

Anisotropic Geometric Diffusion in Image and Image-Sequence Processing

Der Fakultät für Naturwissenschaften
der Universität Duisburg-Essen
(Standort Duisburg)

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

eingereichte Dissertation von

Tobias Preußer

aus
Dernbach

Datum der Einreichung: 5. Mai 2003

Contents

1	Introduction	1
1.1	Scale-space methods in image processing	2
1.2	Background from differential geometry	5
1.3	Curve and surface evolutions	8
1.4	Image-sequence processing	10
2	Anisotropic geometric diffusion	11
2.1	Anisotropic geometric diffusion of static images	11
2.2	Extracting motion velocities — optical-flow	19
2.3	Apparent acceleration	31
2.4	Coupled spatio-temporal smoothing of image-sequences	31
2.5	Comparison with existing models	33
3	Regularized geometric quantities	39
3.1	Global regularizations	40
3.2	Local regularizations	41
3.3	Regularized Shape Operator	42
3.4	Regularized apparent velocity and acceleration	45
4	Existence of viscosity-solutions	49
4.1	Reviewing basic theory of viscosity-solutions	50
4.2	Anisotropic linear geometric diffusion	52
4.3	Toward anisotropic nonlinear geometric diffusion	62
4.4	Existence of a solution of the nonlinear problem	69
5	Discretization and implementation	71
5.1	Finite-Elements on uniform spatial grids	71
5.2	Tensor product finite-elements in sequence-time and space	73
5.3	Geometric quantities on discrete data	73
5.4	Optimal regularization-parameters	77
5.5	Discretization of the evolution of static images	78
5.6	Discretization of the evolution of image-sequences	83
6	Conclusions	93
	Bibliography	95

List of Figures

1.1	Images of equal shapes, but different gray values	2
1.2	Result from the improved Perona Malik model of Cattè et al.	4
1.3	Result from the anisotropic diffusion by Weickert	4
1.4	Characterization of edges by the eigenvalues of the shape operator	7
1.5	Result of the application of mean-curvature-motion	9
2.1	Evolution of a 2D square data-set under the new model	16
2.2	Evolution of a 3D octahedron data-set under the new model	17
2.3	Evolution of an ultrasound heart data-set under the new model	17
2.4	Evolution of a real data-set under the new model	18
2.5	Comparison of different image processing models applied to a noisy 2D data-set	18
2.6	Ill posedness of the optical-flow problem	22
2.7	Velocities of a sample 2D data-set	24
2.8	Velocities of the taxi sequence	25
2.9	Splitting into tangential and normal component of v_{app} for the ellipsoidal data-set	26
2.10	Velocities of the ellipsoidal data-set	26
2.11	Velocities of the saltpool data-set	27
2.12	Velocities of the ultrasound heart data-set	27
2.13	Splitting into tangential and normal component of v_{app} for the ultrasound heart data-set	28
2.14	Extraction of apparent trajectories from an image-sequence during the evolution	34
2.15	Evolution of a 2D sample image-sequence.	35
2.16	Evolution of a real 2D sequence under the coupled diffusion.	35
2.17	Comparison of existing image processing methodology on the noisy octahe- dron data-set	36
2.18	Comparison of existing image processing methodology on one frame of the noisy ultrasound heart data-set	36
2.19	Comparison of existing image-sequence processing methodology on a sample 2D image-sequence	37
3.1	Curvature evaluation via convolutions and projections on an ellipsoid data-set.	44
3.2	Curvature evaluation via convolutions and projections on an octahedron data- set.	45
3.3	Curvature evaluation via convolutions and projections on an ellipsoid data-set.	46
3.4	Curvature evaluation via convolutions and projections on an octahedron data- set.	47
3.5	Curvature evaluation on a segmented cortex data-set	47
3.6	Analysis of the consistency of the regularizations.	48
4.1	Illustration of the jets of a function u	51
4.2	The local L^2 -projection is a local filter	54

4.3	Construction of sub- and super-solution for the linear anisotropic diffusion . .	60
5.1	Schematic sketch of the discrete evaluations of the regularizations.	75
5.2	Influence of the choice of σ on the evolution	79
5.3	Approximation of the singular level-set equation	79
5.4	Splitting of the system matrix for the coupled spatio-temporal diffusion . . .	87
5.5	Schematic sketch of the block solver	91

Notation

$ \cdot _1$	l_1 norm in \mathbb{R}^d	$\mathcal{M}_c, \mathcal{M}_c(t)$	Level-set of value c in an image respectively an image at scale t
$ \cdot , \cdot _2$	Euclidean norm in \mathbb{R}^d	$\mathcal{M}_{c,s}$	Level-set of value c in frame s of an image-sequence
$ \cdot _\varepsilon$	Regularized Euclidean norm in \mathbb{R}^d	\mathcal{M}^σ	Regularized level-set
$ \cdot _\infty$	Maximum norm in \mathbb{R}^d	N	Normal on level-sets
$\ \cdot\ _X$	Norm on vector-space X	N^σ	Regularized normal
$(\cdot, \cdot)_O$	Scalar-product on $L^2(O)$	\mathcal{N}^l	Set of nodes of the triangulation on level l
$\langle \cdot, \cdot \rangle$	Dual pairing on $(H^1(\Omega))' \times H^1(\Omega)$	ϕ_i	Temporal basis function, $i = 0, \dots, M$
\otimes	Tensor product $a \otimes b = (a_i b_j)_{ij}$	$P(O)$	Power set of O
$B(R)$	Ball of radius R	ψ_α	Spatial basis function, $\alpha \in \{0, \dots, N\}^d$
$C_0^0(O)$	Space of compactly supported C^0 -functions on O	Q	Image-sequence domain, $I \times \Omega$
$C^{k,\alpha}(O)$	Hölder space on O , $\alpha \in [0, 1]$.	ρ	Temporal grid-width
D^-	Backward difference quotient	Σ	Extended regularized shape operator
D^+	Forward difference quotient	$S(d)$	Space of symmetric $d \times d$ matrices
D^\pm	Central difference quotient	S	Jacobian of the normal
E_i	Element on the finest level of the quadtree/octtree \mathcal{T}	$S_{T_x} \mathcal{M}$	Shape operator on level-sets of an image
E_i^l	Element on level l of the quadtree/octtree \mathcal{T}	S^σ	Jacobian of the regularized normal
h	Spatial grid-width	$S_{T_x}^\sigma \mathcal{M}^\sigma$	Regularized shape operator on level-sets of an image
$H^m(O), H^{m,2}(O)$	Sobolev space of order m and exponent 2 on O	s_i	Temporal node, $i = 0, \dots, M$
$(H^1)'(O)$	Dual space of $H^1(O)$	τ	Scale-step width
κ^i	Principal curvatures, eigenvalues of $S_{T_x} \mathcal{M}$	$\mathcal{T}, \mathcal{T}^h$	Quadtree/octtree triangulation of Ω or Q
$\kappa^{i,\sigma}$	Regularized principal curvatures, eigenvalues of $S_{T_x}^\sigma \mathcal{M}^\sigma$	$T_x \mathcal{M}$	Tangent space of level-sets, $\text{span}\{w^i\}$
$L^p(O)$	Lebesgue space of exponent $p \in [1, \infty]$		
LSC(O)	Space of lower semi-continuous functions on O		

$T_x \mathcal{M}^\sigma$	Regularized tangent space of level-sets, $\text{span}\{w^{i,\sigma}\}$
$u(x)$	Static image $u : \Omega \rightarrow \mathbb{R}$
$u(s, x)$	Image-sequence $u : I \times \Omega \rightarrow \mathbb{R}$
$u_h(t, x),$ $u_{\rho,h}(t, s, x)$	Discrete finite element function
$u_h^n(x),$ $u_{\rho,h}^n(s, x)$	Discrete finite element function at scale $t = n\tau$
$U_h, U_{\rho,h}$	Vector of nodal values of u_h respectively $u_{\rho,h}$
$U_h^n, U_{\rho,h}^n$	Vector of nodal values of u_h^n respectively $u_{\rho,h}^n$
U_i^n	Vector of nodal values of the frame $u_{\rho,h}^n(s_i, \cdot)$
$U_{i,\alpha}^n$	Nodal value $u_{\rho,h}^n(s_i, x_\alpha)$
$\text{USC}(\mathcal{O})$	Space of upper semi-continuous functions on \mathcal{O}
\mathcal{V}^h	Spatial finite element space
w^i	Principal directions of curvature, eigenvectors of $S_{T_x \mathcal{M}}$
$w^{i,\sigma}$	Regularized principal directions of curvature, eigenvectors of $S_{T_x \mathcal{M}^\sigma}^\sigma$
\mathcal{W}^h	Temporal finite element space
x_α	Spatial node, $\alpha \in \{0, \dots, N\}^d$
Ω	Image domain, $\subset \mathbb{R}^d$.

Chapter 1

Introduction

THE USE OF partial differential equations (PDEs) in image processing has become an interesting research area during the last decade. The idea is to deform a given image with a PDE and obtain a filtered version of the image as the solution of this differential equation. The design of such PDEs has led to *evolution models*, where a given image is treated as initial data for a parabolic initial and boundary value problem. The time parameter of the process represents the so called *scale*, which leads through the evolution process from noisy, detailed original image data to coarse, smooth and enhanced versions of the original image. Advantages of PDE approaches, such as high speed, accuracy, and stability together with a variety of results from numerical analysis, have made these methods a growing field of research. The theory of viscosity solutions provides a framework for the augmentation of PDE-terminology to images, which may neither be differentiable in the classical nor in the weak sense.

The application of such methods is of special interest in three-dimensional (3D) image processing. In medical research and diagnosis, various highly developed data acquisition techniques like computed tomography (CT), nuclear magnetic resonance tomography (MRI) or 3D-ultrasound deliver high resolution images and time series of measured data. Unfortunately, these images – especially those from ultrasound – carry high-frequency noise due to the electronic acquisition process. Also in low-dose CT applications, which become more popular since they lower the risk for the patient, the signal to noise ratio is very poor. This makes it even harder for physicians to give a precise diagnosis or to plan surgical treatment. In medical image-sequence analysis one is interested in the motion, growth and deformation of the underlying tissue. The observed time-scales range from parts of a second to minutes and years.

Generally speaking, image acquisition devices measure certain densities of the observed object and transform them into image intensities (which are usually interpreted as gray values). The observed density may be the tissue density or a photon emission density in medical imaging, but also tracer densities in physical experiments, for example porous medium flow experiments. The consideration of image-sequences taken from an observed object allows to analyze the underlying physical phenomena which move the densities within the object. In the case of porous medium flow this reveals information about the quality of numerical models simulating the flow.

In many applications of PDE methods to image processing the actual image intensity (as a measure for the observed density) is of minor importance, since it mostly depends on the modality of the image taking device (cf. Figure 1.1). The user is much more interested in the iso-surfaces (*level-sets*) of the image and their shapes: It is much more important to have a look at the shape of an organ, than its color in the image. Evolution models, which depend only on the shapes of the level-sets rather than the actual gray values are called *morphologic* or *geometric*.



Figure 1.1: Images of equal shapes which differ by a transformation of the gray-values.

In this thesis, we consider nonlinear anisotropic geometric evolution problems for the de-noising and enhancement of images and image-sequences. The nonlinear and anisotropic character of the models results in the preservation of edges and corners of level-sets while reducing their noise. Appropriate regularization techniques for the intrinsic geometric quantities of the level-sets lead to a rich class of shapes which are invariant under the diffusion. The processing of image-sequences additionally takes the direction of motion and the acceleration of the level-sets in time into account. As a consequence the diffusion is coupled in sequence-time and space and it preserves highly accelerated motions. The results of the application to multiple problems are shown in various forms throughout the dissertation. During the remainder of this introductory chapter we review the basic image processing methodology, together with some tools from differential geometry, which are needed for the subsequent expositions. In Chapter 2 we discuss a geometric evolution model for the processing of static images, which is followed by a model suitable for the de-noising of image-sequences. The necessary regularizations are the topic of Chapter 3. There, we discuss different non-morphological and morphological regularization variants, and compare them on different data-sets. The models are treated analytically in Chapter 4, where we show the existence of viscosity solutions. Discretizational and implementational aspects are discussed in Chapter 5. For the static image processing model a stability estimate is derived and an operator splitting scheme is constructed for the image-sequence evolution. Final conclusions are drawn in Chapter 6.

1.1 Scale-space methods in image processing

The early work on PDEs in image processing goes back to Gabor [34] in 1965, and continued with the independent contributions of Witkin [81] and Koenderink [50], who introduced the notion of *scale-space*, which describes the simultaneous representation of images at multiple scales. Since the application of an evolution process $E(t)$ to a given image data u_0 delivers a family of representations $\{E(t)[u_0]\}_{t \geq 0}$ on successively coarser scales, such a process is also referred to as *multi-scale method* or *scale-space method*. The simplest scale-space method, the heat equation

$$\partial_t u - \Delta u = 0 \quad \text{on } \Omega \subset \mathbb{R}^d,$$

with initial data u_0 and natural boundary conditions results from the minimization of the classical Dirichlet integral

$$u = \operatorname{argmin}_{v \in H_0^1(\Omega)} \mathcal{D}(v) := \operatorname{argmin}_{v \in H_0^1(\Omega)} \int |\nabla v|^2 dx.$$

by means of a gradient descent of \mathcal{D} in the standard L^2 -metric. If $u : \mathbb{R}^+ \times \Omega \rightarrow \mathbb{R}$ solves the above boundary and initial value problem, the corresponding evolution-operator is defined as

$$E(t)[u_0] = u(t)$$

generating the scale-space

$$\{E(t)[u_0] : t \geq 0\}.$$

Besides the natural *causality* assumption

$$E(t+s)[u_0] = E(t)[E(s)[u_0]] \quad \forall s, t \geq 0,$$

scale-spaces generated by various filters have been required to fulfill many other properties [3]. Many of these properties can be characterized as invariance properties such as invariance under gray-level transformations, translations, rotations or affine mappings. Indeed the heat equation is the only evolution, which generates a scale-space and fulfills the *linearity assumption* [6]

$$E(t)[au_0 + bv_0] = aE(t)[u_0] + bE(t)[v_0] \quad \forall t \geq 0, a, b, \in \mathbb{R}$$

as well as the *isometry invariance* [3] for all orthogonal matrices $R \in O(d)$:

$$E(t)[u_0(Rx)] = (E(t)[u_0])(Rx) \quad \forall x \in \Omega \subset \mathbb{R}^d.$$

But the heat equation unfortunately leads to an undesired loss of edges of the image, which themselves are characterized by high gradients. A more successful model was presented in 1987 by Perona and Malik [64], improved by the work of Catté et al. in 1992 [15]. They defined a nonlinear diffusion model with a decreased diffusion coefficient in the vicinity of edges. Denoting the Euclidean norm with $|\cdot|$ the resulting evolution-problem can be written as

$$\partial_t u - \operatorname{div}(G_\lambda(|\nabla u^\sigma|)\nabla u) = 0 \quad \text{on } \Omega \subset \mathbb{R}^d,$$

again with initial data u_0 and natural boundary conditions. With increasing t the initial data $u_0 = u(t=0)$ is being smoothed, its structure is coarsened and the edges are enhanced, if the coefficient $G_\lambda(\cdot)$ reduces the diffusion for high gradients (cf. Figure 1.2). The gradient ∇u of the image itself does not enter the definition of the diffusivity $G_\lambda(|\nabla u^\sigma|)$, but the gradient of a regularized image u^σ . By choosing $G_\lambda(s) = (1 + s^2/\lambda^2)^{-1}$, the edges are characterized by the constant $\lambda > 0$: If $|\nabla u^\sigma| \geq \lambda$, the process results at least theoretically in a backward diffusion, whereas for $|\nabla u^\sigma| < \lambda$ forward diffusion smoothes the image [49]. The use of the regularized image's gradient ∇u^σ in the definition of the diffusivity makes the problem well-posed and furthermore stabilizes the process in presence of noise. Because otherwise on noisy image data, edges may falsely be detected due to the (theoretically unbounded) gradients of the noise.

Weickert [79, 80] presented a further improvement by taking an anisotropic diffusion character into account. In his model the Perona-Malik diffusion is only considered in directions of the gradient of the regularized image u^σ . Perpendicular to that direction (i.e. tangential to the level-sets of u^σ) a linear and constant diffusion is applied. This means that the resulting model also smoothes the edges tangentially (“along themselves”) (cf. Figure 1.3). If the diffusion coefficient depends on the eigenvectors of the *structure tensor* $\nabla u^\sigma \otimes \nabla u^\sigma$ line-structures in the image are especially pronounced. A similar approach was presented by Carmona and Zhong [14] who defined the smoothing directions as the eigenvectors of the Hessian of the image. Another extension considers the dependence of the diffusivity on the moments of the image density [52], because the moments hold information on the anisotropy of an image. In [65, 26] the anisotropic Weickert-diffusion was modified to smooth image



Figure 1.2: Application of the improved Perona-Malik model of Catté et al. [15] to the noisy image depicted on the left in Figure 1.1. From left to right successive scale steps of the evolution on this 513×513 image are depicted.



Figure 1.3: Application of the anisotropic diffusion by Weickert [80] to the noisy image depicted on the left in Figure 1.1. On the left and in the middle successive scale steps of the evolution on this 513×513 image are depicted. The edges of the image are smoothed in tangential direction as the magnified sections illustrate. The top row of the magnifications displays the results from the original Perona-Malik model shown in Figure 1.2. The corresponding magnified sections from the Weickert-diffusion are shown in the bottom row.

data in directions of arbitrary vector fields. The result shows streamline type patterns which visualize the vector field in a multi-scale fashion.

The numerical issues of PDEs in image processing have been considered in a wide variety: Within the computer science community they are mostly treated by finite difference (FD) schemes [2, 4, 82], often coupled with an explicit time stepping [80]. Kačur and Mikula [48] showed convergence of a semi-implicit finite element (FE) scheme for the improved Perona-Malik model [15]. Also for this model, adaptivity of triangular meshes was considered by Bänsch and Mikula [7]. An approach, which treats adaptive quadtree or octtree grids and the resulting matrices procedurally, was presented in [66]. A natural way of discretizing PDEs in image processing is to apply finite volume schemes [51], because they consider images to be a discrete set of constant values on small volumes (the *pixels* (in $2D$) or the *voxels* (in $3D$)), rather than piecewise multi-linear functions. The existence of discrete solutions and their convergence toward the solution of the continuous problem was proven in this context by Mikula and Ramarosy [59] for the improved Perona-Malik model of Catté et al.

None of the evolution-operators presented so far is of geometric nature, which means that they are not invariant under monotone transformations of the gray-value. This fact is obvious for the Perona-Malik model and its improvements by Catté et al. and Weickert,

since these models involve the gradient of a regularized version of the image. Transforming the gray values of the initial image would clearly alter its gradients and thus in general the edge detection and the diffusion tensor of the evolution process. But also the heat-equation is in general a non-morphological filter as becomes clear from the equivalence

$$u(t) = G_\sigma * u_0 \quad \text{with} \quad \sigma = \sqrt{2t},$$

of the solution of the heat equation $u(t)$ and the convolution of the initial data u_0 with the fundamental solution of the heat-equation: the Gaussian kernel G_σ of variance $\sigma = \sqrt{2t}$ (cf. [80]). Only for linear transformations, which are not realistic in applications, the heat-equation behaves geometrically.

Since geometric evolutions are the scope of this work, we continue with a review of geometric evolution methods and draw analogies to the above presented Euclidean models. Curvature quantities describing the morphology of images are a basic ingredient of the models to be presented. Thus, we review the according terminology from differential geometry first.

1.2 Background from differential geometry

Let us start this section with precise definitions of notions we have already used in the last section. For a domain $\Omega \subset \mathbb{R}^d$ for $d = 2, 3$, we consider an image on Ω to be a mapping

$$u : \Omega \rightarrow \mathbb{R}, \quad \text{with} \quad \Omega \subset \mathbb{R}^d.$$

We call the image two dimensional (2D) if $d = 2$ and three dimensional (3D) if $d = 3$. From a given image u one can easily extract a certain level-set

$$\mathcal{M}_c = \mathcal{M}_c(u) := \{x \in \mathbb{R}^d : u(x) = c\} \quad \text{for} \quad c \in \mathbb{R},$$

a process, which is closely related to the so called *thresholding*, i.e. the extraction of *sub-level-sets* $\{x \in \mathbb{R}^d : u(x) \leq c\}$. For the sake of simplicity we skip the subscript c and the argument u whenever they are clear from the context. Thus, we denote the level-set under consideration simply with \mathcal{M} . For de-noising or enhancing purposes the application of curve evolution models on the extracted level-set \mathcal{M}_c is possible [70]. The final composition of all processed level-sets forms again an image.

Definition 1.1. *The morphology $M(u)$ of an image $u : \mathbb{R}^d \rightarrow \mathbb{R}$ is the set of all level-sets \mathcal{M}_c of this image u , i.e.*

$$M(u) := \{\mathcal{M}_c : c \in \mathbb{R}\} \quad \text{with} \quad \mathcal{M}_c := \{x \in \mathbb{R}^d : u(x) = c\}.$$

Two images u and v are of equal morphology if $M(u) = M(v)$, i.e. each level-set of u coincides with a level-set of v , but these level-sets need not necessarily have the same gray values (cf. Figure 1.1).

Image processing operations which depend only on the morphology of an image are called *morphologic*. They yield equal results when applied to images with equal morphologies. Roughly speaking these image processing operations commute with the thresholding. We obtain the same notion for evolution processes:

Definition 1.2. *An evolution process $E(t)[u_0]$, which assigns to an image $u_0 : \mathbb{R}^d \rightarrow \mathbb{R}$ the image u_0 processed at scale $t \geq 0$ is called morphologic (or morphologically invariant), if for any continuous nondecreasing change of contrast $h : \mathbb{R} \rightarrow \mathbb{R}$ we have*

$$h \circ E(t)[u_0] = E(t)[h \circ u_0].$$

If an evolution process is morphologic, we call the corresponding PDE geometric.

Geometric evolution processes are steered by the morphology of an image rather than its gray values. The evolution depends only on the intrinsic geometric quantities of the level-sets.

1.2.1 Implicit surfaces — Level-sets

Clearly the level-sets \mathcal{M}_c of an image u define curves (*level-curves*) for 2D images or surfaces (*level-surfaces*) for 3D images which are not given in a parameterized form, but in an implicit definition

$$u(x) = c \quad \text{for all } x \in \mathcal{M}_c.$$

Assuming that the image u is smooth, we know from the implicit function theorem, that the level-sets \mathcal{M}_c describe smooth curves or surfaces, respectively. In the following we are interested in a characterization of a level-set via intrinsic geometric quantities.

We begin with the definition of a normal to a level-set. If the gradient $\nabla u(x)$ of the image u does not vanish in $x \in \mathcal{M}$, we can define a normal $N(x)$ to the level-set via

$$N(x) := \frac{\nabla u(x)}{|\nabla u(x)|}, \quad (1.1)$$

where again $|\cdot|$ denotes the Euclidean norm. Based on the normal we can decompose \mathbb{R}^d into the *normal-space* $\text{span}\{N(x)\}$ and the *tangent space* $T_x\mathcal{M} \perp \text{span}\{N(x)\}$ being orthogonal in the Euclidean sense. Since our $d - 1$ dimensional level-sets are embedded into \mathbb{R}^d the Euclidean metric of \mathbb{R}^d is directly induced to the tangent space.

The characterization of the level-sets is given by the notion of curvature. We make use of the unique normal (and the orientability of the level-set) we have just defined and consider the variation of this normal. Denoting the tensor product with $v \otimes w := (v_i w_j)_{ij}$, we define

$$S(x) : \mathbb{R}^d \rightarrow \mathbb{R}^d \quad \text{with} \quad S(x) := \nabla N(x)$$

which is the projection of the second derivative D^2u onto the tangent space $T_x\mathcal{M}$. Because

$$\partial_i N_j = \partial_i \frac{\partial_j u}{|\nabla u|} = \frac{\partial_i \partial_j u |\nabla u| - \partial_j u \partial_i |\nabla u|}{|\nabla u|^2} = \frac{1}{|\nabla u|} \left(\partial_i \partial_j u - \frac{d_j u \sum_{l=1}^d (\partial_l u \partial_i \partial_l u)}{|\nabla u|^2} \right)$$

the shape operator can be written as

$$S = \frac{1}{|\nabla u|} \left(D^2u - \frac{\nabla u}{|\nabla u|} \otimes D^2u \frac{\nabla u}{|\nabla u|} \right) = \frac{1}{|\nabla u|} (\text{Id} - N \otimes N) D^2u, \quad (1.2)$$

where we have omitted the argument x for reasons of clearness. By construction we have

$$S(x) : \mathbb{R}^d \rightarrow T_x\mathcal{M}$$

and the preceding projection onto the tangent space delivers the *shape operator*

$$S_{T_x\mathcal{M}} := S(x) (\text{Id} - N(x) \otimes N(x)) \quad \text{with} \quad S_{T_x\mathcal{M}} : T_x\mathcal{M} \rightarrow T_x\mathcal{M} \quad (1.3)$$

on the tangent space $T_x\mathcal{M}$. The shape operator is a symmetric endomorphism on the tangent space [27] and thus it is characterized by two real eigenvalues $\{\kappa^1, \kappa^2\}$ and the corresponding eigenvectors $\{w^1, w^2\}$. In the basis of the eigenvectors the shape operator $S_{T_x\mathcal{M}}$ therefore has diagonal form, i.e. denoting the basis transformation from the standard Euclidean basis $\{e_1, \dots, e_{d-1}\}$ to the basis of eigenvectors $\{\kappa^1, \kappa^2\}$ with $(w^1, w^2)^T$ we have

$$S_{T_x\mathcal{M}} = (w^1, w^2) \begin{pmatrix} \kappa^1 & \\ & \kappa^2 \end{pmatrix} (w^1, w^2)^T$$

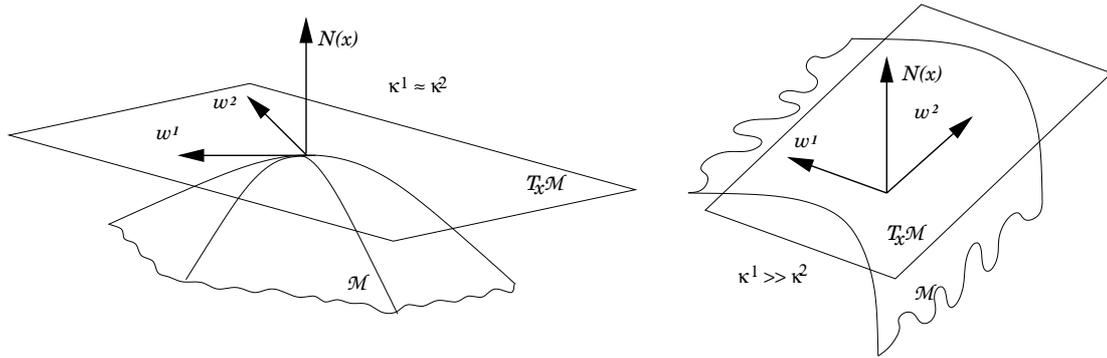


Figure 1.4: Sharp edges are indicated by the eigenvalues (=principal curvatures) and eigenvectors (=principal directions of curvature) of the shape operator $S_{T_x\mathcal{M}}$. Right: A sharp corner is indicated by two high eigenvalues $\kappa^1 \approx \kappa^2 \gg 1$. Left: Along a sharp edge of a level-set \mathcal{M} we have $\kappa^1 \gg \kappa^2$ and the corresponding eigenvectors point across the edge respectively tangentially along the edge.

and furthermore

$$S = (w^1, w^2, N) \begin{pmatrix} S_{T_x\mathcal{M}} & * \\ 0 & 0 \end{pmatrix} (w^1, w^2, N)^T.$$

The eigenvalues $\{\kappa^1, \kappa^2\}$ are called the *principal curvatures* and the eigenvectors are the *principal directions of curvature* of the level-set surface \mathcal{M} . If $\kappa^1 > \kappa^2$ then indeed w^1 and w^2 are the directions in which the normal N has maximum and minimum variation along the surface \mathcal{M} , respectively. We define the *mean curvature* $H = \kappa^1 + \kappa^2$ as the sum of the principal curvatures.

Areas in images containing important geometric information are given by the edges and the corners of the level-sets. In the non-geometric setting edges are characterized as regions of high spatial gradients. But from our geometric point of view, the spatial gradient is not a suitable quantity to describe image information, since we aim at independence from the gray values of the given image. Important information (which we refer to as *features* of the curve or surface, respectively the image) is characterized by curvatures of the level-set. Clearly from the viewpoint of image-processing only an approximation of edges and corners of level-surfaces is possible: Edges of level-surfaces are present in areas which have significantly different principal curvatures, e.g. $\kappa^1 \gg \kappa^2$. Corners of level-curves and level-surfaces are present if the curvatures are “very high”, i.e. $\kappa^i \gg 1$. An exact location of edges and corners of analytical level-sets would involve measure-valued shape operators. Therefore we approximate the features by an indication of high curvature. As seen above, a quantification of the curvature information is given by the principal curvatures as the eigenvalues of the shape operator $S_{T_x\mathcal{M}}$. Moreover the principal directions of curvature, which are the eigenvectors of $S_{T_x\mathcal{M}}$, deliver the orientation of edges on surfaces. In the vicinity of an edge we have one dominant eigenvalue, e.g. κ^1 such that $\kappa^1 \gg \kappa^2$. Then the dominant eigenvector w^1 lies orthogonal to the edge, whereas the remaining eigenvector w^2 points tangentially to the edge (cf. Figure 1.4).

1.2.2 Transformations of the gray-value

We focus, as already pointed out, on morphologically invariant models. By definition 1.2, morphological invariance means invariance under continuous nondecreasing changes of the gray value of the image. We sharpen this definition slightly by ruling out transformations of the gray value which are not twice differentiable and those, which have a vanishing first derivative. A vanishing first derivative would result in a transformation which clusters level-

sets – an undesirable property for real applications. We state our notion precisely in the following definition:

Definition 1.3. *An evolution operator $E(t)[u_0]$ is called geometric if for any monotone increasing transformation of the gray value $h \in C^2(\mathbb{R})$ with $h' > 0$ the relation*

$$h \circ E(t)[u_0] = E(t)[h \circ u_0]$$

holds.

Indeed the intrinsic shape operator, which describes the corners and edges of level-sets, is a good candidate for the modeling of geometric evolution processes, since it is invariant under transformations of the gray value and thus a geometric quantity, as the following lemma ensures:

Lemma 1.4. *The shape operator $S_{T_x\mathcal{M}}$ on the level-sets of an image u is invariant under monotone increasing transformations $h \in C^2(\mathbb{R})$ of the gray value of the image.*

Proof. Let us denote the shape operator resulting from the transformed image $\tilde{u} = h \circ u$ by $\widetilde{S_{T_x\mathcal{M}}}$. Therefore we have to show that

$$\widetilde{S_{T_x\mathcal{M}}} = S_{T_x\mathcal{M}}.$$

Starting with the definition of the shape operator (1.2), we have

$$\begin{aligned} \widetilde{S_{T_x\mathcal{M}}} &= \frac{1}{|\nabla\tilde{u}|} \left(\text{Id} - \frac{\nabla\tilde{u} \otimes \nabla\tilde{u}}{|\nabla\tilde{u}|^2} \right) D^2\tilde{u} \left(\text{Id} - \frac{\nabla\tilde{u} \otimes \nabla\tilde{u}}{|\nabla\tilde{u}|^2} \right) \\ &= \frac{1}{|h'|\nabla u|} \left(\text{Id} - \frac{h'\nabla u \otimes h'\nabla u}{|h'|^2|\nabla u|^2} \right) (h''\nabla u \otimes \nabla u + h'D^2u) \left(\text{Id} - \frac{h'\nabla u \otimes h'\nabla u}{|h'|^2|\nabla u|^2} \right) \end{aligned}$$

and since h is monotone increasing ($h' > 0$)

$$= \frac{1}{h'|\nabla u|} \left(\text{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) (h''\nabla u \otimes \nabla u + h'D^2u) \left(\text{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right).$$

Now, the term $h''\nabla u \otimes \nabla u$ vanishes, since ∇u lies in the kernel of the succeeding projection $(\text{Id} - N \otimes N)$. Thus, we have shown that

$$\widetilde{S_{T_x\mathcal{M}}} = \frac{1}{|\nabla u|} \left(\text{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) D^2u \left(\text{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) = S_{T_x\mathcal{M}}.$$

□

1.3 Curve and surface evolutions

Closely related to the evolution of images under geometric PDEs is the movement of curves and surfaces with curvature-based velocities. Within this area the level-set method of Osher and Sethian [61] has been an important contribution. Their idea was to represent a given and deforming curve or surface as a level-set of a higher dimensional hypersurface. This technique gives a solution to topological issues which had been difficult to handle and moreover it can be treated more accurately in numerical implementations. The concept that relates the level-set approach to PDEs in image processing is the consideration of an image as the collection of all its level-sets (i.e. its morphology).



Figure 1.5: Application of mean curvature motion to a the noisy image depicted on the left in Figure 1.1. From left to right successive scale steps of the evolution on the 513×513 image are shown. One clearly recognizes how highly curved shapes (as e.g. the shape of the trees in the foreground) are smoothed rapidly during the evolution.

The rigorous axiomatic work of Alvarez, Guichard, Lions and Morel [3] delivered an axiomatic approach to image processing via PDEs. Their approach allows to classify evolution processes from the set of fundamental axioms *Causality*, *Regularity*, *Morphological Invariance*, and *Euclidean Invariance* or *Affine Invariance*, respectively. Due to the inclusion of the axiom of morphological invariance, they end up with a class of equations described by

$$\partial_t u - |\nabla u| F \left[\operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right), t \right] = 0. \quad (1.4)$$

Within this class the evolutions $F = \text{const.}$ result in the constant Euclidean motions (*dilation* and *erosion*). The evolution of all level-sets in normal direction according to their mean curvature H is described by

$$\partial_t u - |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = 0.$$

Similar to its Euclidean analogue (i.e. the heat equation) this *mean curvature motion (MCM)* decreases the “geometric” noise [56] but suffers from a loss of surface features in form of corners and edges of the level-sets (cf. Figure 1.5). The incorporation of geometric information into image processing methods has been presented by Pauwels et al. [63], and Sapiro [69] presented an approach which takes into account the gradient of the image via the evolution

$$\partial_t u - |\nabla u| \operatorname{div} \left(g(|\nabla u|^\sigma) \frac{\nabla u}{|\nabla u|} \right) = 0.$$

However none of the above described isotropic methods is capable of retaining the important geometric content while decreasing the geometric noise.

In terms of evolution of curves and surfaces anisotropic models depending on the eigenvalues and the eigenvectors of the shape operator $S_{T_x \mathcal{M}}$ allow the preservation of the surface- or curve-features. The work of Clarenz et al. [17] considers an anisotropic and nonlinear evolution of surfaces which decreases the motion speed in the vicinity of edges and corners. The evolution is modeled in analogy to the anisotropic diffusion of Weickert [80] in the Euclidean case and it therefore uses an anisotropic weighting in the coordinates of the principal directions of curvature to evaluate the evolution speed. The model being described in this work for the processing of static images is closely related to the model by Clarenz et al. [17]. It is the analogue to their model in the level-set context, which incorporates all the advantages

which level-set methods have in contrast to explicit models: Topological changes and tangential movements do not impose conflicts. All level-sets are treated simultaneously, which makes the preceding identification and extraction of particularly interesting surfaces from a given data-set via a marching cube algorithm [54] obsolete. The important difference to existing image processing methodology is, that the anisotropic geometric diffusion model is capable of retaining structures of codimension 2 (i.e. corners/edges), whereas present models are characterized by the preservation of codim 1 features (i.e. lines/planes).

For the parametric setting the convergence of a semi-implicit finite element scheme for MCM was proven by Dziuk [29] already in 1991. The existence of viscosity solutions for the level-set form of MCM was shown independently by Chen et al. [16] and Evans and Spruck [32]. Deckelnick and Dziuk proved convergence of a finite element scheme for the level-set form of MCM in [22, 23]. A more detailed review on the related work in the area of viscosity solutions is given in Chapter 4, where the existence of viscosity solutions is shown for the nonlinear anisotropic diffusion models.

1.4 Image-sequence processing

For the processing of image-sequences it is desired to take into account the information contained in the additional dimension corresponding to the time of the sequence. One tries to analyze the motion of the gray values within the sequence-time. This problem, also known as the *optical-flow problem*, has been studied extensively in the past for sequences of 2D images [37, 5, 24, 31].

In the axiomatic work of Guichard [3] a classification of PDEs has been given for the processing of sequences of 2D images. From a set of fundamental axioms his work derives the model

$$\partial_t u - |\nabla u| F \left[\operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right), t, \operatorname{accel}(u) \right] = 0,$$

which assumes the image-sequence is in *Lambertian motion*, i.e. the gray values do not change along motion paths. Compared to the static image processing models (1.4) this evolution depends additionally on the curvature of the trajectories $\operatorname{accel}(u)$ which corresponds to an acceleration quantity (cf. Section 2.2.2).

Mikula et al. [71] presented an extension of the improved Perona-Malik model [15] toward the processing of sequences of 3D images, by taking into account a modulation of the diffusion speed in terms of the acceleration $\operatorname{accel}(u^\sigma)$ of a regularized image u^σ here written as the curvature of the Lambertian trajectories clt :

$$\partial_t u - \operatorname{clt}(u^\sigma) \operatorname{div} (G(|\nabla u^\sigma|) \nabla u) = 0.$$

Their model increases the diffusivity if the acceleration is high, whereas for uniform motions ($\operatorname{clt}(u^\sigma) = 0$) the equation is reduced to an algebraic identity. This may result in a loss of motion information from the sequence, since highly accelerated trajectories are smoothed significantly. The model behaves as an isotropic filter in the sequence time direction, since it does not take the direction of motion into account as a smoothing direction.

The evolution problem for image-sequence processing considered in this work (cf. Section 2.2) treats all image dimensions in an anisotropic way. This results in a model that behaves anisotropic also in direction of the sequence-time, and therefore is much more capable of preserving accelerated motions.

Chapter 2

Anisotropic geometric diffusion in image and image-sequence processing

THE FOLLOWING CHAPTER first presents the anisotropic geometric diffusion of the level-sets of static images and then deals with image-sequences. When speaking of *static* images, we mean mappings $u : \Omega \rightarrow \mathbb{R}$ from a spatial domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ into the space of image intensities \mathbb{R} . In contrast to that we speak of image *sequences* (also referred to as *movies*) $u : I \times \Omega \rightarrow \mathbb{R}$, $I \subset \mathbb{R}$, if the mapping goes from a spatio-temporal domain $I \times \Omega$ into the space of intensity values \mathbb{R} . For static images, we speak of *2D* or *3D* images, whereas we refer to image-sequences as $(2+1)D$ or $(3+1)D$ data. The section on static images at the beginning of this chapter is followed by a discussion of the optical-flow problem. Afterwards we go into the details of the image-sequence processing model. A comparison of the model with existing methodology of image and surface evolution closes the chapter.

As pointed out in the introduction, the presented methods are so called *level-set methods* which let all level-sets of an image u evolve simultaneously. This is an advantage to parametric methods, if the level-value of interest is not known a priori. Of course a parametric representation of a single level-set can be extracted from the function u e.g. by the marching-cube algorithm [54]. Vice versa from a parametric surface a level-set representation can be obtained by means of a signed distance function [40]. In the following we focus on 3D images, while having in mind, that a reduction to 2D images is straight forward (cf. Remark 2.3).

2.1 Anisotropic geometric diffusion of static images

Let the domain $\Omega \subset \mathbb{R}^d$ for $d = 2, 3$ be bounded, with a Lipschitz boundary $\partial\Omega$ and let an initial image be given in terms of the gray-value function $u_0 : \Omega \rightarrow \mathbb{R}$. The level-sets of u_0 are given by

$$\mathcal{M}_c(0) := \{x \in \Omega : u_0(x) = c\}.$$

We assume that the given image u_0 is noisy, e.g. due to some measurement errors (cf. Chapter 1), and we seek a multi-scale of smoother images $\{u(t, \cdot)\}_{t \geq 0}$ with $u(t, \cdot) : \Omega \rightarrow \mathbb{R}$ and $u(0, \cdot) = u_0(\cdot)$. As common in multi-scale methods the time t denotes the scale-parameter and the family of filtered images delivers a family of filtered level-sets $\{\mathcal{M}_c(t)\}_{t \geq 0}$ via

$$\mathcal{M}_c(t) := \{x \in \Omega : u(t, x) = c\}.$$

As defined above $\mathcal{M}_c(0)$ is the initial level-set described by u_0 . For the time being let us assume that the images $u(t, \cdot)$ are smooth and the gradient does vanish (i.e. $\nabla u(t, x) \neq 0$ for

all $(t, x) \in \mathbb{R}_0^+ \times \Omega$). Thus, we can apply the terminology from Section 1.2 to the level-sets of this image u . We relax the assumption of a non-vanishing gradient when considering suitable regularizations later in Chapter 3.

We are looking for a morphologic smoothing process. Therefore the first choice would be a mean-curvature-motion filter (cf. Section 1.3). But we moreover want the model to retain (or even enhance) corners and edges of an image's level-sets. As already mentioned in the introduction, the features of surfaces are described by the principal directions of curvature κ^i and the principal directions of curvature w^i as eigenvalues and eigenvectors of the shape operator $S_{T_x\mathcal{M}}$.

The idea for the envisaged evolution is to model the diffusion-tensor such that it depends on the principal curvatures κ^i and the principal directions of curvature $w^i, i = 1, 2$. In contrast to the mean curvature motion we would like to take an anisotropic weighting of the principal directions of curvature into account. For MCM the speed of the diffusion depends only on the mean curvature $H = \text{tr}(S_{T_x\mathcal{M}})$, but in our anisotropic model the evolution-speed shall be reduced significantly in presence of an edge or corner feature. We model this by introducing a weighting of the eigenvalues κ^i of the shape operator in coordinates of the principal directions of curvature w^i . This means, that in the vicinity of an edge (i.e. $\kappa^1 \gg \kappa^2$), we prescribe a small diffusion coefficient in direction of the dominant curvature vector w^1 , which points across the edge, and a larger fixed diffusion coefficient in direction of w^2 , i.e. in the subdominant direction of curvature, which lies tangentially to the edge (cf. Figure 1.4). The mechanism steering the weighting of the curvatures is obtained by the function

$$G_\lambda(s) := \frac{1}{1 + s^2/\lambda^2} \quad \text{for } \lambda \in \mathbb{R}^+, \quad (2.1)$$

applied to the $\kappa^i, i = 1, 2$. If the situation is locally spherical, i.e. $\kappa^1 = \kappa^2$, the weights delivered by G_λ coincide: $G_\lambda(\kappa^1) = G_\lambda(\kappa^2)$ and the weighting should become isotropic as in MCM.

Remark 2.1. *The function G which steers the evaluation of the curvatures is well known in image processing. In the prominent Perona-Malik diffusion [64, 15], it plays the role of an edge indicator together with the gradient of the image. We emphasize that for their model the parameter $\lambda \in \mathbb{R}^+$ is exactly the switch between forward and backward diffusion [49].*

Obviously, the evaluation of the shape-operator, which is calculated from the second derivatives of the image u , is an unstable process on noisy images. For non-morphologic methods (e.g. the Perona-Malik model) noise leads to high gradients and therefore the possible detection of false edges. Here the detection of corners and edges of the level can also be erroneous with respect to the true corners and edges of the initial image u_0 . As a consequence we consider regularizations of the shape operator $S_{T_x\mathcal{M}}$, before evaluating its eigenvalues and eigenvectors. The anisotropic weighting of the curvatures will be done in terms of these regularized shape operators. In the following, regularized quantities are denoted with a superscript σ , and σ is the parameter controlling the regularization process.

Let us now suppose, we have regularized the shape operator $S_{T_x\mathcal{M}}$ leading to a symmetric endomorphism

$$S_{T_x\mathcal{M}^\sigma}^\sigma : T_x\mathcal{M}^\sigma \rightarrow T_x\mathcal{M}^\sigma \quad \text{with} \quad T_x\mathcal{M}^\sigma = \text{span}\{w^{1,\sigma}, w^{2,\sigma}\}$$

on a *regularized tangent space* $T_x\mathcal{M}^\sigma$. By construction $S_{T_x\mathcal{M}^\sigma}^\sigma$ diagonalizes with respect to the basis $\{w^{1,\sigma}, w^{2,\sigma}\}$ and we extend the new operator to entire \mathbb{R}^3 with the mapping

$$\Sigma : \mathbb{R}^3 \rightarrow \mathbb{R}^3, \quad \Sigma := \begin{pmatrix} S_{T_x\mathcal{M}^\sigma}^\sigma & 0 \\ 0 & 0 \end{pmatrix} \quad (2.2)$$

in coordinates $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$ of the regularized principal curvatures $w^{1,\sigma}, w^{2,\sigma}$ and the regularized normal $N^\sigma \perp w^{1,\sigma}, w^{2,\sigma}$. We call Σ the *extended regularized shape operator*. Clearly Σ diagonalizes with respect to the basis $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$ and its eigenvectors are $\{\kappa^{1,\sigma}, \kappa^{2,\sigma}, 0\}$.

