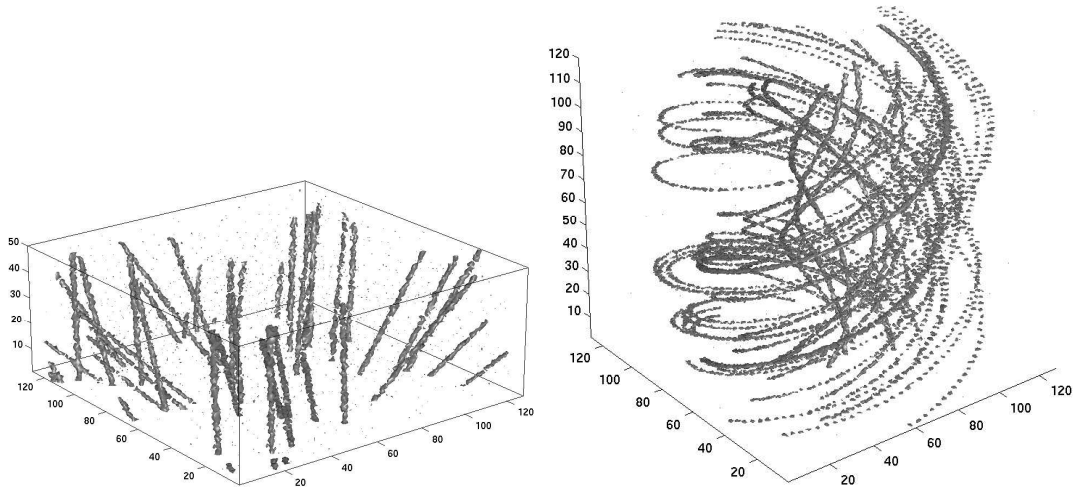
(c) **(d)**

Figure 6.4: a) Random placed rotation blobs with 20dB noise in a image of size $(x, y, t) = (128, 128, 120)$, b) confidence measure c_{1D} eq.(6.40) with $\sigma_g = 1, \sigma_T = 2$, c) flow field, d) rotation/curl with $\sigma = 20$. All displays linear stretched.

$\omega = \frac{2\pi}{120}$ are shown. The size of the image is $(x, y, t) = 128 \times 128 \times 120$, where $t = 60$ is shown. Both time series are hampered with 20dB Gaussian noise.

In figures 6.3,6.4b) the confidence measure c_{1D}, eq.(6.40) is shown and in c) the flow is visualized at those points with high line-likeness ($c_{1D} > 0.7$). The response of the divergence and rotation is shown in d) which are calculated with $\sigma = 20$ (large scale). The divergence gives the highest response in the center and high values along the coordinate axes from which the blobs are moving away. The response is positive indicating a source. The rotation is also positive, which indicates an anti-clockwise rotation and yields a high response in the rotation center.

In figure 6.4b) the confidence for the blobs in the middle is higher than at the image borders, this is in contrast to the confidence measure in figure 6.3b). Here we must remember that the points in figure 6.4 rotate with a constant angular velocity ω around the center. As the actual velocity is $v = \omega r$, the velocity is increasing linearly with the distance from the center, thus we have $v < 1 \frac{\text{pixel}}{\text{frame}}$ for $r \lesssim 20$ and proper sampling as required in eq.(3.33). Until $r < 40, v < 2$ the zero frequency of the Fourier spectrum is not aliased. Only in the outermost part $r > 40$ all frequencies are aliased. In figure 6.5 the (x, t)-images of the flows figure 6.3, 6.4a) are shown. In the divergent case a) the blobs form lines in (x, t)-space, but in b) blobs far from the center are too far apart in subsequent time slices to be seen as a line (at the same scale as in the inner part), therefore the confidence in figure 6.4b) is low at the image borders.



(a) **(b)**

Figure 6.5: (x, t)-image of a) diverging points, b) rotating points.

6.3 Brownian motion

Small suspended particles in a solution move in a random manner in the absence of any flow in the solution. Brown was the first to observe this motion under the microscope of dissolved flower pollen [18]. The cause of this motion is the collision of the suspended particles with the molecules of the solution. If you zoom in on a certain particle then it will move randomly in all directions and the measured displacement in fixed time intervals will vary greatly. On average one observes a typical traveled distance λ, where there is no preferred direction. The probability to find a particle in a certain position after a time t can either be calculated from a random walk process [46] chapter 5 or as the solution to the diffusion equation. The diffusion equation

$$(\partial_t - D\Delta)f(x,t) = 0 \tag{6.46}$$

with diffusion constant D describes such a random motion. The fundamental solution for nD of eq.(6.46) is

$$f(x,t) = \frac{\Theta(t)}{(2\sqrt{Dt\pi})^n}\, e^{-\frac{x^2}{4Dt}}\,, \tag{D.3}$$

where $\Theta(t)$ is the is the Heaviside step function

$$\Theta(t) := \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}\,. \tag{6.47}$$

The average quadratic displacement (variance) is then

$$\left\langle x^2 \right\rangle = \int_{-\infty}^{\infty} x^2 f(x,t)\, dx = 2nDt\,. \tag{D.9}$$

See appendix D for a short derivation of these two equations. A relation between the viscosity of the solvent medium η and the diffusion constant D of the dissolved particles is given by the Einstein relation [32]

$$D = \frac{k_B T}{6\pi\eta r}\,. \tag{6.48}$$

The denominator is the friction coefficient $\mu = \frac{v}{F}$ of a sphere with radius r in a fluid with laminar velocity v given by Stokes' law.

As a consequence of the underlying physics Brownian motion cannot be sampled correctly. However, here it is not of interest to measure the velocity between two collisions but to estimate the parameters of this stochastic motion, i.e. the average displacement over a longer period of time. From this displacement one of the

following material constants D, η or r can be estimated. To do this, the positions of many particles have to be estimated at time t_0 and $t_0 + \Delta t$, where Δt should not be chosen too small (after 30-60s in a water based solution particles with a radius $\approx 1\mu$m should displace on average $6 - 9\mu$m). The particles must be tracked in time to establish correspondence over the time interval Δt. Per particle the average displacement $\sqrt{\langle x^2 \rangle}$ is measured by subtracting the (sub-pixel) positions at frame i and $i + \Delta i$.

The above described method is only applicable if before hand it is known that the motion is Brownian. The recognition of Brownian motion or distinction from motion with constant velocity or acceleration is not possible by investigation of one time interval Δt. The position has to analyzed as a function of time. If this reveals a connection $\sqrt{\langle x^2 \rangle} \propto \sqrt{t}$, then we observe a diffusion process. The discrimination between linear and square root behavior as a function of time cannot be made in the region $t \approx 1$, as the square root is nearly linear there.

As a side note we mention that the Nyquist noise of a resistor is caused by this phenomenon and that thermal noise for example limits the precision of our ears and other uncooled high precision apertures.

Chapter 7

Application 1: Flow patterns in French bread during proofing[8]

Here an application is presented of optic flow measurements on the rising of bread dough.

The crumb structure of bread is an important factor determining the perception of bakery products by the consumer. There is an interest in the controlled manipulation of the bread crumb structure, however this requires insight into how this aspect of the micro-structure evolves during the bread making process. Non-invasive monitoring of gas cells is possible by X-ray tomography. This technique is not able to observe gas cells with small diameters ($<$ 1mm). Magnetic Resonance Imaging (MRI) is a promising technique for the assessment of gas cell development and anisotropy in the growth of the dough during proofing. At the Imaging Science Institute, University Utrecht several time series of dough during proofing (rising) $I(x, y, t) = 256 \times 256 \times 20$ were acquired.

We want to study how different forms of kneading (molding) bread dough affects the crumb structure. With the help of MRI and image analysis we can give experimental evidence for an existing micro-structural chemical model for the rising of dough. The goal is to understand better how the proofing process works and to make machine made bread have the same crumb structure as man made bread.

First we sketch briefly how the rising of dough is thought to be understood in terms of chemistry. For a good readable summary we refer to the text book by Barham chapter 8 [6]. Flour is made of two starch molecules which are polysaccharides (long string-like molecules of sugar molecules). If water is added to the flour, the proteins on the outside of the starch hydrate and become very sticky. If this

[8]The work of this chapter was carried out as part of a contract research for Unilever Research, Vlaardingen in collaboration with the Imaging Science Institute, University Utrecht. The content of this chapter has been published in: P.M. van Duynhoven, G.M.P. van Kempen, R. van Sluis, B. Rieger, P. Weegels, L.J. van Vliet and K. Nicolay, Quantitative Assessment of Gas Cell Development during the Proofing of Dough by Magnetic Resonance Imaging and Image Analysis, *Cereal Chemistry*, 80(4):360-369 2003 [31].

sticky mass is moved along different starch complexes, then via the interaction of the granule proteins *gluten* is formed. Gluten is a protein complex formed out of many starch granules that is organized in layers. It is very elastic like a rubber sheet. During rising CO_2 is produced (by the added or natural abandoned yeast). This gas blows up the gluten sheets like small balloons and rises the bread. Kneading is essential in forming gluten as it stretches the protein molecules. If the dough is not sufficiently kneaded then the gluten sheets are too weak (not enough connections between different starch proteins are made), the little balloons will burst, and the bread will not rise properly. Now, the more you knead the more gluten is formed, the dough in turn will be stiffer, and the gluten sheets will be more isotropically distributed through the dough. A stiffer dough will result mainly in a finer texture and in an on average smaller gas bubble size. Hand kneading will introduce stress in the gluten sheets when the dough is folded. These stress lines are removed by long kneading or machine kneading. To overcome this problem in machine kneading the molding can be done fast, leaving ruptures and internal stress in the gluten sheets. This stress is persevered during rising as we will show.

7.1 Materials and methods

The experiments were conducted at MRI group of the Imaging Science Institute, University Utrecht.

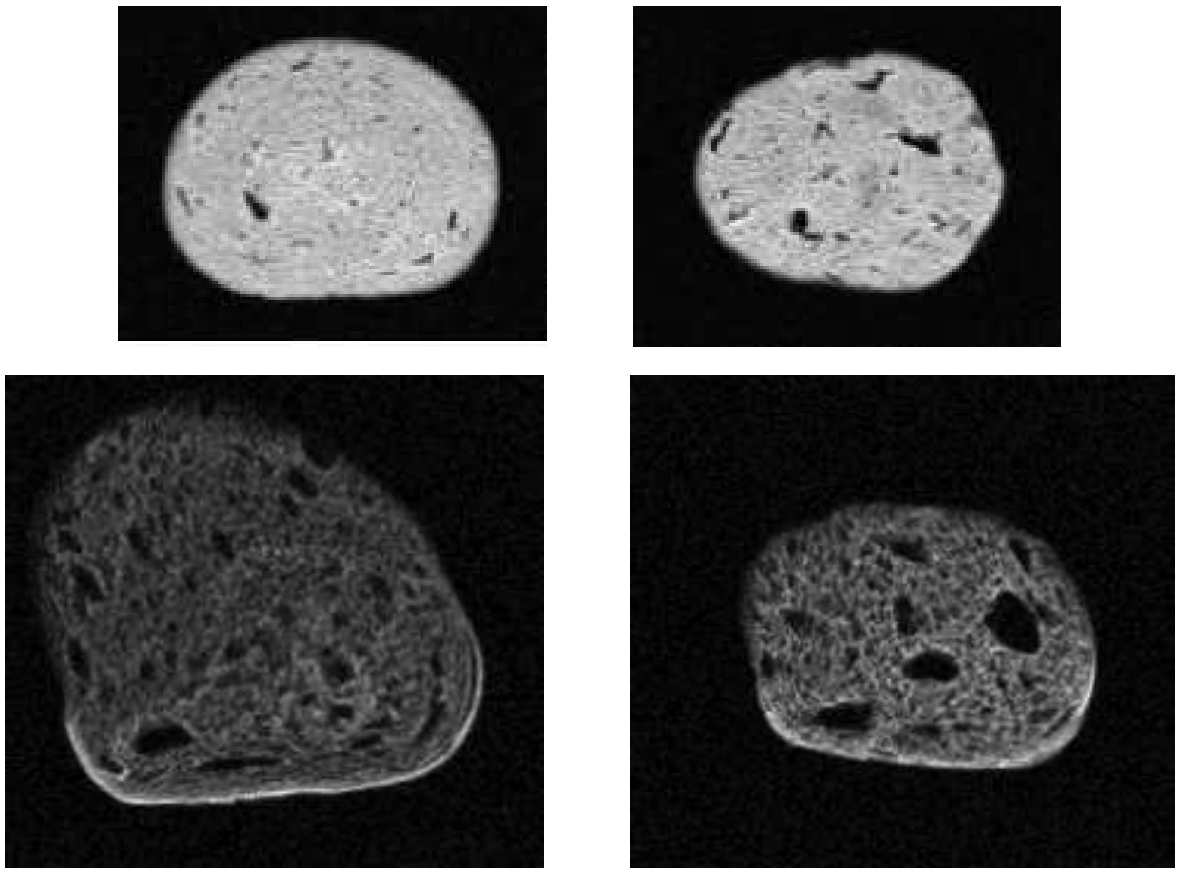
The dough recipe consisted of Baguepi flour with 2.1% salt, 3.2% yeast, 1.75% bread improver (ex Astra Calve) and 54% water (added at 291K). The dough pieces were blast frozen for 35 minutes at 241K, packed in sealed plastic bags, and subsequently stored at 253K, in a temperature controlled freezer. The dough pieces were transported to the MRI facility, after being put on dry ice. After arrival at the MRI facility the frozen dough pieces were again stored at 253K. Within 2 days, the dough pieces were thawed at 277K, during an overnight period. Before the actual measurement, the samples were wrapped in cellophane, and equilibrated at ambient temperature for one hour. The last procedure ensured that the dough piece was at ambient temperature when the MRI measurement started, and prevented condensation of atmospheric water directly on the sample. The dough sample was carefully positioned and fixed on a sample holder using tape.

MRI experiments were performed using a Varian (Palo Alto, CA, USA) Inova spectrometer (operating at a proton frequency of 200MHz), interfaced to an Oxford 40cm bore, 4.7Tesla magnet. A standard spin-echo pulse sequence was used to acquire slices with a thickness of 3.0 mm. Typically, for a field-of-view of 40×40 or 70×70mm, 5 slices were acquired, with 0.1mm spacing, using an echo-time of 9ms and a data matrix of 256×256 points. See figure 7.1 for 2 samples of dough, which were subject to different molding stress. The upper images show the dough

in its initial state and bottom images after proofing (rising).

7.2 Image analysis

A grey-level based image analysis approach was used to characterize the growth behavior during proofing. In a first step the velocity field of the 2D time series is obtained using optic flow as Lucas and Kanade [75], see also eq.(6.36). We compensated for the decreasing intensity in time in the MRI slices. The obtained flow field was then compared with a growth model. Our growth model v_{model} is an isotropic expansion model (see figure 7.2b)), where the growth takes place perpendicular to the contour of the dough, taking the morphology of the dough into account. For this propose we used a heuristic scalar dissimilarity measure dis, which we defined as follows. First we computed for each position in every time



(a) **(b)**

Figure 7.1: Images (70×70mm) of dough pieces, molded with different degree of stress, the top image shows the dough a $t = 0$ and the bottom after baking. a) slow molding/deformation b) fast molding/deformation.

slice the angle between the velocity vector and our model direction weighted by the magnitude of the velocity. These angles are squared and averaged inside a horse shoe like region (see figure 7.2a) and normalized for the size of this area

$$dis(t) = \frac{\sum\limits_{I} \measuredangle^2(v_{model}, v_{meas}) \|v_{meas}\| w(x,t)}{\sum\limits_{I} w(x,t)}\,, \tag{7.1}$$

where the weight function is

$$w(x,t) = \begin{cases} 1 & \text{for} \quad x \in \text{horse shoe} \\ 0 & \qquad\quad \text{otherwise} \end{cases}\,. \tag{7.2}$$

The horse shoe like area is computed for each time step by segmenting the dough and then eroding the binary image 20 times. We keep only the eroded part and there discard the lower third. The reason to evaluate only the upper part of the dough is that in this region the growth restraining influence of the surface on which the dough lies is expected to be small, so deviation from the isotropic expansion can be directly related to the stress contained in the dough during molding. The estimated velocity magnitude from the optical flow acts as a confidence measure for the dissimilarity measure, as it suppresses non reliable angles. To weight strong deviation from the model more heavily we squared the angle between the observed flow field and the model. The computed flow field is shown, for one time frame, in figure 7.2 for samples with slow c) and fast d) molding. The angle of the model with the flow field is shown in false color. As is illustrated in figure 7.3, the dissimilarity measure *dis* is small for fast deformation and larger for slow deformation. Large dissimilarity measures (i.e. a large deviation of isotropic growth) can be attributed to the presence of internal stress in the dough. These stress lines are brought in the dough by the manufacturing procedure, where dough slabs are rolled in a baguette shape. Hence, if stress lines are present in the slab, they will be oriented in a spiral-like manner in the resulting dough sample. In the case that the molding step does not destroy the internal stress lines, the growth of the dough during proofing can locally be restrained by their anisotropic distribution. As a result an anisotropy in the growth will be observed and this is clearly the case for the slowly deformed samples. However, fast deformation has resulted in loss of internal stress, as this results in isotropic growth of the dough. The crumb structure is already defined at an early stage of the bread-making process. Baking is only of minor influence on the crumb structure.

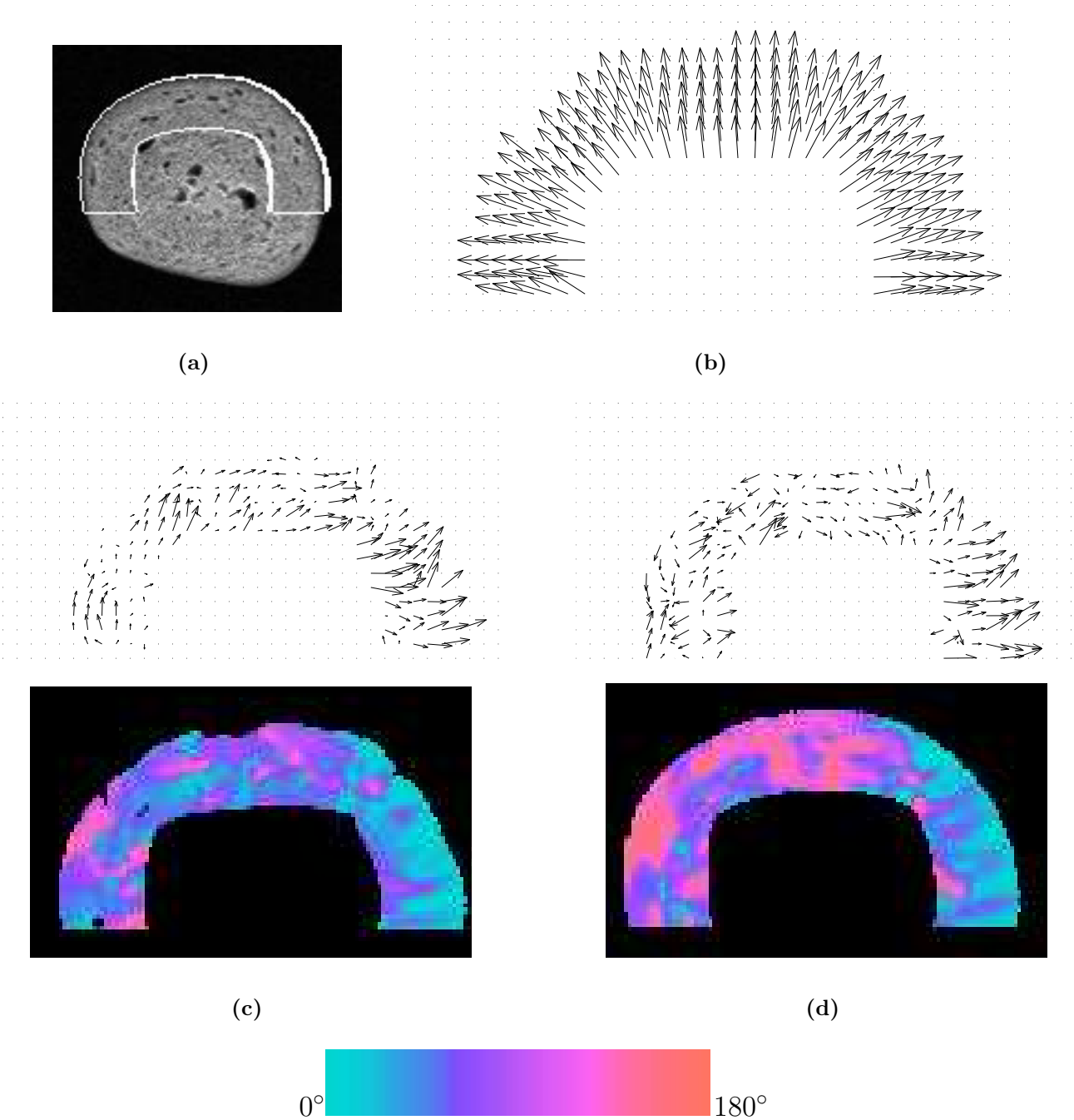


Figure 7.2: a) Upper outer part of the dough being used for the dissimilarity measure, b) Isotropic expansion model perpendicular to the contour, c,d) estimated velocity field (upper image) and angel with the growth model in false color (bottom image), c) slow molding, d) fast molding.

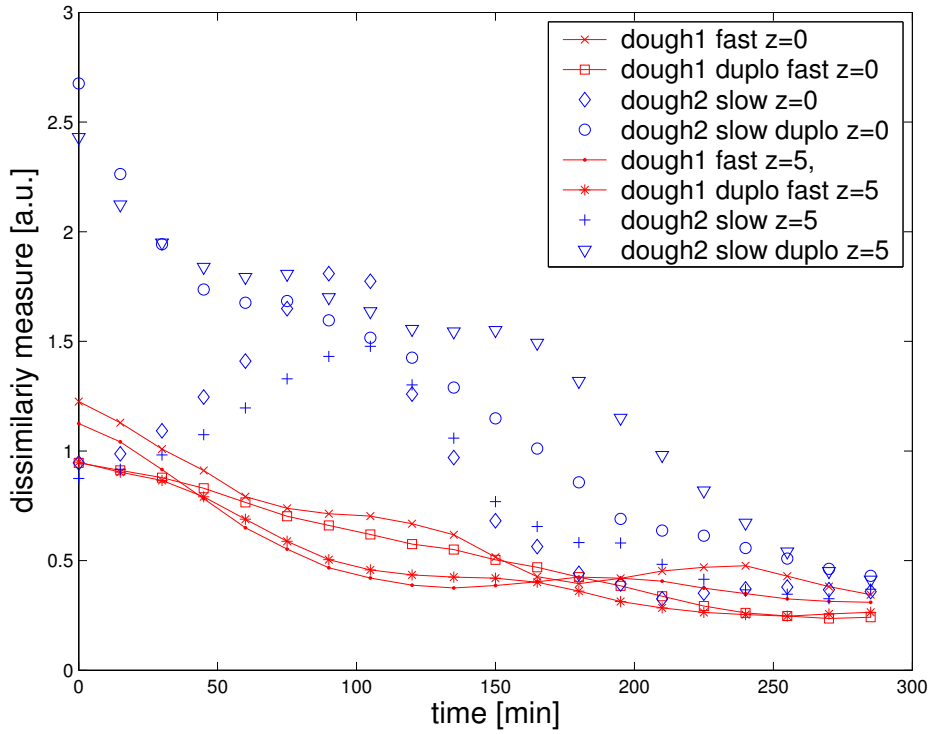


Figure 7.3: Dissimilarity measure as a function of proofing time for dough's with different degrees of imposed molding stress. The non-connected points correspond to slow molding, and the connected curves to fast deformation. The model was applied on 2 duplicate dough samples, and sampled at 2 slices (5mm apart).

7.3 Conclusions

The combined use of MRI and image analysis allows us to draw conclusions on proofing dough and the resulting crumb structure in bread. Our approach shows that it can be successful to combine MRI and image analysis for the quantitative and non-invasive monitoring of the proofing of dough, under relevant industrial conditions. The influence of the molding stress on the rising behavior is made visible with the help of the dissimilarity measure, eq.(7.1). The micro-structural theory of internal stress preservation during proofing (rising) holds in our experiments. Isotropic rising could be attributed to fast molding and anisotropic rising to slow molding. More structural detail could be derived from the MRI data if they would be recorded in 3D mode, since this would improve spatial resolution.

Chapter 8

Application 2: Evaporation induced flow in nanoliter wells[9]

Here we present an image analysis and processing application of adaptive filtering, optic flow and normalized convolution.

Drying of DNA spots on microarrays and spilled coffee yields ring-like stains, because the outward flow transports dissolved particles to the border. Contact line pinning and diffusion limited evaporation of a liquid sample are the two necessary conditions to induce an outward directed liquid flow during evaporation. Evaporation of liquid samples is a key problem in the development of microarray technology (including labs-on-a-chip), especially in the case of open reactors. We have studied the process of evaporation in micromachined subnanoliter wells with a typical radius of $75 - 150\mu$m and a typical depth of 6μm using interference-contrast microscopy [69, 29]. With this microscope-based technique dynamic fringe patterns, showing fringes of equal height, can be observed. Since the depth of the well is only 6μ m, the liquid sample can be regarded as a thin film and the fringe patterns are Newton rings. From these fringe patterns the shape of the air-liquid interface can be retrieved using a temporal phase-unwrapping technique [29]. The rate-of-change of these dynamic height profiles represents the total evaporation rate of the liquid in the subnanoliter well. We have found that the total evaporation rate is linearly proportional to the perimeter of the well [29]. We will present these experimental results in the next section. This counterintuitive result is a consequence of diffusion-limited evaporation. In the next section, we will briefly repeat the most important results of this theory.

