7

Nonlinear analysis of multi-dimensional signals: local adaptive estimation of complex motion and orientation patterns

Christoph S. Garbe[1], Kai Krajsek[2], Pavel Pavlov[1], Björn Andres[1], Matthias Mühlich[2,4], Ingo Stuke[6], Cicero Mota[2,5*], Martin Böhme[5], Martin Haker[5], Tobias Schuchert[3], Hanno Scharr[3], Til Aach[4], Erhardt Barth[5], Rudolf Mester[2], and Bernd Jähne[1]

[1] Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg, 69120 Heidelberg, Germany
[2] Visual Sensorics and Information Processing Lab (VSI), Goethe University Frankfurt, 60054 Frankfurt/M., Germany
[3] Institute for Chemistry and Dynamics of the Geosphere, Institute 3: Phytosphere, Research Centre Jülich GmbH, 52425 Jülich, Germany
[4] Institute of Imaging & Computer Vision, RWTH Aachen University, 52056 Aachen, Germany
[5] University of Lübeck, Institute for Neuro- and Bioinformatics, 23538 Lübeck, Germany
[6] University of Lübeck, Institute for Signal Processing, 23538 Lübeck, Germany

Summary. This chapter presents a new framework for the detection and accurate quantification of motion, orientation, and symmetry in images and image sequences. It focuses on those aspects of motion and orientation that previously could not be handled successfully and reliably by existing methods, for example motion superposition (due to transparency, reflection or occlusion), illumination changes, temporal and/or spatial motion discontinuities, and dispersive non-rigid motion. The solutions to these problems significantly advance the applicability of image sequence processing techniques in technical, medical, and scientific application areas. The novel algorithms presented here result from the combination of modern statistical signal processing, differential geometric analysis, novel estimation techniques, and nonlinear adaptive filtering and regularization techniques. The performance of these algorithms is characterized and their applicability is demonstrated by several key application areas including environmental physics, geology, botany, medical imaging, and technical applications such as microfluidics and quality control.

[*] *Present address:* Federal University of Amazonas, 69077-00 Manaus AM, Brazil

7.1 Introduction

We consider the general task of accurately detecting and quantifying orientations in n-dimensional signals $\boldsymbol{s}$. The main emphasis will be placed on the estimation of motion, which can be thought of as orientation in spatio-temporal signals. Associated problems such as the optimization of matched kernels for deriving isotropic and highly accurate gradients from the signals, optimal integration of local models, and local model selection will also be addressed.

Many apparently different approaches to get a quantitative hold on motion have been proposed and explored, e.g. using the brightness constancy constraint [48, 89], the structure tensor [42, 49], blockwise matching or correlation [9, 10], quadrature filters [22, 45, 51], steerable filters and other filter-based approaches [18, 111], projection on polynomial subspaces [20], autocovariance based analysis [66], and many variants of these approaches. As mentioned previously, the estimation of motion is closely related to the estimation of orientation or linear symmetry in 2D images [11, 39] and the computation of curvature [126, 133] in 2D or higher dimensional spaces.

It is relatively well understood, how these various approaches can be employed in the case of simple motion patterns, which can – at least on a local scale – be approximated by a single rigid translational motion. This applies also to moderate amounts of noise, especially if the noise process is stationary, additive, and its amplitude distribution is unimodal.

However, most of the aforementioned approaches show severe limitations in case of complex motion patterns or strong disturbances, which are characteristic for real-life image data. Problematic situations occur for instance in case of motion superposition (due to transparency or specular reflection), temporal and/or spatial motion discontinuities, and spatio-temporal flow effects (relaxation, diffusion, etc.). It is this area, where the current challenges in motion analysis are found, in which significant contributions will be made throughout this chapter. Improved algorithms will necessarily be locally adaptive, nonlinear and based on modern signal processing tools, such as Total Least Squares, anisotropic diffusion, and Markov Random fields, or extend classical signal theory such as those presented by [22, 123].

An overview of the problem of multiple motions has been given in [12] and robust methods for multiple motions have been proposed. The problem of two motions has been first solved by Shizawa and Mase [109, 110]. Their approach is based on Fourier methods and on solving a six-dimensional eigensystem that limits the method to only two motions. Here we will show how to avoid such an eigensystem in case of one motion resulting in a simpler and faster solution for multiple motions. Important contributions in characterizing the spectral properties of multiple motions have been made [134]. In dealing with

the problem of multiple motions, the useful and intuitive notions of 'nulling filters' and 'layers' have been introduced [16, 127]. Their approach is more general in that it treats the separation of motions into layers, but is also limited to the use of a discrete set of possible motions and a probabilistic procedure for finding the most likely motions out of the set.

The problems mentioned before are dealing with characteristics of the signal itself; but additionally we find several kinds of disturbances in real life image data. For instance: rain, snow or low illumination conditions cause different types of noise. Besides that, outdoor scenes often are afflicted by illumination changes, or specular reflections. Additionally, many scientific or medical applications acquire images at the limit of what is physically and technically feasible. This often introduces strong disturbances such as heavy signal-dependent noise.

As will be outlined, the estimation of complex orientation from image data represents an inverse problem. Paramount to solving this problem are adequate models. In section 7.2 a number of such extended models will be presented, ranging from those incorporating brightness changes along given orientations over those incorporating multiple orientations to those deducing scene depth in motion models. The models constrain orientation to image intensities. Still, generally more unknowns are sought than can be fully recovered from these presented constraint equations. In section 7.3 a number of approaches will be presented that make it feasible to derive the sought parameters by introducing additional constraints. Refined estimators will be presented that take statistics into account to perform maximum likelihood estimates. The presented algorithms are based on differential orientation models, relying on an accurate extraction of gradients from image intensities. In section 7.4 schemes for computing these gradients from optimized filters will be presented. Closely related to estimating the orientation parameter of the introduced models is the task of selecting the correct model, given a noise level by which the image data is corrupted. Two such approaches are presented in Section 7.5. The correct model that can be retrieved from the image data will also depend on the signal structures. These structures can be categorized by their intrinsic dimension, a concept which is also introduced in this section. The inverse problem of estimating model parameters is performed by an optimal regularization. In section 7.6 different regularization schemes are presented, that preserve anisotropies in the image signal. These approaches are also used for suppressing noise prior to performing the estimation, leading to optimum results. The problem of orientation estimation in multiple subspaces is a prominent one, that has significant impact, both on modeling and estimation, but for applications as well. Algorithms tackling this problem are detailed in Section 7.7. The developed algorithms make novel applications feasible. In section 7.8 some exemplary applications are presented stemming from different fields such as environmental-, geo-, and life sciences. This chapter closes with some concluding remarks in section 7.9.

7.2 Modeling multi-dimensional signals

7.2.1 Motion and subspace models

We regard n-dimensional signals $s(\boldsymbol{x})$ defined over a region Ω, e.g. images and image sequences. Motions (translations) and orientations correspond to linear d-dimensional subspaces E of Ω with $1 \leq d < n$, such that

$$s(\boldsymbol{x}) = s(\boldsymbol{x} + k\boldsymbol{u}) \quad \forall k \in \mathbb{R} \text{ and } \forall\, \boldsymbol{x}, \boldsymbol{x} + k\boldsymbol{u} \in \Omega \text{ and } \boldsymbol{u} \in E. \tag{7.1}$$

Often one needs to (i) detect the existence of such a subspace E and (ii) estimate the parameter vector $k\boldsymbol{u}$, which corresponds (not always, see Sec. 7.3.2) to the direction of motion in the regarded volume. The values of s can be scalar as in gray-level images or vector valued (denoted $\boldsymbol{s}$) as in color or multispectral images. The estimation is often based on the fact that constancy of the signal in a certain direction in Ω, such as it is reflected in Eq. (7.1) implies linear differential constraints such as the classical *brightness constancy constraint equation (BCCE)*

$$\frac{\partial s}{\partial \boldsymbol{u}} \;=\; 0 \text{ for all } \boldsymbol{u} \in E\,. \tag{7.2}$$

This is the simplest special case of general partial differential equations which result from applying a suitable differential operator $\alpha(\boldsymbol{u})$ on the signal:

$$\alpha(\boldsymbol{u}) \circ s = 0 \tag{7.3}$$

and we will learn about more sophisticated operators $\alpha(\boldsymbol{u})$ later in this contribution. Assuming the constancy of a moving brightness pattern, motions can be interpreted as local orientations in spatio-temporal signals ($n = 3, d = 1$). Many motion models are based on Taylor expansions of Eq. (7.1) (cf. e.g. [23]). Writing Eq. (7.1) with time t as individual parameter, we obtain

$$s(\boldsymbol{x}(t), t) = s(\boldsymbol{x} + \boldsymbol{u}(x,t)\Delta t, t + \Delta t) \tag{7.4}$$

where s is interpreted as the constant brightness signal produced by a spatial point $\boldsymbol{x}(t)$ moving in time t. First order approximation of Eq. (7.4) yields

$$\frac{\mathrm{d}s}{\mathrm{d}t} = 0 \quad \Leftrightarrow \quad \nabla^T_{(\boldsymbol{x})} s \cdot \boldsymbol{u} + \frac{\partial s}{\partial t} = 0 \quad \Leftrightarrow \quad \nabla^T_{(\boldsymbol{x},t)} s \cdot (\boldsymbol{u}^T, 1)^T = 0 \tag{7.5}$$

where ∇ is the gradient operator with respect to parameters given as indices and the general differential operator $\alpha(\boldsymbol{u})$ from Eq. (7.3) takes the form $\alpha(\boldsymbol{u}) := \nabla^T_{(\boldsymbol{x})}\boldsymbol{u} + \frac{\partial}{\partial t}$. Being based on derivatives, such models are called *differential models*. One may further model the motion field $\boldsymbol{u}(x,t)$ locally by applying a Taylor expansion

$$\nabla_{\boldsymbol{x}}^T s(\boldsymbol{u} + \boldsymbol{A}\Delta\boldsymbol{x}) + \frac{\partial s}{\partial t} = 0 \tag{7.6}$$

where the matrix $\boldsymbol{A} = \nabla_{\boldsymbol{x}}\boldsymbol{u}^T$ contains the spatial derivatives of $\boldsymbol{u}$, and $\Delta\boldsymbol{x} = \boldsymbol{x}_0 - \boldsymbol{x}$ are local coordinates. This is called an *affine motion model*. These and other motion models, i.e. parametrizations of $\boldsymbol{u}$, can be found e.g. in [23].

7.2.2 Multiple motions and orientations

In case of multiple motions and orientations, we are dealing with a (linear, multiplicative, or occluding) superposition of subspaces as defined in Eq. (7.1). In case of two subspaces, and additive superposition, one has

$$s(\boldsymbol{x}) = s_1(\boldsymbol{x} + k\boldsymbol{u}_1) + s_2(\boldsymbol{x} + k\boldsymbol{u}_2). \tag{7.7}$$

The model for multiplicative superposition is simply

$$s(\boldsymbol{x}) = s_1(\boldsymbol{x} + k\boldsymbol{u}_1) \cdot s_2(\boldsymbol{x} + k\boldsymbol{u}_2), \tag{7.8}$$

which can be transformed into Eq. (7.7) by taking the logarithm. The model for occluded superposition is

$$s(\boldsymbol{x}) = \chi s_1(\boldsymbol{x} + k\boldsymbol{u}_1) + (1 - \chi)s_2(\boldsymbol{x} + k\boldsymbol{u}_2), \tag{7.9}$$

where $\chi(\boldsymbol{x})$ is the characteristic function that defines the occlusion. As we shall see in Sec. 7.7, the constraint in Eq. (7.2), and some novel mathematical tricks, can be used to detect and estimate multiple motions and orientations.

7.2.3 Dynamic 3D reconstruction

Since the early work of [47] optical flow has been used for disparity and therefore 3D structure estimation. For disparity estimation time t in Eq. (7.4) is replaced by camera displacement r. Therefore time and camera displacement may be considered equivalent as "time-like" parameters in a data set acquired by a multiple camera setup . Combining equations with N time-like parameters sets the dimension d of subspace E to $d = N$ because brightness constancy applies for each of these parameters. We therefore get a 3-dimensional solution space when using time t and 2 camera displacement directions r_1 and r_2. Following Eq. (7.2) we achieve 3D position and 3D motion depending on the motion field either of the object or the camera. Using Eq. (7.4) we can determine motion of an object whereas replacing time t in this equation with camera position r, called structure from motion, yields disparity. Combining these estimations yields a higher dimensional solution space and the problem of determining the parameters in this space (see [96] and [99]). Still assuming

constant brightness now in time and in camera displacement direction we get according to Eq. (7.5)

$$\frac{\partial s}{\partial u} = 0 \quad \Leftrightarrow \quad \nabla^T s\boldsymbol{u} = 0 \quad \text{with} \quad \boldsymbol{u} = (\mathrm{d}x, \mathrm{d}y, \mathrm{d}r_1, \mathrm{d}r_2, \mathrm{d}t)^T \tag{7.10}$$

Parameters like optical flow (u_1, u_2) or disparity ν are then obtained by combination of subspace solutions, e.g.

$$u_1 = \left.\frac{\mathrm{d}x}{\mathrm{d}t}\right|_{\mathrm{d}r_1,\mathrm{d}r_2=0}, u_2 = \left.\frac{\mathrm{d}y}{\mathrm{d}t}\right|_{\mathrm{d}r_1,\mathrm{d}r_2=0}, \nu_1 = \left.\frac{\mathrm{d}x}{\mathrm{d}r_1}\right|_{\mathrm{d}t,\mathrm{d}r_2=0}, \nu_2 = \left.\frac{\mathrm{d}y}{\mathrm{d}r_2}\right|_{\mathrm{d}t,\mathrm{d}r_1=0} \tag{7.11}$$

where ν_1 and ν_2 are dependent estimates of ν which can be transformed into 2 independent estimates of ν (see Sec. 7.3.5). Further modeling the data by applying a surface patch model extends the above orientation model of Eq. (7.10). In the same way, standard optical flow can be extended to affine optical flow (cf. Eqs. (7.5) and (7.6)).

$$\nabla_{\boldsymbol{x}}^T s(\boldsymbol{u} + \boldsymbol{A}\Delta\boldsymbol{x}) + \sum_{i=1}^{2} \frac{\partial s}{\partial r_i}\mathrm{d}r_i + \frac{\partial s}{\partial t}\mathrm{d}t = 0 \tag{7.12}$$

where u contains parameters for motion and disparity, matrix $\boldsymbol{A}$ parameters for depth motion and surface slopes, and $\Delta\boldsymbol{x} = \boldsymbol{x}_0 - \boldsymbol{x}$ are local coordinates. A detailed derivation of the model is proposed in [96] for special cases and in [99] for the general model.

Comparison of 3D reconstruction models

In [96] a detailed error analysis of special cases of the model, i.e. 1D camera grid without normals with and without z-motion, 2D camera grid without any motion with and without normals, is presented. The analysis shows that with higher z-motion the errors due to linearization increase and that estimating the surface normals within this framework reduces systematic errors. Further an error analysis was done for the full model in [99]. There comparisons between (i) 1D camera grid with and without normals and (ii) 1D camera grid with the 2D camera grid are shown. The full model performed well or better for all parameters and the additional parameters do not lead to instabilities. Also the 2D model is more robust with respect to the aperture problem. All these analyses are performed using the estimation framework presented below (Secs. 7.3.1 and 7.3.5)

7.2.4 Brightness changes

The basic approach of motion estimation requires brightness constancy along motion trajectories, as in the model equations (7.5), (7.6), (7.10) and (7.12) for

single motions and in Eq. (7.7) for multiple motions. In gradient based optical flow techniques, brightness changes can be modeled by partial differential equations in a similar manner as in [41]

$$\alpha(\boldsymbol{u}) \circ \boldsymbol{s} = \frac{\mathrm{d}}{\mathrm{d}t} h(s_0, t, \boldsymbol{\beta}) \ , \tag{7.13}$$

where the brightness change may depend on the initial gray value s_0, the time t and a parameter vector $\boldsymbol{\beta}$. It is modeled by the function $h(s_0, t, \beta)$. Physically motivated brightness changes include exponential ($h(t) \propto \exp(\beta t)$) and diffusive ($h(t) \propto \beta \Delta s$ with the spatial Laplacian Δs) processes. Also, the simple linear brightness change ($h(t) \propto \beta t$) can be used quite advantageously if an accurate model of the actual brightness change model is unknown. In these standard cases, β is a term independent of time t.

In the next sections, specialized application driven brightness change model, as well as incorporations of these brightness changes to dynamic 3D scene reconstruction and multiple motions will be introduced.

Brightness changes for fluid flows

The measurement of fluid flows is an emerging field for optical flow computation. In a number of such applications, a tracer is visualized with modern digital cameras. Due to the projective nature of the imaging process, the tracer is integrated across a velocity profile. For a number of fluid flow configuration, the velocity profile can be approximated to leading order by

$$u(x_3) = A \cdot x_3^n, \tag{7.14}$$

where A is a term independent of the coordinate direction of integration x_3 and time t. Integration across such a profile leads to an intensity change, modeled by the differential equation [27]

$$\frac{ds}{dt} = u_1 \frac{\partial s}{\partial x} + u_2 \frac{\partial s}{\partial y} + \frac{\partial s}{\partial t} = -\frac{1}{n \cdot t} s. \tag{7.15}$$

This equation presents a generalization of the results obtained for Couette flow (shear driven flow, $n = 1$) and Poiseuille flow (pressure driven flow, $n = 2$). These brightness change models take into account effects such as Taylor dispersion and have been applied successfully to microfluidics [32] or in shear driven flows at the air-water interface [30].

Brightness changes in dynamic 3D reconstruction

Brightness changes can be incorporated into dynamic 3D scene reconstruction suitably applying Eq. (7.13). One obtains the following constraint equation

$$\frac{\partial s}{\partial x}\mathrm{d}x + \frac{\partial s}{\partial y}\mathrm{d}y + \frac{\partial s}{\partial r}\mathrm{d}r + \frac{\partial s}{\partial t}\mathrm{d}t = s\frac{\partial h}{\partial t}\mathrm{d}t. \tag{7.16}$$

Brightness changes due to changing viewing direction and bidirectional reflectance distribution function(BRDF, see e.g. [40]) may be modeled only temporally, but changes due to inhomogeneous illumination need additional spatial modeling. A suitable brightness change function $h(\Delta X, \Delta Y, t)$ has been derived by Taylor expansion of changing BRDF influence and illumination intensity [106].

$$h(\Delta X, \Delta Y, t) \approx h(t, \boldsymbol{a}) := \sum_{i=1}^{2} \left(a_i + a_{i,x}\Delta X + a_{i,y}\Delta Y\right) t^i \tag{7.17}$$

with illumination parameter vector $\boldsymbol{a}$.

Comparison of physics based brightness variation models for 3D reconstruction model

A systematic error analysis using sinusoidal patterns in [106] demonstrates, that modeling spatial variations of the BRDF, as shown in equation Eq. (7.17), improves estimation results. Fig. 7.2.4 presents the improvement of the estimations on a reconstructed cube moving with U_Z=2mm/frame while illuminated by a rotating spot light. Fig. 7.2.4 **a** and **e** show the first and the last image of the cube sequence. In Fig. 7.2.4**b-d** the numerical error, i.e. the largest of the three eigenvalues of the structure tensor is depicted as color overlay on the central input image. Finally Fig. 7.2.4**f-h** highlight the estimation of the 3D scene flow for the three different models, i.e. constant, spatially constant but changing in time and both spatially and temporally changing BRDF.