Now we are ready to state the anisotropic geometric diffusion problem (cf. [67]):

For given $u_0 : \Omega \rightarrow \mathbb{R}$ find a function $u : \mathbb{R}^+ \times \Omega \rightarrow \mathbb{R}$ which is a viscosity solution of the following evolution problem

$$\begin{aligned} \partial_t u - |\nabla u| \operatorname{div} \left(a(\Sigma) \frac{\nabla u}{|\nabla u|} \right) &= 0 && \text{in } \mathbb{R}^+ \times \Omega, \\ a(\Sigma) \frac{\partial u}{\partial \nu} &= 0 && \text{on } \mathbb{R}^+ \times \partial\Omega, \\ u(0, \cdot) &= u_0(\cdot) && \text{in } \Omega, \end{aligned} \quad (2.3)$$

where ν denotes the outer normal to Ω . The diffusion tensor is defined via

$$a(\Sigma) = (w^{1,\sigma}, w^{2,\sigma}, N^\sigma) \begin{pmatrix} G_\lambda(\kappa^{1,\sigma}) & & \\ & G_\lambda(\kappa^{2,\sigma}) & \\ & & 0 \end{pmatrix} (w^{1,\sigma}, w^{2,\sigma}, N^\sigma)^T,$$

where G_λ is the well known Perona-Malik function (2.1), and $(w^{1,\sigma}, w^{2,\sigma}, N^\sigma)^T$ is the basis transformation from the standard Euclidean basis to the orthonormal frame $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$.

In the coordinates $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$ the diffusion tensor $a(\Sigma)$ has the diagonal form $\operatorname{diag}(G_\lambda(\kappa^{1,\sigma}), G_\lambda(\kappa^{2,\sigma}), 0)$. Moreover, since $N^\sigma \in \ker a(\Sigma)$, the diffusion tensor

$$a(\Sigma) \Big|_{T_x \mathcal{M}^\sigma} : T_x \mathcal{M}^\sigma \rightarrow T_x \mathcal{M}^\sigma$$

is a positive definite endomorphism on the tangent space $T_x \mathcal{M}^\sigma$. Depending on the curvatures $\kappa^{1,\sigma}, \kappa^{2,\sigma}$ the principal directions $w^{1,\sigma}, w^{2,\sigma}$ are weighted by the function G_λ .

The problem defined in (2.3) is degenerate in normal direction as we will see below. Obviously it is crucial to the model that the diffusion tensor a depends on a regularized shape-operator Σ . Otherwise we have $\ker a(S_{T_x \mathcal{M}}) = \operatorname{span}\{N\}$ such that $\operatorname{div} \left(a(S_{T_x \mathcal{M}}) \nabla u / |\nabla u| \right) = 0$ — the evolution would run with zero velocity, hence resting at the initial image u_0 . The importance of the dependence of the diffusion tensor on regularized quantities has already been observed for the anisotropic Weickert-diffusion in [80]. Below we will see that for our model the difference between regularized and non-regularized quantities is steering the evolution.

The MCM evolution is already a problem which is degenerate in normal direction: Taking into account the relation (1.2) we can write the MCM in level-set form [61] as

$$\begin{aligned} 0 &= \partial_t u - |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \partial_t u - |\nabla u| \operatorname{tr}(DN) \\ &= \partial_t u - |\nabla u| \operatorname{tr} \left(\frac{1}{|\nabla u|} \left(\operatorname{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) D^2 u \right) \\ &= \partial_t u - \operatorname{tr} \left(\left(\operatorname{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) D^2 u \right), \end{aligned}$$

where $\operatorname{Id} - |\nabla u|^{-2} \nabla u \otimes \nabla u = \operatorname{Id} - N \otimes N$ is the projection onto the tangent space. Analogously, starting from (2.3) and assuming enough regularity for $a(\Sigma)$ we see that the

anisotropic level-set problem (2.3) can be written as

$$\begin{aligned} 0 &= \partial_t u - |\nabla u| \operatorname{div}(a(\Sigma)N) = \partial_t u - |\nabla u| \left((\operatorname{div} A(\Sigma)) \cdot N + \operatorname{tr}(a(\Sigma)\nabla N) \right) \\ &= \partial_t u - (\operatorname{div} a(\Sigma)) \cdot \nabla u - \operatorname{tr} \left(a(\Sigma) \left(\operatorname{Id} - \frac{\nabla u \otimes \nabla u}{|\nabla u|^2} \right) D^2 u \right), \end{aligned} \quad (2.4)$$

with

$$\operatorname{div} a(\Sigma) = \left(\sum_{j=0}^d \partial_j (a(\Sigma)_{ij}) \right)_i.$$

Thus, also the new model considers only the projection of the second derivative onto the tangent space — it is degenerate in normal direction. Moreover, in general we cannot assume that the regularized tangent space $T_x \mathcal{M}^\sigma$ coincides with the original tangent space $T_x \mathcal{M}$. Therefore we cannot guarantee that $a(\Sigma)$ still acts as a positive definite endomorphism on $T_x \mathcal{M}$. Indeed, we could have $N^\sigma \in T_x \mathcal{M}$, which could degenerate the problem even more. To avoid instabilities we add a small weight $\alpha \ll 1$ in normal direction and thus substitute

$$a(\Sigma) \rightsquigarrow a^\alpha(\Sigma) = (w^{1,\sigma}, w^{2,\sigma}, N^\sigma) \begin{pmatrix} G_\lambda(\kappa^{1,\sigma}) & & \\ & G_\lambda(\kappa^{2,\sigma}) & \\ & & \alpha \end{pmatrix} (w^{1,\sigma}, w^{2,\sigma}, N^\sigma)^T.$$

However, in our computations we have not experienced any instabilities with the original model. Consequently we stick to the original diffusion tensor.

Let us characterize the behavior of the anisotropic geometric diffusion with respect to the evolution of the level-sets in normal-direction:

Theorem 2.2. *The anisotropic geometric evolution (2.3) is equivalent to the propagation of the level-sets $\mathcal{M}_c(t)$ with normal velocity*

$$f = \operatorname{tr}(a(S^\sigma - S)) + (\operatorname{div} a(\Sigma))(N^\sigma - N), \quad (2.5)$$

where we have used the definition $S^\sigma = DN^\sigma$. Thus, for a parameterization $x(t)$ of the level-set $\mathcal{M}_c(t)$ the following holds:

$$\partial_t x = fN.$$

Proof. Let us examine the divergence term of the evolution problem. As above, we can perform the differentiation to obtain

$$\begin{aligned} \operatorname{div}(a(\Sigma)N) &= (\operatorname{div} a(\Sigma)) \cdot N + \operatorname{tr}(a(\Sigma)DN) \\ &= (\operatorname{div} a(\Sigma)) \cdot N^\sigma - (\operatorname{div} a(\Sigma)) \cdot N^\sigma + (\operatorname{div} a(\Sigma)) \cdot N + \operatorname{tr}(a(\Sigma)DN) \\ &= \operatorname{div}(a(\Sigma)N^\sigma) - \operatorname{tr}(a(\Sigma)DN^\sigma) + (\operatorname{div} a(\Sigma)) \cdot (N - N^\sigma) + \operatorname{tr}(a(\Sigma)DN), \end{aligned} \quad (2.6)$$

where we used the relation $(\operatorname{div} a(\Sigma)) \cdot N^\sigma = \operatorname{div}(a(\Sigma)N^\sigma) - \operatorname{tr}(a(\Sigma)DN^\sigma)$. And because $N^\sigma \in \ker a(\Sigma)$, we deduce

$$\begin{aligned} \operatorname{div}(a(\Sigma)N) &= 0 - \operatorname{tr}(a(\Sigma)DN^\sigma) + \operatorname{tr}(a(\Sigma)DN) + (\operatorname{div} a(\Sigma)) \cdot (N - N^\sigma) \\ &= -\operatorname{tr}(a(\Sigma)(DN - DN^\sigma)) + (\operatorname{div} a(\Sigma)) \cdot (N - N^\sigma). \end{aligned}$$

So, the anisotropic geometric level-set problem in $\mathbb{R}^+ \times \Omega$ can be written as

$$\partial_t u + |\nabla u| \left(\operatorname{tr}(a(\Sigma)(DN - DN^\sigma)) - (\operatorname{div} a(\Sigma)) \cdot (N - N^\sigma) \right) = 0.$$

And due to the equivalence of the level-set evolution $\partial_t u + |\nabla u|f = 0$ and its parametric version $\partial_t x = fN$ (cf. e.g. [74]) the assertion of the theorem is verified. \square

In the last theorem we have defined $S^\sigma = DN^\sigma$, but before we have been working with Σ , which coincides with $S_{T_x\mathcal{M}^\sigma}^\sigma = [S(\text{Id} - N \otimes N)]^\sigma$ on the regularized tangent space $T_x\mathcal{M}^\sigma$. In general we have

$$S^\sigma(\text{Id} - N^\sigma \otimes N^\sigma) \neq [S(\text{Id} - N \otimes N)]^\sigma =: \Sigma.$$

To verify this claim, let us recall, we have assumed that Σ diagonalizes with respect to the basis $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$. In case we obtain Σ by evaluating the curvatures on a globally pre-filtered image u^σ with the definitions

$$N^\sigma(x) := \frac{\nabla u^\sigma(x)}{|\nabla u^\sigma(x)|} \quad \text{and} \quad \Sigma(x) := DN^\sigma(x) \left(\text{Id} - N^\sigma(x) \otimes N^\sigma(x) \right),$$

for each $x \in \Omega$ we obviously have $\Sigma \equiv S^\sigma(\text{Id} - N^\sigma \otimes N^\sigma)$. But for local regularizations in each $x_0 \in \Omega$ and a separate evaluation of the shape operator on the local projections we have $u_{x_0}^\sigma(y)$ and define

$$N^\sigma(y) := \frac{\nabla_y u_{x_0}^\sigma(y)}{|\nabla_y u_{x_0}^\sigma(y)|} \quad \text{and} \quad \Sigma(x) := D_y N^\sigma(y) \left(\text{Id} - N^\sigma(y) \otimes N^\sigma(y) \right) \Big|_{y=x_0},$$

for which in general $\Sigma \neq S^\sigma(\text{Id} - N^\sigma \otimes N^\sigma)$. However there are configurations for which equality still holds. These configurations and the different forms of regularizations are discussed in detail in Section 3.3.

From Theorem 2.2 we have learned how the speed of the evolution depends on the difference between regularized and non-regularized quantities. Since the evaluation of the shape operator is based on second derivatives, whereas the computation of the normals rests on first derivatives, we expect for noisy images that

$$\|S^\sigma(x) - S(x)\| \gg \|N^\sigma(x) - N(x)\| \quad \text{for } x \in \Omega.$$

Thus, the trace-term $\text{tr}(a(\Sigma)(S^\sigma - S))$ is be dominant which allows the following characterization:

The anisotropic geometric level-set evolution is mainly steered by the difference between a shape operator S^σ on regularized data and the true shape operator S weighted by the coefficients of the diffusion tensor $a(\Sigma)$.

Let us remark that the evolution speed vanishes for $N^\sigma(x) = N(x)$ and $S^\sigma(x) = S(x)$. As we see in the sequel this leads to a rich class of invariant shapes under the evolution.

Remark 2.3. *Although we have based the exposition in this section on 3D images, we also consider examples of the analogous problem in 2D. It is clear that in the 2D case the level-sets are curves, which we characterize by one curvature quantity κ . Everything said so far regarding regularizations still applies in the 2D setting and clearly the evolution is steered by a diffusion tensor, which has the form*

$$a(\Sigma) = (w^{1,\sigma}, N^\sigma) \begin{pmatrix} G(\kappa^\sigma) & \\ & 0 \end{pmatrix} (w^{1,\sigma}, N^\sigma),$$

where κ^σ (which in this case is equal to $S_{T_x\mathcal{M}^\sigma}^\sigma$) is the regularized curvature and $w^{1,\sigma} = (N^\sigma)^\perp$ the tangent to the regularized level-sets.

The following figures demonstrate the performance of the method. In all applications, the domain is $\Omega := [0, 1]^d$ and the grid-width for the computations is set to $h = 2^{-8}$. As a test case, Figure 2.1 shows the anisotropic geometric diffusion of a noisy data-set, whose

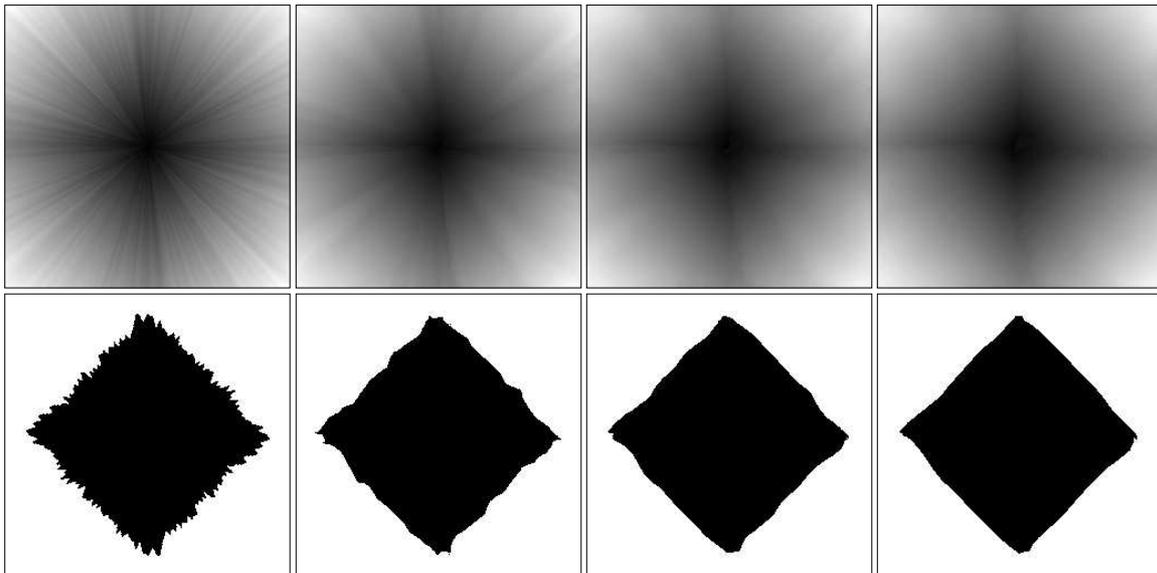


Figure 2.1: *The anisotropic geometric model has been applied to a sample data-set. The level-sets of the initial data are squares which are disturbed randomly in normal direction (see text). From left to right the initial image and several scale steps from the evolution are shown. In the top row the images are displayed, whereas in the bottom row extracted sub-level-sets $\{x : u(x) \leq .43\}$ from the corresponding images are shown.*

level-sets are 2D squares, under the anisotropic geometric diffusion. The smooth initial data is given by $\tilde{u}_0(x) := |x|_1$. The level-sets of this function are perturbed by a random factor $\text{rand}(x) \in [-12h, 12h]$ in their normal direction:

$$u_0(x) := \left| x + \text{rand}(x) \frac{\nabla \tilde{u}_0(x)}{|\nabla \tilde{u}_0(x)|_1} \right|. \quad (2.7)$$

Thus, the initial image for the evolution consists of level-sets, which are noisy squares. Figure 2.1 shows the noisy initial image and different scale steps from the evolution. Additionally a single sub-level-set has been extracted to show the preservation of the squares' corners better.

In Figure 2.2 the evolution of a 3D test data-set of size 65^3 (i.e. the grid-width is $h = 2^{-6}$) is shown. Similar to the 2D case, the level-sets of the initial image consists of perturbed octahedrons, but the perturbation is not restricted to the normal direction. Thus, the noisy initial data is

$$u_0(x) := |x + \text{rand}(x)|_1.$$

Figure 2.2 shows again several scale-steps of the evolution and additionally describes the evaluation of the curvatures of the level-sets. It is clearly visible how features of the level-sets are being retained.

Furthermore we apply the anisotropic geometric diffusion model to a real 3D data-set resulting from echo-cardiography of the human heart (cf. Figure 2.3). From the image-sequence which was acquired with a 3D ultrasound device during one cardiac cycle of the heart, we have taken one fixed image as the input to the model described here. The whole series consists of 16 images of size 129^3 (i.e. the grid-width is $h = 2^{-7}$). For several scale-steps of the evolution the figure shows always the extraction of the same iso-surface which corresponds to the blood concentration in the heart. Features clearly visible to the human eye in the noisy initial data-set are being retained, while the noise is significantly reduced with increasing scale.

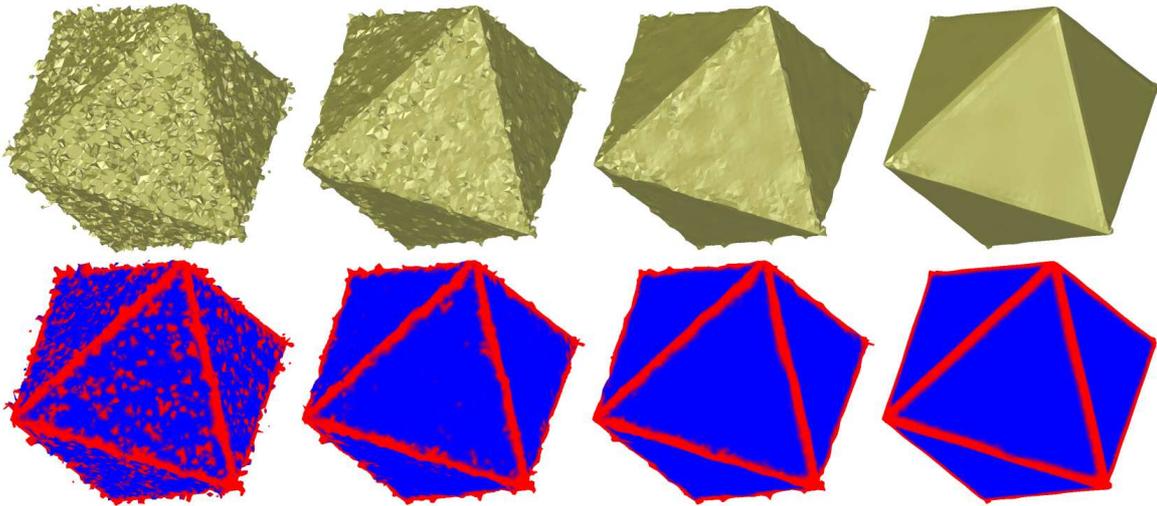


Figure 2.2: *The evolution of the octahedron data-set under the new model is considered. From left to right the noisy initial data and three scale-steps of the evolution are shown. In the top row, the extraction of the same iso-surface from the data is shown, whereas the bottom row illustrates the evaluation of curvature on the corresponding level-sets (cf. Section 3). A color ramp from blue (=small) to red (=high) indicates the value of the dominant curvature. The corners and edges of the original data are retained in a very good way during the evolution.*

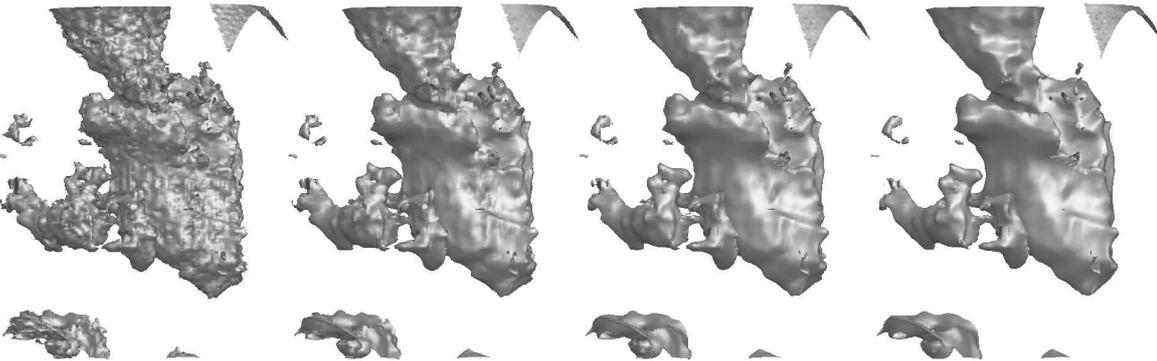


Figure 2.3: *From left to right several scale steps of the application of the anisotropic geometric diffusion to a real 3D ultrasound data-set are shown. Always the same iso-surface (corresponding to the blood concentration in the left ventricle of the heart) was extracted from the data-sets. The level-set of the noisy initial data (left) carries edges, which are kept during the process (see lower right area of the volume). The additional small volumes in front of the main iso-surface correspond to the blood in the other ventricle of the heart.*

The application of the diffusion model to a real data-set is shown in Figure 2.4. Again several scale steps of the evolution are displayed. In the introduction several other image processing models have been shown with the same initial data (cf. Figures 1.2, 1.3, and 1.5). A comparison of the model with the existing methodology which we have discussed in the introduction is given in Figure 2.5. The differences between the models are clearly visible from the magnified parts of the resulting images after the 10th scale step, which are shown in the Figure. Since the contrast of the image is very high for example at the boundary between the mountain and the sky, the non-morphological Perona-Malik model and the Weickert diffusion give good results. The anisotropic geometric diffusion keeps the corner of the shape of the objects very good in contrast to the mean-curvature motion.



Figure 2.4: *The evolution of a real data-set under the new model is shown. From left to right several scale-steps of the diffusion of the noisy image shown on the left in Figure 1.1 are shown. A detailed comparison of different image processing models is shown in Figure 2.5.*

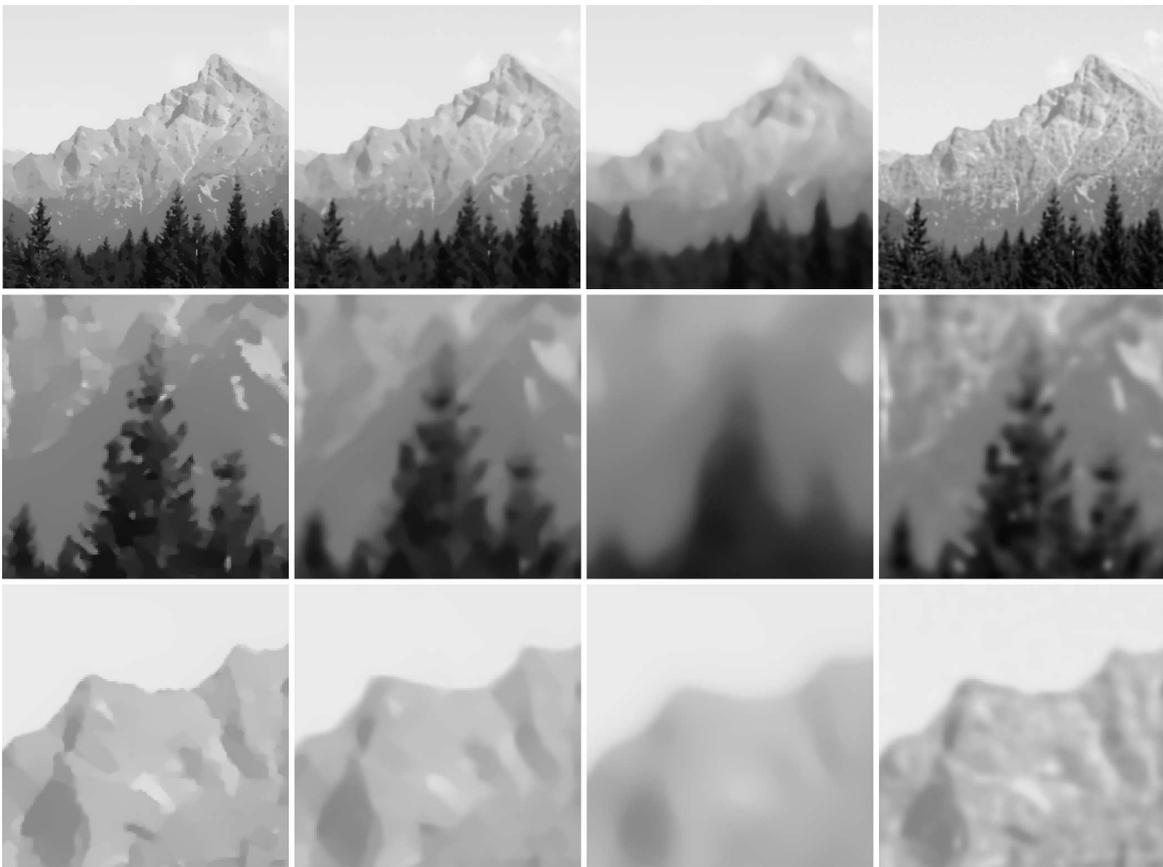


Figure 2.5: *A comparison of different models applied to the noisy data-set shown on the left in Figure 1.1 is displayed. From left to right the result from the Perona-Malik model, the Weickert diffusion, MCM and the anisotropic geometric diffusion model are shown. The top row shows the results of the different diffusion-models after 10 scale steps. The middle and the bottom row show magnified sections from the results in the top row. One clearly sees the difference between the Perona-Malik model and the Weickert diffusion, with its tangential smoothing. In contrast to MCM the anisotropic geometric model keeps the sharp edges of the shape of the trees and the mountains. For this application the non-morphological Perona-Malik model and the Weickert diffusion give very good results, since the contrast – and thus the gradients – in the image is quite high.*

2.2 Extracting motion velocities — optical-flow

We leave the processing of static images now and start working with image-sequences. The goal is to augment the anisotropic geometric diffusion for static images, which has been presented in the last section. The analysis of the underlying motion of the images (the *optical-flow problem*) is indispensable for the construction of an anisotropic model, which takes into account a nonlinear diffusion in the direction of this motion.

As already explained in the introduction, image-sequences result from the observation of a time-dependent physical process. An image acquisition device (e.g. MRI, CT, ultrasound, or a standard camera) takes static images $u_i : \Omega \rightarrow \mathbb{R}$ at certain (usually equally distant) times $s_i \in I$ within the observation interval $I \subset \mathbb{R}$. We call these single static images u_i , which constitute the image-sequence, the *frames* of the sequence. For the subsequent analysis it is convenient to relax this setting. Let us assume that the image-sequence data is continuous in time and space. Thus, we are working with a family of images on a time interval $I := [0, T], T \in \mathbb{R}^+$, which is smooth in time and space. We denote this continuous sequence by

$$u : I \times \Omega \rightarrow \mathbb{R}, \quad (s, x) \mapsto u(s, x),$$

and we call

$$u(s) : \Omega \rightarrow \mathbb{R}, \quad x \mapsto u(s, x),$$

the *frame at time s* of the image-sequence. Here, again $\Omega \subset \mathbb{R}^d, d = 2, 3$ is the spatial domain, which we assume to be fixed over time. Time variables are denoted by r or s , whereas x and y are spatial coordinates. Below, the multi-scale parameter is again denoted by t . We refer to the time of the image-sequence always as the *sequence-time*, whereas the evolution-time stays the *scale* just as before.

The focus now lies on the analysis of the intrinsic motion in the image-sequence. As before, we are interested in the iso-surfaces, for $c \in \mathbb{R}$ thus

$$\mathcal{M}_{c,s} := \{x \in \Omega : u(s, x) = c\}.$$

If the considered image-sequence is smooth and, as usually the case, non constant in time, the level-sets $\mathcal{M}_{c,s}$ change smoothly in time, too. Clearly we have the notions of a normal $N = N(s, x)$ of the level-set and the tangent space $T_x \mathcal{M}_{c,s}$ as before. One wants to express the motion of the level-sets in space and time (also referred to as *optical-flow*) by the vector field

$$v : [0, T] \times \Omega \rightarrow \mathbb{R}^3 \quad (s, x) \mapsto v(s, x).$$

such that a single trajectory $\{x(s)\}$, describing the path of a single point in space-time, is given by

$$\dot{x}(s) = v(s, x(s)).$$

At any point $x = x(s) \in \mathcal{M}_{c,s}$, we can project the velocity onto the tangent space $T_x \mathcal{M}_{c,s}$ and the normal space $\text{span}\{N(s, x)\}$ and thus obtain a splitting of $v = v_n + v_{\text{tg}}$ into a normal component v_n and a tangential part v_{tg} :

$$v_n(s, x) := v(s, x) \cdot N(s, x)N(s, x) \quad \text{and} \quad v_{\text{tg}}(s, x) := v(s, x) - v_n(s, x).$$

The optical-flow problem consists of the extraction of the velocity from the given image data. Early work on the optical-flow problem in a variational framework goes back to Horn and Schunck in 1981 [41]. Since then, the optical-flow problem has been considered widely for (2+1)D image-sequences [60]. A standard assumption easily leads to a formula describing the normal velocity v_n . Unfortunately, a formula for the tangential part is not obtained that easily. Rather the extraction of tangential components of v turns out to

be an ill-posed task, since tangential movements correspond to reparameterizations of the level-sets, which are impossible to observe. To treat this ill-posedness, various regularization techniques have been derived. For example one can consider elastic stresses or viscous fluid effects to aim for tangential components of v [33, 13, 47]. These approaches work well in case of large deformations, but are computationally expensive. Stochastic approaches have been considered as well [9, 55]. Various other techniques in the related area of image matching have been developed [77, 75, 28], where, however, one has only two images: a reference and a template image. The goal is to find a deformation which maps the template image onto the reference image.

2.2.1 Normal velocity

In the following we make different assumptions on the underlying image-sequence. This enables us to derive equations which describe the motion:

(A1) *Intensities are preserved along motion trajectories:*

$$u(s_0, x(s_0)) = u(s_0 + \tau, x(s_0 + \tau)) \quad \text{for} \quad -s_0 \leq \tau \leq T - s_0.$$

This assumption is reasonable since it is related to the invariance of the image acquisition device, which measures physical quantities like density and concentration. If this physical quantity moves in time, so does the corresponding image intensity, since at any time s same densities are mapped to the same intensities. With this assumption we rule out image-sequences in which e.g. the observed objects changes luminosity, shadows or opacity within time. In the literature [37, 71, 58] this assumption is referred to as *Lambertian motion assumption* or *brightness conservation constraint equation (BCCE)*.

The first assumption (A1) provides us with a condition for the normal part $v_n := v \cdot N N$ of the velocity. Since we have assumed smoothness of the sequence, we can define the normal N and tangent space $T_x \mathcal{M}$ to a level-set — as before in the setting for static images. Starting from a point $x_0 := x(s_0) \in \Omega$ within a frame $u(s_0)$ we differentiate (A1) with respect to τ and obtain

$$\partial_\tau \left(u(s_0 + \tau, x(s_0 + \tau)) - u(s_0, x_0) \right) \Big|_{\tau=0} = 0 \quad \Rightarrow \quad \partial_s u(s_0, x_0) + \nabla u(s_0, x_0) \cdot v(s_0, x_0) = 0.$$

Assuming $|\nabla u(s_0, x_0)| \neq 0$ and dividing by $|\nabla u(s_0, x_0)|$ we obtain an expression for the normal velocity

$$v_n(s_0, x_0) = v(s_0, x_0) \cdot N(s_0, x_0) N(s_0, x_0) = - \frac{\partial_s u(s_0, x_0)}{|\nabla u(s_0, x_0)|} N(s_0, x_0).$$

If we knew that v is orthogonal to the level-sets, by additional assumptions on the observed underlying physical process, we would have $v = v_n N$. In this case the motion would be completely described by the above equation. Such settings are considered in some applications at the end of this section.

2.2.2 Apparent velocity

In general one cannot assume that the tangential component v_{t_g} vanishes. But extracting these tangential motions is an ill-posed task. Consider, for example, a parameterized curve

$C(s, p) : [0, T) \times S^1 \rightarrow \mathbb{R}^2$ which moves in time s according to

$$\frac{\partial C}{\partial s} = v \quad \text{with} \quad v = v_n N + v_{\text{tg}} N^\perp,$$

and an arbitrary reparameterization $\tilde{C}(s, p) = C(s, \tilde{p}(s, p))$. Then the reparameterization corresponds to a tangential motion, because

$$\frac{\partial \tilde{C}}{\partial s} = \frac{\partial C}{\partial s} + \frac{\partial C}{\partial p} \frac{\partial \tilde{p}}{\partial s} = v_n N + v_{\text{tg}} N^\perp + \frac{\partial C}{\partial p} \frac{\partial \tilde{p}}{\partial s} = v_n N + v_{\text{tg}} N^\perp + \frac{\partial \tilde{p}}{\partial s} \frac{\partial l}{\partial p} N^\perp,$$

where we used the fact that for the arclength l the relation

$$\frac{\partial C}{\partial p} = \frac{\partial C}{\partial l} \frac{\partial l}{\partial p} = \frac{\partial l}{\partial p} N^\perp$$

holds [27]. Thus, the reparameterization corresponds to the tangential motion $w := \frac{\partial \tilde{p}}{\partial s} \frac{\partial l}{\partial p} N^\perp$ which does not change the curve as can be seen easily from the level-set formulation: Let \mathcal{M}_c be the level-set defining the curve C . As above, we have for any $x \in \mathcal{M}_c$

$$u(s, x(s)) = c \quad \Rightarrow \quad \partial_s u + \nabla u \cdot \partial_s x = \partial_s u + \nabla u \cdot v = 0,$$

but since $\nabla u \parallel N$, we have $w \cdot \nabla u = 0$. Consequently, tangential motions of level-sets cannot be extracted from an image-sequence since they only change parameterizations of the level-sets, still defining the same sets in \mathbb{R}^d .

With the following assumption we restrict the set of possible tangential velocities which leads us to the notion of the *apparent velocity*.

(A2) *Locally the underlying motion is a translation:*

$$N(s_0, x(s_0)) = N(s_0 + \tau, x(s_0 + \tau)) \quad \text{for} \quad -s_0 \leq \tau \leq T - s_0.$$

Assuming that the scenery consists of solid objects being translated in space, this assumption is of course fulfilled.

This time we differentiate (A2) with respect to τ and evaluate the result at $\tau = 0$ to obtain

$$\partial_\tau N(s_0 + \tau, x(s_0 + \tau)) \Big|_{\tau=0} = 0 \quad \Rightarrow \quad \partial_s N(s_0, x_0) + DN(s_0, x_0)v(s_0, x_0) = 0. \quad (2.8)$$

We analyze this equation more closely: We already know from Section 1.2 that $DN = S : \mathbb{R}^3 \rightarrow T_x \mathcal{M}$. Moreover $\partial_s N \perp N$, because

$$0 = \partial_s(N \cdot N) = 2\partial_s N \cdot N,$$

and therefore $\partial_s N$ is also contained in the tangent space $T_x \mathcal{M}$. Thus, equation (2.8) can be interpreted pointwise as an equation on the tangent space, and we write

$$\partial_s N(s_0, x_0) + S_{T_x \mathcal{M}} v_{\text{tg}}(s_0, x_0) = 0,$$

because $S_{T_x \mathcal{M}} := S(\text{Id} - N \otimes N) = S|_{T_x \mathcal{M}}$. Now, if the shape operator $S_{T_x \mathcal{M}}$ is invertible, we can solve this equation for v_{tg} to obtain

$$v_{\text{tg}}(s_0, x_0) = -S_{T_x \mathcal{M}}^{-1} \partial_s N(s_0, x_0). \quad (2.9)$$

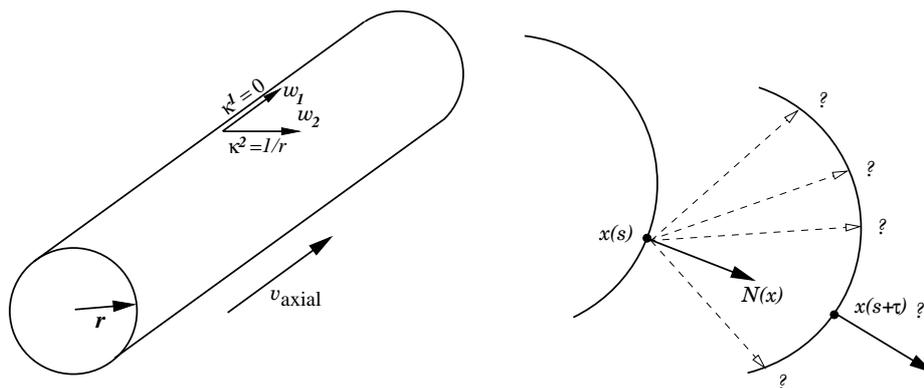


Figure 2.6: *Left: In case of an (infinitely extended) moving cylinder the velocity component v_{axial} parallel to the axis of the cylinder can not be extracted. The image-sequence is indeed invariant under motions along this axis. This corresponds to the fact that the axis of the cylinder is the principal direction of curvature for the eigenvalue zero. Right: From the set of all possible solutions the apparent velocity v_{app} is the one minimizing the variations of the normal in direction of the trajectory (cf. Guichard [37]).*

At this point it becomes clear, which circumstances allow us to speak of a tangential velocity: If the shape operator $S_{T_x\mathcal{M}}$ is invertible, assuming the presence of local translations, we can define a tangential velocity v_{tg} by means of the latter formula. The invertibility of $S_{T_x\mathcal{M}}$ implies that its eigenvalues do not vanish. Since the eigenvalues are the principal directions of curvature of the level-set a necessary condition for the extraction of v_{tg} is a non-vanishing Gaussian curvature ($K = \kappa^1\kappa^2$) of the level-surfaces. Such a situation may correspond to the presence of a surface feature such as a corner or an edge. In contrast to that we know that for a non-invertible shape operator the level-set is flat in at least one direction (cf. Figure 2.6). For example we can imagine (infinitely extended) concentric cylinders, moving in time and this way generating an image-sequence. On one hand we cannot see this motion, if the cylinders move in direction of their axis. On the other hand the direction of their axis is the principal direction in which the cylinders are flat, i.e. their curvature is zero. For situations which are locally like this cylindrical case, we cannot expect to be able to extract the tangential motion. For planar cases ($\kappa^1 = \kappa^2 = 0$), where both curvatures vanish, the extraction of tangential motions is impossible as well.

As a logical consequence of this example, we would like to treat situations, which are locally like the cylindrical or planar case, such that our definition of the tangential velocity still makes sense. Here we use a tool from numerical linear algebra, where the inversion of singular systems of equations is treated by the *singular value decomposition* or the *pseudo-inverse* [25]. If in our case the Shape operator $S_{T_x\mathcal{M}}$ is singular (corresponding to the cylindrical or planar case), we replace the inverse $S_{T_x\mathcal{M}}^{-1}$ in the definition (2.9) of the tangential velocity v_{tg} by its pseudo-inverse $S_{T_x\mathcal{M}}^\dagger$. For any matrix $W \in \mathbb{R}^{n \times m}$ the pseudo-inverse W^\dagger is uniquely defined by the Penrose axioms

$$(W^\dagger W)^T = W^\dagger W, \quad (W W^\dagger)^T = W W^\dagger, \quad W^\dagger W W^\dagger = W^\dagger, \quad W W^\dagger W = W.$$

From these axioms one derives that $W^\dagger W$ is an orthogonal projection from \mathbb{R}^m onto $\{v \in \mathbb{R}^m : Wv = 0\}^\perp$. In view of numerical approximations and noisy image data, we modify this definition slightly, by considering a projection $S_{T_x\mathcal{M}}^\dagger S_{T_x\mathcal{M}}$ from \mathbb{R}^d onto the space

$$T_\delta := \text{span}\{w^i : |\kappa^i| > \delta, i = 1, 2\}.$$

Thus, $S_{T_x\mathcal{M}}^\dagger$ inverts $S_{T_x\mathcal{M}}$ only on T_δ , i.e.

$$S_{T_x\mathcal{M}}^\dagger = (w^{1,\sigma}, w^{2,\sigma}) \begin{pmatrix} f_\delta(\kappa^{1,\sigma}) & 0 \\ 0 & f_\delta(\kappa^{2,\sigma}) \end{pmatrix} (w^{1,\sigma}, w^{2,\sigma})^T \quad \text{with} \quad f_\delta(s) = \begin{cases} \frac{1}{s} & \text{if } |s| > \delta, \\ 0 & \text{else,} \end{cases}$$

and so

$$S_{T_x\mathcal{M}}^\dagger S_{T_x\mathcal{M}} = \begin{cases} \text{Id} & \text{on } T_\delta, \\ 0 & \text{else.} \end{cases}$$

Obviously for invertible Shape operator and $\delta = 0$ the pseudo inverse $S_{T_x\mathcal{M}}^\dagger$ coincides with the inverse $S_{T_x\mathcal{M}}^{-1}$. Finally we have the definition

$$v_{\text{tg}}(s_0, x_0) = -S_{T_x\mathcal{M}}^\dagger \partial_s N(s_0, x_0),$$

which in regular situations with invertible Shape operator and for $\delta = 0$ gives us a continuous tangential velocity v_{tg} . However, we emphasize here that for $\delta > 0$ the resulting tangential velocity need not be continuous.

Together with the equation for the normal component we obtain the following formula

$$v_{\text{app}}(s_0, x_0) = - \left(\frac{\partial_s u}{|\nabla u|} N + S_{T_x\mathcal{M}}^\dagger \partial_s N \right) (s_0, x_0), \quad (2.10)$$

which defines the so called *apparent velocity*. It is a generalization of the velocity which was derived for 2D image-sequences by Guichard in [36, 37]. Equation (2.10) is valid for $d \geq 2$, since it involves the intrinsic geometric shape operator $S_{T_x\mathcal{M}}$. Finally we can characterize the apparent velocity as follows for $\delta = 0$ and existing inverse $S_{T_x\mathcal{M}}^{-1}$:

From the set of all possible tangential motions the apparent velocity v_{app} selects the one that corresponds to a translation of the normal. Thus, it minimizes the variations of the normal along the trajectories of the level-sets (cf. Figure 2.6).

The fact, that the apparent velocity minimizes the variations of the normal along trajectories is a direct consequence of its construction. Denoting the *material derivative* (i.e. the space-time derivative along trajectories) with $\frac{D}{\partial s}$ we have

$$\frac{D}{\partial s} N := \partial_s N + DN v_{\text{app}} = \partial_s N - DN \left(\frac{\partial_s u}{|\nabla u|} N + S_{T_x\mathcal{M}}^{-1} \partial_s N \right),$$

but according to (2.8) $\partial_s N = -DN v_{\text{app}}$ and so

$$\frac{D}{\partial s} N = 0. \quad (2.11)$$

Still we stick to the assumption that the definition of a normal is possible, i.e. the spatial gradient of the images does not vanish. This assumption is dropped as soon as the apparent velocity is going to be regularized in an appropriate way in Section 3. Let us finally ensure that the apparent velocity indeed is a morphological quantity.

Lemma 2.4. *The apparent velocity v_{app} depends only on the morphology of the image.*

Proof. We are going to show that the apparent velocity v_{app} is invariant under a strictly monotone C^1 -transformation $h : \mathbb{R} \rightarrow \mathbb{R}$ of the gray-value. To this end let us consider the

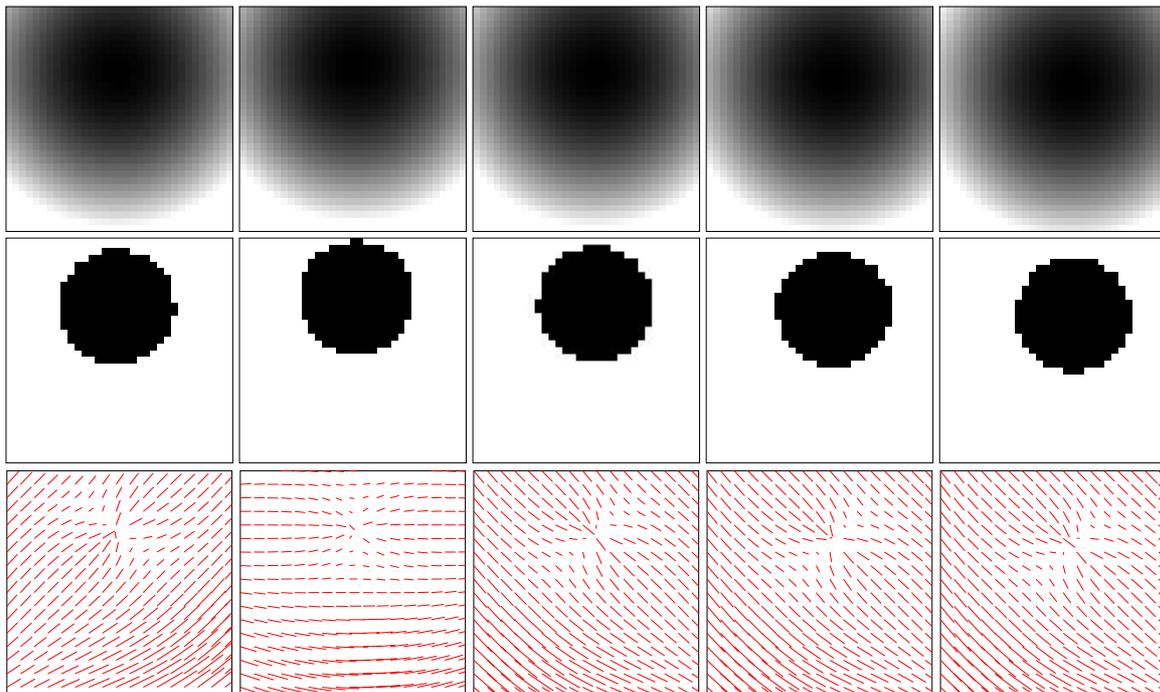


Figure 2.7: The extraction of motion velocities from a 2D sample data-set is displayed. From left to right several frames of the sequence are shown, whose level-sets are squares bouncing at some invisible object. The top row shows the images of the sequence, and in the middle row the extraction of always the same level-set from the frames is shown. Finally the bottom row shows the extracted velocities. The computations were performed on a 33×33 grid.

terms in (2.10) separately and denote the quantities resulting from the transformed image with a tilde ($\widetilde{}$). We already know from Section 1.2 that

$$\widetilde{S_{T_x \mathcal{M}}} = S_{T_x \mathcal{M}}$$

and therefore using $h' > 0$

$$\widetilde{v_{\text{tg}}} = \widetilde{S_{T_x \mathcal{M}}}^\dagger \partial_s \widetilde{N} = S_{T_x \mathcal{M}}^\dagger \frac{h'(u)}{|h'(u)|} \partial_s N = S_{T_x \mathcal{M}}^\dagger \partial_s N = v_{\text{tg}}.$$

Analogously for the normal component we have

$$\widetilde{v}_{\text{n}} = \frac{\partial_s h(u)}{|\nabla h(u)|} \widetilde{N} = \frac{h'}{|h'|} \frac{\partial_s u}{|\nabla u|} \frac{h'}{|h'|} N = \frac{\partial_s u}{|\nabla u|} N = v_{\text{n}},$$

and so $\widetilde{v_{\text{app}}} = v_{\text{app}}$. \square

In Figure 2.7 the extraction of velocities of a sample 2D data-set is shown. The image-sequence consists of 40 frames

$$u(s_i, x) := |x - d(s_i)|, \quad i = 1, \dots, 40, \quad \text{with} \quad d(s_i) = \begin{cases} s_i & \text{if } s_i < 20, \\ 40 - s_i & \text{else,} \end{cases}$$

i.e. $d(s)$ models the bouncing of the images at the upper boundary. Each level-set traced during the image-sequence therefore shows a circle bouncing at some invisible object at the top of the domain.