The theory of diffusion-limited evaporation predicts an outward directed flow

[9]This work was carried out is close collaboration with L.R. van den Doel who did all the experimental work and the application of Deegan's theory to the nanoliter wells. The content has been published in: B. Rieger, L.R. van den Doel and L.J. van Vliet, Ring Formation in Nanoliter Cups: Quantitative Measurements of Flow in Micromachined Wells, *Physical Review E*, 68(3):036312, 2003 [100]. New material is presented in section 8.6.

in a droplet, which transports dissolved particles to the perimeter of the droplet, a phenomenon responsible for ring-like stains on flat surfaces: this is the explanation for the formation of coffee rings [25, 26]. In macroscopic droplets this evaporation induced flow field has been coarsely sampled by tracking microspheres only near the edge of the droplet and only at the end of the evaporation process [26]. On the basis of this theory we expected a similar outward directed liquid flow in our wells, but this flow could not be observed with interference-contrast microscopy. In order to visualize this flow field and to validate the theory of ring formation, we conducted experiments in which subnanoliter wells were filled with a liquid in which fluorescent microspheres were introduced. The motion, including Brownian motion, of the microspheres (radius 0.25μ m) during evaporation of the liquid sample was monitored using digital fluorescence microscopy. The acquired digital recordings (image sequences) clearly showed the transportation of the microspheres to the perimeter of the well. Figure 8.1a) shows the distribution of the fluorescent microspheres right after injection of the mixture in the well. Figure 8.1b) shows the trajectories of the dissolved microspheres during evaporation. See also the cover of this thesis for a color version of figure 8.1b). There the time is color coded from blue for the inital stage to red for the end of the evaporation. All trajectories point outwards, when observed over a long time ($\approx$ 3 min.). When observed on a finer time-scale ($\approx$ s), the random behavior due to Brownian motion is clearly visible, but only in the early stage of the evaporation process. Figure 8.1c) shows the microspheres almost at the end of the evaporation process; practically all microspheres have been transported to the edge of the well.

First, we summarize the most important aspects of diffusion-limited evaporation regarding droplets. We will argue that these aspects are applicable to our nanoliter wells. Second, we will discuss in detail the effects of the liquid flow on the microspheres. Third, we will present a digital image analysis approach, using optic flow, to measure the flow field. As the well contains a limited number ($\approx$ 180) of microspheres, we obtain a sparse spatio-temporal measurement space. The local velocity (flow field) and a confidence value is computed from this measurement space. The confidence is high at (x, y, t)-positions where a sphere is present and low elsewhere. The confidence map allows for a transformation of the flow field $v(x, y, t)$ into an estimate of the radial velocity $\hat{v}(r, t)$ as a function of the radial position r and time t. Finally, we will compare the measured flow field with the predicted flow field.

8.1 Diffusion-limited evaporation

Recently, the formation of ring stains in "macroscopic" droplets, e.g. coffee droplets, was explained by Deegan et al. as the result of contact line pinning at the edge of the droplet, and evaporation-limited diffusion of molecules from the saturated

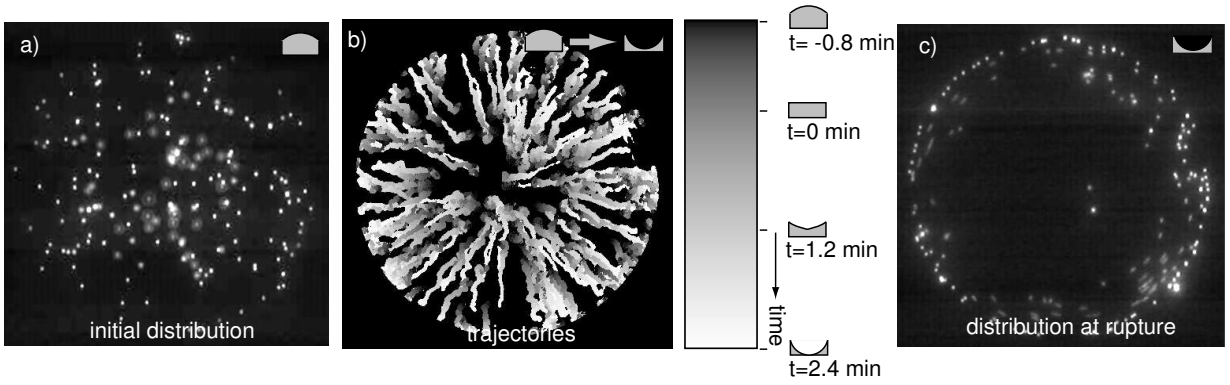


Figure 8.1: a) The distribution of the microspheres (radius 0.25μ m) directly after filling the circularly shaped well (radius 100μ m) with an evaporating ethylene-glycol sample, b) the trajectories of the fluorescent beads (see also the cover of this thesis for a color version). The grey-scale map represents time, c) the distribution of the microspheres just before the meniscus (air-liquid interface) touches the ground in the middle of the well and breaks. The instant $t = 0$ corresponds to the moment that the meniscus is perfectly flat. The corresponding meniscus shape is illustrated next to the grey-scale-map.

vapor layer to the environment [26, 25]. These two conditions are responsible for the outward directed flow and explain our observations of the transportation of the fluorescent microspheres as shown in figure 8.1.

Deegan et al. state that the rate limiting step of the evaporation process determines the functional form of the local evaporation rate $j(r)$ along the air-liquid interface. If the evaporation process is *diffusion* limited, then the diffusion process will rapidly attain a steady state: the diffusion equation reduces to the Laplace equation. This results in a boundary value problem with boundary conditions $u = u_s$ (saturated vapor pressure) at the interface and $u = u_\infty$ (ambient vapor pressure) far away from the interface, which can be solved analytically, under the assumption that the shape of the droplet is a spherical cap. Deegan verified experimentally that a spherical cap describes the shape of the air-liquid interface of a droplet [26]. If the footprint of the droplet has a constant radius R and the height in the center of the droplet is $H(t)$ as a function of the time t, then it is straightforward to derive that the height $h(r,t)$ of the interface at a distance r from the center of the droplet is given by:

$$h(r,t) = \frac{H^2(t) - R^2 + \sqrt{H^4(t) - 4H^2(t)r^2 + 2H^2(t)R^2 + R^4}}{2H(t)}\,. \tag{8.1}$$

Deegan et al. give a very good approximation for the analytical solution of the boundary value problem given the geometry of the droplet defined in eq. (8.1):

$$j(r) \propto \frac{1}{\sqrt{1 - \left(\frac{r}{R}\right)^2}}\,. \tag{8.2}$$

Equation (8.2) shows that the evaporation rate is strongly enhanced towards the edge of the droplet. As a result, the loss of liquid at the edge due to evaporation must be replenished by liquid from the bulk of the droplet. This induces an outward directed liquid flow $v(r,t)$. Given the expressions for the dynamic height of the air-liquid interface of the droplet $h(r,t)$ (eq.(8.1)), the (static) local evaporation rate $j(r)$ (eq.(8.2)), and the rate of change of the height in the center of the droplet, the outward flow field $v(r,t)$ as a function of the radial position r and time t follows from a mass balance [26]:

$$v(r,t) = -\frac{1}{\rho r h}\int_0^r dr'\, r' \left\{ j(r')\sqrt{1+\left(\frac{\partial h}{\partial r'}\right)^2} + \rho\frac{\partial h}{\partial t}\right\}, \tag{8.3}$$

with ρ the liquid density. From this equation it is clear that a nonzero v arises when there is a mismatch between the local evaporation rate $j(r)$ and the rate of change of the height profile $h(r,t)$. There are three differences between our experiments and the experiments of Deegan et al.

First, the geometry of the liquid sample in our circular wells differs from the geometry of a droplet on a flat surface. The shape of the air-liquid interface is measured using interference-contrast microscopy [69, 29]. With this microscope-based technique, fringe patterns showing Newton rings are generated. The depth of the wells is only 6μ m and the liquid sample in the well can therefore be regarded as a thin film. Analysis of the fringe patterns acquired by interference-contrast microscopy results in the height profile of the meniscus as a function of time [29]. Figure 8.2 shows nine (out of a large series) retrieved height profiles (solid lines) of the air-liquid interface as a function of the radial position. The time difference between two successive profiles is $20s$. The dashed lines show the height profiles computed from eq.(8.1) with $R = 100\mu m$ and the best fit to the center

$$H(t) = (7.38\mu\text{m} - 6.13\mu\text{m}) - 0.042\frac{\mu\text{m}}{\text{s}}\,t\,. \tag{8.4}$$

The depth of the well (estimated value $6.13\mu m$) is added to $h(r,t)$ to compute the height of the air-liquid interface with respect to the bottom of the well. The expression for $H(t)$ follows from fitting a first-order polynomial to the measured heights of the air-liquid interface in the center of the well. Figure 8.2 confirms that the meniscus can be modeled by spherical caps. In other words: eq.(8.1) is valid for the air-liquid interface of the liquid sample in our wells.

Second, we have used ethylene glycol for our experiments rather than water. The evaporation process is extended to a time scale on the order of a few minutes instead of a few seconds. With the technique of interference-contrast microscopy we have found that the total evaporation rate for a circular well is constant in time [29]. Furthermore, any thermal effect, cooling of the liquid due to evaporation,

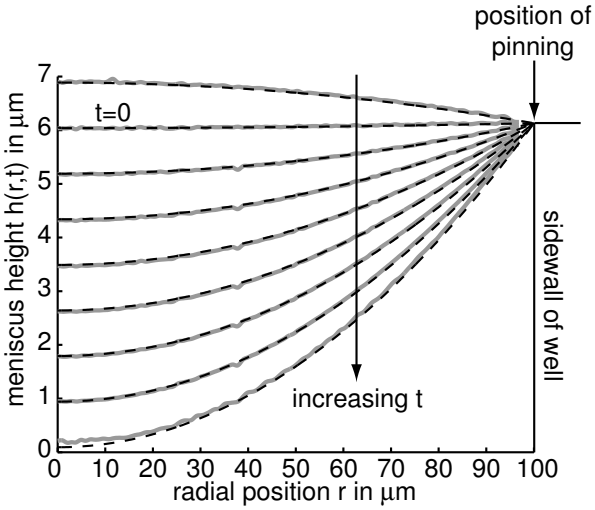


Figure 8.2: The measured height profiles of the meniscus (solid lines) and height profiles representing spherical caps (dashed lines) obtained by fitting as a function of evaporation time.

is negligible. The large silicon volume in which the well is fabricated is a source of thermal energy that can easily compensate for any loss of thermal energy due to evaporation, especially since the evaporation is such a slow process. Furthermore, the total evaporation rate as a function of the well radius R is a linear function, which implies that the evaporation process in our wells is indeed diffusion limited.

Third, at the start of our experiment, the meniscus is convex and the geometry of the meniscus is identical to the geometry of a droplet on a flat surface. The fact that we have a cylindrical liquid volume with a radius R and a depth of 6μm is not important for the boundary value problem to be solved. No molecules can escape via the bottom or via the side walls of the well, just as no molecules can escape through the footprint of a droplet on the surface. At the start of our experiment, we have the same geometry as Deegan et al., the same type of diffusion process, the same boundary value problem and therefore the same form of the local evaporation rate. In other words, at the start of our experiment eq.(8.2) is valid. Furthermore, since a diffusion-limited evaporation process implies a static local evaporation rate [26] ($j(r,t) = j(r)$), we conclude that this form is valid during the entire evaporation process. Considering the three differences described above, we can still use the expressions for the dynamic height of the air-liquid interface of the droplet $h(r,t)$ (eq.(8.1)), the (static) local evaporation rate $j(r)$ (eq.(8.2)), and the rate of change of the height in the center of the well (eq.(8.4)). The outward flow field $v(r,t)$ as a function of the radial position r and time t follows from eq.(8.3).

Figure 8.3a) shows the flow $v(r,t)$ for a circular well (radius $100\mu m$, depth $6.13\mu m$) according to eq.(8.3). As can be seen in this figure, the flow in the initial phase of the evaporation is very slow ($\approx 0.1\mu m/s$). At the end of the evaporation

process the velocity of the flow increases ($\approx 1\mu$m/s, in figure 8.3a) the velocity is clipped at 0.5μm/s for good visualization). Due to symmetry the velocity in the center of the well is zero at all instants. Note that the theory predicts that the flow at the edge of the well is not zero. Furthermore, the form of the local evaporation rate in our wells is identical to the form in the case of droplets. The flow field in our wells, however, is not identical to the flow field in a droplet. The difference is the 6μm deep cylindrical volume in our wells. This keeps the flow field at the edge of the well at some finite value, whereas in the case of a droplet, the flow field at the edge of the droplet diverges (mathematically). Figure 8.2 suggests that the change of the shape of the air-liquid interface is significant. This is not the case. The minimum radius of curvature of the air-liquid interface, at the instant that the meniscus hits the bottom of the well, is approximately $1.1cm$. Compared to the radius of the well, $R = 150\mu$m, the meniscus is practically flat during the complete process of evaporation.

8.2 Motion of microspheres in the fluid

In this section, we will discuss if the microspheres are well suited to monitor the flow. The first question is: are the spheres small enough, not to influence the flow and to represent it accurately? Secondly, are they big enough, such that Brownian motion is not a major effect? Finally, is the diameter of the spheres small enough compared to the depth of the well?

1. The outward directed flow is laminar, since the Reynolds number $Re = \frac{\rho v l}{\eta} \ll 1$, with l a characteristic length, i.e. the radius of the microspheres, $v \approx 1\mu$m/s the velocity, $\eta = 16.1 \cdot 10^{-3}$Pas the viscosity of ethylene glycol ($C_2H_4(OH)_2$) at 298K [72] and $\rho = 1.1 \cdot 10^3$kg/m^3 the density of ethylene glycol. Stokes's law gives the friction force on a microsphere as

$$F = 6\pi r \eta (v - v_k), \tag{8.5}$$

with v the velocity of the fluid and v_k the velocity of the sphere. From this follows the acceleration of the sphere as $\dot{v}_k = C(v - v_k)$ with $C = \frac{9\eta}{2\rho_k r_k^2}$, with $\rho_k = 905$kg/m^3 the density of the microspheres (polypropylene). The velocity of the fluid changes very slowly in time. Consequently, we assume v to be constant and obtain

$$v_k(t) = v(1 - e^{-Ct}), \quad C = 8 \cdot 10^{-5} \cdot \frac{1}{r_k^2} \frac{\text{m}^2}{\text{s}} \tag{8.6}$$

as the solution of the equation of motion with boundary condition $v_k(0) = 0$. For $r_k = 0.25\mu$m we conclude that $v_k \to v$ instantaneously. Whereas if the

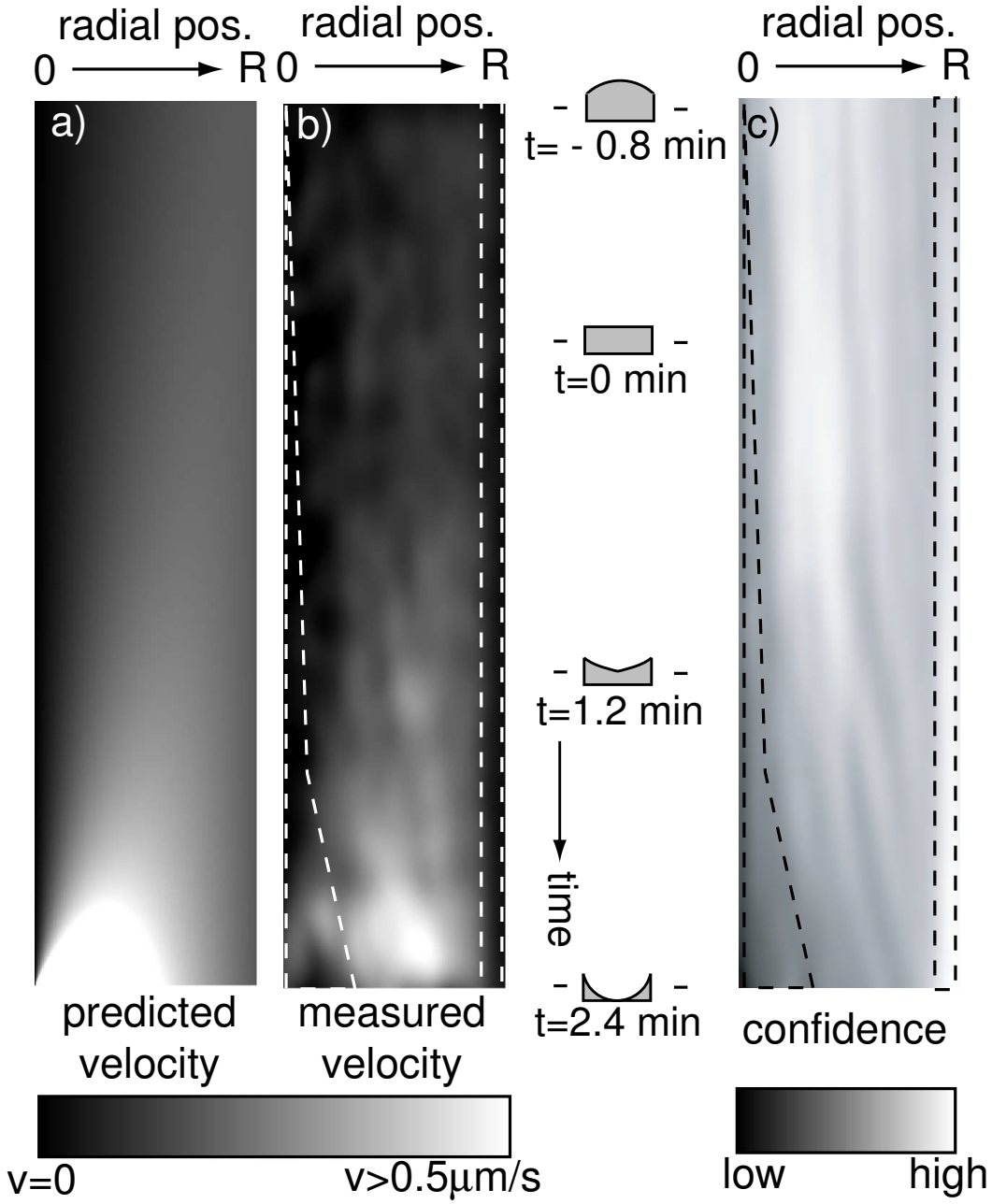


Figure 8.3: a) The predicted outward-directed flow field $v(r,t)$ as a function of time t and radial position r, b) the measured outward-directed flow field $\hat{v}(r,t)$, c) the radial confidence field $\hat{c}(r,t)$. The instant $t = 0$ corresponds to the moment that the meniscus is perfectly flat. In the dashed region for small radial positions the confidence is very low, because of the absence of microspheres. In the dashed region for large radial positions the microspheres at rest do not represent the flow.

constant C were on the order of unity (for $r_k \approx 5$mm), the time to reach $v_k = 0.95v$ starting from initial rest would then be ≈ 5s.

2. The average quadratic displacement in three dimensions of a microsphere due to Brownian motion in a static solution in a time-interval t is given by $\langle x^2 \rangle = 6Dt$, with D the diffusion constant defined by the Einstein relation [32]

$$D = \frac{k_B T}{6\pi\eta r_k}\,. \tag{6.48}$$

In our experimental setup we only observe the projected 2D displacement $\sqrt{\langle x_{ob}^2 \rangle}$ rather than the real 3D displacement $\sqrt{\langle x^2 \rangle}$. Their relation is given by the stereological factor, the average projected displacement of all 3D displacements

$$\sqrt{\langle x_{ob}^2 \rangle} = \frac{\int_0^1 \sqrt{1-z^2}\,dz}{\int_0^1 dz}\sqrt{\langle x^2 \rangle} = \frac{\pi}{4}\sqrt{\langle x^2 \rangle}\,. \tag{8.7}$$

Finally, we obtain $\langle x_{ob}^2 \rangle = \frac{3}{8}\pi^2 Dt \approx 3.7Dt$. Recall that $\langle x^2 \rangle = 4Dt$ for 2D Brownian motion. With the values for η and r_k, we have $D = 0.054\mu\text{m}^2/\text{s}$. In the early stage of the evaporation $v \approx 0.1\mu$m/s and we conclude that Brownian motion is a significant effect, but Brownian motion becomes negligible towards the end of the evaporation process with $v > 1\mu m/s$.

3. The choice for microspheres with a radius of 0.25μ m seems reasonable given the depth of our wells (6μm). The particles are small enough that they immediately drift with the flow. Smaller particles would be more influenced by Brownian motion. Larger particles would reduce the contribution of Brownian motion, but larger particles would strand at the bottom of the well early in the evaporation process when the fluid level in the middle of the well decreases.

8.3 Imaging microspheres in the fluid

In the previous section we argued that the microspheres were suited to visualize the flow from a transportation point of view. In this section we will discuss if the microspheres can be imaged properly onto a detector and recorded, such that the analysis of the acquired image sequence results in a good description of the liquid flow. Figure 8.1a) clearly shows that not all microspheres are in focus. Right below the center of the well some out-of-focus microspheres can be seen; instead of a small bright spot, a bright and dark ring can be seen around a dimmer spot. In microscopy the following approximation is used to describe the depth-of-focus

(DOF) for a quarter of a wavelength defocus error [149]:

$$DOF = \frac{\lambda}{4n\left(1 - \sqrt{1 - \frac{NA^2}{n^2}}\right)}\,. \tag{8.8}$$

with $\lambda = 605$nm the emission wavelength of the Rhodamine-stained microspheres, $n = 1.43$ the refractive index of ethylene glycol and $NA = 0.75$ the numerical aperture of the objective used for microscopic imaging. This is the maximum tolerance of defocusing without significant distortion of the image (which is in fact the $\lambda/4$ criterion for the wave front distortion of the marginal ray of the lens). Inserting these values in eq.(8.8) yields a total depth-of-focus (defocusing error $|\Delta z| < 5/4\lambda$) of $1.5\mu m$. If we position the focal plane at half the depth of our wells, then the 6μ m depth of the wells implies a maximum defocusing error of one wavelength disregarding evaporation. The out-of-focus microspheres in figure 8.1a) may very well have this defocusing error. In our experiments, we have positioned the focal plane below half the depth of the wells, since the height of the liquid decreases during the evaporation process. This implies that during the evaporation process the out-of-focus microspheres tend to move in focus. This can be seen in figure 8.1c), where practically all microspheres are sharply imaged. The unsharpness here is due to very fast motion at the instant that the meniscus hits the bottom of the well. Equation 8.8 reduces for $NA/n \ll 1$ to the more well-know expression

$$DOF = \frac{\lambda n}{4NA^2}\,. \tag{8.9}$$

Any analysis that requires sharply imaged objects, such as segmentation and tracking of objects, as used by Deegan et al. [26], can only be applied in a short time interval and for small changes in the height of the air-liquid interface. The technique we propose takes these intensity fluctuations due to defocusing into account. Consequently, we can measure the flow field during the complete course of evaporation and at all radial positions.

8.4 Image analysis

In this section we will present a grey-level based image analysis approach to estimate the average radial velocity of the microspheres in the fluid. This approach does neither require segmentation nor tracking of the individual microspheres.

One option to measure the flow is standard particle image velocimetry (PIV) [142]. PIV is used for monitoring and analyzing laminar and turbulent flow behavior mainly in pipes, based on the cross correlation of subsequent images. PIV assumes that *many* tracer particles are *uniformly* distributed in the fluid. Computation of

the local shift between two successive images requires at least one particle to be present in the interrogation window. For an image with a size of about $218 \times 218 \mu m^2$ (figure 8.1a) and about 180 visible dissolved microspheres, the correlation window must be at least $16 \times 16 \mu m^2$. Note that the estimation of the correlation peak from one marker is not very robust. Furthermore, the flow is not expected to be uniform in larger windows, but radially diverging from the center of the well. Therefore we conclude that we do not satisfy the assumptions necessary for robust velocity estimation via PIV. The use of more markers in this micron scale flow study results in indistinguishable objects after imaging. Smaller microspheres do not emit sufficient light for a reasonable signal to noise ratio. The average shift of a microsphere between two successive images in our experiments is small (0.01–0.07 pixel units) and PIV with sub-pixel interpolation can only obtain measurements with an uncertainty of 0.1 pixel units [145]. Another possibility is to leave out several images in the sequence and compute the cross-correlation between them. This is not necessary in our approach, therefore we can use all available images for improving the signal-to-noise ratio allowing high precision velocity estimation.

A second option to measure the flow field is segmentation of the microspheres and tracking over the course of the fluid evaporation. This is not very promising. As discussed earlier we see all microspheres in a projection. Not all microspheres are in the focal plane, therefore some of the microspheres appear blurred. This makes segmentation a difficult task, that can be avoided by our grey-value approach.

The technique we propose uses optic flow [75], compare chapter 6.1.2 for a detailed discussion on different methods to compute velocity in images. Optic flow computes the velocity per pixel for the whole image sequences $I(x, y, t)$. The trajectories of the microspheres form spaghetti-like structures in $I(x, y, t)$, see figure 8.4. Where PIV uses only two successive images to compute the shift of the tracers, optic flow uses more images. Velocity is interpreted as orientation in the spatio-temporal image sequence. The use of several images simultaneously yields a higher precision and accuracy for the velocity estimate. Optic flow starts from the assumption that the intensity per object $I(x, t)$ is conserved, such that

$$I(x, t) = I(x + dx, t + dt)\,. \tag{8.10}$$

Assuming a uniform rectilinear motion $v = const.$ we can use a first order Taylor expansion of the right hand side and combine it with eq.(8.10) This yields the well-known flow equation

$$\nabla I \cdot v + \partial_t I = 0\,. \tag{6.27}$$

This is a scalar equation, which does not allow the computation of all components of v. This is known as the aperture problem. Expanding the constraint of constant velocity to a small neighborhood $V = (\Delta x, \Delta t)$, introduced first by Lucas and

Kanade [75], we can solve eq.(6.27) for v using a least squares approach:

$$v(x,y,t) = -\frac{1}{V^2}\int dV(\nabla I \nabla I^t)^{-1}\int dV \nabla I \partial_t I\,. \tag{6.36}$$

The images are recorded with a CCD camera and processed with a computer, thus we deal with discrete signals rather than continuous functions. Equation (6.36) therefore requires discrete derivatives. Finite differences of image values divided by the pixel distance is not the ideal derivative for images [19, 83]. Gaussian derivative filters (or at least approximations) are used ordinarily in image processing as they combine noise suppression and localization in an optimal way [19, 83]. A discrete derivative is computed with the help of a regularized derivative, as in distribution theory, i.e. using a convolution of the image with derivatives of Gaussians characterized by their standard deviation σ. The parameter or scale σ defines the smoothing. A larger σ results in a higher noise suppression and blurring. The effective width of a Gaussian filter is $(-3\sigma, 3\sigma)$. Equation (6.36) involves three scales: the spatial derivatives $\nabla(\sigma_g)$, the temporal derivative $\partial_t(\sigma_t)$, and the averaging in $V(\sigma_T)$. We have set $\sigma_g = 0.52\mu$m, $\sigma_t = 0.2$s and $\sigma_T = (2.6\mu\text{m}, 0.3\text{s})$. The reason for these choices is as follows: σ_g should be small enough to detect abrupt changes in intensity transitions from the background intensity level to the object level. We want a standard deviation of $\sigma_t = 0.2$s to average over several (≈ 18) successive images. The regularization kernel σ_T has to be large since the structure is estimated in a spatio-temporal neighborhood defined by σ_T. Our velocity estimation is not critically dependent on the choice of the scale parameters, i.e. no fine tuning is needed. A range of values will only lead to very small changes in the estimated velocity.