Brightness changes in multiple motions and orientations

The multiple-motions model can be extended to account for brightness changes [102]. As with the operator $\alpha(\boldsymbol{u})$ for brightness constancy, one may define operators for additive, multiplicative or diffusive brightness changes. E.g. for multiplicative brightness changes Eq. (7.7) becomes

$$s(\boldsymbol{x}, t) = s_1(\boldsymbol{x} - \boldsymbol{u}_1 t)k_1(t) + s_2(\boldsymbol{x} - \boldsymbol{u}_2 t)k_2(t) \tag{7.18}$$

where k_1 and k_2 are scalar functions. It can be nullified via $\beta(\boldsymbol{u}_1, c_1) \circ \beta(\boldsymbol{u}_2, c_2) \circ s = 0$ using

$$\beta(\boldsymbol{u}, c) := u_x\partial_x + u_y\partial_y + \partial_t - c \tag{7.19}$$

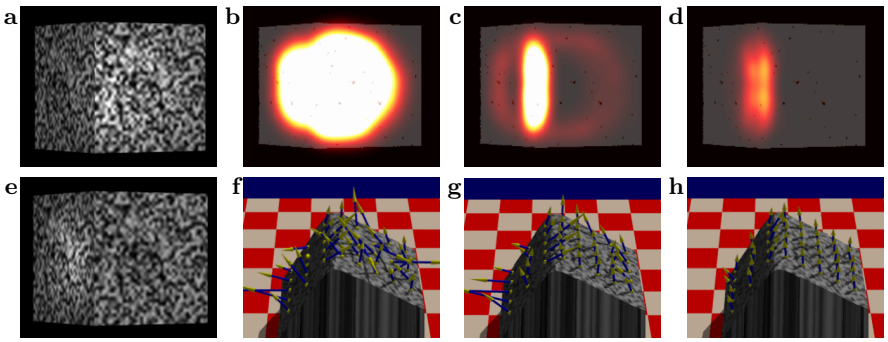

Fig. 7.1. Motion estimation of cube moving towards camera with spot light moving around cube center. (**a**, **e**): first and last image taken with central camera. (**b**–**d**): color coded model errors (projected on contrast reduced cube) for models without (**b**), constant temporal (**c**), and spatially varying temporal brightness change (**d**). Below the model errors, scaled motion estimates for the models are depicted, respectively (**f**–**h**).

if $k_1(t) \propto \exp(c_1 t)$ and $k_2(t) \propto \exp(c_2 t)$. As in the constant brightness case (see Sec. 7.2.2) the constraint equation is linear in mixed and therefore nonlinear parameters. A detailed analysis reveals that parameter estimation and disentangling motion parameters can be done as in the constant brightness case, but disentangling brightness parameters is done by solving a real instead of a complex polynomial [102].

7.3 Estimation of local orientation and motion

7.3.1 The structure tensor approach

All of the motion models presented in Sec. 7.2 linearly relate optical flow parameters, brightness changes and image signals. These constraints can be formulated generally as an inner product of a data vector $\boldsymbol{d}$ and a parameter vector p, resulting in

$$\boldsymbol{d}^\top \boldsymbol{p} = \alpha(\boldsymbol{u}) \circ s - \frac{\mathrm{d}}{\mathrm{d}t} h(s_0, t, \boldsymbol{\beta}) = 0, \tag{7.20}$$

where $\boldsymbol{d}$ contains image spatio-temporal derivatives and depending on the actual constraint, the gray values s themselves. The parameter vector $\boldsymbol{p}$ consists of the image velocities $\boldsymbol{u}$ as well as additional parameters such as those of brightness change $\boldsymbol{\beta}$ or those of higher order motion [8, 24, 25].

Eq. (7.20) provides one constraint for several unknowns we seek to estimate. An additional constraint is that of local constancy of parameters. To this

end, the constraints of type Eq. (7.20) can be constructed for each pixel in a spatio-temporal neighborhood, leading to a linear system of equations

$$\boldsymbol{D} \cdot \boldsymbol{p} = \boldsymbol{0}, \tag{7.21}$$

where $\boldsymbol{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}]^\top$. Assuming identical isotropic Gaussian noise (iid) in all measurements, the maximum likelihood estimate for the unknown parameter vector is given by the total least squares (TLS) solution [25, 91, 92, 124, 128]. The total least squares (TLS) method seeks to minimize $||\boldsymbol{D} \cdot \boldsymbol{p}||^2$, subject to the constraint that $\boldsymbol{p}^\top \boldsymbol{p} = 1$ to avoid the trivial solution. Usually the rows of $\boldsymbol{D}$ are weighted according to their distance from the central pixel by Gaussian weights w with standard deviation ρ. The structure tensor $\boldsymbol{J}_\rho$ then results from $||\boldsymbol{D} \cdot \boldsymbol{p}||^2$

$$||\boldsymbol{D} \cdot \boldsymbol{p}||^2 = \boldsymbol{p}^T \boldsymbol{D}^T \boldsymbol{W}_\rho \boldsymbol{D} \boldsymbol{p} =: \boldsymbol{p}^T \boldsymbol{J}_{\boldsymbol{\rho}} \boldsymbol{p} \tag{7.22}$$

where $\boldsymbol{W}_\rho$ is a diagonal matrix containing the Gaussian weights w. This formulation yields a solution, $\hat{\boldsymbol{p}}$, given by the right singular direction associated with the smallest singular value of row weighted $\boldsymbol{D}$ [74, 124] or the respective eigenvectors of $\boldsymbol{J}_\rho$. The sought parameter vector $\boldsymbol{p}$ is found by normalizing the last component of $\hat{\boldsymbol{p}}$ to unity [124]. The algorithmic aspects of TLS parameter estimation have been explored in some detail [73, 74, 124].

7.3.2 Beyond the differential approach: the generalized structure tensor

The attempt to express brightness constancy along the motion trajectory by a first-order partial differential equation, that is: by using the BCCE of Eq. (7.2) is not the unique and not the most expressive way of specifying a relation between the entity that is sought (the motion vector $\boldsymbol{u}$) and the signal that can be observed. The BCCE describes the situation for a *continuous signal*, and it does not explicitly consider the different error terms that are caused by observation noise, spatio-temporal pixel aperture, and by the necessary discretization of the problem. Beyond that, the formulation in terms of derivatives or gradients does not lend itself so much for the development of motion estimation procedures that take into account the spectral characteristics of the image signal and the spectral characteristics of the noise.

Assuming brightness constancy along the motion trajectory, *all* higher order directional derivatives vanish in the motion direction:

$$\frac{\partial s}{\partial \boldsymbol{u}} \stackrel{!}{=} 0 \quad \bigcap \quad \frac{\partial^2 s}{\partial \boldsymbol{u}^2} \stackrel{!}{=} 0 \quad \bigcap \quad \ldots \tag{7.23}$$

A condition which is less stringent than Eq. (7.23), but nevertheless comprises as much as possible from these multitude of conditions in a single linear equation can be obtained by summing up the constraints:

$$\alpha(\boldsymbol{u}) \circ s \quad = \quad \alpha_1 \frac{\partial s}{\partial \boldsymbol{u}} + \alpha_2 \frac{\partial^2 s}{\partial \boldsymbol{u}^2} + \alpha_3 \frac{\partial^3 s}{\partial \boldsymbol{u}^3} \quad \stackrel{!}{=} 0 \tag{7.24}$$

The middle part of this equation is a generator for a very rich class of filter operators, parameterized by direction vector $\boldsymbol{u}$:

$$h(\boldsymbol{x} \mid \boldsymbol{u}) * s(\boldsymbol{x}) \stackrel{!}{=} 0$$

This means that all linear operators that do not let an ideal oriented signal $s(\boldsymbol{x})$ pass[7], have the structure of Eq. (7.24). Since all oriented signals have power spectra that are concentrated on lines or planes in the Fourier domain, we can denote these filters as *oriented nulling filters*.

Like in the case of the normal BCCE, this equation will be satisfied almost never for a real image signal. Thus, we end up with optimization criteria like

$$\int_{\boldsymbol{x}} w(\boldsymbol{x}) \cdot |h(\boldsymbol{x} \mid \boldsymbol{u}) * s(\boldsymbol{x})|^2 \; d\boldsymbol{x} \quad \longrightarrow \; \min \tag{7.25}$$

where $h(\boldsymbol{x} \mid \boldsymbol{u})$ comprises the combination of directional derivatives of different order, and an optional *pre-filter* $p(\boldsymbol{x})$. This means: the *frequency-weighted* and *localized* directional variation of the signal is minimized in the direction of motion.

In standard differential motion estimation schemes, so-called *pre-filters* are used to compensate for the varying precision of the (discrete) derivative filters for different frequencies, and in particular to obtain isotropic performance of gradient estimates. Standard pre-filter design assumes that the Fourier spectra of the input signal and the noise are both white. For real signals, this is clearly not the case, as can be seen by inspecting the spatio-temporal autocovariance function of video signals. Furthermore, the possibly scenario-dependent distribution of motion vectors significantly controls the temporal part of the autocovariance function of the signal (cf. [53, 67]). This means that such pre-filters should be designed in consideration of the actual autocovariance function.

We will now generalize the concept of the *structure tensor* in consideration of Eq. (7.24), building upon a wider interpretation of the directional derivative operator.

We proceed by restating the relation between directional derivatives and steerable filters, which have been explored e.g. in [19, 26, 111]. The partial derivative in a direction specified by a unit vector $\boldsymbol{e}_r \in \mathbb{R}^3$ parameterized via spherical angles $\theta = (\theta_1, \theta_2)$ as

[7] **Multiple motions:** The constraint equation (7.71) generalized to

$$h(\boldsymbol{x} \mid \boldsymbol{u}_1) * h(\boldsymbol{x} \mid \boldsymbol{u}_2) * s(\boldsymbol{x}) \stackrel{!}{=} 0$$

$$\boldsymbol{e}_r = (a_1(\theta),\ a_2(\theta),\ a_3(\theta))$$

is given by

$$\frac{\partial}{\partial \boldsymbol{e}_r} s(\boldsymbol{x}) \ = \ \boldsymbol{e}_r^T \cdot \boldsymbol{g}(x) \ = \ \boldsymbol{e}_r^T \cdot \nabla s(\boldsymbol{x}) \ = \ \sum_{i=1}^{3} a_i(\theta) \cdot \frac{\partial s(\boldsymbol{x})}{\partial x_i} \tag{7.26}$$

Following the reasoning on pre-filters presented in Sec. 7.3.2, we may insert a prefilter $p(\boldsymbol{x})$ (see e.g. [95, 111])

$$\frac{\partial}{\partial \boldsymbol{e}_r}(s(\boldsymbol{x}) * p(\boldsymbol{x})) = \ = \ \left(\sum_{i=1}^{3} a_i(\theta_1, \theta_2) \cdot \left(\frac{\partial}{\partial x_i} p(\boldsymbol{x}) \right) \right) * s(\boldsymbol{x}) \tag{7.27}$$

For $p(\boldsymbol{x})$ there are therefore many more functions under consideration than only a simple Gaussian kernel[8]. We can design $p(\boldsymbol{x})$ in a way that optimizes the signal/noise ratio at the output of the prefilter; this is (again) the *Wiener-type prefilter approach* [53, 69].On the other hand, we may generalize the structure of the analysis scheme described by Eq. (7.27) and arrive at a *generalized class of structure tensors*, as will be shown in the following.

Steerable oriented signal energy determination

We abstract now from derivative filters and regard a family of steerable filter operators which can be written in the form [26]

$$h(\boldsymbol{x} \mid \theta) = \sum_{i=1}^{N} a_i(\theta) \cdot b_i(\boldsymbol{x})$$

Since the original signal is *sheared* instead of being rotated by motion, it is appropriate to design $h(\boldsymbol{x} \mid \theta)$ accordingly; however, we will not deal here with details of such *shearable filters*. The symbol θ stands for a general parameter (or parameter vector) that controls the direction in which the filter operator is being steered. The $b_i(\boldsymbol{x})$ are basis functions;$a_i(\theta)$ and $b_i(\boldsymbol{x})$ are subject to certain conditions discussed in [26]. This operator will now be applied to an input signal $s(\boldsymbol{x})$:

$$h(\boldsymbol{x} \mid \theta) * s(\boldsymbol{x}) = \sum_{i=1}^{N} a_i(\theta) \cdot (b_i(\boldsymbol{x}) * s(\boldsymbol{x}))$$

As before, the local energy of the resulting signal will be computed. The localization of the computation is again ensured by the weight function $w(\boldsymbol{x})$:

[8] In general, a binomial filter does much better than a sampled (and truncated) Gaussian.

$$Q(\theta) = \int_{\boldsymbol{x}} w(\boldsymbol{x}) \cdot \left(h(\boldsymbol{x} \mid \theta) * s(\boldsymbol{x})\right)^2 \, d\boldsymbol{x}$$

A closer look reveals (using $g_i(\boldsymbol{x}) \equiv s(\boldsymbol{x}) * b_i(\boldsymbol{x})$):

$$\begin{aligned}\left(h(\boldsymbol{x} \mid \theta) * s(\boldsymbol{x})\right)^2 &= \left(\sum_{i=1}^{N} a_i(\theta) \cdot \left(b_i(\boldsymbol{x}) * s(\boldsymbol{x})\right)\right)^2 \;=\; \left(\sum_{i=1}^{N} a_i(\theta) \cdot g_i(\boldsymbol{x})\right)^2 \\ &= \sum_{i=1}^{N} \sum_{k=1}^{N} a_i(\theta) \cdot a_k(\theta) \cdot g_i(\boldsymbol{x}) \cdot g_k(\boldsymbol{x})\end{aligned}$$

If now a local integration is performed across this squared signal we obtain:

$$Q(\theta) = \sum_{i=1}^{N} \sum_{k=1}^{N} a_i(\theta) \cdot a_k(\theta) \int_{\boldsymbol{x}} w(\boldsymbol{x}) \cdot g_i(\boldsymbol{x}) \cdot g_k(\boldsymbol{x}) \, d\boldsymbol{x} \tag{7.28}$$

With the shorthand notation

$$J_{ik} \;\stackrel{def}{=}\; \int_{\boldsymbol{x}} w(\boldsymbol{x}) \cdot g_i(\boldsymbol{x}) \cdot g_k(\boldsymbol{x}) \, d\boldsymbol{x} \tag{7.29}$$

$$\text{we obtain} \qquad Q(\theta) = \sum_{i=1}^{N} \sum_{k=1}^{N} a_i(\theta) \cdot a_k(\theta) \cdot J_{ik} \tag{7.30}$$

This is a quadratic form

$$Q(\theta) \;=\; \begin{pmatrix} a_1(\theta) \\ \vdots \\ a_N(\theta) \end{pmatrix}^T \begin{pmatrix} J_{11} & \cdots & J_{1N} \\ \vdots & \ddots & \vdots \\ J_{N1} & \cdots & J_{NN} \end{pmatrix} \begin{pmatrix} a_1(\theta) \\ \vdots \\ a_N(\theta) \end{pmatrix} \;=\; \boldsymbol{a}^T(\theta) \cdot \mathbf{J} \cdot \boldsymbol{a}(\theta) \tag{7.31}$$

$$\text{with} \quad \boldsymbol{a}(\theta) \;\stackrel{def}{=}\; \begin{pmatrix} a_1(\theta) \\ \vdots \\ a_N(\theta) \end{pmatrix} \quad \text{and} \quad \mathbf{J} \;\stackrel{def}{=}\; \begin{pmatrix} J_{11} & \cdots & J_{1N} \\ \vdots & \ddots & \vdots \\ J_{N1} & \cdots & J_{NN} \end{pmatrix}$$

In the standard structure tensor approach, $N = 3$, and $h(\boldsymbol{x} \mid \theta)$ is the first order directional derivative which can be represented by a steerable set of $N = 3$ filters (each of them representing the directional derivative in one of the principal directions of space-time). It is not very surprising that in this case $\boldsymbol{a}(\theta)$ is a unit vector in $\mathbb{R}^3$, and the determination of the argument θ which minimizes $Q(\theta)$ boils down to a simple eigensystem problem, as given already in Sec. 7.3.1.

For synthesizing and steering a more general filter operator $h(\boldsymbol{x} \mid \theta)$, we know that the basis functions $b_i(\boldsymbol{x})$ should be polar-separable harmonic functions.The coefficient functions $a_i(\theta)$ will then be trigonometric functions of

different (harmonic) frequencies [135], and the optimization problem will not be so simple to solve, though well-behaved. The design of the localization function $w(\boldsymbol{x})$ and the generalization of the directional derivative can be adapted to the signal and noise power spectra, respectively [53, 69]. Within this framework, a wide class of orientation selective steerable filters can be used to find principal orientations; if necessary they can equipped with a much more pronounced selectivity, offering the potential for higher accuracy.

7.3.3 A mixed OLS-TLS estimator

Local estimators of motion pool a constraint equation in a local neighborhood constructing an overdetermined system of equations. The parameters of the motion model can then be solved by ordinary least squares (OLS) [62] or by total least squares (TLS), resulting in the structure tensor approach. Using (OLS) techniques, the temporal derivatives are treated as erroneous observations and the spatial gradients as error free. This approach will lead to biases in the estimates, as all gradients are generally obscured by noise [43]. Under these circumstances the use of a total least squares (TLS) method [125] is the estimator of choice [72]. A number of physically induced brightness changes as well as those caused by inhomogeneous illumination can be modeled quite accurately by a source term in the constraint equation. Additionally does the computation of surface motion from range data lead to the same type of constraints [113]. The equation of motion is given by

$$\alpha(\boldsymbol{u}) \circ \boldsymbol{s} - 1 \cdot c = 0 \;, \tag{7.32}$$

where c is a constant, modeling the local brightness change linearly.

The data matrix of such a model for the TLS estimator contains a column of exactly known elements (the elements $D_{i,1} = -1$ for $i \in \{1, \ldots, n\}$ where n is the number of pixel in the local neighborhood) thus inducing a strong bias in the estimation. This bias can be efficiently eliminated by mixing the OLS and TLS estimator as presented by [33].

The data matrix $\boldsymbol{D}$ can be split into two submatrices $\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2]$, where $\boldsymbol{D}_1$ contains the p_1 exactly known observations. A QR factorization of $\boldsymbol{D}$ is performed, leading to

$$(\boldsymbol{D}_1, \boldsymbol{D}_2) = \boldsymbol{Q} \begin{pmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ \boldsymbol{0} & \boldsymbol{R}_{22} \end{pmatrix}, \tag{7.33}$$

with $\boldsymbol{Q}$ being orthogonal and $\boldsymbol{R}_{11}$ upper triangular. The QR factorization is justified because the singular vectors and singular values of a matrix are not changed by multiplying it by an orthogonal matrix [38].

The solution for the sub system of equations $\boldsymbol{R}_{22}\boldsymbol{p}_2 = 0$ is computed in a TLS sense, which boils down to a singular value analysis of the data matrix $\boldsymbol{R}_{22}$ [125].

With the known estimate of $\boldsymbol{p}_2$ the system of equations $\boldsymbol{R}_{11}\boldsymbol{p}_1 + \boldsymbol{R}_{12}\boldsymbol{p}_2 = 0$ is solved for $\boldsymbol{p}_1$ by back-substitution. A comparative analysis has shown, that the error in the mixed OLS-TLS estimates can be reduced by a factor of three as compared to standard TLS [33].

7.3.4 Simultaneous estimation of local and global parameters

Local estimation schemes, like all estimation schemes presented so far, e.g. the structure tensor method (cf. Sec. 7.3.1) or mixed OLS-TLS scheme (cf. Sec. 7.3.3), can be implemented efficiently in terms of storage needed and CPU time used. This is due to local formulation of the models and their parameters, because then all estimations can be done separately for each pixel neighborhood. In other words the model equation matrix is a block diagonal matrix with one block per pixel and one block after the other is processed. This is no longer true if global parameters have to be estimated as well. They introduce additional full rows in the model matrix, thus coupling all blocks. Thus the optimization problem occurring in the estimation process has to be treated as a large scale problem and can only be solved for practical applications if the problem structure is carefully exploited. In [17] an OLS estimation method is presented for simultaneous estimation of local and global parameters which full exploits the structure of the estimation matrix. It has comparable complexity and memory requirements as pure local methods. The numerical solution method makes use of the so called *Sherman-Morrison-Formula* [37], which allows to efficiently obtain the inverse of an easily invertible matrix (the block diagonal matrix) when it is modified by a low rank matrix (the few full rows).