The application of the optical-flow extraction to a real data-set is shown in Figure 2.8. The classical taxi-sequence is one of the standard applications in the image processing

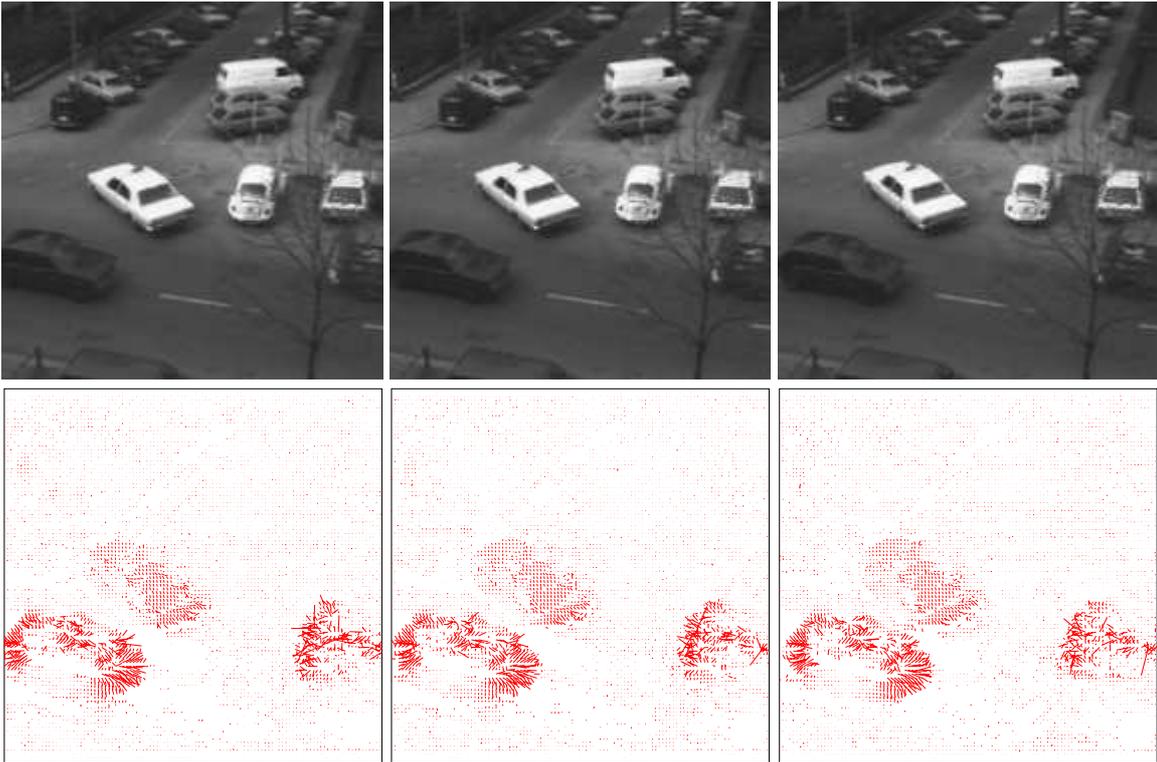


Figure 2.8: *The extraction of velocities from the classical taxi-sequence data-set is shown. The top row shows several frames from the sequence, whereas the extracted velocities are drawn in the bottom row.*

community. However, we are not going to compare our results from the taxi sequence with existing methodology, since the main scope of the approach are 3D applications.

A sample 3D image-sequence is considered in computations whose results are depicted in Figure 2.10. The image-sequence consists of ellipsoidal level-sets, which oscillate over one half-axis in time. Thus, each frame of the sequence is given by

$$u(s, x) := \frac{x_1^2}{a(s)} + \frac{x_2^2}{b(s)} + x_3^2, \quad \text{with} \quad a(s) = 4s - (1 - s), b(s) = s - 4(1 - s)$$

for sequence times $s \in [0, 1]$. Figure 2.9 shows on a certain level-set the splitting of the velocity into the normal and the tangential component.

Figure 2.11 shows the extraction of velocities from the observation of a porous medium flow experiment. The *salt pool experiment* [62] consists of a cubical container containing small solid balls, thus modeling a porous medium. First the container is filled with fresh water. Then a hole is opened at the bottom of this container and a salt-concentration is flowing in. Finally an outlet is opened at one of the top corners of the container through which the liquid leaves the pool. During the experiment the salt concentration is measured using a MRI device and the resulting image-sequence is taken as an input for the velocity extraction. Since the porous medium flow obeys Darcy's law, we conclude that in this simple setting the flow is aligned with the concentration gradient, i.e. the normal to the level-sets. Therefore in this case it is indeed sufficient to extract the normal velocities from the given image-sequence.

The processing of a medical data-set is shown in Figure 2.12. From the noisy ultrasound image-sequence showing the human heart during one cardiac cycle the motions of the level-sets of the blood concentration have been extracted. Since the image data is very noisy each frame has been treated with the level-set method for anisotropic geometric diffusion

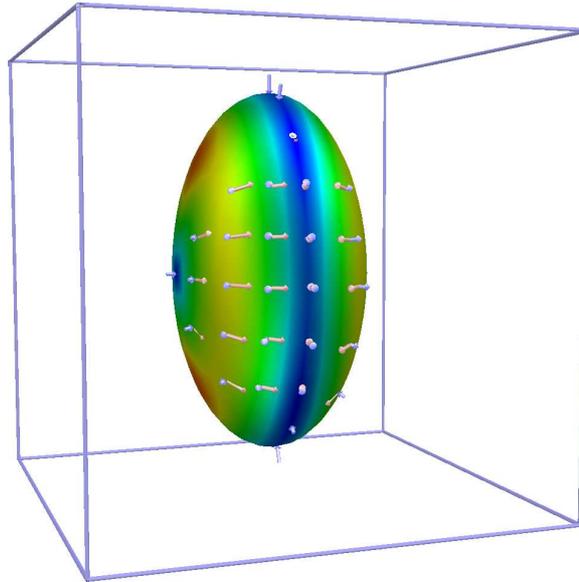


Figure 2.9: For one frame of the image-sequence showing oscillating ellipsoidal level-sets, the splitting of the apparent velocity into its normal (blue arrows) and tangential (red arrows) component is shown. The color coding from blue to green to red indicates the absolute value $|v_{\text{tg}}|$ of the tangential velocity.

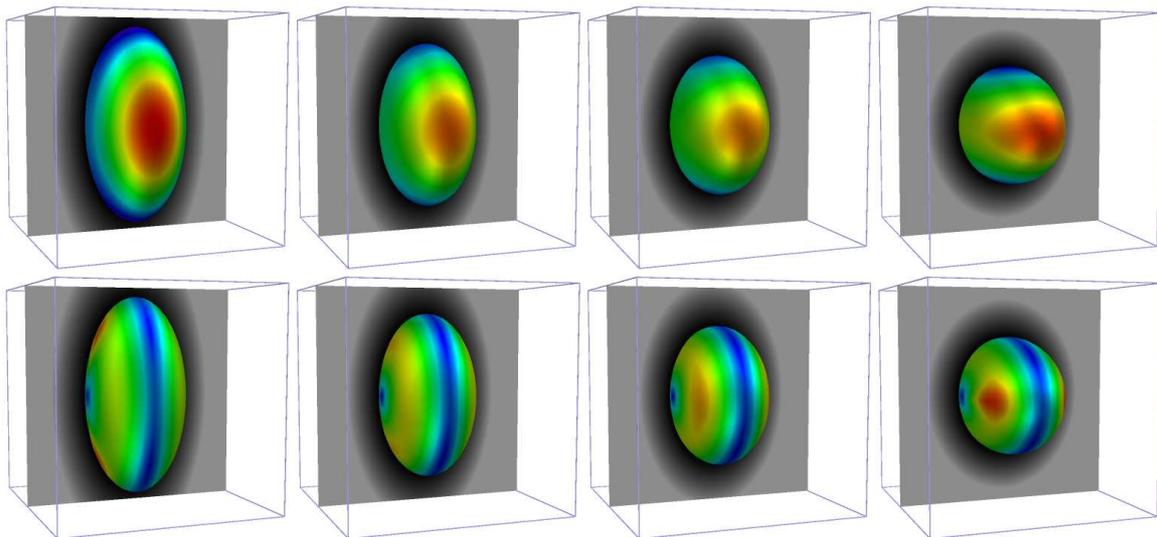


Figure 2.10: Velocities have been extracted from a 3D sample data-set. The image-sequence shows level-sets which are ellipsoids oscillating over one half axis (see text). From left to right several frames of the sequence are shown. Always the same iso-surface was extracted and — to emphasize the 3D character of the data — drawn together with a slice through the volume. In the top row the color coding shows the normal component of the velocity v_{n} (blue = moving inward, red = moving outward). In the bottom row the color ramp from blue to green to red shows the absolute value of the tangential component $|v_{\text{tg}}|$.

(cf. Section 2.1) in advance. This can be considered as a sophisticated kind of regularization which however does not take the sequence character of the images into account. The splitting of the apparent velocity v_{app} into the normal and the tangential component is demonstrated in Figure 2.13, where the regularizing effect of the pseudo-inverse is shown.

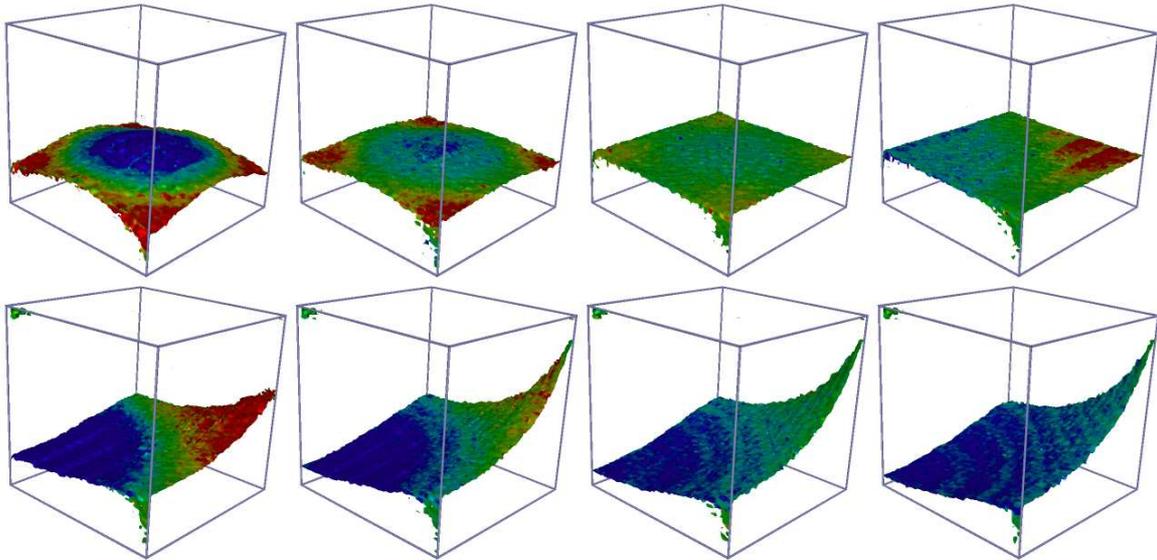


Figure 2.11: A real 3D data-set is considered as the input for the motion extraction process. The physical experiment being observed is the flow of a fluid in a porous medium container. From top left to bottom right several stages of the experiment are depicted. Always the same level-set has been extracted from the data. Again the color-coding shows the normal velocity of the iso-surfaces (cf. Figure 2.10). The upper row shows data from the first stage of the experiment, where a salt concentration flows into a pool filled with fresh water. In the lower row the second part of the experiment is shown, where the fluid leaves the container through a hole at the top left corner. All computations were performed on a 65^3 grid. The experimental data has been used by courtesy of S. Oswald [62].

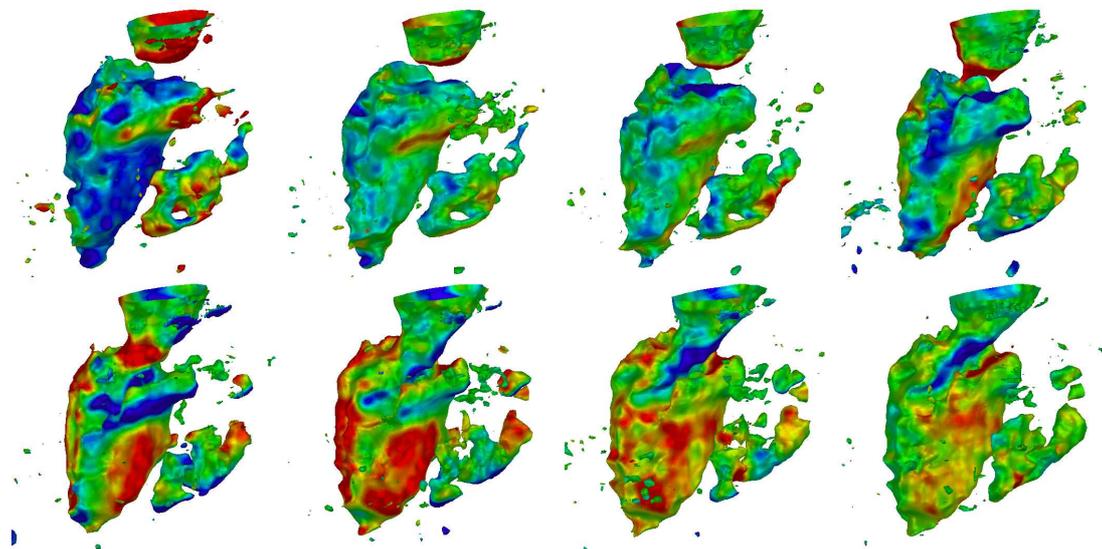


Figure 2.12: From an echo-cardiographical data-set, showing the left ventricle of the human heart during one cardiac cycle, velocities of the level-sets have been extracted. From top left to bottom right a color coding (cf. Figure 2.10) shows the normal component of the velocity always on the same iso-surface of different frames of the sequence. Each frame has been processed with the anisotropic geometric diffusion method for static images. The splitting of the apparent velocity into its normal and tangential component is shown in Figure 2.13. All computations have been performed on a 129^3 grid. The data has been used by courtesy of C. Lamberti from DEIS, Bologna University.

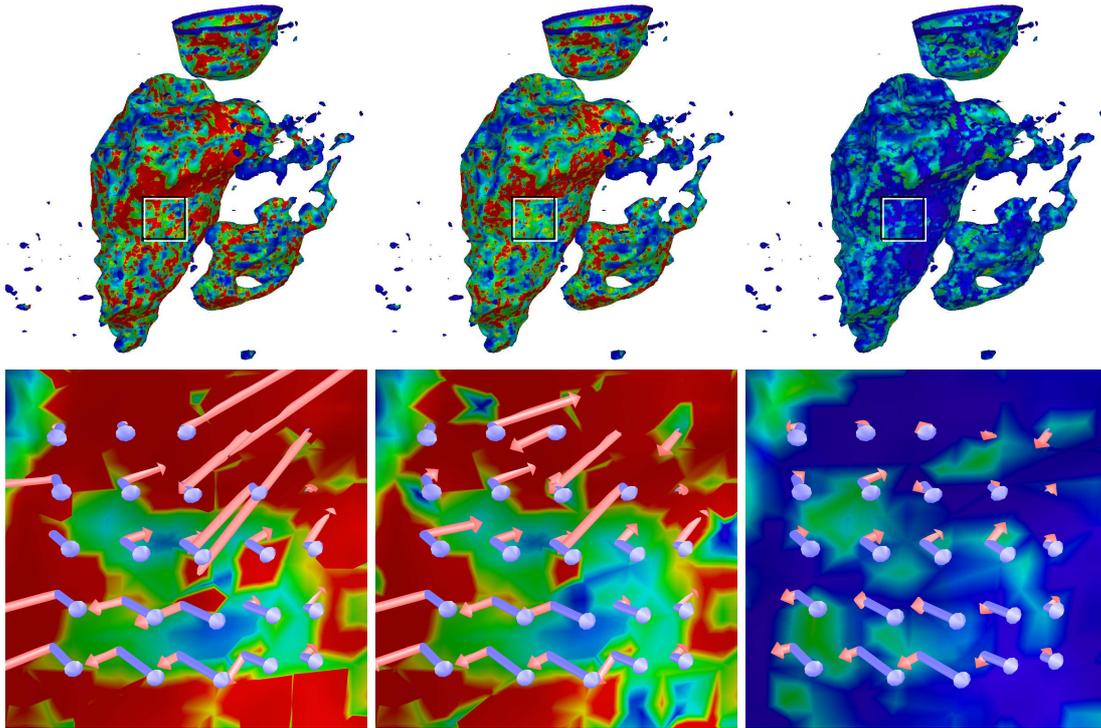


Figure 2.13: For one frame of the ultrasound data-set the splitting of the apparent velocity v_{app} into its tangential and normal component is shown. The top row shows the absolute value $|v_{\text{tg}}|$ of the tangential velocity with a color coding from blue to green to red, whereas the bottom row shows magnified sections of the extracted iso-surfaces. Arrow symbols show the tangential (red arrows) and the normal (blue arrows) part of v_{app} . From left to right the threshold value δ for the computation of the pseudo inverse of the regularized shape operator Σ^\dagger (cf. Section 3) is 0.1, 1 and 10, corresponding to local radii $r := 1/\delta$ of the objects being damped out, while the computational domain is $\Omega = [0, 1]^3$. Clearly the tangential motion becomes more regular, as the threshold δ suppresses more and more geometric noise.

2.2.3 General motions

The apparent velocity is based on the assumption that the underlying motion is locally a translation. Based on this assumption we could state the local invariance of the normals along motion trajectories (A2) and from that derive a formula for the tangential part v_{tg} . Let us for a while consider motion described by an arbitrary vector field. As already discussed in the last section the extraction of the motion velocities is an ill-posed task, because for any given v respecting the gray value invariance (A1) we can add any $w \in T\mathcal{M}$ from the tangent bundle, with $v + w$ generating the same image-sequence: There is no continuous dependence of the solution of the optical-flow problem on the input data and the resulting vector-fields can become arbitrary irregular. From the set of all possible solutions, the apparent velocity v_{app} is one choice which minimizes the variations of the normal (cf. (2.11)) by definition and due to assumption (A2). In the following paragraph we try to relax this assumption (A2) and take general motions into account. Unfortunately, this still gives no more information on the tangential velocity but leads to a condition for the equality $v = v_{\text{app}}$, moreover it does not deliver a comparison between the real velocity and the apparent velocity for arbitrary motions.

The velocity v induces a flux for the points $x_0 = x(s_0) \in \Omega$ contained in a level-set $\mathcal{M}(s_0)$, which is denoted by

$$\Theta_\tau(s_0, x_0) = x(s_0 + \tau) \quad \text{with} \quad \dot{x}(r) = v(r, x(r)).$$

This flux also describes the deformation of the image frame $u(s_0)$ to the frame $u(s_0 + \tau)$. Of course the normals of the level-sets deform according to the flux. Precisely, the transformation of the normals is controlled by the transposed inverse of the Jacobian of the flux:

$$N(s_0 + \tau, x(s_0 + \tau)) = \frac{(\nabla\Theta_\tau)^{-T}N}{|(\nabla\Theta_\tau)^{-T}N|}(s_0, x_0). \quad (2.12)$$

For the following analysis it is convenient to consider a first order series expansion of the flux' Jacobian:

$$\nabla\Theta_\tau = \text{Id} + \tau\nabla v + o(\tau^2),$$

and thus, for the transposed inverse of the Jacobian of Θ_τ

$$(\nabla\Theta_\tau)^{-T} = \text{Id} - \tau(\nabla v)^T + o(\tau^2).$$

Now, we differentiate the identity (2.12) above, written as

$$|(\nabla\Theta_\tau)^{-T}(s_0, x_0)N(s_0, x_0)| N(s_0 + \tau, x(s_0 + \tau)) = (\nabla\Theta_\tau)^{-T}(s_0, x_0)N(s_0, x_0), \quad (2.13)$$

with respect to τ and evaluate it at the starting point, i.e. $\tau = 0$. Recalling that for any vector-valued mapping $W(\tau)$ the derivative of the norm of W can be computed with

$$\partial_\tau |W(\tau)| \Big|_{\tau=0} = \frac{W(0)}{|W(0)|} \cdot \left(\partial_\tau W(\tau) \Big|_{\tau=0} \right),$$

we can compute the first factor in the left-hand-side of (2.13) to obtain

$$\begin{aligned} \partial_\tau |(\nabla\Theta_\tau)^{-T}N| &= N \cdot \left(\partial_\tau (\nabla\Theta_\tau)^{-T} \Big|_{\tau=0} N \right) \\ &= -N \cdot (\nabla v)^T N. \end{aligned}$$

Using this, one gets from (2.13)

$$\begin{aligned} 0 &= |(\nabla\Theta_\tau)^{-T}(s_0, x_0)N(s_0, x_0)| \partial_\tau N((s_0) + \tau, x((s_0) + \tau)) \Big|_{\tau=0} \\ &\quad + \partial_\tau |(\nabla\Theta_\tau)^{-T}N| \Big|_{\tau=0} N(s_0, x_0) - \partial_\tau (\nabla\Theta_\tau)^{-T} \Big|_{\tau=0} N - (\nabla\Theta_\tau)^{-T} \partial_\tau N(s_0, x_0) \\ &= \partial_s N + \nabla N v - (\nabla v)^T N \cdot N N + (\nabla v)^T N. \end{aligned}$$

In the last row, we have omitted the arguments for the clearness of the presentation, keeping in mind the terms are evaluated at (s_0, x_0) . Using the projection $\Pi := (\text{Id} - N \otimes N)$ on the tangent space $T_x \mathcal{M}$, and $S = DN = |\nabla u|^{-1} \Pi D^2 u$ (cf. (1.2)), we can express this result as

$$0 = [\partial_s N + Sv] + (\text{Id} - N \otimes N) ((\nabla v)^T N). \quad (2.14)$$

As (2.8), this is an equation on the tangent space $T_x \mathcal{M}$ as well, and it is the key to a better understanding of the apparent velocity v_{app} . Writing

$$\nabla(v \cdot N) = (\nabla v)^T N + (\nabla N)^T v = (\nabla v)^T N + S^T v$$

and therefore

$$(\nabla v)^T N = \nabla(v \cdot N) - S^T v,$$

and substituting this into (2.14), which we interpret on the tangent space $T_x \mathcal{M}$, we find

$$0 = \partial_s N + S_{T_x \mathcal{M}} v_{\text{tg}} + \Pi (\nabla(v \cdot N) - S^T v). \quad (2.15)$$

This equation can be simplified by using the relation

$$S_{T_x \mathcal{M}} = S\Pi \quad \Leftrightarrow \quad S_{T_x \mathcal{M}} = S_{T_x \mathcal{M}}^T = \Pi S^T$$

and the linearity of the projection to obtain

$$0 = [\partial_s N + S_{T_x \mathcal{M}} v_{\text{tg}}] + \Pi \nabla v_n - S_{T_x \mathcal{M}} v_{\text{tg}} = \partial_s N + \Pi \nabla v_n. \quad (2.16)$$

So if the shape operator $S_{T_x \mathcal{M}}$ is invertible and the velocity v solves the partial differential equation

$$\Pi \nabla v_n - S_{T_x \mathcal{M}} v_{\text{tg}} = 0,$$

then $v = v_{\text{app}}$, because the remaining term in brackets ($[\partial_s N + S_{T_x \mathcal{M}} v_{\text{tg}}] = 0$) uniquely defines $(v_{\text{app}})_{\text{tg}}$ (cf. equation (2.10)). Unfortunately there is not much additional information contained in (2.16): The tangential part has vanished from the equation. This again shows the ill-posedness of the problem, but also that it is impossible to obtain a comparison between v and v_{app} . Moreover we see from (2.16) that the variation of the normals in direction of the trajectory must be captured by a tangential change of the normal velocity, which is indeed the case for a velocity that fulfills assumption (A1) as can be verified by a straight-forward calculation.

Coming back to the case $v = v_{\text{app}}$, we can insert the apparent velocity into the left part of equation (2.16) to see that

$$\begin{aligned} 0 &= \Pi \left(\nabla(v_{\text{app}} \cdot N) - S_{T_x \mathcal{M}} v_{\text{app}} \right) \\ &= \Pi \nabla(v_{\text{app}} \cdot N) - S_{T_x \mathcal{M}} v_{\text{app}}, \end{aligned}$$

which is an equation on the tangent space. And so

$$(v_{\text{app}})_{\text{tg}} = S_{T_x \mathcal{M}}^{-1} \Pi \nabla(v_{\text{app}})_n = S^{-1} \nabla(v_{\text{app}})_n.$$

Thus, the tangential component of the apparent velocity is the variation of the normal velocity weighted on the tangent space according to the shape of the level-set.

2.3 Apparent acceleration

The final goal is to derive a model for image-sequence de-noising, simultaneously in space and sequence-time. Since we have derived an expression, which allows the extraction of an apparent velocity from the underlying data, we can now analyze how much the apparent trajectories $x(s)$ with

$$\dot{x}(s) = v_{\text{app}}(s, x(s))$$

of the level-sets are curved within sequence-time. This would enable to distinguish between uniform and rapidly changing motions. The physical quantity describing the variation of the velocity along the trajectories is the *acceleration vector*, defined by

$$\begin{aligned} \mathbf{a}(v) &:= \partial_\tau v(s + \tau, x(s + \tau)) \Big|_{\tau=0} \\ &= \partial_s v + (\nabla v)v. \end{aligned}$$

Since we are not going to give any further interpretation of the apparent acceleration vector \mathbf{a}_{app} we close this section with the definition of the *apparent acceleration* $\text{accel}_{\text{app}}$ as the length of \mathbf{a}_{app} :

$$\text{accel}_{\text{app}} := |\partial_s v_{\text{app}} + (\nabla v_{\text{app}})v_{\text{app}}|,$$

assuming we have enough regularity to perform the differentiation. Later we will perform the differentiation coupled with a regularization, which ensures the existence of the derivatives. The apparent acceleration is a measure for the curvature of the apparent trajectories in sequence-time. In the work of Guichard [37] for 2D image-sequences this quantity was also referred to as the *curvature of Lambertian trajectories* (CLT).

2.4 Coupled spatio-temporal smoothing of image-sequences

We now consider the case that the given image-sequence is noisy, where this noise results for example from the electronic acquisition process of the sequence. One possibility to de-noise the sequence is the application of the anisotropic geometric evolution from Section 2.1 to each frame of the sequence. This has been done for the data-set shown in Figure 2.12 (cf. [57, 58]). Unfortunately this approach does not respect the underlying correlation between the different frames. In the worst case it can destroy the correlation and the processed image-sequence may be non-smooth in the sequence-time direction.

Approaches taking into account the apparent curvature of the trajectories $\text{accel}_{\text{app}}$ have been considered by Mikula et al. in [71, 58]. Their models consider a smoothing in sequence time direction which is directly proportional to the value of the apparent acceleration. For highly accelerated motions this results in a significant smoothing of the highly curved trajectories in sequence-time (cf. Section 2.5), resulting in a decrease of this temporal curvature. Such a behavior however is undesired in many applications. In this section we derive a model, which also behaves nonlinear in the sequence-time direction and thus avoids this smoothing out of “temporal edges”. The model takes into account the additional information of the sequence — given in form of the motion v_{app} and acceleration $\text{accel}_{\text{app}}$ of the level-sets, which we are now able to approximate. Again suitable regularizations are indispensable with respect to noisy image-sequences.

As the static image processing model we base the image-sequence processing model on regularized quantities which are again denoted by a superscript σ . We do not want to sacrifice the good properties of the anisotropic level-set method from Section 2.1. Therefore the purely spatial diffusion does not differ from the one presented for static images. In

addition, we consider a smoothing in direction of the apparent velocity v_{app} . Thus, we introduce the space-time apparent velocity

$$V_{\text{app}}^\sigma := \frac{1}{\sqrt{1 + |v_{\text{app}}^\sigma|}} \begin{pmatrix} 1 \\ v_{\text{app}}^\sigma \end{pmatrix},$$

which has the direction of the apparent velocity in space, but points ‘‘across’’ the image frames in the sequence-time direction. The diffusivity in that direction is steered by the curvature of the trajectory in sequence-time, i.e. the apparent acceleration $\text{accel}_{\text{app}}$. We end up with the following coupled spatio-temporal evolution problem in $Q := I \times \Omega$ (cf. [57]):

For a given sequence $u_0 : Q \rightarrow \mathbb{R}$ find a function $u : \mathbb{R}^+ \times Q \rightarrow \mathbb{R}$ which is a viscosity solution of the following initial-value problem

$$\begin{aligned} \partial_t u - |\nabla_{(s,x)} u| \operatorname{div}_{(s,x)} \left(a_{(s,x)} \frac{\nabla_{(s,x)} u}{|\nabla_{(s,x)} u|} \right) &= 0 && \text{in } \mathbb{R}^+ \times Q, \\ u(0, s, x) &= u_0(s, x) && \text{in } Q, \end{aligned} \quad (2.17)$$

where $\nabla_{(s,x)} := (\partial_s, \nabla)$ and $\operatorname{div}_{(s,x)}$ is the dual operator to $\nabla_{(s,x)}$ in the $L^2(Q)$ -sense. The function u shall satisfy one of the following boundary conditions:

(BC1) *It satisfies a Neumann boundary condition on the whole spatio-temporal boundary $\partial Q := \{0, T\} \times \Omega \cup (0, T) \times \partial\Omega$, i.e. if $\nu_{(s,x)}$ denotes the outer normal to the sequence-time/space cube Q , then*

$$\nabla_{(s,x)} u \cdot \nu_{(s,x)} = 0 \quad \text{on } \mathbb{R}^+ \times \partial Q,$$

or

(BC2) *if the initial data u_0 is periodic in sequence time, u retains this periodicity and fulfills a Neumann condition on the spatial domain, i.e. if ν denotes the outer normal to $\partial\Omega$, then*

$$\nabla u(t, s, \cdot) \cdot \nu = 0 \quad \text{on } \partial\Omega, \quad \text{and} \quad u(\cdot, 0, \cdot) = u(\cdot, T, \cdot) \quad \text{in } \mathbb{R}^+ \times \Omega.$$

Moreover, for $V_{\text{app}}^\sigma := (1, v_{\text{app}}^\sigma)^T / |(1, v_{\text{app}}^\sigma)|$ (cf. (2.10)) the spatio-temporal diffusion tensor $a_{(s,x)}$ is defined by

$$a_{(s,x)} := a_v V_{\text{app}}^\sigma \otimes V_{\text{app}}^\sigma + \begin{pmatrix} 0 & 0 \\ 0 & a(\Sigma) \end{pmatrix} \quad (2.18)$$

with the anisotropic spatial diffusion tensor $a(\Sigma)$ from the level-set evolution for static images (cf. Section 2.1) and a regularized apparent velocity v_{app}^σ . The temporal diffusion coefficient a_v depends on the apparent acceleration via the definition

$$a_v := G_\lambda(\text{accel}_{\text{app}}^\sigma) \quad \text{with} \quad G_\lambda(s) := \frac{1}{1 + s^2/\lambda}, \quad \lambda > 0,$$

where G_λ is the indicator function already defined in (2.1).

As before the variable t is the multi-scale parameter. The two different boundary conditions have the following meaning: In (BC1) we prescribe generally natural boundary conditions to the whole sequence, i.e. we have no flux across the spatial boundary of the

single frames and no flux at the beginning and the end of the sequence. It may be more convenient to impose natural boundary conditions in space and periodicity in sequence-time which is stated in (BC2), if this is consistent with the initial sequence u_0 .

Since the decomposition given in (2.18) is in general not orthogonal, there is an additional spatial diffusion part resulting from the velocity. Only in case the apparent velocity v_{app} vanishes, we have $V^\sigma = (1, 0)$ and the process performs a linear diffusion in direction of the sequence-time coordinate. For non-vanishing v_{app}^σ this is the real coupling of the space and sequence-time diffusion, as we will see below.

In Figure 2.15 we have applied the coupled anisotropic diffusion to an image-sequence similar to the one used in Figure 2.7. This time the level-sets are noisy squares, i.e. we obtain the noisy initial image from

$$\tilde{u}_0(x) = |x|_1$$

by adding random distortions in normal direction (cf. (2.7)):

$$u(s_i, x) := \left| x - d(s_i) + \text{rand}(x) \frac{\nabla \tilde{u}_0(x - d(s_i))}{|\nabla \tilde{u}_0(x - d(s_i))|} \right|_1, \quad i = 1, \dots, 40.$$

Each level-set is distorted in normal direction by a random factor $\text{rand}(x) \in [-12h, 12h]$, where h is the spatial grid-width for the computations, and the motion $d(s)$ models bouncing at some object as in Figure 2.15. Again the sequence consists of 40 frames each given on a 129×129 grid. Clearly the image-sequence is smoothed, while the edges of the quadratical level-sets are preserved. The anisotropic character becomes even more obvious when considering Figure 2.19, which is discussed in more detail in the next section. For a similar noisy image-sequence, we have integrated apparent trajectories through the sequence-time/space cube in Figure 2.14. This time the image-sequence consists of circular level-sets being disturbed in normal direction and again the motion reflects the circles' bouncing off a solid object. Clearly for the noisy initial image data the apparent trajectories do not reflect the motion at all. After several scale steps of the nonlinear coupled anisotropic diffusion we are able to integrate the trajectories from the start of the sequence to the end following a good approximation of the original motion. In fact, the coupled evolution can be seen as a sophisticated regularization variant for the extraction of motion velocities. In contrast to linear filters (as convolutions, heat-equation, projections, MCM) it preserves edges and corners of the level-sets in space and accelerations of the trajectories in sequence-time.

The application to a real (2+1)D image-sequence is shown in Figure 2.16. From the (3+1)D image-sequence, taken from ultrasound imaging of the human heart during a cardiac cycle, one slice through the 3D volume has been extracted and taken as an input for the anisotropic coupled diffusion. Unfortunately it is hard to comment on the quality of the diffusion process by looking at the static frames of the resulting sequence. Therefore, we identify the (2+1)D sequences with 3D space in the next section. This allows a better comparison of the coupled diffusion method with existing methods.

2.5 Comparison with existing models

Finally we would like to rank the diffusion models, we have defined so far, within the existing methodology of image processing. We thereby refer to the models that have been reviewed in the introduction. In particular we focus on the non-morphologic improved Perona-Malik model by Catté et al. [15], on the non-morphologic anisotropic Weickert model [80], on morphological mean curvature motion and for the processing of image-sequences on the clt-model of Mikula et al. presented in [58].

At the very beginning of the discussion, it must be emphasized that a fair comparison between existing image processing methodology is very hard and it might not be possible

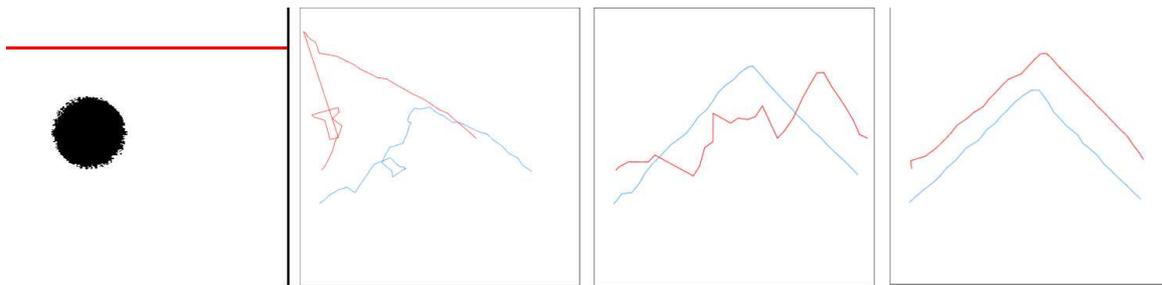


Figure 2.14: From an image-sequence showing noisy spherical level-sets (left) bouncing at some solid object (left, thick red line) we have integrated apparent trajectories (right, blue and red line). Obviously from the initial noisy image-sequence (middle left), we are not able to extract the underlying motion. For later scale-steps of the evolution (middle right: 5th scale, right: 9th scale), we are able to integrate a good approximation of the real trajectories.

at all. All models depend on many parameters to be provided by the user. Especially the choice of the edge indication function $G(s)$ (defined in (2.1)) and even more the stopping scale t , at which we evaluate the result of the models, are crucial parameters. Since the results are mostly compared visually, there is no objective control of errors. Furthermore, the results depend heavily on the discretization. Here we mention just a few parameters: explicit, semi-implicit or fully-implicit time-stepping, discretization with finite differences or finite elements, grid-width h , range of the image data in $[0, 256]$ or $[0, 1]$, stopping criterion of the iterative solver, etc. Moreover many of these parameters are not independent from the others.

In the following we try to compare the qualitative behavior of the present models. We choose most of the above parameters to be equal for each computation. Thus, the results reflect only a principle behavior of the models.

Static images

In Figure 2.17 a comparison of applications to the noisy octahedron data-set (cf. Figure 2.2) is shown. For the evolution always the result at the scale $t = 0.00375$ is shown. All models have been discretized using a semi-implicit time-stepping scheme and finite elements on hexahedral 65^3 grids. The built-in regularizations have been adjusted such that they correspond to a filter width $\sigma = 4h$. For the anisotropic diffusion we use local L^2 projections to regularize, whereas for the non-morphological models a short time step of size $\sigma^2/2$ of the heat-equation is used (cf. Section 3). A similar comparison for a real data-set on a 129^3 grid is shown in Figure 2.18.

As expected, the mean curvature motion has a good smoothing character, which however also smoothes out the edges and corners. Much more interesting is the behavior of the isotropic and the anisotropic Perona-Malik models. For the octahedron data-set the Perona-Malik model behaves similar to MCM although edges seem to be preserved better. The Weickert-model instead delivers a fairly good result: The edges are kept quite well, however straight edges become curved and corners are rounded very much. The disadvantage of the Perona-Malik model as a non-morphological model becomes clear in Figure 2.18. Due to the presence of high gradients in the noisy ultrasound data, the model tries to keep much of that noise, which is removed by the tangential smoothing of the anisotropic Weickert model. In both applications however the anisotropic geometric evolution visually gives the best result. It preserves edges very well, and in the case of the sample-data it is visually closest to the original non-noisy input.

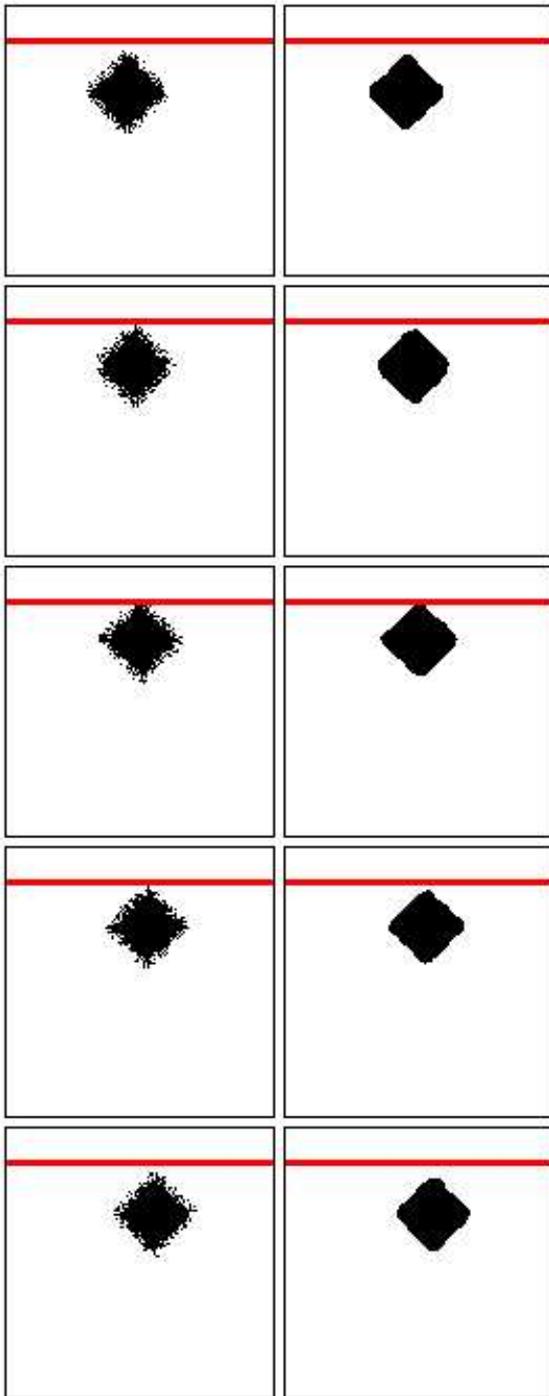


Figure 2.15: The application to a sample $(2+1)D$ image-sequence is illustrated. The input data is given by $\tilde{u}_0(s, x) := |x - d(s)|_1$, where $d(s)$ is such that the motion reflects bouncing at the red line. The level-sets of \tilde{u}_0 have been disturbed in normal direction. From top to bottom the extraction of the same sub-level-set from several frames of the sequence is shown. The left column shows the initial data; the third scale step is shown in the right column for successive frames.

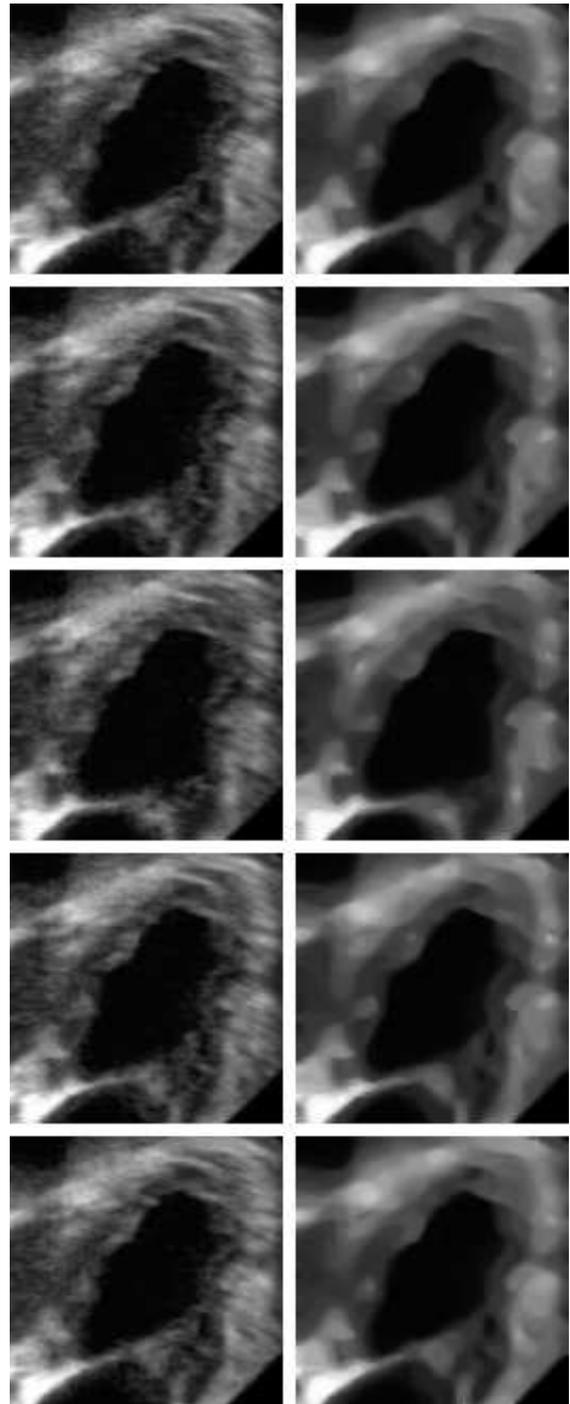


Figure 2.16: From the $(3+1)D$ ultrasound image-sequence shown in figure 2.12 one slice has been extracted from the 3D volume and taken as $(2+1)D$ image-sequence. From top to bottom successive frames of the sequence are displayed. The left column shows the noisy image-sequence and the right column the result after the third scale step.