Given the lowering of the meniscus during the evaporation, the changes of the focal plane, and the Brownian motion in the axial direction, the microspheres drift in and out of focus. This causes a varying recorded intensity per microsphere, which is not compatible with the assumption of optic flow. An adaptive 1D filter along the trajectories is applied before computing the optic flow to suppress noise and to reduce the intensity variations without spurious blurring of the image [40, 4]. Figure 8.4 shows for some trajectories in a quarter of the well after pre-processing. Adaptive filtering requires knowledge about the local structure in the image in order to adapt to it. The orientations of the trajectories of the microspheres are estimated by an eigenvalue analysis of the gradient structure tensor [66, 45, 61]. This is a generic tool to estimate the local structure and the dimensionality of the structure in images. The gradient structure tensor is closely related to the inertia tensor from classical mechanics. For line-like structures the eigenvector associated with the smallest eigenvalue corresponds to the orientation of the trajectory and the eigenvalues are distributed as $\lambda_1 \approx \lambda_2 \gg \lambda_3$. Furthermore, this permits the

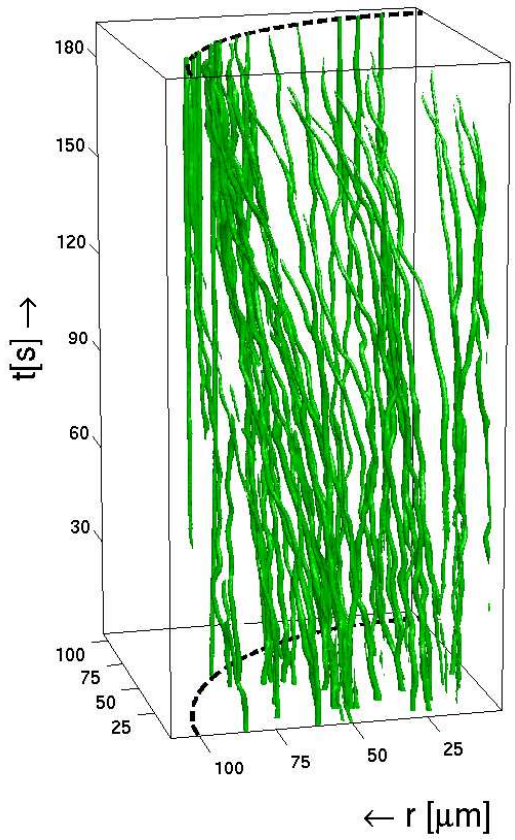


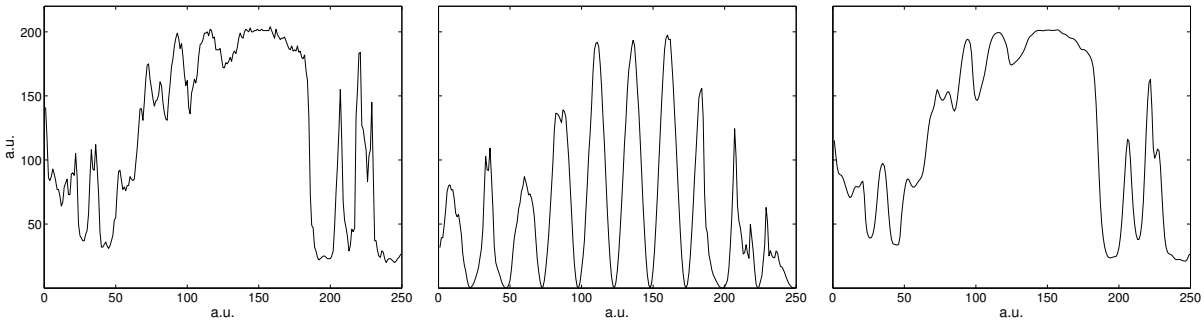
Figure 8.4: Iso-surface plot of the trajectories of microspheres in a quarter of the well after 1D adaptive filtering. The time axis points upwards and we only show every 7^{th} time frame of the (x, y, t) image.

computation of a robust normalized confidence field

$$c(x, y, t) = \frac{\lambda_2 - \lambda_3}{\lambda_2 + \lambda_3}, \tag{8.11}$$

where $c \approx 1$ for lines in the spatio-temporal image and zero elsewhere.

After adaptive filtering and optic flow estimation, the radial velocity needs to be estimated against fluctuations of individual particle movements. The limited number of particles result in a (x, y, t)-image that is sparsely filled with trajectories. As a consequence, the velocity field $v(x, y, t)$ only contains reliable data along/near the trajectories, i.e. (x, y, t)-locations with high confidence. This is especially the case in the center of the well near the end of the evaporation process, since most of the microspheres that were present in the center at the start of the experiment have been displaced to the border of the well by the liquid flow. In regions with low confidence (few or no microspheres present) the velocity can be estimated by normalized convolution [66, 144, 45], which interpolates the signal between parts of high confidence. See figure 8.5 for an example of the use of normalized convolution for interpolation purposes.



(a) **(b)** **(c)**

Figure 8.5: Demonstration of interpolation using normalized convolution, a) original signal s, b) signal confidence product $s \cdot c$ with some confidence $c \in [0, 1]$, c) reconstructed signal $\frac{(s \cdot c) * w}{c * w}$ with a Gaussian weight function w where $*$ is the convolution operator.

The circular shape of our well justifies angular averaging of the radial components of the velocities. Given the velocity field $v(x, y, t)$, averaging over all measurements in a ring between radius r and radius $r + dr$ results in a more robust estimate of $v(r, t)$ and interpolation between measurements with high confidence compensates for the sparse field. The averaging and interpolation are performed simultaneously. This requires the radial component of the velocity field to be weighted with the confidence field, or $v_r(x, y, t) = (v(x, y, t) \cdot \hat{r})c(x, y, t)$ with $\hat{r} = \frac{r}{|r|}$. The combination of radial averaging and spatio-temporal interpolation on measurements with high confidence uses the weight function $w(x, y, t; r)$, which is a ring in the xy-plane that has a Gaussian profile, as center point the center of the well (x_0, y_0, t), and radius r:

$$w(x, y, t; r) = e^{-\frac{1}{2\sigma_{xy}^2}\left(\sqrt{(x-x_0)^2+(y-y_0)^2}-r\right)^2} e^{-\frac{t^2}{2\sigma_t^2}}, \tag{8.12}$$

where the temporal blurring is chosen to be much larger (in image pixels) than the spatial blurring as the predicted velocity is very smooth along the time-dimension, i.e. $\sigma_t \gg \sigma_{xy}$. This weight function is applied to the velocity field as well as to the confidence field. The ratio of the two weighted fields gives an estimation for the flow field at radius r:

$$\hat{v}(r, t) = \frac{\sum_{x,y,\tau} v_r(x, y, \tau) w(x, y, t - \tau; r)}{\sum_{x,y,\tau} c(x, y, \tau) w(x, y, t - \tau; r)}. \tag{8.13}$$

The operation with the weight function in eq.(8.13) is a scalar product in space and a convolution in time. As the time and spatial components are separable and the

ordering of summation can be interchanged we can rewrite eq.(8.13) to

$$\hat{v}(r,t) = \frac{\sum_{\tau} w_2(t-\tau) \sum_{x,y} v_r(x,y,\tau) w_1(x,y;r)}{\sum_{\tau} w_2(t-\tau) \sum_{x,y} c(x,y,\tau) w_1(x,y;r)}\,, \tag{8.14}$$

which allows for a convenient computational implementation. The radial confidence field follows as

$$\hat{c}(r,t) = \frac{\sum_{x,y,\tau} c(x,y,\tau) w(x,y,t-\tau;r)}{\sum_{x,y,\tau} w(x,y,t-\tau;r)}\,. \tag{8.15}$$

Figure 8.3b) shows the computed radial velocity distribution $\hat{v}(r,t)$. Figure 8.3c) represents the confidence $\hat{c}(r,t)$. In order to compare the predicted flow field with the computed flow field, we plot them for different radii in a well with a radius of 100μm as shown in figure 8.6. Figure 8.7 shows the same comparison for a well with a radius of 150μm.

The estimated velocity $\hat{v}$ corresponds well with the predicted velocity, given the fact that the estimate is based on sparse measurements. The microspheres move towards larger radii during the evaporation and make the center of the well empty. This empty region grows during the evaporation process. Thus, for small radii, it is practically impossible to measure the velocity for $t \gg 0$. This explains the region with low confidence in the lower left corner (small radial position at the end of the experiment) in figure 8.3c). For large radial positions, at the sidewall of the well, the spheres will collide against the wall, stop moving, and no longer represent the flow.

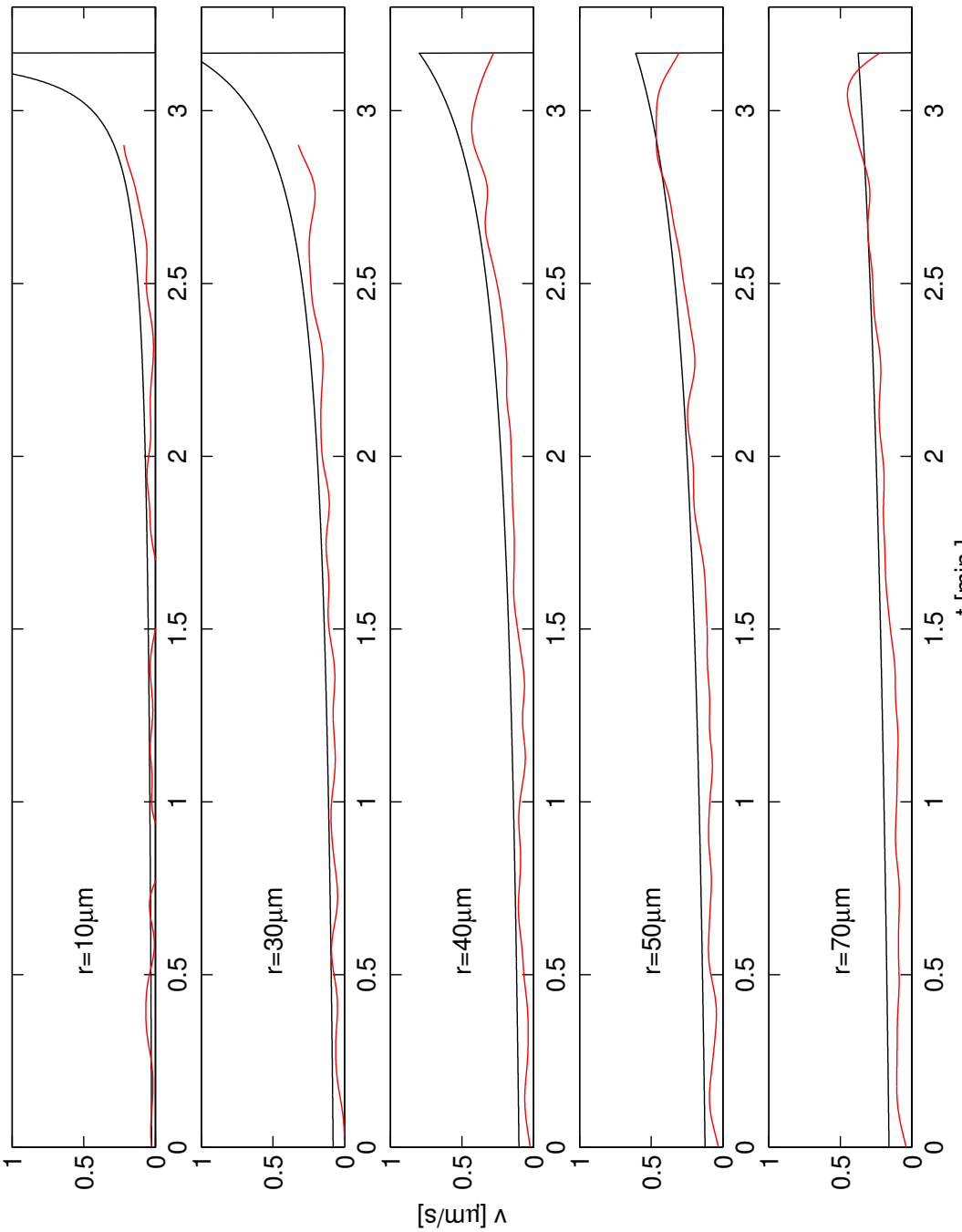


Figure 8.6: Theoretical and estimated velocity $\hat{v}$ for different radii as a function of time. The radius of the well is 100μm.

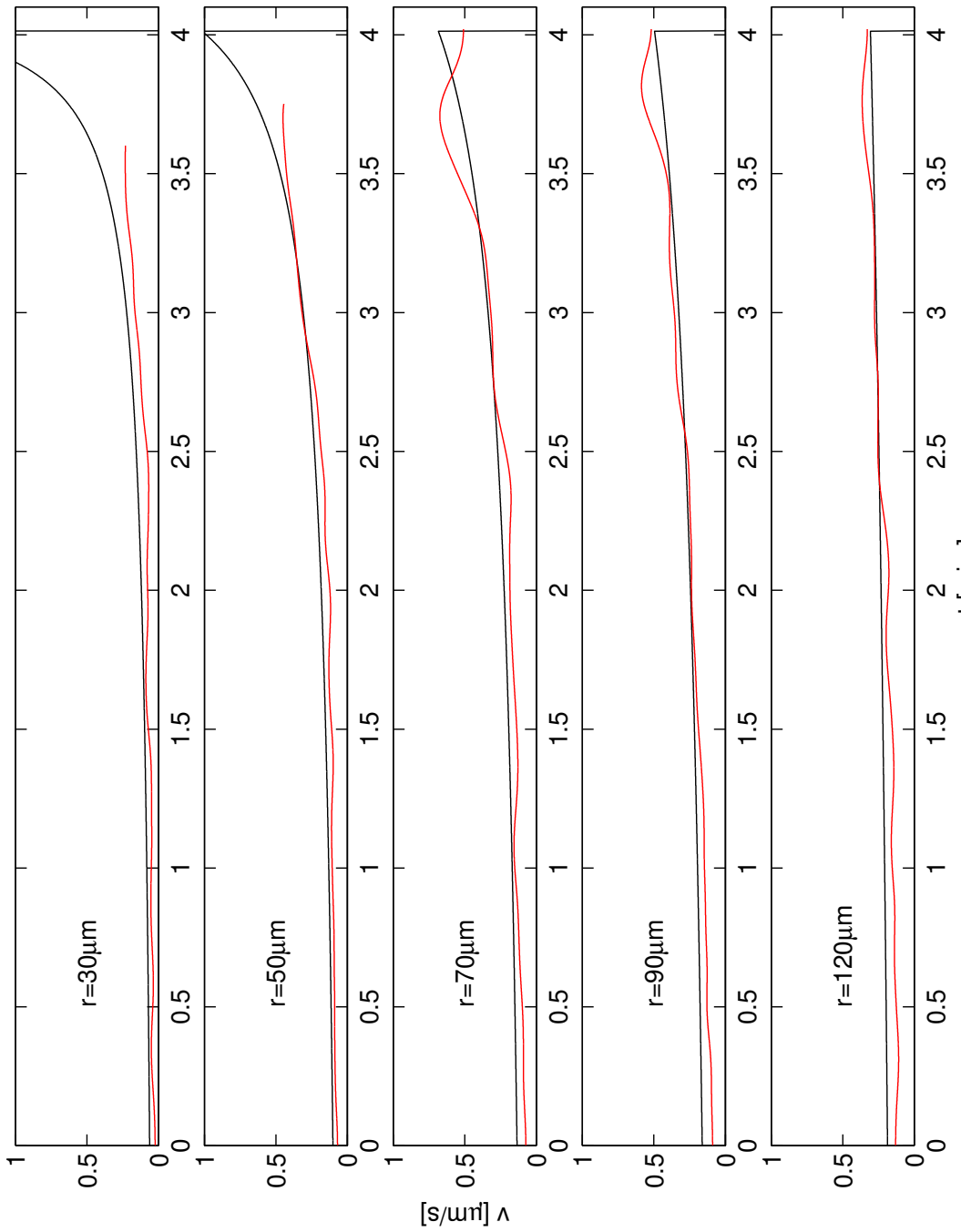


Figure 8.7: Theoretical and estimated velocity $\hat{v}$ for different radii as a function of time. The radius of the well is 150μm.

8.5 Conclusions

In this chapter we presented a method to measure the flow induced by an evaporating liquid sample in circular wells ($R = 100\mu$m, 150μm, depth $= 6\mu$m) and compare the estimated flow field with the predicted flow field. The outward directed flow is a consequence of the pinning of the liquid at the edge of the well in combination with the diffusion-limited evaporation. Small ($r_k = 0.25\mu$m) microspheres are used to visualize the flow. The Brownian motion of the microspheres is only of influence in the very early stages of the evaporation when the flow velocity is small. We use optic flow to process the acquired image sequence and retrieve an estimate of the flow field (velocity image). Together with a confidence measure we are able to interpolate the relatively sparse measurement points in a generic manner. The final estimate of the radial velocity $\hat{v}(r,t)$ is in good agreement with the theoretical prediction, except at the side walls of the well.

Previous measurements of flow in droplets to support the theory have been limited to a small number of points in the lower right region of figure 8.3b). The novel approach described here, which allows for a detailed quantitative verification of the elegant theory of ring formation in pinned droplets (and wells), significantly outperforms the earlier presented results, which has been coarsely sampled by tracking microspheres only near the edge of macroscopic droplets and only at the end of the evaporation process [26].

Methods and materials

A small volume of Rhodamine-stained microspheres (Fluospheres, Molecular Probes, Leiden, The Netherlands, excitation wavelength: 580nm and emission wavelength 605nm) with a radius of 0.25μm was added to a droplet of pure ethylene glycol. The experiments were performed on a Zeiss Axioskop microscope with a $20\times/0.75$ FLUAR objective and a fluorescence filter set from Zeiss (filter set #15, excitation: BP 546/12, beam splitter: FT 580, emission: LP 590). A Sony DXC-960MD 3CCD color video camera was mounted on the microscope via a $2.5\times$ camera mount from Zeiss. The CCD camera was attached to a Matrox Marval G200 AGP frame grabber. Digital recordings were acquired at a frame rate of $15 frames/s$. For the well with a radius of 100μm the acquired image sequence after cropping and resampling was $I(x,y,t) = 420\times416\times2849$ voxels. From the cut-off frequency of a lens $2NA/\lambda$ we get the Nyquist sampling criterion $4NA/\lambda = 4.95$ samples/μm. This sampling would be necessary to reconstruct the image correctly from an observation with this lens system. However, we could only achieve a spatial sampling density of 1.92 samples/μm. We wanted to observe the whole well in one image, making a trade off between the field of view and the sampling density. Nevertheless, we have one pixel per 0.52μm which is a microsphere diameter. On the other hand the time sampling

is very fine, the velocities are much smaller than one pixel per frame. The overall sampling is sufficient for our analysis task. If we would use smaller microspheres only image the PSF would be imaged.

For the well with a radius of 150μm the acquired image sequence after cropping and resampling was $I(x, y, t) = 473\times384\times3618$ voxels. The spatial sampling density was 1.29 samples/μm. In both cases the acquisition started when the meniscus was convex. The instant $t = 0$ corresponding to a flat meniscus was at frame 711 for the 100μm well and at frame 513 for the 150μm well. Microarrays with different sized wells were etched in silicon dioxide at DIMES (Delft Institute for Microelectronics and Submicron Technology).

8.6 Brownian motion in nanoliter wells

In several experiments we investigate the Brownian motion of microspheres in nanoliter wells. Brownian motion cannot be studied in the former experiments independently of the outward directed flow. Even in the early phase of the evaporation the flow is of the same order as the expected Brownian motion. Therefore we perform experiments in which we fill the wells ($R = 200\mu$m, $d = 40\mu$m) with different aqueous solutions of ethylene glycol, add microspheres and then seal the wells by a glass cover slip to prevent evaporation. This ensures that the liquid is stationary.

From the study of the Brownian motion it is possible to extract one of the following quantities: the diffusion constant D, the viscosity of the medium η, the radius of the spheres r or the temperature T. The connection between the random motion of the microspheres and their average displacement is given by the diffusion equation (6.46). The expected position of a tracer particle after a time t is the initial position, but the variance of the observation is growing (linearly) with time

$$\left\langle x^2 \right\rangle = 2nDt\,. \tag{D.9}$$

Here we have to keep in mind that only the projected 3D Brownian motion is observed and we have to apply the stereological factor $\pi/4$

$$\left\langle x_{ob}^2 \right\rangle = \frac{3}{8}\pi^2 Dt\,. \tag{8.7}$$

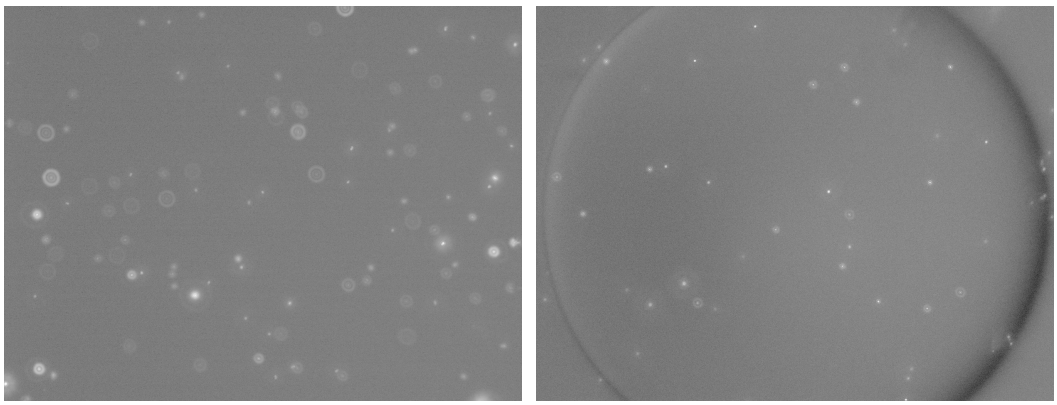
A connection between the macroscopic diffusion constant and the microscopic parameters is established by the Einstein relation [32]

$$D = \frac{k_B T}{6\pi\eta r}\,. \tag{6.48}$$

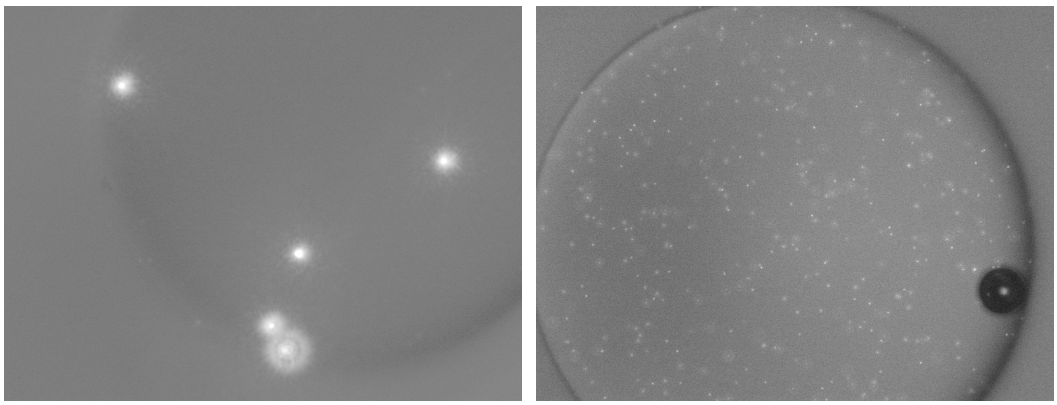
In our experiments we want to measure the diffusion constant via observations of the variation of the displacement eq.(8.7) as a function of the radius of the

microspheres and as a function of the viscosity of the aqueous solution of ethylene glycol ($C_2H_4(OH)_2$). Finally we can compare the measurements with the theoretical values from eq.(6.48).

In figures 8.8a-d) we show typical images of wells filled with fluorescent microspheres. We used microspheres with radii of 55, 95, 280, 485 and 1050nm in solutions ranging from pure water to pure ethylene glycol. For all experiments we acquired in between 150 and 360 time frames (100-600s).



(a) **(b)**



(c) **(d)**

Figure 8.8: Images of different microspheres in aqueous solutions of ethylene glycol in nanoliter wells (all displays log stretch). The image size is 1334×1024 pixels or $408 \times 311 \mu m^2$. a) square well with pure water r=95nm, b) round well with pure ethylene glycol r=95nm, c) 10% ethylen glycol r=1050nm, d) 50% ethylen glycol, r=55nm. For the processing a region of interest is manually selected such that the well borders are excluded.

8.6.1 Estimation algorithm

To estimate the displacement per tracer particle we have to follow the individual particle over time and record its space-time position. In recorded images as in figures 8.8a,b,d) there are many visible microspheres present (dependent on the radius of the spheres). Tracking theses 'blobs' is difficult if there is a high density of blobs. In the following we will describe a straightforward and simple algorithm. For more sophisticated tracking methods see for example [8, 20].

The first step in our algorithm is to segment the blobs from the background. For larger microspheres the signal-to-noise-ratio (SNR) is good as there is a lot of fluorescent material present, for smaller radii this decreases with the third power, compare figure 8.8c) and d). As a preprocessing step to enhance the blobs, if needed, we use a tophat filter $I - maxf(minf(I))$ [131]. Then a manually selected threshold is applied to segment the objects. The objects in each time frame are labeled automatically. Starting from all detected objects in the first time frame the nearest object in the following time frame is located (within a search radius of a few times the expected motion in this time interval). Due to merging or detection failure of objects in time frames, this procedure can produce ambiguous paths. With a post-processing step only unique paths are extracted, where at merging paths the shortest path is considered. From theses paths the diffusion constant is calculated from non-overlapping pairs at time points $\{i, i+\Delta t\}, \{i+\Delta t+1, i+2\Delta t+1\}, \ldots$ of positions of an object. With this method several estimates for D can be obtained from one tracer particle. A high number of pairs N is desirable to get a robust estimate for the diffusion constant.