Ordinary least squares (OLS) means the minimization problem is posed as

$$\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 \quad \Rightarrow \quad \bar{\boldsymbol{x}} = (A^T A)^{-1} A^T b \tag{7.34}$$

where $\boldsymbol{x}$ are the sought for parameters, $\boldsymbol{A}$ and $\boldsymbol{b}$ are defined by the model. The solution vector $\bar{\boldsymbol{x}}$ is given by the Moore-Penrose pseudo-inverse $(A^T A)^{-1} A^T$, if $A^T A$ is invertible. The matrix A has the following block structure

$$A = \left[\begin{array}{ccc|c} B_1 & & & V_1 \\ & \ddots & & \vdots \\ & & B_N & V_N \end{array}\right] = \left[\begin{array}{c|c} B & V \end{array}\right] \quad \text{and thus} \quad A^T A = \left[\begin{array}{c|c} B^T B & B^T V \\ \hline V^T B & V^T V \end{array}\right]. \tag{7.35}$$

with $n_\Omega \times N_{\mathrm{lp}}$-blocks B_i and $n_\Omega \times N_{\mathrm{gp}}$-blocks V_i. Finally, the squared matrix can be decomposed as

$$A^T A = M + RSR^T \text{ with } M = \left[\begin{array}{c|c} B^T B & 0 \\ \hline 0 & V^T V \end{array}\right],\ R = \left[\begin{array}{c|c} B^T V & 0 \\ \hline 0 & \mathbf{I} \end{array}\right],\ S = \left[\begin{array}{c|c} 0 & \mathbf{I} \\ \hline \mathbf{I} & 0 \end{array}\right], \tag{7.36}$$

with matrix M block diagonal and matrix R low rank, $2N_{\mathrm{gp}}$, so that the Sherman-Morrison-Woodbury formula gives an efficiently computable inverse:

$$(A^T A)^{-1} = (\mathbf{I} - M^{-1}R(S^{-1} + R^T M^{-1} R)^{-1} R^T) M^{-1}. \tag{7.37}$$

In addition to the matrix blocks $B_i^T B_i$ and $\sum_{i=1}^{N} V_i^T V_i$ of M one therefore only has to invert one further $(2N_{\mathrm{gp}}) \times (2N_{\mathrm{gp}})$ matrix, $(S^{-1} + R^T M^{-1} R)$, and all remaining calculations for computation of $\bar{x} = (A^T A)^{-1} A^T b$ can be performed as matrix vector products. As the inversion of the matrix blocks $B_i^T B_i$ is by far the most time consuming step in the computations of the algorithm, the computational burden is comparable to that of an OLS estimation *without* global parameters.

For an illustrative example of combined estimation of local optical flow and global camera gain as well as for further details on the numerical solution we refer to [17].

7.3.5 Simultaneous estimation of 3D position and 3D motion

Modeling dynamic data acquired with a 2D camera grid, the solution space is a three dimensional subspace ($n = 5$, $d = 3$, cf. Secs. 7.2.1 and 7.2.3). Using the weighted TLS estimation scheme presented in Sec. 7.3.1), it is spanned by the 3 right singular vectors corresponding to the 3 smallest singular values of the data matrix $\boldsymbol{D}$. From these singular vectors the sought for parameters are derived by linear combination such that all but one component of $\mathrm{d}r_1$, $\mathrm{d}r_2$, $\mathrm{d}t$ vanish. The parameters for disparity, surface slopes and depth motion occur twice in the model (see Eq. (7.11)). Their estimates cannot be combined straightforward because they are not independent in the original coordinate system. In order to decouple these measurements we first estimated their error covariance matrix as proposed in [90]. After diagonalizing this matrix via a rotation in r_1-r_2-space (for disparity and surface slopes) or x-y-space (for u_z), we achieve independent measurements which can be combined respecting their individual errors (see [96]). A more detailed description of decoupling the motion estimates is presented in [99].

7.3.6 Motion estimation with the energy operator

For continuous signals $\boldsymbol{s} = \boldsymbol{s}(\boldsymbol{x})$ the energy operator E is defined as [75, page 177]

$$E(\boldsymbol{s}) := D(\boldsymbol{s})^2 - \boldsymbol{s} \cdot D^2(\boldsymbol{s}) \; . \tag{7.38}$$

Here D denotes an abstract derivative or pseudo-derivative operator.

Implementation for optic flow estimation

We implemented the energy operator in three dimensions for optic flow estimation based on three dimensional derivative filters in the spatio-temporal domain in a straightforward manner using Eq. (7.38) by replacing D^2 with the Hessian of the sequence. We used the modified Sobel filters as described in [103] and [95, page 155]. For comparison between different implementations, the Hessian of the spatio-temporal image data was computed either by twice applying first order derivative filters or second order derivative masks.

Numerical experiments

In our experiments we computed the energy operator on original resolution. We conducted measurements of the average angular error for a synthetic sequence without noise, for synthetic sequences with noise and a real world test sequence acquired by a camera.

Since we estimated image sequences with ground truth, we compared the best results of a total least squares local approach for the energy operator and the structure tensor. For results see Tab. 7.1.

Sequence	Derivative filter	Optimal integration scale	Average angular error
Sinus pattern	**structure tensor**	0.24	4.589143
	energy operator by ...		
	first order derivative	3.54	10.2803
	second order derivative	3.43	11.59
Marble	**structure tensor**	3.2	3.509298
	energy operator by ...		
	first order derivative	2.59	3.215193
	second order derivative	6.02	4.498214
Street	**structure tensor**	1.57	4.589143
	energy operator by ...		
	first order derivative	6.1	10.251023
	second order derivative	5.4	9.240537

Table 7.1. Results for optic flow estimation with the energy tensor and comparison with the structure tensor.

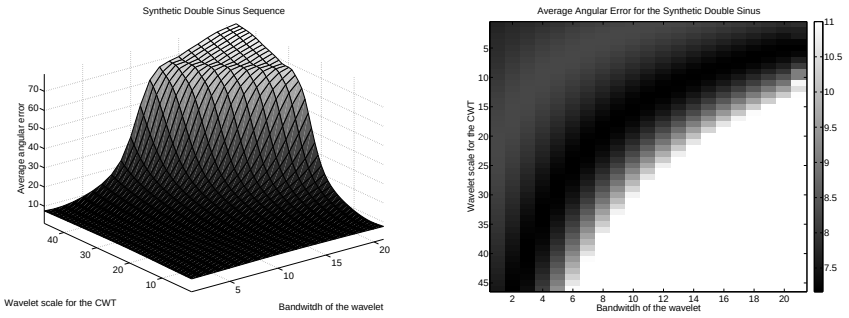

Fig. 7.2. The optimal average angular error as a function of the bandwidth and scale of the Mexican hat wavelet for a synthetic sinus pattern sequence with discontinuity. In the left image the range is set between 6 and 11 degrees, in order to show the region around the minimum.

On the effect of the bandpass filtering: filter bandwidth versus wavelet scale

We investigated the dependency of the average angular error as a function of the bandwidth and wavelet scale simultaneously. Here we mean the bandwidth of the bandpass filter or the spread of the wavelet, used to filter the input image sequence. As a result it comes out, that there is an optimal point in the bandwidth-scale plane which minimizes the error measure, see also figures 7.2 and 7.3.

Despite the preprocessing of the image sequence, our experiments showed, that there is a need of post-integration for the energy operator to achieve optimal average angular error of the estimated flow fields. For orientation estimation with the energy operator [21, page 498] reported similar results and applied a Gaussian post-filtering with $\sigma = 1$ and a smoothing window size of 7x7.

The accuracy gain for the real world Marbled Block sequence is in the first digit behind the decimal point or approximately 7.7 %. This improvement is acchieved under optimal parameter setting for both operators, the structure tensor and the energy operator. For the synthetic sequence with discontinuity and the Street sequence we measured higher accuracy with the structure tensor.

The energy operator requires additional computation steps, this is the calculation of the Hessian of the image data. For this reason and because of the measurements in our experiment we recommend the structure tensor for motion estimation in the scale space.

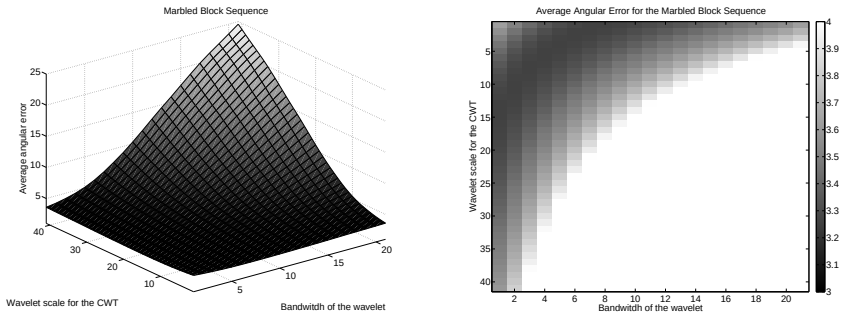

Fig. 7.3. The optimal average angular error as a function of the bandwidth and scale of the Mexican hat wavelet for the Marbled Block sequence. In the left image the range is set between 3 and 4 degrees, in order to show the region around the minimum.

7.4 Filter design

All models described above contain derivative filters of first order (e.g. Eqs. (7.2), (7.5), or (7.11)), second order (Eq. (7.71)) or higher orders, e.g. if brightness changes are due to diffusive processes or more than two orientations are present. They are discretized by finite differences using convolution kernels optimized with respect to the model assumptions, scales and/or noise present in the data.

7.4.1 Optimal filters for linear models

As shown in Sec. 7.2 linear models describe linear subspaces in multi-dimensional data in the form $\boldsymbol{d}^T\boldsymbol{p} = 0$ (cmp. Eq. (7.20)). Thus a parameter vector $\boldsymbol{p}$ is a solution to Eq. (7.20) if it is normal to the data vector $\boldsymbol{d}$. In all the models above except the ones with a data independent source-term, the data vector $\boldsymbol{d}$ can be formulated as an operator vector $\boldsymbol{\mathcal{O}}$ applied to the signal s. Consequently a discrete filter family is *optimal for a model* if the *orientation* of the operator vector calculated by the filter family is as precise as possible. This observation can be formulated in Fourier domain (see below for an example). Selecting fixed size filter sets, their transfer functions (TFs, i.e. Fourier transforms) are known. The coefficients of the whole filter set may then be optimized simultaneously by adapting a TF optimization scheme first presented in [95]. There a weighted 2-norm of the difference of an ansatz function $f_a(\tilde{\boldsymbol{k}})$ and a reference function $f_r(\tilde{\boldsymbol{k}})$ is minimized

$$c(\boldsymbol{h}) = \sqrt{\int w^2(\tilde{\boldsymbol{k}}) \left(f_r(\tilde{\boldsymbol{k}}) - f_a(\tilde{\boldsymbol{k}}, \boldsymbol{h})\right)^2 \mathrm{d}\tilde{\boldsymbol{k}}} \quad \text{with} \quad \int w^2(\tilde{\boldsymbol{k}}) \mathrm{d}\tilde{\boldsymbol{k}} = 1 \quad (7.39)$$

with normalized wave-numbers $\tilde{\boldsymbol{k}}$, i.e. Nyquist wave-number equal to 1. The normalized weight function $w(\tilde{\boldsymbol{k}})$ allows to specify statistical importance of different wave vectors $\tilde{\boldsymbol{k}}$ and thus allows for scale selection.

Example: Filter families for two transparent motions

The operator vector $\boldsymbol{\mathcal{O}}$ for transparent motion can be extracted from Eq. (7.71)

$$\boldsymbol{\mathcal{O}} = [\partial_{xx}, \partial_{xy}, \partial_{yy}, \partial_{xt}, \partial_{yt}, \partial_{tt}]^T \tag{7.40}$$

and its TF is $\hat{\boldsymbol{\mathcal{O}}} = -\pi^2 \left[\tilde{k}_x^2, \tilde{k}_x\tilde{k}_y, \tilde{k}_y^2, \tilde{k}_x\tilde{k}_t, \tilde{k}_y\tilde{k}_t, \tilde{k}_t^2\right]^T$. The reference function therefore is $\hat{\boldsymbol{\mathcal{O}}}$ normalized by its length

$$f_r(\tilde{k}) = \frac{[\tilde{k}_x^2, \tilde{k}_x\tilde{k}_y, \tilde{k}_y^2, \tilde{k}_x\tilde{k}_t, \tilde{k}_y\tilde{k}_t, \tilde{k}_t^2]^T}{\sqrt{\tilde{k}_x^4 + \tilde{k}_x^2\tilde{k}_y^2 + \tilde{k}_y^4 + \tilde{k}_x^2\tilde{k}_t^2 + \tilde{k}_y^2\tilde{k}_t^2 + \tilde{k}_t^4}} \tag{7.41}$$

We discretize $\boldsymbol{\mathcal{O}}$ (from Eq. (7.40)) using fixed size separable kernels. The filter family consists of only four 1D filters: a first order derivative D^1, a second order derivative D^2 and two smoothing kernels I^1 and I^2. Please note the upper indices. The 3D filters are then $\partial_{xy} = \mathcal{D}_x^1 * \mathcal{D}_y^1 * \mathcal{I}_t^1$ and $\partial_{xx} = \mathcal{D}_x^2 * \mathcal{I}_y^2 * \mathcal{I}_t^2$ where $*$ denotes convolution and lower indices denote the application direction. All filters not introduced above, can be derived by suitably exchanging lower indices. All one-dimensional filters are constrained to numerical consistency order 2 (cmp. [95]). We refer to [98] for further details. Using the transfer functions $\hat{\mathcal{D}}_{xx}$, $\hat{\mathcal{D}}_{xy}$, $\hat{\mathcal{D}}_{yy}$, $\hat{\mathcal{D}}_{xt}$, $\hat{\mathcal{D}}_{yt}$, and $\hat{\mathcal{D}}_{tt}$ of these filter kernels, we get the ansatz function

$$f_a(\tilde{k}) = \frac{[\hat{\mathcal{D}}_{xx}, \hat{\mathcal{D}}_{xy}, \hat{\mathcal{D}}_{yy}, \hat{\mathcal{D}}_{xt}, \hat{\mathcal{D}}_{yt}, \hat{\mathcal{D}}_{tt}]^T}{\sqrt{\hat{\mathcal{D}}_{xx}^2 + \hat{\mathcal{D}}_{xy}^2 + \hat{\mathcal{D}}_{yy}^2 + \hat{\mathcal{D}}_{xt}^2 + \hat{\mathcal{D}}_{yt}^2 + \hat{\mathcal{D}}_{tt}^2}} \tag{7.42}$$

E.g. for $5 \times 5 \times 5$- filters the optimization results in the kernels ($c = 1.6e-12$)

$$\begin{aligned} I^1 &= [0.01504, 0.23301, 0.50390, 0.23301, 0.01504] \\ I^2 &= [0.01554, 0.23204, 0.50484, 0.23204, 0.01554] \\ D^1 &= [0.06368, 0.37263, 0, -0.37263, -0.063681] \\ D^2 &= [0.20786, 0.16854, -0.75282, 0.16854, 0.20786] \end{aligned} \tag{7.43}$$

For larger kernels we refer to [98].

7.4.2 Steerable and quadrature filters

Quadrature filters have become an appropriate tool for computing the local phase and local energy of one dimensional signals. In the following, we describe

the relation between steerable filter and quadrature filter. For a detailed description on steerable filters we refer to [58].

The main idea of a quadrature filter is to apply two filters to a signal such that the sum of the square filter responses reflect the local energy of the signal. Furthermore, the local energy should be group invariant, i.e. the filter outputs should be invariant with respect to the deformation of the signal by the corresponding group. In order to achieve group invariance, we construct our quadrature filter from the basis of a unitary group representation. Groups with a unitary representation are compact groups and Abelian groups [131]. The even $\boldsymbol{h}_e$ and odd $\boldsymbol{h}_o$ components of the quadrature filter are constructed by a vector valued impulse response consisting of the basis functions of a unitary representation of dimension m_e and m_o, respectively.

$$\boldsymbol{h}_e = \begin{pmatrix} h_{e1}(\boldsymbol{x}) \\ h_{e2}(\boldsymbol{x}) \\ \vdots \\ h_{em_e}(\boldsymbol{x}) \end{pmatrix}, \quad \boldsymbol{h}_o = \begin{pmatrix} h_{o1}(\boldsymbol{x}) \\ h_{o2}(\boldsymbol{x}) \\ \vdots \\ h_{om_o}(\boldsymbol{x}) \end{pmatrix} . \tag{7.44}$$

It can be shown that all basis functions belonging to the same subspace attain the same parity. The filter responses of $\boldsymbol{h}_e$ and $\boldsymbol{h}_o$ are denoted as the filter channels $\boldsymbol{c}_e = s(\boldsymbol{x}) * \boldsymbol{h}_e(\boldsymbol{x})$ and $\boldsymbol{c}_o = s(\boldsymbol{x}) * \boldsymbol{h}_o(\boldsymbol{x})$, respectively. The square of the filter response of each channel are denoted as even and odd energies. Due to the unitary representation, both energies are invariant under the corresponding group action

$$E_s = \left(\mathbf{D}(g)\boldsymbol{c}_s\right)^T \left(\mathbf{D}(g)\boldsymbol{c}_s\right) = \boldsymbol{c}_s^T \boldsymbol{c}_s \quad s \in \{e, o\} \ .$$

Note that the inner product is taken with respect to the invariant subspace, not with respect to the function space. The local energy of the signal is given by the sum of the even and odd energy. In the following we will examine the properties of the filter channels required to achieve a phase invariant local energy when applied to bandpass signals. In the ideal case, a simple[9] bandpass filtered signal consists of only one wave vector $\boldsymbol{k}_0$ and its Fourier transform[10] reads with the Dirac delta distribution $\delta(\boldsymbol{k})$

$$S(\boldsymbol{k}) = S_0 \delta(\boldsymbol{k} - \boldsymbol{k}_0) + S_0^* \delta(\boldsymbol{k} + \boldsymbol{k}_0) \ . \tag{7.45}$$

We start with examining the Fourier transform of the even and odd energies

$$E_s = \boldsymbol{c}_s^T \boldsymbol{c}_s = \sum_{j=1}^{m_s} \left(s(\boldsymbol{x}) * h_{sj}(\boldsymbol{x})\right)^2 \ . \tag{7.46}$$

Applying the convolution theorem to E_s reads

[9] simple signal: signal with intrinsic dimension one.

[10] Note that the Fourier transformed entities are labeled with capital letters.

$$\mathcal{F}\{E_s\}(\boldsymbol{k}) = \sum_{j=1}^{m_s} (S(\boldsymbol{k})H_{sj}(\boldsymbol{k})) * (S(\boldsymbol{k})H_{sj}(\boldsymbol{k})) \quad .$$

Inserting the signal Eq. (7.45) in the equation above, computing the convolution and performing the inverse Fourier transformation results in a phase invariant local energy

$$E = 2|S_0|^2 \left(\sum_{j=1}^{m_e} |H_{ej}(\boldsymbol{k}_0)|^2 + \sum_{k=1}^{m_o} |H_{ok}(\boldsymbol{k}_0)|^2 \right) \quad .$$

Thus, each steerable filter whose basis can be decomposed as described above is a quadrature filter which is invariant with respect to the corresponding group action.

7.4.3 Design and application of Wiener-optimized filters and average masks

The filter masks described in Sec. 7.4 have been optimized for ideal noiseless signals. However, the fact that all real world images are with different extents corrupted by noise has thus been neglected. In the following we present how these filters have to be adapted in case of noisy signals. A detailed treatment can be found in [52].

The signal and noise adapted filter approach

The signal and noise adapted (SNA)-filter approach is motivated by the fact that we can exchange the directional derivative filter $d_{\boldsymbol{r}}(\boldsymbol{x}_n)$ in the BCCE by any other steerable filter $h_{\boldsymbol{r}}(\boldsymbol{x}_n)$ which only nullifies the signal when applied in the direction of motion [70]. The shape of the frequency spectrum of any *rank 2* signal[11] is a plane $K_{\boldsymbol{r}}$ going through the origin of the Fourier space and its normal vector $\boldsymbol{n}$ points to the direction of motion $\boldsymbol{r}$ ([44],p.316). Thus, the transfer function[12] $H_{\boldsymbol{r}}(\boldsymbol{f})$ has to be zero in that plane, but the shape of $H_{\boldsymbol{r}}(\boldsymbol{f})$ outside of plane $K_{\boldsymbol{r}}$ can be chosen freely as long as it is not zero at all. If the impulse response $h_{\boldsymbol{r}}(\boldsymbol{x}_n)$ shall be real-valued, the corresponding transfer function $H_{\boldsymbol{r}}(\boldsymbol{f})$ has to be real and symmetric or imaginary and antisymmetric or a linear combination thereof. The additional degrees of freedom to design the shape outside $K_{\boldsymbol{r}}$ make it possible to consider the spectral characteristics of the signal and the noise which are encoded in the second order statistical moments in the filter design. In the following section the derivation of an optimal filter is shown which is a special case of the more general framework presented in [87] and for the special case of motion estimation in [53].