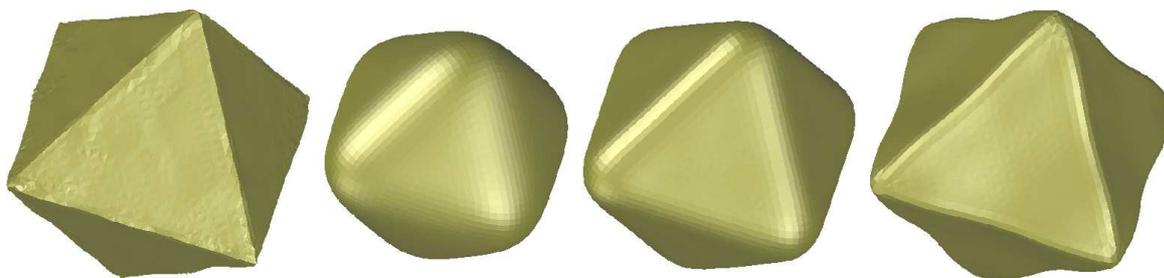


Figure 2.17: Comparison of the principle behavior of different image processing models on the noisy octahedron data-set shown in Figure 2.2. From left to right, the result of the anisotropic geometric diffusion, mean curvature motion, the isotropic improved Perona-Malik model [15], and the anisotropic Weickert diffusion [80] are displayed. Always the same iso-surface has been extracted from the data.

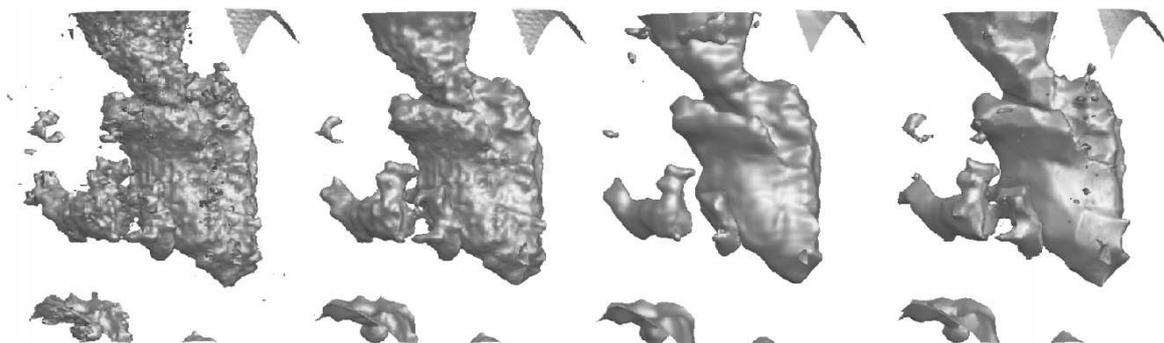


Figure 2.18: One frame of the noisy 3D echo-cardiographical data-set (left) shown in Figure 2.3 is taken as input for different image processing methodology. From middle left to right the results from the application of the isotropic improved Perona-Malik model [15], the anisotropic Weickert diffusion [80], and the anisotropic geometric diffusion are displayed. Always the extraction of the same iso-surface from the data-sets is shown.

Image-sequences

As already noticed in the last section, a judgment on the coupled spatio-temporal diffusion is difficult from static images of the sequences. A comparison between different models becomes even harder. Therefore we are restricting our comparison to (2+1)D image-sequences and identify the sequence-time direction with a third space-dimension. For a comparison we then draw iso-surfaces in the 3D space, which show the movement of single level-sets within the sequence. In Figure 2.19 such iso-surfaces of the noisy image-sequence already shown in Figure 2.7 are shown. The image-sequence now consists of 61 frames

$$\tilde{u}_0(s_i, x) := |x - d(s_i)|, \quad \text{for } i = 0, \dots, 60, \quad \text{and } 0 = s_0 \leq \dots \leq s_{60} = 1,$$

where $d(\cdot)$ models the bouncing of the image-sequence at some solid object as for the data shown in Figure 2.7. We add a small amount of noise in normal direction to the original image-sequence (cf. Figure 2.15). The application of a diffusion model then delivers a scale of image-sequences $u : \mathbb{R}^+ \times [0, 1] \times [0, 1]^2 \rightarrow \mathbb{R}$, $(t, s, x) \mapsto u(t, s, x)$ as the solution of the parabolic PDE. From this scale of image-sequences, we generate a scale of 3D images $U : \mathbb{R}^+ \times [0, 1]^3 \rightarrow \mathbb{R}$ by

$$U(t, x) := u(t, x_1, y), \quad \text{for } x_1 = 0, \dots, 60 \quad \text{and } x = (x_1, x_2, x_3), y = (x_2, x_3).$$

If we fix a scale t_0 , the 3D image $U(t_0, \cdot)$ then shows the motion of the sequence smoothed at scale t_0 . The frames of the smoothed sequence correspond to slices through the 3D image.

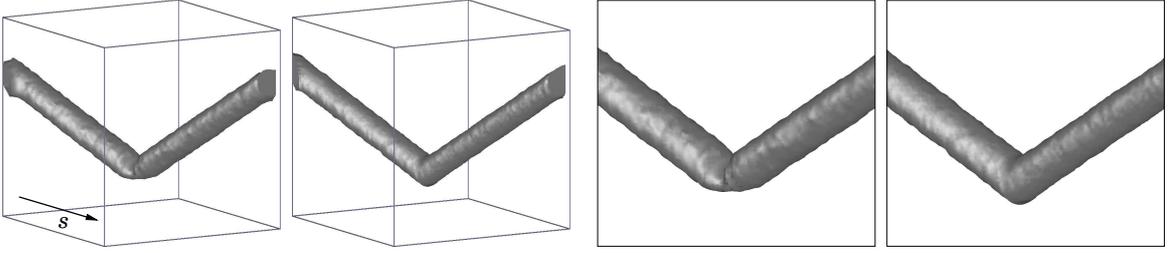


Figure 2.19: We compare existing image processing methodology applied to a noisy image-sequence, which shows noisy spherical level-sets bouncing at some solid object (cf. Figures 2.7 and 2.15). On the left one iso-surface from the 3D representation of the image-sequence smoothed at scale 3 is displayed. The leftmost image shows the result of the model (2.19) which smooths out the highly accelerated motion of the level-sets. In contrast to that, the anisotropic geometric model (middle left) preserves this behavior quite well. The right images show magnified sections from the iso-surface representations (middle right: model (2.19), right: anisotropic geometric model).

In Figure 2.19 we compare the different behavior of the new space-time anisotropic diffusion model from the last section with the model presented by Mikula et al. in [71]:

$$\partial_t u - \text{clt}(u^\sigma) \text{div}(G(|\nabla u^\sigma|) \nabla u) = 0. \quad (2.19)$$

which steers the diffusion speed by the apparent acceleration $\text{accel}_{\text{app}}$, here written as the curvature of the trajectories clt (cf. Section 1.4). The above non-morphological model (2.19) does not have an anisotropic character and in contrast to our anisotropic geometric model it increases the diffusion if the apparent acceleration is high. This is the main reason for the different behavior of the two models in case of highly accelerated motions. The numerical example shown in Figure 2.19 shows that in case of the model (2.19) the high acceleration of the trajectories is not preserved well during the evolution. In particular the temporal corner, where the motion changes its direction, i.e. the point where the balls bounce at the solid object, is rounded during the evolution, whereas it is kept sharp for the anisotropic geometric model. The magnified sections shown in Figure 2.19 underline even more how the acceleration increases the diffusion speed at the motion-corner.

Chapter 3

Regularized geometric quantities

IN THE EXPOSITION of the last chapter, we have intensively made use of a regularized shape operator $S_{T_x\mathcal{M}}^\sigma$, an extended regularized shape operator Σ , a shape operator on regularized data $S^\sigma = DN^\sigma$, regularized normals N^σ , a regularized apparent velocity v_{app}^σ and moreover a regularized apparent acceleration $\text{accel}_{\text{app}}^\sigma$, which are part of the definition of the diffusion tensors $a(\Sigma)$ and $a_{(s,x)} = a_{(s,x)}(\Sigma, v_{\text{app}}^\sigma, \text{accel}_{\text{app}}^\sigma)$. These regularizations are discussed in this chapter. They are necessary for various reasons:

- Robustness of the edge, velocity and acceleration detection with respect to noisy image and image-sequence data is needed;
- Omitting the regularization would result in a trivial evolution $u(t, \cdot) = u_0(\cdot)$ for the static image smoothing model;
- The geometric quantities $S_{T_x\mathcal{M}}$, N and v_{app} become singular for a vanishing image gradient $|\nabla u|$.

In terms of the robustness of the resulting models, the regularization process is equivalent to a classification of the noise. Naturally, the driving force of an evolution process should depend on such a classification. In particular for the static-image processing model presented in Chapter 2 we have shown that the force depends on the difference between regularized noisy and real data (Theorem 2.2). In general there are different methods for suitable regularizations or classifications at hand. As we have seen in the last Chapter (cf. Figure 2.14), it is crucial to have a regularization simultaneous in sequence-time and space for image-sequence processing. Only such a coupled regularization ensures the robustness of the model with respect to sequence-time dependent quantities, such as v_{app} and $\text{accel}_{\text{app}}$. Although the discretization of the regularized quantities with linear finite elements is later the topic in Chapter 5, we already emphasize at this point that the computation of all the geometric quantities $S_{T_x\mathcal{M}}$, v_{app} , and $\text{accel}_{\text{app}}$ involves higher order derivatives of image functions, which are typically given as piecewise constant or piecewise bi- or trilinear functions. Therefore it is useful to combine the regularization aspect with the consistent definition of higher order derivatives on data having low regularity.

First we discuss various regularization methods themselves, before going into the details of the regularizations of the geometric quantities. The regularization methods under consideration are divided into global and local variants. The different behavior of suitable regularizations is shown for test-data as well as for real data-sets. A section on optimal regularization parameters concludes the chapter. Although the regularization of spatial data is presented first, corresponding remarks comment on the easy augmentation to the regularization of spatio-temporal data.

3.1 Global regularizations

A very first attempt to regularize the geometric quantities $S_{T_x\mathcal{M}}$, v_{app} , $\text{accel}_{\text{app}}$ is the component-wise smoothing of these quantities with a standard linear filter such as the heat-equation. Regardless of the question how to compute the initial non-regularized quantities, one might encounter inconsistencies with this approach: For example it is no longer guaranteed that $S_{T_x\mathcal{M}}^\sigma$ diagonalizes with respect to $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$ if N is regularized separately from $S_{T_x\mathcal{M}}$. As a consequence, we do not consider such component-wise regularizations in the following. Instead we always base the regularization method on a regularization of the image u and derive Σ , v_{app}^σ , and $\text{accel}_{\text{app}}^\sigma$ from this regularized image data.

(R1) Morphologic regularizations

For a morphologic model, it is desirable to consider regularizations, which are also invariant under monotone transformations of the gray-value. Consequently the morphologic character of the whole model would not be altered. The simplest morphological filter possible is the mean curvature flow (cf. Section 1.3): For a given image $u : \Omega \rightarrow \mathbb{R}$ (or an image-sequence $u : Q \rightarrow \mathbb{R}$) one considers the evolution

$$\partial_t w - |\nabla w| \operatorname{div} \left(\frac{\nabla w}{|\nabla w|} \right) = 0$$

with initial data $w(0, \cdot) = u$ and natural boundary conditions. For a fixed stopping time $t := \sigma$ the definition $u^\sigma(\cdot) := w(\sigma, \cdot)$ defines the regularized data, which can be taken for the computation of geometric quantities. However, the definition of the higher order derivatives needed for the computations is still a problem. For example one could use difference quotients, which have a poor approximation order though. For our model, we therefore consider one of the regularizations discussed below, which combine the aspects of regularization and definition of the derivatives.

(R2) Convolution with Gaussian kernels

Closely related to the first attempt at the beginning of this section is the most prominent way of regularizing image data in the image processing community. We consider the convolution of the image u with a smoothing kernel K^σ . If we choose a Gaussian kernel K^σ having variance σ , the convolution $u^\sigma := K^\sigma * u$ coincides with the evolution of u under the heat equation

$$\partial_t w - \Delta w = 0 \tag{3.1}$$

with initial data $w(0, \cdot) = u$ and natural boundary conditions, evaluated at a time $t = \sigma^2/2$. Thus, $u^\sigma(\cdot) = w(\sigma^2/2, \cdot)$. Still this approach does not provide a definition of the second derivative directly. But if for a multi-index γ we compute the convolution of the image data with the derivatives $D^\gamma K^\sigma$ of the kernel, the convolution property

$$D^\gamma(K^\sigma * u) = (D^\gamma K^\sigma) * u = K^\sigma * (D^\gamma u) \tag{3.2}$$

would provide us with approximations of the desired derivatives of the image function u . Unfortunately, due to the non compact support of K^σ , a good approximation of the convolution $D^\gamma K^\sigma * u$ is computationally expensive and an efficient approximation of (3.1) is possible only with multi-grid-methods [38].

(R3) Convolution with derivatives of C_0^∞ -kernels

The convolution property (3.2) is now used to define regularizations u^σ and the naturally assigned spatial as well as temporal derivatives $D^\gamma u^\sigma$ via

$$u^\sigma := K^\sigma * u \quad \text{and} \quad D^\gamma u^\sigma := (D^\gamma K^\sigma) * u.$$

To allow an efficient implementation, we use kernels with compact support in the implementation and then base the convolution on discrete summations with appropriate weights. For a spatial regularization, a suitable C_0^∞ -kernel which has already successfully been used in [59] is given by

$$K^\sigma(x) := \begin{cases} \frac{1}{M} \exp\left(\frac{|x|^2}{|x|^2 - \sigma^2}\right) & \text{if } |x|^2 \leq \sigma^2, \\ 0 & \text{else.} \end{cases} \quad (3.3)$$

For a spatio temporal regularization, we use the following compactly supported kernel

$$K^\sigma(s, x) := \begin{cases} \frac{1}{M} \exp\left(\frac{s^2 + |x|^2}{s^2 + |x|^2 - \sigma^2}\right) & \text{if } |x|^2 + s^2 \leq \sigma^2, \\ 0 & \text{else.} \end{cases} \quad (3.4)$$

The constant M is chosen such that $\int K^\sigma = 1$. For this type of regularization, we set $\mathcal{B}_\sigma(x) := \text{supp } K^\sigma(x)$ or $\mathcal{B}_\sigma(s, x) := \text{supp } K^\sigma(s, x)$, respectively. Based on the regularized data u^σ the geometric quantities can be computed.

3.2 Local regularizations

So far we have presented global regularization variants. Since the latter variant is based on compactly supported kernels we can consider it as a local process as well. In the following, we are going to present another regularization variant which combines the regularization with a consistent definition of derivatives. The method is based on local projections of the image data onto polynomial spaces. It turns out to have a more robust regularization effect than the convolution with derivatives of smoothing kernels.

(R4) Local L^2 -projections

Let us study a purely spatial local regularization first. We consider a finite dimensional space $Q^* \subset C^2(\Omega)$ of higher regularity than the given image u . Later we choose $Q^* = \mathcal{P}_l(\mathbb{R}^d)$ to be the space of polynomials of degree $l \geq 2$; other projection spaces might be useful in certain applications. To regularize the image data, we compute a least squares fit (a local L^2 -projection) of the data onto the space Q^* . To this end let us fix a point $x \in \Omega$ contained in the level-set \mathcal{M} and a neighborhood \mathcal{B}_σ of x and denote a basis of Q^* by

$$Q^* = \text{span}\{q_i(y), i = 1, \dots, \dim Q^*\}.$$

By definition the L^2 -projection $u_x^\sigma(y)$ of the image u onto Q^* is defined via the orthogonality relation

$$\int_{\mathcal{B}_\sigma(x)} (u_x^\sigma - u) q \, dy = 0 \quad \forall q \in Q^*. \quad (3.5)$$

Locally we obtain the iso-surface of u_x^σ in x , whose tangent space in general does not coincide with the the tangent space $T_x \mathcal{M}$ of the iso-surface of u in x . But as before we base the computations of the geometric quantities on the regularized image data.

For the spatio-temporal regularization we proceed in a similar manner. Let $Q^* \subset \mathcal{P}_l(Q)$ and a basis of Q^* shall be given by $\text{span}\{q_i(r, y)\}$. Then the projection $u_{(s,x)}^\sigma$ in sequence-time and space fulfills the orthogonality relation

$$\int_{\mathcal{B}_\sigma(s,x)} \left(u_{(s,x)}^\sigma - u \right) q \, dr \, dy = 0 \quad \forall q \in Q^*, \quad (3.6)$$

for a neighborhood $\mathcal{B}_\sigma(s, x)$ of a space-time point $(s, x) \in Q$. Although this approach delivers very robust local regularizations of image-sequences, it turns out to be computationally too expensive in applications, particularly in the spatio-temporal applications (cf. Section 5).

3.3 Regularized Shape Operator

Now we discuss the application of the different regularization approaches to the quantities defining our diffusion tensors. Since the regularization variants (R1) and (R2) do not deliver a definition of higher order derivatives on the image data, we do not consider them any longer. Instead we apply the variants (R3) and (R4) to the definition of a regularized shape operator and moreover examine the consistency of these definitions.

So far, we have not yet treated the singularities which appeared in the definition of the geometric quantities N , $S_{T_x\mathcal{M}}$, and v_{app} . In general, we cannot assume $\nabla u \neq 0$ even if the initial data fulfills the regularity assumption $\nabla u \neq 0$. Evans and Spruck [32] gave an example of the level-set approach of mean curvature motion, where a certain level-set degenerates and fattens, such that it does not describe curves or surfaces any longer.

In case the image gradient ∇u vanishes, the definition of a normal does not make sense. Therefore also the definition of a normal velocity $(v_{\text{app}})_n$ is not meaningful any more. Hence, the definition of the shape operator $S_{T_x\mathcal{M}}$ fails as well. Nevertheless we have to define a suitable diffusion tensor $a(\Sigma)$ in that case. For discretization issues (in Chapter 5) we follow the proposal of Evans and Spruck [32] who replaced the Euclidean norm $|\cdot|$ with an approximation $|\cdot|_\varepsilon$, where for small $\varepsilon \ll 1$

$$|\cdot|_\varepsilon := \sqrt{|\cdot|^2 + \varepsilon^2}.$$

Here, we define the diffusion tensor $a(\Sigma) = \text{Id}$ in case $|\nabla u| = 0$. This reduces the anisotropic diffusion to an isotropic mean curvature motion, which was handled successfully by Evans and Spruck with the above regularization $|\cdot|_\varepsilon$.

Let us now give the definitions of the regularized shape operator on a level-set going through a point $x_0 \in \Omega$ assuming that we have excluded the latter case $\nabla u = 0$. Since we base the computations on regularized images rather than component-wise regularized geometric quantities, we ensure that always

$$S_{T_x\mathcal{M}^\sigma}^\sigma : T_x\mathcal{M}^\sigma \rightarrow T_x\mathcal{M}^\sigma \quad \text{with} \quad T_x\mathcal{M}^\sigma \perp \text{span}\{N^\sigma\}.$$

Finally as defined in (2.2), we get the extended regularized shape operator Σ from

$$\Sigma := \begin{pmatrix} S_{T_x\mathcal{M}^\sigma}^\sigma & 0 \\ 0 & 0 \end{pmatrix}$$

in coordinates $\{w^{1,\sigma}, w^{2,\sigma}, N^\sigma\}$ of the regularized principal curvatures $w^{1,\sigma}, w^{2,\sigma}$ and the regularized normal $N^\sigma \perp w^{1,\sigma}, w^{2,\sigma}$.

Variant (R3): Convolution with derivatives of C_0^∞ kernels

The regularization u^σ is given by convolution of u with the kernel K^σ and the derivatives $D^\gamma u^\sigma$ are given by convolution with $D^\gamma K^\sigma$. Thus, we have the definitions

$$\begin{aligned} N^\sigma(x_0) &= \frac{\nabla K^\sigma * u}{|\nabla K^\sigma * u|}(x_0), \\ S_{T_x}^\sigma \mathcal{M}^\sigma(x_0) &= \frac{1}{|\nabla K^\sigma * u|} \left((\text{Id} - N^\sigma \otimes N^\sigma) (D^2 K^\sigma * u) (\text{Id} - N^\sigma \otimes N^\sigma) \right)(x_0) \end{aligned} \quad (3.7)$$

and in particular we have $\Sigma = DN^\sigma (\text{Id} - N^\sigma \otimes N^\sigma) = S^\sigma (\text{Id} - N^\sigma \otimes N^\sigma)$, because

$$\begin{aligned} \partial_i (N^\sigma)_j &= \partial_i \frac{\partial_j K^\sigma * u}{|\nabla K^\sigma * u|} = \frac{\partial_i \partial_j K^\sigma * u |\nabla K^\sigma * u| - \partial_j K^\sigma * u \partial_i |\nabla K^\sigma * u|}{|\nabla K^\sigma * u|^2} \\ &= \frac{1}{|\nabla K^\sigma * u|} \left(\partial_i \partial_j K^\sigma * u - \frac{\partial_j K^\sigma * u \sum_{l=1}^d (\partial_l K^\sigma * u \partial_i \partial_l K^\sigma * u)}{|\nabla K^\sigma * u|^2} \right) \\ &= \frac{1}{|\nabla K^\sigma * u|} (\text{Id} - N^\sigma \otimes N^\sigma) D^2 K^\sigma * u. \end{aligned}$$

Variant (R4): Local L^2 -projections

In the local projection approach on $\mathcal{B}_\sigma(x)$ the local regularization $u_x^\sigma(y)$ is given by the solution of the linear system (3.5). The normal and the shape operator are then defined via

$$\begin{aligned} N^\sigma(x_0) &= \frac{\nabla_y u_x^\sigma(y)}{|\nabla_y u_x^\sigma(y)|} \Big|_{y=x_0}, \\ S_{T_x}^\sigma \mathcal{M}^\sigma(x_0) &= \frac{1}{|\nabla_y u_x^\sigma(y)|} \left((\text{Id} - N^\sigma \otimes N^\sigma) D_y^2 u_x^\sigma(y) (\text{Id} - N^\sigma \otimes N^\sigma) \right) \Big|_{y=x_0}, \end{aligned} \quad (3.8)$$

and since we evaluate the projection and consequently the normal and the shape operator separately for each $x_0 \in \Omega$, we have in general $\Sigma \neq D_x N^\sigma (\text{Id} - N^\sigma \otimes N^\sigma) = S^\sigma (\text{Id} - N^\sigma \otimes N^\sigma)$. But there is one special configuration for which we have the identity $\Sigma = S^\sigma (\text{Id} - N^\sigma \otimes N^\sigma)$: Consider an image which has the form

$$u(x) = \sum_{i \leq \dim Q^*} u_i q_i(x) \in Q^*,$$

thus being an element of our projection space Q^* . Then, obviously, the projection onto Q^* is the identity, i.e. we have $u^\sigma = u$ globally. Then u has enough regularity to define N^σ and $S_{T_x}^\sigma \mathcal{M}^\sigma$ as in (1.1) and (1.3), and clearly $N^\sigma = N$ and $\Sigma = S^\sigma (\text{Id} - N \otimes N) = S(\text{Id} - N \otimes N)$ hold. Indeed

$$u_x^\sigma(y) = u(x) \quad \forall x \in \Omega \quad \forall y \in \Omega$$

and thus for any multi-index γ

$$D_y^\gamma u_x^\sigma(y) = D_x^\gamma u(x).$$

We have seen in Chapter 2 that the diffusion is steered by the difference between the regularized quantities and their non-regularized counterparts, according to Theorem 2.2. Since for the case $u \in Q^*$ these differences vanish, all the elements of Q^* remain invariant under the evolution. For example, if $Q^* = \mathcal{P}_2(\mathbb{R}^d)$ the invariant images are quadratic functions, and their level-sets are ellipsoids.

For a 2D data-set of size 257×257 , whose level-sets are ellipsoids, the evaluation of curvature is shown in Figure 3.1. The regularization variants (R3) and (R4) are compared for different sizes of σ (cf. Chapter 5). In the next step the latter 2D data-set is disturbed using random noise, which is added to the nodal values. We note that in the presence

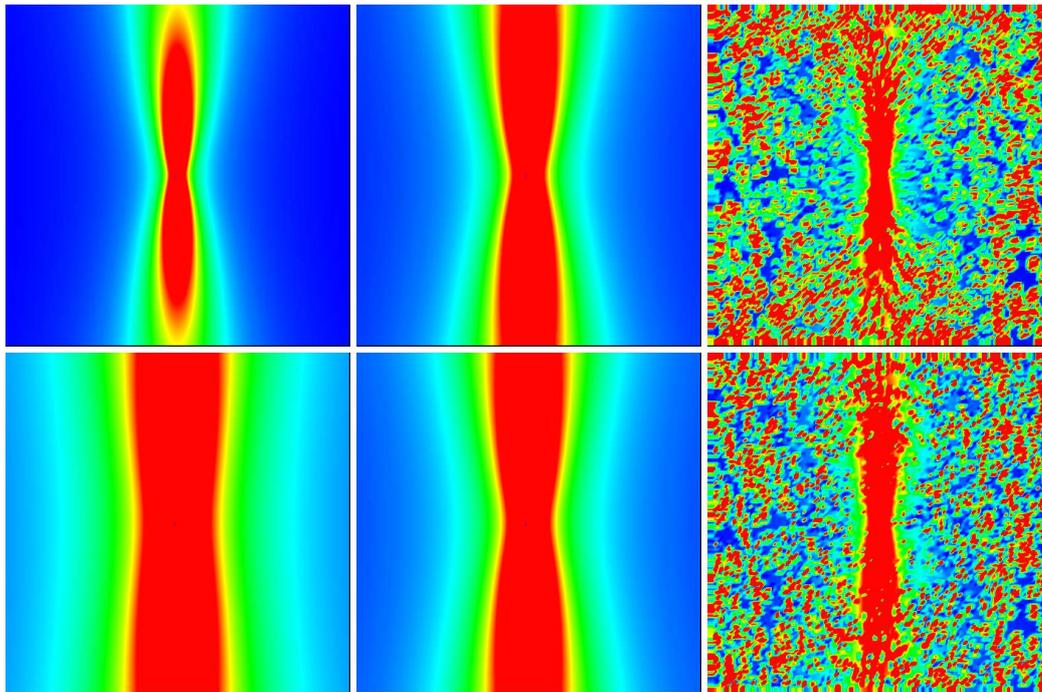


Figure 3.1: *Curvature evaluation on a 2D ellipsoid data-set, i.e. $u(x) = x_1^2 + x_2^2/b$, $b > 1$. Top row: Evaluation via convolutions. Bottom row: Evaluation via projections. Left column: Smooth data-set $l = 2$. Middle column: Smooth data-set $l = 4$. Right column: Noisy data-set $l = 2$. A color ramp from blue to green to red indicates the curvature of the level-sets.*

of noise the local projections behave more robust, i.e. the curvature evaluated using local projections is visually closer to the curvature of a smooth image. This behavior becomes even more obvious when comparing the 3D evaluations in Figure 3.3. In Figure 3.1 the same comparison has been performed for a data-set whose level-sets are squares. Again the projection approach behaves more robust in the presence of noise.

In Figures 3.3 and 3.4 we have compared the regularization variants (R3) and (R4) on 3D data-sets. Again we see that the L^2 -projection behaves more robust in case of noisy data. Especially for the octahedron data-set, the edges and corners are detected much better by the regularization via local projections. The evaluation of curvature on a 3D medical data-set is illustrated in Figure 3.5, where the curvatures of the gray matter of the human brain are shown.

In the remaining paragraphs we examine the consistency of the approximations we make with the regularization process. In particular it is of interest, how good we can approximate the curvature of a given level-set with the two different approaches. We consider the spherical level-sets of the image $u(x) := |x - r|^2$, $r = (R, \dots, R) \in \mathbb{R}^d$ and define an error

$$e_i := \left\| \kappa^{\sigma,i} - 1/R \right\|_{L^2(\Omega_\sigma)}$$

which measures the difference between the curvature $\kappa^{\sigma,i}$ of the projection and the real curvature $1/R$ in the L^2 -norm on $\Omega_\sigma \subset \Omega = [0, 1]^d$. To avoid interference of the convolution or the projection with the boundary of Ω we have chosen Ω_σ such that $\text{dist}(\Omega_\sigma, \Omega) > 1/4$. This is the maximum subset such that the convolution or projection stencil is always completely contained in Ω (cf. Figure 3.6). In Tables 3.1 and 3.2 we have collected the error values for different grid sizes and stencil widths for both regularization variants (R3) and (R4). The computation is performed with the parameter $R = 4$, different grid-widths h and increasing size of $\sigma \sim \text{diam } \mathcal{B}_\sigma(x)$ which corresponds to an increasing size of the projection stencil $\mathcal{B}_\sigma(x)$. Later in Chapter 5 we will use balls in the $|\cdot|_\infty$ norm to define \mathcal{B}_σ such

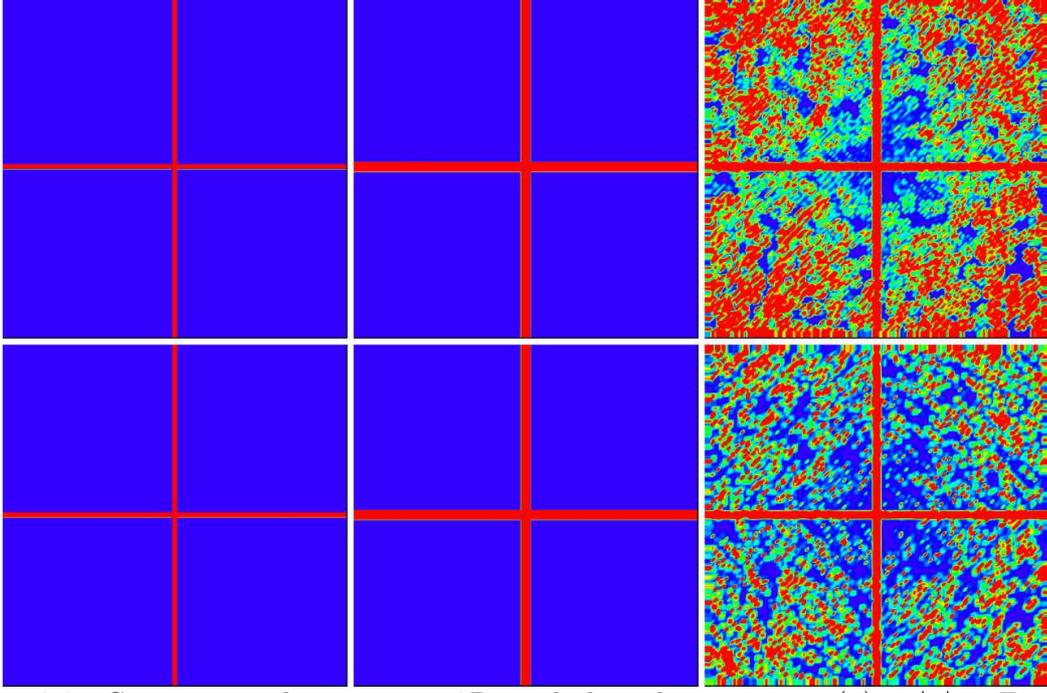


Figure 3.2: *Curvature evaluation on a 2D octahedron data-set, i.e. $u(x) = |x|_1$. Top row: Evaluation via convolutions. Bottom row: Evaluation via projections. Left column: Smooth data-set $l = 2$. Middle column: Smooth data-set $l = 4$. Right column: Noisy data-set $l = 2$. Again a color ramp from blue to green to red indicates the curvature of level-sets.*

that $\sigma = lh = \text{diam}\mathcal{B}_\sigma$. From the table we see that the approximation of the curvatures is second order: An enlargement of the projection stencil by a factor 2 decreases the error by a factor $1/4$. As expected the error is independent of the grid-width h . In Section 5.4 we deal with the choice of optimal regularization parameters.

3.4 Regularized apparent velocity and acceleration

So far we have discussed the regularization of the normals and the shape operators, but for the processing of image-sequences we also need a regularized apparent velocity and acceleration. In general, the definitions of the regularized shape operator Σ and the normal N^σ are done in exactly the same way as in the last section, where of course the corresponding spatial part from the regularizations is used. Earlier we have already seen that the definition of an apparent normal velocity $(v_{\text{app}}^\sigma)_n$ makes no sense if we do not have the definition of a normal ($\nabla u = 0$). Therefore we exclude this case in the following. Furthermore we had replaced the inverse of the shape operator by the pseudo inverse $S_{T_x\mathcal{M}}^\dagger$, consequently we replace the inverse of the regularized shape operator by its pseudo inverse Σ^\dagger .

Variant (R3): Convolution with derivatives of C_0^∞ kernels

We are going to treat the sequence-time derivatives we need as a convolution with the derivative $\partial_s K^\sigma$ of the smoothing kernel $K^\sigma(s, x)$ defined in (3.4). Consequently we can define

$$\begin{aligned} v_{\text{app}}^\sigma(s_0, x_0) &= - \left(\frac{\partial_s K^\sigma * u}{|\nabla K^\sigma * u|} N^\sigma + \Sigma^\dagger(\partial_s K^\sigma * N^\sigma) \right) (s_0, x_0) \\ \text{accel}_{\text{app}}^\sigma(s_0, x_0) &= (\partial_s K^\sigma * v_{\text{app}}^\sigma + (\nabla K^\sigma * v_{\text{app}})v_{\text{app}}) (s_0, x_0) \end{aligned} \quad (3.9)$$

Variant (R4): Local L^2 -projections

The local projections delivered a consistent approximation of the shape operator $S_{T_x\mathcal{M}}$ on

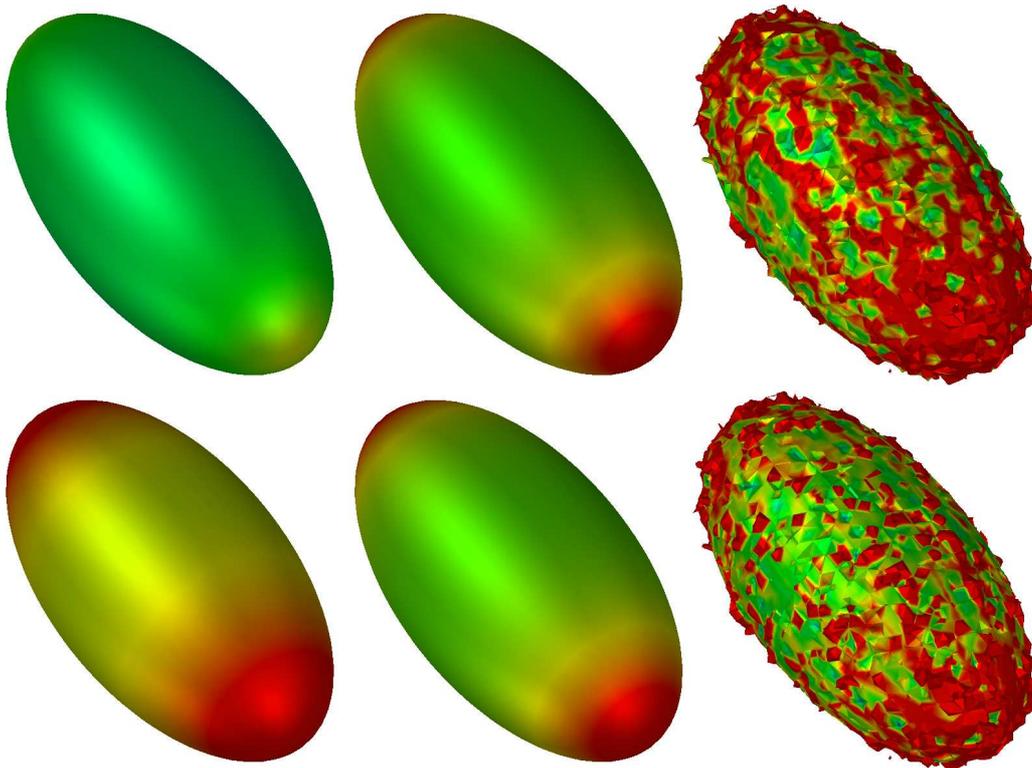


Figure 3.3: *Curvature evaluation on a 3D ellipsoid data-set, i.e. $u(x) = x_1^2 + (x_2^2 + x_3^2)/b, b > 1$. Top: Evaluation via convolutions. Bottom: Evaluation via projections. Left column: Smooth data-set $l = 2$. Middle column: Smooth data-set $l = 4$. Right column: noisy data-set $l = 2$. A color ramp from blue to red indicates the dominant curvature of the level-sets.*

data of low regularity. Thus, we would like to carry on this good approximation character to the computation of the regularized apparent velocity v_{app}^σ and the regularized apparent acceleration $\text{accel}_{\text{app}}^\sigma$. Unfortunately, we have to enlarge the projection space for this purpose, because even more regularity is required. For the computation of Σ it has been sufficient to have $Q^* \subset \mathcal{P}_2(\Omega)$ whereas for the additional sequence-time quantities we need tensor product polynomials in sequence-time and space having the appropriate regularity. In particular we need the following spaces:

$$\begin{aligned} v_{\text{app}}^\sigma(s_0, x_0) &: u^\sigma(y) \in \mathcal{P}_1(I) \otimes \mathcal{P}_2(\Omega), \\ \text{accel}_{\text{app}}^\sigma(s_0, x_0) &: u^\sigma(y) \in \mathcal{P}_2(I) \otimes \mathcal{P}_3(\Omega). \end{aligned} \quad (3.10)$$

Assume we have computed the local projection $u_{(s,x)}^\sigma(r, y) \in \mathcal{P}_1(I) \otimes \mathcal{P}_2(\Omega)$ as the solution of a linear system. We then define

$$v_{\text{app}}^\sigma(s_0, x_0) = - \left(\frac{\partial_r u_{(s,x)}^\sigma(r, y)}{|\nabla_y u_{(s,x)}^\sigma(r, y)|} N^\sigma + \Sigma^\dagger(r, y) (\partial_r N^\sigma)(r, y) \right) \Big|_{r=s_0, y=x_0}. \quad (3.11)$$

For the computation of the regularized apparent acceleration $\text{accel}_{\text{app}}^\sigma$ a closed formula in terms of the derivatives of $u_{(s,x)}^\sigma(r, y) \in \mathcal{P}_2(I) \otimes \mathcal{P}_3(\Omega)$ can be used. Alternatively, a separate projection of v_{app}^σ would lead to a different definition of $\text{accel}_{\text{app}}^\sigma$. But as we see in Chapter 5 the enlarged projection spaces make the regularization variant impracticable, since they involve huge amounts of numerical integrations. Therefore we are not going into the details of the definition of $\text{accel}_{\text{app}}^\sigma$ for the projection regularization approach. In applications we will always use variant (R3).

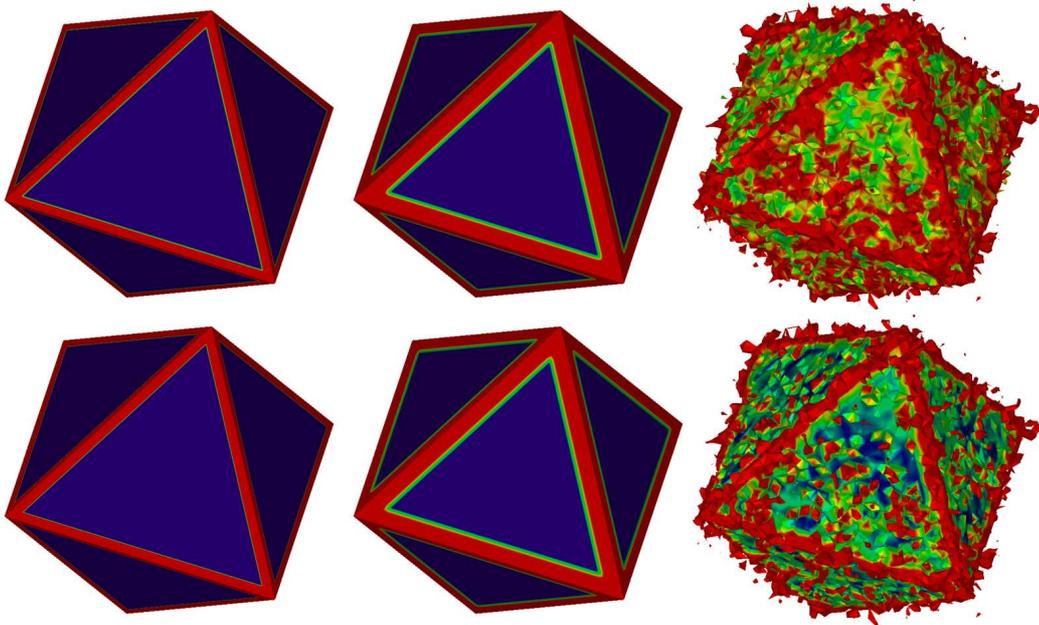


Figure 3.4: *Curvature evaluation on a 3D octahedron data-set, i.e. $u(x) = |x|_1$. Top: Evaluation via convolutions. Bottom: Evaluation via projections. Left column: Smooth data-set $l = 2$. Middle column: Smooth data-set $l = 4$. Right column: noisy data-set $l = 2$. A color ramp from blue to red indicates the dominant curvature of the level-sets.*

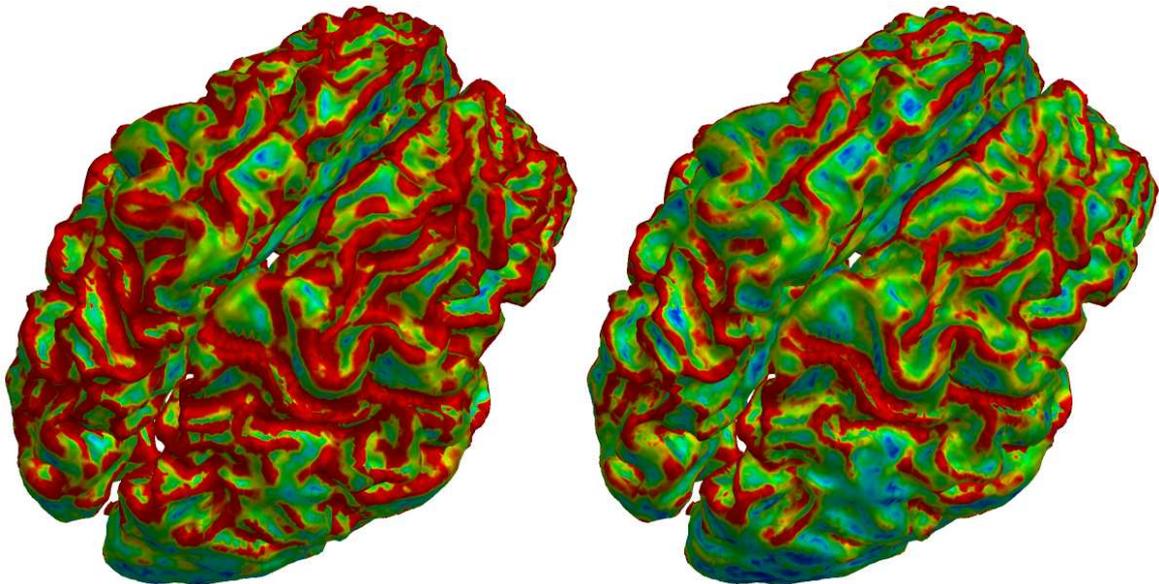


Figure 3.5: *The computations of curvature on a segmented 3D medical data-set of the gray matter of the human brain are shown. On the left the evaluation with convolutions is displayed, whereas on the right the result of the projection method is shown. Again a color ramp from blue to green to red codes the dominant curvature of the level-sets. The data-set was of size 129^3 .*

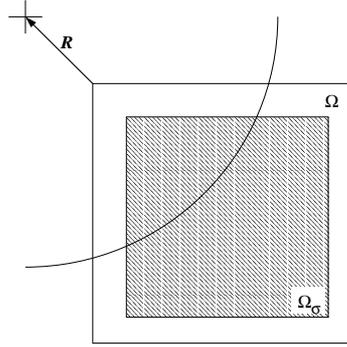


Figure 3.6: Analysis of the consistency of the regularizations for the evaluation of curvatures on spherical level sets.

h	$l=\sigma/h$	Projection e	Convolution e
1/32	2	5.998e-02	9.327e-02
	4	1.559e-02	1.627e-02
	8	3.934e-03	5.974e-03
	16	9.858e-04	2.683e-04
1/64	2	5.998e-02	9.309e-02
	4	1.559e-02	1.619e-02
	8	3.934e-03	5.962e-03
	16	9.857e-04	2.677e-04
	32	2.466e-04	2.414e-05

h	$l=\sigma/h$	Projection e	Convolution e
1/128	2	5.998e-02	9.301e-02
	4	1.559e-02	1.617e-02
	8	3.934e-03	5.956e-03
	16	9.857e-04	2.675e-04
	32	2.466e-04	2.412e-05
	64	6.165e-05	3.285e-07

Table 3.1: Consistency errors for the evaluation of curvature on regularized 2D data for different sizes of h and increasing stencil width l of the L^2 -projection and the convolution with derivatives of smoothing kernels, respectively. We see that the consistency error is rather independent of the grid-size, i.e. decreasing h does not decrease the error if l is kept fixed. If the value of l is increased by a factor 2 we see that the error is decreased by approximately a factor 1/4.

h	$l=\sigma/h$	Projection $e_1 = e_2$	Convolution $e_1 = e_2$
1/32	2	5.451e-02	4.987e-02
	4	1.398e-02	1.723e-03
	8	3.517e-03	
	16	8.806e-04	
1/64	2	5.479e-02	5.016e-02
	4	1.405e-02	1.460e-03
	8	3.535e-03	
	16	8.852e-04	
	32	2.214e-04	

h	$l=\sigma/h$	Projection $e_1 = e_2$	Convolution $e_1 = e_2$
1/128	2	5.493e-02	4.974e-02
	4	1.409e-02	1.718e-03
	8	3.544e-03	
	16	8.875e-04	
	32	2.220e-04	
	64	5.549e-05	

Table 3.2: Consistency errors for the evaluation of curvature on regularized 3D data for different sizes of h and increasing stencil width l of the L^2 -projection and the convolution with derivatives of smoothing kernels, respectively. If the value of l is increased by a factor 2 we see that the error is decreased by approximately a factor 1/4. Moreover for fixed l the error is rather independent of h .