It is to be expected that the outlined algorithm will not work well under all circumstances. The tracking will always choose the closest particle (which might not be the right one). The tracking will fail or choose the wrong particle more often for a high diffusion constant and for high densities of tracer particles. For these cases the algorithm is to be expected to have a negative bias (underestimation of the true displacement). Evaluation of the algorithm on simulated Brownian motion of blobs in images has been done to verify this behavior.

r[nm]	% ethylene glycol in aqueous solution η[mPas] at $T = 20°$C 0 1.00	10 1.27	50 3.8*	70 6.3*	100 19.9	$D[\frac{\mu m^2}{s}]$
55			1.03 1.69 ± 0.23 (124/417)			D_{theo} D_{exp} (n/N)
95	2.26 1.34 ± 0.16 (58/266)	1.78 1.49 ± 0.18 (105/331)	0.594 $0.594 \pm 0.10^{\dagger}$ (98/382)	0.358 0.357 ± 0.051 (235/273)	0.113 0.109 ± 0.016 (23/273)	D_{theo} D_{exp} (n/N)
280	0.766 $0.921 \pm 0.083^{\ddagger}$ (6/296)	0.603 0.60 ± 0.22 (11/27)	0.201 0.213 ± 0.084 (6/30)	0.122 0.137 ± 0.016 (48/265)	0.0384 0.0520 ± 0.0069 (30/219)	D_{theo} D_{exp} (n/N)
485	0.442 0.374 ± 0.16 (3/20)	0.348 0.640 ± 0.18 (3/51)	0.116 0.127 ± 0.017 (18/194)			D_{theo} D_{exp} (n/N)
1050	0.204 0.476 ± 0.15 (19/106)	0.160 0.174 ± 0.051 (2/44)				D_{theo} D_{exp} (n/N)

Table 8.1: Estimations of the diffusion constant $D = \frac{k_B T}{6\pi\eta r}$. Measurements were preformed with different combinations of microsphere radii and aqueous solutions of ethylene glycol. Viscosity values from [73], where values with * have been interpolated. 0% indicates pure water and 100% pure ethylene glycol. The mean value and the 95% confidence intervall is shown for all measurements. n is the number of tracer particles extracted in the first time frame and N the number of position pairs used to computed D with $\Delta t \approx 25$s. †the average of three repetitions of the same experiments. ‡here $\Delta t \approx 1$s because during the experiment there is a visible (evaporation) flow.

8.6.2 Evaluation

In table 8.1 we show the estimated diffusion constant for different aqueous solutions of ethylene glycol. We show the values computed for pairs separated by $\Delta t \approx 25$s. The number of initial tracers is given along with the number of pairs N used. We display the mean and the 95% confidence interval in which the mean lies. The statistics are based on the estimation of an unknown mean from N independent normal distributed measurements (Student t-distribution). The estimated mean μ lies with $1-\alpha$ probability within $\mu \pm \frac{s}{\sqrt{N}} t_{\alpha/2}(N-1)$, where $\alpha = 0.05$, N the number of pairs, s the measured standard deviation and $t_{\alpha/2}(N-1)$ a tabulated value for the left and right sided percentile of the Student t-distribution [17]. For our measurements we have $N \gtrsim 20,\ t_{\alpha/2}(N-1) \approx 2$.

In figure 8.9 we plot the estimates for Dr (scale invariant) as a function of $1/\eta$ and for $D\eta$ (viscosity invariant) as a function of $1/r$. In general, the estimated values for the macroscopic diffusion constant are in agreement with the microscopic theory. A few measurements, however, stick out with a high overestimation $\{\eta = 1.0\text{mPas}, r = 1050\text{nm}\}$, $\{\eta = 1.27\text{mPas}, r = 485\text{nm}\}$ and $\{\eta = 3.8\text{mPas}, r = 55\text{nm}\}$. We have no explanation for this overestimation. A leak in the sealing of the well by the glass cover slip could be an explanation, but there is no visible flow in these time series.

Finally, we investigate the estimation behavior as a function of the time Δt, which is correlated to the number of pairs N available from one measurement. In figure 8.10 we plot this dependency for two experiments ($\eta = 1.27$mPas, $r = 95$nm and $\eta = 6.3$mPas, $r = 95$nm). We see that the choice of Δt is not critical for the estimation. However, with increasing Δt the number of pairs N is decreasing resulting in a higher uncertainty of the mean.

8.6.3 Conclusions

We showed that a relative classification of diffusion constants is possible with an image analysis technique based on an experiment in the line with the one by the botanist Brown who noticed that pollen grains suspended in water jiggle about under the lens of the microscope [18].

For our algorithm it is desirable to only have a few number of initial tracers resulting in a low density so that the tracking will not fail. To compensate for a lower number of pairs, the measurement time should be long, which enables more independent estimates from the same tracer particle.

A different approach to measure the diffusion constant would be to introduce one droplet of a colored liquid and to observe the spread of the droplet in time. All the color particles are used as tracers of the diffusion and the tracking can

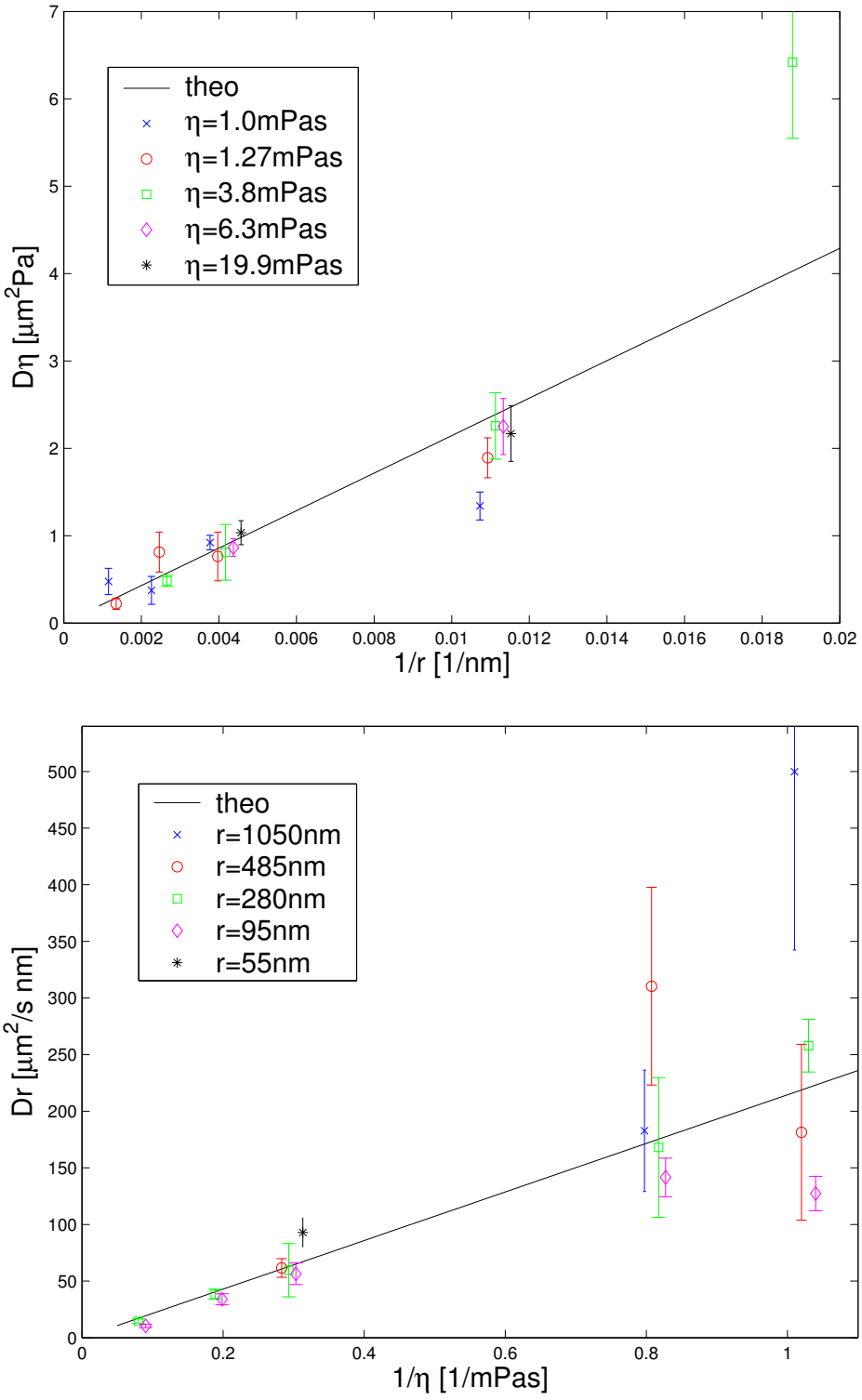
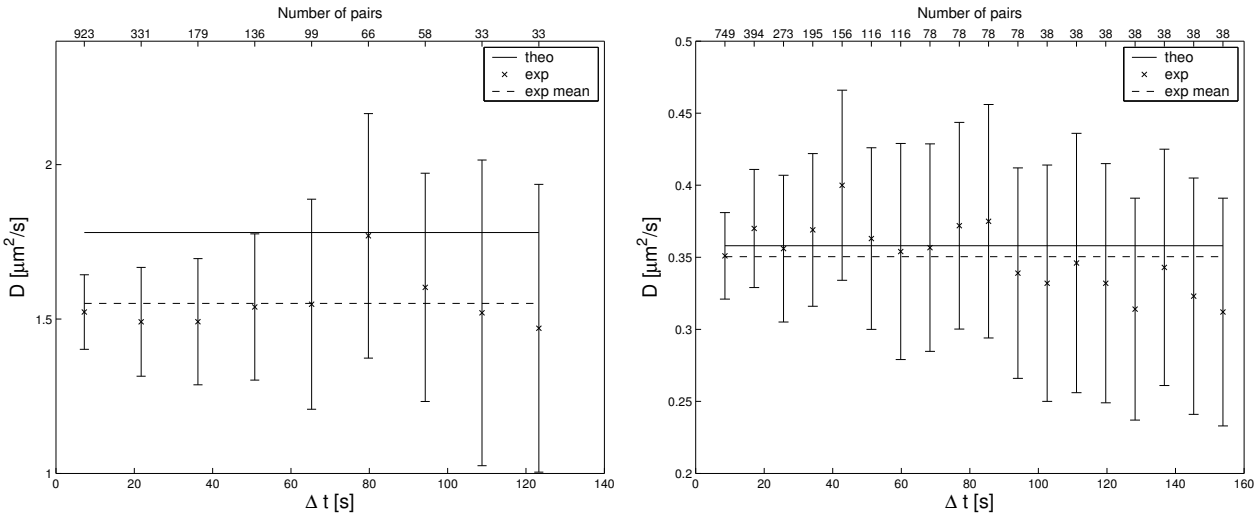


Figure 8.9: Plots of the estimated diffusion constants of table 8.1. a) $D\eta$ as a function of $1/r$ and b) Dr as a function of $1/\eta$.



(a) **(b)**

Figure 8.10: Estimation of the diffusion constant for two experiments as a function of Δt which is correlated to the number of pairs N that can be extracted from the data a) $\eta = 1.27$mPas, $r = 95$nm b) $\eta = 6.3$mPas, $r = 95$nm.

be omitted and replaced by an area/intensity measurement. In general, another possibility – however not applicable to our problem – is to monitor a fixed volume in time and count the number of tracers in it or measure the intensity from that area if the intensity is proportional to the number of tracers. From the fluctuations over time the diffusion constant can be calculated.

Materials and Methods

The experiments were performed with a Leica DM RXA confocal microscope, 20x/0.40 lens (in wide-field mode) with a Hamamatsu ORCA-ER C4742-95 CCD camera. The images were recorded with 12 bpp resolution on 1344×1024 pixels. The sampling density was measured by placing a stage micrometer under the microscope $\Delta_x = \Delta_y = 0.304\mu$m or 3.29 samples/$\mu$m. The microarray of wells ($R = 200\mu$m, d=40μm) was placed on a cooling stage which kept the temperature constant at $T = 20°$C. Rhodamine-stained microspheres (Fluospheres, Molecular Probes, Leiden, The Netherlands, $\lambda_{ex} = 580$nm and $\lambda_{em} = 605$nm) were used as tracer particles in the solution. Microarrays with wells were etched in silicon dioxide at DIMES (Delft Institute for Microelectronics and Submicron Technology).

Chapter 9

Application 3: Group motion correction applied to cell nuclear motion[10]

Nowadays, confocal microscopy is used for the quantitative 4D imaging of spatio-temporal processes in living cells. The transport of proteins in the cell during the cell cycle is presumably a process that plays a role in RNA transcription and DNA repair. Fluorescent labeling of such a protein reveals tens of moving spots in the cell. This motion, however, is superimposed on the translation and rotation of the cell as a whole. In this chapter we propose a fully automatic method to remove this superimposed motion without first establishing point-to-point correspondence. After removing the global motion the protein transport can be analyzed. The correction method is robust against noise and different axial and lateral resolution typical for confocal microscopy.

Acquisition and analysis of 4D image data (x, y, z, t) is an important tool in molecular cell biology to study the function of certain proteins in dynamic cellular processes such as RNA transcription/transport and DNA repair/replication. Fluorescent labeling facilitates 4D imaging of specific proteins using confocal microscopy. The targets show up as moving bright spots. Often their velocity, absolute or with respect to each other, is of interest [8]. Superimposed on the spot motion itself is the motion of the cell nucleus during the acquisition time (typically 30-120 min.). We have developed an automatic procedure to estimate and remove the superimposed motion without the need to establish correspondence. The affine motion can be separated into translation and rotation [111]. First the translation is estimated via the center of mass motion and, secondly the rotation parameters are estimated by constructing an intrinsic body system in each time frame via the inertia tensor. Note that we learned from mechanics that translation and rotation can always

[10]The experimental work was done at the Leiden University Medical Center by C. Molenaar and R.W. Dirks, which are part of the interdisciplinary project in which this thesis is carried out.

be treated separately. Both techniques work on the grey-value information of the image and do neither require tracking nor segmentation. The construction of an intrinsic body system via the inertia tensor has the big advantage that the rotation in 3D needs not to be done via the Euler angles. To compute the rotation matrix from two given systems via the Euler angels is conceptually complex. A generic 3D rotation $\mathbf{R}$ consists of three consecutive rotations in a 2D plane around the normal axis. The generic rotation $\mathbf{R} = \mathbf{R}_1\mathbf{R}_2\mathbf{R}_3$ obtained by matrix multiplication is not commutative, requiring an iterative one-by-one method to estimate the rotation angles. It is much easier to calculate the rotation matrix by combining the basis transformation in two time frames to an unique intrinsic coordinate system induced by the inertia tensor.

Often the correction for the cell motion is done by hand or by an algorithm that requires segmentation and matching of the objects. Soo-Chang Pei et al. [97] have presented another correspondences-less approach where they obtain the motion parameters via different order moments. However, they cannot give a confidence measure if the algorithms fails. For symmetric bodies they fall back to a correspondence based approach. Our algorithm can automatically detect symmetric body constellations (for which the body system is not unique). As long as the body is not a sphere we can reduce the problem's dimensionality and solve the rotation by an explicit 2D rotation, which is easily characterized by only one angle. Again we can avoid the conceptual difficult Euler angles.

The goal of our approach is to compute the overall cell motion by observation of the protein motion only. The hypothesis that we need to check is: do the labeled protein spots move approximately the same as the cell nucleus (stain)? The motion of the stain and spots cannot be entirely correlated as the spots have an interesting biological function that we want to study. To test the hypothesis we acquire protein spots and the cell nucleus at the same time by using two different fluorescent markers and compute the motion parameter on both independently. If these parameters show good correspondence we can conclude that our approach is suitable. Furthermore, we want to study the effect of the correction on the motion trajectories of the spots. An accelerated motion of the cell should be visible via a curved spot trajectory in 4D before correction and a more straight trajectory after correction.

9.1 Theory of motion parameter estimation

Using techniques based on the intensity of the image we are going to retrieve the motion parameters of the affine transformation of the overall body motion without establishing correspondence of special points. We assume that only a single structure of interest, for example a cell, is in the field of view during the acquisition time.

Furthermore, shifting and rotating the acquired 3D time series is computationally complex if the voxels are anisotropic, which is often the case in confocal microscopy as the lateral resolution is about 3-5 times higher than the axial. Therefore, as a preprocessing step the data is resampled to make the voxels isotropic.

9.1.1 Translation

If no movement of the individual spots and only movement due to drift of the cell nucleus occurs then the motion of the center of mass of the spot image perfectly characterizes the drift. Biological mass (protein/DNA) in images is identified through the intensity values $I(x)$ of the measured fluorophores. As long as the displacement of individual spots between two successive time frames is small, the center of mass will not be effected significantly, as the number of bright (heavy) spots is large. The center of mass is calculated in each time frame

$$x_{CM} := \frac{1}{\sum I(x)} \sum I(x) x \,. \tag{9.1}$$

After computation of the x_{CM} for all time frames, we impose a shift on all images such that their x_{CM} coincide. We choose to shift all x_{CM} to the image center. This ensures that we keep maximal distance of the cell body to the images boundaries. For image processing purposes it is desirable to avoid border effects of filtering operation, which is practically achieved by a distance of ≈ 3 filter sizes of the object from the border (at a distance of 3σ the Gaussian filters will only be marginally influenced by the border).

After subtracting the center of mass motion we will correct for the rotation, which is the second step in this registration technique.

9.1.2 Rotation

A suitable way to find the rotation of a body without identifying special points and track them in time, is to construct a unique intrinsic coordinate system of the body at every time step. Such a system is induced by the principal axes of the inertia tensor [111]

$$J_{\mu\nu} := \frac{1}{\sum I(x)} \sum I(x) \left(x^2 \delta_{\mu\nu} - x_\mu x_\nu \right) , \tag{9.2}$$

where the coordinate origin is the center of mass and $\delta_{\mu\nu}$ is the Kronecker-delta [17]. The inertia tensor for images is normalized by the sum of the grey-values in contrast to the inertia tensor used in classical mechanics [111, 56]. The inertia tensor is real, symmetric and positive. Therefore it can be diagonalized by an orthonormal

transformation $\mathbf{D} \in \mathrm{SO(n)}$ as

$$\mathbf{DJD^{-1}} = \tilde{\mathbf{J}}\,, \tag{9.3}$$

where $\tilde{\mathbf{J}}$ is diagonal. The eigenvectors of $\mathbf{J}$ form an orthogonal system and are called *principal axes*. Furthermore, the eigenvalues of $\mathbf{J}$ have the following property: $\lambda_i \geq 0$ and the sum of any two eigenvalues is greater or equal than the remaining one. We assume that $\tilde{\mathbf{J}}$ can be considered constant between two successive time frames. The rotation between two time frames t and $t+1$ of a group of objects described by their inertia tensor can be formulated with a rotation matrix $\mathbf{R} \in \mathrm{SO(n)}$ as

$$\mathbf{R}\mathbf{J}_t\mathbf{R}^{-1} = \mathbf{J}_{t+1}\,. \tag{9.4}$$

In 3D one or two rotations around some axes are in general not enough to let an object coincide. A possible way is the formulation by Euler angles as

$$\mathbf{R} = \mathbf{R}_{3''}(\gamma)\mathbf{R}_{2'}(\beta)\mathbf{R}_3(\alpha)\,. \tag{9.5}$$

The first rotation is about angle α around the initial 3-axis. The second about angleβ around the intermediate 2-axis and the last about an angle γ around the final 3-axis. The intervals for the angles are $\alpha \in [0, 2\pi],\ \beta \in [0, \pi],\ \gamma \in [0, 2\pi]$ [111] [11]. Solving eq.(9.4) for $\mathbf{R}$ and the three Euler angels in closed-form is not feasible with this rotation matrix

$$\mathbf{R} = \begin{pmatrix} \cos\gamma\cos\alpha - \cos\beta\sin\alpha\sin\gamma & \cos\gamma\sin\alpha + \cos\beta\cos\alpha\cos\gamma & \sin\beta\sin\gamma \\ -\sin\gamma\cos\alpha - \cos\beta\sin\alpha\sin\gamma & -\sin\gamma\sin\alpha + \cos\beta\cos\alpha\cos\gamma & \sin\beta\cos\gamma \\ \sin\beta\sin\alpha & -\sin\beta\cos\alpha & \cos\beta \end{pmatrix} \tag{9.6}$$

Fortunately we are in the situation that the matrix under consideration $\mathbf{J}$ is real and symmetric, so it can be diagonalized. For two time frames t and $t+1$ we compute

$$\mathbf{J}_t = \mathbf{D}_t^{-1}\tilde{\mathbf{J}}\mathbf{D}_t \tag{9.7}$$

$$\mathbf{J}_{t+1} = \mathbf{D}_{t+1}^{-1}\tilde{\mathbf{J}}\mathbf{D}_{t+1}\,. \tag{9.8}$$

Solving for $\tilde{\mathbf{J}}$ we get

$$\mathbf{J}_{t+1} = \mathbf{D}_{t+1}^{-1}\mathbf{D}_t\mathbf{J}_t\mathbf{D}_t^{-1}\mathbf{D}_{t+1}\,. \tag{9.9}$$

The overall rotation matrix is

$$\mathbf{R} = \mathbf{D}_{t+1}^{-1}\mathbf{D}_t\,. \tag{9.10}$$

For the actual computation we can use the orthogonality of $\mathbf{R}$ and $\mathbf{D}$. This a major advantage over the use of classical Euler rotation. This trick is only applicable where an intrinsic body system can be constructed at each time frame independently.

[11] There is another definition of the Euler angles. The second rotation is then about the intermediate 1-axis. This convention is traditionally used in the mechanics of the rigid body. The angles are called ψ, θ, ϕ and their relation with the other angles is $\phi = \gamma - \frac{\pi}{2} (\mathrm{mod}\ 2\pi),\ \theta = \beta,\ \psi = \alpha + \frac{\pi}{2} (\mathrm{mod}\ 2\pi)$.

9.1.3 Degeneration of the eigenvalue system; a confidence measure for correspondence

The computation of the rotation matrix $\mathbf{R}$ is unique as long as the eigenvalue system of $\mathbf{J}$ is not degenerated. Let $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ be the eigenvalues of $\mathbf{J}$ and $\lambda_2 + \lambda_3 \geq \lambda_1$ due to the definition of the inertia tensor. We propose a normalized measure to indicate degeneration as

$$c_{deg} = \frac{\min(\lambda_1 - \lambda_2, \lambda_2 - \lambda_3)}{\lambda_1/3} \in [0, 1]\,. \tag{9.11}$$

Where $c_{deg} = 1$ indicates maximal variance in the distribution of the eigenvalues and $c_{deg} \searrow 0$ indicates degeneration. The confidence measure can be interpreted geometrically. Imagine the λ_i's as distances in Euclidean space, then the condition

$$\lambda_i + \lambda_j \geq \lambda_k,\ \ i, j, k \in \{1, 2, 3\} \text{ cyclic}\,, \tag{9.12}$$

is the triangle inequality, where $c_{deg} \searrow 0$ indicates symmetry of the triangle constructed by the line segments of length λ_i, as $c_{deg} \nearrow 1$ the triangle becomes more "asymmetric". As long as λ_1^t corresponds to the same axis as λ_1^{t+1} etc. the rotation is computed correctly. If the initial eigensystem is not degenerated then the correspondence can only be broken by a transition via degenerated state, which we can detect.

If the body is truly symmetric, and not only the inertia, then for registration purposes it does not matter how the body axes are chosen. If the inertia is symmetric (two-fold degeneration) but the distribution of the spots is not, then we can resolve the ambiguous coordinate system as follows. We project all spots onto the symmetry plane reducing the dimensionality of the problem to 2D. By cross-correlation, grey-value difference, 2D inertia tensor or Fourier-Mellin analysis [21, 117] applied in this symmetry plane on two successive time frames, we obtain a 2D rotation angle about the symmetry axis. The use of Euler angles is again circumvented by projection and easy 2D rotation.

9.1.4 Stability of the eigenvalue analysis

Numerical eigenvalue analysis of noisy data is not very robust. In our case the direction of corresponding eigenvectors in subsequent time frames should not flip. A flip can be detected via the sign of the volume spanned by the eigen system $\{v_i\}$ of $\mathbf{J}$, i.e. $sgn(\det(\{v_i\}))$. To stabilize the computed eigen system we require the system to have the same orientation for all time frames, i.e. either left or right handed.

9.1.5 Rotation of images

We want the rotation point of a 3D image to be the center of mass that we fixed in section 9.1.1 to be the image center. The easiest way to do this is by translation of the image about a vector v. A well known formalism in computer graphics to combine a translation, rotation and another translation into one matrix operation is through the concept of the *homogeneous coordinates* [49]. The translation can be expressed in a matrix by

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 & -v_x \\ 0 & 1 & 0 & -v_y \\ 0 & 0 & 1 & -v_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, \ \tilde{\mathbf{T}} = \begin{pmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{9.13}$$

The rotation matrix $\mathbf{R}$ is expanded into a 4×4 matrix

$$\hat{\mathbf{R}} = \begin{pmatrix} & & & 0 \\ & \mathbf{R} & & 0 \\ & & & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{9.14}$$

Combining these matrices the transformation of an old vector $(x, y, z)^t$ to a new vector $(x', y', z')^t$ becomes [49]

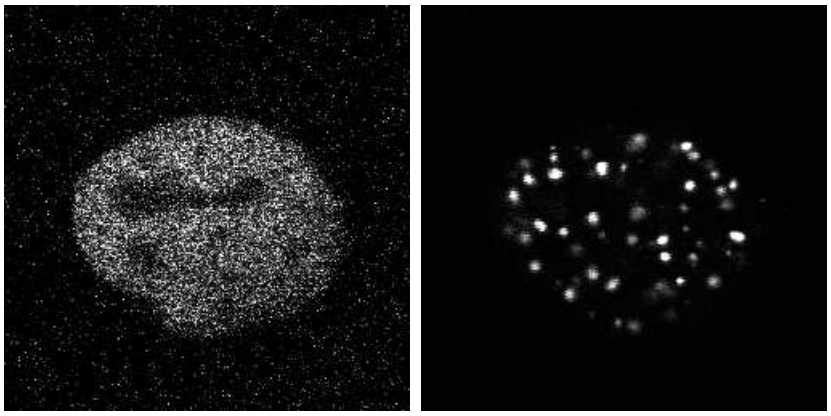
$$\begin{pmatrix} x' \\ y' \\ z' \\ \cdot \end{pmatrix} = \tilde{\mathbf{T}}\hat{\mathbf{R}}\mathbf{T} \begin{pmatrix} x \\ y \\ z \\ \cdot \end{pmatrix}. \tag{9.15}$$

For a rotation around the center of the image the translation vector must be chosen half the image size.