[11] The rank of a signal is defined by the rank of the corresponding structure tensor.
[12] The Fourier transforms of functions are denoted here by capital letters.

General model of the observed signal

The general idea of the SNA-filter proposed first in [68] is to combine Wiener's theory of optimal filtering with a desired ideal impulse response. The term ideal, in this case, means that the filter is designed for noise free signal $s(\boldsymbol{x})$. But signals are always corrupted by noise. Our goal is now to adapt the ideal filter $h_{\boldsymbol{r}}(\boldsymbol{x})$ to more realistic situations where signal is corrupted by noise. We model the observed image signal z at position i, j, k in a spatio-temporal block of dimension $N \times N \times N$ by the sum of the ideal (noise free) signal s and a noise term v: $z(i,j,k) = s(i,j,k) + v(i,j,k)$. For the subsequent steps, it is convenient to arrange the s, v, and z of the block in vectors $\boldsymbol{s} \in \mathbb{R}^M, \boldsymbol{v} \in \mathbb{R}^M$ and $\boldsymbol{z} \in \mathbb{R}^M$. The extraction of a single filter response value $\hat{g}$ can thus be written as the scalar product $\hat{g} = \boldsymbol{x}^T\boldsymbol{z}$ using a filter coefficient vector $\boldsymbol{x} \in \mathbb{R}^M$. The corresponding equation for the actual filter output $\hat{g}$ reads:

$$\hat{g} = \boldsymbol{x}^T\boldsymbol{z} = \boldsymbol{x}^T(\boldsymbol{s} + \boldsymbol{v}) = \boldsymbol{x}^T\boldsymbol{s} + \boldsymbol{x}^T\boldsymbol{v} \; . \tag{7.47}$$

Our task is to choose $\boldsymbol{x}^T$ in such a way that the filtered output $\hat{g}$ approximates, on an average, the desired output $g = \boldsymbol{h}^T\boldsymbol{s}$ for the error-free case as closely as possible. The next step is to define the statistical properties of the signal and the noise processes, respectively. Let the noise vector $\boldsymbol{v} \in \mathbb{R}^N$ be a zero-mean random vector with covariance matrix $\mathsf{E}\left[\boldsymbol{v}\boldsymbol{v}^T\right] = \mathbf{C}_v$ (which is in this case equal to its correlation matrix $\mathbf{R}_v$). Furthermore, we assume that the process which has generated the signal $\boldsymbol{s} \in \mathbb{R}^N$ can be described by the expectation $\mathsf{E}[\boldsymbol{s}] = \boldsymbol{m}_s$ of the signal vector, and an autocorrelation matrix $\mathsf{E}\left[\boldsymbol{s}\boldsymbol{s}^T\right] = \mathbf{R}_s$. Our last assumption is that noise and signal are uncorrelated $\mathsf{E}\left[\boldsymbol{s}\boldsymbol{v}^T\right] = 0$.

Designing the optimized filter

Knowing these first and second order statistical moments for both the noise as well as the signal allows the derivation of the optimum filter $\boldsymbol{x}$. For this purpose, we define the approximation error $e := \hat{g} - g$ between the ideal output g and the actual output $\hat{g}$. The expected squared error Q as a function of the vector $\boldsymbol{x}$ can be computed from the second order statistical moments:

$$\begin{aligned} Q(\boldsymbol{x}) &= \mathsf{E}\left[e^2\right] = \mathsf{E}\left[\hat{g}^2\right] - 2\mathsf{E}\left[g\hat{g}\right] + \mathsf{E}\left[g^2\right] \\ &= \boldsymbol{h}^T\mathbf{R}_s\boldsymbol{h} - 2\boldsymbol{x}^T\mathbf{R}_s\boldsymbol{h} + \boldsymbol{x}^T(\mathbf{R}_s^T + \mathbf{R}_v)\boldsymbol{x} \end{aligned}$$

We see that a minimum mean squared error (MMSE) estimator can now be designed. We set the derivative $\partial Q(\boldsymbol{x})/\partial\boldsymbol{x}$ to $\mathbf{0}$ and after solving for $\boldsymbol{x}$ we obtain

$$\boldsymbol{x} = \underbrace{(\mathbf{R}_s^T + \mathbf{R}_v)^{-1}\mathbf{R}_s}_{\mathbf{M}}\boldsymbol{h} = \mathbf{M}\boldsymbol{h} \; . \tag{7.48}$$

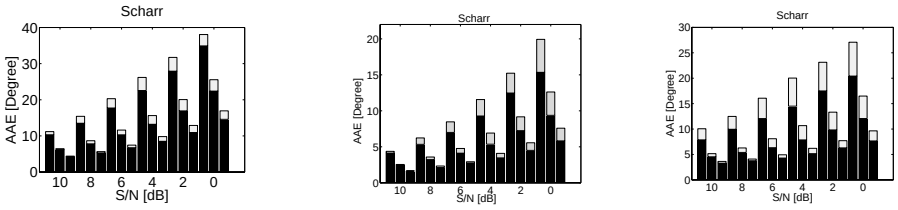

Fig. 7.4. Bar plots of the average angular error (AAE) vs. the signal to noise (S/N) ratio. For every S/N ratio the AAE for three different densities is depicted: From left to right: 86%, 64% and 43%. The gray bar denotes the AAE with the original, the black ones the AAE with the optimized filters. Note that the experiment is performed at rather low S/N ratio in range 10 dB to 0 dB.

Thus, the desired SNA-filter is obtained by a matrix vector multiplication. The ideal filter designed for the ideal noise free case is multiplied with a matrix **M** composed out of the correlation matrices of signal and noise. In principle, this result is a direct consequence of the Gauss-Markov theorem.

Experimental results

In this section, we present examples which show the performance of our optimization method. For the test we use three image sequences, together with the true optical flow: 'Yosemite' (without clouds) 'diverging tree' and 'translating tree' sequences[13]. The optical flow has been estimated with the tensor based method described in Sec. 7.3.1. For all experiments, an averaging volume of size $11 \times 11 \times 11$ and filters of size $5 \times 5 \times 5$ are applied. For the weighting function $w(\boldsymbol{x})$, we chose a sampled Gaussian function with width $\sigma = 8$ (in pixels) in all directions. For performance evaluation, the average angular error (AAE) [4] is computed. The AAE is computed by taking the average over 1000 trials with individual noise realization. In order to achieve a fair comparison between the different filters but also between different signal-to-noise ratios S/N, we compute all AAEs for three different but fixed densities [14] determined by applying the *total coherence measure* and the *spatial coherence measure* [44]. We optimized the described in Sec. 7.4 denoted as SCHARR filter in the following, for every individual signal to noise ratio S/N in a range from 10 dB to 0 dB (for i. i. d. noise). We then applied both the original SCHARR and its corresponding SNA-filters.

[13] The diverging and translating tree sequence has been taken from Barron's web-site and the Yosemite sequence from *http://www.cs.brown.edu/people/black/images.html*

[14] The density of an optical flow field is defined as the percentage of estimated flow vectors which have been used for computing the AAE with respect to all estimated flow vectors.

As expected and shown in Fig. 7.4, the SNA-filters yield a better performance than the original non-adapted filters in case the image sequence being corrupted by noise for all types of ideal filters . The performance of the SCHARR filter is increased by optimizing it to the corresponding images sequence. We can conclude that for these cases the optimum shape of the filter is mainly determined by the signal and noise characteristics, whereas for higher signal to noise ratios the systematical optimization plays a greater role.

7.5 Model selection and intrinsic dimension

7.5.1 Intrinsic dimension of multispectral signals

A basic model selection is the classification of signals according to their intrinsic dimension. Based on Eq. (7.1) the intrinsic dimension [136] is defined by the dimension of the subspace E to which the signal is confined. More precisely, the intrinsic dimension of an n-dimensional signal s is $n-d$ if s satisfies the constraint in Eq. (7.1) [78, 85]. Therefore, when estimation subspaces, it is essential to know the intrinsic dimension of the signal.

Furthermore, the intrinsic dimension is relevant to image and video coding due to the predominance of regions with intrinsic dimension 0 and 1 in natural images [137], and the fact that images and videos are fully determined by the regions with intrinsic dimension 2, i.e. the whole image information is contained in the 2D regions [5, 76].

The intrinsic dimension of scalar images can be estimated with differential methods based, for example, on the structure tensor, the Hessian, and the energy tensor. All these methods have been generalized to vector-valued (multispectral) images in [85]. A further extension is based on the concept of fractional intrinsic dimension for combinations of subspaces [81], e.g. multiple-motion layers. More general approaches for estimating the intrinsic dimension are based on the compensation principle [136] and the Volterra-Wiener theory of nonlinear systems [60].

7.5.2 Model selection by residual analysis

In local optical flow estimation under parametric model assumptions, the parameters of optical flow models are allowed to depend on non-trivial subsets of the spatiotemporal volume. The exploitation of the full potential of this approach involves the problem of selecting appropriate motion models for each of these subsets. While a simple optical flow model fails to approximate data of higher intrinsic complexity under low noise conditions, a complex model

is prone to over-fitting in the presence of noise. Various information criteria (e.g. AIC [2], BIC [107]) have been proposed that penalize model complexity in order to avoid over-fitting. In the context of motion estimation, the model selection problem has been discussed by Wechsler et al. [129] as well as by Gheissari et al. [36]. Gheissari et al. point out that "[...] *the available information theoretic model selection criteria are based on the assumptions that noise is very small and the data size is large enough*" and that this assumption is often violated in computer vision applications [36]. Hence, they suggest to consider the constraint surfaces of parametric models as thin plates and to penalize the strain energy of these plates according to a physical model. As this penalization incorporates only second order derivatives of the model surfaces, it cannot be used to distinguish different linear models. Moreover, if information on the distribution of noise is available from camera calibration measurements or noise estimation, probabilistic model selection criteria that incorporate this information should be employed. Residual analysis fills the gap between information theoretic penalization and heuristic surface modeling. Following the general idea of Cootes et al. [15], we suggest in [3] to assess parametric optical flow models by measuring the discrepancy between the empirical distribution of regression residuals and the pdf predicted from theory. The additive Errors-in-Variables (EIV) model claims the existence of a true signal $\tau : \Omega \to \mathbb{R}$ and, for all $\boldsymbol{x} \in \Omega$, a random variable $\epsilon(\boldsymbol{x})$ (noise) such that

$$\forall \boldsymbol{x} \in \Omega : s(\boldsymbol{x}) = \tau(\boldsymbol{x}) + \epsilon(\boldsymbol{x}) \ . \tag{7.49}$$

Optical flow estimation under the assumption of the BCCE is performed on the partial derivatives of the signal which are approximated by linear shift invariant operators. The overlap of the finite impulse response masks of these operators in the computation of derivatives at nearby pixels introduces correlation to the entries of the data matrix D_s and data term $\boldsymbol{d}_s$ used in TLS estimation. As these entries are linear in the derivatives, they can be decomposed with respect to Eq. (7.49) into

$$D_s(\boldsymbol{x}) = D_\tau(\boldsymbol{x}) + D_\epsilon(\boldsymbol{x}) \quad \text{and} \quad \boldsymbol{d}_s(\boldsymbol{x}) = \boldsymbol{d}_\tau(\boldsymbol{x}) + \boldsymbol{d}_\epsilon(\boldsymbol{x}) \ . \tag{7.50}$$

Equilibration as proposed by Mühlich [71] is used to derive from the covariance matrices of the vectors $\mathrm{vec}([D_s(\boldsymbol{x}), \boldsymbol{d}_s(\boldsymbol{x})])$ (column-wise vectorization of the matrix $[D_s(\boldsymbol{x}), \boldsymbol{b}_s(\boldsymbol{x})]$) square equilibration matrices $W_L(\boldsymbol{x})$ and $W_R(\boldsymbol{x})$ to estimate $\hat{\boldsymbol{p}}(\boldsymbol{x})$ by TLS on the data $W_L(\boldsymbol{x})[D_s(\boldsymbol{x}), \boldsymbol{d}_s(\boldsymbol{x})]W_R^T(\boldsymbol{x})$ instead of $[D_s(\boldsymbol{x}), \boldsymbol{d}_s(\boldsymbol{x})]$. $W_R^T(\boldsymbol{x})\hat{\boldsymbol{p}}(\boldsymbol{x})$ is then taken as an estimate of the initial problem. If the distribution of noise in the signal is known, regression residuals can be tested to be in accordance with the theoretically expected distribution. Given ETLS estimates $\hat{\boldsymbol{p}} : \Omega \to \mathbb{R}^k$ (k being the number of model parameters), the residuals are given by the mapping $\hat{\boldsymbol{r}} : \Omega \to \mathbb{R}^m$ (m being the number of pixels in the neighborhood for which a consistent model is assumed) such that

$$\hat{\boldsymbol{r}} := W_L[D_s, \boldsymbol{d}_s]W_R^T \begin{pmatrix} \hat{\boldsymbol{p}} \\ -1 \end{pmatrix} \ . \tag{7.51}$$

In principle, the theoretical pdf of these residuals is determined by the joint pdf of the entries of D_s and $\boldsymbol{d}_s$. The latter is obtained from the EIV model, the motion models, and the derivative operators. However, there is a direct influence to the residual pdf by the factor $[D_s, \boldsymbol{d}_s]$ as well as an indirect influence by the pdf of the estimates $\hat{\boldsymbol{p}}$. In the following, we assume $\hat{\boldsymbol{p}}$ to be deterministic. Then, the residuals Eq. (7.51), expressed as

$$\forall \boldsymbol{x} \in \Omega : \quad \hat{\boldsymbol{r}} = \underbrace{\left(\begin{pmatrix} \hat{\boldsymbol{p}} \\ -1 \end{pmatrix}^T W_R \otimes W_L \right)}_{=:\, R} \operatorname{vec}([D_s, \boldsymbol{d}_s]) \ , \tag{7.52}$$

are obtained from the deterministic linear mapping defined by the matrix R, applied to the vector $\operatorname{vec}([D_s, \boldsymbol{d}_s])$ of which the covariance matrix C is known. The covariance matrices of the residual vectors are therefore given by $C_r := \operatorname{cov}(\hat{\boldsymbol{r}}) = RCR^T$. From the Cholesky factorization $LL^T = C_r$ follows that $\hat{\boldsymbol{r}}' := L^{-1}\hat{\boldsymbol{r}}$ is decorrelated i.e.,

$$\operatorname{cov}(\hat{\boldsymbol{r}}') = \mathbb{1}_m \ , \tag{7.53}$$

while $\mathbb{E}(\hat{\boldsymbol{r}}') = L^{-1}R\mathbb{E}(\operatorname{vec}([D_s, \boldsymbol{d}_s]))$. From Eq. (7.50) follows $\mathbb{E}(\operatorname{vec}([D_s, \boldsymbol{d}_s])) = \operatorname{vec}([D_\tau, \boldsymbol{d}_\tau]) + \mathbb{E}(\operatorname{vec}([D_\epsilon, \boldsymbol{d}_\epsilon]))$. Under the assumption that the entries of $[D_\epsilon, \boldsymbol{d}_\epsilon]$ have zero mean, it follows

$$\mathbb{E}(\hat{\boldsymbol{r}}') = L^{-1}W_L[D_\tau, \boldsymbol{d}_\tau]W_R^T \begin{pmatrix} \hat{\boldsymbol{p}} \\ -1 \end{pmatrix} \ . \tag{7.54}$$

In practice, it depends on the appropriateness of the parametric model as well as on the empirical distribution of noise whether or not

$$[D_\tau, \boldsymbol{d}_\tau]W_R^T \begin{pmatrix} \hat{\boldsymbol{p}} \\ -1 \end{pmatrix} = 0 \tag{7.55}$$

holds, in which case Eq. (7.54) implies that

$$\mathbb{E}(\hat{\boldsymbol{r}}') = 0 \ . \tag{7.56}$$

If, in addition, noise is i.i.d. according to a normal distribution with known variance then, it follows from Eq. (7.53) and Eq. (7.56) that the entries of the decorrelated residual vector $\hat{\boldsymbol{r}}'$ from ETLS estimation form a set of independent standard normally distributed random variables. We therefore suggest to test this set of residuals to be standard normally distributed. Deviations from the standard normal distribution are then taken as indications of inappropriateness of the motion model. Testing for this deviation is performed by the Kolmogorov-Smirnov test, Pearson's χ^2 test, the Anderson-Darling test as well as the absolute difference of the vectors of the first j moments of the

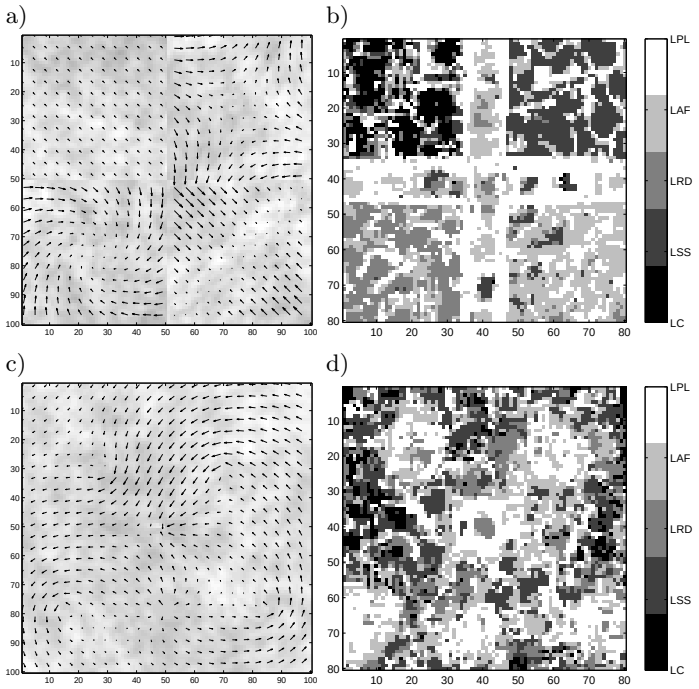

Fig. 7.5. Model selection from 11x11x3 motion neighborhoods of simulated sequences at 0.5% noise-to-signal amplitude ratio by comparison of 5 moments of the residual distribution. a) displacement field of the types sequence, b) according model selection, c) displacement field of the current sequence, d) according model selection.

empirical and theoretical distribution. In order to specifically test for properties of the model selector, we generated a variety of sequences from given two-dimensional displacement fields by warping of an initial frame. Grayvalue structure on multiple scales was introduced to this frame in order to avoid the aperture problem. Zero mean Gaussian noise was added to the sequences. Results of model selection from the models LC (locally constant), LSS (local shear/stretch), LRD (local rotation/divergence), LAF (local affine) and LPL (local planar) are shown in Fig. 7.5 for a sequence featuring motion patterns of different parametric form (top) as well as for a simulated continuous current (bottom). From the different shading in Fig. 7.5b, it can be seen that model selection is in accordance with the true displacement field. Motion patterns are identified correctly. The incidental choice of overly complex models is explained by the fact that a higher order model with the additional parameters correctly estimated as zero cannot be distinguished from the simpler model by means of residual analysis. The most complex model is correctly selected at motion discontinuities.

7.5.3 Bayesian model selection

In the following we consider energy functionals of the form

$$E(\boldsymbol{u}) = \int \gamma_1 \psi_1(\alpha_1(\boldsymbol{u})s) + \gamma_2 \psi_2(\alpha_2(\boldsymbol{u})s) + \beta \psi_3(|\nabla \boldsymbol{u}|^2) \tag{7.57}$$

where $\psi_i(\alpha_i(\boldsymbol{u})s)$, $i = 1, 2$ are different data terms and we define $|\nabla \boldsymbol{u}|^2 = |\nabla u_1|^2 + |\nabla u_2|^2$. The goal is to determine all hyper-parameters $\boldsymbol{\gamma} = (\gamma_1, \gamma_2), \beta$ directly from the data, i.e. the relative weights of the data term as well as the regularization term are to be estimated simultaneously with the optical flow. For a detailed description of the method we refer to [59].