Chapter 4

Existence of viscosity-solutions

IN THE FOLLOWING we consider the evolution problems we have defined in the last chapter from a theoretical point of view. The interest now lies in the existence of solutions in the viscosity sense. The notion of viscosity solutions was introduced by Crandall and Lions in [20] for Hamilton-Jacobi equations. At this time uniqueness results were only available for the first order case, thus the first presentation of the theory was based on first order partial differential equations. Meanwhile the theory has been extended to second order PDEs, allowing merely continuous functions to be solutions of fully nonlinear (possibly degenerate) equations.

In our setting we are dealing with parabolic fully nonlinear degenerate equations of second order, which moreover have a singularity if the gradient of the solution vanishes. For mean curvature motion the existence and uniqueness of a viscosity solution have been proved independently by Evans and Spruck [32] and by Chen et al. [16]. It has been shown for non-convex initial surfaces $\mathcal{M}_c(0)$ that topology changes of this surface can occur in finite time [32]. If u is the viscosity solution of the MCM problem given a smooth level-set $\mathcal{M}_c(0) := \{u_0(x) = c\}$ as initial data, then $\mathcal{M}_c(t) = \{u(t, x) = c\}$ indeed represents the classical evolution of a parameterization of $\mathcal{M}_c(0)$ under mean curvature up to this time T . But this is not a problem for the level-set approach where also in less regular settings the evolution of the initial level-set is defined via $\mathcal{M}_c(t) := \{u(t, x) = c\}$ for all scales $t \in \mathbb{R}^+$. The existence of a viscosity solution is also guaranteed, even if its gradient vanishes and the definition of a normal does not make sense any more. The convergence of the solution of a finite element scheme toward the viscosity solution has been proven by Deckelnick and Dziuk [22] for MCM of graphs and MCM in the level-set context.

The goal of this chapter is first to prove existence of solutions for a linear anisotropic level-set problem, which results from the models we have considered in the last chapter by replacing the nonlinear diffusion tensor $A(u^\sigma)$ by a diffusion tensor $A(v)$ that depends on a fixed image or image-sequence $v \in C^0$. The existence proof uses the Perron method introduced by Ishii [42] and a comparison theorem for sub- and super-solutions [35]. The second part of the chapter deals with the continuity of the solution-operator, which means the continuous dependence of the problems on the functions v , which define the diffusion tensor. This leads to the existence of a solution of the nonlinear problem with the Schauder Fixed-Point Theorem.

Unfortunately, the theory currently present for MCM does not transfer to the nonlinear evolution problems we have modeled in the last chapter. Also existing work on fully nonlinear problems [45, 44, 46] and degenerate equations [72] is not applicable, since we do not have any monotonicity assumptions on the diffusion tensor. Fortunately the notion of viscosity solutions exhibits a great flexibility in passing to limits in various settings, which enables us to prove the results in the second part of the chapter. Thereby we make use of the notion of $\Gamma^{+/-}$ limits, which were introduced first by De Giorgi [21]. But before we go

into the details, let us recall some facts of the basic theory.

4.1 Reviewing basic theory of viscosity-solutions

The notion of viscosity solution is a purely pointwise definition. As already pointed out, solutions u can be merely continuous. In general, derivatives of u do not have the classical meaning. Indeed the theory of viscosity solutions encompasses classes of problems that do not have solutions which are differentiable in the classical sense [19]. In the theory of viscosity solutions the notion of derivatives of u is replaced by the so called *jets* of the function u . To precise this notion, it is useful to have the following notations

$$\begin{aligned} USC(\mathcal{O}) &:= \{u : \mathcal{O} \rightarrow \mathbb{R} : u \text{ is upper semi-continuous} \}, \\ LSC(\mathcal{O}) &:= \{v : \mathcal{O} \rightarrow \mathbb{R} : v \text{ is lower semi-continuous} \}. \end{aligned}$$

Furthermore, in the following $S(d)$ denotes the symmetric matrices of dimension $d \times d$. The power set of a set \mathcal{O} is denoted with $P(\mathcal{O})$. As usual we denote the Hölder spaces with $C^{k,\alpha}(\mathcal{O})$, for $\alpha \in [0, 1]$. We denote by $C_0^0(J \times \mathcal{O})$ for $J \subset \mathbb{R}^+$ the space of all continuous functions $u : J \times \mathcal{O} \rightarrow \mathbb{R}$ such that $u(t, \cdot)$ has compact support in \mathcal{O} for all $t \in J$. Since in this chapter all solutions are to be seen in the viscosity sense, we will often simply speak of solutions omitting the term “viscosity”.

Definition 4.1. *For the time space cylinder $Q := J \times \Omega$, $J \subset \mathbb{R}_0^+$ over the spatial domain $\Omega \subset \mathbb{R}^d$ consider a function $u \in USC(Q)$. The parabolic second order super-jet of u , denoted with $\mathcal{P}_Q^{2,+}u$, is a mapping*

$$\mathcal{P}_Q^{2,+}u : Q \rightarrow P\left(\mathbb{R} \times \mathbb{R}^d \times S(d)\right),$$

from Q into the power-set of $\mathbb{R} \times \mathbb{R}^d \times S(d)$, where $S(d)$ denotes the symmetric $d \times d$ matrices. For any $(t, x) \in Q$, $\mathcal{P}_Q^{2,+}u(t, x)$ is the set of $(a, p, X) \in \mathbb{R} \times \mathbb{R}^d \times S(d)$ such that

$$u(s, y) \leq u(t, x) + a(s - t) + \langle p, y - x \rangle + \frac{1}{2} \langle X(y - x), (y - x) \rangle + o(|s - t| + |y - x|^2)$$

for $(s, y) \rightarrow (t, x)$ in Q . For $v \in LSC(Q)$ the parabolic second order sub-jet $\mathcal{P}_Q^{2,-}v$ is defined analogously. The relation $\mathcal{P}_Q^{2,-}v(t, x) = -\mathcal{P}_Q^{2,+}(-v(t, x))$ holds.

Remark 4.2. *As long as (t, x) is an interior point of Q the jets $\mathcal{P}_Q^{2,+}u$ and $\mathcal{P}_Q^{2,-}u$ are independent of Q . We indicate this fact by omitting the subscript Q and writing $\mathcal{P}^{2,+}u$ and $\mathcal{P}^{2,-}u$ whenever we are working on open sets $Q = J \times \Omega$. Furthermore for twice continuously differentiable functions $u \in C^2(Q)$ the classical derivatives lie in the intersection of the sub- and super-jet. In particular, we then have*

$$\{(\partial_t u, \nabla u, D^2 u)(t, x)\} = \mathcal{P}^{2,+}u(t, x) \cap \mathcal{P}^{2,-}u(t, x).$$

For semi-continuous functions u the elements of the jets $\mathcal{P}^{2,+}u$ and $\mathcal{P}^{2,-}u$ replace the (possibly non existing) classical derivatives of u . Figuratively speaking the jets contain all paraboloids lying locally above or below u (cf. Figure 4.1). This is furthermore motivated by the following equivalent definition (a proof of the equivalence is given in [53]), which is used more often throughout our analysis:

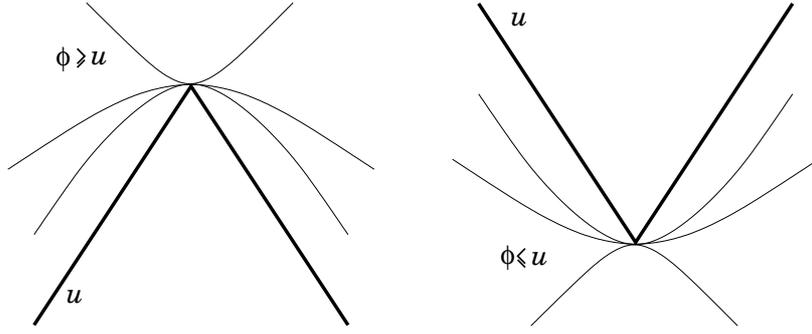


Figure 4.1: *Left: The super-jet $\mathcal{P}^{2,+}u$ contains all paraboids ϕ lying locally above the function u . In the case illustrated here, the sub-jet $\mathcal{P}^{2,-}u$ is empty. Right: The sub-jet $\mathcal{P}^{2,-}u$ contains paraboids ϕ lying locally below the function u . In the setting displayed in the figure the super-jet $\mathcal{P}^{2,+}u$ is empty.*

Lemma 4.3. *The following definitions of the super- and sub-jet are equivalent to definition 4.1 (cf. Figure 4.1):*

$$\begin{aligned} \mathcal{P}_Q^{2,+}u(t,x) &:= \{(a,p,X) \in \mathbb{R} \times \mathbb{R}^d \times S(d) : \exists \phi \in C^2(Q) \text{ such that} \\ &\quad (a,p,X) = (\partial_t \phi, \nabla \phi, D^2 \phi) \text{ and } u - \phi \text{ has a local maximum at } (t,x)\}, \\ \mathcal{P}_Q^{2,-}v(t,x) &:= \{(a,p,X) \in \mathbb{R} \times \mathbb{R}^d \times S(d) : \exists \phi \in C^2(Q) \text{ such that} \\ &\quad (a,p,X) = (\partial_t \phi, \nabla \phi, D^2 \phi) \text{ and } v - \phi \text{ has a local minimum at } (t,x)\}. \end{aligned}$$

We call the C^2 -functions ϕ whose derivatives form the super-jet and sub-jet the test-functions.

Using the jets, we are now able to give the definition of viscosity sub- and super-solutions (cf. [19]):

Definition 4.4. *A function $\bar{u} \in USC(Q)$ is called a viscosity sub-solution of $F(t,x,u,\partial_t u, \nabla u, D^2 u) = 0$, if for all $(t,x) \in Q$*

$$F(t,x,\bar{u}(t,x),a,p,X) \leq 0 \quad \forall (a,p,X) \in \mathcal{P}_Q^{2,+}\bar{u}(t,x).$$

A function $\underline{u} \in LSC(Q)$ is called a viscosity super-solution of $F(t,x,u,\partial_t u, \nabla u, D^2 u) = 0$, if for all $(t,x) \in Q$

$$F(t,x,\underline{u}(t,x),a,p,X) \geq 0 \quad \forall (a,p,X) \in \mathcal{P}_Q^{2,-}\underline{u}(t,x).$$

Finally, a function $u \in C^0(Q)$ is called a viscosity solution of $F(t,x,u,\partial_t u, \nabla u, D^2 u) = 0$, if it is a sub- and a super-solution of $F = 0$.

This definition will not be the one, we use for our existence analysis here. Because of the fact, that the geometric problems are degenerate in normal direction, we have to relax the definition slightly as we will see below.

An important device in viscosity solution theory is the doubling of spatial variables, where the sum of viscosity sub- and super-solutions is considered. Let us record the following important lemma concerning possible choices of test-functions for the doubling of variables:

Lemma 4.5 (Crandall, Ishii [18]). *Let $u_i \in USC((0,T) \times \bar{\Omega})$, with $u_i < \infty$ in $(0,T) \times \Omega$ for $i = 1, \dots, k$, $T > 0$, and $\Omega \subset \mathbb{R}^d$ open and bounded. Furthermore let the function w be defined on $(0,T) \times \Omega \times \dots \times \Omega$ via*

$$w(s,z) := u_1(s,z_1) + \dots + u_k(s,z_k)$$

with $z = (z_1, \dots, z_k) \in \Omega \times \dots \times \Omega$. For $(s, z) \in (0, T) \times (\Omega \times \dots \times \Omega)$ let

$$(\tau, p, Z) \in \mathcal{P}_{\Omega \times \dots \times \Omega}^{2,+} w(s, z) \subset \mathbb{R} \times \mathbb{R}^{kd} \times S(kd),$$

with $p = (p_1, \dots, p_k)$, $p_i \in \mathbb{R}^d$, and let the functions u_i be viscosity solutions of

$$\partial_t u + F_i(t, x, \nabla u, D^2 u) \leq 0$$

in a neighborhood of $(s, z_i) \in (0, T) \times \Omega$ for $i = 1, \dots, k$, where the functions $F_i : (0, T) \times \Omega \times \mathbb{R}^d \times S(d) \rightarrow \mathbb{R}$ are lower semi-continuous. Then for each $\gamma > 0$ there exist $X_i \in S(d)$ such that

$$(\tau, p_i, X_i) \in \mathcal{P}^{2,+} u_i(s, z_i) \quad \text{and so} \quad \tau + \sum_{i=1}^k F_i(s, z_i, p_i, X_i) \leq 0.$$

Furthermore the estimate

$$-\left(\frac{1}{\gamma} + \|Z\|\right) \text{Id} \leq \begin{pmatrix} X_1 & \dots & O \\ \vdots & & \vdots \\ O & \dots & X_k \end{pmatrix} \leq Z + \gamma Z^2$$

holds. Here Id is the identity matrix, and $\|\cdot\|$ denotes the spectral norm on $S(kd)$.

Concerning the test functions of $w(s, z) = u_1(s, z_1) + \dots + u_k(s, z_k)$ we need the following elementary lemma, whose proof can be found in e.g. [19].

Lemma 4.6. *Let $\Omega \subset \mathbb{R}^d$ and $J := (0, T)$. Consider a non-negative test function $\Phi \in \text{USC}(\Omega \times \dots \times \Omega)$, $\Phi \geq 0$ and for w as in Lemma 4.5 the suprema*

$$M_\alpha = \sup_{J \times (\Omega \times \dots \times \Omega)} (w(t, x) - \alpha \Phi(x))$$

for $\alpha > 0$ and $x = (x_1, \dots, x_k)$. Furthermore let $-\infty < \lim_{\alpha \rightarrow \infty} M_\alpha < \infty$ and let the family $((t_\alpha, x_\alpha))_\alpha \subset J \times (\Omega \times \dots \times \Omega)$ be chosen such that

$$\lim_{\alpha \rightarrow \infty} (M_\alpha - (w(t_\alpha, x_\alpha) - \alpha \Phi(x_\alpha))) = 0.$$

Then

$$(i) \quad \lim_{\alpha \rightarrow \infty} \alpha \Phi(x_\alpha) = 0$$

$$(ii) \quad \Phi(\hat{x}) = 0 \quad \text{and}$$

$$\lim_{\alpha \rightarrow \infty} M_\alpha = w(\hat{t}, \hat{x}) = \sup_{\{\Phi(x)=0\}} w(t, x)$$

if $(t_\alpha, x_\alpha) \rightarrow (\hat{t}, \hat{x}) \in J \times (\Omega \times \dots \times \Omega)$ for $\alpha \rightarrow \infty$.

4.2 Anisotropic linear geometric diffusion

After this review of the basic concepts of the viscosity solution theory, let us first consider the linear anisotropic problem. In the following we no longer make a distinction between the static-image model and the model for image-sequence processing. The assumptions we make below on the diffusion tensor A immediately apply to both of the models. In one case we work on $\Omega \subset \mathbb{R}^d$ and in the other case on $\Omega \subset \mathbb{R}_0^+ \times \mathbb{R}^d$. Therefore we make a small change of notation here and write Ω regardless whether we are working on a static image or an image-sequence.

But furthermore for reasons of simplicity we replace the domain $\Omega \subset \mathbb{R}^d$ by whole \mathbb{R}^d . Thus, we do not have to deal with the Neumann boundary conditions any longer. However, the comparison and existence results we discuss below, consider functions in

$$C_0^0(Q) := \{v \in C^0(Q) : \exists R > 0 \text{ such that } v(t, x) = 0 \text{ for } x \notin B(R)\},$$

i.e. functions which are constant outside some spatial ball around the origin. Here and in the following for $\Omega \subset \mathbb{R}^d$ the parabolic domain is $Q := J \times \Omega$ where $J := [0, T]$ for any finite scale $T > 0$. For bounded Ω we denote the parabolic boundary of $J \times \Omega$ by $\partial_p Q := \{0\} \times \Omega \cup J \times \partial\Omega$.

In this section we are considering the following problem:

Given $g \in C_0^0(\mathbb{R}^d)$ and $v \in C_0^0(J \times \mathbb{R}^d)$ find $u \in C^0(J \times \mathbb{R}^d)$ such that

$$\begin{aligned} \partial_t u - |\nabla u| \operatorname{div} \left(A(v) \frac{\nabla u}{|\nabla u|} \right) &= 0 && \text{in } J \times \mathbb{R}^d, \\ u(0, \cdot) &= g(\cdot) && \text{in } \mathbb{R}^d. \end{aligned}$$

4.2.1 Compactness of the diffusion tensor

In the problem above the nonlinearity A is always a compact operator

$$A : C^0(Q) \rightarrow C^{1,1}(Q, S(d)), \quad 0 \leq A(v)(t, x) \leq \Lambda \operatorname{Id}, \quad \lambda < \infty, \quad (4.1)$$

giving symmetric, bounded and elliptic diffusion coefficients $A(v)(t, x)$. In Section 2 we have constructed the diffusion tensor A such that it depends on regularized curvatures $\kappa^{i,\sigma}$ or additionally the apparent velocity v_{app}^σ and acceleration $\operatorname{accel}_{\text{app}}^\sigma$. The weighting is thereby steered by the function $G(\cdot) \leq 1$, thus giving a bounded diffusion tensor $A(v) \leq \operatorname{Id}$.

Lemma 4.7. *The diffusion tensors $a(\Sigma)$ and $a_{(s,x)}(\Sigma, v_{\text{app}}^\sigma)$, whose arguments result from the regularization variants (R3) and (R4) are compact operators from $C^0(Q)$ into $C^{1,1}(Q, S(d))$.*

Proof. Since $a(\Sigma)$ and $a_{(s,x)}(\Sigma, v_{\text{app}}^\sigma)$ depend on the first and second derivative of a regularized version of the image, we assume that we can write them as

$$(A(v^n)(t, x))_{ij} = \mathbf{a}_{ij}(\nabla(v^n)^\sigma(t, x), D^2(v^n)^\sigma(t, x)) \quad \text{for } \mathbf{a}_{ij} \in C^{1,1}(\mathbb{R}^d \times S(d)).$$

Now let $(v^n)_n$ be a bounded sequence in $C^0(Q)$. We show that there exists a $A^* \in C^{1,1}(Q, S(d))$ such that for a subsequence v^{n_j}

$$A(v^{n_j}) \rightarrow A^* \quad \text{in } C^{1,1}(Q, S(d))$$

for $j \rightarrow \infty$. Because $\mathbf{a}_{ij} \in C^1(\mathbb{R}^d \times S(d))$, it suffices to show the existence of $w \in C^4(Q)$ and the convergence in $C^2(Q)$ of

$$\nabla(v^n)^\sigma \rightarrow \nabla w \quad \text{and} \quad D^2(v^n)^\sigma \rightarrow D^2 w. \quad (4.2)$$

This is what we show below for the two regularization variants under consideration:

- *Convolution with C_0^∞ kernels.* For a multi-index γ , let D^γ be a spatial derivative of order less than six, i.e. $|\gamma| < 6$. Substituting the definition of $(v^n)^\sigma = K^\sigma * v^n$, we have

$$\|D^\gamma(K^\sigma * v^n)\|_{L^\infty(Q)} \leq \sup_{(t,x) \in Q} \int_{\Omega} |(D^\gamma K^\sigma)(x-y)| |v^n(t,y)| dy \leq C \|v^n\|_{L^\infty(Q)}.$$

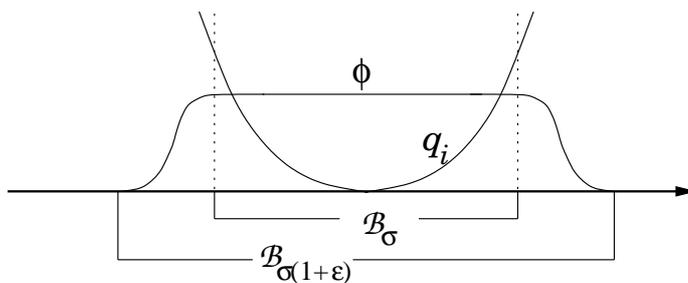


Figure 4.2: We interpret the local L^2 -projection as a local filter, where the C_0^∞ function ϕ cuts off the monomials q_i outside $\mathcal{B}_{\sigma(1+\varepsilon)}$.

This is a bound of $D^\gamma(v^n)^\sigma$ in $C^0(Q)$ and we can use Arzela-Ascoli's Theorem and the compact inclusion $C^5 \hookrightarrow C^{4,\alpha}$ to get the existence of a $w \in C^4(Q)$ such that $(v^{n_j})^\sigma \rightarrow w$ in $C^4(Q)$ for a subsequence v^{n_j} .

• *Local L^2 -projection onto a polynomial space.* We consider the local L^2 projection $v_{(t,x)}^{n,\sigma} \in Q^* \subset \mathcal{P}_l(\mathbb{R}^d)$ of $v^n(t, \cdot)$ onto Q^* . Let us denote basis-functions of Q^* with q_i , i.e. $Q^* := \text{span}\{q_i : i = 1, \dots, K\}$. We have seen in Chapter 3 that the projection is given as the solution of a linear system of equations for a fixed $t \in J$

$$\int_{\mathcal{B}_{\sigma(x)}} v_{(t,x)}^{n,\sigma}(t, y) q_i(y) dy = \int_{\mathcal{B}_{\sigma(x)}} v^n(t, y) q_i(y) dy \quad \text{for } i = 1, \dots, K.$$

If we write the projection in terms of the basis q_i

$$v_{(t,x)}^{n,\sigma}(t, y) = \sum_{i=1}^K w_i^t q_i(y)$$

for $w_i^t = w_i^t(x) \in \mathbb{R}$. Note that here and below the superscript t indicates the dependence on $t \in J$ and not as usual the transposition operation. We can rewrite the above system of equations as

$$H w^t(x) = R^t(x),$$

with $w^t(x) = (w_i^t(x))_i$, $R^t(x) = (R_i^t(x))_i$, $H = (H_{ij})_{ij}$ where

$$H_{ij} = \int_{\mathcal{B}_{\sigma(0)}} q_i q_j(y) dy \quad \text{and} \quad R_i^t(x) = \int_{\mathcal{B}_{\sigma(x)}} v^n(t, y) q_i(x - y) dy.$$

So the coefficients $w^t(x)$ are computed via $w^t(x) = H^{-1} R^t(x)$. Now, we replace the integration of the components of the right-hand-side $(R_i^t(x))$ by the following global filters

$$\tilde{R}_i^t(x) = \int_{\mathcal{B}_{\sigma(1+\varepsilon)}(x)} \phi(x - y) q_i(x - y) v^n(t, y) dy,$$

where $\varepsilon > 0$ and $\phi \in C_0^\infty(\Omega)$ is chosen such that (cf. Figure 4.2)

$$\phi(x - y) = 1 \quad \text{for } y \in \mathcal{B}_{\sigma}(x) \quad \text{and} \quad \phi(x - y) = 0 \quad \text{for } y \notin \mathcal{B}_{\sigma(1+\varepsilon)}(x).$$

Considering the solution of the system $\tilde{w}^t(x) = H^{-1} \tilde{R}^t(x)$, we see that the following estimate

holds for any multi-index γ :

$$\begin{aligned} \|D^\gamma \tilde{w}^t(x)\|_{L^\infty(Q)} &\leq \sup_{(t,x) \in Q} C |D^\gamma \tilde{R}^t(x)| \\ &\leq \sup_{(t,x) \in Q} C \left(\int_{\mathcal{B}_\sigma(x)} |D^\gamma q_i(x-y) v^n(t,y)| dy + \right. \\ &\quad \left. \int_{\mathcal{B}_{\sigma(1+\varepsilon)}(x) \setminus \mathcal{B}_\sigma(x)} |D^\gamma (\phi(x-y) q_i(x-y)) v^n(t,y)| dy \right) \\ &\leq \sup_{(t,x) \in Q} C \left(\|v^n(t, \cdot)\|_{L^\infty(\mathcal{B}_\sigma(x))} + \varepsilon \|v^n(t, \cdot)\|_{L^\infty(\mathcal{B}_{\sigma(1+\varepsilon)}(x) \setminus \mathcal{B}_\sigma(x))} \right). \end{aligned}$$

But since the v^n are bounded in $C^0(Q)$, this estimate means that for $\varepsilon \rightarrow 0$ we have

$$\|D^\gamma w^t\| \leq C \|v^n\|_{L^\infty(Q)},$$

which is again a bound on the derivatives of the regularization in $C^0(Q)$. Now we apply the Theorem of Arzela-Ascoli again, to finish the proof as above. \square

Remark 4.8. *If $(v^n)_n \subset C^0(Q)$ such that $v^n \rightarrow w$ in $C^0(Q)$ then this convergence in $C^0(Q)$ is transformed by A into convergence $A(v^n) \rightarrow A(w)$ in $C^{1,1}(Q)$. Indeed, for the convolution regularization variant, we can write for any multi-index γ :*

$$\|D^\gamma ((v^n)^\sigma - w^\sigma)\|_{L^\infty(Q)} \leq \sup_{(t,x) \in Q} \int_\Omega |D^\gamma K^\sigma(x-y)| |v^n - w|(t,y) dy \leq C \|v^n - w\|_{L^\infty(Q)},$$

which in particular means the convergence of $D^\gamma (v^n)^\sigma \rightarrow D^\gamma w^\sigma$ in $C^0(Q)$.

For the projection approach, we denote the coefficients of the projection $v_{(t,x)}^\sigma$ by $\tilde{W}^t(x)$ and as in the proof of the lemma, we obtain

$$\|D^\gamma (\tilde{w}^t(x) - \tilde{W}^t(x))\|_{L^\infty(Q)} \leq C \|v^n - v\|_{L^\infty(Q)},$$

which means the $C^0(Q)$ convergence of the derivatives as well.

4.2.2 A comparison theorem

Let us first outline the course of the existence proof for the linear problem, which is the topic of the following sections. A comparison principle was proven by Giga et al. [35] for bounded domains $\Omega \subset \mathbb{R}^d$. We provide one sub- and one super-solution, which fulfill this requirement, and the application of Ishii's Perron-Method delivers a solution which has compact support as well.

Let us provide a strong formulation for the problem above. Similar to the computations in (2.4) the differentiation of the term $|\nabla u| \operatorname{div}(A(v)N)$ leads to the definition of $F[v] : J \times \mathbb{R}^d \times \mathbb{R}^d \setminus \{0\} \times S(d) \rightarrow \mathbb{R}$ with

$$F[v](t, x, p, X) := -\operatorname{div}(A(v)(t, x)) \cdot p - \operatorname{tr}((A(v))(t, x)\Pi X), \quad (4.3)$$

where $\Pi := \operatorname{Id} - (p \otimes p)/|p|^2$ is the projection onto the tangent space and $(\operatorname{div} A(v)(t, x))_i = \sum_{j=1}^d \partial_j (a_{ij}(v))(t, x)$. Now we can rewrite our problem for $\Omega \subset \mathbb{R}^d$ as:

For $g \in C_0^0(\Omega)$ and arbitrary but fixed $v \in C_0^0(J \times \Omega)$ find a viscosity solution $u \in C^0(J \times \Omega)$ of

$$\left. \begin{aligned} \partial_t u + F[v](t, x, \nabla u, D^2 u) &= 0 && \text{in } J \times \Omega, \\ u(0, \cdot) &= g(\cdot) && \text{in } \Omega. \end{aligned} \right\} \quad (\text{P}[v])$$

Still the function $F[v]$ is singular in points where the gradient vanishes. But fortunately the theory of viscosity solutions allows the application to semi-continuous problems. Therefore we consider the semi-continuous envelopes of the given F to provide a value at the point $p = 0$. This gives us semi-continuous functions $\underline{F}[v]$ and $\overline{F}[v]$ on which we base the existence analysis below. For the construction of these functions, we require the following observation:

Lemma 4.9. *The function $F[v]$ defined in (4.3) has the following properties:*

(i) *The estimate*

$$-\operatorname{div}(A(v)(t, x)) \cdot p - d\Lambda\|X\| \leq F[v](t, x, p, X) \leq -\operatorname{div}(A(v)(t, x)) \cdot p + d\Lambda\|X\|, \quad (4.4)$$

holds, or in other words we have

$$|\operatorname{tr}((A(v))(t, x)\Pi X)| \leq d\Lambda\|X\|,$$

where $\|\cdot\|$ denotes the spectral-norm on $S(d)$.

(ii) *F is degenerate elliptic, i.e. for all $(t, x, p, X, Y) \in J \times \mathbb{R}^d \times \mathbb{R}^d \setminus \{0\} \times S(d) \times S(d)$ with $X \leq Y$*

$$F(t, x, p, X) \geq F(t, x, p, Y).$$

Proof. (i) Since Π is the projection onto a $d-1$ dimensional subspace (the tangent space), Π has eigenvalues $\{0, 1, \dots, 1\}$ and so it is clear that $\|\Pi X\| \leq \|X\|$. Moreover we have $-d\|X\| \leq \operatorname{tr}(X) \leq d\|X\|$, from which the claim follows at once using the upper bound on A .

(ii) For any $W \geq 0$ we have $\operatorname{tr}(W) \geq 0$ and thus $\operatorname{tr}(\Pi W) \geq 0$ as well. Since the diffusion tensor is positive definite, we conclude that for $W := Y - X \geq 0$ we have $\operatorname{tr}(A(t, x)\Pi(Y - X)) \geq 0$ and so

$$-(\operatorname{div}A)(t, x) \cdot p - \operatorname{tr}(A(t, x)\Pi X) \geq -(\operatorname{div}A)(t, x) \cdot p - \operatorname{tr}(A(t, x)\Pi Y).$$

□

Now we define the function $\underline{F}[v] \in \operatorname{LSC}(J \times \mathbb{R}^d \times \mathbb{R}^d \times S(d))$ and the function $\overline{F}[v] \in \operatorname{USC}(J \times \mathbb{R}^d \times \mathbb{R}^d \times S(d))$ as

$$\begin{aligned} \underline{F}[v](t, x, p, X) &:= \begin{cases} F[v](t, x, p, X) & \text{if } p \neq 0, \\ -d\Lambda\|X\| & \text{else,} \end{cases} \\ \overline{F}[v](t, x, p, X) &:= \begin{cases} F[v](t, x, p, X) & \text{if } p \neq 0, \\ d\Lambda\|X\| & \text{else,} \end{cases} \end{aligned} \quad (4.5)$$

and the first estimate of Lemma 4.9 ensures that they are indeed the lower and upper semi-continuous envelope of $F[v]$. We emphasize the important fact

$$\underline{F}[v](t, x, 0, O) = \overline{F}[v](t, x, 0, O), \quad (4.6)$$

where O denotes the “zero” matrix in $S(d)$. Furthermore for the later use, let us introduce the positive function

$$l : [0, \infty) \rightarrow \mathbb{R}, \quad s \mapsto s \|\operatorname{div}A(v)\|_{L^\infty(Q)} + d\Lambda,$$

such that we can rewrite the estimate (4.4) for $X = \pm \operatorname{Id}$ in the form

$$\underline{F}(t, x, p, -\operatorname{Id}) \leq l(|p|) \quad \text{and} \quad \overline{F}(t, x, p, \operatorname{Id}) \geq -l(|p|). \quad (4.7)$$

Remark 4.10. Throughout the remainder of this section, notation is chosen such that upper semi-continuous functions are indicated by an upper bar, e.g. \bar{u} , lower semi-continuous functions carry a lower bar, e.g. \underline{v} . The correspondence between functions and problems is always such that upper and lower bars appear as couples, e.g. \bar{u} solves $\underline{F} \leq 0$. Similar notation is chosen for the * super-/subscripts, which appear later.

Now we can formulate our definition of viscosity solutions of the problem $(P[v])$ as usual for singular problems (cf. e.g. [19, 43]) based on $\underline{F}[v]$ and $\bar{F}[v]$:

Definition 4.11. An upper semi-continuous function $\bar{u} \in USC(J \times \Omega)$ is called viscosity sub-solution of $(P[v])$ on $\Omega \subset \mathbb{R}^d$, if

$$\begin{aligned} a + \underline{F}[v](t, x, p, X) &\leq 0 && \text{for all } (t, x) \in J \times \Omega, \quad \text{and } (a, p, X) \in \mathcal{P}^{2,+}\bar{u}(t, x), \\ \bar{u}(0, x) &\leq g(x) && \text{for all } x \in \Omega. \end{aligned}$$

Analogously, a lower semi-continuous function $\underline{u} \in LSC(J \times \Omega)$ is called viscosity super-solution of $(P[v])$ on $\Omega \subset \mathbb{R}^d$, if

$$\begin{aligned} a + \bar{F}[v](t, x, p, X) &\geq 0 && \text{for all } (t, x) \in J \times \Omega, \quad \text{and } (a, p, X) \in \mathcal{P}^{2,-}\underline{u}(t, x), \\ \underline{u}(0, x) &\geq g(x) && \text{for all } x \in \Omega. \end{aligned}$$

A function $u \in C^0(J \times \Omega)$ is called viscosity solution of $(P[v])$ if it is a sub-solution and a super-solution of $(P[v])$.

Obviously this definition makes sense, because for non-singular problems it results in the original definition of viscosity solutions 4.4. The following lemma is a basic fact, which we need later for the construction of an initial sub- and a super-solution. A proof of the lemma can be found in e.g. [19].

Lemma 4.12. Let \mathcal{A} be a nonempty family of sub-solutions of $(P[v])$ and the function u be defined via

$$u(x) := \sup\{v(x) : v \in \mathcal{A}\} \quad \text{for } x \in Q.$$

If u is bounded, then u is a sub-solution of $(P[v])$. An analogue result holds for super-solutions.

Before we come to the comparison theorem let us note a result concerning a temporal perturbation of sub-solutions, which acts as a temporal barrier for $t \rightarrow T$:

Lemma 4.13. Let the function u be the solution of a problem $F(t, x, \nabla u, D^2 u) \leq 0$, i.e. a sub-solution of $F(t, x, \nabla u, D^2 u) = 0$ fulfilling the initial condition $u(0, \cdot) = g(\cdot)$. Then the function

$$u_\varepsilon := u - \frac{\varepsilon}{T-t}$$

solves the problem with strict inequality, i.e. $F(t, x, \nabla u_\varepsilon, D^2 u_\varepsilon) < 0$, and it fulfills the initial condition

$$u_\varepsilon(0, x) \leq g(x) \quad \text{for } x \in \Omega.$$

Proof. We choose $(a, p, X) \in \mathcal{P}^{2,+}u_\varepsilon(t, x)$ for any $(t, x) \in Q$. Then

$$\left(a + \frac{\varepsilon}{(T-t)^2}, p, X \right) \in \mathcal{P}^{2,+}u(t, x)$$

holds and therefore we can insert the latter into the problem to obtain

$$a + \frac{\varepsilon}{(T-t)^2} + F(t, x, p, X) \leq 0.$$

In other words

$$a + F(t, x, p, X) \leq -\frac{\varepsilon}{(T-t)^2} \leq -\frac{\varepsilon}{T^2} =: -c(\varepsilon) < 0.$$

Clearly the original initial condition is weakened by the subtraction of $\varepsilon/(T-t)$, i.e. $u_\varepsilon(0, \cdot) \leq u(0, \cdot) = g(\cdot)$. \square

Theorem 4.14 (Comparison principle [35]). *Let $\bar{u} \in USC(\overline{J \times \Omega})$ be a sub-solution, and $\underline{u} \in LSC(\overline{J \times \Omega})$ be a super-solution of $(P[v])$ for bounded $\Omega \subset \mathbb{R}^d$. Suppose furthermore $\bar{u} \leq \underline{u}$ on the parabolic boundary $\partial_p \Omega = \{0\} \times \Omega \cup J \times \partial\Omega$. Then $\bar{u} \leq \underline{u}$ on $\overline{J \times \Omega}$.*

Proof. This comparison principle is Theorem 4.9 in the work of Giga et al. [35] for problems described by functions H , which fulfill the following conditions

(F1) $H : J \times \Omega \times (\mathbb{R}^d \setminus \{0\}) \times S(d) \rightarrow \mathbb{R}$ is continuous

(F2) H is degenerate elliptic

(F3) $-\infty < \underline{H}(t, x, 0, O) = \overline{H}(t, x, 0, O) < \infty$, where \underline{H} and \overline{H} are the semi-continuous relaxations of H

(F8) There is a modulus σ_2 such that for $x, y \in \Omega$, $p \neq 0$

$$|H(t, x, p, X) - H(t, y, p, X)| \leq \sigma_2(|x - y|(|p| + 1))$$

(F11) H is geometric, i.e. for $p \neq 0$ and $\lambda > 0$, $\mu \in \mathbb{R}$

$$H(t, x, \lambda p, \lambda X + \mu p \otimes p) = \lambda H(t, x, p, X)$$

(F12) There exists a function $c \in C^1([0, \infty))$ such that $c(0) \geq c_0 > 0$ for some constant c_0 and

$$\underline{H}(t, x, p, -\text{Id}) \leq c(|p|) \quad \text{and} \quad \overline{H}(t, x, p, \text{Id}) \geq -c(|p|).$$

Thus, we have to check these assumptions for $H = F$. The assertion of (F1)-(F3) is already given by Lemma 4.9 and equation (4.6). Let us construct the modulus σ_2 from (F8). We have

$$\begin{aligned} |F[v](t, x, p, X) - F[v](t, y, p, X)| &= |(\text{div} A(v)(t, x)) \cdot p - (\text{div} A(v)(t, y)) \cdot p| \\ &\leq |(\text{div} A(v)(t, x)) - (\text{div} A(v)(t, y))| |p| \\ &\leq C|x - y| |p|, \end{aligned}$$

because $\text{div} A(v)$ is Lipschitz due to assumption (4.1). Clearly F is geometric, due to the built-in projection $(\text{Id} - |p|^2(p \otimes p))$, i.e.

$$\begin{aligned} F(t, x, \lambda p, X + \mu p \otimes p) &= -(\text{div} A(v)(t, x)) \cdot (\lambda p) - \text{tr}(A(v)(t, x) \Pi(\lambda X + p \otimes p)) \\ &= -\lambda(\text{div} A(v)(t, x)) \cdot p - \lambda \text{tr}(A(v)(t, x) \Pi X). \end{aligned}$$

This ensures (F11). Finally the function c from (F12) is equal to the function l , which we have constructed in (4.7). \square

4.2.3 Existence of a viscosity solution

We continue the course of this chapter with the validity of the above comparison theorem for bounded domains $\Omega \subset \mathbb{R}^d$. A unique viscosity solution can now be obtained by the application of Ishii's Perron Method [42]: We have to construct one sub-solution and one super-solution, which is done in the following paragraph. But before, we need to improve our results concerning the transformations of gray values:

Lemma 4.15. *Let $u \in USC(Q)$ be a bounded viscosity solution of $\underline{F}[v] \leq 0$. Then for any continuous nondecreasing transformation of the gray-values $h : \mathbb{R} \rightarrow \mathbb{R}$, the function $h(u)$ is also a viscosity solution of $\underline{F}[v] \leq 0$. An analogue result holds for super-solutions.*

Proof. The proof of the lemma, which is given in e.g. [16], is only sketched roughly here: First the assertion is obtained for strictly increasing $h \in C^2(\mathbb{R})$. Thereby, the proof makes use of sup-convolutions, which are semi-convex and Lipschitz. The degenerate ellipticity of $\underline{F}[v]$ can be used with the second derivative of $D^2h(u)$ at a maximum point \hat{y} of $h(u) - \phi$, where ϕ is a suitable test function. To approximate this maximum point with a sequence $y_k \rightarrow \hat{y}$ the semi-continuity of $\underline{F}[v]$ is utilized. In a final step $h \in C^0(\mathbb{R})$ is approximated by a sequence $h_k \in C^2(\mathbb{R})$ with $h'_k > 0$. \square

Theorem 4.16 (Global existence and uniqueness). *There exists a unique viscosity solution $u \in C^0(J \times \mathbb{R}^d)$ of the problem $(P[v])$ on $\Omega = \mathbb{R}^d$. Furthermore there exists a constant*

$$R_1 = R_1(g, \|\operatorname{div} A(v)\|_{L^\infty(J \times \mathbb{R}^d)}), \quad (4.8)$$

such that $u(t, x) = 0$ for $|x| > R_1$ and $t \in J$, i.e. $u \in C_0^0(J \times \mathbb{R}^d)$. The following estimate holds:

$$\|u\|_{L^\infty(J \times \mathbb{R}^d)} \leq \|g\|_{L^\infty(\mathbb{R}^d)}. \quad (4.9)$$

Proof. In the proof we construct a sub- and a super-solution \bar{u} and \underline{u} which are compactly supported, to show that also the solution u has compact support. The constant R_1 is then obtained from this construction. If we construct \bar{u} and \underline{u} such that they fulfill the initial condition

$$\bar{u}(0, x) = \underline{u}(0, x) = g(x) \quad \text{for } x \in \mathbb{R}^d,$$

we can apply the Perron Method [42] to obtain a continuous $u \in C^0(J \times \mathbb{R}^d)$ such that

$$\bar{u} \leq u \leq \underline{u}$$

by comparison. For the construction of \bar{u} and \underline{u} , we are inspired by the ideas of Chen et al. [16] for MCM.

• *Step 1.* Let us first construct barriers to handle the unbounded domain $\Omega = \mathbb{R}^d$. We define

$$\psi^\pm(t, x) = \begin{cases} \mp(|x| - \omega t)^4 & \text{if } |x| > \omega t \\ 0 & \text{else,} \end{cases}$$

where $\omega > 0$ will be chosen appropriately below. Then ψ^- and ψ^+ are a sub- respectively a super-solution of the problem. We prove this assertion only for ψ^- . Because $\psi^- \in C^2$ is convex, we can use the degenerate ellipticity of $\underline{F}[v]$ to derive for $\nabla \psi^- \neq 0$

$$\begin{aligned} \underline{F}[v](t, x, \nabla \psi^-, D^2 \psi^-) &\leq \underline{F}[v](t, x, \nabla \psi^-, O) \\ &= -(\operatorname{div} A(v)(t, x)) \cdot \nabla \psi^- \leq \|\operatorname{div} A(v)\|_{L^\infty(Q)} |\nabla \psi^-|. \end{aligned}$$

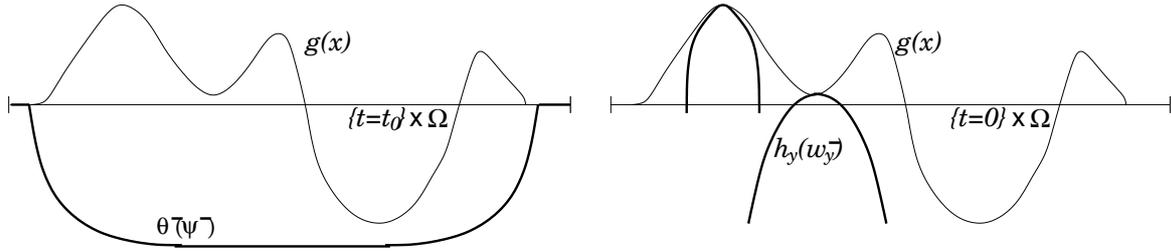


Figure 4.3: *Left: Illustration of the barrier function $\theta^-(\psi^-)$ which is a sub-solution of the problem. Right: Illustration of the transformed initial sub-solutions $h_y(w_y^-)$.*

A straightforward calculation shows that ψ^- solves $\partial_t \psi^- + \omega |\nabla \psi^-| = 0$ so choosing $\omega > \|\operatorname{div} A(v)\|_{L^\infty(J \times \mathbb{R}^d)}$ leads to

$$\begin{aligned} \partial_t \psi^- + \underline{F}[v](t, x, \nabla \psi^-, D^2 \psi^-) &\leq \partial_t \psi^- + \|\operatorname{div} A(v)\|_{L^\infty(Q)} |\nabla \psi^-| \\ &\leq \partial_t \psi^- + \omega |\nabla \psi^-| = 0. \end{aligned}$$

Additionally for $\nabla \psi^- = 0$, we have

$$\partial_t \psi^- + \underline{F}[v](t, x, 0, D^2 \psi^-) = \partial_t \psi^- - d\Lambda \|D^2 \psi^-\| \leq 0,$$

since $\partial_t \psi^- \leq 0$ for all $(t, x) \in Q$.