The actual application to an image requires some more steps due to the quantization of the image. For a given voxel in the output image there is in general no associated voxel in the input image, so that value must be obtained through interpolation. Different methods of interpolation are known in the literature, as linear (first order hold), bispline, cubic, etc.. In our case the simplest of them (linear), already turned out to be sufficient.

9.1.6 Measure for rotations

The Euler angles do not give an easy interpretation of how much an object is rotated. More intuitive measures for the amount/degree of rotation are the rotation matrix $\mathbf{R}$ and the angles between corresponding pairs of eigenvectors of the inertia tensor



(a) **(b)**

Figure 9.1: a) Stain H4-CFP, b) spots PML-YFP at time frame 8, z-slice 25.

J. For the rotation matrix the deviation from the unit matrix defines a measure for rotation. For the eigenvectors of **J** the angles give a very good measure for the rotation.

9.2 Application to real data

Three time series of 15 time frames each were recorded in a time-span of about one hour. The image has two channels for two nuclear proteins (channels). The 3D images are of size $256 \times 256 \times 31$ voxels with a physical voxel size of $148 \times 148 \times 244$nm and were recorded with 8 bit integer values. The cells are not spheres in this development stage, but rather flat, comparable to a bread dough before rising. The cell is captured in an image of about $38 \times 38 \times 8\mu$m. One protein (H4-CFP) is distributed over the cell body, which we will refer to as stain. The other one (PML-YFP) is localized at distinct spots, see figure 9.1. The acquisition of the stain along with the spots in another channel enables us to test our correction algorithm.

9.2.1 Experimental setup

The living cell recordings where done by C. Molenaar and R.W. Dirks at the Leiden University Medical Center.

Two nuclear proteins were visualized in living cells: Histon H4 and PML. The DNA-helix in eukaryotic cells is highly condensed. The first degree of packaging is formed by nucleosomes. About 146 base pairs are wrapped around protein octamers containing two copies of each of the four core histones, H2A, H2B, H3 and

H4. The H3-H4 tetramer forms the inner core of the nucleosomes and these proteins are stable bound to DNA. The PML protein is typically found concentrated in 20-50 discrete nuclear spots, called PML bodies. The PML body has been associated with several human disorders, such as acute promyelocytic leukemia and AIDS. The molecular functions of PML bodies are not completely understood, but have been ascribed roles in RNA transcription/transport and DNA repair/replication. Using DNA-recombinant techniques, two auto fluorescent (AFP) fusion proteins were constructed: Histon H4-CFP and PML-YFP and expressed in human osteaosarcoma cells (U-2 OS). The chimeric proteins mimic the function and localization of the endogenous proteins and enable dynamic live cell imaging. Expression of Histon H4-CFP results in a total nuclear staining and expression of PML-YFP in staining of ≈ 30 PML bodies.

9.2.2 Methods: Cloning of GFP-fusion proteins

The cDNAs encoding PML and Histon H4 were generated by RT-PCR on cDNA copied from mRNA isolated from human U-2 OS cells and cloned into the pECFP-C1 or pEYFP-C1 vectors (Clontech, Palo Alto, CA).

9.2.3 Live cell imaging

The AFP-fusion proteins were transiently expressed in U-2-OS cells using DOTAP (Roche Diagnostics GmbH). Cells were analyzed 24-48 h after transfection and were selected for moderate expression and protein-specific localization. The temperature of the cells was maintained at 37°C using a heated ring surrounding the culture chamber and a microscope objective heater (Bioptechs, Butler, PA). Images were acquired on a Leica TCS/SP2 confocal microscope system using a 100× NA 1.4PL APO lens. The 457 nm and 514 nm lines of the Argon laser were used for respectively CFP and YFP excitation. CFP and YFP were sequentially scanned to avoid cross-talk through in the emission window.

9.2.4 Results

For all three time series and in both channels the degeneration confidence measure (9.11) is $c_{deg} > 0.75\ \forall t$, i.e. there is no transition via a symmetrical state. This indicates that our approach is valid under real circumstances. In figures 9.2 and 9.4a) we have plotted for the series the computed center of mass shift and rotation of the eigenvectors, figures 9.2, 9.4 and 9.6b-d), for the stain and spots. The shift estimate of the stain and spots is in very good agreement. The overall shift is shown in table 9.2. The rotation of corresponding eigenvectors between time frames is

also well correlated. Therefore it is reasonable to say that the motion parameters of the whole cell can be computed from the spots only, if they are distributed over the whole cell. More important, it is not to be expected that the spot and stain rotation is exactly the same. The spots move within the stain, but perform a motion correlated to their biological activity, which we want to study. Even stronger, if the stain and the spot motion would not differ at all, then the labeled spots, would not have a biological interesting function, as they just stand still inside the nucleus. In table 9.1 we show the computed correlation between the stain and spots rotation of the different eigenvectors. We compute the correlation as the cosine of the angle between the vectors and as the normalized difference between the vectors formed by the rotation angle of the eigenvectors of the stain and the spots plotted in figures 9.2, 9.4 and 9.6b-d). Furthermore, we see a strong correlation between the

	Series 1		Series2		Series 3	
	ρ	diff	ρ	diff	ρ	diff
corr(stain,spots) v_1	0.79	0.37	0.88	0.31	0.80	0.35
corr(stain,spots) v_2	0.81	0.36	0.88	0.34	0.83	0.32
corr(stain,spots) v_3	0.93	0.24	0.94	0.24	0.89	0.24
corr(spots,spots) v_1, v_2	1.0	0.049	0.99	0.050	0.99	0.048

Table 9.1: Correlation between the spots and stain rotation for the different eigenvectors. Last row: Correlation between the first and second eigenvector of the spots. $\rho := \frac{v_i(stain)\cdot v_i(spots)}{\|v_i(stain)\|\|v_i(spots)\|}$, diff:$=\frac{\|v_i(stain)-v_i(spots)\|}{\|v_i(stain)\|+\|v_i(spots)\|}$. A ρ close to one indicates good correlation.

motion of the first v_1 and second eigenvector v_2 in figures 9.2, 9.4, 9.6b) and c). They rotate more than v_3 in figures 9.2, 9.4, 9.6d). The correlation between the v_1 and v_2 rotation comes from the flat shape of the cell. A typical cell is 4-5 times wider than high. Based on this shape of cell we conclude therefore: v_1 and v_2 lie approximately in the xy-plane, i.e. the elongated direction of the cell, and v_3 close to the axial direction. Thus the cell rolls little in the axial direction during acquisition compared to the rotation in the lateral plane.

	Shift stain	Shift spots	$\bar{c}_{1D}$ before	$\bar{c}_{1D}$ after
Series 1	3.32μm	3.53μm	0.145	0.175
Series 2	4.32μm	4.31μm	0.145	0.187
Series 3	2.52μm	2.56μm	0.142	0.191

Table 9.2: Total shift of the center of mass computed form the stain and the spot channels. The mean confidence $\bar{c}_{1D}$ before and after the correction.

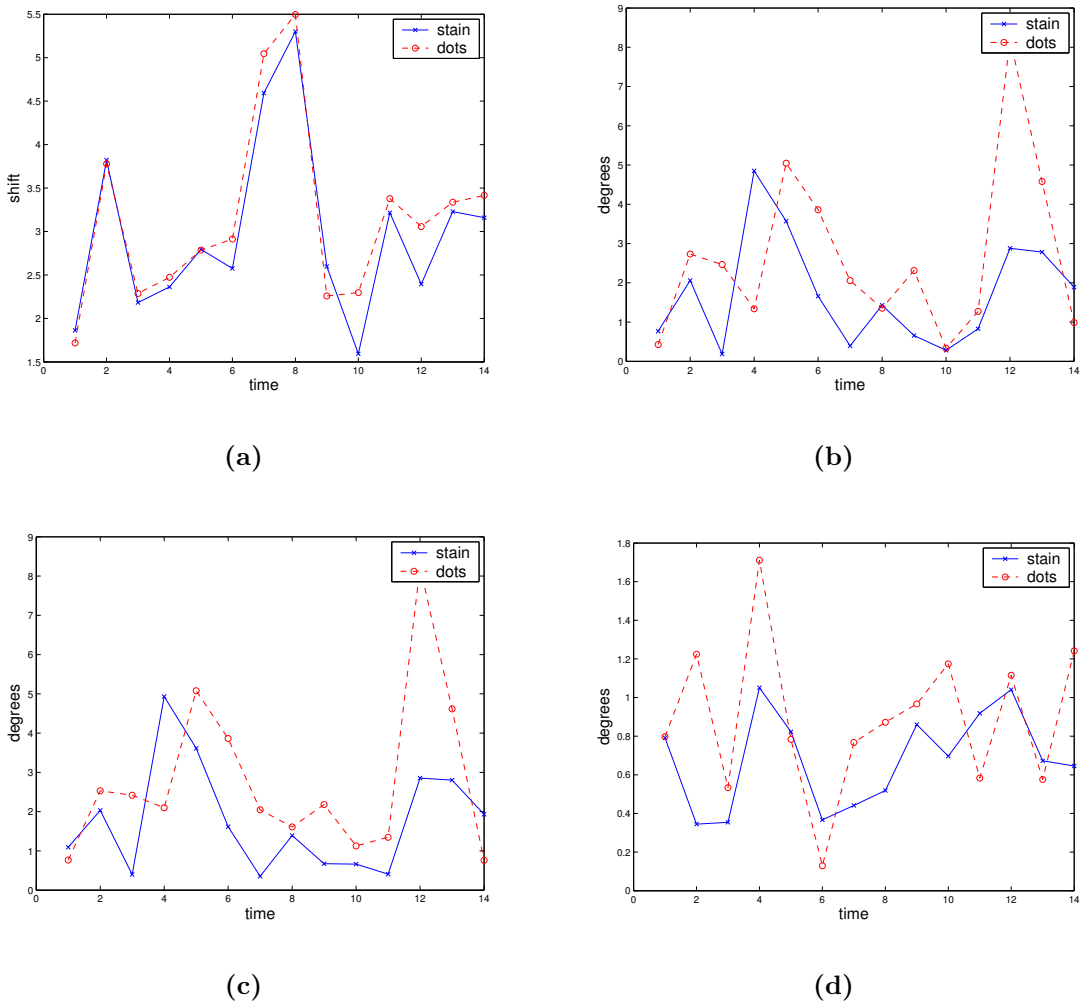


Figure 9.2: Series 1: a) Shift of the center of mass in pixel, b) Rotation of the first eigenvector in degrees between time frames, rotation of the second c) and third d) eigenvector in degrees between time frames.

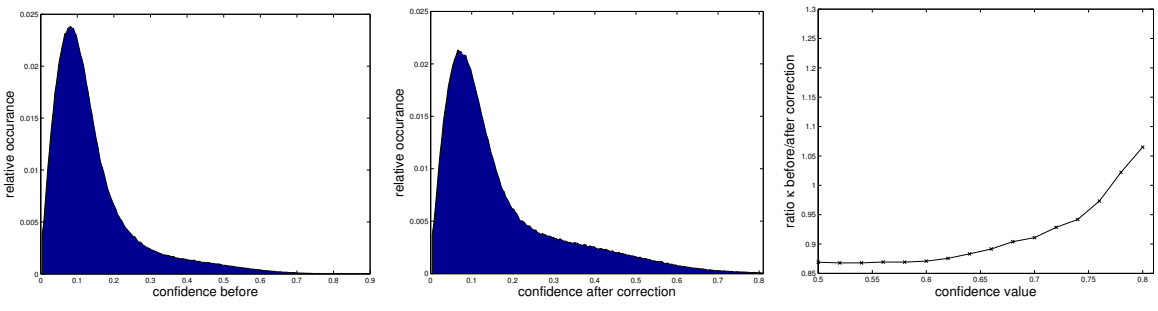


Figure 9.3: Series 1: a) Histogram of c_{1D} before correction, b) after correction, c) $\bar{\kappa}_{before}/\bar{\kappa}_{after}$ as a function of the regions with c_{1D} higher than the threshold.

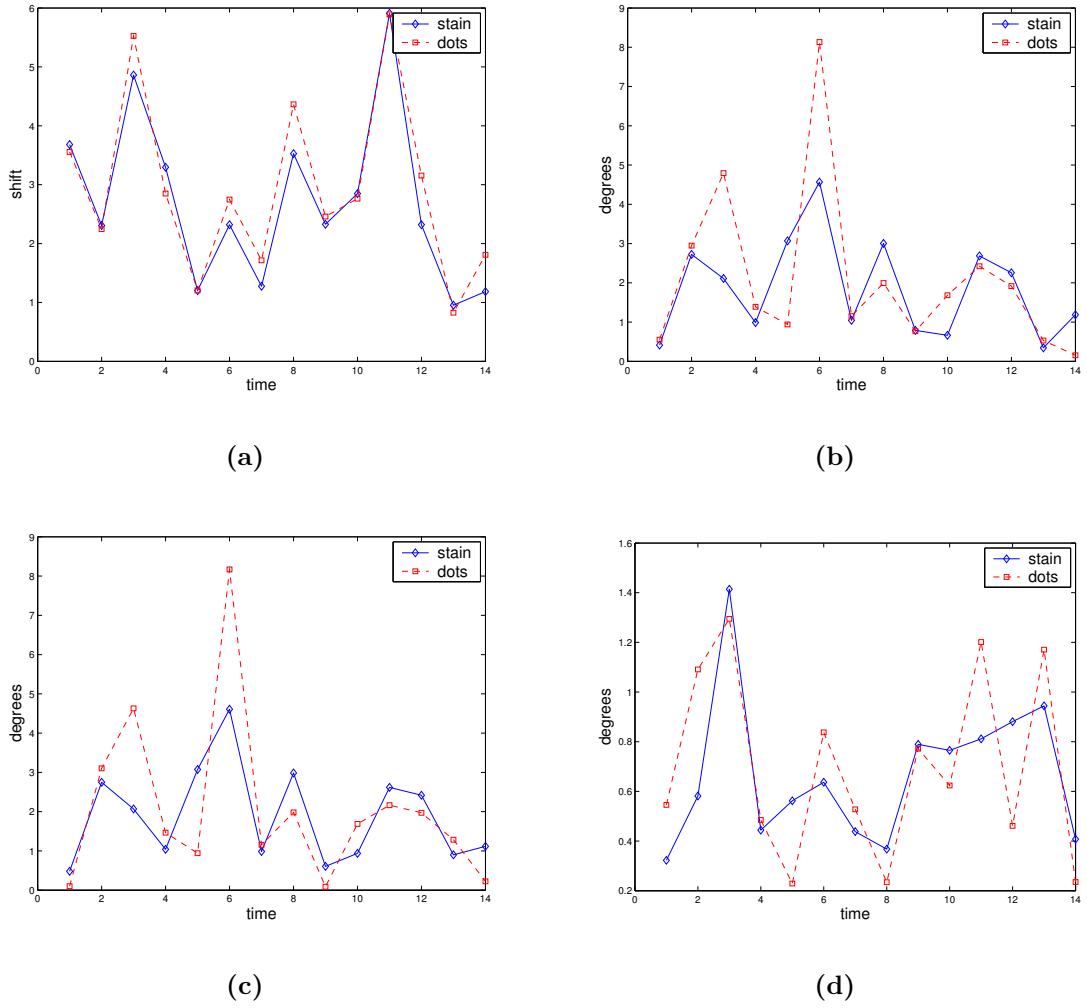


Figure 9.4: Series 2: a) Shift of the center of mass in pixel, b) Rotation of the first eigenvector in degrees between time frames, rotation of the second c) and third d) eigenvector in degrees between time frames.

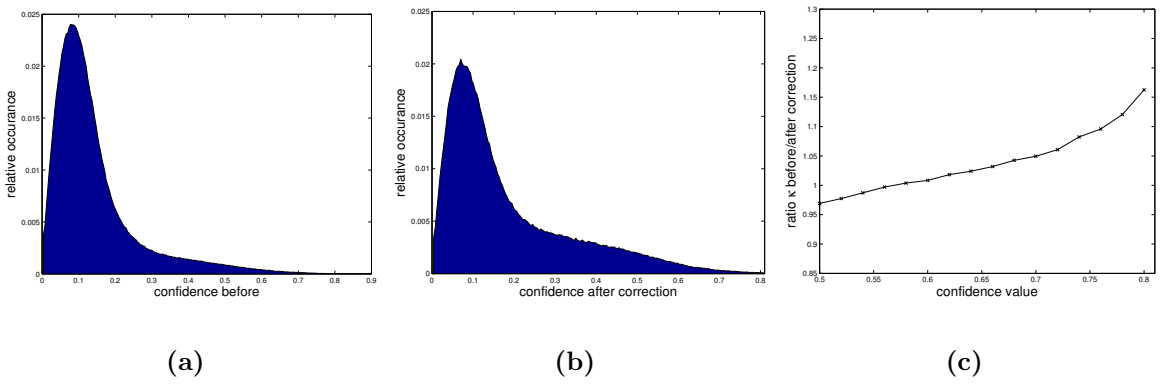


Figure 9.5: Series 2: a) Histogram of c_{1D} before correction, b) after correction, c) $\bar{\kappa}_{before}/\bar{\kappa}_{after}$ as a function of the regions with c_{1D} higher than the threshold.

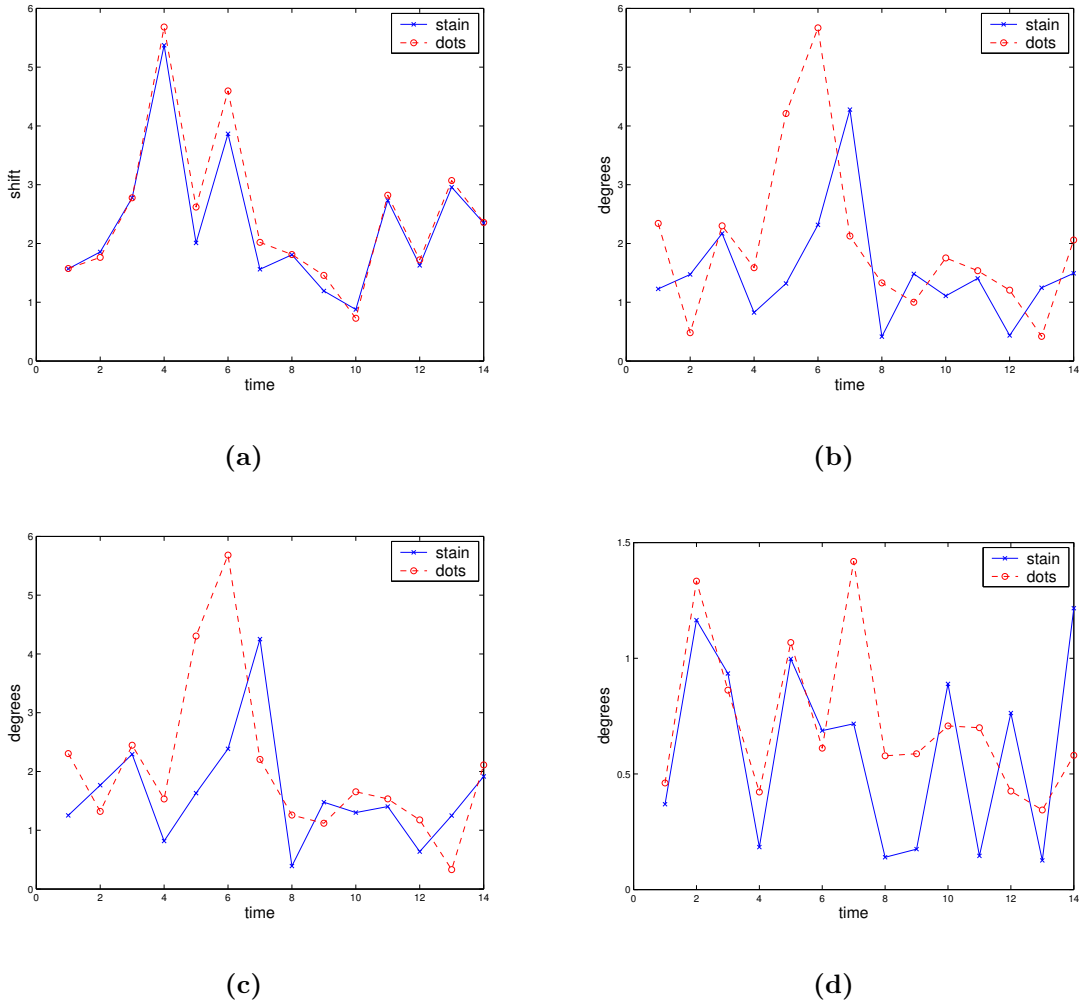


Figure 9.6: Series 3: a) Shift of the center of mass in pixel, b) Rotation of the first eigenvector in degrees between time frames, rotation of the second c) and third d) eigenvector in degrees between time frames.

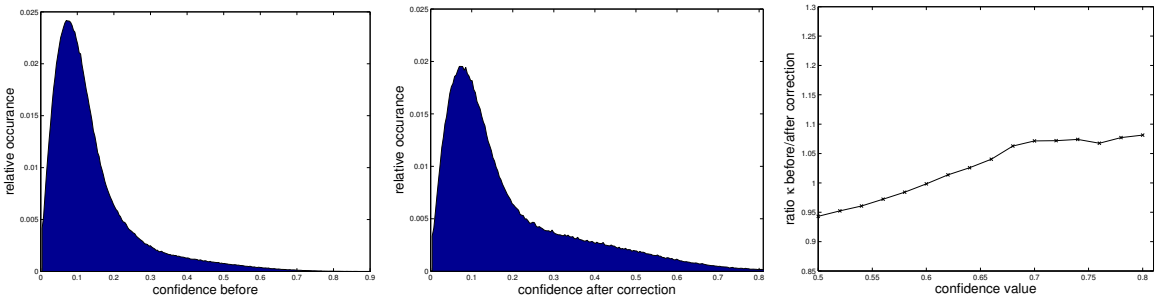


Figure 9.7: Series 3: a) Histogram of c_{1D} before correction, b) after correction, c) $\bar{\kappa}_{before}/\bar{\kappa}_{after}$ as a function of the regions with c_{1D} higher than the threshold.

9.2.5 4D measures

The result of the cell body motion on the spots has been studied as follows: Before and after the motion parameter correction we compute a line-likeness measure in the 4D image of the spots. We use

$$c_{1D} = \frac{l_3 - l_4}{l_3 + l_4} \in [0, 1]\,, \tag{9.16}$$

as a normalized measure, where $l_i \geq l_{i+1} \geq 0$ are the eigenvalues of the gradient structure tensor [10, 60, 56, 61]

$$\bar{G} := \overline{\partial_\alpha I \partial_\beta I}\,,\ \alpha, \beta = 1, \ldots 4 \tag{9.17}$$

applied to the 4D (x, t)-image, see appendix A. Here the overhead bar denotes averaging over a local neighborhood.

In figures 9.3, 9.5 and 9.7 the histograms of the confidence measures are shown before a) and after b) the correction. The peak position of the confidence histogram before and after correction does not change. However, the mean confidence increases about 20% for all series, compare also table 9.2. Furthermore, we investigate the acceleration of the spots. A measure for the force/acceleration a is the curvature of the 4D spot trajectory, compare section 4.6

$$a = |\dot{v}|T + |v|^2 \kappa(s) N\,. \tag{4.24}$$

The curvature is compute by eq.(4.7). In figures 9.3,9.5 and 9.7c) we show the ratio of the average curvature $\bar{\kappa}$ before and after the correction as a function of the regions with a least a certain line-likeness value c_{1D}. The mean curvature before and after correction as a function of the confidence increases only slightly for higher confidences. Therefore, we conclude that the corrected motion is not accelerated. The cell does not roll on the stage as seen from the angles of the third eigenvector and the drift in the water basin seems to be force free.

9.3 Conclusions

We presented a fully automatic algorithm to obtain motion parameters in 3D without establishing correspondence of individual points. The inertia tensor provides a robust intrinsic coordinate system in every time frame. This enable us to calculate the rotation matrix by combining basis transformations to this intrinsic system, instead of using the Euler angle formalism. A measure to detect failure of the rotation correction is presented (9.11). The motion parameters can reliably be computed from the spots. If it is possible the motion parameters of the stain observation

should be used to correct the spot movement as the stain is not influenced by dislocations of labeled material. This automatic correction should be the first step in analyzing the relative motion of individual spots. It enables us to only observe the spots and at the same time to correct for the cell nucleus motion, which is an experimental advantage.

Chapter 10

Conclusions

This thesis is devoted to measurements of local properties, especially orientation and curvature, in multi-dimensional grey-value images and the possibility to compute global shape properties from them. We aimed to derive sampling-error free measures. Therefore, we used grey-value measurements throughout the thesis for orientation and curvature. A measurement value is always accompanied by a confidence value that indicates the trustworthiness of the measurement value. The confidence is computed from the structure present in the image taking into account the model assumptions of the measurement procedure. To unambiguously compute curvature at all image position, we argue that the standard orientation representation is not suitable and show how a known solution to the orientation representation problem can be extended to compute curvature reliably. In the following we fuse all conclusions from the previous chapters into one global conclusion.

In chapter 2 we have reviewed how to map an orientation representation (discontinuity modulo π) to a continuous representation in arbitrary dimensions. We generalized on the findings of Knutsson [63, 64]

$$M(x) = \frac{xx^t}{\|x\|} \tag{2.4}$$

and made a general scheme available to construct a basis (minimal set) from M, necessary to close the orientation representation in arbitrary dimensions. This is advantageous because it reduces the dimensionality of the problem.