Thus, different data models are selected for the given data set. For estimating the hyper-parameter we explore the well known relation between variational and probabilistic methods, i.e. each (discrete approximated) energy functional can be interpreted as the energy of the posterior pdf $p(\boldsymbol{u}|\boldsymbol{g}, \boldsymbol{\gamma}, \beta)$ of the optical flow. The main idea to estimate the hyper-parameters are to explore the evidence framework, that has been developed in [63] and firstly been applied to motion estimation in [57]. The hyper-parameters are estimated by a MAP estimate of the posterior pdf $p(\boldsymbol{\gamma}, \beta|\boldsymbol{g}) \propto p(\boldsymbol{g}|\boldsymbol{\gamma}, \beta)p(\boldsymbol{\gamma}, \beta)$ of the hyper-parameters given the observed gradient of the signal. The derivation of the of the likelihood $p(\boldsymbol{g}|\boldsymbol{\gamma}, \beta)$ require some approximation steps whose detailed derivation can be found in [55, 56, 59] leading to the approximated likelihood function

$$p(\boldsymbol{g}|\boldsymbol{\gamma}, \beta, \hat{\boldsymbol{u}}) = \frac{(2\pi)^N}{Z_L(\boldsymbol{\gamma}) Z_p(\beta) \det \mathbf{Q}^{\frac{1}{2}}} \exp\left(-\hat{J}\right) \quad , \tag{7.58}$$

where $\hat{\boldsymbol{u}}$ denotes the optical flow field that maximizes the posterior pdf $p(\boldsymbol{u}|\boldsymbol{g}, \boldsymbol{\gamma}, \beta)$, $\hat{J}$ the energy of the joint pdf $p(\boldsymbol{g}, \boldsymbol{u}|\boldsymbol{\gamma}, \beta)$ taken at $\hat{\boldsymbol{u}}$ and $Z_L(\boldsymbol{\gamma}), Z_p(\beta)$ denote the partition functions of Gaussian distributions. The matrix $\mathbf{Q}$ denotes the Hessian of the joint pdf energy $J(\boldsymbol{u}, \boldsymbol{g})$ taken at the maximum of the posterior pdf $p(\boldsymbol{u}|\boldsymbol{g}, \boldsymbol{\gamma}, \beta)$. Since $\hat{\boldsymbol{u}}$ itself depends on the hyper-parameters $\boldsymbol{\gamma}, \beta$ we have to apply an iterative scheme for estimating the optical flow field and the hyper-parameters simultaneously, i.e. we estimate the optical flow for fixed hyper-parameters and estimate then the hyper-parameters using the previously estimated optical flow.

$$\begin{aligned} \boldsymbol{u}^{k+1} &= \arg\min_{\boldsymbol{u}} \left\{ p(\boldsymbol{u}|\boldsymbol{g}, \hat{\boldsymbol{\gamma}}^k, \beta^k) \right\} \\ \boldsymbol{\gamma}^{k+1} &= \arg\max_{\boldsymbol{\gamma}} \left\{ p(\boldsymbol{\gamma}, \beta^k|\boldsymbol{g}) \right\} \\ \beta^{k+1} &= \arg\max_{\beta} \left\{ p(\boldsymbol{\gamma}^k, \beta|\boldsymbol{g}) \right\} \quad . \end{aligned} \tag{7.59}$$

This procedure is repeated until convergence.

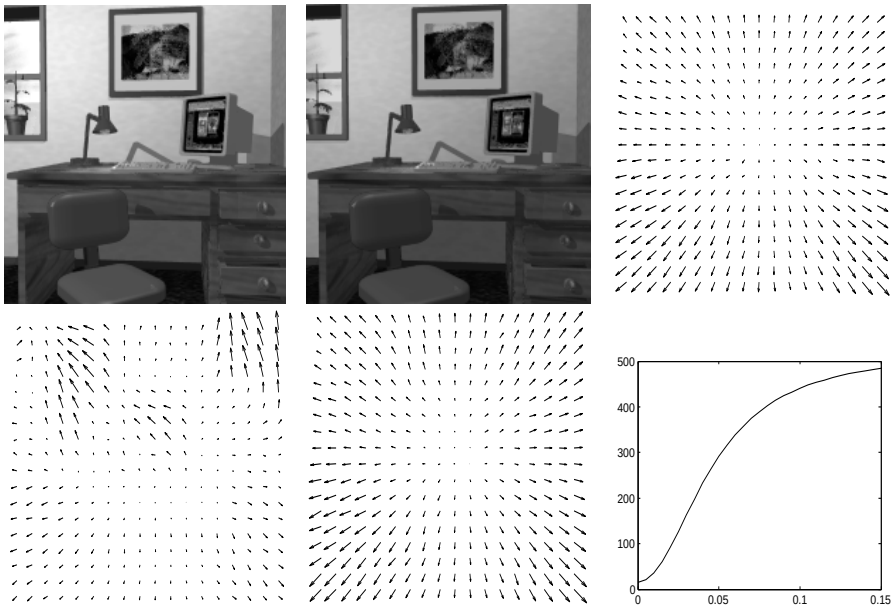

Fig. 7.6. Upper figures (from left to right): first frame of the 'Office' sequence; second frame of the 'Office sequence' with a brightness decay; estimated flow field using $\gamma_1 = 1$ and $\gamma_2 = 0$; Lower figures(from left to right):estimated flow field using $\gamma_1 = 1$ and $\gamma_2 = 0$; estimated flow field using the Bayesian model selection approach for; ratio of both estimated likelihood hyper-parameters γ_2/γ_1 vs. the gradient of the brightness change.

7.6 Anisotropic regularization for orientation estimation

So far, we focused on modeling, model selection and estimation of parameters in multidimensional signals. A major source of estimation inaccuracy is due to the data being noisy, or corrupted, and thus not fulfilling the constraint equations selected. As long as the data belongs to the same population or distribution, one can reduce the influence of noise by smoothing, i.e. grouping measurements belonging together. For single orientation or single motion data this can be done by anisotropic diffusion with a diffusion tensor. A suitable diffusion process is described in Sec. 7.6.1. As parameter selection is non-trivial, a learning approach interpreting the data as a Markovian random field may be advantageous (see Sec. 7.6.2). This approach does not end in anisotropic diffusion in the same way as it is usually applied for data denoising. Anisotropic diffusion with a diffusion tensor can be derived from a cost functional where the *expectation* of the motion constraint Eq. (7.5). This is not only of theoretic interest and opens a connection to stochastic partial differential equations,

but also allows to construct diffusion-like denoising schemes for other (linear) models, e.g. two transparently overlaid motions (see Sec. 7.6.3).

7.6.1 Flow adapted anisotropic diffusion

Anisotropic diffusion filtering evolves the acquired, noisy initial multidimensional signal $s(\boldsymbol{x}, 0)$ via an evolution equation:

$$\frac{\partial s}{\partial t} = \nabla \cdot (D \nabla s) \; . \tag{7.60}$$

Here D is the diffusion tensor, a positive definite symmetric matrix and $s(\boldsymbol{x}, t)$ is the evolving signal. Here t is diffusion time not to be confused with the time coordinate x_3 if s is an image sequence. As we will see in Sec. 7.6.3 this diffusion is appropriate for signals with up to single orientation only.

The diffusion tensor D usually applied in anisotropic diffusion uses the same eigenvectors e_i as the structure tensor J_ρ (see Eq. (7.22)) constructed for single orientation (Eq. (7.2)) or single motion constraint (Eq. (7.5)). Thus smoothing is applied according to the signal structures. Smoothing strengths along these structures are given by the corresponding eigenvalues λ_i of D. Given a diagonal matrix L with $L_{ii} = \lambda_i$, the diffusion tensor D is given by

$$D = (\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)\, L\, (\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)^T \; . \tag{7.61}$$

The directional diffusivities $\lambda_i, i \in \{1, \ldots, n\}$ determine the behavior of the diffusion.

For denoising they shall be large for small eigenvalues μ_i of J_ρ and vice versa. For *orientation-enhancing anisotropic diffusion* introduced in [100], all λ_i are calculated according to the same rule. This is in contrast to the well established *edge-enhancing anisotropic diffusion*, where one λ is fixed to 1, enforcing single orientation everywhere, even if the structure tensor indicates model violation.

One of the major problems in anisotropic diffusion application is to find an appropriate stopping criterion. For optical flow the reliability of the estimate can be determined by a simple normalized confidence measure. It initially rises when the input data is smoothed and decays or reaches a plateau when the data is over-smoothed. Consequently we stop the diffusion when this measure stops to raise significantly. Details can be found in [100, 112]. A typical smoothing result is shown in Fig. 7.7.

7.6.2 Anisotropic diffusion as maximum likelihood estimator

A crucial ingredient to successfully use anisotropic diffusion for image denoising is the appropriate selection of directional diffusivities λ_i (cf. Eq. (7.61)).

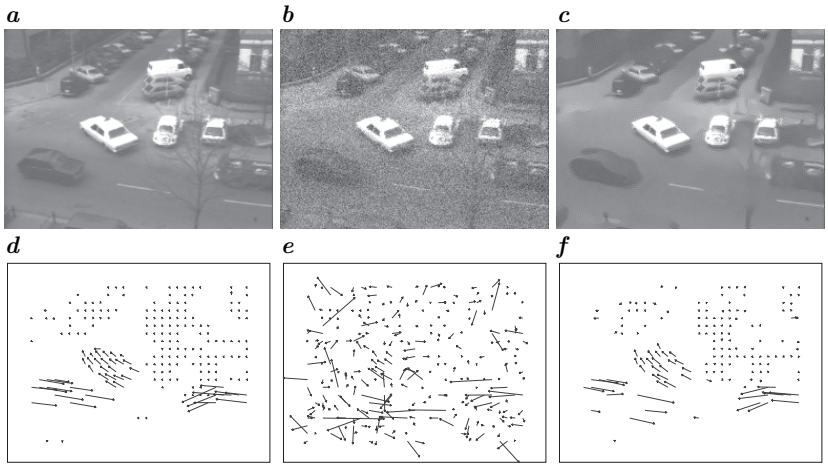

Fig. 7.7. Hamburg taxi sequence: **a** one frame in the original sequence, **b** the same frame with noise (std. dev. $\sigma = 20$) added and **c** reconstructed frame. **d** Velocity field on the original data, **e** on the noisy data and **f** on the reconstructed data.

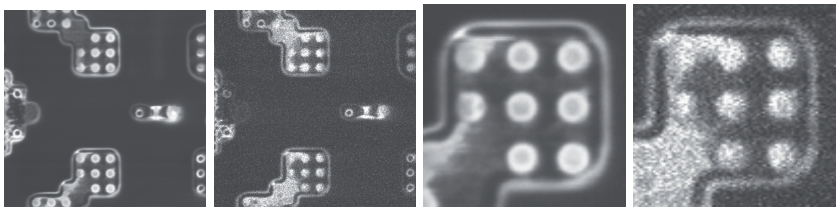

Fig. 7.8. Noise-free and noisy silicon chip images acquired by a focused-ion-beam tool. From left to right: high quality image, lower quality image, sub-regions of high and low quality images.

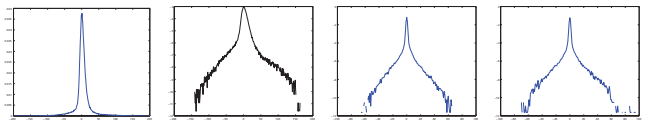

Fig. 7.9. Empirical image noise statistics for silicon chip images. **(a)** distribution of image noise $(f_i - g_i)$. **(b)** log of image noise distribution. **(c, d)** log probability of horizontal and vertical image derivatives.

In [101] it turned out that isotropic nonlinear diffusion and anisotropic diffusion correspond to isotropic and directional statistical models, respectively. Having training data at hand, as e.g. noisy images f_i and noise-free images g_i derived in different operation modes of a focused-ion-beam tool (cf. Fig. 7.8), image statistics are computable as histograms (cf. Fig. 7.9).

The problem of recovering the image g from f can then be posed as the maximization of

$$p(g|f) \propto \prod_i \left(p(f_i|g_i) \prod_{j=1}^{J} p(\boldsymbol{n}_j \nabla g_i) \right) \tag{7.62}$$

where $p(g|f)$ approximates the posterior probability of the image g conditioned on the observed, noisy, image f. The likelihood term, $p(f_i|g_i)$, at every pixel, i, is defined by noise statistics. The spatial prior term exploits a Markov Random Field assumption [35] which defines the prior in terms of local neighborhood properties. Here it is defined in terms of the spatial derivatives, ∇g_i, at a pixel i, in J different directions $\boldsymbol{n}_j$, and uses learned image statistics to assess the prior probability. If ∇g_i is computed using neighborhood differences (as in Fig. 7.9), then Eq. (7.62) can be viewed as a standard Markov Random Field (MRF) formulation of the regularization problem [35]. Calculating such a spatial prior as histograms of the eigenvalues μ of the structure tensor $\boldsymbol{J}_\rho$ (see Eq. (7.22)), results in anisotropic diffusion. Exploiting this relationship provides a principled way of formulating anisotropic diffusion problems and results in a fully automatic algorithm in which all parameters are learned from training data. The resulting anisotropic diffusion algorithm has many of the benefits of Bayesian approaches along with a well-behaved numerical discretization. For further details on this approach and its performance we refer to [101].

7.6.3 Anisotropic diffusion for multiple motions

Standard diffusion may be derived from a cost function

$$E(g) = \int_\Omega (g - f)^2 + \alpha |\nabla g|^2 \mathrm{d}x \tag{7.63}$$

The first term in Eq. (7.63) is usually called *data term*, corresponding to the posterior probability term in Eq. (7.62). Modeled purely quadratic is equivalent to $p(g|f)$ being zero-mean Gaussian. The second term is the *smoothness term*, corresponding to the prior probability, modeled by (the positive part of) a zero-mean Gaussian of $|\nabla g|$. An extension of this constraint has been proposed by Mumford and Shah [88]. Its connection to (not tensor driven) anisotropic diffusion can be found in [105].

In [97] a cost function penalizing violation of a linear model was introduced. Using e.g. Eq. (7.5) one gets

$$E(g) = \int_\Omega (g - f)^2 \mathrm{d}x + \alpha \int_\Omega < (\nabla^T g \boldsymbol{u})^2 > \mathrm{d}x \tag{7.64}$$

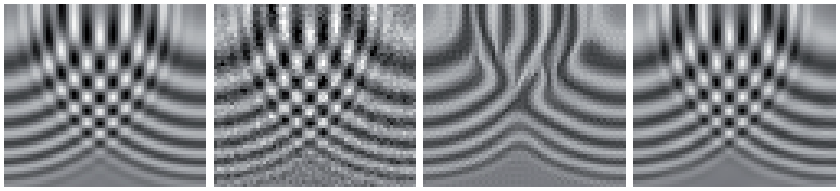

Fig. 7.10. Denoising of transparently overlaid ring pattern. From left to right: original image, image with noise added, reconstruction via edge-enhancing single orientation anisotropic diffusion, reconstruction via double orientation diffusion.

Minimizing Eq. (7.64) by iteratively fulfilling the minimality condition given by calculous of variations yields exactly anisotropic diffusion with a diffusion tensor $mD =< \boldsymbol{u}\boldsymbol{u}^T >$. In [97] it is shown that this tensor can be approximated by a diffusion tensor as constructed in Eq. (7.61). This observation allows to construct diffusion-like reconstruction schemes for linear models. Plugging the 2D equivalent of Eq. (7.71) into Eq. (7.64) and minimizing it as before yields such a scheme, which has been implemented using optimized filters for transparent motion as derived in Sec. 7.4.1. A denoising result on synthetic data is depicted in Fig. 7.10. One observes that standard edge-enhancing diffusion produced artifacts by enforcing a single orientation model, while enforcing the novel double orientation diffusion yields results visually indistinguishable from the original noise-free image.

7.6.4 Optimal integration of the structure tensor

In the following we generalized the signal and noise adapted filter approach discussed in 7.4.3 such that it is able to preserve edges (edge preserving Wiener (EPW) filter) and generalize it from scalar valued signals to tensor valued signals. For a detailed description we refer to [54].

The scalar valued edge preserving Wiener filter

The estimation of a true underlying image value s_j at position j from a linear but not shift invariant filtering of the observable image $\boldsymbol{z}$ can be written in the form $\hat{s}_j = \boldsymbol{m}_j^T \boldsymbol{z}$. Our task is to choose $\boldsymbol{m}_j$ in such a way that the filtered output $\hat{s}_j$ approximates, on an average, the desired output s_j for the error-free case as closely as possible in the least mean squares sense. Therefore, it is necessary to model the statistical properties of the signal and the noise processes, respectively. Let the noise vector $\boldsymbol{v} \in \mathbb{R}^N$ be a zero-mean random vector with covariance matrix $\mathbf{C}_{\boldsymbol{v}} \in \mathbb{R}^{N \times N}$ (which is in this case equal to its correlation matrix $\mathbf{R}_{\boldsymbol{v}}$). Furthermore, we assume that the process that generates the signal $\boldsymbol{s} \in \mathbb{R}^N$ can be described by the expectation $\boldsymbol{w}_s =$

$\mathsf{E}[\boldsymbol{s}]$ of the signal vector, and its autocorrelation matrix $\mathbf{R}_s$. Furthermore, let $\mathbf{R}_{\boldsymbol{s}s_j} \in \mathbb{R}^{1\times N}$ denote the correlation matrix between the image value s_j and the whole image $\boldsymbol{s}$. The filter $\boldsymbol{m}_j$ is then determined by minimizing the mean squared error between the estimated signal value and the actual one

$$\boldsymbol{m}_j = \arg\min_{\tilde{\boldsymbol{m}}_j} \left\{ \mathsf{E}\left[||\tilde{\boldsymbol{m}}_j^T \boldsymbol{z} - s_j||^2 \right] \right\} \ . \tag{7.65}$$

Knowing the second order statistical moments for both the noise and signal as well as the observation matrix, the Gauss-Markov theorem delivers the optimal filter (for a detailed derivative of mean squared error based filters see e.g. [50])

$$\boldsymbol{m}_j = \left(\mathbf{K}_j \mathbf{R}_{\boldsymbol{s}} \mathbf{K}_j^T + \mathbf{R}_{\boldsymbol{v}} \right)^{-1} \mathbf{K}_j \mathbf{R}_{\boldsymbol{s}s_j} \ . \tag{7.66}$$

In following, we discuss the extension of this concept to matrix valued data.