• *Step 2.* In the second step we will modify the functions ψ^\pm such that they are constant on $J \times \{x \in \mathbb{R}^d : |x| > R_1\}$ for some $R_1 > 0$. Let us consider the following continuous and nondecreasing transformations of the gray-values for $L > 0$

$$\theta^-(s) := \min\{s - L, 0\} \quad \text{and} \quad \theta^+(s) := \max\{s + L, 0\},$$

which again deliver a sub-solution $\theta^-(\psi^-)$ and a super-solution $\theta^+(\psi^+)$ in view of Lemma 4.15. We choose the constant $L > 0$ large enough so that $\theta^-(\psi^-(0, x)) \leq g(x) \leq \theta^+(\psi^+(0, x))$ (cf. Figure 4.3). Thus, $\theta^-(\psi^-)$ and $\theta^+(\psi^+)$ respect the initial conditions and we have

$$\begin{aligned} \theta^-(\psi^-(t, x)) &= 0 & \text{for } (t, x) \in J \times \{x \in \mathbb{R}^d : |x| > R_1\}, \\ \theta^+(\psi^+(t, x)) &= 0 & \text{for } (t, x) \in J \times \{x \in \mathbb{R}^d : |x| > R_1\}, \end{aligned}$$

for $R_1 = R_1(g, \|\operatorname{div} A(v)\|_{L^\infty(J \times \mathbb{R}^d)})$ appropriately chosen. Obviously g has compact support in $B(R_1)$.

• *Step 3.* So far the sub- and super-solution $\theta^-(\psi^-)$ and $\theta^+(\psi^+)$ do not attain the initial value at scale $t = 0$. We now construct functions which fulfill the initial condition and then combine these functions with $\theta^-(\psi^-)$ and $\theta^+(\psi^+)$ in the last step.

For any fixed $y \in \Omega$ we define the functions

$$w_y^\pm(t, x) = \pm(t + e(|x - y|)) \quad \text{with} \quad e(\rho) = \int_0^\rho s/l(s) ds.$$

Then w_y^- is a solution of $\underline{F}[v] \leq 0$ and w_y^+ a solution of $\overline{F}[v] \geq 0$. Again we ensure this property for w_y^- . Obviously for $x = y$ (i.e. $\rho := |x - y| = 0$) the function w_y^- is not twice continuously differentiable and thus, we have to consider the jet of w_y^- and show that

$$a + \underline{F}[v](t, x, p, X) \leq 0 \quad \text{for } (a, p, X) \in \mathcal{P}_Q^{2,+} w_y^-(t, x).$$

But for $x = y$ (i.e. $\rho := |x - y| = 0$) the super-jet of w_y^- is given by $\mathcal{P}_Q^{2,+} w_y^-(t, x) = \{-1\} \times \{0\} \times S(d)$ and so

$$-1 + \underline{F}[v](t, x, 0, X) = -1 - d\Lambda \|X\| \leq 0 \quad \text{for } X \in S(d).$$

It remains to show that w_y^- is a sub-solution for $x \neq y$, i.e. $\rho \neq 0$. We have

$$\nabla e(\rho) = e'(\rho)\nabla\rho, \quad D^2 e(\rho) = e''(\rho)\nabla\rho \otimes \nabla\rho + e'(\rho)D^2\rho, \quad D^2\rho = \frac{1}{\rho}(\text{Id} - \nabla\rho \otimes \nabla\rho).$$

and since $e'(\rho) = \rho/l(\rho)$

$$\nabla e(\rho) = \frac{\rho\nabla\rho}{l(\rho)}, \quad D^2 e(\rho) = \left(e''(\rho) + \frac{\rho}{l(\rho)\rho} \right) \nabla\rho \otimes \nabla\rho + \frac{1}{l(\rho)}\text{Id}.$$

Because $\underline{F}[v]$ considers only the projection Π of the second derivative onto the tangent space $\text{span}\{\nabla\rho\}^\perp$, a calculation yields

$$\begin{aligned} \underline{F}[v](t, x, \nabla w_y^-, D^2 w_y^-) &= \underline{F}[v] \left(t, x, -\frac{x-y}{l(|x-y|)}, -\frac{1}{l(|x-y|)}\text{Id} \right) \\ &= \frac{1}{l(|x-y|)} \underline{F}[v](t, x, -(x-y), -\text{Id}). \end{aligned}$$

Together with the temporal derivative we now have due to (4.7)

$$\partial_t w_y^- + \underline{F}[v](t, x, \nabla w_y^-, D^2 w_y^-) = -1 + \frac{1}{l(|x-y|)} \underline{F}[v](t, x, -(x-y), -\text{Id}) \leq 0,$$

which shows that w_y^- is a sub-solution if $\rho \neq 0$. Thus we have asserted that w_y^- is a solution of $\underline{F}[v] \leq 0$. Analogously one ensures that w_y^+ is a solution of $\overline{F}[v] \geq 0$.

Since $w_y^-(t, x)$ is nondecreasing with respect to $|x-y|$ and t , the continuity of g guarantees the existence of a continuous nondecreasing function $h_y : \mathbb{R} \rightarrow \mathbb{R}$ with $h_y(0) = g(y)$ and such that

$$h_y(w_y^-(t, x)) \leq g(x)$$

(cf. Figure 4.3). Due to the fact that $\underline{F}[v]$ is geometric, the function $h_y(w_y^-)$ is a sub-solution and by Lemma 4.12 the function

$$w_h^-(t, x) := \sup\{h_y(w_y^-(t, x)) : y \in \Omega\}$$

is again a sub-solution of $\underline{F}[v] \leq 0$. But this function now fulfills the initial condition, because $h_y(w_y^-(0, y)) = g(y)$. Similarly we construct a super-solution

$$w_h^+(t, x) := \inf\{h_y(w_y^+(t, x)) : y \in \Omega\}$$

• *Step 4.* Finally we can combine the sub- and super-solutions we have found in the last steps. Due to Lemma 4.12 the following functions again are sub- and super-solution

$$\overline{u} := \max\{w_h^-, \theta^-(\psi^-)\} \quad \text{and} \quad \underline{u} := \min\{w_h^+, \theta^+(\psi^+)\},$$

such that $\overline{u}(0, x) = g(x)$ because of $\theta^-(\psi^-(0, x)) = g(x)$ and $w_h^-(0, x) \leq g(x)$ and similarly $\underline{u}(0, x) = g(x)$. Moreover \overline{u} and \underline{u} have compact support in $B(R_1(g, \|\text{div} A(v)\|_{L^\infty(Q)}))$.

We apply the Perron Method to obtain a continuous solution, for which $\overline{u} \leq u \leq \underline{u}$ holds. Thus, the function u has compact support as well. This verifies the assertion of the theorem.

It remains to show that u is bounded by the initial value $\|g\|_{L^\infty(\mathbb{R}^d)}$. Clearly the function, which is constant equal $-\|g\|_{L^\infty(J \times \mathbb{R}^d)}$ is a sub-solution and analogously $\|g\|_{L^\infty(J \times \mathbb{R}^d)}$ a super-solution. If we consider these constant functions on the domain $J \times B(R_1)$, we can apply the comparison theorem 4.14, to obtain the desired estimate $\|u\|_{L^\infty(J \times B(R_1))} \leq \|g\|_{L^\infty(\mathbb{R}^d)}$ and thus $\|u\|_{L^\infty(J \times \mathbb{R}^d)} \leq \|g\|_{L^\infty(\mathbb{R}^d)}$. \square

4.3 Toward anisotropic nonlinear geometric diffusion

In the following sections we prove the existence of a solution for the nonlinear problem $(P[u])$. We consider the operator $P[\cdot]$ from above, i.e.

$$P : C_0^0(J \times \Omega) \rightarrow C_0^0(J \times \Omega), \quad v \mapsto \text{Solution of } P[v],$$

such that $P[v]$ is the solution of the problem $(P[v])$ in which the diffusion tensor is defined by v . The results of the last section ensure the existence of this solution. Below we show the continuity of this operator, which enables us to apply the Schauder Fixed-Point Theorem to assert the existence of a solution of the nonlinear problem $(P[u])$. The central result of this section is the following:

Theorem 4.17 (Continuity of $P[\cdot]$). *Let $(v^n)_{n \in \mathbb{N}} \subset C_0^0(J \times \mathbb{R}^d)$ be such that $v^n \rightarrow v$ in $C_0^0(J \times \mathbb{R}^d)$ for $n \rightarrow \infty$ and let u^n be the viscosity solution of the problem $(P[v^n])$ on $\Omega = \mathbb{R}^d$. Then there exists a continuous function u such that $u^n \rightrightarrows u$, and u is the viscosity solution of the problem $(P[v])$.*

The results from the last sections ensure the existence of the solution u^n of $(P[v^n])$. Since all the problems $(P[v^n])$ have the same initial condition, we have

$$\|u^n\|_{L^\infty(Q)} \leq \|g\|_{L^\infty(J \times \mathbb{R}^d)}$$

independently of n . Moreover, the compactness of A delivers a bound on $\|\operatorname{div} A(v^n)\|_{L^\infty(J \times \mathbb{R}^d)}$ for the bounded sequence $(v^n)_n \subset C_0^0(J \times \mathbb{R}^d)$ independently of n . Indeed we can bound any derivative of $A(v^n)$ by the C^0 norm of v^n and thus in view of (4.8) there exists a constant

$$R(g) := \sup_{n \in \mathbb{N}} R_1 \left(g, \|\operatorname{div} A(v^n)\|_{L^\infty(J \times \mathbb{R}^d)} \right) < \infty$$

which is independent of n , and such that the functions u^n are constant outside a ball of radius $R(g)$ around the origin, i.e.

$$u^n(t, x) = 0 \quad \text{for } (t, x) \in J \times \{x \in \mathbb{R}^d : |x| > R(g)\}. \quad (4.10)$$

Concerning the sequence v^n , we have seen in Remark 4.8 that the compactness of A transforms the convergence in $C^0(J \times \mathbb{R}^d)$ into convergence in $C^{1,1}(J \times \mathbb{R}^d)$. This means in particular

$$A(v^n) \rightrightarrows A(v) \quad \text{and} \quad \operatorname{div}(A(v^n)(\cdot)) \rightrightarrows \operatorname{div}(A(v)(\cdot)) \quad \text{for } n \rightarrow \infty, \quad (4.11)$$

as well, and we denote the semi-continuous problems resulting from the diffusion tensor $A(v)$ with \overline{F}^∞ and \underline{F}^∞ . Furthermore we introduce the notation \underline{F}^n and \overline{F}^n instead of $\underline{F}[v^n]$ and $\overline{F}[v^n]$ as well to simplify the notation.

4.3.1 $\Gamma^{+/-}$ - limits of the linear problems

Now we associate the following limits with the sequence of solutions u^n and the sequence of problems \underline{F}^n and \overline{F}^n :

Definition 4.18. *For a subset $L \subset B$ of a Banach-Space B and a sequence of functions $h_k : L \rightarrow \mathbb{R}$, $k = 1, 2, \dots$ the Γ^- -limit $h^* \in LSC(\overline{L})$ is defined via*

$$h^*(x) := \liminf_{k \rightarrow \infty} h_k(x) := \lim_{k \rightarrow \infty} \inf \left\{ h_l(y) : \|y - x\| \leq \frac{1}{k}, l \geq k \right\}.$$

The Γ^+ -limit $h_* \in USC(\bar{L})$ is defined by

$$h_*(x) := \limsup_{k \rightarrow \infty}^* h_k(x) := \lim_{k \rightarrow \infty} \sup \left\{ h_l(y) : \|y - x\| \leq \frac{1}{k}, l \geq k \right\}.$$

Remark 4.19. If $h_k = h$ for all k the Γ^- -limit h^* is called the lower semi-continuous relaxation of h , because $h \in LSC(\bar{L})$. Of course an analogue remark holds for the Γ^+ -limit.

The notion of $\Gamma^{+/-}$ -limits was introduced by De Giorgi [21]. It is especially important in the calculus of variations and also in stochastic control theory. From there we take the next result concerning the stability of viscosity solutions.

Theorem 4.20 (Barles, Perthame [8]). For $\Omega \subset \mathbb{R}^d$ and $J \subset \mathbb{R}$, let \bar{w}^n and \underline{w}^n be sequences of sub- and super-solutions of the problems $\underline{G}^n(t, x, \partial_t u, \nabla u, D^2 u) = 0$ and $\bar{G}^n(t, x, \partial_t u, \nabla u, D^2 u) = 0$ on $J \times \Omega$ with initial value $f \in C^0(\Omega)$. Then the function

$$W_*(t, x) := \limsup_{n \rightarrow \infty}^* \bar{w}^n(t, x) \in USC(Q)$$

is a sub-solution of the problem

$$G^* := \liminf_{n \rightarrow \infty}^* \underline{G}^n = 0,$$

supposed W_* and G^* are locally bounded. Similarly the function

$$W^*(t, x) := \liminf_{n \rightarrow \infty}^* \underline{w}^n(t, x) \in LSC(Q)$$

is a super-solution of the problem

$$G_* := \limsup_{n \rightarrow \infty}^* \bar{G}^n = 0,$$

again supposed W^* and G_* are locally bounded.

This theorem also considers the $\Gamma^{+/-}$ -limits of the problems \underline{G}^n and \bar{G}^n . The application of definition 4.18 (for $L := J \times \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d \times S(d)$) gives us the precise meaning: The Γ^- -limit of \underline{G}^n is given by

$$\liminf_{n \rightarrow \infty}^* \underline{G}^n(t, x, a, p, X) := \lim_{n \rightarrow \infty} \inf \left\{ \underline{G}^l(s, y, b, q, Y) : |s - t| \leq \frac{1}{n}, |y - x| \leq \frac{1}{n}, \right. \\ \left. |b - a| \leq \frac{1}{n}, |q - p| \leq \frac{1}{n}, \|Y - X\| \leq \frac{1}{n}, l \geq n \right\}$$

and analogously the Γ^+ -limit of the problems \bar{G}^n is defined.

Remark 4.21. It is quite natural to consider the $\Gamma^{+/-}$ - limits as in the theorem. Indeed, if \underline{G}^n was independent of n , the theorem would reduce to the observation that the limsup of sub-solutions is again a sub-solution, which is exactly the device that is employed with Perron's method for existence [42] (cf. also Lemma 4.12). In this sense passing to the $\Gamma^{+/-}$ - limits can be considered as a kind of diagonal version of Perron's method.

Although the proof of the theorem is given in [8] we repeat it here since we use a similar technique to derive a subsequent result. We need the following result according the approximation of the $\Gamma^{+/-}$ -limits, whose proof is given in e.g. [8, 19]:

Lemma 4.22. *Let $(\bar{w}^n)_{n \in \mathbb{N}} \subset USC(Q)$ be a sequence of upper semi-continuous functions, $(s, z) \in Q$ and furthermore $W_* < \infty$. For any triple $(a, p, X) \in \mathcal{P}^{2,+}W_*(s, z)$ there exist sequences*

$$n_j \rightarrow \infty, \quad (s_j, z_j) \in Q, \quad (a_j, p_j, X_j) \in \mathcal{P}^{2,+}\bar{w}^{n_j}(s_j, z_j)$$

such that

$$\begin{aligned} (s_j, z_j) &\rightarrow (s, z), \\ \bar{w}^{n_j}(s_j, z_j) &\rightarrow W_*(s, z), \\ a_j &\rightarrow a, \quad p_j \rightarrow p, \quad X_j \rightarrow X. \end{aligned}$$

The analogue result holds for lower semi-continuous functions.

Proof of Theorem 4.20. For the sequence of sub-solutions $(\bar{w}^n)_{n \in \mathbb{N}} \subset USC(Q)$, $(s, z) \in Q$ and $(a, p, X) \in W_*(s, z)$ let us consider the sequences provided by Lemma 4.22. We then clearly have

$$\underline{G}^{n_j}(s_j, z_j, a_j, p_j, X_j) \leq 0,$$

since $(a_j, p_j, X_j) \in \mathcal{P}^{2,+}\bar{w}^{n_j}(s_j, z_j)$ and because the \bar{w}^{n_j} are sub-solutions of $\underline{G}^{n_j} = 0$. In the definition of G^* we now insert these sequences we have found and therefore get an upper estimate

$$G^*(s, z, a, p, X) = \liminf_{n \rightarrow \infty} \underline{G}^n(s, z, a, p, X) \leq \liminf_{j \rightarrow \infty} \underline{G}^{n_j}(s_j, z_j, a_j, p_j, X_j) \leq 0,$$

because passing to the sub-sequence n_j increases the limes inferior. Since we have chosen $(a, p, X) \in W_*(s, z)$ the function W_* is a sub-solution of $G^* = 0$. The proof for W^* and G_* follows the same pattern. \square

Assigning the $\Gamma^{+/-}$ -limits with the sub-sequence of solutions u^n of the problems $(P[v^n])$, we learn from the last theorem that

$$U^* := \liminf_{n \rightarrow \infty} u^n \quad \text{and} \quad U_* := \limsup_{n \rightarrow \infty} u^n,$$

which are bounded in consequence of Theorem 4.16, solve the problems

$$\bar{F}_* := \limsup_{n \rightarrow \infty} (a + \bar{F}^n) = a + \limsup_{n \rightarrow \infty} \bar{F}^n \geq 0$$

and

$$\underline{F}^* := \liminf_{n \rightarrow \infty} (a + \underline{F}^n) = a + \liminf_{n \rightarrow \infty} \underline{F}^n \leq 0,$$

supposed \underline{F}^* and \bar{F}_* are locally bounded, which we verify in Section 4.3.4.

To prove Theorem 4.17 it remains to assert the following statements, which is the goal of the rest of this section

- The limit function of the $\Gamma^{+/-}$ -convergence is continuous, i.e. we have $U_* = U^* =: u \in C^0(J \times \mathbb{R}^d)$. This is shown in Sections 4.3.2 and 4.3.3.
- The $\Gamma^{+/-}$ -limit function u fulfills the initial conditions. This is asserted in Section 4.3.2.
- The limit problem is the anisotropic level-set problem corresponding to v , i.e. $\bar{F}_*(t, x, a, p, X) = a + \bar{F}[v](t, x, p, X)$ and $\underline{F}^*(t, x, a, p, X) = a + \underline{F}[v](t, x, p, X)$. This identification is treated in Section 4.3.4.

4.3.2 Continuity of the limit function

Theorem 4.23. *Let $(u^n)_{n \in \mathbb{N}} \subset C^0(J \times \mathbb{R}^d)$ be the sequence of solutions of $(P[v^n])_{n \in \mathbb{N}}$. Concerning the limits*

$$U_* := \limsup_{n \rightarrow \infty}^* u^n \quad \text{and} \quad U^* := \liminf_{n \rightarrow \infty}^* u^n$$

we have $U_* = U^* =: u \in C^0(J \times \mathbb{R}^d)$.

Proof. Per definition we know that $U^* \leq U_*$. Thus, it remains to show that $U_* \leq U^*$. Again we divide the proof into smaller steps.

- *Step 1.* For a fixed $\varepsilon > 0$ let us consider the functions u_ε^n defined in Lemma 4.13 and

$$U_{*\varepsilon} := \limsup_{n \rightarrow \infty}^* u_\varepsilon^n.$$

We then have

$$U_{*\varepsilon}(t, x) = \limsup_{n \rightarrow \infty}^* \left(u^n - \frac{\varepsilon}{T-t} \right) = \limsup_{n \rightarrow \infty}^* u^n - \frac{\varepsilon}{T-t}$$

and therefore $U_{*\varepsilon} \nearrow U_*$ in $J \times \mathbb{R}^d$ with $\varepsilon \rightarrow 0$. So if we could prove $U_{*\varepsilon} \leq U^*$, we would in the limit $\varepsilon \rightarrow 0$ obtain the desired result. Moreover from Lemma 4.13 we know that $U_{*\varepsilon}$ solves $\underline{F}^* \leq -c(\varepsilon) < 0$.

- *Step 2.* We show $U_{*\varepsilon} \leq U^*$. Assume there exists a point (s, z) for which

$$0 < \delta := U_{*\varepsilon}(s, z) - U^*(s, z). \quad (4.12)$$

For $\alpha > 0$ fixed we take the test function $\phi(x, y) := \frac{\alpha}{4}|x - y|^4$ and consider the maximum point $(\hat{t}, \hat{x}, \hat{y})$ of

$$w(t, x, y) - \phi(x, y) := U_{*\varepsilon}(t, x) - U^*(t, y) - \phi(x, y).$$

Such a maximum point exists in view of the semi-continuity of U^* and $U_{*\varepsilon}$. And for α large enough this maximum point lies in the interior of the domain $(0, T) \times B(R(g)) \times B(r(g))$ due to the initial condition, $U_{*\varepsilon} \rightarrow -\infty$ for $t \rightarrow T$ and the bound (4.10).

We apply the parabolic Ishii-Lemma (Lemma 4.5, $k = 2$, $u_1 = U_{*\varepsilon}$, $u_2 = -U^*$, $Z = D^2\phi(\hat{x}, \hat{y})$, $p = \nabla\phi(\hat{x}, \hat{y})$) and get

$$(a, q, X) \in \mathcal{P}^{2,+} U_{*\varepsilon}(\hat{t}, \hat{x}) \quad \text{and} \quad (-b, -r, -Y) \in \mathcal{P}^{2,+}(-U^*)(\hat{t}, \hat{y})$$

such that again

$$\begin{aligned} a &= b = \partial_t \phi(\hat{x}, \hat{y}) = 0 \\ q &= r = \nabla_x \phi(\hat{x}, \hat{y}) = \alpha |\hat{x} - \hat{y}|^2 (\hat{x} - \hat{y}) \end{aligned} \quad (4.13)$$

and for the setting $\gamma = \|Z\|^{-1}$ in Lemma 4.5

$$-2\|Z\|\text{Id} \leq \begin{pmatrix} X & 0 \\ 0 & -Y \end{pmatrix} \leq Z + \frac{1}{\|Z\|} Z^2 \quad (4.14)$$

with

$$Z = D^2\phi(\hat{t}, \hat{x}, \hat{y}) = 3\alpha(\hat{x} - \hat{y}) \otimes (\hat{x} - \hat{y}) \begin{pmatrix} \text{Id} & -\text{Id} \\ -\text{Id} & \text{Id} \end{pmatrix}.$$

From (4.14) we conclude that for $\mu := 6\alpha\|(\hat{x} - \hat{y}) \otimes (\hat{x} - \hat{y})\|$

$$\|X\|, \|Y\| \leq \mu. \quad (4.15)$$

Using $\mathcal{P}^{2,+}(-U^*) = -\mathcal{P}^{2,-}U^*$, we have now $(a, q, X) \in \mathcal{P}^{2,+}U_{*\varepsilon}(\hat{t}, \hat{x})$ and $(b, r, Y) \in \mathcal{P}^{2,-}U^*(\hat{t}, \hat{y})$, which we approximate by sequences given from Lemma 4.22. Thus, we have

$$\begin{aligned} t_j &\rightarrow \hat{t}, & s_j &\rightarrow \hat{t}, \\ x_j &\rightarrow \hat{x}, & y_j &\rightarrow \hat{y}, \\ a_j &\rightarrow a, & b_j &\rightarrow b, \\ q_j &\rightarrow q, & r_j &\rightarrow r, \\ X_j &\rightarrow X, & Y_j &\rightarrow Y, \\ u_{\varepsilon^j}^{n_j}(t_j, x_j) &\rightarrow U_{*\varepsilon}(\hat{t}, \hat{x}), & u^{m_j}(s_j, z_j) &\rightarrow U^*(\hat{t}, \hat{y}). \end{aligned}$$

Obviously we cannot choose the same sequences to approximate the jets. We insert these sequences into the problems, which our candidates $U_{*\varepsilon}$ and U^* solve, to obtain (as before in the proof of Theorem 4.20)

$$\begin{aligned} \underline{F}^*(\hat{t}, \hat{x}, a, q, X) &\leq \liminf_{j \rightarrow \infty} (a_j + \underline{F}^{n_j}(t_j, x_j, q_j, X_j)) \leq -c(\varepsilon) \\ \overline{F}_*(\hat{t}, \hat{y}, b, r, Y) &\geq \limsup_{j \rightarrow \infty} (b_j + \overline{F}^{m_j}(s_j, y_j, r_j, Y_j)) \geq 0. \end{aligned}$$

Subtracting the two rightmost inequalities delivers

$$\limsup_{j \rightarrow \infty} (b_j + \overline{F}^{m_j}(s_j, y_j, r_j, Y_j)) - \liminf_{j \rightarrow \infty} (a_j + \underline{F}^{n_j}(t_j, x_j, q_j, X_j)) \geq c(\varepsilon) > 0$$

and thus by passing to sub-sequences (for which we do not change notation)

$$\lim_{j \rightarrow \infty} (b_j - a_j + \overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \underline{F}^{n_j}(t_j, x_j, q_j, X_j)) \geq c(\varepsilon) > 0. \quad (4.16)$$

Let us closer analyze the term inside the limes. Because we could not choose the same sub-sequences for the \underline{F}^* and the \overline{F}_* we have to add zeroes in the following form:

$$\begin{aligned} b_j - a_j + \overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \underline{F}^{n_j}(t_j, x_j, q_j, X_j) &= b_j - b + a - a_j \\ &\quad + \overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \overline{F}^\infty(s_j, y_j, r_j, Y) \quad (4.17) \end{aligned}$$

$$+ \overline{F}^\infty(s_j, y_j, r_j, Y) - \underline{F}^\infty(t_j, x_j, q_j, X) \quad (4.18)$$

$$+ \underline{F}^\infty(t_j, x_j, q_j, X) - \underline{F}^{n_j}(t_j, x_j, q_j, X_j). \quad (4.19)$$

We estimate these terms separately. Clearly $b_j - b + a - a_j$ goes to zero as $j \rightarrow \infty$. For the rest, we choose $\eta > 0$ arbitrary and using the notation

$$\Pi_\theta := \left(\text{Id} - \frac{\theta \otimes \theta}{|\theta|^2} \right)$$

for vectors $\theta \in \mathbb{R}^d \setminus \{0\}$ we make a distinction between the following cases:

Case 1: $p = q \neq 0$. Without loss of generality we can assume that $q_j \neq 0 \neq p_j$ for all j . We consider the first term (4.17) and have

$$\begin{aligned} \overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \overline{F}^\infty(s_j, y_j, r_j, Y) &= -\text{div}(A(v^{m_j})(s_j, y_j)) \cdot r_j + \text{div}(A(v)(s_j, y_j)) \cdot r_j \\ &\quad - \text{tr}(A(v^{m_j})(s_j, y_j)\Pi_{r_j}Y_j) + \text{tr}(A(v)(s_j, y_j)\Pi_{r_j}Y_j) \\ &\quad - \text{tr}(A(v)(s_j, y_j)\Pi_{r_j}Y_j) + \text{tr}(A(v)(s_j, y_j)\Pi_{r_j}Y) \\ &\leq \eta \end{aligned} \quad (4.20)$$

for all $j > j_1$ large enough. The same estimate holds with $j > j_2$ large enough for the third term (4.19). Let us now consider the middle term (4.17). For all $j > j_3$ large enough:

$$\begin{aligned} \overline{F}^\infty(s_j, y_j, r_j, Y) - \underline{F}^\infty(t_j, x_j, q_j, X) &= -\operatorname{div}(A(v)(s_j, y_j)) \cdot r_j + \operatorname{div}(A(v)(s_j, y_j)) \cdot r \\ &\quad + \operatorname{div}(A(v)(t_j, x_j)) \cdot q_j - \operatorname{div}(A(v)(t_j, x_j)) \cdot q \\ &\quad + \operatorname{div}(A(v)(t_j, x_j)) \cdot q - \operatorname{div}(A(v)(s_j, y_j)) \cdot r \\ &\quad + \operatorname{tr}(A(v)(s_j, y_j)\Pi_r Y) - \operatorname{tr}(A(v)(s_j, y_j)\Pi_r Y) \\ &\quad + \operatorname{tr}(A(v)(t_j, x_j)\Pi_{q_j} X) - \operatorname{tr}(A(v)(t_j, x_j)\Pi_{q_j} X) \\ &\quad + \operatorname{tr}(A(v)(\hat{t}, \hat{y})\Pi_r Y) - \operatorname{tr}(A(v)(s_j, y_j)\Pi_r Y) \\ &\quad + \operatorname{tr}(A(v)(t_j, x_j)\Pi_{q_j} X) - \operatorname{tr}(A(v)(\hat{t}, \hat{x})\Pi_{q_j} X) \\ &\quad + \operatorname{tr}(A(v)(\hat{t}, \hat{x})\Pi_{q_j} X) - \operatorname{tr}(A(v)(\hat{t}, \hat{y})\Pi_r Y) \\ &\leq \eta + \operatorname{tr}(A(v)(\hat{t}, \hat{x})\Pi_{q_j} X) - \operatorname{tr}(A(v)(\hat{t}, \hat{y})\Pi_r Y). \end{aligned}$$

But we can choose α large so that $|\hat{x} - \hat{y}| \leq \eta$ according to Lemma 4.6 and thus $\|A(v)(\hat{t}, \hat{x}) - A(v)(\hat{t}, \hat{y})\| \leq C\eta$ and finally

$$\begin{aligned} \operatorname{tr}(A(v)(\hat{t}, \hat{x})\Pi_{q_j} X) - \operatorname{tr}(A(v)(\hat{t}, \hat{y})\Pi_r Y) \\ = \operatorname{tr}(A(v)(\hat{t}, \hat{y})\Pi_{q_j}(X - Y)) + \operatorname{tr}(\Pi_{q_j}(A(v)(\hat{t}, \hat{x}) - A(v)(\hat{t}, \hat{y}))X) \leq C\eta \end{aligned}$$

because $X \leq Y$.

Case 2: Let us now assume that $q = r = 0$. If $q_j \neq 0 \neq r_j$ we are back in the first case so let us now first assume $q_j = r_j = 0$. Then obviously for term (4.17) we have for $j > j_4$ large enough

$$\overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \overline{F}^\infty(s_j, y_j, r_j, Y) = d\Lambda(\|Y\| - \|Y_j\|) \leq \eta. \quad (4.21)$$

And for the middle term (4.18) we use $X \leq Y$

$$\overline{F}^\infty(s_j, y_j, r_j, Y) - \underline{F}^\infty(t_j, x_j, q_j, X) = d\Lambda(\|X\| - \|Y\|) \leq 0 \leq \eta.$$

Finally we assume $q_j \neq 0 = r_j$ (the case $q_j = 0 \neq r_j$ is treated equivalently). Then for (4.17) we can use the estimate (4.21) and for (4.19) the estimate (4.20) from above. It remains to estimate (4.18). We use the information $\|X\|, \|Y\| \leq C\alpha|\hat{x} - \hat{y}|^2$ from the Ishii Lemma (estimate (4.15)) to conclude that if $r = q = \alpha|\hat{x} - \hat{y}|^2(\hat{x} - \hat{y}) = 0$ then $\|X\| = \|Y\| = 0$ and thus

$$\overline{F}^\infty(s_j, y_j, r_j, Y) - \underline{F}^\infty(t_j, x_j, q_j, X) = \operatorname{div}(A(v)(t_j, x_j)) \cdot q_j \leq \|\operatorname{div}A(v)\|_{L^\infty(Q)} |q_j| \leq \eta.$$

Summarizing the last results, we find for arbitrary $\eta > 0$ a fixed constant α large enough and $j > \max\{j_1, j_2, j_3, j_4, j_5, j_6, j_7\}$ such that

$$\overline{F}^{m_j}(s_j, y_j, r_j, Y_j) - \underline{F}^{m_j}(t_j, x_j, q_j, X_j) < \eta.$$

But this is a contradiction to (4.16) for $\eta < c(\varepsilon)$. □

4.3.3 Uniform convergence toward the $\Gamma^{+/-}$ -limit

In the last paragraph we have shown that the limit function u is continuous. We now conclude that the u^n converge uniformly toward u . Then obviously u fulfills the initial condition of $(P[v])$.

Lemma 4.24. *The functions u^n converge uniformly toward the $\Gamma^{+/-}$ -limit u .*

Proof. Because according to (4.10) the functions u^n have a compact support in $B(R_1)$ independently of n , it suffices to show the uniform convergence on the bounded set $J \times B(R_1)$: Assume the convergence were not uniform, i.e. there exists an $\varepsilon > 0$ and a sub-sequence $n_j \rightarrow \infty$, $\overline{J \times B(R_1)} \ni (s_j, x_j) \rightarrow (s, x) \in \overline{J \times B(R_1)}$ such that

$$u^{n_j}(s_j, x_j) - u(s_j, x_j) > \varepsilon \quad \text{or} \quad u^{n_j}(s_j, x_j) - u(s_j, x_j) < -\varepsilon.$$

We then would derive

$$\begin{array}{l} \varepsilon \leq \limsup_{j \rightarrow \infty} (u^{n_j}(s_j, x_j) - u(s_j, x_j)) \quad \text{or} \quad \varepsilon \leq \liminf_{j \rightarrow \infty} (u(s_j, x_j) - u^{n_j}(s_j, x_j)) \\ \leq \limsup_{j \rightarrow \infty}^* (u^{n_j}(s, x) - u(s, x)) \quad \left| \quad \leq \liminf_{j \rightarrow \infty}^* (u(s, x) - u^{n_j}(s, x)) \right. \\ = \limsup_{j \rightarrow \infty}^* u^{n_j}(s, x) - u(s, x) \quad \left| \quad = U(s, x) - \liminf_{j \rightarrow \infty}^* u^{n_j}(s, x) \right. \\ = u(s, x) - u(s, x) = 0 \quad \left| \quad = u(s, x) - u(s, x) = 0 \right. \end{array}$$

where we have used the continuity of u . But this is a contradiction and thus the convergence uniform. \square

A trivial consequence of the uniform convergence is that u fulfills the initial condition

$$u^n(0, x) = g(x) \quad \Rightarrow \quad u(0, x) = g(x).$$

Finally u has compact support as it is claimed by Theorem 4.17, since $u^n \in C_0^0(J \times \mathbb{R}^d)$.

4.3.4 Identification of the limit problems

To finish the compactness result it remains the identification of the limit problems. So far we know that u solves $\underline{F}^* \leq 0$ and $\overline{F}^* \geq 0$ with the appropriate initial condition. Thus, it remains to show that

$$\underline{F}^*(t, x, a, p, X) = a + \underline{F}^\infty(t, x, p, X) \quad \text{and} \quad \overline{F}^*(t, x, a, p, X) = a + \overline{F}^\infty(t, x, p, X).$$

Let us recall the definition:

$$\begin{aligned} \underline{F}^*(t, x, a, p, X) &= a + \liminf_{n \rightarrow \infty}^* \underline{F}^n(t, x, p, X) \\ &= a + \liminf_{n \rightarrow \infty} \left\{ \underline{F}^l(s, y, q, Y) : l \geq n, |s - t|, |y - x|, |q - p|, \|Y - X\| \leq \frac{1}{n} \right\} \end{aligned} \quad (4.22)$$

Assume that $p = 0$. Due to the lower estimate (4.4) we know the infimum in (4.22) is reached in $\underline{F}^n(s, y, 0, Y)$ independently of s, y , and Y , thus obviously

$$\underline{F}^*(t, x, a, p, X) = a + \liminf_{n \rightarrow \infty} d\Lambda\|X\| = a + d\Lambda\|X\| = a + \underline{F}^\infty(t, x, 0, X).$$

So let us now assume $p \neq 0$. For n large enough we have $q \neq 0$ for all $|q - p| \leq \frac{1}{n}$, and \underline{F}^n is continuous on $J \times \mathbb{R}^d \times \mathbb{R}^d \setminus \{0\} \times S(d)$. Moreover since $A(v^n) \rightrightarrows A(v)$ and $\text{div}A(v^n) \rightrightarrows \text{div}A(v)$ we can easily pass to the limit to obtain

$$\begin{aligned} \liminf_{n \rightarrow \infty}^* \underline{F}^n(t, x, p, X) &= \liminf_{n \rightarrow \infty} \left\{ \underline{F}^l(s, y, q, Y) : l \geq n, |s - t|, |y - x|, \right. \\ &\quad \left. |q - p|, \|Y - X\| \leq \frac{1}{n} \right\} \\ &= -\text{div}(A(v)(t, x)) \cdot p + \text{tr}(A(v)(t, x)\Pi X) \\ &= \underline{F}^\infty(t, x, p, X), \end{aligned}$$

because the functions converge locally uniformly (cf. [11]) for $p \neq 0$. Analogously we ensure the identification of the upper semi-continuous limit problem $\overline{F}^* = \overline{F}[v]$.

4.4 Existence of a solution of the nonlinear problem

With the tools from the last section we are now able to prove the existence of a solution of the nonlinear problem:

Theorem 4.25. *For the problems (2.3) and (2.17) there exists a solution $u : J \times \Omega \rightarrow \mathbb{R}$ in the sense that for given initial condition $g \in C_0^0(\mathbb{R}^d)$ the function u is defined as the viscosity solution of $P[u]$. Moreover $u \in C_0^0(J \times \Omega)$ and the following estimate holds:*

$$\|u\|_{L^\infty(J \times \mathbb{R}^d)} \leq \|g\|_{L^\infty(\mathbb{R}^d)}.$$

Proof. We consider functions $v \in C_0^0(J \times \mathbb{R}^d)$ such that $\|v\|_{L^\infty(J \times \mathbb{R}^d)} \leq \|g\|_{L^\infty(\mathbb{R}^d)}$ and thus

$$v \in B := \{w \in C_0^0(J \times \mathbb{R}^d) : \|w\|_{L^\infty(J \times \mathbb{R}^d)} \leq \|g\|_{L^\infty(\mathbb{R}^d)}\} \subset C_0^0(J \times \mathbb{R}^d).$$

Clearly B is a convex subset of $C_0^0(J \times \mathbb{R}^d)$ and due to the estimate (4.9) we can interpret $P[\cdot]$ as a mapping

$$P : B \rightarrow B.$$

Moreover we know from Theorem 4.17 that P is a continuous operator, because for a convergent sequence $w^n \rightarrow w$ in $C^0(J \times \mathbb{R}^d)$ we have $w^n \rightarrow w$ in $L^2(J \times \mathbb{R}^d)$. And the corresponding solutions u^n again converge uniformly toward u .

Thus, the assumptions for the application of the Schauder Fixed-Point-Theorem are fulfilled, which finally gives us the existence of a fixed-point $u \in B$ of the continuous and compact operator $P[\cdot]$. \square

Chapter 5

Discretization and implementation

IN THIS CHAPTER the focus lies on the discretization and the implementation of the previously defined models. First we will consider the spatial discretization of the image domain Ω and the image-sequence domain \mathcal{Q} . After a discussion of the discrete variants of the involved geometric quantities, we deal with the scale-discretization and derive linear systems of equations via the standard continuous Galerkin method. Thereby for the image-sequence evolution a detailed analysis with suitable integration schemes leads to a simple block-system of equations, where each block corresponds to an image in the sequence. The resulting system is solved by a symmetric block Gauß-Seidel solver.

Throughout the chapter we make use of multi-indices, when denoting grid-elements and -nodes. For an index set Y we denote the components of a multi-index $i \in Y^d$ by (i_1, \dots, i_d) . The multiplication of i with a scalar λ is defined by $\lambda i := (\lambda i_1, \dots, \lambda i_d)$. Furthermore for open sets \mathcal{O} we denote by $L^p(\mathcal{O})$ the Lebesgue space of exponent $p \in [1, \infty]$ on \mathcal{O} and by $H^{m,2}(\mathcal{O})$ or just $H^m(\mathcal{O})$ the Sobolev spaces for $m \in \mathbb{N}$. For the dual space of $H^1(\mathcal{O})$ we write $(H^1)'(\mathcal{O})$. The dual pairing on $(H^1)'(\mathcal{O}) \times H^1(\mathcal{O})$ is denoted with $\langle \cdot, \cdot \rangle$. Moreover $(\cdot, \cdot)_{\mathcal{O}}$ is the scalar product on $L^2(\mathcal{O})$ and the norm on $L^p(\mathcal{O})$ is denoted with $\|\cdot\|_{L^p(\mathcal{O})}$.

5.1 Finite-Elements on uniform spatial grids

To begin with, let us focus on the spatial discretization and introduce suitable triangulations on Ω . Naturally, we assume that $\Omega = [0, 1]^d$, because images are usually given on rectangular areas. Obviously, for image processing purposes it is most efficient to discretize the domain Ω with a regular quadrilateral or hexahedral grids, where the unknowns on the grid correspond to the given pixels or voxels of the image.

Definition 5.1. For given $l_{\max} \in \mathbb{N}$ a Quadtree-grid (Octtree-grid) \mathcal{T}^* of $\Omega := [0, 1]^d$, for $d = 2$ ($d = 3$) is the set

$$\mathcal{T}^* := \left\{ E_i^l : l \in \{0, \dots, l_{\max}\}, i \in \{0, \dots, 2^l - 1\}^d \right\}$$

with the elements, which carry a multi-index subscript i ,

$$\Omega \supset E_i^l := \begin{cases} [i_1 2^{-l}, (i_1 + 1) 2^{-l}] \times [i_2 2^{-l}, (i_2 + 1) 2^{-l}] & \text{if } d = 2, \\ [i_1 2^{-l}, (i_1 + 1) 2^{-l}] \times [i_2 2^{-l}, (i_2 + 1) 2^{-l}] \times [i_3 2^{-l}, (i_3 + 1) 2^{-l}] & \text{if } d = 3. \end{cases}$$

We call l_{\max} the depth of the Quadtree (Octtree), l the level of the element E_i^l and i its index. For a multi-index $\alpha \in \{0, \dots, 2^l\}^d$ we denote the set of the Nodes

$$\Omega \ni x_\alpha^l := 2^{-l} \alpha = \begin{cases} (\alpha_1 2^{-l}, \alpha_2 2^{-l})^T & \text{if } d = 2, \\ (\alpha_1 2^{-l}, \alpha_2 2^{-l}, \alpha_3 2^{-l})^T & \text{if } d = 3, \end{cases}$$

on level l by

$$\Omega \supset \mathcal{N}^l := \left\{ x_\alpha^l : \alpha \in \{0, \dots, 2^l\}^d \right\}.$$

For the quadtree- or octtree-grid of Ω we have the following properties:

- For a fixed $0 \leq l_0 \leq l_{\max}$ the elements of level l_0 cover Ω

$$\bigcup_{i \in \{0, \dots, 2^{l_0-1}\}^d} E_i^{l_0} = \Omega.$$

- For a fixed $0 \leq l_0 \leq l_{\max}$ and $i \neq j$

$$E_i^{l_0} \cap E_j^{l_0} = \begin{cases} \emptyset, \\ \text{common node } x_\alpha^{l_0}, \\ \text{common edge } (d=2, 3) \text{ or common face } (d=3). \end{cases}$$

- Each element E_i^l on level $0 \leq l < l_{\max}$ has 2^d child-elements on level $l+1$

$$E_i^l = \bigcup_{j \in \{1, \dots, 2^d\}} E_j^{l+1}, \quad \text{with } i_j \in \{0, \dots, 2^{l+1} - 1\}^d.$$

Per definition quadtree- and octtree-grids induce a hierarchy of grids. This allows for the efficient handling of multi-grid as well as adaptivity [66]. However, adaptivity has so far not been implemented for the evolution problems we consider here, thus, we will only use the finest level l_{\max} of a grid in the following. Consequently the triangulation \mathcal{T}^h given by

$$\mathcal{T}^* \supset \mathcal{T}^h := \left\{ E_i := E_i^{l_{\max}} : i \in \{0, \dots, N-1\}^d \right\} \quad \text{with } N := 2^{l_{\max}}$$

is the base for our discretization. The super-index h denotes here the *grid-width* $h = 2^{-l_{\max}}$. Below we omit the superscript l for the elements E_i and the nodes x_α .

Obviously, the nodes of the quadtree and octtree grids cannot carry pixel data for arbitrary image dimensions. We assume that the resolution of the given image data is $2^{l_{\max}} + 1$, $l_{\max} \in \mathbb{N}$ in each space-dimension, i.e. the images consist of $(2^{l_{\max}} + 1)^d$ pixels or voxels. This is not a severe restriction, since an image that does not fit into this size can be enlarged e.g. by mirroring it at its boundaries.