Following the approach of sampling-error free measurements in images, we have elaborated on the closed orientation representation using its uniform stretch property and developed a new curvature estimator for lines

$$\kappa = \frac{1}{\sqrt{2}} \left\| \frac{\partial M(T)}{\partial T} \right\| \tag{4.7}$$

and surfaces

$$|\kappa_{1,2}| = \frac{1}{\sqrt{2}} \|\nabla_{v_{2,3}} M(v_1)\| \,. \tag{5.12}$$

The principal tangent directions v_1, v_2 can be found using an eigenvalue analysis of the Gradient Structure Tensor (GST). The curvature estimation formulas work also for ridge-like structures where isophote curvature fails. Furthermore, they are more robust against noise than the GST in parabolic coordinates. The formulas are especially useful for patterns of lines or surfaces. Noise suppression while at the same time computing the curvature on a small local scale is not possible using isophote curvature. Higher noise suppression ultimately smooths out the pattern of interest. The curvature estimator for surfaces is not unbiased, however, the error remains small ($\epsilon < 7\%$ for $\kappa < 0.08$). The estimation is dependent on the magnitude of the curvature and the change of the curvatures in a local neighborhood. The bias is qualitatively understood as a problem in the orientation estimation of the GST leading to biased tangent and normal directions, but no analytic correction term can be given (up to now).

From the principal curvatures, shape descriptors can be computed: the bending energy

$$\mathrm{BE}_{3D} = \int \kappa_1^2 + \kappa_2^2 \, dA \tag{5.7}$$

and the Euler characteristic χ

$$\int\limits_M \kappa_1 \kappa_2 \, dA = 2\pi\chi(M) \,. \tag{5.8}$$

The integration of the curvatures over the object surface is done by replacing the infinitesimal area element dA by the sum of the erf-clipped gradient magnitude [132, 122],

$$dA \leftrightarrow \|\nabla I_{\mathrm{clip}}\| \, dV \,, \tag{3.24}$$

which is an unbiased estimator.

Regarding velocity estimation, we have used optic flow as introduced by Lucas and Kanade [75] for our applications as the involved parameters are given by the knowledge of the structure at hand. A confidence measure can be computed from the line-likeness obtained from the GST, as optic flow models the motion as a local unimodal orientation distribution in (x, t)-space.

The rising behavior of bread dough was studied by optic flow computed on MRI recordings made during rising. Two differently pre-processed doughs were studied. We could demonstrate in which manner the internal stress imposed during kneading on the dough, which is clearly visible in the end result, is preserved during proofing. Isotropic rising could be attributed to fast moulding and anisotropic rising to slow moulding.

Evaporation of ethylene glycol in micron scale wells ("Labs on a chip") has been studied. We could verify quantitatively that the theory of diffusion limited evaporation introduced by Deegan [25, 26] is applicable to these nano liter wells. Furthermore, we could study the entire evaporation process in nearly the whole well. To obtain these results, two image processing problems had to be overcome. The flow field during the evaporation is visualized by introducing fluorescent microspheres in the liquid, which are recorded with a CCD camera mounted on a microscope. First, we observe all microspheres (some unsharp) at all times in a projection, which lead to a varying intensity and shape per microsphere in time. We could compensate for this using adaptive 1D filtering along the motion trajectory without unnecessary blurring [40, 4]. Second, the flow field can only be measured accurately at position where microspheres are present, but even at those locations the measurement can be faulty if the model assumptions of optic flow are not met. We have estimated the line-likeness of the motion trajectories from the GST as a confidence measure. Angular averaging of all measurements in a ring (round wells) and at the same time performing a normalized convolution with the confidence value, has resulted in a dense velocity estimation.

Finally, we have investigated the possibility to compute the affine motion parameters of a cell nucleus drifting on a microscope stage in an in-vivo confocal microscopy application. The computation is based on the labeled material within the nucleus and not on the nucleus itself. Establishing correspondence between individual spots is not necessary in our approach as we compute the moments and axis of inertia based on the spot distribution, which provides an intrinsic coordinate system. This system can be registered very robustly, as the rotation matrix can be obtained from the axis of inertia. The registration performed on the nucleus itself and on the labeled material is working equally well. Practically, this means that we only need to observe the labeled material and still can do a motion correction for the cell nucleus as a whole without the need to use a stain.

Samenvatting

Structuur door middel van beweging in nD beeldanalyse

Dit proefschrift behandelt het meten van lokale eigenschappen in meerdimensionale grijswaarde beelden. Bijzondere aandacht wordt gegeven aan de representatie van oriëntatie en het schatten van de kromming. Vervolgens bestuderen wij de mogelijkheid globale vorm beschrijvende kenmerken te berekenen uit deze lokale eigenschappen. Wij trachten meetmethoden zonder bemonsteringsfout af te leiden. Een dergelijke meetmethode toegepast op een juist bemonsterd beeld levert hetzelfde resultaat op als de bemonstering van de analoge meting (hoofdstuk 3).

Er wordt een nieuw robuuste krommingsschatter voor lijnen (hoofdstuk 4) en voor oppervlakten (hoofdstuk 5) in nD grijswaarde beelden gepresenteerd. Een beter inzicht in, en een veralgemening van een door Knutsson geïntroduceerde continue oriëntatie representatie, stelt ons in staat, de afgeleiden van het oriëntatieveld te nemen, wat leidt tot een nieuwe krommingsschatter. In onze aanpak worden beelden als meerdimensionale functies beschouwd en gaan wij ervan uit dat er in die beelden lokaal slechts één oriëntatie aanwezig is. Er is geen parametrische beschrijving nodig van de objecten. De kromming kan betrouwbaar in dalen en op bergruggen worden berekend, daar waar de conventionele isophoot-krommingsschatter in gebreke blijft. De isophoot-krommingsschatter kan gebieden met een verdwijnende gradiënt niet aan omdat de kromming genormaliseerd wordt met de grootte van de gradiënt. De lokale eigenschap kromming wordt gebruikt om globale eigenschappen als buigingsenergie en het Euler getal te schatten (hoofdstuk 4 en 5). De integratie over een objectoppervlak in een discrete representatie wordt gerealiseerd door het optellen van de gradiënt magnitude in het hele beeld na soft-clipping.

Deze lokale en globale meetmethoden zijn toepasbaar op tijdreeksen: (x, t)-beelden. Hierin komt oriëntatie overeen met snelheid en kromming met versnelling. Een aantal snelheidsschatters worden uiteen gezet (hoofdstuk 6) en voorts wordt het verband tussen spatiële en temporele bemonstering bloot gelegd (hoofdstuk 3).

De ontwikkelde methoden worden in drie toepassingen gebruikt: 1) De beschrijving van het dynamische gedrag van brooddeeg tijdens het rijzen daarvan, 2) een

stromingsveldanalyse van de beweging van micro-bolletjes in nanoliter putten tijdens het verdampen van de vloeistof, waar zowel de vloeistofstroming als de Brownse beweging aan de orde komen, 3) een biologische toepassing waarin met fluorescentie gelabeld chromatine bestudeerd wordt in een levende cel met behulp van een confocale microscoop. We presenteren een stabiele methode om de samengestelde beweging van het chromatine in de cel te scheiden van de cel beweging. Dit maakt een betere studie van de dynamica van chromatine mogelijk.

Appendix A

The gradient structure tensor

The Gradient Structure Tensor (GST) is a tool to analyse local structure in images by its dominant orientation. It was introduced in the late 1980s by Bigün, Granlund, Kass and Witkin [10, 60] and later adopted by Knutsson, Rao and Schunck [64, 98]. The concept is based on the idea that locally only a single dominant orientation is present (the image is locally one-dimensional). A region with only one such orientation is called a *simple neighborhood* [45, 11, 56]. It is defined by the property that it is at least in one direction s_i shift invariant, i.e. for $x \in \mathbb{R}^n$

$$I(x) = I(x+s) \text{ where } s = 0 \text{ expect at coordinates } i\,. \tag{A.1}$$

The i-th components can be omitted as they carry no information. The dimensionality is therefore reduced to $n-i$. The concept of *local* is defined by the scales $\sigma_i, s_i \in [-\sigma_i, \sigma_i]$ for which eq.(A.1) holds. The coordinate directions s_i are linear subspaces of constant grey-value. In the case of more than one local orientation the GST will, in general, not find the correct response in the orientation histogram. Other more advanced methods are needed. A possible solution yields the concept of orientation space in 2D [42] and 3D [33].

The GST is defined as

$$G_{\alpha\beta} := \partial_\alpha I \partial_\beta I^t, \quad \bar{G}_{\alpha\beta} := \overline{\partial_\alpha I \partial_\beta I^t},\ \alpha, \beta = 1 \ldots n\,, \tag{A.2}$$

where the gradient $\partial_\alpha I$ is computed at scale σ_g and the overlining indicates averaging of the individual elements of $\bar{G}$ at scale σ_T. The tensor scale is usually chosen in the range: $3\sigma_g \leq \sigma_T \leq 10\sigma_g$. This suppresses gradients orientation contributions due to noise and yields a smooth, robust orientation output. This is also an example for a multi-scale approach to images. Where σ_g is the scale of the local detail, σ_T is the scale of the local organization. Averaging of gradient information is possible in this representation as discussed in chapter 2.

As a remark, a structure tensor can also be constructed from other than gradient filters. Angular separable quadrature filter can also be used as suggested by

	Gradient	Hessian	Quadrature
2D	2	3	6
3D	3	6	12 [64]
4D	4	10	24 [48]
nD	n	$\frac{n(n+1)}{2}$	

Table A.1: Number of convolutions needed to compute the gradient, Hessian and Quadrature filters per dimension.

Knutsson [63, 64, 42] based on the linear symmetry representation by Bigün and Granlund [10]. Quadrature filters give response for even (ridge) and odd (edge) structures at one time, whereas derivative filters only for odd structures. Here it is not a drawback as luckily the GST approach lets us combine gradient information without cancellation. For pure lines or line patterns the response of the quadrature filters is twice as high as the GST response resulting in a better signal-to-noise ratio. However, the GST has some implementation advantages especially in higher dimensions as the number of angular separable filter increases faster than the number of convolutions needed to compute the gradient. In 3D six quadrature filters are needed [64]. Each filter consists of a pair, therefore we need a total of twelve convolutions. The GST derivatives need three and if we want to incorporate even structures we can make use of second derivatives via the Hessian which are again six convolutions, which is a total of nine convolutions. In table A.1 we summarize the number of convolutions needed to compute the Gradient (odd), the Hessian (even) and the Quadrature filters (even and odd responses). Mapping from the gradient to the structure tensor is straightforward, for the quadrature filters this is more complex.

The GST as friction

Another way to picture the GST is as the local mean squared friction that one would feel rubbing over a smooth washboard where the friction is proportional to the gradient magnitude in a certain orientation $\vec{o}$. This was pointed out by Verbeek [123]. The locality is given by the averaging of G with σ_T. Across the ribs gives a high friction, whereas sliding parallel to them is easy, compare figure A.1. The orientations obtained by the eigenvalue analysis of the GST return the orientation with the least and the highest friction, along and perpendicular to the ribs.

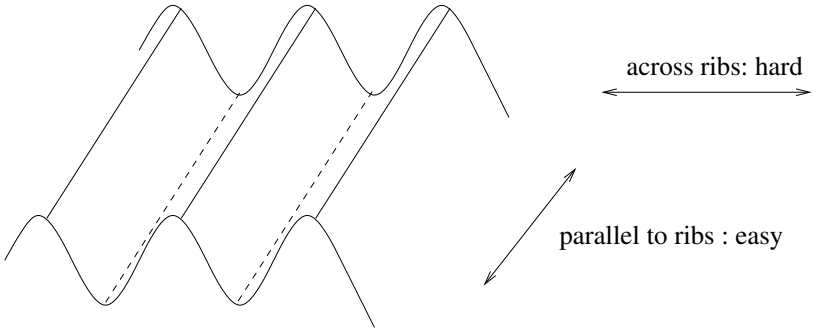


Figure A.1: The GST as the local mean squared friction in orientation o; $\vec{o}^t \bar{G} \vec{o}$.

The GST as the inertia of the gradients

The elements G_{ij} represent gradient energy, and we are interested in the distribution of this energy. The GST is symmetric by construction and semi-positive definite [42] appendix B. Therefore $\bar{G}$ can be diagonalized and the eigenvalues are non-negative, $\lambda_i \geq 0$ [17]. The eigenvectors are the basis of a local coordinate system. This system is much alike the intrinsic body system constructed by the inertia tensor [55, 111]. It is defined as

$$J_{\mu\nu} := \int dx\, \varrho(x) \left(x^2 \delta_{\mu\nu} - x_\mu x_\nu \right), \ \mu, \nu = 1, \ldots n\,, \tag{A.3}$$

in the center of mass system $x_{CM} := \frac{1}{\int dx\, \varrho(x)} \int dx\, \varrho(x) x$ and where $\varrho(x)$ is the mass density. This reads explicitly in 3D

$$\mathbf{J} = \int d^3x\, \varrho(x) \begin{pmatrix} x_2^2 + x_3^2 & -x_1 x_2 & -x_1 x_3 \\ -x_2 x_1 & x_3^2 + x_1^2 & -x_2 x_3 \\ -x_3 x_1 & -x_3 x_2 & x_1^2 + x_2^2 \end{pmatrix}. \tag{A.4}$$

The relation to the GST is

$$J_{ij} = \mathrm{tr}\,(G_{ij}) \delta_{ij} - G_{ij}\,. \tag{A.5}$$

In figure A the gradient vectors at one position are shown, then the eigenvectors of the GST correspond to the two principal directions of the ellipse shown in figure. In this interpretation the inertia of the endpoint of the gradient vectors translated to the a common origin is estimated by the GST. The step from the gradient vectors to the endpoints removes the direction information from the process. The eigenvectors of the GST do not contain direction information but only orientation information. This has to be kept in mind when further processing of the eigenvectors is desired. How to cope with the processing of orientation representation data has been shown and comprehensively discussed in chapter 2. The estimated orientation for symmetric neighborhoods is unbiased, as gradient contributions from one side of

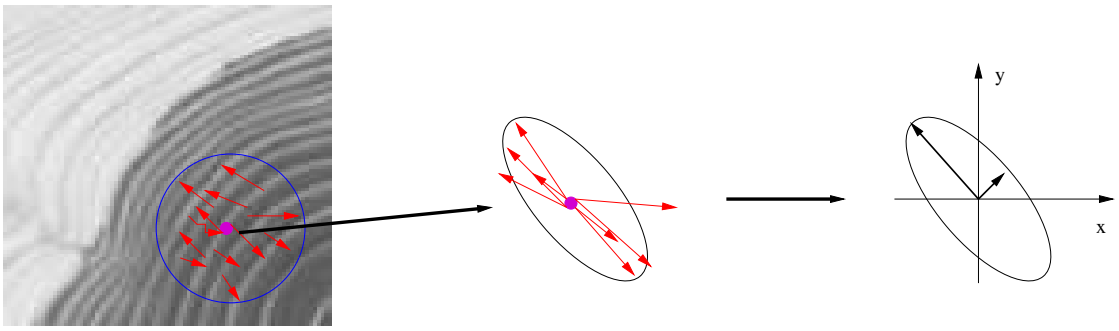


Figure A.2: The gradient structure tensor and the principal directions. The magnitudes of the principal axes are $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$.

the symmetry axis (or symmetry plane) are balanced by their mirrored counterpart. For asymmetric neighborhoods a small bias is introduced. The GST estimates the 'best' orientation (in a least squares sense) [10, 14].

The GST as a projection operator

The GST can also be seen as a projection operator of a vector x onto an orientation axis $\hat{v}$ as pointed out by Verbeek [124]

$$P_{\hat{v}}x := (\hat{v}x)\hat{v} = (\hat{v}\hat{v}^t)x\,. \tag{A.6}$$

The projection operator is idempotent and independent of the sign of $\hat{v}$. The averaged GST $\bar{G}$ maximizes the mean square projection

$$\hat{x}^t\bar{P}_{\hat{v}}\hat{x} = \hat{x}^t\overline{\hat{v}\hat{v}^t}\hat{x} = \overline{\hat{x}^t\hat{v}\hat{v}^t\hat{x}} = \overline{(\hat{x}\hat{v})^2}\,, \tag{A.7}$$

where the projection is maximal when x is the eigenvector of $\bar{G}$ corresponding to the largest eigenvalue. In linear algebra this is known as a optimization of a quadratic form [17].

Confidence measures

The eigenvalues of $\bar{G}$ are a measure of the energy in the corresponding eigen direction. The trace of the GST is the total gradient energy. Thus groups of eigenvalues that have about the same magnitude belong to one invariant subspace. The local dimensionality can therefore be described by the ratio between the eigenvalues. If the signal energy is distributed over all dimensions equally, we encounter $\lambda_i \approx 0, \forall i$, or isotropic noise ($\lambda_i \approx \lambda_{i+1} \neq 0\ \forall i$) or objects. In the following we will only consider objects. Furthermore, we will sort the eigenvalues in decreasing size $\lambda_i \geq \lambda_{i+1} \geq 0$. For 2D and 3D contrast independent measures have been developed. A well-posed

measure c is a continuous function of (a pair of) the eigenvalues, that has a range of zero to one, $c(\{\lambda\}) \in [0, 1]$. The response should be one for the desired subspace and zero for the others. In 2D there is only one linear subspace, a line. Then we have one large and one small eigenvalue, i.e. $\lambda_1 \gg \lambda_2$ and a measure is

$$c_{1D} = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}\,. \tag{A.8}$$

Where $c_{1D} \nearrow 1$ for line like structures. In 3D there are two non-trivial subspaces, 1D line-like structures and 2D plane-like structures.

Subspace	Eigenvalues	Measure	Description
1D	$\lambda_1 \approx \lambda_2 \gg \lambda_3$	$\frac{\lambda_2-\lambda_3}{\lambda_2+\lambda_3}$	line-like structure
2D	$\lambda_1 \gg \lambda_2 \approx \lambda_3$	$\frac{\lambda_1-\lambda_2}{\lambda_1+\lambda_2}$	plane-like structure

In 4D we have four eigenvalues and there are five pairs of eigenvalues. However, by ordering the eigenvalues, in the case of pure integer dimensional subspaces, for the confidence measure it is sufficient to count the number of large and small eigenvalues, i.e. detect the position of "$\gg$". Logical pairs of eigenvalues thus consists of pairs $(\lambda_i, \lambda_{i+1})$:

Subspace	Eigenvalues	Confidence measure
1D	$\lambda_1 \approx \lambda_2 \approx \lambda_3 \gg \lambda_4$	$\frac{\lambda_3-\lambda_4}{\lambda_3+\lambda_4}$
2D	$\lambda_1 \approx \lambda_2 \gg \lambda_3 \approx \lambda_4$	$\frac{\lambda_2-\lambda_3}{\lambda_2+\lambda_3}$
3D	$\lambda_1 \gg \lambda_2 \approx \lambda_3 \approx \lambda_4$	$\frac{\lambda_1-\lambda_2}{\lambda_1+\lambda_2}$

The measure $c_{1D} = \frac{\lambda_3-\lambda_4}{\lambda_3+\lambda_4}$ indicates the weakest line-likeness, through the ordering of the eigenvalues. In another direction the line-likeness could be higher, if you think for example of a line that is blurred in one direction only. Thus a linear combination of $\frac{\lambda_2-\lambda_4}{\lambda_2+\lambda_4}$ and $\frac{\lambda_3-\lambda_4}{\lambda_3+\lambda_4}$ could take into account a possible sharper transition. For the interpretation of a confidence image you have to keep in mind that it is only meaningfully in those regions where (gradient) energy is present, i.e. $\sum \lambda_i$.

One little drawback is the fact that the above mentioned confidence measures over the different subspaces do not add up to one (or another constant like the trace)

$$\sum_i c_{iD} \neq const.\,. \tag{A.9}$$

This indicates that not all possible structures are covered by the here defined c_{iD}. There is no measure for isotropic/pure noise regions for example.

Appendix B

Curvature of space curves

Here we present the mathematics of differential geometry that describe space curves. In 2D, the curvature of a curve in every point describes the shape of this curve completely. In 3D, a second parameter, the torsion, is needed to give a full description, in 4D another and so forth. For a n-dimensional curve we know from the central theorem of space curves, that for given curvatures $\kappa_i, 1 \leq i \leq n-1$ there exists a curve with these κ_i and any two such curves differ only by a translation followed by a rotation (element of SO(n)). The curvatures therefore totally determine the shape of a space curve but give no information about its position. This makes these parameters well suited as curve descriptors. The curvature κ_1 is a first order feature of a nD curve.

Let $I \subseteq \mathbb{R}$ be an interval, then a C^1-mapping $k: I \to \mathbb{R}^n$ is called a parameter curve and $k(I)$ is called a space curve. Let further $t \in I$ be the parameter of k then the tangent $\frac{dk}{dt} =: \dot{k}$ exists for all curves. The arc length $s(t)$ is

$$s(t) := \int_{t_0}^{t} \sqrt{\dot{k}^2}\, d\tilde{t}\,. \tag{B.1}$$

Intermezzo

Motivation for the arc length formula
Let k be a nD space curve and $k(t), t \in [a,b]$ a parametrization. We divide the interval $[a,b]$ in N pieces. Let us now approximate the arc length of k by straight lines connecting consecutive points

$$s = \sum_{i}^{N} \|k(t_i) - k(t_{i+1})\| = \sum_{i}^{N} \sqrt{\sum_{j}^{n} [k_j(t_i) - k_j(t_{i+1})]^2}\,. \tag{B.2}$$

From the mean value theorem we know that there exist $\tilde{t}_i^j$ such that

$$k_j(t_i) - k_j(t_{i+1}) = \dot{k}(\tilde{t}_i^j)(t_i - t_{i+1}),\ t_i < \tilde{t}_i^j < t_{i+1},\ j = 1, 2, \ldots n \tag{B.3}$$

In the transition $N \to \infty$ the sum becomes a Riemann integral, $(t_i - t_{i+1})$ becomes dt and $\tilde{t}_i^j = t$, so

$$s = \int_{t_0}^{t_n} dt \sqrt{\sum_j^n \dot{k}(t)^2 dt}\,. \tag{B.4}$$

This is another notation for

$$s(t) = \int_{t_0}^{t} \|\dot{k}\|\, d\tilde{t} = \int_{t_0}^{t} \sqrt{\dot{k}^2}\, d\tilde{t}\,. \tag{B.5}$$

END INTERMEZZO

If $\frac{ds}{dt} = \sqrt{\dot{k}^2} \neq 0,\ \forall t$, i.e. k is a regular curve then $s \to t$ and $t \to s$ are valid parameter transformations. In the following s will be the arc length and $'$ the derivative with respect to it. In this parametrization we have some favorable properties

$$|k'| = 1, \quad \text{because} \quad k'^2 = \left(\frac{dk}{dt}\frac{dt}{ds}\right)^2 = \left(\dot{k}\frac{1}{\sqrt{\dot{k}^2}}\right)^2 = 1 \tag{B.6}$$

$$k'k'' = 0, \quad \text{because} \quad \frac{d}{ds}\left(k'^2 = 1\right) \Rightarrow 2k'k'' = 0. \tag{B.7}$$

A local orthonormal basis can be constructed for a curve $k : I \to E^n$ if $\dot{k}(t), \ddot{k}(t), \ldots, k^{(n-1)}(t)$ are linearly independent by :

$$v_1(t) = \frac{\dot{k}(t)}{|\dot{k}(t)|} = k'(s), \qquad 2 \leq i \leq n-1 \tag{B.8}$$

$$v_i = \frac{n_i(t)}{|n_i(t)|} \quad \text{with} \quad n_i(t) = k^{(i)}(t) - \sum_{j=1}^{i-1}[v_j(t)k^{(i)}(t)]v_j(t) \tag{B.9}$$

$$v_n \perp v_i \qquad \forall i \leq n-1\,. \tag{B.10}$$

With the above definitions and the same prerequisites as for the orthonormal basis we can write down the Frenet equations. For a parameter curve $k : I \to E^n$ there exist unambiguous numbers $\kappa_1, \ldots, \kappa_{n-1}$ with:

$$v_1' = \kappa_1 v_2 \tag{B.11}$$

$$v_i' = -\kappa_{i-1}v_{i-1} + \kappa_i v_{i+1}, \text{ for } 2 \leq i \leq n-1 \tag{B.12}$$

$$v_n' = -\kappa_{n-1}v_{n-1}. \tag{B.13}$$

For a column vector $V := (v_1, \ldots, v_n)$ one can write $V' = VK$, with

$$K = \begin{pmatrix} 0 & -\kappa_1 & 0 & \cdots & \cdots & 0 \\ \kappa_1 & 0 & -\kappa_2 & \cdots & \cdots & 0 \\ 0 & \kappa_2 & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & -\kappa_{n-1} \\ 0 & 0 & 0 & \cdots & \kappa_{n-1} & 0 \end{pmatrix}. \tag{B.14}$$

Here κ_i is the i-th curvature of the curve and κ_{n-1} is called the torsion τ.

Sketch of the proof
This gives an insight into the symmetric structure of K. After the prerequisites there exists a n dimensional basis $\Rightarrow v_j'$ is a linear combination of $v_1, \ldots v_n \Rightarrow \exists$ a matrix K with $V' = VK$. $v_i v_k = \delta_{ik} \Rightarrow v_i' v_k + v_i v_k' = 0$ with $v_i' = v_l K_i^l \Rightarrow v_l K_i^l v_k + v_i v_l K_k^l = 0 = K_i^k + K_k^i$, i.e. out of the orthonormality of $\{v_i\}$ we get the skew symmetry $K_k^i = -K_i^k$. Through the iterative construction of $\{v_i\}$ we know: $v_j \in [\dot{k}, \ldots k^{(j)}]$, $v_j' \in [\dot{k}, \ldots k^{(j+1)}] = [v_1, \ldots v_{j+1}] \Rightarrow v_j' = v_s K_j^s$ with $K_j^s = 0$ for $s > j+1$ and together with $K_k^i = -K_i^k$ we get $K_l^i = 0$ for $i < l + 1_\square$

In three dimensional space the Frenet formulas are:

$$T'(t) = \kappa N(t) \tag{B.15}$$
$$N'(t) = -\kappa T(t) + \tau B(t) \tag{B.16}$$
$$B'(t) = -\tau N(t), \tag{B.17}$$

where $T(t) \equiv v_1$ is the tangent of the curve, $N \equiv v_2$ the normal and $B(t) \equiv v_3$ the binormal. So the curvature κ is the deviation from lineness and the torsion τ is the deviation from flatness. Another definition for the curvature is therefore $\kappa := \frac{d\theta}{ds}$, where θ is the angle of the curve to a reference vector.
The connection to the known osculating circle with radius R and the curvature being $1/R$ is made by this theorem: a circle touching the curve $k \subset E^n$ in ξ_0 in second order has radius $1/\kappa_1$, middle point $m = \xi_0 + 1/\kappa_1 v_2$ and lies in the plane spanned by v_1 and v_2. Thereby it is unambiguously defined. The planes defined by the three vectors T, N, B in a point $k(s_0)$ are called:

- the *osculating* plane for T, N,
- the *rectifying* plane for T, B and
- the *normal* plane for N, B.