The edge preserving tensor valued Wiener filter

As already mentioned in the introduction, most important tensors for image processing are square positive (semi-)definite matrices denoted by $P(n)$ in the following where n is the size of the matrix. This set of tensors does not form a subspace of the tensor vector space. For example, multiplying a positive definite matrix by -1 yields a negative definite matrix and hence leads out of the set $P(n)$. Thus, applying image processing techniques to $P(n)$ requires additional care since even simple linear operations might destroy the basic structure of the data. In [119] the proposed nonlinear diffusion scheme is shown to preserve positiveness of the processed tensor field. An equivalent proof based on discrete filtering can be found in [130] which uses the fact that the proposed diffusion filters are convex filters. This is also the basis for the design of our tensor valued EPW-filter, i.e. we design the tensor valued EPW-filter as a convex filter. A map $\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^N$ is denoted as a convex filter (see e.g. [132]) if for each $\boldsymbol{z} \in \mathbb{R}^N$ there are weights $w_{ij}(\boldsymbol{z})$ with

$$(\mathbf{F}\boldsymbol{z})_k = \sum_{t=1}^{N} w_{kt}(\boldsymbol{z}) z_t, \quad w_{tk}(\boldsymbol{z}) \geq 0 \ \ \forall k, \ \ \sum_{t=1}^{N} w_{kt}(\boldsymbol{z}) = 1 \ . \tag{7.67}$$

If each component of the tensor-field is processed with the same convex filter, it is simple to prove the positiveness of the processed tensor field. This implies that we have to model each matrix component by the same process and thus use the same statistical model as in the scalar case for each matrix element. We have to design a filter mask whose sum is equal one and where each element is non-negative. The first requirement can easily be obtained by a proper normalization. The second requirement is not guaranteed by Eq. (7.65). In

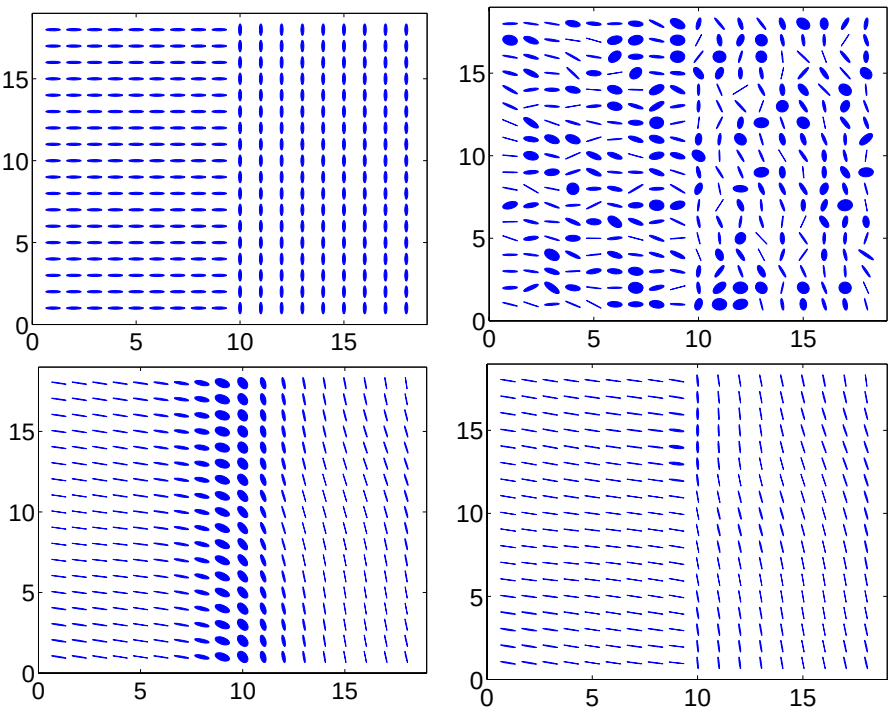

Fig. 7.11. Upper left: original tensor field; upper right: left tensor field corrupted by additive Gaussian noise ($\sigma_v = 0.3$ on each matrix element); lower left: processed tensor field by our EPW-filter with $\beta_1, \beta_2 = 0$; lower right: EPW-filter with $\beta_1 = 6$, $\beta_2 = 1$.

order to keep each element non-negative, further constraints are introduced to the optimization procedure

$$\boldsymbol{m}_j = \arg\min_{\tilde{\boldsymbol{m}}_j} \left\{ \mathsf{E}\left[||\tilde{\boldsymbol{m}}_j^T \boldsymbol{z} - s_j||^2 \right] \right\} \quad \text{such that} \quad (\boldsymbol{m}_j)_k \geq 0 \ . \tag{7.68}$$

In contrast to Eq. (7.65)), a closed form solution does not exist for the non-negative least squares problem and numerical methods (chapter 23, pp. 161 in [61]) need to be applied.

7.6.5 Tensor field integration by nonlinear diffusion

Here we present a method for adaptive integration of tensor fields with respect to motion estimation. The smoothing is fulfilled by a suitable nonlinear diffusion strategy, which is then applied on the manifold of the matrices with same eigenvalues, cf. [120].

As already discussed in [120, 121, 122], there is a general smoothing on corners, when we use the nonlinear diffusion based on total variation flow. In order to circumvent this drawback, [13] defines a coherence dependent map, which stops the diffusion near corners. We propose the formalism of curvature preserving partial differential equations(PDE's) to avoid oversmoothing on corners. This is just a result of the analysis in [120, 121, 122]. It is a direct approach following the formalism for the heat flow equation, constrained on a curve. The adaptive, curve dependent metrics drives the diffusion according to the desired behavior by itself and avoids dependency of the smoothed tensor field on the derivatives of the steering geometry.

The next theoretical part of this section is to choose a proper integration scheme, which constraints the diffusion flow on the manifold of the matrices with the same spectrum. We represent different flows and conduct experiments with the isospectral flow. In optic flow and motion estimation not only the computation of the flow field is important, but also the confidence of the estimated flow. Most of the confidence measures rely on the eigenvalues of the second order tensor in the estimated pixel. The isospectral flow leaves the eigenvalues of second order tensors untouched. This is an intrinsic property of the flow, at least analytically. This means, the confidence measure is preserved locally. That's why we decided to employ an integration scheme, based on the isospectral flow. Additionally it should be mentioned, that the isospectral flow represents a good trade-off between performance and computational costs, [120, page 149].

Diffusion tensor regularization can be used to general symmetric and semi-positive definite matrices such as structure tensors or covariance matrices.

Let $\Omega \in \mathbb{R}^n$. $T : \Omega \to P^{n\times n}$, P: positive semi-definite matrices. The multi-valued regularization process can be expressed in variational, divergence and trace-based formulation.

$$\int_\Omega \phi(\|\nabla T\|)\, dx \to \min$$

$$\frac{\partial T_i}{\partial t} = \operatorname{div}\left(\frac{\phi\prime\|\nabla T_i\|}{\|\nabla T_i\|}\nabla T_i\right), \qquad i = 1, ..., n$$

$$\frac{\partial T_i}{\partial t} = \operatorname{trace}(DH_i), \qquad i = 1, ..., n,$$

where D is the smoothing geometry and H_i is the Hessian of T_i.

We choose the discontinuity-preserving smoothing geometry

$$D := \sum a_i \frac{1}{(1+\sum \lambda_i)^p}\theta_i\theta_i^T,$$

where λ_i are the eigenvalues, θ_i are the eigenvectors of T, $a_i \in \mathbb{R}$ is an anisotropy weight and p is the discontinuity preservation amount for i=1,...,3. a_i and p are set by the user.

We constrained this diffusion process on the submanifold N of the matrices with the given set of eigenvalues by using the isospectral flow

$$\frac{\partial T}{\partial t} = [T, [T, -\mathcal{L} + \mathcal{L}^T]],$$

where L is the matrix, corresponding to an unconstrained Lagrangian, describing the regularization process.

A suitable integration scheme is

$$\begin{aligned} T_{t+dt} &:= A_t(x)^T \; T_t(x) \; A_t(x) \\ A_t(x) &:= e^{-dt[\mathcal{L}(x)\mathcal{L},T]} \; . \end{aligned} \tag{7.69}$$

Numerical experiments

We measured the performance of both techniques on a synthetic sequence without noise, containing a moving sinus pattern with discontinuity.

We achieved a small accuracy gain, compared to the isotropic smoothing, for synthetic sequences without noise by using the isospectral flow. The accuracy gain vanisches after adding Gaussian noise to the synthetic sequence, thus the method is noise sensitive. The tuning of parameters is crucial and it's difficult to find a trade-off between preserving of discontinuities and noise sensitivity of the diffusion flow. There is a thin line for the proper parameter setting, at least in our implementation of the isospectral flow by matrix exponential for the integration scheme Eq. (7.69).

7.6.6 Adaptive anisotropic filtering

Sec. 7.6 emphasized the importance of reducing the noise inherent in the data for modeling, model selection, and estimation of parameters in multidimensional signals. Standard procedures smooth the data by averaging measurements $s(\boldsymbol{x})$ over neighborhoods in the domain of s. Apparently, size and shape of the neighborhoods are crucial to the performance of such procedures.

The method of Adaptive Anisotropic Filtering, based on Adaptive Weights Smoothing [93], determines size and shape of such neighborhoods making two structural assumptions on s. First, in a neighborhood of local flow homogeneity, s takes the form of Eq. (7.1) (with $d = 2, n = 2$) and can thus be

approximated by a univariate function $f(u) \approx f(\nabla^T s \cdot \boldsymbol{x})$. A second assumption of local linearity regarding f leads to an approximation of s of the form

$$s(\boldsymbol{x}) \approx a + c\beta^T(\boldsymbol{x} - \boldsymbol{x}_0), \tag{7.70}$$

where β denotes a vector parallel to the gradient of s.

Noise reduction in the data with respect to the above formulation can be achieved by estimating the parameters a for each data point. This can be accomplished in an iterative procedure where an initial estimate of β determines an elliptic neighborhood which is used to estimate the parameters a and c. In a subsequent step the parameters a and c are used to improve the estimate of β and so on. In each iteration the elliptic neighborhoods are increased until further extension of the neighborhoods will consider data that significantly deviate from the estimated model with respect to a given threshold.

7.7 Estimation of multiple subspaces

7.7.1 Introduction to multiple subspaces

Multiple superimposed subspaces as defined in Sec. 7.2.2 can be estimated by extending the constraint in Eq. (7.2) based on the observation that all directional derivatives within the different subspaces must equal zero. Therefore, the concatenated, directional derivatives of s must equal zero, a condition that leads to a nonlinear constraint on the components of the $\boldsymbol{u}_i$. This constraint can be linearized by introducing the mixed-motion or mixed-orientation parameters (MMPs and MOPs), which then need to be separated such as to yield the $\boldsymbol{u}_i$´s.

For the case of scalar image sequences with two additively superimposed motions, such a constraint was first used in [108]. After successively applying $\alpha(\boldsymbol{u}_1)$ and $\alpha(\boldsymbol{u}_2)$ to Eq. (7.7), the resulting constraint $\alpha(u)\alpha(v)f = 0$, is linear in the MMPs, i.e. we obtain a model that is linear in its non-linear parameters. The non-linear parameters can then be estimated using linear estimation techniques for the MMPs, which are then separated by solving a complex polynomial as shown below.

Here we summarize the comprehensive results obtained for multiple motions in videos and multiple orientations in images. The final goal, however, is the estimation of any combination of any number of subspaces with arbitrary intrinsic dimensions and signal dimension. First steps toward a comprehensive classification of such combinations have been done in [79, 80, 81]. In [86, 118] the problem of estimating multiple orientations in n-dimensional signals has been solved.

7.7.2 Multiple motions

Analytical solutions for multiple motions

First consider two additive motion layers s_1, s_2 as defined by Eq. (7.7). The constraint for the velocities Eq. (7.71) becomes

$$\alpha(\boldsymbol{u}_1) \circ \alpha(\boldsymbol{u}_2) \circ s = c_{xx}s_{xx} + c_{xy}s_{xy} + c_{xt}s_{xt} + c_{yy}s_{yy} + c_{yt}s_{yt} + c_{tt}s_{tt} = 0, \tag{7.71}$$

with MMPs defined by:

$$\begin{aligned} c_{xx} &= u_{1,x}u_{2,x},\ c_{yy} = u_{1,y}u_{2,y},\ c_{xy} = (u_{1,x}u_{2,y} + u_{1,y}u_{2,x}) \\ c_{xt} &= (u_{1,x} + u_{2,x}),\ c_{yt} = (u_{1,y} + u_{2,y}),\ c_{tt} = 1. \end{aligned} \tag{7.72}$$

Eq. (7.71) is non-linear in the motion parameters themselves, but linear in the MMPs, which therefore can be estimated by standard linear techniques, e.g. [77, 115].

MMP decomposition with complex polynomials

In [77] a general solution for decomposing an arbitrary number of superimposed motions has been proposed. Here we sketch the idea for the case of two motions. The interpretation of motion vectors as complex numbers $\boldsymbol{u} = u_x + iu_y$ enables us to find the motion parameters as the roots of the complex polynomial

$$Q(z) = (z - \boldsymbol{u}_1)(z - \boldsymbol{u}_2) = z^2 - (c_{xt} - ic_{yt})z + (c_{xx} - c_{yy} + ic_{xy}), \tag{7.73}$$

whose coefficients are expressed in terms of the MMPs. The generalization of this approach to N overlaid motion layers is straightforward.

Solutions for multiple motions based on regularization

A major benefit of the above approach to multiple motions is that it involves a linearization of the problem such that it becomes mathematically equivalent to the problem of estimating only one motion. As a consequence, the regularization methods used for single motions can be applied. In [115, 116], the well-known algorithm for single-motion estimation proposed by Horn and Schunck has been extended to the case of multiple motions by the use of the following regularization term:

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

The MMPs c are obtained as the values that minimize the above term together with the squared optical-flow term in Eq. (7.71), i.e.

$$\int\int (\alpha(\boldsymbol{u}_1) \circ \alpha(\boldsymbol{u}_2) \circ s)^2 + \lambda^2 N \, \mathbf{d}\Omega.$$

λ is the regularization parameter and Ω the image plane over which we integrate. Note that working on the MMPs has the great advantage that we obtain an Euler-Lagrange system of differential equations that is linear, which would not be the case, when working directly on the motion vectors themselves.

Block-matching for multiple motions

The framework for multiple motions has been extended such as to include block-matching techniques for estimating an arbitrary number of overlaid motions [114]. To estimate N motions, $N+1$ images of the sequence are needed. A regularized version of the algorithm has also been derived based on Markov Random Fields [117], and the increased robustness has been demonstrated.

Separations of motion layers

A benefit of multiple-motion estimation is that the parameters of the multiple motions can be used to separate the motion layers. This problem has been solved in the Fourier domain, where the inherent singularities can be better understood and interpolated than in the spatio-temporal domain [83, 115].

Occluded motions

For occluded motions as defined by Eq. (7.9), one obtains the constraint

$$\alpha(\boldsymbol{u}_1) \circ \alpha(\boldsymbol{u}_2) \circ s = -\alpha(\boldsymbol{u}_1) \circ \left[\chi(\boldsymbol{x} - t\boldsymbol{u}_1)\right] (\boldsymbol{u}_1 - \boldsymbol{u}_2) \cdot \nabla s_2(\boldsymbol{x} - t\boldsymbol{u}_2). \quad (7.74)$$

In [84], it has been shown that

$$\alpha(\boldsymbol{u}_1) \circ \alpha(\boldsymbol{u}_2) \circ s = q(\boldsymbol{x}, t, \boldsymbol{u}_1, \boldsymbol{u}_2) \delta(B(\boldsymbol{x} - t\boldsymbol{u}_1)), \quad (7.75)$$

where

$$q(\boldsymbol{x}, t, \boldsymbol{u}_1, \boldsymbol{u}_2) = -(\boldsymbol{u}_1 - \boldsymbol{u}_2) \cdot \boldsymbol{N}(\boldsymbol{x} - t\boldsymbol{u}_1) (\boldsymbol{u}_1 - \boldsymbol{u}_2) \cdot \nabla s_2(\boldsymbol{x} - t\boldsymbol{u}_2). \quad (7.76)$$

$B(\boldsymbol{x}) = 0$ determines the occluding boundary of $\boldsymbol{N}(\boldsymbol{x}) = \nabla B(\boldsymbol{x})$ is the unit normal to the boundary. Eq. (7.75) shows that the occlusion distortion is (i) restricted to the occluding boundary, (ii) minimal when the normal to the boundary is orthogonal to the relative motion (the difference between fore- and background motions) and maximal when the two vectors are aligned, (iii) proportional to the intensity gradient of the background pattern. By a Fourier analysis of occluded motions, it has been revealed that the decay of the distortion is hyperbolic for both straight and curved boundaries and the exact expression for the distortion term has been derived for the case of straight boundaries [84]. Based on these results, a hierarchical motion-estimation algorithm has been designed that obtains very good estimates at the occluding boundary by avoiding the there localized distortion [6, 7, 82, 84].

7.7.3 Multiple orientations

Orientation estimation in tensor form

If m orientations in n-variate signals are to be found, this problem can either be written using the mixed orientation parameters (MOP) vector or, alternatively, in tensor notation. The latter form was presented in [86], including the generalization to p-dimensional signals, for instance color or multi-spectral images, which will not be handled here.

In order to express orientation estimation in tensor form, we first define the sought entity, the orientation tensor, as outer product of all individual orientation unit vectors $\boldsymbol{u}_i$:

$$\mathcal{U} = \boldsymbol{u}_1 \otimes \cdots \otimes \boldsymbol{u}_n \; . \tag{7.77}$$

With the tensor scalar product

$$\langle \mathcal{O}, \mathcal{U} \rangle := \sum_{k_1,\ldots,k_m=1}^{n} (\mathcal{O})_{k_1 \cdots k_m} (\mathcal{U})_{k_1 \cdots k_m} \; , \tag{7.78}$$

we can define the data constraint as

$$\langle \mathcal{O}, \mathcal{U} \rangle = 0 \tag{7.79}$$

where

$$(\mathcal{O})_{k_1 \cdots k_m} = \prod_{i=1}^{m} \frac{\partial s}{\partial x_{k_i}} \tag{7.80}$$

is the data tensor generated for the occluding orientations model. In a similar way, the additive superposition of multiple orientations leads to

$$\langle \mathcal{T}, \mathcal{U} \rangle = 0 \quad \text{with} \quad (\mathcal{T})_{k_1 \cdots k_m} = \frac{\partial^m s}{\partial x_{k_1} \cdots \partial x_{k_m}} \; . \tag{7.81}$$

Symmetry properties of the data tensors

The commutativity in the definitions of Eq. (7.79) and Eq. (7.81) is the key to the understanding of multiple orientations. The data tensors are invariant against any arbitrary permutation of indices and therefore have some very pronounced symmetry properties.

For $m = 2$, the data tensors $\mathcal{O}$ and $\mathcal{T}$ are symmetric $n \times n$ matrices, but for higher m, concepts from matrix algebra will not suffice and a tensor notation becomes necessary. We therefore define the *space of fully symmetric m-tensors* which are invariant to *any arbitrary* permutation of indices – and not just to some special permutations only. We define

$$\mathbb{R}_{\oplus}^{n\times\cdots\times n} = \left\{ \mathcal{T} \in \mathbb{R}^{n\times\cdots\times n} \middle| (\mathcal{T})_{i_1\cdots i_m} = (\mathcal{T})_{P(i_1\cdots i_m)} \right\} \tag{7.82}$$

with $P(i_1 \cdots i_m)$ denoting any arbitrary permutation of the indices $i_1 \cdots i_m$.

For both models – occluding or additive superposition –, the resulting data tensors are fully symmetric and from this symmetry property follows an import consequence: the sought orientation tensor cannot be estimated uniquely, but only up any arbitrary permutation of indices. One cannot distinguish between a "first" orientation, "second" orientations and so on; all we can compute a *set* of m orientations.

Fortunately, this problem can be resolved by restricting the sought orientation tensor to those tensors which are invariant to index permutations, i.e. to $\mathbb{R}_{\oplus}^{n\times\cdots\times n}$. Within this set, the solution becomes uniquely determined – at least in non-degenerate cases.

Estimation of the orientation tensor

Stacking the independent elements of data tensor $\mathcal{O}$ (resp. $\mathcal{T}$) and of the symmetrized (and therefore uniquely determined) orientation tensor $\mathcal{U}$ generates two vectors which can be understood as generalization of the double-orientation constraint handled in [1]. See [86] for details.

In many applications, for instance in feature extraction for tracking, this is already sufficient. If, on the other hand, a decomposition into the underlying individual orientations is necessary, then an additional problem arises: the orientation tensor estimated so far is overparameterized because 'true' orientation tensors must belong to the set

$$\mathbb{R}_{\circledast}^{n\times\cdots\times n} = \left\{ \sum_{P(i_1\cdots i_m)} \boldsymbol{u}_{i_1} \otimes \cdots \otimes \boldsymbol{u}_{i_m} \middle| \boldsymbol{u}_{i_1}, \ldots, \boldsymbol{u}_{i_m} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\} \right\} \tag{7.83}$$

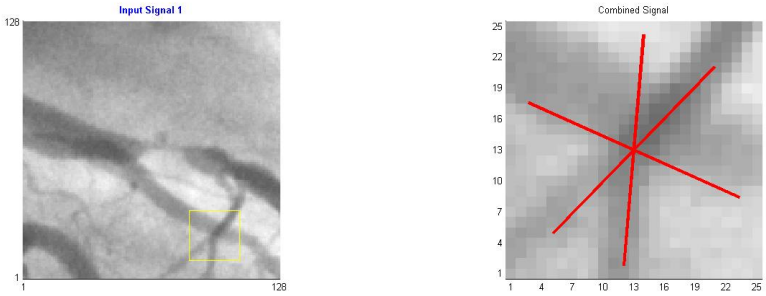

Fig. 7.12. Modelling blood vessels in x-ray images often requires *three* orientations. Left image: input image. Right image: region of interest (light square in left image) and detected orientations.

which we will denote as set of *minimal fully symmetric tensors*. They are constructed by summing up all possible permutations of outer products and obviously form a subset of $\mathbb{R}^{n\times\cdots\times n}_{\oplus}$.

