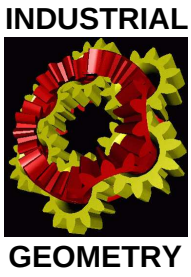
National Research Network S92

Industrial Geometry

http://www.industrial-geometry.at



NRN Report No. 85

Morphological Partial Differential Equations for Optical Flow Computations and Applications to Image Interpolation

Frank Lenzen and Otmar Scherzer

May 2009

FWF
Der Wissenschaftsfonds.

universität innsbruck

IJCV manuscript No.
(will be inserted by the editor)

Morphological Partial Differential Equations for Optical Flow Computations and Applications to Image Interpolation

Frank Lenzen · Otmar Scherzer

Received: date / Accepted: date

Abstract In this paper, for imaging applications, we introduce partial differential equations, which allow for correcting for displacement errors in multi-channel data. These equations are derived via a semi-group for a non–convex energy functional. For gray valued data, it turns out that one of the equations develops optical flow displacements according to mean curvature in normal direction of the level lines. Therefore, the paper links morphological partial differential equations (such as the mean curvature motion), non-convex variational principles, and the optical flow. As a further application for correction of displacement errors we study image interpolation of digital color images. For actual image interpolation, the solutions of the suggested PDEs are projected onto a space of functions satisfying interpolation constraints. A comparison of the test results with standard and state-of-the-art interpolation algorithms shows the competitiveness of this approach.

Keywords Optical Flow · Morphological Partial Differential Equations · Image Interpolation

Frank Lenzen
Department of Mathematics, University of Innsbruck,
Technikerstraße 21a, A-6020 Innsbruck, Austria,
Tel.: +43-512-507-6115
Fax: +43-512-507-2758
E-mail: Frank.Lenzen@uibk.ac.at

Otmar Scherzer
Department of Mathematics, University of Innsbruck,
Technikerstraße 21a, A-6020 Innsbruck, Austria,
Tel.: +43-512-507-6118
Fax: +43-512-507-2758
Johann Radon Institute for Computational and
Applied Mathematics (RICAM),
Austrian Academy of Sciences,
Altenbergerstraße 69, A-4040 Linz, Austria
E-mail: Otmar.Scherzer@oeaw.ac.at

1 Introduction

In this paper, for imaging applications, we introduce partial differential equations, which allow for correcting displacement errors in multi-channel data, such as for instance color images. This work is motivated by the following image acquisition model: Data $\mathbf{u}^{(0)}$ of some $\mathbf{u}$ are given, which are obtained by a (random) displacement perturbation. That is $\mathbf{u}^{(0)}$ and $\mathbf{u}$ satisfy

$$\mathbf{u}^{(0)} = \mathbf{u} \circ \mathbf{\Phi}\,, \tag{1}$$

where $\mathbf{\Phi} : \Omega \to \Omega$ is a displacement vector field.

For filtering such data $\mathbf{u}^{(0)}$ novel variational methods and partial differential equations (PDEs) are derived and numerically studied. We show that the derived variational equations are morphological and the simultaneously evolving displacement fields are the optical flows, which evolve according to the mean curvature in direction of the normals to the level lines.

The second objective is to apply the partial differential equations for image *interpolation*, which we refer to as the process of assigning a discrete set of pixel positions and according discrete multi-channel image data an interpolating *function*. Interpolation is frequently used for *zooming into* or *scaling* digital images. A special kind of image interpolation problems is *inpainting*, i.e., the problem of reconstructing lost or corrupted parts of images.

Linear interpolation (that is convolution methods) [18], such as for example nearest neighbor, spline, Whittaker-Shannon and the Lanczos interpolation [14,4], is computationally efficient but produce unpleasant artifacts. On the other hand, *nonlinear* methods adapting to geometrical structures can produce more visually attractive results but are computationally more demanding.

Nowadays, most of these nonlinear methods are motivated by energy minimization or by scale spaces of partial differential equations, see for example [1,22,21, 18]. In particular for inpainting such nonlinear methods are widely used, see for example [2,5,6,23].

In this paper we derive partial differential equations that are designed to correct and filter for displacement errors in multi-channel data. When combined with the interpolation ideas of [11,16], these PDEs are very well suited for image interpolation. In Section 7 we compare results of the proposed methods with interpolation methods from the scale space literature. In particular we take into account the GREYCstoration software of Tschumperlé [21] and the interpolation method proposed by Roussos & Maragos [18,19].

The paper is organized as follows: In Section 2 we present the connection between optical flow and displacement errors. In Section 3 we consider a variational ansatz for correcting displacement errors. Application of the semi-group concepts (see Section 4) yields PDEs, which can be considered the gradient flows of the variational problems. Moreover, a relationship of our PDEs to the *Mean Curvature Flow* (MCF) equation is established. Section 5 clarifies the relation of the partial differential equations to optical flow displacement. The application of the proposed PDEs to image interpolation, i.e. combining our approach with interpolation constraints, is described in Section 6. Numerical tests for image interpolation are shown in Section 7. The paper ends with a conclusion in Section 8.