From $v_1' = \kappa_1 v_2$ eq.(B.11) we see immediately how to compute the curvature $\kappa_1 \equiv \kappa$ as $v_1 = k'$ and from the iterative construction of the basis $v_2 = k''/\|k''\|$ eq.(B.9) we obtain

$$\kappa = \|k''\|. \tag{4.2}$$

Here we see that the curvature is always greater or equal to zero. Indeed for space curves it does not make sense to speak of a signed curvature in a coordinate independent description. In contrary to closed surfaces there is no border separating two distinct parts of space. By choosing an origin one can speak of signed curvature also for space curves.

Example: The circular helix, a parametrization is

$$k(t) = (r\cos t, r\sin t, ht),\ t \in \mathbb{R}\,, \tag{B.18}$$

where h is the pitch, r the radius and the helix is winding around the z-axis. With the parametrization of the curve and the arc length factor $\frac{ds}{dt} = \sqrt{r^2+h^2}$ the curvature and torsion can be computed with the help of the Frenet equations as follows:

$$k'(s) = \frac{dk}{dt}\frac{dt}{ds} = \frac{1}{\sqrt{r^2+h^2}}\begin{pmatrix} -r\sin t \\ r\cos t \\ h \end{pmatrix} =: T \tag{B.19}$$

$$k''(s) = \frac{dk'}{dt}\frac{dt}{ds} = \frac{1}{r^2+h^2}\begin{pmatrix} -r\cos t \\ -r\sin t \\ 0 \end{pmatrix} \tag{B.20}$$

$$\kappa := \|k''(s)\| = \frac{r}{r^2+h^2} \tag{4.15}$$

$$N := \frac{k''(s)}{|k''(s)|} = \begin{pmatrix} -\cos t \\ -\sin t \\ 0 \end{pmatrix} \tag{B.21}$$

$$B = T \times N = \frac{1}{\sqrt{r^2+h^2}}\begin{pmatrix} h\sin t \\ -h\cos t \\ r \end{pmatrix} \tag{B.22}$$

$$B' = \frac{dB}{dt}\frac{dt}{ds} = \frac{1}{r^2+h^2}\begin{pmatrix} h\cos t \\ h\sin t \\ 0 \end{pmatrix} \tag{B.23}$$

$$\Rightarrow B' = -\tau N \Rightarrow \tau = \frac{h}{r^2+h^2}\,. \tag{B.24}$$

Example: The curvature of the elliptic helix $e(t) = (a\cos t, b\sin t, ht)$ is easier compute to with the help of the following formula

$$\kappa^2 = \frac{\dot{k}^2\ddot{k}^2 - (\dot{k}\ddot{k})^2}{(\dot{k}^2)^3}\,, \tag{B.25}$$

which can be derived as follows with $ds/dt = \sqrt{\dot{k}^2}$ and $d^2t/ds^2 = \frac{-\dot{k}\ddot{k}}{(\dot{k}^2)^2}$

$$\begin{aligned}
\kappa^2 &= k''^2 && \text{(B.26)}\\
&= \left[\frac{d^2k}{dt^2}\left(\frac{dt}{ds}\right)^2 + \frac{dk}{dt}\frac{d^2t}{ds^2}\right]^2 && \text{(B.27)}\\
&= \left[\ddot{k}\frac{1}{\dot{k}^2} - \dot{k}\frac{\dot{k}\ddot{k}}{(\dot{k}^2)^2}\right]^2 && \text{(B.28)}\\
&= \frac{1}{(\dot{k}^2)^2}\left[\ddot{k}^2 - 2\ddot{k}\dot{k}\frac{\dot{k}\ddot{k}}{\dot{k}^2} + \dot{k}^2\frac{(\dot{k}\ddot{k})^2}{(\dot{k}^2)^2}\right] && \text{(B.29)}\\
&= \frac{\ddot{k}^2\dot{k}^2 - (\ddot{k}\dot{k})^2}{(\dot{k}^2)^3}\ \square && \text{(B.30)}
\end{aligned}$$

Filling in the parametrization of the elliptic helix in eq.(B.25) we obtain

$$\kappa^2 = \frac{a^2b^2(\sin^4 t + \cos^4 t) + 2a^2b^2\sin^2 t\cos^2 t + h^2(a^2\cos^2 t + b^2\sin^2 t)}{(a^2\sin^2 t + b^2\cos^2 t + h^2)^3}\,. \qquad \text{(B.31)}$$

This formula reduces for $h = 0$ to the planar helix

$$\kappa = \frac{ab}{(a^2\sin^2 t + b^2\cos^2 t)^{3/2}} \qquad \text{(B.32)}$$

and for the $a = b = r$ for the circular helix to eq.(4.15).

Appendix C

Curvatures on surfaces

Here the mathematics of differential geometry that describe surfaces in 3D are presented [39, 28]. We have adopted a formulation that can intuitively be transferred to our approach where we deal with grey-value images as in chapter 5. As surfaces in grey-level images can always be viewed as embedded (in the image), we use the formalism of embedded manifolds $M^2 \subset \mathbb{R}^3$. Vectors will be printed bold opposite to coordinates in this section only. In the following we will use the Einstein summation convention, i.e. a summation is implied over any index that appears as lower and upper index in a single term; for example: $\sum_i a_i b^i = a_i b^i$.

Let the surface M^2 be parameterized by u^1, u^2. A curve $\mathbf{x} = \mathbf{x}(t)$ that lies on M^2 can be expressed in the parametrization $u^\alpha = u^\alpha(t)$, so $\mathbf{x} = \mathbf{x}[u(t)]$ and

$$\frac{d\mathbf{x}}{dt} = \mathbf{x}_\alpha \frac{du^\alpha}{dt}, \quad \mathbf{x}_\alpha := \frac{\partial \mathbf{x}}{\partial u^\alpha}, \ \alpha = 1, 2 \tag{C.1}$$

where $\mathbf{x}_\alpha$ form a basis of the tangent space to M^2 at each point. The first fundamental form is then defined as

$$ds^2 = \langle d\mathbf{x}, d\mathbf{x} \rangle = \left\langle \mathbf{x}_\alpha du^\alpha, \mathbf{x}_\beta du^\beta \right\rangle = g_{\alpha\beta} du^\alpha du^\beta \,, \tag{C.2}$$

with the metric tensor $g_{\alpha\beta} = \langle \mathbf{x}_\alpha, \mathbf{x}_\beta \rangle$. A surface area element is then $dA = \sqrt{\det g}\ du^1 \wedge du^2$. Let $\mathbf{N} = \frac{\mathbf{x}_1 \times \mathbf{x}_2}{\|\mathbf{x}_1 \times \mathbf{x}_2\|}$ be the unit normal to M^2 and $\mathbf{X}$ a tangent vector, then the derivative of $\mathbf{N}$ with respect to $\mathbf{X}$ is $d\mathbf{N}/dt = \mathbf{N}_\alpha X^\alpha, \mathbf{N}_\alpha := \partial \mathbf{N}/\partial u^\alpha$. Now if we define a linear transformation

$$b(\mathbf{X}) := -\mathbf{N}_\alpha X^\alpha \,. \tag{C.3}$$

then we can write down the Weingarten equations $b(\mathbf{x}_\alpha) = \mathbf{x}_\alpha b^\alpha_\beta = -\mathbf{N}_\beta$. With b a bilinear form B is associated by $B(\mathbf{X}, \mathbf{Y}) = \langle \mathbf{X}, b(\mathbf{Y}) \rangle = -\left\langle \mathbf{x}_\alpha X^\alpha, \mathbf{N}_\beta Y^\beta \right\rangle$. Now we can formulate the second fundamental form

$$B = b_{\alpha\beta} du^\alpha du^\beta, \quad b_{\alpha\beta} = g_{\alpha\gamma} b^\gamma_\beta \,. \tag{C.4}$$

Form the definition of B we get

$$b_{\alpha\beta} = -\left\langle \mathbf{x}_\alpha, \mathbf{N}_\beta \right\rangle . \tag{C.5}$$

Because $\mathbf{N}$ is a normal vector $\langle \mathbf{x}_\alpha, \mathbf{N} \rangle = 0$, and after differentiating we get another expression for $b_{\alpha\beta} = \langle \mathbf{x}_{\alpha\beta}, \mathbf{N} \rangle$. Here we see that B is symmetric.

Assume we have a space curve k on M^2 given in arclength parametrization by $\mathbf{x}$(s), then the unit tangent is

$$\mathbf{T} = \frac{d\mathbf{x}}{ds} = \mathbf{x}_\alpha \frac{du^\alpha}{ds}, \quad \mathbf{x}_\alpha := \frac{\partial \mathbf{x}}{\partial u^\alpha}, \quad \alpha = 1, 2, \tag{C.6}$$

compare figure C.1. The curvature κ of the space curve is given by the Frenet-

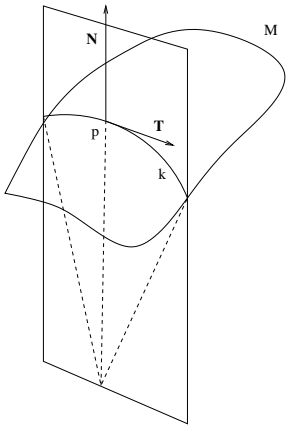


Figure C.1: Curvature on a surface

equations, with $\tilde{\mathbf{N}}$ the unit normal of k, as $\frac{d\mathbf{T}}{ds} = \kappa \tilde{\mathbf{N}}$,

$$\frac{d\mathbf{T}}{ds} = \mathbf{x}_{\alpha\beta} \frac{du^\alpha}{ds} \frac{du^\beta}{ds} + \mathbf{x}_\alpha \frac{d^2 u^\alpha}{ds^2} . \tag{C.7}$$

The projection of $\kappa\tilde{\mathbf{N}}$ along the surface normal $\mathbf{N} = \frac{\mathbf{x}_1 \times \mathbf{x}_2}{\|\mathbf{x}_1 \times \mathbf{x}_2\|}$ is

$$\left\langle \kappa \tilde{\mathbf{N}}, \mathbf{N} \right\rangle = \left\langle \mathbf{x}_{\alpha\beta}, \mathbf{N} \right\rangle \frac{du^\alpha}{ds} \frac{du^\beta}{ds} \tag{C.8}$$

$$= b_{\alpha\beta} \frac{du^\alpha}{ds} \frac{du^\beta}{ds} = B(\mathbf{T}, \mathbf{T}) . \tag{C.9}$$

The curvature is thus a function of the second fundamental B and only dependent on the tangent. The principal curvatures in a point p are now defined as

$$\begin{aligned} \kappa_1(p) &= \max B(\mathbf{T}, \mathbf{T}) \\ \kappa_2(p) &= \min B(\mathbf{T}, \mathbf{T}) . \end{aligned} \tag{C.10}$$

The associated directions $\mathbf{T}_\alpha$ are called principal directions. As mentioned above B is symmetric, so the principal curvatures are the real eigenvalues of

$$b(\mathbf{T}_\alpha) = \kappa_\alpha \mathbf{T}_\alpha \,, \tag{C.11}$$

with corresponding eigenvectors $\mathbf{T}_1, \mathbf{T}_2$ and if the eigenvalues are not degenerated ($\kappa_1 \neq \kappa_2$) then the principal directions are orthogonal. The principal curvature are therefore the magnitude of the change of the normal $\mathbf{N}$ in the principal directions $\mathbf{T}_\alpha$ or short in abstract notation

$$\kappa_\alpha = \|\nabla_{\mathbf{T}_\alpha} \mathbf{N}\| \,. \tag{5.3}$$

Example: The spherical torus, generated by rotating a circle, radius r, around an axis lying in the circle plane outside of the circle, radius R. A parametrization is

$$((R + r\cos v)\cos u, (R + r\cos v)\sin u, r\sin v) \,, \tag{C.12}$$

with $u, v \in [0, 2\pi]$. The principal curvatures can be computed by hand to

$$\kappa_1 = -\frac{\cos v}{R + r\cos v} \text{ and } \kappa_2 = -\frac{1}{r} \,. \tag{C.13}$$

Example: Curvatures of an ellipsoid

$$(a\cos u \sin v, b \sin u \sin v, c\cos v), u \in [0, 2\pi], v \in [0, \pi] \,. \tag{C.14}$$

The principal curvatures are

$$K = \frac{a^2b^2c^2}{[a^2b^2\cos^2 v + c^2\sin^2 v(b^2\cos^2 u + a^2\sin^2 u)]^2} \tag{C.15}$$

$$H = \frac{abc(5(a^2+b^2) + 6c^2 - 3(a^2-b^2)\cos 2u)\sin v}{16[a^2b^2\cos^2 v + c^2\sin^2 v(b^2\cos^2 u + a^2\sin^2 u)]^{3/2}|\sin v|} + \frac{(a^2+b^2-2c^2+(a^2-b^2)\cos 2u)\sin 3v}{16[a^2b^2\cos^2 v + c^2\sin^2 v(b^2\cos^2 u + a^2\sin^2 u)]^{3/2}|\sin v|} \tag{C.16}$$

$$\kappa_{1,2} = H \pm \sqrt{H^2 - K} \tag{C.17}$$

Higher dimensional surfaces

Curvature is an intrinsic property of a surface V^r or in other words: an ant walking on an apple can conduct local measurements and measure the deviation from flatness. Therefore the concept of curvature does not need an embedding space M^n and the curvature of a 2D surface does not change if the embedding space is three or four dimensional.

That being said, the proper treatment of surface properties in higher dimension is substantial and requires mathematical concept and detail that is not within the scope of this thesis. In general the curvature measure of a surface is given by the Riemann curvature tensor or equally by the curvature 2-forms [39, 84]. They can be computed directly from the metric (intrinsic).

Our approach to curvature of surfaces via eq.(5.3) requires tangents to the surface and a normal. In image processing the embedding space is Euclidean $M^n = E^n$. If $r = n - 1$ there is a unique normal and $n - 1$ tangents and the GST provides a method to find them. From them the $n - 1$ principal curvatures can be computed. If $r < n - 1$ the normal space to the surface gives rise to more normal vectors. As the curvatures at a point on the surface is only dependent on the tangent plane any normal vector is sufficient to compute the sectional curvatures.

Appendix D

Diffusion equation

The general diffusion equation with diffusion constant $D(x,t) = D$ is

$$(\partial_t - D\Delta) f(x,t) = \delta(x,t)\,. \tag{D.1}$$

Fourier-transformation along x yields

$$\begin{aligned} \partial_t \hat{f}(\xi,t) + D\xi^2 \hat{f}(\xi,t) &= (2\pi)^{-n/2}\delta(t) && \text{(D.2)} \\ [23] \Rightarrow \hat{f}(\xi,t) &= (2\pi)^{-n/2}\Theta(t)e^{-Dt\xi^2}\,, && \text{(D.3)} \end{aligned}$$

where $\Theta(t)$ is the Heaviside step function

$$\Theta(t) := \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}\,. \tag{6.47}$$

The fundamental solution $f(x,t)$ is then obtained via inverse Fourier transformation

$$\begin{aligned} \Rightarrow f(x,t) &= (2\pi)^{-n/2}\Theta(t)\mathcal{F}_\xi^{-1}\{e^{-Dt\xi^2}\}(x) && \text{(D.4)} \\ &= \cdots \mathcal{F}_\xi\{e^{-Dt\xi^2}\}(-x) && \text{(D.5)} \\ &= \cdots \mathcal{F}_\xi\{e^{-\frac{\xi^2}{2\sigma^2}}\}(-x),\ \sigma^2 = \frac{1}{2Dt} && \text{(D.6)} \\ &= \cdots \sigma^n e^{-\frac{1}{2}\sigma^2 x^2} && \text{(D.7)} \\ &= \frac{\Theta(t)}{(2\sqrt{\pi Dt})^n} e^{-\frac{x^2}{4Dt}}\,. && \text{(D.8)} \end{aligned}$$

The second order moment of the distribution $f(x,t)$ then becomes

$$\left\langle x^2 \right\rangle = \int x^2 f(x,t)\,dx = -(2\pi)^{n/2}\,\frac{d^2\hat{f}(\xi,t)}{d\xi^2}\Big|_{\xi=0} = 2nDt\,. \tag{D.9}$$

Appendix E

Proof of equations (4.7) and (5.12)

Here we show a proof that the proposed curvature estimators for lines eq.(4.7) and surfaces eq.(5.12) are formally equal to the text book expressions eq.(4.2) and eq.(5.3). We assume that the normal and tangent direction are estimated correctly. In practice, however, the GST estimation of these direction is only unbiased for constant curvature neighborhoods. It is to show that

$$\kappa = \|k''\| \quad \longleftrightarrow \quad \kappa = \tfrac{1}{\sqrt{2}} \left\| \tfrac{\partial M(T)}{\partial T} \right\| \tag{4.2} \tag{4.7}$$

$$\kappa = \|\nabla_T N\| \quad \longleftrightarrow \quad \kappa = \tfrac{1}{\sqrt{2}} \|\nabla_T M(N)\| \tag{5.3} \tag{5.12}$$

with $M(v) = vv^t/\|v\|$ the Knutsson mapping. As the proof for the line formula is very similar to the surface formula we will only show the latter. Furthermore, we will use the index notation with summation over same indices, i.e. $\sum_i a_i b^i = a_i b^i$ and assume a normalized normal $\|N\| = N_l N_l = 1$. In this notation eq.(5.3) reads $\kappa^2 = \sum_l [T^i \partial_i N_l]^2$, the Knutson mapping $M(N) = N_l N_k$ and eq.(5.12) becomes

$$2\kappa^2 = \sum_{l,k} [T^i \partial_i N_l N_k]^2 \tag{E.1}$$

$$= \sum_{l,k} N_l^2 (T^i \partial_i N_k)^2 + N_k^2 (T^i \partial_i N_l)^2 + 2 N_l N_k (T^i \partial_i N_k)(T^i \partial_i N_l) \tag{E.2}$$

$$= \underbrace{\sum_l N_l^2}_{=1} \sum_k (T^i \partial_i N_k)^2 + \underbrace{\sum_k N_k^2}_{=1} \sum_l (T^i \partial_i N_l)^2 \tag{E.3}$$

$$+ 2 \sum_l N_l (T^i \partial_i N_l) \sum_k N_k (T^i \partial_i N_k) \tag{E.4}$$

$$= 2\sum_l [T^i \partial_i N_l]^2 + 2\left[\sum_l N_l (T^i \partial_i N_l)\right]^2 \tag{E.5}$$

$$= 2\sum_l [T^i \partial_i N_l]^2 + 2\left[\sum_l T^i \partial_i \frac{1}{2} N_l^2\right]^2 \tag{E.6}$$

$$= 2\sum_l [T^i \partial_i N_l]^2 \quad \square \tag{E.7}$$

Bibliography

[1] M. Aronsson. *On 3D Fibre Measurement of Digitized Paper*. PhD thesis, Swedish University of Agricultural Science, Uppsala, Sweden, 2002. Referred to on p.: 80

[2] H. Bårman, G. H. Granlund, and H. Knutsson. Estimation of curvature in 3-D images using tensor field filtering. In O. Faugeras, editor, *Computer Vision – ECCV90. Proceedings, 1990*, pages 563–565. Springer-Verlag, 1990. Referred to on p.: 64, 78

[3] H. Bårman, L. Haglund, G. H. Granlund, and H. Knutsson. Estimation of velocity, acceleration and disparity in time sequences. In *Proc. Conf. Pattern Recognition ICPR'91*, pages 44–51, 1991. Referred to on p.: 91

[4] P. Bakker. *Image structure analysis for seismic interpretation*. PhD thesis, Delft University of Technology, The Netherlands, 2002. Referred to on p.: 41, 58, 59, 81, 119, 149

[5] P. Bakker, P.W. Verbeek, and L.J. van Vliet. Confidence and curvature estimation of curvilinear structures in 3-D. In *ICCV'01 (Vancouver, Canada)*, volume II, pages 139–144. IEEE, July 9-12 2001. Referred to on p.: 40, 41, 58

[6] P. Barham. *The Science of Cooking*. Springer, Berlin, Heidelberg, New York, 2001. Referred to on p.: 103

[7] J.L. Barron, D.J. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994. Referred to on p.: 85, 92

[8] C.B.J. Bergsma, G.J. Streekstra, A.W.M. Smeulders, and E.M.M. Manders. Velocity estimation of spots in 3d confocal images sequences of living cells. *Cytometry*, 43(4):261–272, 2001. Referred to on p.: 33, 61, 128, 133

[9] M. Bezuijen. Improved sub-pixel shift estimation in 2d gray-value images. Master's thesis, Delft University of Technology, Delft, The Netherlands, September 2003. Referred to on p.: 87

[10] J. Bigün and G.H. Granlund. Optimal orientation detection of lineasr symmetry. In *Proceedings of the IEEE first International Conference on Computer Vision*, pages 433–438, London, June 8-11 1987. IEEE Computer Society Press. Referred to on p.: 9, 11, 14, 15, 20, 41, 67, 145, 153, 154, 156

[11] J. Bigün, G.H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991. Referred to on p.: 9, 11, 90, 153

[12] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, New York, 1998. Referred to on p.: 11

[13] R. van den Boomgaard. Affine invariant deformation curves a tool for shape characterization. *Image and Vision Computing*, 17:375–380, 1999. Referred to on p.: 66, 82

[14] R. van den Boomgaard and J. van de Weijer. Robust estimation of orientation for texture analysis. In *Texture 2002, The 2nd international workshop on texture analysis and synthesis*, pages 135–138, Copenhagen, Denmark, 1 June 2002. Referred to on p.: 156

[15] M. Boutin. Numerically invariant signature curves. *International Journal of Computer Vision*, 40(3):235–248, 2000. Referred to on p.: 40

[16] P. Breton and S.W. Zucker. Shadows and shading flow fields. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition - CVPR'96*, pages 782–789, San Francisco, USA, 1996. Referred to on p.: 39, 44, 63

[17] I.N. Bronstein, K.A. Semendjajew, G. Musiol, and H. Mühlig. *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Thun, Frankfurt (Main), 4th edition, 1999. Referred to on p.: 14, 18, 43, 57, 97, 130, 135, 155, 156

[18] R. Brown. Additional remarks on active molecules. *The Philosophical Magazine and Annals of Philosophy*, 6(34):161–166, 1829. Referred to on p.: 101, 130

[19] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–697, November 1986. Referred to on p.: 119

[20] A. Cavallo. *Four Dimensional Particle Tracking in Biological Dynamic Processes*. PhD thesis, Ruprechts-Karls-Universität, Heidelberg, Germany, June 2002. Referred to on p.: 128

[21] Q. Chen, M. Defrise, and F. Deconinck. Symmetric phase only matched filtering of Fourier-Mellin transforms for image registration and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(12):1156–1168, 1994. Referred to on p.: 137

[22] D. Coeurjolly, S. Miguet, and L. Tougne. Discrete curvature based on osculating circle estimation. In C. Arcelli, L.P. Cordella, and G. Sanniti fi Baja, editors, *Visual Form 2001*, volume 2059 of *Lecture Notes on Computer Science*, pages 303–312, Capri, Italy, May 2001. Springer-Verlag. Referred to on p.: 40, 60

[23] F. Constantinescu. *Distributionen und ihre Anwendung in der Physik*. Teubner, Stuttgart, 1974. Referred to on p.: 169

[24] J.G. Daugman. Pattern and motion vision without laplacian zero-crossings. *Journal of the Optical Society of America A*, 5(7):1142–1148, 1988. Referred to on p.: 85

[25] R.D. Deegan, O. Bakajin, T.F. Dupont, G. Huber, S.R. Nagel, and T.A. Witten. Capillary flow as the cause of ring stains from dried liquid drops. *Nature*, 389:827–829, 1997. Referred to on p.: 110, 111, 149

[26] R.D. Deegan, O. Bakajin, T.F. Dupont, G. Huber, S.R. Nagel, and T.A. Witten. Contact line deposits in an evaporating drop. *Physical Review E*, 62(1):756–765, 2000. Referred to on p.: 110, 111, 112, 113, 117, 125, 149

[27] N. Diehl and H. Burkhardt. Planar motion estimation with a fast converging algorithm. In *Proc. 8th Int. Conf. Pattern Recognition, ICPR'86*, pages 1099–1102, Paris, France, 1989. Referred to on p.: 86

[28] M.P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, New Jersey, 1976. Referred to on p.: 41, 64, 165

[29] L.R. van den Doel and L.J. van Vliet. Temporal phase unwrapping algorithm for dynamic interference pattern analysis in interference-contrast microscopy. *Applied Optics*, 40(25):4487–4500, 2001. Referred to on p.: 10, 109, 112

[30] J.S. Duncan, F.A. Lee, A.W.M. Smeulders, and B.L. Zaret. A bending energy model for measurement of cardiac shape deformaty. *IEEE Transactions on Medical Imaging*, 10(3):307–319, 1991. Referred to on p.: 56, 66

[31] P.M. van Duynhoven, G.M.P. van Kempen, R. van Sluis, B. Rieger, P. Weegels, L.J. van Vliet, and K. Nicolay. Quantitative assessment of gas cell development during the proofing of dough by magnetic resonance imaging and image analysis. *Cereal Chemistry*, 80(4):360–369, 2003. Referred to on p.: 103

[32] A. Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der Physik*, 17:549–560, 1905. Referred to on p.: 101, 116, 126

[33] F.G.A. Faas and L.J. van Vliet. 3d-orientation space; filters and sampling. In *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, June 29 -July 2 2003. Referred to on p.: 153

[34] G. Farnebäck. Fast and Accurate Motion Estimation using Orientation Tensors and Parametric Motion Models. In *Proceedings of 15th International Conference on Pattern Recognition*, volume 1, pages 135–139, Barcelona, Spain, September 2000. IAPR. Referred to on p.: 96

[35] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 363–370, Berlin, June 29 -July 2 2003. Springer-Verlag. Referred to on p.: 86

[36] M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 49(12):3136–3144, 2001. Referred to on p.: 91