In [86], methods for estimating an arbitrary number of orientations in images (arbitrary m for $n = 2$) and for double orientation estimation ($m = 2$) in general n-variate signals are presented. The second problem boils down to the approximation of a general $n \times n$ matrix by a rank-2 matrix and the first problem, the estimation of multiple orientations in images, can be formulated as finding the roots of a degree m real polynomial.

Especially the estimation of more than two orientations in images has many applications in image processing and computer vision, for instance in industrial quality control or medical image processing. Fig. 7.7.3 shows an x-ray image of superposed blood vessels and the estimated orientations.

7.8 Applications

7.8.1 Transport processes

The transport of energy, mass and momentum is one ubiquitous phenomenon in our world, spanning all branches of science. It is one condition for the continued existence of complex dynamic systems. As such, only through this transport living organisms can coordinate and keep upright their metabolism and other processes of life. Plants, for example, have developed complex vascular systems (the xylem) to transport nutrients and water from the roots to the rest of the plant. At the same time, all living tissue is continuously supplied with energy from the leaves, in form of the organic products of photosynthesis.

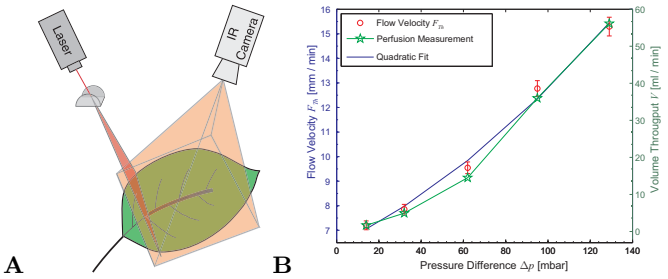

Fig. 7.13. The set-up for measuring the xylem flow in plant leaves with active thermography in **A** and the results of a ground truth measurement in **B**.

In plant physiology, it is a longstanding question how water and nutrients are transported in the xylem of the plant's leaf and which external factors influence it. While bulk dependencies are known, a detailed analysis has eluded research due to inadequate measurement techniques. The transport processes in leaves are especially important since it is known that they can be regulated by plants according to environmental forcing. Also, due to these regulatory mechanisms, the plant can cope with cuts in the leaf and still supply areas affected by these cuts with xylem sap by other pathways. Still, very little is known about these mechanisms. To shed light on the transport of the xylem inside leafs the advanced techniques presented in this chapter have been employed on thermographic image sequences of heated water parcels in plant leaves [31]. The experimental set-up as well as a comparative measurement to ground truth is presented in Fig. 7.13. Through these measurements, quantitative measurements could be made in different parts of the leaf in dependence of external environmental forcings acting on the leaf [31].

On much smaller scales, a current trend in chemical and biochemical analytics as well as in medical diagnostics is the move to microfluidic mixers and "lab-on-a-chip" applications. Huge surface to volume ratios are achievable by micro channels with highly controlled boundary conditions, leading to more efficient reaction kinetics with less by-products. Even on these minute structures, the transport of energy, mass and momentum as well as the measurements thereof is vital for a better understanding of the processes involved. In these flow regimes, molecular tagging velocimetry (MTV) is an alternative approach to the standard technique of micro particle imaging velocimetry (μPIV) for measuring fluid flows. In MTV, a pattern is written to the fluid with an UV laser to uncage dyes and thus making them fluorescent. Although this approach has been used since the late eighties of the last century, measuring fluid flow with this technique has had one significant drawback: Due to the flow profile, the uncaged dyes would appear to diffuse in a process termed Taylor dispersion. Two frames of a typical image sequence can be seen in Fig. 7.14. This dispersive process leads to significant uncertainties in the measurements,

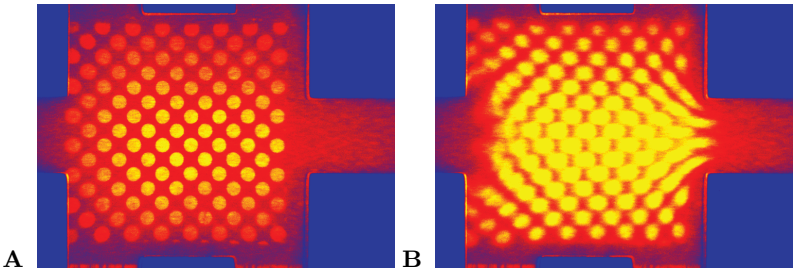

Fig. 7.14. In **A** and **B** two frames of a microfluidic image sequences are shown. The implication of Taylor dispersion can clearly be observed. Structures seem to diffuse in the direction of fluid flow.

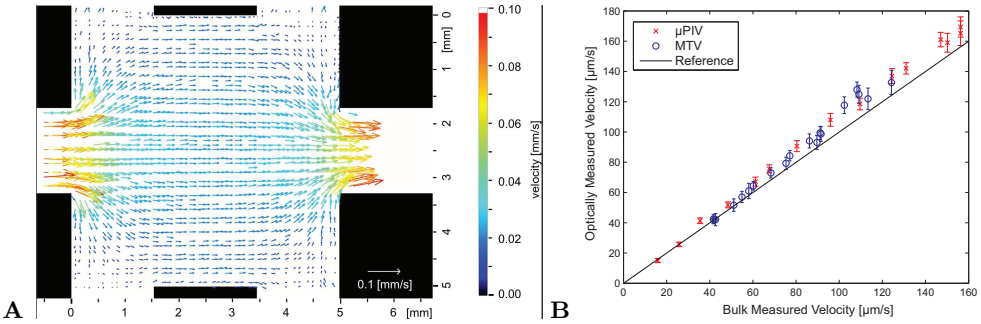

Fig. 7.15. The vector field computed for an inhomogeneous flow in the mixing chamber in **A**. Comparison of measured values from MTV (blue circles) and µPIV (red crosses) compared to ground truth measurement obtained from a flow meter (solid black line) in **B**.

as it is difficult to correct for this dispersion. Due to Taylor dispersion, the use of MTV is very limited. Nevertheless, it represents the only technique available for situations in which particles cannot be used for visualizing the flow, such as packed columns. Motion models such as those described in Sec. 7.2.4 were used to make feasible a highly accurate technique for measuring microfluidic flows based on MTV [29, 32, 94]. A comparison of MTV, µPIV and ground truth can be seen in Fig. 7.15.

In environmental sciences, it is the transport of the same quantities, energy, momentum and mass, that is the driving force in most weather conditions on short scales and in climatic variability on longer time periods. It plays the dominant role on aquatic processes in the oceans such as ocean currents and the thermohaline circulation, the meridional overturning circulation (MOC). The two compartments of atmosphere and ocean are in contact at the ocean surface where energy, mass and momentum is exchanged between them. Due to experimental difficulties of measuring inside te boundary layer, extending

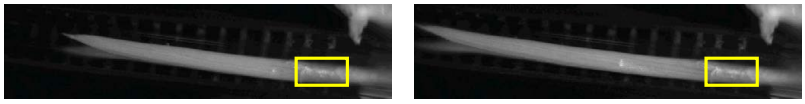

Fig. 7.16. Growing pine needle. 1^{st} (left) and 100^{th} (right) image of the sequence, the frames indicate the 256×128 sheath region.

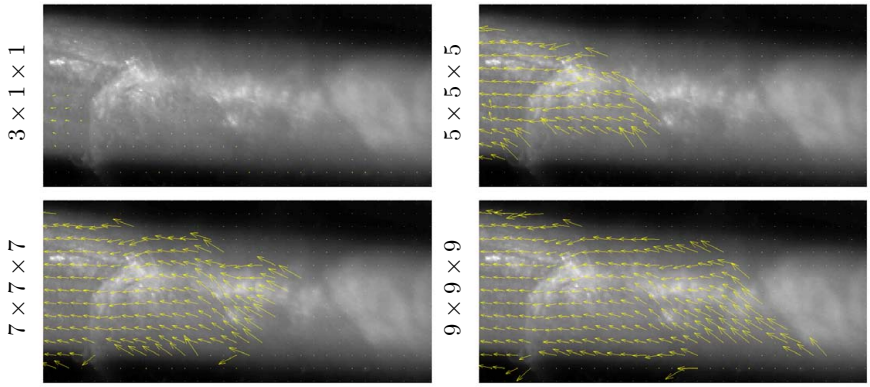

Fig. 7.17. Growing pine needle. From left to right and top to bottom: motion estimation results on the sheath region using $3 \times 1 \times 1$, $5 \times 5 \times 5$, $7 \times 7 \times 7$, and $9 \times 9 \times 9$-filters. Vectors longer than 4 pixels/frame are cut off. Vectors are scaled by a factor of 4.

less than one millimeter into the water body which is undulated by waves of several centimeters heights. Applying the advanced techniques for the estimation of complex motion presented in this chapter has lead to significant advances in the field of air-sea interactions. Estimating the total derivative of temperature T with respect to time $dT/dt = \alpha(\boldsymbol{u}) \circ T = c$ from thermographic image sequences of the air-water interface has made it possible to accurately estimate the net heat flux as well as the transfer velocity of heat [28, 34, 104]. The viscous shear stress could be deduced from active thermography [30]. This will make it possible to perform process studies, relating the transport of heat with that of momentum in the same footprint spatially and temporally highly resolved.

7.8.2 Plant growth

In Sec. 7.4.1 optimal filters for transparent motion estimation have been derived. As demonstrated in [98] using these filters reduce systematical errors in transparent motion estimation. Motion estimation of a growing pine needle (see Fig. 7.16) shows the effect of using different filter families. The sheath

Fig. 7.18. reconstruction of plant using 49 camera positions. From left to right: central image of input sequence, rendered depth estimates, close-up view on leaf and estimated depth with surface slopes

region consists of a transparent layer, becoming more and more opaque the closer to the base to the needle. Motion underneath this layer shall be measured in order to quantify local growth. In the current data sets only rigid motion is visible, indicating that the growth zone is completely hidden in the opaque region at the base of the needle. Fig. 7.17 depicts that using simple central distances 3x1x1-filters, reliable results can be achieved nowhere in the transparent region (same holds for 3x3x3-filters). The larger the optimized filters, the larger and visibly more accurate the motion fields become.

The 3D model proposed in Sec. 7.2.3 has been integrated in a screening setup established at ICG 3, Jülich. Depth and surfaces slopes of a plant (kalanchoe) reconstructed from an input sequence of 49 images, i.e. 7×7 camera positions, demonstrate the accuracy of the model. Fig. 7.18 shows the central image of the input sequence and of depth estimates rendered by *povray* [14]. Furthermore Fig. 7.18 shows a close-up view to compare a leaf with the estimated parameters in detail. Reconstructed depth and surfaces slopes match the original quite well.

7.8.3 Analysis of seismic signals

The analysis of seismic surface waves can provide valuable information about the subsurface structure, which is of interest for engineering purposes and seismic hazard assessment (see e.g. [46]).

A seismic signal typically contains several superposed propagation modes, each with its own dispersion and attenuation characteristics. A typical propagation model is

$$S^t(x,f) = \sum_{l=1}^{L} \mathrm{e}^{-x(\alpha_l(f)+\mathrm{i}k_l(f))} R_l(f), \tag{7.84}$$

where $S^t(x,f)$ is the spectrum of the signal recorded at distance x from the source, L is the number of modes, $R_l(f)$ is the spectrum of the source event for each mode, and $\alpha_l(f)$ and $k_l(f)$ are the frequency-dependent attenuation and

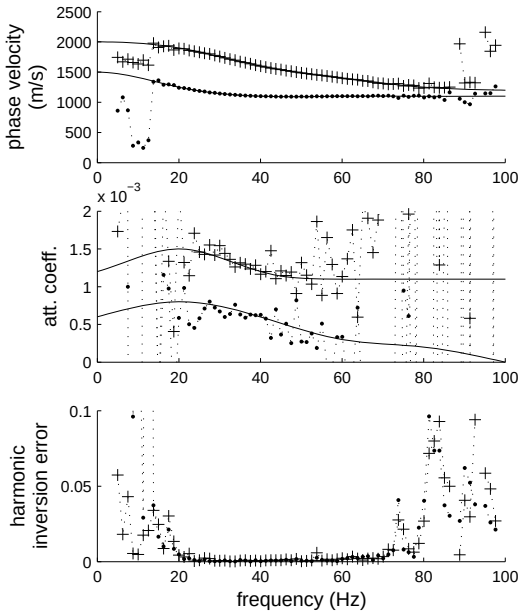

Fig. 7.19. Estimation results obtained on a signal containing two superposed propagation modes; noise level $\sigma = 0.001$, 20 recording stations.

wavenumber, respectively. (The wavenumber is related to the phase velocity by the expression $k_l(f) = \frac{2\pi f}{c_l(f)}$).

The problem is now to estimate $\alpha_l(f)$ and $k_l(f)$ from signals $s(x,t)$ recorded at stations $x \in \{x_1, \ldots, x_m\}$; we call this the "Multiple Modes" problem.

Note that the Multiple Motions problem is a specialization of the Multiple Modes problem where we have $k_l(f) = 2\pi \frac{f}{c_l}$ and $\alpha_l(f) \equiv 0$ for all $l = 1, \ldots, L$, i.e. all frequency components propagate with the same phase velocity c_l and there is no attenuation.

There are several existing approaches to solving the Multiple Modes problem, e.g. by estimating a wavelet operator [46]. We have developed a solution that is based on the technique of harmonic inversion by filter diagonalization [64, 65]. This technique successfully separates modes that overlap in both time and frequency, but it is quite sensitive to noise (Fig. 7.19 shows a sample result). However, our interest in the technique stems from the fact that, like the Multiple Motions technique, it is based on solving an eigenproblem. Based on this observation, we conjecture that the ideas used in the solution of the Multiple Motions problem can be generalized to yield a solution for the Multiple Modes problem; however, we have not been able to find such a solution so far.

7.9 Conclusions

In this chapter, the results of a fruitful and highly collaborative research initiative have been presented. A number of previously untackled and long-standing problems of estimating local and complex orientations were addressed in a multi-disciplinary effort. We have presented novel formulations for constraint equations, linking multi-dimensional signals to local orientations. These constraint equations make estimation of complex motions feasible. Also, a number of algorithms have been presented that make use of these orientation constraint equations and compute the model parameters in a statistically sound and efficient way. The novel algorithms that were presented in this chapter result from the combination of modern statistical signal processing, differential geometric analysis, novel estimation techniques, and nonlinear adaptive filtering and diffusion techniques. Moreover, these novel algorithms were applied to a number of applications, making digital image processing feasible to a number of them for the first time. The advanced algorithms were used in environmental-, earth-, bio-, and life sciences, leading to significant advances and contributions within these fields.

Acknowledgement. The authors gratefully acknowledge financial support from the German Science Foundation (DFG) within the priority program SPP1114.

References

[1] Aach T, Mota C, Stuke I, Mühlich M, Barth E (2006) Analysis of superimposed oriented patterns. IEEE Transactions on Image Processing 15(12):3690–3700

[2] Akaike H (1974) A new look at statistical model identification. IEEE Transactions on Automatic Control 19:716–723

[3] Andres B, Hamprecht FA, Garbe CS (2007) Selection of local optical flow models by means of residual analysis. In: Pattern Recognition, Springer, no. 4713 in LNCS

[4] Barron JL, Fleet DJ, Beauchemin SS (1994) Performance of optical flow techniques. Int Journal of Computer Vision 12:43–77

[5] Barth E, Caelli T, Zetzsche C (1993) Image encoding, labeling, and reconstruction from differential geometry. CVGIP: Graphical Models and Image Processing 55(6):428–46, URL http://www.inb.uni-luebeck.de/ barth/papers/BaCaZe93.html

[6] Barth E, Stuke I, Mota C (2002) Analysis of motion and curvature in image sequences. In: Proceedings IEEE Southwest Symp. Image Analysis and Interpretation, IEEE Computer Press, Santa Fe, NM, pp 206–10

[7] Barth E, Stuke I, Aach T, Mota C (2003) Spatio-temporal motion estimation for transparency and occlusion. In: Proceedings IEEE Int. Conf. Image Processing, IEEE Signal Processing Soc., Barcelona, Spain, vol III, pp 69–72
[8] Bergen J, Anandan P, Hanna K, Hingorani R (1992) Hierarchical model-based motion estimation. In: Proc. European Conference on Computer Vision, Springer, Berlin, Lecture Notes in Computer Vision, pp 237–252
[9] Bierling M (1988) Displacement estimation by hierarchical blockmatching. Cambridge, Mass.
[10] Bierling M, Thoma R (1986) Motion compensating field interpolation using a hierarchically structured displacement estimator. Signal Processing 11
[11] Bigün, J, Granlund, G H (1987) Optimal orientation detection of linear symmetry. In: ICCV, London, UK, pp 433–438
[12] Black, M J, Anandan, P (1996) The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. Computer Vision and Image Understanding 63(1):75–104
[13] Brox, Thomas, Weickert, Joachim, Burgeth, Bernhard, Mrázek, Pavel (2006) Nonlinear structure tensors. Image and Vision Computing 24(1):41–55
[14] Cason C (2005) Persistence of vision ray tracer (POV-Ray), version 3.6, Windows. www.povray.com
[15] Cootes TF, Thacker N, Taylor CJ (2002) Automatic model selection by modelling the distribution of residuals. In: ECCV 2002, LNCS 2353, Springer, pp 621–635
[16] Darrel, T, Simoncelli, E (1993) "nulling" filters and the separation of transparent motions. In: Darrel, T, Simoncelli, E (eds) CVPR, New York City, NY, pp 738–739
[17] Diehl M, Küsters R, Scharr H (2002) Simultaneous estimation of local and global parameters in image sequences. In: Forth Workshop Dynamic Perception, Bochum, Germany
[18] Elad M, Teo P, Hel-Or Y (1999) Optimal filters for gradient-based motion estimation. In: Proc. Intern. Conf. on Computer Vision (ICCV'99)
[19] Farid H, Simoncelli EP (1997) Optimally rotation-equivariant directional derivative kernels. In: 7th Intern. Conf. on Computer Analysis of Images and Patterns, Kiel, Germany
[20] Farnebäck G (1999) Spatial Domain Methods for Orientation and Velocity Estimation. Lic. Thesis, Linköping University, Linköping, Sweden
[21] Felsberg M, Jonsson E (2005) Energy tensors: Quadratic, phase invariant image operators. In: Kropatsch, W, Sablatnig, R, Hanbury, A (eds) Pattern Recognition, Springer, Berlin, Heidelberg, pp 493–500
[22] Felsberg M, Sommer G (2000) The multidimensional isotropic generalization of quadrature filters in geometric algebra. In: Algebraic Frames for the Perception-Action Cycle, Kiel

[23] Fleet D, Weiss Y (2005) Optical flow estimation. In: Paragios N, Chen Y, Faugeras O (eds) Mathematical Models in Computer Vision: The Handbook, Springer

[24] Fleet DJ, Black MJ, Yacoob Y, Jepson AD (2000) Design and use of linear models for image motion analysis. Int J Computer Vision 36:171–193

[25] Florac L, Niessen W, Nielsen M (1998) The intrinsic structure of optical flow incorporating measurement duality. Int J Computer Vision 27:263–286

[26] Freeman WT, Adelson EH (1991) The design and use of steerable filters. IEEE Transactions on Pattern Analysis and Machine Intelligence 13(9)

[27] Garbe CS (2007) Fluid flow estimation through integration of physical flow configurations. In: Hamprecht F, Schnörr C, Jähne B (eds) Pattern Recognition, Springer, Heidelberg, accepted

[28] Garbe CS, Schimpf U, Jähne B (2004) A surface renewal model to analyze infrared image sequences of the ocean surface for the study of air-sea heat and gas exchange. Journal of Geophysical Research 109(C08S15):1–18