Now we introduce a finite dimensional space \mathcal{V}^h on \mathcal{T}^h as a subspace of $H^1(\Omega)$

$$\mathcal{V}^h := \{v \in H^1(\Omega) : v|_{E_i} \text{ is bilinear (d=2) or trilinear (d=3) for all } E_i \subset \mathcal{T}^h\},$$

on which the H^1 norm is induced. Functions $u \in H^2$ can be approximated by functions $u_h \in \mathcal{V}^h$ and the approximation error between u and the approximation u_h can be measured by [10, 12]

$$\|u - u_h\|_{H^m(\Omega)} \leq Ch^{2-m} \|u\|_{H^2(\Omega)} \quad \text{for } 0 \leq m \leq 2. \quad (5.1)$$

A basis of \mathcal{V}^h is given by the so called *hat-functions*

$$\left\{ \psi_\alpha \in \mathcal{V}^h : \psi_\alpha(x_\beta) = \delta_{\alpha\beta}, x_\beta \in \mathcal{N}^{l_{\max}} \right\}_{\alpha \in \{0, \dots, 2^{l_{\max}}\}^d} \quad \text{with } \delta_{\alpha\beta} = \begin{cases} 1 & \text{if } \alpha = \beta, \\ 0 & \text{else,} \end{cases}$$

and as a consequence each function $u_h \in \mathcal{V}^h$ can be written as

$$u_h(x) = \sum_{\alpha} (U_h)_\alpha \psi_\alpha(x). \quad (5.2)$$

The vector $U_h \in \mathbb{R}^{d(N+1)}$ of the *nodal values* $u_h(x_\alpha)$ contains the degrees of freedom of the approximation u_h . Here and in the following we will denote discrete quantities by upper case letters and discrete functions will carry a subscript h if an explicit distinction is necessary.

5.2 Tensor product finite-elements in sequence-time and space

We continue with the introduction of the sequence-time discretization. We assume that the frames have a uniform distance $\rho > 0$ in sequence-time. Recalling $I = [0, T]$, we define $M := T/\rho$ and introduce the intervals

$$I_j := [j\rho, (j+1)\rho] \quad \text{for } j = 0, \dots, M-1,$$

and the corresponding temporal nodes

$$s_j := j\rho \quad \text{for } j = 0, \dots, M.$$

As in the spatial case, we define a finite dimensional subspace \mathcal{W}^ρ of $H^1(I)$ by

$$\mathcal{W}^\rho := \{w_\rho \in H^1(I) : w_\rho|_{I_j} \text{ is linear for } j = 0, \dots, M-1\}.$$

Again a basis is given with the hat-functions

$$\{\phi_i \in \mathcal{W}^\rho : \phi_i(s_j) = \delta_{ij}, j = 0, \dots, M\}_{i=0, \dots, M}$$

and the above approximation estimates (5.1) hold.

We combine now the finite elements in space and the ones in sequence-time to discretize image-sequences. For this purpose, we consider the tensor product space

$$\mathcal{W}^\rho \otimes \mathcal{V}^h,$$

for which a basis is given by tensor products of the hat-functions, i.e.

$$\mathcal{W}^\rho \otimes \mathcal{V}^h = \text{span} \left\{ \phi_i \psi_\alpha : i = 0, \dots, M, \alpha \in \{0, \dots, N\}^d \right\}.$$

Thus, each discrete image-sequence function $u_{\rho,h}(s, x) \in \mathcal{W}^\rho \otimes \mathcal{V}^h$ can be expressed as

$$u_{\rho,h}(s, x) = \sum_i \sum_\alpha (U_{\rho,h})_{i,\alpha} \phi_i(s) \psi_\alpha(x). \quad (5.3)$$

The vector $U_{\rho,h} \in \mathbb{R}^{(M+1)d(N+1)}$ contains the nodal values $u_{\rho,h}(s_i, x_\alpha)$ of the approximation $u_{\rho,h}$. To distinguish between temporal and spatial nodes, we will denote the temporal nodes with Latin indices, and the spatial nodes with Greek indices.

5.3 Geometric quantities on discrete data

In Chapter 3 we have already introduced various methods for the regularization of the image data, which lead to the definition of regularized geometric quantities involving higher order derivatives of the image. Let us now turn to the actual implementation of these regularization approaches and again pick up the two promising variants (R3) and (R4).

In general we will compute geometric quantities on nodes and not on elements, although element values are needed for the final integration of the local matrices. In case of the convolution regularization an element-wise evaluation would not be as efficient as the nodal approach. Moreover for further post-processing nodal values are more convenient.

5.3.1 Discrete shape-operator on discrete data

Let us first consider the static-image evolution model (2.3), whose diffusion tensor $a(\Sigma)$ takes into account an extended regularized shape operator Σ .

Variant (R3): Convolution with derivatives of C_0^∞ kernels

We have to evaluate the shape operator according to the formula given in (3.7), which involves the convolution of the image with derivatives of the smoothing kernel K^σ defined in (3.3). We will only discuss how this convolution is computed, since the rest of the task is just to substitute the convolutions into (3.7).

Now let us fix a point $x \in \Omega$. The kernel is designed such that the kernel itself and also its derivatives $D^\gamma K^\sigma$ have the same compact support, so we can replace the integration on Ω by the integration on the support of K^σ . Naturally we should take into account the uniform structure of the quadtree or octtree. Thus, we define the neighborhood \mathcal{B}_σ in the maximum norm. In particular, we set $l \in \{2m : m \in \mathbb{N}\}$ such that

$$\text{supp } D^\gamma K^\sigma \subset \mathcal{B}_\sigma := [-lh, +lh] \times \cdots \times [-lh, +lh].$$

Setting $\sigma = \text{diam } B_\sigma(x) = 2lh$ we obtain a relation between the regularization parameter σ and the discrete neighborhood. For the integration we enlarge the latter neighborhood slightly (cf. Figure 5.1), to define appropriate nodal weights below. In particular we define the neighborhood $\tilde{\mathcal{B}}_\sigma$ used for the numerical integration as

$$\tilde{\mathcal{B}}_\sigma := [-(l+1/2)h, (l+1/2)h] \times \cdots \times [-(l+1/2)h, (l+1/2)h] = \bigcup_i \tilde{E}_i$$

such that still $\text{supp } D^\gamma K^\sigma \subset \tilde{\mathcal{B}}_\sigma$, but the integration is performed on the *dual cells* \tilde{E}_i . These dual cells \tilde{E}_i for a multi-index $i \in \{-l, \dots, l\}^d$ are given by

$$\tilde{E}_i := [(i_1 - 1/2)h, (i_1 + 1/2)h] \times \cdots \times [(i_d - 1/2)h, (i_d + 1/2)h].$$

Consequently we can write

$$(D^\gamma K^\sigma * u_h)(x) = \int_{\tilde{\mathcal{B}}_\sigma} D^\gamma K^\sigma(y) u_h(x-y) dy \approx \sum_{\tilde{E}_i \subset \tilde{\mathcal{B}}_\sigma} u_h(x-x_i) \int_{\tilde{E}_i} D^\gamma K^\sigma(y) dy,$$

where $x_i \in \mathcal{N}$ is the node that is located at the center of the dual cell \tilde{E}_i . Clearly, the integral in this sum can be precomputed for fixed choices of σ . A lookup table containing the weights

$$\omega_i := \int_{\tilde{E}_i} D^\gamma K^\sigma(y) dy$$

can be used to replace the convolution with a weighted sum over the nodes $x_i \in (B_\sigma + x)$, i.e.

$$(D^\gamma K^\sigma * u_h)(x) = \sum_{x_i \in (B_\sigma + x)} u_h(x_i) \omega_i.$$

Variant (R4): Local L^2 -projections

For the local projection variant, we start with the formula given in (3.8) to compute the shape operator. Since we treat a purely spatial projection for the computation of the shape operator, we again fix a point $x \in \Omega$. A translation does not affect the local projection and for the evaluation of derivatives the subtraction of a constant does not matter. Therefore we can assume without loss of generality that $x = 0$ and moreover $u_h(x) = 0$.

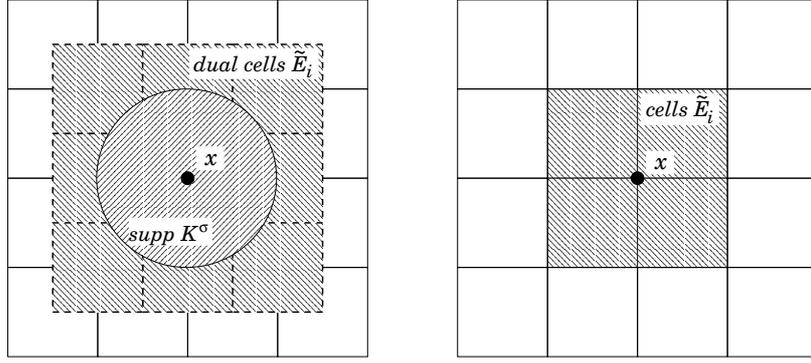


Figure 5.1: *Left: The convolution of the image data with the compactly supported kernel is performed on the dual cells \tilde{E}_i (shaded rectangles) which cover the support of the kernel K^σ (shaded disc). This allows the definition of suitable nodal integration weights. Right: The local projection of the image data onto a polynomial space is performed on the elements E_i (shaded rectangles), which constitute the neighborhood \mathcal{B}_σ .*

We do not work with the dual cells mentioned above, but consider a patch of elements around x (cf. Figure 5.1). As before we select an even $l \in \{2m : m \in \mathbb{N}\}$ and define

$$\mathcal{B}_\sigma := [-lh, +lh] \times \cdots \times [-lh, +lh] = \bigcup_i E_i,$$

i.e. \mathcal{B}_σ is the patch consisting of $(2l)^d$ elements having $x = 0$ at its center. If we set $\sigma = \text{diam } \mathcal{B}_\sigma$ we obtain $\sigma = 2lh$ for the regularization parameter.

The assumption $u_h(x) = 0$ leads to a projection space $\mathcal{Q} \subset \mathcal{Q}^*$, which does not contain the constant functions any more. In particular for the computations we consider the subspaces

$$\mathcal{Q} := \begin{cases} \text{span}\{y_1^2, y_2^2, y_1 y_2, y_1, y_2\} & \text{if } d = 2, \\ \text{span}\{y_1^2, y_2^2, y_3^2, y_1 y_2, y_1 y_3, y_2 y_3, y_1, y_2, y_3\} & \text{if } d = 3, \end{cases}$$

of \mathcal{P}_2 , which do not contain the constant monomial. Any $u_x^\sigma \in \mathcal{Q}$ can then be written as a linear combination

$$u_x^\sigma(y) := \sum_{i=1}^{\dim \mathcal{Q}} b_i q_i(y),$$

in the above canonical basis, where the dimension of \mathcal{Q} is 5 or 9. If u_x^σ is the local projection, the vector of coefficients $(b_i)_i \in \mathbb{R}^{\dim \mathcal{Q}}$ solves the system (cf. equation (3.5)):

$$\sum_{j=1}^{\dim \mathcal{Q}} \left(\int_{\mathcal{B}_\sigma(x)} (q_i q_j)(y) dy \right) b_j = \int_{\mathcal{B}_\sigma(x)} u_h(y) q_i(y) dy \quad \text{for } i = 1, \dots, \dim \mathcal{Q}. \quad (5.4)$$

As before we pre-compute the inverse of the matrix

$$H := \left(\int_{\mathcal{B}_\sigma(x)} (q_i q_j)(y) dy \right)_{i,j}.$$

From the given coefficients b_i of the projection we can now easily derive the expressions for the geometric quantities we are interested in. In two dimensions ($d = 2$) we have

$$\nabla_y u_x^\sigma(y) \Big|_{y=x=0} = \begin{pmatrix} b_4 \\ b_5 \end{pmatrix}, \quad D_y^2 u_x^\sigma(y) \Big|_{y=x=0} = \begin{pmatrix} 2b_1 & b_3 \\ b_3 & 2b_2 \end{pmatrix}$$

l	diam \mathcal{B}_σ	(R3)	(R4)
1	$2h$	3 sec	179 sec
2	$4h$	10 sec	622 sec
3	$6h$	52 sec	4912 sec

Table 5.1: *Computational cost for the evaluation of apparent velocities via regularization variant (R3) and (R4) for different σ and a fixed problem size of $129 \times 129 \times 129$. The execution times were measured on an Intel P4, 1.7 GHz PC.*

and in the three-dimensional case ($d = 3$)

$$\nabla_y u_x^\sigma(y) \Big|_{y=x=0} = \begin{pmatrix} b_7 \\ b_8 \\ b_9 \end{pmatrix}, \quad D_y^2 u_x^\sigma(y) \Big|_{y=x=0} = \begin{pmatrix} 2b_1 & b_4 & b_5 \\ b_4 & 2b_2 & b_6 \\ b_5 & b_6 & 2b_3 \end{pmatrix}.$$

These expressions can be substituted into (3.8), to compute N^σ and Σ .

It remains to comment on the quadrature rules, which are applied for the computation of the right hand side of (5.4). Since the basis functions $q_i(y)$ are of second order at most and the image function $u_h(y)$ is bi-/trilinear on an element E_i , a quadrature rule for the integration of third order polynomials should be used. We employ a tensor product of a two point quadrature rule, thus using 2^d interpolated values on an element E to compute the integral on E . This ensures an exact integration of the right-hand-side of (5.4).

Finally, in Table 5.1 we have compared the computation times for the evaluation of apparent velocities for different σ via regularization variant (R3) and (R4). In Chapter 3 we have noted that variant (R4) behaves much more robust in the presence of noise, but as we see from the table the computational price is very high. This is due to the computation of the right hand side of (5.4), which involves many local 2^d -point integrations on the elements — in contrast to the convolution approach which just computes weighted sums on nodal values without any interpolations. This can be handled very efficiently on state-of-the-art processors, e.g. Intel P4. A possible improvement could be to use inexact quadrature rules for the integration in (5.4), but this is not considered here.

5.3.2 Apparent velocity and apparent acceleration on discrete data

So far we have considered the discrete evaluation of the regularized normal N^σ and the extended regularized shape operator Σ for static images. The discretization of the regularization approaches does not change much for image-sequences.

Variant (R3): Convolution with derivatives of C_0^∞ kernels

Here we take the kernel defined in (3.4), which has compact support in space and sequence-time, and evaluate the apparent velocity v_{app}^σ and acceleration $\text{accel}_{\text{app}}^\sigma$ according to (3.9). With the given equidistant sequence-time stepping, we consider dual cells $\tilde{I}_j \times \tilde{E}_i$ in space and sequence-time, where

$$\tilde{I}_j := [(j - 1/2)\rho, (j + 1/2)\rho],$$

and pre-compute the weights

$$\omega_{j,i} = \int_{\tilde{I}_j \times \tilde{E}_i} D^\gamma K^\sigma(s, x),$$

where the multi-index γ stands for sequence-time derivatives as well as spatial derivatives. Consequently the spatio-temporal convolution can be replaced by a weighted sum over the frames and the pixels or voxels within the frames.

Variant (R4): Local L^2 -projections

As already described earlier, the computation of the local L^2 -projections is even more expensive than in the purely spatial case and therefore not applicable to the computation of the acceleration. Indeed, if we consider the projection spaces from (3.10) we notice that in three-dimensions ($d = 3$)

$$\begin{aligned}\dim(P_1(I) \otimes P_2(\Omega)) &= 20, \\ \dim(P_2(I) \otimes P_3(\Omega)) &= 51.\end{aligned}$$

Not all monomials of the canonical bases are needed to compute v_{app}^σ and $\text{accel}_{\text{app}}^\sigma$. For example the projection can be restricted to a 13-dimensional subspace to compute the apparent velocity, because only the derivatives of these 13 monomials define v_{app} as an easy calculation shows. However, the integrations require enormous costs that do not justify the use of this approach for the spatio-temporal computations. As a compromise for the computation of v_{app}^σ we restrict the diameter of \mathcal{B}_σ in sequence-time direction to contain only three frames. To be more precise, let us fix a point (s, x) in \mathcal{Q} . For an even $l \in \{2m : m \in \mathbb{N}\}$ we define

$$\mathcal{B}_\sigma := [s - \rho, s + \rho] \times [x_1 - lh, x_1 + lh] \times \cdots \times [x_d - lh, x_d + lh] = \bigcup_{i,j} I_j \times E_i,$$

to be the patch containing $(2l)^d$ elements and having (s, x) at its center. As before the projection $u_{(s,x)}^\sigma(y)$ is now given by the solution of the system (5.4) or (3.6), respectively, which delivers vectors $(b_i)_i \in \mathbb{R}^{\dim \mathcal{Q}}$. Finally b can be substituted into the equation (3.11) to compute the apparent velocity.

For the integration of the right hand side of the system, in direction of the sequence-time a trapezoidal rule is sufficient, because the sequence-time basis-functions are only linear. In the space dimension again a tensor product 2^d -point integration rule is applied.

5.4 Optimal regularization-parameters

In this section we discuss the consistency of the regularizations from a theoretical point of view. We consider only a convolution regularization variant here, since we expect the L^2 projection onto polynomials (R4) to behave similar concerning the choice of h and σ . The smoothing kernel used for the regularization was defined by (3.3), but for the subsequent analysis, we consider the scaled kernel $\tilde{K}(y)$ for $y = x/\sigma$, which has support in $B_1(0)$:

$$\tilde{K}^\sigma(y) := \begin{cases} \frac{1}{M} \exp\left(\frac{|y|^2}{|y|^2 - 1}\right) & \text{if } |y|^2 \leq \sigma^2, \\ 0 & \text{else.} \end{cases}$$

We intend to estimate the difference between the regularized discrete geometric quantities and their true geometric counterpart. For this purpose we assume that the image is contained in the Sobolev space $H^2(\Omega)$. For the sake of simplicity, we derive an estimate for the second derivatives of u instead of deriving an estimate for the shape operator:

$$\|D^2(\tilde{K}^\sigma * u_h) - D^2u\|_{L^2(\Omega)} \leq \|D^2\tilde{K}^\sigma * (u_h - u)\|_{L^2(\Omega)} + \|\tilde{K}^\sigma * D^2u - D^2u\|_{L^2(\Omega)}. \quad (5.5)$$

We estimate these terms separately. For the first term we apply the approximation inequality (5.1) and obtain

$$\|D^2\tilde{K}^\sigma * (u_h - u)\|_{L^2(\Omega)} \leq \|D^2\tilde{K}^\sigma\|_{L^\infty(\Omega)} \|(u_h - u)\|_{L^2(\Omega)} \leq C \frac{h^2}{\sigma^2} \|u\|_{H^2(\Omega)},$$

because for the derivatives of \tilde{K}^σ we have for $|y|^2 \leq \sigma^2$

$$\frac{\partial}{\partial x_i} \tilde{K}^\sigma(y) = \frac{\frac{2}{\sigma} y_i (|y|^2 - 1) - |y|^2 \frac{2}{\sigma} y_i}{(|y|^2 - 1)^2} \exp\left(\frac{|y|^2}{|y|^2 - 1}\right) = \frac{2y_i}{\sigma(|y|^2 - 1)^2} \exp\left(\frac{|y|^2}{|y|^2 - 1}\right)$$

and for $|y|^2 \leq \sigma^2$ as well

$$\frac{\partial}{\partial x_j} \frac{\partial}{\partial x_i} \tilde{K}^\sigma(y) = \left(\frac{4y_i y_j}{\sigma^2 (y^2 - 1)^4} + 2 \frac{(\delta_{ij} - 4y_i y_j)(y^2 - 1)^2}{\sigma^2 (y^2 - 1)^4} \right) \exp\left(\frac{|y|^2}{|y|^2 - 1}\right).$$

This yields

$$\|D^2 K\|_{L^\infty(\Omega)} \leq \frac{1}{\sigma^2} C$$

for our estimate above. The second term in (5.5) is estimated using standard results for convolutions with Gaussian kernels [1]

$$\|K * D^2 u - D^2 u\|_{L^2(\Omega)} \leq C \sigma \|D^2 u\|_{L^2(\Omega)}.$$

Altogether we have

$$\|D^2(K * u_h) - D^2 u\|_{L^2(\Omega)} \leq C \left(\frac{h^2}{\sigma^2} + \sigma \right) \|u\|_{H^2(\Omega)},$$

in which we would like to balance the term in parentheses. Assuming $\sigma = h^\gamma$ and thus for the balance

$$h^{2-2\gamma} = h^\gamma$$

we obtain an optimal regularization parameter with the choice $\sigma = h^{2/3}$. We conclude that for decreasing h the regularization parameter must be increased.

For non-smooth images $u \notin H^2(\Omega)$ the above estimate is non-realistic, but for computations the size of the stencil \mathcal{B}_σ has to be fixed somehow. In the applications shown in this thesis the value $l = 2$ (i.e. $\sigma = 4h$) is suitable if not described different. We emphasize that the actual choice of l is of minor importance, since the qualitative behavior of the method is nearly invariant under changes of σ as Figure 5.2 illustrates. The method only needs values indicating high curvatures for the classification of edges. The real curvatures of the regularized surface are not important, since they are not a descriptor of the optimal limit shape.

5.5 Discretization of the evolution of static images

So far we have shown how to obtain the geometric quantities consistently in a discrete setting, we can proceed with the discretization of the evolution models themselves. We start with the variational form of the problem and continue to derive the linear system of equations, which needs to be solved.

We still have the problem of defining a normal in case the gradient of the image ∇u vanishes. To prove existence of solutions for a linear anisotropic problem, we had overcome this singularity earlier by a relaxation of the problems (cf. Section 4) defining a semi-continuous continuation into the point $\nabla u = 0$. Here we follow the method presented by Evans and Spruck in [32], who replaced the Euclidean norm $|\nabla u|$ with

$$|\nabla u|_\varepsilon := \sqrt{\varepsilon^2 + |\nabla u|^2}$$

for a small $\varepsilon > 0$ (cf. Section 3.3). In the computations we set $\varepsilon \approx h$ as proposed by Deckelnick and Dziuk [23].

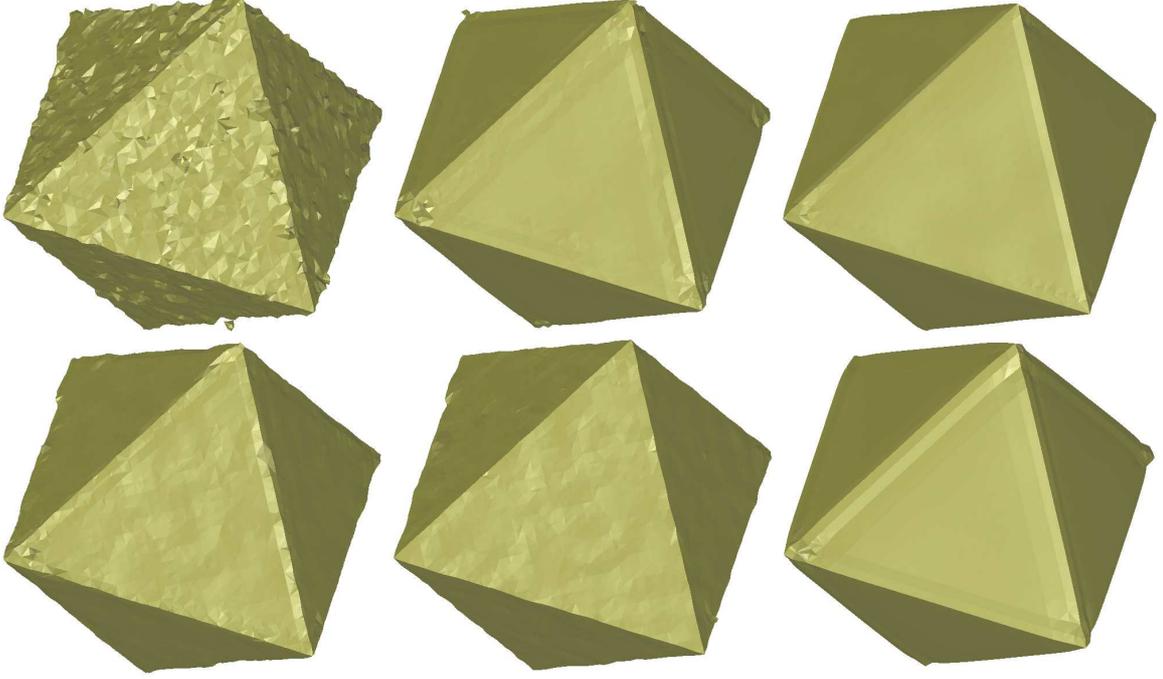


Figure 5.2: The influence of the choice of σ on the evolution is shown for the projection regularization variant (R4). From left to right results from the evolution of the noisy octahedron data-set are shown for $\sigma = 2h$, $\sigma = 4h$, and $\sigma = 8h$. The top row displays the first scale step of the evolution, the bottom row displays the third scale step. The qualitative behavior of the method is rather invariant under these choices of σ .

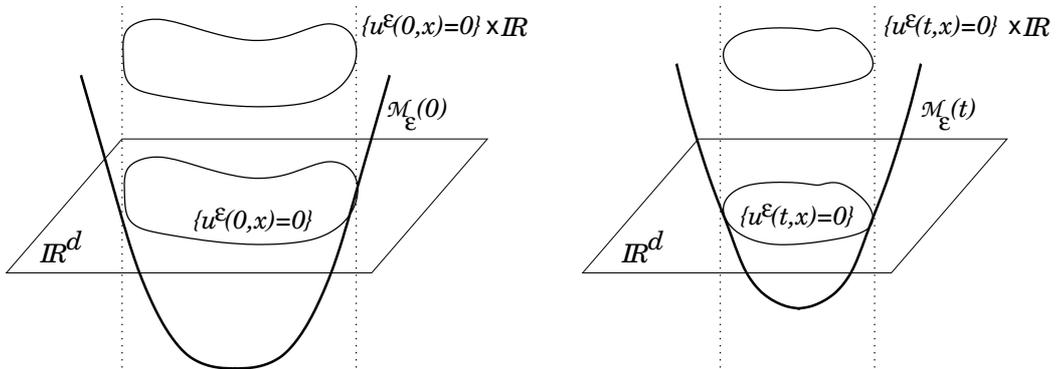


Figure 5.3: The problem which uses the regularized Euclidean norm $|\cdot|_\varepsilon$ corresponds to the graph-evolution in \mathbb{R}^{d+1} . The original evolution thereby corresponds to the evolution of the zero level-set. Choosing $\varepsilon > 0$ small approximates the cylinder $\{u^\varepsilon(t, x) = 0\} \times \mathbb{R}$ (cf. Evans and Spruck [32])

Remark 5.2. The use of the regularized Euclidean norm $|\cdot|_\varepsilon$ corresponds to an embedding of the problem into \mathbb{R}^{d+1} as follows: Assume $u^\varepsilon(t, x)$ is a smooth solution, write $y = (x, x_{d+1})$ and define $v^\varepsilon(t, y) := u^\varepsilon(t, x) - \varepsilon x_{d+1}$. Then $|\nabla_y v^\varepsilon|^2 = |\nabla_x u^\varepsilon|^2 + \varepsilon^2 = |\nabla_x u^\varepsilon|_\varepsilon^2$. Thus, the problem becomes

$$\partial_t v^\varepsilon - \operatorname{tr} \left(\tilde{A} \left(\operatorname{Id} - \frac{\nabla_y v^\varepsilon \otimes \nabla_y v^\varepsilon}{|\nabla_y v^\varepsilon|^2} \right) D_y^2 v^\varepsilon \right) = 0 \quad \text{in } I \times \mathbb{R}^{d+1}, \quad (5.6)$$

where

$$\tilde{A} := \begin{pmatrix} A(\Sigma) & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad D_y^2 v^\varepsilon = \begin{pmatrix} D^2 u^\varepsilon & 0 \\ 0 & 0 \end{pmatrix}.$$

Now each level-set of v^ε evolves according to (5.6) and particularly the zero level-set

$$\mathcal{M}_\varepsilon(t) := \{y \in \mathbb{R}^{d+1} : v^\varepsilon(t, y) = 0\},$$

for which $x_{d+1} = \varepsilon^{-1}u^\varepsilon(t, x)$. Consequently the problem with the regularized norm $|\cdot|_\varepsilon$ corresponds to the evolution of the graph $\{x_{d+1} = \varepsilon^{-1}u^\varepsilon(t, x)\}$ (cf. Ecker and Huisken [30]). For small $\varepsilon > 0$ this graph approximates the cylinder $\{u^\varepsilon(0, x) = 0\} \times \mathbb{R}$ (cf. Figure 5.3). With the approximation it is expected that the complicated behavior, which might be singular, is approximated by the smooth evolution of the graph (cf. [32]).

Variational form

The variational form of the problem (2.3) is obtained as usual: Let $\vartheta \in H^1(\Omega)$ be a test function. We multiply the equation with ϑ and integrate over Ω . Clearly the boundary terms vanish because of the Neumann conditions, we imposed on $\partial\Omega$. Furthermore we denote the L^2 -scalar product on Ω by (\cdot, \cdot) and the dual-space of $H^1(\Omega)$ by $(H^1)'(\Omega)$. With the corresponding dual pairing on $(H^1)'(\Omega) \times H^1(\Omega)$ denoted by $\langle \cdot, \cdot \rangle$, we obtain the following weak problem

Find $u \in L^2(\mathbb{R}^+; H^1(\Omega))$ such that $\partial_t u \in L^2(\mathbb{R}^+; (H^1)'(\Omega))$ and for almost every $t \in \mathbb{R}^+$

$$\left\langle \frac{\partial_t u}{|\nabla u|_\varepsilon}, \vartheta \right\rangle + \left(a(\Sigma) \frac{\nabla u}{|\nabla u|_\varepsilon}, \nabla \vartheta \right) = 0, \quad \forall \vartheta \in H^1(\Omega), \quad (5.7)$$

$$u(0, \cdot) = u_0(\cdot) \quad \text{in } \Omega.$$

Here $L^2(\mathbb{R}^+; H^1(\Omega))$ and $L^2(\mathbb{R}^+; (H^1)'(\Omega))$ denote the Bochner spaces equipped with the norms

$$\|v\|_{L^2(\mathbb{R}^+; H^1(\Omega))} := \int_{\mathbb{R}^+} \|v\|_{H^1(\Omega)} \quad \text{and} \quad \|v\|_{L^2(\mathbb{R}^+; (H^1)'(\Omega))} := \int_{\mathbb{R}^+} \|v\|_{(H^1)'(\Omega)}.$$

Let us mention that even though for the solutions of the linear problem we had only found $u \in C^0(Q)$ in Chapter 4, we ask for H^1 regularity in (5.7) for the approximate solutions. We obviously have this regularity for the FE solutions of the discrete problems, but the convergence of the discrete solutions toward the viscosity solution is not proven. For mean curvature motion the convergence of a finite element scheme towards the viscosity solution was shown by Deckelnick and Dziuk in [22, 23].

Time-discrete problem

To proceed, we discretize the scale t with an equidistant stepping of size τ and a semi-implicit backward Euler scheme. The scale-derivative ∂_t is replaced by the backward difference quotient

$$\partial_t u \rightsquigarrow \frac{u^n - u^{n-1}}{\tau} =: D_t^- u^n,$$

where $u^n := u(t_n)$ and $t_n := n\tau$. This backward-scheme becomes semi-implicit, if on scale $(n+1)\tau$ the nonlinear terms $a(\Sigma)$ and $|\nabla u|^{-1}$ are evaluated on the previous scale $n\tau$. We will use the notation $a^n := a(\Sigma(u(t^n)))$ in the following, i.e. a^n is the diffusion tensor evaluated at the scale-step t^n . The time-discrete problem now reads:

Find a sequence $(u^n)_{n \in \mathbb{N}}$, $u^n \in H^1(\Omega)$ such that

$$\left(\frac{u^n - u^{n-1}}{\tau |\nabla u^{n-1}|_\varepsilon}, \vartheta \right) + \left(a^{n-1}(\Sigma) \frac{\nabla u^n}{|\nabla u^{n-1}|_\varepsilon}, \nabla \vartheta \right) = 0 \quad \forall \vartheta \in H^1(\Omega), \quad (5.8)$$

$$u^0(\cdot) = u_0(\cdot) \quad \text{in } \Omega,$$

where a^n means the evaluation of the diffusion tensor a at scale t_n .

We thus have transformed the parabolic problem into a sequence of elliptic problems. Since the resulting bilinear forms on $H^1(\Omega) \times H^1(\Omega)$

$$b^n(v, w) := \left(\frac{v}{\tau |\nabla u^{n-1}|_\varepsilon}, w \right) + \left(a^{n-1}(\Sigma) \frac{\nabla v}{|\nabla u^{n-1}|_\varepsilon}, \nabla w \right)$$

are coercive and continuous, the Lax-Milgram Theorem [1] guarantees the existence of the u^n . Finally we consider the Rothe-approximation

$$u^{(n)}(t) := u^n + (t - t_n) D_t^- u^{n+1} \quad \text{for } t_n \leq t \leq t_{n+1}$$

as an approximation to the continuous solution.

Fully-discrete problem

We proceed in the discretization process with the spatial approximations. We replace the solution space $H^1(\Omega)$ with the finite dimensional subspace \mathcal{V}^h , which was introduced above. We only have to test with functions in \mathcal{V}^h , and furthermore we can substitute the basis decomposition (5.2) for the approximation u_h . Testing subsequently with all basis functions ψ_α , one obtains a system of equations for each scale step. In matrix form the system is written as

$$(M^{n-1} + \tau L^{n-1}) U_h^n = \tau F^{n-1}, \quad (5.9)$$

where the definitions

$$\begin{aligned} M^n &:= \left(\left(\frac{\psi_\alpha}{|\nabla u^n|_\varepsilon}, \psi_\beta \right) \right)_{\alpha, \beta} && \text{mass matrix,} \\ L^n &:= \left(\left(a^n \frac{\nabla \psi_\alpha}{|\nabla u^n|_\varepsilon}, \nabla \psi_\beta \right) \right)_{\alpha, \beta} && \text{stiffness matrix,} \\ F^n &:= M^n U_h^n && \text{right hand side} \end{aligned} \quad (5.10)$$

have been used. These systems of equations are symmetric, positive definite and they are solved by a diagonally preconditioned conjugate gradient method (CG) [73]. All integrations are performed element-wise and a midpoint integration rule is used to approximate the integrals. Obviously, the integration has to be performed over the support of $\psi_\alpha \psi_\beta$ only. For example, the entries of the stiffness matrix are computed via

$$\begin{aligned} (L^n)_{\alpha, \beta} &= \left(a^n \frac{\nabla \psi_\alpha}{|\nabla u^n|_\varepsilon}, \nabla \psi_\beta \right) \\ &= \sum_{E \in \text{supp}(\psi_\alpha \psi_\beta)} \int_E a^n \frac{\nabla \psi_\alpha}{|\nabla u^n|_\varepsilon} \cdot \nabla \psi_\beta \, dx \\ &\approx \sum_{E \in \text{supp}(\psi_\alpha \psi_\beta)} \frac{1}{|\nabla u^n(c_E)|_\varepsilon} \int_E a^n(c_E) \nabla \psi_\alpha \cdot \nabla \psi_\beta \, dx, \end{aligned}$$

where c_E is the center of an element E . Thus, we have assumed the gradient ∇u_h^n and the diffusion tensor a^n being element-wise constant.

Mass-lumping

Alternatively to the application of the standard mass matrix (5.10) the use of lumped masses is possible. The lumped mass-matrix [76] is diagonal due to one of the following equivalent approximations:

1. The integration is performed with a nodal integration formula. This means an integral is replaced by a weighted sum over the nodal values. Denoting the weights with ω_γ we have

$$(M_{\text{lumped}})_{\alpha\beta} = \int \psi_\alpha \psi_\beta = \sum_{x_\gamma \in \text{supp}(\psi_\alpha \psi_\beta)} \omega_\gamma \psi_\alpha(x_\gamma) \psi_\beta(x_\gamma) = \omega_\alpha \delta_{\alpha\beta}$$

and in particular for the quadtree- and octtree-grids, we have $\omega_\gamma = h^d$.

2. For the integration of the masses as little regularity as needed is imposed for the basis functions. Thus, the original basis functions are replaced with functions, which are constant on the dual cells \tilde{E}_i . Then clearly

$$\text{supp}(\psi_\alpha \psi_\beta) = \begin{cases} \emptyset & \text{if } \alpha \neq \beta, \\ \text{supp } \psi_\alpha & \text{else.} \end{cases} \quad \Rightarrow \quad (M_{\text{lumped}})_{\alpha\beta} = \delta_{\alpha\beta} |\text{supp } \psi_\alpha| = \delta_{\alpha\beta} h^d.$$

3. The lumped mass matrix arises from a contraction of the row-entries of the original mass matrix to the diagonal elements. This means

$$(M_{\text{lumped}})_{\alpha\beta} = \delta_{\alpha\beta} \sum_{\gamma} M_{\alpha\gamma}.$$

In any case we have a convergence of the lumped mass matrix M_{lumped} to the original mass matrix M for $h \rightarrow 0$ (cf. [76]).

If we use lumped masses in the discretization instead of the matrix (5.10), we can easily prove L^∞ stability of the whole discrete evolution.

Theorem 5.3. *There exists a limit u_h^n of a subsequence $u_h^{\varepsilon,n}$ for $\varepsilon \rightarrow 0$, where $u_h^{\varepsilon,n}$ is the solution of the system (5.9) with mass lumping. For this generalized solution u_h^n , we have the following estimate independent of ε :*

$$\|u_h^n\|_{\infty, \Omega} \leq \|u_h^0\|_{\infty, \Omega}.$$

Proof. We follow the ideas of Walkington [78] and Handlovičová et al. [39] in the proof. Let us denote the nodal values of $u_h^{\varepsilon,n}$ by $U_h^{\varepsilon,n} := (U_\alpha^{\varepsilon,n})_\alpha$. To prove the inequality we further assume that the maximum of $U_h^{\varepsilon,n}$ is achieved in $U_\beta^{\varepsilon,n}$. We write down the equation from (5.9) corresponding to the node x_β with lumped masses:

$$(M_{\text{lumped}}^{n-1})_{\beta,\beta} U_\beta^{\varepsilon,n} + \tau \left(U_\beta^{\varepsilon,n} (L^{n-1})_{\beta,\beta} + \sum_{x_\alpha \neq x_\beta} U_\alpha^{\varepsilon,n} (L^{n-1})_{\beta,\alpha} \right) = (M_{\text{lumped}}^{n-1})_{\beta,\beta} U_\beta^{n-1} \quad (5.11)$$

Since the stiffness matrix is an M -matrix, the term in parentheses is non-negative, as an easy calculation shows. Thus, we have

$$U_\beta^{\varepsilon,n} \leq U_\beta^{n-1} \leq \max \left\{ U_\alpha^{n-1} : \alpha \in \{0, \dots, N\}^d \right\}.$$

Analogously we obtain the following estimate for the minimum

$$\min \left\{ U_\alpha^{n-1} : \alpha \in \{0, \dots, N\}^d \right\} \leq U_\beta^{\varepsilon,n},$$

from which we recursively obtain

$$\|u_h^{\varepsilon,n}\|_{\infty,\Omega} \leq \|u_h^0\|_{\infty,\Omega}.$$

Since this estimate is independent of ε we have the existence of a subsequence of vectors $U_h^{\varepsilon,n}$ converging to U_h^n as $\varepsilon \rightarrow 0$. We denote the corresponding finite element function by $u_h^n \in \mathcal{V}^h$ and it is clear that u_h^n fulfills the first estimate of the theorem. \square

5.6 Discretization of the evolution of image-sequences

The expositions of the following section deal with the discretization of the anisotropic evolution for image-sequences. The discretization is in many parts similar to the discretization presented in the last section. Again we use the regularized Euclidean norm and again we assume H^1 regularity in the variational form. However, the image-sequence evolution is a four or five dimensional problem (scale, sequence-time, space \mathbb{R}^d) for which especially in the case $d = 3$ a finite element approach is non-desirable, due to the high numerical efforts involved. In case of the ultrasound image-sequence of the human heart, this would result in a problem having $16 \times 129 \times 129 \times 129 \approx 2^{25} \approx 3 \cdot 10^7$ unknowns. The corresponding matrix would have bandwidth 81 and a supercomputer would be needed for its storage and an iterative solution. By choosing appropriate quadrature rules and a mass-lumping in sequence time, we derive a coupled finite difference/finite element scheme. But we start with the introduction of the weak form of the problem.

Variational form

As before we denote the sequence-time/space cylinder with $Q := (0, T) \times \Omega$. Again, we take a test-function $\vartheta \in H^1(Q)$, multiply (2.17) with ϑ and integrate over Q . Since we have either imposed natural boundary conditions (BC1) or periodicity of the sequence (BC2), the boundary terms from the integration vanish: Green's formula yields

$$\begin{aligned} \int_Q \operatorname{div} \left(a_{(s,x)} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \right) \vartheta \, dx &= \int_Q a_{(s,x)} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nabla_{(s,x)} \vartheta \, dx \\ &\quad + \int_{\partial Q} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nu_{(s,x)} \vartheta \, d\mathcal{H}^{n-1}, \end{aligned}$$

where $\nu_{(s,x)}$ is the outer normal to the cylinder Q and $d\mathcal{H}^{n-1}$ is the area element. The area integral over $\partial\Omega$ clearly vanishes if (BC1) is satisfied. But also if u fulfills the second boundary condition variant (BC2), the boundary integral is zero, because we have $\partial Q = (\partial I \times \Omega) \cup (I \times \partial\Omega)$. And we can split up the integral

$$\begin{aligned} \int_{\partial Q} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nu_{(s,x)} \vartheta \, d\mathcal{H}^{n-1} &= \int_\Omega \left(\frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nu_{(s,x)} \vartheta \right) \Big|_{s=T} d\mathcal{H}^{n-1} \\ &\quad - \int_\Omega \left(\frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nu_{(s,x)} \vartheta \right) \Big|_{s=0} d\mathcal{H}^{n-1} \\ &\quad + \int_I \int_{\partial\Omega} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon} \cdot \nu_{(s,x)} \vartheta \, d\mathcal{H}^{n-1}. \end{aligned}$$

The first two terms cancel out, due to the assumed periodicity and the last term vanishes, too, because on $I \times \partial\Omega$ the condition $\nabla u \cdot \nu = 0$ implies $\nabla_{(s,x)} u \cdot \nu_{(s,x)} = (\partial_s u, \nabla u) \cdot (0, \nu) = 0$, if ν denotes the outer normal to $\partial\Omega$.

Using again the regularized Euclidean norm $|\cdot|_\varepsilon$, we have derived the following weak form of the evolution problem, denoting the dual pairing on $(H^1)'(Q) \times H^1(Q)$ by $\langle \cdot, \cdot \rangle$ and the L^2 scalar-product on Q by (\cdot, \cdot) :

$$\begin{aligned} & \text{Find } u \in L^2(\mathbb{R}^+; H^1(Q)) \text{ such that } \partial_t u \in L^2(\mathbb{R}^+; (H^1)'(Q)) \text{ and for almost every } \\ & t \in \mathbb{R}^+ \\ & \left\langle \frac{\partial_t u}{|\nabla u_{(s,x)}|_\varepsilon}, \vartheta \right\rangle + \left(a_{(s,x)} \frac{\nabla_{(s,x)} u}{|\nabla u_{(s,x)}|_\varepsilon}, \nabla_{(s,x)} \vartheta \right) = 0 \quad \forall \vartheta \in H^1(Q), \\ & u(0, \cdot) = u_0(\cdot) \quad \text{in } Q. \end{aligned} \quad (5.12)$$

Time-discrete problem

We proceed as for the static image processing model. Dividing the scale into equidistant steps of size τ , using a backward semi-implicit Euler scheme and again denoting $t_n := n\tau$, $u^n := u(t_n)$, we obtain the time-discrete problem:

$$\begin{aligned} & \text{Find a sequence } (u^n)_{n \in \mathbb{N}} \text{ } u^n \in H^1(Q) \text{ such that} \\ & \left(\frac{u^n - u^{n-1}}{\tau |\nabla u_{(s,x)}^{n-1}|_\varepsilon}, \vartheta \right) + \left(a_{(s,x)}^{n-1} \frac{\nabla_{(s,x)} u^n}{|\nabla u_{(s,x)}^{n-1}|_\varepsilon}, \nabla_{(s,x)} \vartheta \right) = 0 \quad \forall \vartheta \in H^1(Q), \\ & u^0(\cdot) = u_0(\cdot) \quad \text{in } Q, \\ & \text{where } a_{(s,x)}^n \text{ means the evaluation of the diffusion tensor } a_{(s,x)} \text{ at time } t_n. \end{aligned} \quad (5.13)$$

Again the resulting bilinear forms

$$b^n(v, w) := \left(\frac{v}{\tau |\nabla u_{(s,x)}^{n-1}|_\varepsilon}, w \right) + \left(a_{(s,x)}^{n-1} \frac{\nabla_{(s,x)} v}{|\nabla u_{(s,x)}^{n-1}|_\varepsilon}, \nabla_{(s,x)} w \right)$$

on $H^1(Q) \times H^1(Q)$ are coercive and continuous, which ensures the existence of the u^n via the Lax-Milgram Theorem.

Fully-discrete problem

At least formally we now derive the fully discrete problem, although this is not the variant, which is used for the actual solution. As mentioned before, we will utilize appropriate quadrature rules to simplify the scheme significantly.