[37] D.J. Fleet and A.D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990. Referred to on p.: 85

[38] L.M.J. Florack, W.J. Niessen, and M. Nielsen. The intrinsic structure of optic flow incorporating measurement duality. *International Journal of Computer Vision*, 27(3):263–286, 1998. Referred to on p.: 92

[39] T. Frankel. *The Geometry of Physics*. Cambridge University Press, Cambridge, 1997. Referred to on p.: 41, 61, 64, 165, 168

[40] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, September 1991. Referred to on p.: 119, 149

[41] D. Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers*, 93, Part III(26):429–457, November 1946. Referred to on p.: 22

[42] M. van Ginkel. *Image Analysis using Orientation Space based on Steerable Filters*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2002. digital version available:
`http://www.ph.tn.tudelft.nl/PHDTheses/MvGinkel/thesis_vanginkel.html`.
Referred to on p.: 40, 42, 87, 91, 153, 154, 155

[43] M. van Ginkel, J. van de Weijer, P.W. Verbeek, and L.J. van Vliet. Curvature estimation from orientation fields. In B.K. Ersboll and P. Johansen, editors, *SCIA'99, Proc. 11th Scandinavian Conference on Image Analysis (Kangerlussuaq, Greenland)*, pages 545–551. Pattern Recognition Society of Denmark, Lyngby, June 7-11 1999. Referred to on p.: 11, 12, 39, 40, 42, 44, 64, 73

[44] G.H. Golub and C.F. van Loan. *Matrix Computations*. John Hopkins University Press, Baltimore, 3 edition, 1996. Referred to on p.: 42, 95

[45] G.H. Granlund and H. Knutsson. *Signal processing for computer vision*. Kluwer Academic Publishers, Boston/Dordrecht/London, 1995. Referred to on p.: 9, 11, 119, 120, 153

[46] W. Greiner, L. Neise, and H. Stöcker. *Theoretische Physik, Thermodynamik und statische Mechanik*. Verlag Harri Deutsch, Frankfurt, 2 edition, 1993. Referred to on p.: 101

[47] B.M. ter Haar Romeny, L.M.J. Florack, A.H. Salden, and M.A. Viergever. Higher order differential structure of images. *Image and Vision Computing*, 12(6):317–325, 1994. Referred to on p.: 63, 64, 78

[48] L. Haglund. *Adaptive Multidimensional Filtering*. PhD thesis, Linköping University, Linköping, Sweden, 1992. Referred to on p.: 154

[49] D. Hearn and M.P. Baker. *Computer Graphics*. Prentice Hall, New Jersey, 2 edition, 1997. Referred to on p.: 138

[50] E.C. Hildreth. The computation of the velocity field. *Proceedings of the Royal Society of London B*, 221:189–220, 1984. Referred to on p.: 85

[51] A. Hilton, J. Illingworth, and T. Windeatt. Statistics of surface curvature estimates. *Pattern Recognition*, 28(8):1201–1221, 1995. Referred to on p.: 74

[52] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981. Referred to on p.: 93

[53] D.H. Hubel and T.N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160:106–154, 1962. Referred to on p.: 4

[54] L.A. Iverson and S.W. Zucker. Logical/linear operators for image curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):982–996, October 1995. Referred to on p.: 64, 78

[55] B. Jähne. *Spatio-Temporal Image Processing*. Number 751 in Lecture Notes in Computer Science. Springer-Verlag, Berlin, 1993. Referred to on p.: 21, 61, 85, 92, 94, 155

[56] B. Jähne. *Digital Image Processing*. Springer, 4th edition, 1997. Referred to on p.: 11, 12, 14, 21, 39, 40, 41, 135, 145, 153

[57] A.K. Jain, R.P.W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000. Referred to on p.: 11

[58] E.A. Kaltenbacher and R.C. Hardie. High resolution infrared image reconstruction using multiple, low resolution, aliased frames. In *SPIE Proceedings*, volume 2751, pages 142–152, 1996. Referred to on p.: 86

[59] J. Karlholm. *Local Signal Models for Image Sequence Analysis*. PhD thesis, Linköping University, Linköping, Sweden, 1998. Referred to on p.: 61

[60] M. Kass and A. Witkin. Analyzing oriented patterns. *Computer Vision, Graphics and Image Processing*, 37:362–385, 1987. Referred to on p.: 9, 11, 20, 41, 67, 145, 153

[61] G.M.P. van Kempen, N. van den Brink, L.J. van Vliet, M. van Ginkel, and P.W. Verbeek. The application of a local dimensionality estimator to the analysis of 3d microscopic network structures. In *SCIA'99, Proc. 11th Scandinavian Conference on Image Analysis (Kangerlussuaq, Greenland)*, pages 447–455, June 7-11 1999. Referred to on p.: 12, 41, 42, 119, 145

[62] H. Knutsson. *Filtering and Reconstruction in Image Processing*. PhD thesis, Linköping University, Linköping, Sweden, 1982. Referred to on p.: 91

[63] H. Knutsson. Producing a continuous and distance preserving 5-d vector representation of 3-d orientation. In *IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management*, pages 175–182. Miami Beach, Florida, November 18-20 1985. Referred to on p.: 12, 17, 18, 20, 40, 42, 64, 68, 147, 154

[64] H. Knutsson. Representing local structure using tensors. In *The 6th Scandinavian Conference in Image Analysis*, pages 244–251, Oulu, Finland, June 19-22 1989. Referred to on p.: 12, 13, 18, 20, 41, 42, 91, 147, 153, 154

[65] H. Knutsson and M. Anderson. Loglets: Generalized quadrature and phase for local spatio-temporal structure estimation. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 741–748, Berlin, June 29 -July 2 2003. Springer-Verlag. Referred to on p.: 91

[66] H. Knutsson and C.-F. Westin. Normalized convolution - a technique for filtering incomplete and uncertain data. In K. A. Høgda, B. Braathen, and K. Heia, editors, *SCIA'93, Proceedings of the 8th Scandinavian Conference on*

Image Analysis, volume 2, pages 997–1006, Tromsø Norway, 1993. Norwegian Society for Image Processing and Pattern Recognition. Referred to on p.: 119, 120

[67] J.J. Koenderink and A.J. van Doorn. Receptive field families. *Biological Cybernetics*, 63:291–297, 1990. Referred to on p.: 5

[68] J.J. Koenderink and A.J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557–565, 1992. Referred to on p.: 66

[69] A. Lambacher and P. Fromherz. Fluorescence interference-contrast microscopy on oxidized silicon using a monomolecular dye layer. *Applied Physics A*, 63(3):207–216, 1996. Referred to on p.: 109, 112

[70] K.G. Larkin. Natural demodulation of two-dimensional fringe patterns. II. stationary phase analysis of the spiral phase quadrature transform. *Journal of the Optical Society of America A*, 18(8):1871–1881, August 2001. Referred to on p.: 10, 91

[71] M.S. Lee and G. Medioni. Grouping ., -, →, ?, into regions, curves, and junctions. *Computer Vision and Image Understanding*, 76(1):54–69, October 1999. Referred to on p.: 11

[72] D.R. Lide, editor. *Handbook of Chemistry and Physics*. CRC Press, 60th edition, 1980. Referred to on p.: 114

[73] D.R. Lide, editor. *Handbook of Chemistry and Physics*. CRC Press, 82th edition, 1992. Referred to on p.: 129

[74] T. Lindeberg. On scale selection for differential operators. In *Proc. 8th Scandinavian Conference on Image Analysis (Tromsø Norway)*, pages 857–866, May 1993. Referred to on p.: 96

[75] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In P.J. Hayes, editor, *Proceedings of the Seventh Int. Conf. on Artifical Intelligence*, pages 674–679, Vancouver, B.C., Canada, August 1981. William Kaufmann. Referred to on p.: 86, 93, 95, 105, 118, 119, 148

[76] C.L. Luengo Hendriks, B. Rieger, M. van Ginkel, and L.J. van Vliet. DIPimage: a scientific image processing toolbox for MATLAB, 1999-. Delft University of Technology, `http://www.ph.tn.tudelft.nl/DIPlib`. Referred to on p.: 81

[77] C.L. Luengo Hendriks and L.J. van Vliet. Improving resolution to reduce aliasing in an undersampled image sequence. In M. M. Blouke, G.M. Williams,

N. Sampat, and T. Yeh, editors, *Sensors and Camera Systems for Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications*, volume 3965 of *Proc. SPIE Electronic Imaging 2000 Conf.*, pages 2134–222, San Jose, 2000. Referred to on p.: 86

[78] C.L. Luengo Hendriks and L.J. van Vliet. A rotation-invariant morphology for shape analysis of anisotropic objects and structures. In C. Arcelli, L.P. Cordella, and G. di Baja Sanniti, editors, *Proc. 4th International Workshop on Visual Form*, volume 2059 of *Lecture Notes on Computer Science*, pages 378–387, Capri, Italy, 2001. Springer-Verlag. Referred to on p.: 66, 82

[79] V.K. Madisetti and D.B. Williams, editors. *The Digital Signal Processing Handbook*. IEEE Press/CRC Press, 1997. Referred to on p.: 21

[80] J.B.A. Maintz, P.A. van den Elsen, and M.A. Viergever. Evaluation of ridge seeking operators for multimodality medical image matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):353–365, 1996. Referred to on p.: 63

[81] K.V. Mardia, R.J. Morris, A.N. Walder, and J.J Koenderink. Estimation of torsion. *Journal of Applied Statistics*, 26:373–381, 1999. Referred to on p.: 40

[82] D. Marr. *Vision*. Freeman, San Francisco, USA, 1982. Referred to on p.: 85

[83] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London B*, 207:187–217, 1980. Referred to on p.: 42, 119

[84] C.W. Misner, K.S. Thorne, and J.A. Wheeler. *Gravitation*. W.H. Freeman and Company, New York, 1999. Referred to on p.: 64, 168

[85] O. Monga, N. Ayache, and P.T. Sander. From voxel to intrinsic surface features. *Image and Vision Computing*, 10(6):403–417, 1992. Referred to on p.: 64, 78

[86] H.H. Nagel. Displacement vectors derived from second-order intensity variations in image sequences. *Computer Graphics and Image Processing*, 21:85–117, 1983. Referred to on p.: 92, 93

[87] H.H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields form image sequences. *IEEE Transactions of Pattern Recognition and Machine Intelligence*, 8:168–176, 1986. Referred to on p.: 93

[88] Y. Nakajima, H. Inomata, H. Nogawa, Y. Sato, S. Tamura, K. Okazaki, and S. Torii. Physics-based flow estimation of fluids. *Pattern Recognition*, 36:1203–1212, 2003. Referred to on p.: 96

[89] W.J. Niessen, J.S. Duncan, M. Nielsen, L.M.J. Florack, B.M. ter Haar Romeny, and M.A. Viergever. A mulitscale approach to image sequence analysis. *Computer Vision and Image Understanding*, 65(2):259–268, 1997. Referred to on p.: 96

[90] K. Nordberg, H. Knutsson, and Granlund G. Signal representation using operators. Technical report, Linköping University, 1992. LiTH-ISY-I-1342. Referred to on p.: 39

[91] K. Nordberg, H. Knutsson, and G. Granlund. Local curvature from gradients of the orientation tensor field. Technical report, Linköping University, Linköping, Sweden, 1995. LiTH-ISY-R-1783. Referred to on p.: 40

[92] K. Nummiaro, E. Koller-Meier, and L. van Gool. An adaptive color-based particle filter. *Image and Vision Computing*, 21(1):99–110, 2002. Referred to on p.: 86

[93] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the AIEE*, pages 617–644, 1928. [reprinted in: Proceedings of the IEEE, vol. 90, no. 2, pp. 280-305, February 2002]. Referred to on p.: 22

[94] B.H. Okker. Contour continuation using space variant mathematical morphology. Master's thesis, Delft University of Technology, January 2003. Referred to on p.: 34

[95] A.V. Oppenheim, A.S. Willsky, and I.T. Young. *Signals and Systems*. Prentice-Hall International, London, 1983. Referred to on p.: 21, 24

[96] P. Parent and S.W. Zucker. Trace inference, curvature consistency, and curve detection. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 11(8):823–839, August 1989. Referred to on p.: 78

[97] S.-C. Pei and L.-G. Liou. Using moments to acquire the motion parameters of a deformable object without correspondence. *Image and Vision Computing*, 12(8):475–485, 1994. Referred to on p.: 134

[98] A.R. Rao and B.G. Schunck. Computing oriented texture fields. *CVGIP: Graphical Models and Image Processing*, 53(2):157–185, March 1991. Referred to on p.: 153

[99] Wolfram Research. Mathematica 4, 1998. `www.wolfram.com`. Referred to on p.: 74

[100] B. Rieger, L.R. van den Doel, and L.J. van Vliet. Ring formation in nanoliter cups: Quantitative measurements of flow in micromachined wells. *Physical Review E*, 68(3):036312, 2003. Referred to on p.: 109

[101] B. Rieger, F.J. Timmermans, L.J. van Vliet, and P.W. Verbeek. Curvature estimation of surfaces in 3d grey-value images. In C. Suen R. Kasturi, D. Laurendeau, editor, *ICPR'02, Proc. 16th Int. Conf. on Pattern Recognition*, volume 1. IEEE Computer Society, August 11-15 2002. Referred to on p.: 63

[102] B. Rieger, F.J. Timmermans, L.J. van Vliet, and P.W. Verbeek. On curvature estimation of iso-surfaces in 3d grey-value images and the computation of shape descriptors. *IEEE Transactions on Pattern Recoginition and Machine Intelligence*, October 2003. accepted. Referred to on p.: 63

[103] B. Rieger and L.J. van Vliet. Curvature of n-dimensional space curves in grey-value images. *IEEE Transactions on Image Processing*, 11(7):738–745, 2002. Referred to on p.: 39

[104] B. Rieger and L.J. van Vliet. Representing orientation in n-dimensional spaces. In N. Petkov and M.A. Westenberg, editors, *CAIP'03, 10th International Conference on Computer Analysis of Images and Patterns (Groningen, The Netherlands)*, LNCS 2756, pages 17–24, Berlin, August 25-27 2003. Springer Verlag. Referred to on p.: 9

[105] B. Rieger and L.J. van Vliet. A systematic approach to nd orientation representation. *Image and Vision Computing*, November 2003 2004. accepted. Referred to on p.: 9

[106] B. Rieger, L.J. van Vliet, and P.W. Verbeek. Estimation of curvature based shape properties of surfaces in 3d grey-value images. In J. Bigün and T. Gustavsson, editors, *SCIA'03, Proceedings of the 13th Scandinavian Conference on Image Analysis (Göteborg, Sweden)*, LNCS 2749, pages 262–267, Berlin, June 29 -July 2 2003. Springer-Verlag. Referred to on p.: 63

[107] A. Roberts. Curvature attributes and their application to 3d interpreted horizons. *First Break*, 19.2:85–100, February 2001. Referred to on p.: 81

[108] W. Rudin. *Functional Analysis*. McGraw-Hill, second edition, 1991, 1973. Referred to on p.: 21

[109] D. Rutovitz. Datastructures for operations on digital images. In G.C. Cheng, R.S. Ledley, D.K. Pollock, and A. Rosenfeld, editors, *Pictorial Pattern Recognition*, pages 105–133, Washington, DC, 1968. Thompson. Referred to on p.: 70

[110] A.H. Salden, B.M. ter Haar Romeny, and M.A. Viergever. Differential integral geometry of linear scale-space. *Journal of Mathematical Imaging and Vision*, 9:5–27, 1998. Referred to on p.: 41, 64, 78

[111] F. Scheck. *Mechanics: From Newton's Law to Deterministic Chaos*. Springer, Berlin, 1999. Referred to on p.: 12, 89, 133, 135, 136, 155

[112] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, London, 1982. Referred to on p.: 66, 82

[113] C.E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949. [reprinted in: Proceedings of the IEEE, vol. 86, no. 2, pp. 447-457, February 1998]. Referred to on p.: 22

[114] I.A. Shevelev. Second-order features extraction in the cat visual cortex: selective and invariant sensitivity of neurons to the shape and orientation of crosses and corners. *BioSystems*, 48:195–204, 1998. Referred to on p.: 4

[115] A. Singh. *Optic Flow Computation: A Unified Perspective*. IEEE Computer Society Press, 1992. Referred to on p.: 85

[116] M. Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 2. Publish or Perish, Berkley, 1979. Referred to on p.: 41, 64

[117] B. Srinivasa Reddy and B.N. Chatterji. A fft-based technique for translation, rotation and scale-invariant image registration. *IEEE Transactions on Image Processing*, 5(8):1266–1271, 1996. Referred to on p.: 137

[118] H. Stöcker. *Taschenbuch der Physik*. Verlag Harri Deutsch, Frankfurt (Main), 4 edition, 2000. Referred to on p.: 56

[119] Texas Instruments Technology. www.dlp.com. Referred to on p.: 36

[120] J.P. Thirion and A. Gourdon. Computing the differential characteristics of isointensity surfaces. *Computer Vision and Image Understanding*, 61(2):190–202, 1995. Referred to on p.: 63, 64, 65, 78, 79, 81, 83

[121] O. Tretiak and L. Pastor. Velocity estimation for mimage sequences with second order differential operators. In *Int. Conf. on Pattern Recognition (Montreal, Canada)*, pages 20–22. IEEE Computer Society Press, Los Alamitos, USA, 1984. Referred to on p.: 92

[122] P.W. Verbeek. A class of sampling-error free measures in oversampled bandlimited images. *Pattern Recognition Letters*, 3:287–292, 1985. Referred to on p.: 26, 27, 28, 39, 56, 63, 70, 148

[123] P.W. Verbeek. The GST as local mean squared friction. private communications, 1993. Referred to on p.: 11, 154

[124] P.W. Verbeek. private communications, 1999. The GST as a projection operator. Referred to on p.: 156

[125] P.W. Verbeek and J. Dijk. The D-dimensional inverse vector-gradient operator and its application for scale free image enhancement. In N. Petkov and M.A. Westenberg, editors, *10th International Conference on Computer Analysis of Images and Patterns (Groningen, The Netherlands)*, LNCS 2756, pages 738–745, Berlin, August 25-27 2003. Springer. Referred to on p.: 70, 91

[126] P.W. Verbeek, L.J. van Vliet, and J. van de Weijer. Improved curvature and anisotropy estimation for curved line bundles. In A.K. Jain, S. Venkatesh, and B.C. Lovell, editors, *ICPR'98, Proc. 14th Int. Conference on Pattern Recognition*, pages 528–533. IEEE Computer Society Press, Los Alamitos, 17-20 August 1998. Referred to on p.: 40, 58

[127] P.W. Verbeek and B.J.H Verwer. Shading from shape, the eikonal equation solved by grey-weighted distance transform. *Pattern Recognition Letters*, 11:681–690, 1990. Referred to on p.: 70

[128] P.W. Verbeek and L.J. van Vliet. An estimator of edge length and surface area in digitized 2D and 3D images. In *Proc. 11th IAPR Int. Conf. on Pattern Recognition*, volume 3, pages 1–5. IEEE Computer Society Press, 1992. Referred to on p.: 27

[129] P.W. Verbeek and L.J. van Vliet. Estimators of 2D edge length and position, 3D surface area and position in sampled grey-valued images. *BioImaging*, 1(1):46–61, 1993. Referred to on p.: 27, 69, 70

[130] P.W. Verbeek and L.J. van Vliet. On the location error of curved edges in low-pass filtered 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(7):726–733, 1994. Referred to on p.: 70

[131] P.W. Verbeek, H.A. Vrooman, and L.J. van Vliet. Low-level image processing by max-min filters. *Signal Processing*, 15(3):249–258, 1988. Referred to on p.: 128

[132] L.J. van Vliet. *Grey-Scale Measurements in Multi-Dimensional Digitized Images*. PhD thesis, Delft University of Technology, Delft, The Netherlands, October 1993. Referred to on p.: 22, 25, 27, 28, 34, 40, 56, 57, 64, 69, 71, 95, 148

[133] L.J. van Vliet and C.L. Luengo Hendriks. Improving spatial resolution in exchange of temporal resolution in aliased image sequence. In B.K. Ersboll and P. Johansen, editors, *Proc. 11th Scandinavian Conference on Image Analysis*, pages 493–499, Lyngby, June 7-11), June 7-11 1999. Pattern Recognition Society of Denmark. Referred to on p.: 86

[134] L.J. van Vliet and B. Rieger. private communications, May 2003. Referred to on p.: 34

[135] L.J. van Vliet and P.W. Verbeek. Curvature and bending energy in digitized 2d and 3d images. In *8th Scandinavian Conference on Image Analysis, Tromsø Norway*, pages 1403–1410, 1993. Referred to on p.: 27, 29, 39, 41, 44, 45, 63, 64, 66, 78

[136] L.J. van Vliet and P.W. Verbeek. Better geometric measurements based on photometric information. In *Proc. IEEE Instrumentation and Measurement Technology Conf. IMTC94*, pages 1357–1360, 1994. Referred to on p.: 45

[137] H.A. Vrooman. *Quantitative analysis of interferograms.* PhD thesis, Delft University of Technology, Delft, the Netherlands, 1991. Referred to on p.: 10

[138] J. Weickert and C. Schörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245 – 255, 2001. Referred to on p.: 93

[139] J. Weickert, B.M. ter Haar Romeny, and M.A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410, 1998. Referred to on p.: 70

[140] J. van de Weijer, L.J. van Vliet, P.W. Verbeek, and M. van Ginkel. Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):1035 –1042, 2001. Referred to on p.: 12, 40, 58, 63

[141] J. Westerweel. *Digital particle image velocimetry - theory and application.* PhD thesis, Delft University of Technology, Delft, The Netherlands, 1993. Referred to on p.: 88, 89

[142] J. Westerweel. Fundamentals of digital particle image velocimetry. *Measurement Science and Technology*, 8:1379–1392, 1997. Referred to on p.: 88, 117

[143] J. Westerweel. Theoretical analysis of the measurement precision in particle image velocimetry. *Experiments in Fluids*, 29(7):S3–S12, 2000. Referred to on p.: 88

[144] C.F. Westin. *A Tensor Framework for Multidimensional Signal Processing.* PhD thesis, Linköping University, Linköping, Sweden, 1994. Referred to on p.: 120

[145] C.E. Willert and M. Gharib. Digital particle image velocimetry. *Experiments in Fluids*, 10:181–193, 1991. Referred to on p.: 118

[146] M. Worring and A.W.M. Smeulders. The accuracy and precision of curvature estimation methods. In *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, pages 139–142, The Hague, The Netherlands, 1992. IEEE Computer Society Press. Referred to on p.: 39

[147] M. Worring and A.W.M. Smeulders. Digital curvature estimation. *CVGIP: Image Understanding*, 58(3):336–382, 1993. Referred to on p.: 39, 60

[148] I.T. Young, J.E. Walker, and J.E. Bowie. An analysis technique for biological shape I. *Information and Control*, 25(4):357–370, August 1974. Referred to on p.: 56, 66

[149] I.T. Young, R. Zagers, L.J. van Vliet, J. Mullikin, F.R. Boddeke, and H. Netten. Depth-of-focus in microscopy. In K. A. Høgda, B. Braathen, and K. Heia, editors, *SCIA'93, Proceedings of the 8th Scandinavian Conference on Image Analysis*, pages 493–498, Tromsø Norway, May 25-28 1993. Norwegian Society for Image Processing and Pattern Recognition. Referred to on p.: 117

[150] A.L. Yuille and T. Poggio. Scaling theorems for zero crossings. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 8(1):15–25, January 1986. Referred to on p.: 85

[151] S. di Zenzo. A note on the gradient of a multi-image. *Computer Vision, Graphics and Image Processing*, 33:116–125, 1986. Referred to on p.: 41, 67

[152] C. Zetzsche and E. Barth. Direct detection of flow discontinuities by 3d curvature operators. *Pattern Recognition Letters*, 12:771–779, 1991. Referred to on p.: 96

Curriculum vitae

Bernd Rieger was born in Landsberg am Lech, Germany, on the 3^{th} of July, 1973. In 1992 he graduated from the Dominikus-Zimmermann-Gymnasium in Landsberg am Lech, where he started his community service at the Sozialstation St. Martin in the same year. In the fall of 1993 he took up a study in physics at the Technische Universität München, Germany and obtained his Masters Degree in 1999 from the theoretical bio and molecule physics group under the supervision of Prof.Dr. Fischer on the study of vibration relaxation of carbon monoxide on metal surfaces. During his studies, he visited the Royal Institute of Technology, KTH, in Stockholm, Sweden from August 1995 to July 1996 on an Erasmus exchange program.

In October 1999 he started a PhD at the Pattern Recognition Group of the Delft University of Technology, The Netherlands on a project called '4D grey-level image processing' under the supervision of Prof.dr.ir L.J. van Vliet. During his PhD he worked for six month on a project in collaboration with Unilever Research, Vlaardingen, The Netherlands, on time-dependent structural changes in low water foods.

In January 2004 he joint the Max-Plank-Institute for Bio-physical Chemistry, Department of Molecular Biology in Göttingen, Germany as a postdoctoral researcher.

Es gibt nur eine Ausflucht vor der Arbeit: Andere für sich arbeiten zu lassen.
Immanuel Kant, 1724-1804

Acknowledgments

Credit to whom it belongs.

Lucas van Vliet for being a promoter like everybody would want one. Piet Verbeek for having more ideas than I could handle. Michael van Ginkel for helping me with whatever was necessary and for being a friend. The same is true for Cris Luengo Hendriks who made DIPlib usable.

Geert van Kempen, John van Duynhoven and Gerard van Dalen from Unilever Research, Vlaardingen for a very nice project together and to get a taste of the real world. Partners in the 4D NWO project: Chris Molenaar (LUMC) and Erik Manders (UvA) for a good collaboration and useful data sets.

Peter Bakker, Judith Dijk, Tuan Pham, Frank Faas and Kees van Wijk for creating a good atmosphere in the room.

Richard van den Doel for bringing physics in my thesis. Dick de Ridder for a never ending desire to go for a beer. The rest of PH AIO for being there. Ted Young for being the boss. Heidi Dietrich and Yuval Garini for a nice day behind a microscope.

The support staff Ronald Ligteringen, Klara Schaafsma and Wim van Oel for being supportive.

My sister Sabina Rieger for reading the introduction, conclusion and summary and freeing it from mistakes in the English language.

Manon.