[29] Garbe CS, Roetmann K, Jähne B (2006) An optical flow based technique for the non-invasive measurement of microfluidic flows. In: 12th International Symposium on Flow Visualization, Göttingen, Germany, pp 1–10

[30] Garbe CS, Degreif K, Jähne B (2007) Estimating the viscous shear stress at the water surface from active thermography. In: Garbe CS, Handler RA, Jähne B (eds) Transport at the Air Sea Interface - Measurements, Models and Parametrizations, Springer-Verlag, Berlin, Heidelberg, pp 223–239

[31] Garbe CS, Pieruschka R, Schurr U (2007) Thermographic measurements of xylem flow in plant leaves. New Phytologist Submitted

[32] Garbe CS, Roetmann K, Beushausen V, Jähne B (2007) An optical flow MTV based technique for measuring microfluidic flow in the presence of diffusion and Taylor dispersion. Experiments in Fluids Accepted

[33] Garbe, C S, Spies, Hagen, Jähne, B (2002) Mixed OLS-TLS for the estimation of dynamic processes with a linear source term. In: Van Gool L (ed) Pattern Recognition, Springer-Verlag, Zurich, CH, Lecture Notes in Computer Science, vol LNCS 2449, pp 463–471

[34] Garbe, C S, Spies, H, Jähne, B (2003) Estimation of surface flow and net heat flux from infrared image sequences. Journal of Mathematical Imaging and Vision 19(3):159–174

[35] Geman S, Geman D (1984) Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. PAMI 6(6):721–741

[36] Gheissari N, Bab-Hadiashar A, Suter D (2006) Parametric model-based motion segmentation using surface selection criterion. Computer Vision and Image Understanding 102:214–226

[37] Golub GH, van Loan CF (1996) Matrix Computations, 3rd edn. Johns Hopkins University Press
[38] Golub, G H, van Loan, C F (1996) Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore and London
[39] Granlund, G H, Knutsson, H (1995) Signal Processing for Computer Vision. Kluwer Academic, Dordrecht, The Netherlands
[40] Haußecker H (1999) Radiation. In: Jähne, B, Haußecker, H, Geißler, P (eds) Handbook of Computer Vision and Applications, vol 1, Academic Press, San Diego,CA, chap 2, pp 7–35
[41] Haußecker, H, Fleet, D J (2001) Computing optical flow with physical models of brightness variation. IEEE PAMI 23(6):661–673
[42] Haußecker, H, Spies, H (1999) Motion. In: Jähne, B, Haußecker, H, Geißler, P (eds) Handbook of Computer Vision and Applications, vol 2, Academic Press, chap 13
[43] Haußecker, H, Garbe, C, others (1999) A Total Least Squares for low-level analysis of dynamic scenes and processes. In: DAGM, Springer, Bonn, Germany, pp 240–249
[44] Haussecker H, Spies H (1999) Motion. In: Handbook of Computer Vision and Applications, Academic Press
[45] Heeger D (1988) Optical flow from spatiotemporal filters. IJCV 1
[46] Holschneider M, Diallo MS, Kulesh M, Ohrnberger M, Lück E, Scherbaum F (2005) Characterization of dispersive surface waves using continuous wavelet transforms. Geophysical Journal International 163:463–478
[47] Horn B, Schunck B (1981) Determining optical flow. Artificial Intelligence 17:185–204
[48] Horn, B K P, Schunk, B (1981) Determining optical flow. Artificial Intelligence 17:185–204
[49] Jähne B (1993) Spatio-temporal image processing. Lecture Notes in Computer Science, Springer Verlag
[50] Kay SM (1993) Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory. Prentice Hall PTR
[51] Knutsson H, Granlund G (1983) Texture analysis using two-dimensional quadrature filters. In: IEEE Workshop Computer Architecture for Pattern Analysis and Image Data Base Management., Pasadena (CA)
[52] Krajsek K, Mester R (2005) Signal and noise adapted filters for differential motion estimation. In: DAGM-Symposium, pp 476–484
[53] Krajsek K, Mester R (2005) Wiener-optimized discrete filters for differential motion estimation. In: Jähne B, Barth E, Mester R, Scharr H (eds) First International Workshop on Complex Motion, Günzburg, Germany, Oct. 2004, Springer Verlag, Berlin, Lecture Notes in Computer Science, vol 3417
[54] Krajsek K, Mester R (2006) The edge preserving wiener filter for scalar and tensor valued images. In: DAGM-Symposium, pp 91–100

[55] Krajsek K, Mester R (2006) Marginalized maximum a posteriori hyperparameter estimation for global optical flow techniques. In: Bayesian Inference and Maximum Entropy Methods In Science and Engineering, Paris, France
[56] Krajsek K, Mester R (2006) A maximum likelihood estimator for choosing the regularization parameters in global optical flow methods. In: IEEE International Conference on Image Processing, Atlanta, USA
[57] Krajsek K, Mester R (2006) On the equivalence of variational and statistical differential motion estimation. In: Southwest Symposium on Image Analysis and Interpretation, Denver, Colorado/U.S.A.
[58] Krajsek K, Mester R (2006) A unified theory for steerable and quadrature filters. In: VISAPP (1), pp 48–55
[59] Krajsek K, Mester R (2007) Bayesian model selection for optical flow estimation. In: DAGM-Symposium, pp 142–151
[60] Krieger G, Zetzsche C, Barth E (1995) Nonlinear image operators for the detection of local intrinsic dimensionality. In: Proc. IEEE Workshop Nonlinear Signal and Image Processing, pp 182–185
[61] Lawson C, Hanson R (1974) Solving Least-Squares Problems. Prentice-Hall
[62] Lucas, B, Kanade, T (1981) An iterative image registration technique with an application to stereo vision. In: DARPA Image Understanding Workshop, pp 121–130
[63] MacKay DJC (1992) Bayesian interpolation. Neural Computation 4(3):415–447
[64] Mandelshtam VA (2003) On harmonic inversion of cross-correlation functions by the filter diagonalization method. Journal of Theoretical and Computational Chemistry 2(4):1–9
[65] Mandelshtam VA, Taylor HS (1997) Harmonic inversion of time signals and its applications. Journal of Chemical Physics 107(17):6756–6769
[66] Mester R (2000) Orientation estimation: conventional techniques and a new approach. In: Proc. European Signal Processing Conference (EUSIPCO2000), Tampere (FI)
[67] Mester R (2002) Some steps towards a unified motion estimation procedure. In: Proc. 45th IEEE MidWest Symposium on Circuits and Systems (MWSCAS)
[68] Mester R (2002) A system-theoretical view on local motion estimation. In: Proc. IEEE SouthWest Symposium on Image Analysis and Interpretation, IEEE Computer Society, Santa Fé (NM)
[69] Mester R (2003) A new view at differential and tensor-based motion estimation schemes. In: Michaelis B (ed) Pattern Recognition 2003, Springer Verlag, Magdeburg, Germany, Lecture Notes in Computer Science
[70] Mester R (2003) On the mathematical structure of direction and motion estimation. In: Workshop on Physics in Signal and Image Processing, Grenoble, France

[71] Mühlich M (2005) Estimation in projective spaces and application in computer vision. PhD thesis, Johann Wolfgang Goethe Universität Frankfurt am Main

[72] Mühlich, M, Mester, R (1998) The role of total least squares in motion analysis. In: ECCV, Freiburg, Germany, pp 305–321

[73] Mühlich, M, Mester, R (1999) Subspace methods and equilibration in computer vision. Tech. Rep. XP-TR-C-21, Institute for Applied Physics, Goethe-Universitaet, Frankfurt, Germany

[74] Mühlich, Matthias, Mester, Rudolf (1998) The role of total least squares in motion analysis. In: Proceedings European Conference on Computer Vision ECCV 1998, Lecture Notes on Computer Science, pp 305–321

[75] Mitra, SK, Li, H, Lin, I-S, Yu, T-H (1991) A new class of nonlinear filters for image enhancement. In: International Conference on Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991, IEEE, Toronto, Ontario, Canada, vol 4, pp 2525–2528

[76] Mota C, Barth E (2000) On the uniqueness of curvature features. In: Baratoff G, Neumann H (eds) Dynamische Perzeption, Infix Verlag, Köln, Proceedings in Artificial Intelligence, vol 9, pp 175–178, URL http://www.inb.uni-luebeck.de/ barth/papers/ulm2000.html

[77] Mota C, Stuke I, Barth E (2001) Analytic solutions for multiple motions. In: Proceedings IEEE Int. Conf. Image Processing, IEEE Signal Processing Soc., Thessaloniki, Greece, vol II, pp 917–20

[78] Mota C, Aach T, Stuke I, Barth E (2004) Estimation of multiple orientations in multi-dimensional signals. In: IEEE Int. Conf. Image Processing, Singapore, pp 2665–8

[79] Mota C, Door M, Stuke I, Barth E (2004) Categorization of transparent-motion patterns using the projective plane. Tech. rep., Preprint series of the DFG priority program 1114 "Mathematical methods for time series analysis and digital image processing", URL http://www.math.uni-bremen.de/zetem/DFG-Schwerpunkt/preprints/prep062.pdf

[80] Mota C, Dorr M, Stuke I, Barth E (2004) Analysis and synthesis of motion patterns using the projective plane. In: Human Vision and Electronic Imaging Conference. 16th Symposium on Electronic Imaging Science and Technology, San Jose, CA, to appear

[81] Mota C, Dorr M, Stuke I, Barth E (2004) Categorization of transparent-motion patterns using the projective plane. International Journal of Computer & Information Science 5(2):129–140

[82] Mota C, Stuke I, Aach T, Barth E (2004) Spatial and spectral analysis of occluded motions. Tech. rep., Preprint series of the DFG priority program 1114 "Mathematical methods for time series analysis and digital image processing", URL http://www.math.uni-bremen.de/zetem/DFG-Schwerpunkt/preprints/prep064.pdf

[83] Mota C, Stuke I, Aach T, Barth E (2005) Divide-and-conquer strategies for estimating multiple transparent motions. In: Jähne B, Barth E, Mester R, Scharr H (eds) Complex Motion, 1. Int. Workshop, Günzburg,

Oct. 2004, Springer Verlag, Berlin, Lecture Notes in Computer Science, vol 3417

[84] Mota C, Stuke I, Aach T, Barth E (2005) Spatial and spectral analysis of occluded motions. Signal Processing: Image Communication Elsevier Science 20-6:529–536

[85] Mota C, Stuke I, Barth E (2006) The Intrinsic Dimension of Multispectral Images. In: MICCAI Workshop on Biophotonics Imaging for Diagnostics and Treatment, pp 93–100

[86] Mühlich M, Aach T (2006) A theory for multiple orientation estimation. In: Bischof H, Leonardis A (eds) Proceedings European Conference on Computer Vision 2006, Springer, no. 3952 in LNCS, pp (II) 69–82

[87] Mühlich M, Mester R (2004) A statistical unification of image interpolation, error concealment, and source-adapted filter design. In: Proc. Sixth IEEE Southwest Symposium on Image Analysis and Interpretation, Lake Tahoe, NV/U.S.A.

[88] Mumford D, Shah J (1989) Optimal approximations by piecewise smooth functions and associated variational problems. Comm Pure Appl Math XLII pp 577–685

[89] Nagel, H H, Enkelmann, W (1986) An investigation of smoothness constraints for the estimation of dispalcement vector fields from image sequences. PAMI 8(5):565–593

[90] Nestares O, Fleet D, Heeger D (2000) Likelihood functions and confidence bounds for total-least-squares problems. In: IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina, Vol. I, pp 523–530

[91] Nestares, O, Fleet, D J, Heeger, D (2000) Likelihood functions and confidence bounds for total-least-squares problems. In: CVPR'00, vol 1

[92] Ohta N (1996) Optical flow detection using a general noise model. IEICE Transactions on Information and Systems E79-D(7):951–957

[93] Polzehl J, Spokoiny V (2000) Adaptive weights smoothing with applications to image restoration. Journal of the Royal Statistical Soc B 62(2):335–354

[94] Roetmann K, Garbe CS, Schmunk W, Beushausen V (2006) Microflow analysis by molecular tagging velocimetry and planar ramanscattering. In: Proc. of 12th International Symposium on Flow Visualization, Göttingen, Germany

[95] Scharr H (2000) Optimal operators in digital image processing. PhD thesis, Interdisciplinary Center for Scientific Computing, Univ. of Heidelberg

[96] Scharr H (2005) Towards a multi-camera generalization of brightness constancy. In: Jähne B, Barth E, Mester R, Scharr H (eds) Complex Motion, 1. Int. Workshop, Günzburg, Oct. 2004, Springer Verlag, Berlin, Lecture Notes in Computer Science, vol 3417

[97] Scharr H (2006) Diffusion-like reconstruction schemes from linear data models. In: Pattern Recognition 2006, Springer Verlag, Berlin, Lecture Notes in Computer Science, submitted
[98] Scharr H (2007) Optimal second order derivative filter families for transparent motion estimation. In: EUSIPCO
[99] Scharr H, Schuchert T (2006) Simultaeous estimation of depth, motion and slopes using a camera grid. In: L Kobbelt TA T Kuhlen, Westermann R (eds) Vision Modeling and Visualization 2006, Aachen, pp 81–88
[100] Scharr H, Spies H (2005) Accurate optical flow in noisy image sequences using flow adapted anisotropic diffusion. Signal Processing: Image Communication 20(6):537–553
[101] Scharr H, Black M, Haussecker H (2003) Image statistics and anisotropic diffusion. In: Int. Conf. on Computer Vision, ICCV 2003, Nice, France, pp 840–847
[102] Scharr H, Stuke I, Mota C, Barth E (2005) Estimation of transparent motions with physical models for additional brightness variation. In: EUSIPCO
[103] Scharr, H, Körkel, S, Jähne, B (1997) Numerische Isotropieoptimierung von FIR-Filtern mittels Querglättung. In: Wahl, F M, Paulus, E (eds) DAGM, pp 199–208
[104] Schimpf U, Garbe CS, Jähne B (2004) Investigation of transport processes across the sea surface microlayer by infrared imagery. Journal of Geophysical Research 109(C08S13)
[105] Schnörr C (1998) A study of a convex variational diffusion approach for image segmentation and feature extraction. J Math Im and Vis 8(3):271–292
[106] Schuchert T, Scharr H (2007) Simultaneous estimation of surface motion, depth and slopes under changing illumination. In: DAGM 2007
[107] Schwarz G (1978) Estimating the dimension of a model. Annals of Statistics 6(461):464
[108] Shizawa M, Mase K (1990) Simultaneous multiple optical flow estimation. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, Atlantic City, pp 274–8
[109] Shizawa, M, Mase, K (1990) Simultaneous multiple optical flow estimation. In: ICPR, Atlantic City, NJ, USA, pp 274–278
[110] Shizawa, M, Mase, K (1991) Principle of superposition: A common computational framework for analysis of multiple motion. In: Proc. IEEE Workshop on Visual Motion, Princeton,NJ, pp 164–172
[111] Simoncelli EP (1994) Design of multi-dimensional derivative filters. In: Intern. Conf. on Image Processing, Austin TX
[112] Spies H, Scharr H (2001) Accurate optical flow in noisy image sequences. In: ICCV 2001, Vancouver, Canada
[113] Spies, H, Beringer, O, others (1999) Analyzing particle movements at soil interfaces. In: Jähne, B, Haussecker, H, Geißler, P (eds) Handbook

on Computer Vision and Applications, vol 3, Academic Press, pp 699–718

[114] Stuke I, Aach T, Barth E, Mota C (2003) Estimation of multiple motions by block matching. In: Dosch W, Lee RY (eds) Proceedings ACIS 4th Int. Conf. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, Lübeck, Germany, pp 358–62

[115] Stuke I, Aach T, Mota C, Barth E (2003) Estimation of multiple motions: regularization and performance evaluation. In: Vasudev B, Hsing TR, Tescher AG, Ebrahimi T (eds) Image and Video Communications and Processing 2003, Proceedings of SPIE, vol 5022, pp 75–86

[116] Stuke I, Aach T, Mota C, Barth E (2003) Linear and regularized solutions for multiple motion. In: Proceedings IEEE Int. Conf. Acoustics, Speech and Signal Processing, IEEE Signal Processing Soc., Hong Kong, vol III, pp 157–60

[117] Stuke I, Aach T, Barth E, Mota C (2004) Estimation of multiple motions using block-matching and Markov random fields. In: Panchanathan S, Vasudev B (eds) Visual Communications and Image Processing 2004, IS&T/SPIE 16th Annual Symposium Electronic Imaging, San Jose, California

[118] Stuke I, Barth E, Mota C (2006) Estimation of Multiple Orientations and Multiple Motions in Multi-Dimensional Signals. In: IEEE XIX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'06), pp 341–348

[119] T Brox BB J Weickert, Mrázek P (2004) Nonlinear structure tensors. Revised version of technical report no. 113, Saarland University, Saarbrücken, Germany

[120] Tschumperle D (2002) PDE's based regularization of multivalued images and applications. PhD thesis, Université de Nice-Sophia

[121] Tschumperle D (2005) Fast anisotropic smoothing of multi-valued images using curvature preserving pde's. Tech. rep., Equipe Image/GREYC, UMR CNRS 6072, 6 Bd du Maréchal Juin, 14050 Caen Cedex, France

[122] Tschumperle, D, Deriche, R (2005) Vector-valued image regularization with pdes: a common framework for different applications. IEEE Transactions on PAMI 27(4):506–517

[123] Unser M (1999) Splines: A perfect fit for signal and image processing. IEEE Signal Processing Magazine 16(6):22–38

[124] Van Huffel, S, Vandewalle, J (1989) Analysis and properties of the generalized total least squares problem $\boldsymbol{Ax} \approx \boldsymbol{B}$ when some or all columns in A are subject to error. SIAM Journal on Matrix Analysis and Applications 10(3):294–315

[125] Van Huffel, S, Vandewalle, J (1991) The Total Least Squares Problem: Computational Aspects and Analysis. Society for Industrial and Applied Mathematics, Philadelphia

[126] Walkington NJ (1996) Algorithms for computing motion by mean curvature. SIAM J Numer Anal 33(6):2215–2238, DOI http://dx.doi.org/10.1137/S0036142994262068

[127] Wang, JYA, Adelson, EH (1994) Spatio-temporal segmentation of video data. In: Proceedings of the SPIE: Image and Video Processing II, vol. 2182, San Jose, February 1994

[128] Weber, J, Malik, J (1995) Robust computation of optical flow in a multi-scale differential framework. International Journal of Computer Vision 14(1):67–81

[129] Wechsler H, Duric Z, Li FY, Cherkassky V (2004) Motion estimation using statistical learning theory. PAMI 26(4):466–478

[130] Weickert J, Brox: T (2002) Diffusion and regularization of vector- and matrix-valued images. In: M. Z. Nashed, O. Scherzer (Eds.) : Inverse Problems, Image Analysis, and Medical Imaging. Contemporary Mathematics, AMS, Providence, pp 251–268

[131] Wigner E (1959) Group Theory and its Application to Quantum Mechanics of Atomic Spectra. Academic Press, New York

[132] Winkler G (2002) Image Analysis, Random Fields and Markov Chain Monte Carlo Methods. A Mathematical Introduction. Springer, Berlin

[133] Worring M, Smeulders AWM (1993) Digital curvature estimation. CVGIP: Image Underst 58(3):366–382, DOI http://dx.doi.org/10.1006/ciun.1993.1048

[134] Yu W, Daniilidis K, Beauchemin S, Sommer G (1999) Detection and characterization of multiple motion points. In: 18th IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, pp 171–177

[135] Yu W, Daniilidis K, Sommer G (2000) A new 3D orientation steerable filter. In: Proc. DAGM 2000, Springer

[136] Zetzsche C, Barth E (1990) Fundamental limits of linear filters in the visual processing of two-dimensional signals. Vision Research 30:1111–7

[137] Zetzsche C, Barth E, Wegmann B (1993) The importance of intrinsically two-dimensional image features in biological vision and picture coding. In: Watson AB (ed) Digital Images and Human Vision, MIT Press, pp 109–38, URL http://www.inb.uni-luebeck.de/ barth/papers/ZeBaWe93a.html

Index