The solution space is replaced by the finite dimensional subspace $\mathcal{W}^\rho \times \mathcal{V}^h$. Inserting the basis decomposition (5.3) into the equation and testing with each basis function, one again obtains a linear system

$$(M^{n-1} + \tau L^{n-1}) U_{\rho,h}^n = \tau F^{n-1},$$

where

$$\begin{aligned} M^n + \tau L^n &= \left(\left(\frac{\psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \psi_\beta \right) + \tau \left(a_{(s,x)}^n \frac{\nabla_{(s,x)} \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \nabla_{(s,x)} \psi_\beta \right) \right)_{(i,\alpha),(j,\beta)} \\ F^n &= M^n U_{\rho,h}^n. \end{aligned}$$

The system can be solved using an iterative solver, but the effort is enormous, as motivated above: The use of the spatio-temporal finite elements is non-desirable especially in the case

$t \in \mathbb{R}_0^+$	scale
$s \in I$	sequence-time coordinate
$x \in \Omega$	spatial coordinate
$n \in \mathbb{N}$	scale step
$i, j \in \{0, \dots, M\}$	temporal node (= sequence-frame)
$\alpha, \beta \in \{0, \dots, N\}^d$	spatial node (= Pixel/Voxel in a frame)
τ	scale step width
ρ	temporal step width (= sequence step width)
h	spatial step width

Table 5.2: *Meaning of indices in the discretized coupled diffusion problem.*

of (3+1)D image-sequences. As a consequence we proceed with a strong simplification of the system matrix, we have formally obtained above. The simplified scheme is a coupled finite-difference/finite-element scheme, which is solved by a block solver.

But before we proceed to the simplification of the system matrix $M^n + \tau L^n$, let us analyze how the diffusion tensor $a_{(s,x)}^n$ acts within the bilinear form. We obtain for two functions u and w :

$$\begin{aligned}
a_{(s,x)}^n \nabla_{(s,x)} u \cdot \nabla_{(s,x)} w &= a_v^n (V^\sigma \cdot \nabla_{(s,x)} u) (V^\sigma \cdot \nabla_{(s,x)} w) + a^n (\Sigma) \nabla u \cdot \nabla w \\
&= \tilde{a}_v^n (\partial_s u + v_{\text{app}}^\sigma \cdot \nabla u) (\partial_s w + v_{\text{app}}^\sigma \cdot \nabla w) + a^n (\Sigma) \nabla u \cdot \nabla w \quad (5.14) \\
&=: \tilde{a}_v^n \frac{D}{\partial s} u \frac{D}{\partial s} w + a^n (\Sigma) \nabla u \cdot \nabla w,
\end{aligned}$$

where $\frac{D}{\partial s} := \partial_s + v_{\text{app}}^\sigma \cdot \nabla$ denotes the material derivative along the apparent trajectory and $\tilde{a}_v^n = a_v^n / |(1, v_{\text{app}}^\sigma)|^2$. In the following, we assume v_{app}^σ being adjusted such that $|(1, v_{\text{app}}^\sigma)| = 1$ and thus $\tilde{a}_v^n = a_v^n$. The decomposition given in the latter equation, is used in the next section to split up and simplify the system matrix.

5.6.1 Finite-Elements in space – finite differences in time

For the detailed analysis of the matrix system it is necessary to refine the notation: As introduced above the temporal degrees of freedom (i.e. the frames of the sequence) are denoted with Latin indices $i, j \in \{0, \dots, M\}$. The spatial degrees of freedom (i.e. the pixels/voxels within the frames) are denoted with Greek indices $\alpha, \beta \in \{0, \dots, N\}^d$. For a scale-step n we write a super-index n and define $u_i^n(\alpha h) = u^n(i\rho, \alpha h)$. Furthermore we introduce the following abbreviations for the vectors containing the nodal values:

$$\begin{aligned}
U^n &:= U_{\rho,h}^n = (U_i^n)_i = (U_0^n, \dots, U_M^n), \\
U_i^n &= (U_{i,\alpha}^n)_\alpha = (U_{i,0}^n, \dots, U_{i,N^d-1}^n).
\end{aligned}$$

For the sake of the presentation's clearness we omit the index ρ, h for the discrete functions $u_{\rho,h}^n$ and the vectors of the nodal values here and below. Consequently for each $u \in \mathcal{W}^\rho \otimes \mathcal{V}^h$ we have the abbreviation

$$u(n\tau, i\rho, \alpha h) = U_{i,\alpha}^n.$$

To avoid any confusion, a list of all indices in use is given in Table 5.2.

Let us again write down the matrix resulting from the discretization of the image-sequence

evolution. Using the observation (5.14), we obtain

$$(M^n + \tau L^n)_{(i,\alpha),(j,\beta)} = \left(\frac{\phi_i \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \phi_j \psi_\beta \right)_Q \quad (\text{MM})$$

$$+ \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \frac{D}{\partial s}(\phi_i \psi_\alpha), \frac{D}{\partial s}(\phi_j \psi_\beta) \right)_Q \quad (\text{CP})$$

$$+ \tau \left(\frac{a^n(\Sigma)}{|\nabla_{(s,x)} u^n|_\varepsilon} \nabla(\phi_i \psi_\alpha), \nabla(\phi_j \psi_\beta) \right)_Q. \quad (\text{ML})$$

The first term (MM) corresponds to a spatio-temporal mass matrix, the third term (ML) contains the purely spatial stiffness-matrix, whereas the middle term (CP) consists of the actual coupling in sequence-time and space (cf. Figure 5.4). The key for the simplification is a mass lumping in sequence time, which results in a diagonalization of the mass matrix (see Section 5.5 and p. 82).

Spatio-temporal mass-matrix

Our first observation concerns the mass matrix (MM). In all terms, containing the temporal mass, we approximate the denominator $|\nabla_{(s,x)} u^n|_\varepsilon$ by a central temporal difference $|D_{(s,x)}^\pm u_i^n|_\varepsilon$ on the intervals containing s_i . For one entry $(i, \alpha), (j, \beta)$ we thus have

$$\begin{aligned} (\text{MM})_{(i,\alpha),(j,\beta)} &= \left(\frac{\phi_i \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \phi_j \psi_\beta \right)_Q = \int_0^T \int_\Omega \frac{\phi_i \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon} \phi_j \psi_\beta \, dx \, ds \\ &\approx \sum_k \int_{s_k}^{s_{k+1}} \int_\Omega \frac{\phi_i \psi_\alpha}{|D_{(s,x)}^\pm u_k^n|_\varepsilon} \phi_j \psi_\beta \, dx \, ds. \end{aligned}$$

In direction of sequence time, the integration has to be performed of course only on the support of $\phi_i \phi_j$. Incorporating the mass lumping in our setting, we need to consider only the diagonal entries $i = j$ for which $\int \phi_i^2 = 1$. Using Fubini's Theorem we obtain

$$(\text{MM})_{(i,\alpha),(j,\beta)} \approx \delta_{ij} \int_{s_{i-1}}^{s_{i+1}} \phi_i^2 \, ds \int_\Omega \frac{\psi_\alpha \psi_\beta}{|D_{(s,x)}^\pm u_i^n|_\varepsilon} \, dx = \delta_{ij} \rho \int_\Omega \frac{\psi_\alpha \psi_\beta}{|D_{(s,x)}^\pm u_i^n|_\varepsilon} \, dx.$$

Denoting the spatial mass matrix of frame i in scale n with $M_i^n(1)$, the above simplifications result in the following block-diagonal form for the spatio-temporal mass matrix

$$\begin{pmatrix} \ddots & & & \\ & \rho M_i^n(1) & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix}, \quad \text{with} \quad M_i^n(b) := \left(\left(\frac{b \psi_\alpha}{|D_{(s,x)}^\pm u_i^n|_\varepsilon}, \psi_\beta \right)_\Omega \right)_{\alpha,\beta}. \quad (5.15)$$

$$\begin{array}{c}
\left(\frac{\phi_i \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \phi_j \psi_\beta \right)_Q + \tau \left(a_v^n \frac{D}{|\nabla_{(s,x)} u^n|_\varepsilon} \phi_i \psi_\alpha, \frac{D}{\partial s} \phi_j \psi_\beta \right)_Q + \tau \left(a^n(\Sigma) \frac{\nabla \phi_i \psi_\alpha}{|\nabla_{(s,x)} u^n|_\varepsilon}, \nabla \phi_j \psi_\beta \right)_Q \\
\downarrow \qquad \qquad \qquad \downarrow \qquad \qquad \qquad \downarrow \\
\text{Sequence-temporal/spatial} \qquad \qquad \text{Coupled diffusion terms} \qquad \qquad \text{Spatial stiffness matrix} \\
\text{mass matrix (MM)} \qquad \qquad \qquad \text{(CP)} \qquad \qquad \qquad \text{(ML)} \\
\downarrow \\
\left[\begin{array}{l}
\left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \partial_s(\phi_i \psi_\alpha), \partial_s(\phi_j \psi_\beta) \right)_Q \rightarrow \text{Temporal stiffness matrix (CP}_1) \\
+ \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \partial_s(\phi_i \psi_\alpha), v \cdot \nabla(\phi_j \psi_\beta) \right)_Q \\
+ \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} v \cdot \nabla(\phi_i \psi_\alpha), \partial_s(\phi_j \psi_\beta) \right)_Q \\
+ \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} v \cdot \nabla(\phi_i \psi_\alpha), v \cdot \nabla(\phi_j \psi_\beta) \right)_Q \rightarrow \text{Anisotropic spatial stiffness matrix (CP}_4)
\end{array} \right] \left. \begin{array}{l} \\ \\ \\ \end{array} \right\} \begin{array}{l} \\ \\ \text{Mixed nonlinear stiffness matrix (CP}_2) \text{ and (CP}_3) \end{array}
\end{array}$$

Figure 5.4: The matrix of the system splits up into various parts. The sequence-time/space matrix (upper row, left term) and the spatial stiffness matrix (upper row, right term) lead to diagonal blocks in the resulting scheme, whereas the coupled diffusion terms (upper row, middle term) split up further (inner box).

Temporal mass — spatial diffusion

In a similar manner the term (ML) can be handled. Since we again have a temporal mass involved, we can apply the lumping of these masses, which results in

$$\begin{aligned}
(\text{ML})_{(i,\alpha),(j,\beta)} &\approx \tau \sum_k \int_{s_k}^{s_{k+1}} \int_{\Omega} a^n(\Sigma) \frac{\phi_i \nabla \psi_\alpha}{|D_{(s,x)}^\pm u_k^n|_\varepsilon} \cdot (\phi_j \nabla \psi_\beta) \, dx \, ds \\
&= \tau \delta_{ij} \int_{s_{i-1}}^{s_i} \phi_i^2 \int_{\Omega} a_i^n(\Sigma) \frac{\nabla \psi_\alpha}{|D_{(s,x)}^\pm u_i^n|_\varepsilon} \cdot \nabla \psi_\beta \, dx \, ds \\
&= \tau \rho \delta_{ij} \int_{\Omega} a_i^n(\Sigma) \frac{\nabla \psi_\alpha}{|D_{(s,x)}^\pm u_i^n|_\varepsilon} \cdot \nabla \psi_\beta \, dx.
\end{aligned}$$

Here, a_i^n is the spatial diffusion tensor on frame i . As before, the simplification results in a block diagonal matrix, here having the form

$$\begin{pmatrix} \ddots & & \\ & \tau \rho L_i^n & \\ & & \ddots \end{pmatrix}, \quad \text{with} \quad L_i^n := \left(\left(a_i^n(\Sigma) \frac{\nabla \psi_\alpha}{|D_{(s,x)}^\pm u_i^n|_\varepsilon}, \nabla \psi_\beta \right)_\Omega \right)_{\alpha,\beta}, \quad (5.16)$$

where the L_i^n denotes the spatial stiffness matrix of frame i at scale n .

Spatio-temporal diffusion

Finally, the simplification of the coupled spatio-temporal diffusion term (CP) remains. According to the definition of the material derivative along the apparent trajectory v_{app}^σ , we

can write

$$\begin{aligned} (\text{CP})_{(i,\alpha),(j,\beta)} &= \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \frac{D}{\partial s} (\phi_i \psi_\alpha), \frac{D}{\partial s} (\phi_j \psi_\beta) \right)_Q \\ &= \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \partial_s (\phi_i \psi_\alpha), \partial_s (\phi_j \psi_\beta) \right)_Q \end{aligned} \quad (\text{CP}_1)$$

$$+ \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} \partial_s (\phi_i \psi_\alpha), v_{\text{app}}^\sigma \cdot \nabla (\phi_j \psi_\beta) \right)_Q \quad (\text{CP}_2)$$

$$+ \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} v_{\text{app}}^\sigma \cdot \nabla (\phi_i \psi_\alpha), \partial_s (\phi_j \psi_\beta) \right)_Q \quad (\text{CP}_3)$$

$$+ \tau \left(\frac{a_v^n}{|\nabla_{(s,x)} u^n|_\varepsilon} v_{\text{app}}^\sigma \cdot \nabla (\phi_i \psi_\alpha), v_{\text{app}}^\sigma \cdot \nabla (\phi_j \psi_\beta) \right)_Q, \quad (\text{CP}_4)$$

whose terms are again considered separately.

Temporal stiffness matrix. The first component (CP₁) is the sequence-time elliptic term. As before, we split up the integration to the small sequence-time intervals, but now we use a forward difference quotient $D^+ u_k^n$ for the evaluation of the denominator on an interval I_k . Thus, an entry of the corresponding matrix is given by

$$\begin{aligned} (\text{CP}_1)_{(i,\alpha),(j,\beta)} &= \tau \int_0^T \int_\Omega a_v^n \frac{\partial_s (\phi_i \psi_\alpha)}{|\nabla_{(s,x)} u^n|_\varepsilon} \partial_s (\phi_j \psi_\beta) dx ds \\ &= \tau \sum_k \int_{s_k}^{s_{k+1}} \partial_s \phi_i \partial_s \phi_j ds \int_\Omega \frac{(a_v^n)_k \psi_\alpha \psi_\beta}{|D_{(s,x)}^+ u_k^n|_\varepsilon} dx, \end{aligned}$$

because the temporal derivative is constant on each interval I_k . Obviously the supports of ϕ_i and ϕ_j overlap only in the case $|i - j| \leq 1$, therefore we consider the following two cases:

- $i = j$: Using $\partial_s \phi_i = 1/\rho$, where ρ is the temporal grid-width, we get for the diagonal entries

$$\begin{aligned} (\text{CP}_1)_{(i,\alpha),(i,\beta)} &= \tau \int_{s_{i-1}}^{s_i} \partial_s \phi_i \partial_s \phi_j ds \int_\Omega \frac{(a_v^n)_{i-1} \psi_\alpha \psi_\beta}{|D_{(s,x)}^+ u_{i-1}^n|_\varepsilon} dx \\ &\quad + \tau \int_{s_i}^{s_{i+1}} \partial_s \phi_i \partial_s \phi_j ds \int_\Omega \frac{(a_v^n)_i \psi_\alpha \psi_\beta}{|D_{(s,x)}^+ u_i^n|_\varepsilon} dx \\ &= \frac{\tau}{\rho} \int_\Omega \left(\frac{(a_v^n)_{i-1}}{|D_{(s,x)}^+ u_{i-1}^n|_\varepsilon} + \frac{(a_v^n)_i}{|D_{(s,x)}^+ u_i^n|_\varepsilon} \right) (\psi_\alpha \psi_\beta) dx, \end{aligned}$$

where $(a_v^n)_i$ denotes the local temporal diffusivity in scale step n which we assume to be constant on an interval I_i .

- $i + 1 = j$: For the off-diagonal entries we use $\partial_s \phi_i = -\partial_s \phi_{i+1} = 1/\rho$ to obtain

$$(\text{CP}_1)_{(i,\alpha),(i+1,\beta)} = \tau \int_{s_i}^{s_{i+1}} \partial_s \phi_i \partial_s \phi_j ds \int_\Omega \frac{(a_v^n)_i \psi_\alpha \psi_\beta}{|D_{(s,x)}^+ u_i^n|_\varepsilon} dx = -\frac{\tau}{\rho} \int_\Omega \frac{(a_v^n)_i \psi_\alpha \psi_\beta}{|D_{(s,x)}^+ u_i^n|_\varepsilon} dx.$$

Consequently, each row and the resulting matrix has 3-band form

$$\begin{pmatrix} \ddots & & & & \\ & -\frac{\tau}{\rho} B_{i-1}^n & \frac{\tau}{\rho} \{B_{i-1}^n + B_i^n\} & -\frac{\tau}{\rho} B_i^n & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \ddots \end{pmatrix}, \quad \text{with } B_i^n := M_i^n((a_v^n)_i) \quad (5.17)$$

We emphasize that this scheme is similar to a finite difference scheme with the stencil $[-1, 2, -1]$.

Mixed derivatives. We consider the mixed derivative terms (CP₂) and (CP₃) next. They contain temporal and spatial derivatives and carry the actual coupling of the spatial and temporal diffusion. If the temporal diffusion coefficient were constant, these terms would cancel out for symmetry reasons. But in general we have to take them into account. Obviously, only indices i, j for $|i - j| \leq 1$ have to be considered. For the diagonal $i = j$ we get the sum

$$\begin{aligned} & ((\text{CP}_2) + (\text{CP}_3))_{(i,\alpha),(i,\beta)} \\ &= \tau \int_{s_{i-1}}^{s_i} \partial_s \phi_i \phi_i \int_{\Omega} \frac{(a_v^n)_{i-1}}{|D_{(s,x)}^+ u_{i-1}^n|_{\varepsilon}} (\psi_{\alpha} v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta} + v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha} \psi_{\beta}) dx \\ & \quad + \tau \int_{s_i}^{s_{i+1}} \partial_s \phi_i \phi_i \int_{\Omega} \frac{(a_v^n)_i}{|D_{(s,x)}^+ u_i^n|_{\varepsilon}} (\psi_{\alpha} v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta} + v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha} \psi_{\beta}) dx. \end{aligned}$$

Using $\int_{I_{i-1}} \partial_s \phi_i \phi_i = -\int_{I_i} \partial_s \phi_i \phi_i = -\rho/2$, we obtain for the diagonal components

$$\begin{aligned} & ((\text{CP}_2) + (\text{CP}_3))_{(i,\alpha),(i,\beta)} \\ &= \frac{\tau \rho}{2} \int_{\Omega} \left(\frac{(a_v^n)_i}{|D_{(s,x)}^+ u_i^n|_{\varepsilon}} - \frac{(a_v^n)_{i-1}}{|D_{(s,x)}^+ u_{i-1}^n|_{\varepsilon}} \right) (\psi_{\alpha} v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta} + v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha} \psi_{\beta}) dx. \end{aligned}$$

The temporal off diagonal entries vanish, as one can see from $\int_{I_i} \partial_s \phi_{i+1} \phi_i = -\int_{I_i} \partial_s \phi_i \phi_{i+1}$, which leads to

$$\begin{aligned} & ((\text{CP}_2) + (\text{CP}_3))_{(i,\alpha),(i+1,\beta)} \\ &= \tau \int_{s_i}^{s_{i+1}} (\partial_s \phi_i \phi_{i+1} + \partial_s \phi_{i+1} \phi_i) ds \int_{\Omega} \frac{(a_v^n)_i}{|D_{(s,x)}^+ u_i^n|_{\varepsilon}} (\psi_{\alpha} v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta} + v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha} \psi_{\beta}) dx \\ &= 0. \end{aligned}$$

Summarizing the last results, we see that the matrix resulting from the terms (CP₂) and (CP₃) has block diagonal form

$$\begin{pmatrix} \ddots & & & \\ & \frac{\tau \rho}{2} R_i^n & & \\ & & \ddots & \\ & & & \ddots \end{pmatrix},$$

with

$$R_i^n = \left(\int_{\Omega} \left(\frac{(a_v^n)_i}{|D_{(s,x)}^+ u_i^n|_{\varepsilon}} - \frac{(a_v^n)_{i-1}}{|D_{(s,x)}^+ u_{i-1}^n|_{\varepsilon}} \right) (\psi_{\alpha} v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta} + v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha} \psi_{\beta}) dx \right)_{\alpha,\beta}. \quad (5.18)$$

Anisotropic elliptic matrix. The remaining term is an anisotropic spatial elliptic term, which contains anisotropic spatial derivatives. Since there is no temporal derivative involved, we

can again perform a mass lumping. The denominator is again replaced with the central difference quotient. We have

$$(\text{CP}_4)_{(i,\alpha),(i,\beta)} = \frac{\tau\rho}{2} \int_{\Omega} \frac{\{(a_v^n)_{i-1} + (a_v^n)_i\}}{|D_{(s,x)}^{\pm} u_i^n|_{\varepsilon}} (v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha}) (v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta}) dx.$$

And so denoting the anisotropic spatial stiffness matrix for scale n and frame i with Q_i^n the resulting matrix has block-diagonal form

$$\begin{pmatrix} \ddots & & & \\ & \frac{\tau\rho}{2} Q_i^n & & \\ & & \ddots & \end{pmatrix},$$

with

$$Q_i^n = \left(\left(\int_{\Omega} \frac{\{(a_v^n)_{i-1} + (a_v^n)_i\}}{|D_{(s,x)}^{\pm} u_i^n|_{\varepsilon}} (v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\alpha}) (v_{\text{app}}^{\sigma} \cdot \nabla \psi_{\beta}) dx \right) \right)_{\alpha,\beta}. \quad (5.19)$$

Let us summarize the analysis of the system matrix. We have incorporated a mass lumping in sequence time which has led to a 3-band block-matrix. Within this block matrix the off-diagonal blocks solely result from the temporal derivatives. Furthermore we observe, that each row of the block-matrix corresponds to one single frame of the sequence. This fact helps us in the next section to derive a simple block solving scheme.

5.6.2 An operator-splitting scheme

To discuss the operator-splitting scheme, let us consider one block-row of the simplified matrix. The system described by this row is (cf. (5.15) - (5.19))

$$\begin{aligned} \rho M_i^n U_i^{n-1} = & \left(\rho M_i^n + \tau\rho \left[L_i^n + \frac{1}{\rho^2} (B_{i-1}^n + B_i^n) + \frac{1}{2} (R_i^n + Q_i^n) \right] \right) U_i^n \\ & - \frac{\tau}{\rho} (B_{i-1}^n U_{i-1}^n - B_i^n U_{i+1}^n) \end{aligned} \quad (5.20)$$

and due to the 3-band structure only the frames U_{i-1}^n, U_i^n and U_{i+1}^n are coupled in this system. Thus, one block-row of the system can be rewritten in the form

$$0 = \langle E_i^n(U_{i-1}^n, U_i^n, U_{i+1}^n), \psi_{\alpha} \rangle \quad \text{for all spatial test-functions } \psi_{\alpha}, \alpha \in \{0, \dots, N\}^d,$$

where the operator $E_i^n(\cdot, \cdot, \cdot)$ has the matrix decomposition (5.20) and the vectors $U_i^n \in \mathbb{R}^{Nd}$ correspond to the frames $(u_{\rho,h}^n)_i$ of the discrete function $(u_{\rho,h}^n)$. The successive solving of the systems E_i^n for $i = 0, \dots, M$ is — figuratively speaking — the movement of a 3-frame window over the whole sequence (cf. Figure 5.5).

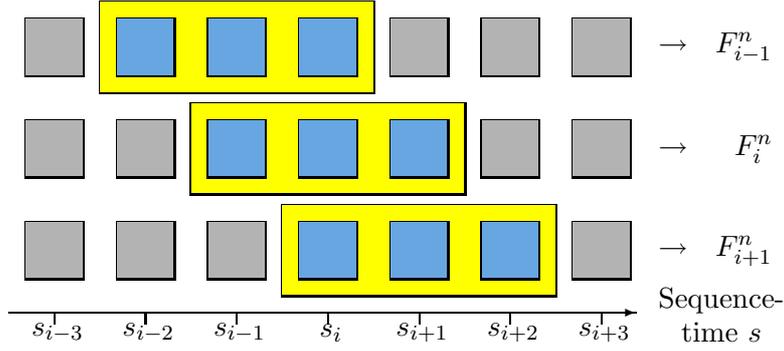
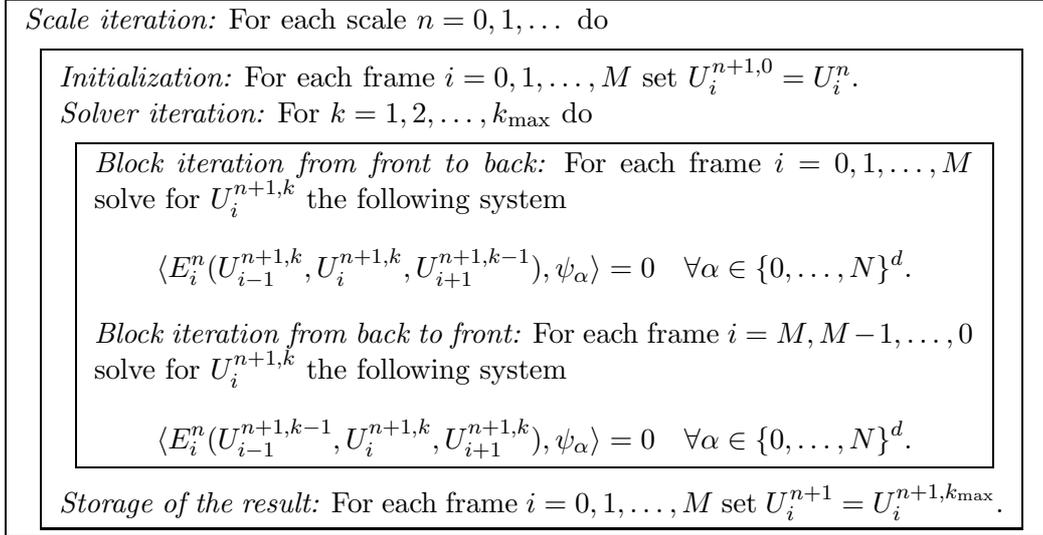


Figure 5.5: In each step of the inner frame loop, the block solver always considers only three successive images of the sequence. This corresponds to the fact that the resulting system matrix has a 3-band block-structure.

Finally, we use the following symmetric block Gauß-Seidel solver, in which we consider the appropriate terms to be handled implicitly respectively explicitly with respect to the iteration cycle. Schematically the block-solver can be sketched as follows:



In this scheme we have to respect the temporal boundary conditions. If we have prescribed the Neumann boundary conditions in sequence-time (BC1), we set

$$U_{-1}^n = U_0^n \quad \text{and} \quad U_{M+1}^n = U_M^n$$

to evaluate the previous and next image-frame at the beginning and at the end of the sequence. In case of the periodic boundary conditions in sequence-time (BC2), we set

$$U_{-1}^n = U_M^n \quad \text{and} \quad U_{M+1}^n = U_0^n.$$

Running the above scheme with $k_{\max} = 1$ one obtains a solver which is explicit in sequence-time, because the coupling between successive frames is not considered any more. This is not desirable since, obviously the strength of the evolution model lies in the nonlinear sequence-time behavior. A good compromise between effort and implicitness of the scheme in sequence-time is the fixation of a small k_{\max} . In the applications shown in Figures 2.14, 2.15, and 2.16 we always choose $k_{\max} = 3$. Within the block Gauß-Seidel solver the solution of the subsystems E_i^n is done by a conjugate gradient (CG) method, which is preconditioned by diagonal scaling.

Chapter 6

Conclusions

IN THIS THESIS anisotropic geometric diffusion models for the processing of images and image-sequences have been discussed. The diffusion for static images is driven by the intrinsic geometric quantities of the level-sets of the images: The shape operator characterizes the corners and edges of the level-sets in terms of its eigenvalues – the principal curvatures – and the corresponding eigenvectors – the principal directions of curvature. The presence of such a surface feature is indicated by (at least) one high principal curvature. The orientation of an edge is given by the principal directions of curvature, which point across and along the edge, respectively.

In particular on noisy image data a regularization of these geometric quantities is needed to obtain a robust anisotropic model. The diffusion tensor for the static image processing model depends on a regularized shape operator. In coordinates of the regularized principal directions of curvature and the regularized normal it weights the directions nonlinearly according to the corresponding regularized principal curvature. The well known Perona-Malik edge indicator function is used for this weighting of the curvatures. The resulting model is capable of retaining corners and edges of level-sets while smoothing their geometric noise. It turns out that the diffusion is steered by the difference between regularized and true geometric quantities. Dependent on the regularization, this leads to a rich class of shapes which are invariant under the diffusion.

The model which has been presented for the diffusion of image-sequences shares this corner and edge preserving behavior. Moreover it takes into account the correlation of the level-sets between successive frames of the sequence. The extraction of this correlation is an ill-posed task. One possible regularization is the consideration of the apparent velocity. Of all possible solutions it is the one, which minimizes the variations of the normal along the trajectories of level-sets. The curvature of these apparent trajectories is measured by the apparent acceleration. In the image-sequence diffusion model this apparent acceleration is weighted by the Perona-Malik edge indicator function to distinguish uniform motions (which can be smoothed significantly) from highly accelerated motions (which must be kept). This temporal diffusion coefficient steers the anisotropic smoothing in direction of the apparent velocity. The resulting model is a coupled spatio/temporal diffusion, which smooths image-sequence data while retaining spatial edges and corners as well as highly accelerated motions. Obviously on noisy image data a suitable spatio-temporal regularization is indispensable.

Different local and global variants for the regularization of the image data have been presented in this work. Since the definition of the shape operator involves higher order derivatives on low-regularity image data, it is convenient to combine the regularization method with a consistent definition of the necessary spatial derivatives of the regularized data. The convolution of the data with the derivatives of compactly supported smoothing kernels is one possible regularization variant. A different approach utilizes local L^2 -projections onto polynomial spaces. A comparison of the two approaches on test data-sets has shown that the

projection regularization behaves much more robust in the presence of noise. However the convolution approach is much faster in terms of computing time, since in an implementation it results in the computation of weighted sums. Both approaches however deliver a consistent approximation of the curvature of data-sets whose level-sets are spheres. Thereby, the approximation is of second order, i.e. a doubling of the stencil width of the regularization reduces the difference between real and computed curvature by a factor $1/4$.

For the geometric diffusion models related to the two approaches we have proven the existence of viscosity solutions on unbounded domains. Since the problems are undefined if the definition of a normal is not possible, semi-continuous relaxations had to be considered as common in the theory of viscosity solutions. The construction of one sub- and one super-solution then lead to a continuous solution via the Perron method by Ishii and a comparison principle by Giga et al. In a second step the continuity of the solution operator was shown. For a sequence of problems defined by a sequence of continuous functions the corresponding solutions converge uniformly toward the solution of the limit problem. Thereby, the semi-continuous limit problem and the limit solutions are the $\Gamma^{+/-}$ limits of the linear problems and their solutions. Since the solutions are bounded by the initial data, this continuity result allowed the application of the Schauder Fixed-Point Theorem to obtain a solution for the nonlinear problem.

The two models have been implemented using finite elements on quadtree- and octtree-grids, respectively. For the time stepping an semi-implicit scheme was employed, which treats the nonlinearities at the old time-step. The Euclidean norm in the denominator was regularized following Evans and Spruck. For the static image model the resulting system of equations is solved by a diagonally preconditioned CG method. But in the case of the image-sequence model, which is a 5D problem for (3+1)D image-sequences, the solution of such a system resulting from space/time finite elements involves huge computational efforts. Therefore a suitable operator splitting scheme has been discussed, which simplifies the system of equations significantly. The resulting final system has a 3-band block-structure and is very similar to a mixed finite difference/finite element scheme. Since each block corresponds to one image of the sequence, the system can be solved by a symmetric block Gauß-Seidel solver, which loops over the separate frames. To keep the effort reasonable only a few steps of this block solver are performed.

Future work directions in the theoretical analysis of the models include the convergence of the finite element scheme. Concerning the discretization of the model, the consideration of faster regularizations based on local projections and adaptive schemes is needed for larger data-sets. Since this thesis has shown that curvature based diffusion models in level-set formulation yield good results for the de-noising of images, the extension of these models to other areas of image processing is a future research direction. Ongoing work is being done for example in the area of *image in-painting*, where destroyed areas of images (i.e. areas on which the image is completely lost) are to be restored.

Bibliography

- [1] H. W. Alt. *Lineare Funktionalanalysis*. Springer, 1985.
- [2] L. Alvarez. Images and PDE's. In M.-O. Berger, R. Deriche, I. Herlin, J. Jaffré, and J.-M. Morel, editors, *ICAOS '96: Images, wavelets and PDEs*, Lecture notes in Control and Information Sciences, pages 3–14. Springer, London, 1996.
- [3] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel. Axioms and fundamental equations of image processing. *Arch. Ration. Mech. Anal.*, 123(3):199–257, 1993.
- [4] L. Alvarez and L. Mazorra. Signal and image restoration using shock filters and anisotropic diffusion. *SIAM J. Numer. Anal.*, 31:590–605, 1994.
- [5] L. Alvarez, J. Weickert, and J. Sánchez. A scale-space approach to nonlocal optical flow calculations. In *Scale-Space Theories in Computer Vision, Proceedings of Scale-Space '99*, Lecture Notes in Computer Science, pages 235–246. Springer, 1999.
- [6] J. Babaud, A. P. Witkin, B. Baudin, and R. O. Duda. Uniqueness of the Gaussian kernel for scale-space filtering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8:26–33, 1986.
- [7] E. Bänsch and K. Mikula. A coarsening finite element strategy in image selective smoothing. *Computing and Visualization in Science*, 1:53–63, 1997.
- [8] G. Barles and B. Perthame. Discontinuous solutions of deterministic optimal stopping time problems. *Modél. Math. Anal. Num.*, 21:557–579, 1987.
- [9] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proceedings IEEE Comp. Soc. Conf. on Comp. Vis. and Patt. Rec. (CVPR '91)*, 1991.
- [10] D. Braess. *Finite Elemente*. Springer, 1992.
- [11] A. Braides. *Gamma - convergence for beginners*. Oxford Lecture Series in Mathematics and its Application, 2002.
- [12] S. Brenner and L. Scott. *The mathematical Theory of Finite Element Methods*. Springer, 1994.
- [13] M. Bro-Nielsen and C. Gramkow. Fast fluid registration of medical images. *Lecture Notes in Computer Science*, 1131:267–276, 1996.
- [14] R. A. Carmona and S. Zhong. Adaptive smoothing respecting feature directions. *IEEE Transactions on Image Processing*, 7(3):353–358, 1998.
- [15] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182–193, 1992.
- [16] Y.-G. Chen, Y. Giga, and S. Goto. Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations. *J. Diff. Geom.*, 33(3):749–786, 1991.

-
- [17] U. Clarenz, U. Diewald, and M. Rumpf. Nonlinear anisotropic diffusion in surface processing. In *Proc. Visualization 2000*, pages 397–405, 2000.
- [18] M. G. Crandall and H. Ishii. The maximum principle for semicontinuous functions. *Differential and Integral Equations*, 3:1001–1014, 1990.
- [19] M. G. Crandall, H. Ishii, and P. L. Lions. User’s guide to viscosity solutions of second order partial differential equations. *Bull. Am. Math. Soc.*, 27(1):1–67, 1992.
- [20] M. G. Crandall and P. L. Lions. Viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, 277:1–43, 1983.
- [21] F. De Giorgi. Gamma-convergenza e G-convergenza. *Boll. Un. Mat. Ital.*, 14-A(5):213–220, 1977.
- [22] K. Deckelnick and G. Dziuk. Discrete anisotropic curvature flow of graphs. *Mathematical Modelling and Numerical Analysis, RAIRO*, 33(6):1203–1222, 1999.
- [23] K. Deckelnick and G. Dziuk. A fully discrete numerical scheme for weighted mean curvature flow. Technical Report 30, Mathematische Fakultät Freiburg, 2000.
- [24] R. Deriche, P. Kornobst, and G. Aubert. Optical-flow estimation while preserving its discontinuities: A variational approach. In *Proc. Second Asian Conf. Computer Vision*, volume 2, pages 290–295, 1995.
- [25] P. Deuffhard and A. Hohmann. *Numerische Mathematik I*. de Gruyter, 1993.
- [26] U. Diewald, T. Preußner, and M. Rumpf. Anisotropic diffusion in vector field visualization on Euclidean domains and surfaces. *Trans. Vis. and Comp. Graphics*, 6(2):139–149, 2000.
- [27] M. P. do Carmo. *Differential Geometry*. Birkhäuser, 1993.
- [28] M. Droske and M. Rumpf. A variational approach to non-rigid morphological registration. *SIAM Appl. Math.*, 2002. submitted.
- [29] G. Dziuk. An algorithm for evolutionary surfaces. *Numer. Math.*, 58:601–611, 1991.
- [30] K. Ecker and G. Huisken. Mean curvature evolution of entire graphs. *Ann. of Math.*, 130:453–471, 1989.
- [31] W. Enkelmann. Investigation of multigrid algorithms for the estimation of optical flow fields in image sequences. *Computer Vision, Graphics and Image Processing*, 43:150–177, 1988.
- [32] L. Evans and J. Spruck. Motion of level sets by mean curvature I. *J. Diff. Geom.*, 33(3):635–381, 1991.
- [33] M. Ferrant, S. K. Warfield, C. R. Guttmann, R. V. Mulkern, F. A. Jolesz, and R. Kikinis. 3D image matching using a finite element based elastic deformation model. In *Proceedings Medical image computing and computer-assisted intervention - MICCAI 1999*, pages 202–209. Springer, 1999.
- [34] D. Gabor. Information theory in electron microscopy. *Lab. Invest.*, 14:801–807, 1965.
- [35] Y. Giga, S. Goto, H. Ishii, and M.-H. Sato. Comparison principle and convexity preserving properties for singular degenerate parabolic equations on unbounded domains. *Ind. Math. J.*, 40(2):443–470, 1991.

-
- [36] F. Guichard. *Axiomatisation des analyses multi-échelles d'images et de films*. PhD thesis, University Paris IX Dauphine, 1994.
- [37] F. Guichard. A morphological, affine, and galilean invariant scale-space for movies. *IEEE Transactions on Image Processing*, 7(3):444–456, 1998.
- [38] W. Hackbusch. *Multi-Grid Methods and Applications*. Springer, 1985.
- [39] A. Handlovičová, K. Mikula, and F. Sgallari. Semi-implicit complementary volume scheme for solving level set like equations in image processing and curve evolution. *Numer. Math.*, 2001. (DOI) 10.1007/s002110100374.
- [40] E. Harabetian and S. Osher. Regularization of ill-posed problems via the level set approach. *SIAM J. Appl. Math.*, 58(6):1689–1706, 1998.
- [41] B. Horn and B. Schunck. Determining optical flow. *Artif. Intell.*, 17:185–203, 1981.
- [42] H. Ishii. Perron's method for Hamilton-Jacobi equations. *Duke Math J.*, 55:369–384, 1987.
- [43] H. Ishii. A boundary value problem of the Dirichlet type for Hamilton-Jacobi equations. *Ann. Scuola Norm. Sup. Pisa Cl. Sci.*, 16(4):105–135, 1989.
- [44] H. Ishii. On uniqueness and existence of viscosity solutions of fully nonlinear second-order elliptic PDE's. *Comm. Pure Appl. Math.*, 42:14–45, 1989.
- [45] H. Ishii and P. L. Lions. Viscosity solutions of fully nonlinear second-order elliptic partial differential equations. *J. Differential Equations*, 21:26–78, 1990.
- [46] R. Jensen. The maximum principle for viscosity solutions of fully nonlinear second order partial differential equations. *Arch. Rat. Mech. Anal.*, 101:1–27, 1988.
- [47] S. C. Joshi and M. I. Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Trans. Medical Imaging*, 9(8):1357–1370, 2000.
- [48] J. Kačur and K. Mikula. Solution of nonlinear diffusion appearing in image smoothing and edge detection. *Appl. Numer. Math.*, 17 (1):47–59, 1995.
- [49] B. Kawohl and N. Kutev. Maximum and comparison principle for one-dimensional anisotropic diffusion. *Math. Ann.*, 311 (1):107–123, 1998.
- [50] J. J. Koenderink. The structure of images. *Biol. Cyber.*, 50:363–370, 1984.
- [51] Z. Kriva and K. Mikula. An adaptive finite volume scheme for solving nonlinear diffusion equations in image processing. *J. Vis. Comm. and Image repres.*, 13:22–35, 2002.
- [52] F. Lenzen. 3D Rekonstruktion von DNA Strukturen. Diplomarbeit Universität Bonn, 2001.
- [53] P. L. Lions. Optimal control of diffusion process and Hamilton-Jacobi-Bellman equations, Part II: Viscosity solutions and uniqueness. *Comm. Partial Differential Equations*, 8:1229–1276, 1983.
- [54] W. E. Lorensen and H. E. Cline. Marching cube: A high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4), 1987.

- [55] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multi-modal volume registration by maximization of mutual information. *IEEE Trans. Medical Imaging*, 16(7):187–198, 1997.
- [56] R. Malladi and J. A. Sethian. Image processing: Flows under min/max curvature and mean curvature. *Graphical Models and Image Processing*, 58(2):127–141, March 1996.
- [57] K. Mikula, T. Preußer, and M. Rumpf. Morphological image sequence processing. *Computing and Visualization in Science*, 2003. submitted.
- [58] K. Mikula, T. Preußer, M. Rumpf, and F. Sgallari. On anisotropic geometric diffusion in 3D image processing and image sequence analysis. In *Proc. Trends in Nonlinear Analysis, Heidelberg*, pages 305–319. Springer, 2002.
- [59] K. Mikula and N. Ramarosy. Semi-implicit finite volume scheme for solving nonlinear diffusion equations in image processing. *Numerische Mathematik*, 2001.
- [60] A. Mitchie and P. Bouthemy. Computation and analysis of image motion: a synopsis of current problems and methods. *Int. J. Comput. Vision*, 19:29–55, 1996.
- [61] S. J. Osher and J. A. Sethian. Fronts propagation with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. *J. Comp. Phys.*, 79:12–49, 1988.
- [62] S. Oswald. *Dichteströmungen in porösen Medien: Dreidimensionale Experimente und Modellierung*. PhD thesis, ETH, Zürich, Switzerland, 1999.
- [63] E. Pauwels, P. Fiddelaers, and L. van Gool. Enhancement of planar shape through optimization of functionals for curves. *IEEE Trans. Pattern Anal. Mach. Int.*, 17(1):1101–1105, 1995.
- [64] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12:629–639, 1990.
- [65] T. Preußer and M. Rumpf. Anisotropic nonlinear diffusion in flow visualization. In *Proc. Visualization 1999*, pages 325–332, 1999.
- [66] T. Preußer and M. Rumpf. An adaptive finite element method for large scale image processing. *Journal of Visual Comm. and Image Repres.*, 11:183–195, 2000.
- [67] T. Preußer and M. Rumpf. A level set method for anisotropic diffusion in 3D image processing. *SIAM J. Appl. Math.*, 62(5):1772–1793, 2001.
- [68] T. Preußer and M. Rumpf. Extracting motion velocities from 3D image sequences and coupled spatio-temporal smoothing. In *Proc. SPIE Visual Data Analysis*, 2003. to appear.
- [69] G. Sapiro. Vector (self) snakes: a geometric framework for color, texture, and multiscale image segmentation. In *Proc. IEEE Int. Conf. Image Processing (ICIP-96, Lausanne)*, 1996.
- [70] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, 2001.
- [71] A. Sarti, K. Mikula, and F. Sgallari. Nonlinear multiscale analysis of 3D echocardiography sequences. *IEEE Transactions on Medical Imaging*, 18(6):453–466, 1999.

-
- [72] M. Sato. Comparison principle for singular degenerate elliptic equations on unbounded domains. *Proc. Japan Acad. Ser. A*, 66:252–256, 1990.
- [73] R. Schaback and H. Werner. *Numerische Mathematik*. Springer, 1992.
- [74] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.
- [75] J. P. Thirion. Image matching as a diffusion process: An analogy with Maxwell’s demons. *Med. Imag. Anal.*, 2:243–260, 1998.
- [76] V. Thomée. *Galerkin Finite Element Methods for Parabolic Problems*. Springer, 1997.
- [77] P. A. Viola and W. Wells III. Alignment by maximization of mutual information. Technical Report AITR-1548, 1995.
- [78] N. J. Walkington. Algorithms for computing motion by mean curvature. *SIAM J. Numer. Anal.*, 33(6):2215–2238, 1996.
- [79] J. Weickert. Anisotropic diffusion filters for image processing based quality control. In A. Fasano and M. Primicerio, editors, *Proc. Seventh European Conf. on Mathematics in Industry*, pages 355–362. Teubner, 1994.
- [80] J. Weickert. *Anisotropic diffusion in image processing*. Teubner, 1998.
- [81] A. P. Witkin. Scale-Space filtering. In *Proc. International Joint Conference on Artificial Intelligence*, pages 1019–1021. ACM Inc. New York, 1983.
- [82] S. W. Zucker, A. Dobbins, L. Iverson, B. Kimia, and A. Tannenbaum. A new implementation of discrete multiscale filtering. In *Proc. IEEE Int. Conf. Image Processing (ICIP-96, Lausanne)*, volume 1, pages 383–386, 1996.