2 Displacement Errors and the Optical Flow Problem

Let $\mathbf{u} : \Omega \to \mathbb{R}^M$ be an multi-channel function representing *continuous* multi-channel data on a bounded open domain $\Omega \subseteq \mathbb{R}^n$.

In the following we assume that $\mathbf{u}$ is a smooth function, so that we can make a first order Taylor series expansion. Then it follows from our modeling assumptions that

$$\begin{aligned}\mathbf{u}^{(0)}(\mathbf{x}) &= (\mathbf{u}\circ\mathbf{\Phi})(\mathbf{x}) = \mathbf{u}(\mathbf{x}+(\mathbf{\Phi}(\mathbf{x})-\mathbf{x}))\\ &\approx \mathbf{u}(\mathbf{x}) + \nabla\mathbf{u}^T(\mathbf{x})\,(\mathbf{\Phi}(\mathbf{x})-\mathbf{x})\,.\end{aligned} \tag{2}$$

Here

$$\nabla\mathbf{u} = \begin{pmatrix}\partial_1 u_1 & \partial_1 u_2 & \partial_1 u_3\\ \partial_2 u_1 & \partial_2 u_2 & \partial_2 u_3\end{pmatrix},$$

and $\approx$ symbolizes that the left hand side of (2) approximates the right hand side for small displacements $\mathbf{\Phi}-\mathrm{Id}$. In the following we assume that equality holds instead of $\approx$, which implicitly implies that only small displacements are relevant.

Remark 1 We emphasize that in the case of scalar data (that is, for $M = 1$) (2) is the standard *optical flow* equation:

$$\frac{\partial\mathbf{u}}{\partial t}(\mathbf{x},t) + (\nabla\mathbf{u})^T(\mathbf{x},t)(\mathbf{\Phi}(\mathbf{x},t)-\mathbf{x}) = 0\,. \tag{3}$$

The relation is immediate if one considers $\mathbf{\Phi}-\mathrm{Id}$ the optical flow and $\mathbf{u}-\mathbf{u}^{(0)}$ is considered a time discretization of a time dependent function $\mathbf{u}(t)$.

Equation (2) shows a different behavior for scalar valued functions and multi-channel data. In the scalar case ($M = 1$) the system is *under*determined, and aside at positions, where $\nabla\mathbf{u}$ degenerates, solvable. For $\Omega = \mathbb{R}^d$ and $M > d$ the system is *over*determined, in which case it is seems necessary to consider minimizers of the least squares problem,

$$\left|\nabla\mathbf{u}^T(\mathbf{x})(\mathbf{\Phi}(\mathbf{x})-\mathbf{x}) - \mathbf{u}^{(0)}(\mathbf{x}) + \mathbf{u}(\mathbf{x})\right|^2, \tag{4}$$

instead. A minimizer of (4) is given by

$$\mathbf{\Phi}(\mathbf{x}) - \mathbf{x} = (\nabla\mathbf{u}^T(\mathbf{x}))^\dagger(\mathbf{u}^{(0)}(\mathbf{x}) - \mathbf{u}(\mathbf{x})), \tag{5}$$

where $(\nabla\mathbf{u}^T(\mathbf{x}))^\dagger$ denotes the *Moore–Penrose pseudo–inverse* (see [17]) of $\nabla\mathbf{u}^T(\mathbf{x})$. The above ansatz, to use the pseudo–inverse to estimate a displacement $\mathbf{\Phi}$ from a given pair of images, provides already a simplistic solution to the optical flow problem (see Figure 1). However, this approach is only capable to capture *small* displacement and very sensitive to noise.

3 Displacement Regularization

In what follows, we consider the problem of finding the pairs $(\mathbf{u},\mathbf{\Phi})$ of minimal energy satisfying (1). For solving this problem approximately, we use a variational method consisting in minimization of the functional (with small positive α)

$$\frac{1}{2}\int_\Omega |\mathbf{\Phi}(\mathbf{x})-\mathbf{x}|^2\,d\mathbf{x} + \alpha\int_\Omega |\nabla\mathbf{u}(\mathbf{x})|\,d\mathbf{x} \tag{6}$$

over the set of functions satisfying $\mathbf{u}^{(0)} = \mathbf{u}\circ\mathbf{\Phi}$. Here

$$|\nabla\mathbf{u}(\mathbf{x})| = \left(\sum_{j=1}^{M}\sum_{i=1}^{2}(\partial_i u_j(\mathbf{x}))^2\right)^{1/2}.$$

In the following, for notational convenience, we leave out the space dependency and write $\mathbf{u}$ instead of $\mathbf{u}(\mathbf{x})$. Inserting (5) into (6) gives the functional

$$\begin{aligned}&\mathcal{F}^0_{\mathbf{u}^{(0)}}(\mathbf{u})\\ &\quad:= \frac{1}{2}\int_\Omega (\mathbf{u}-\mathbf{u}^{(0)})^T((\nabla\mathbf{u})^T\nabla\mathbf{u})^\dagger(\mathbf{u}-\mathbf{u}^{(0)})\\ &\qquad + \alpha\,|\nabla\mathbf{u}|\,d\mathbf{x}\,.\end{aligned} \tag{7}$$

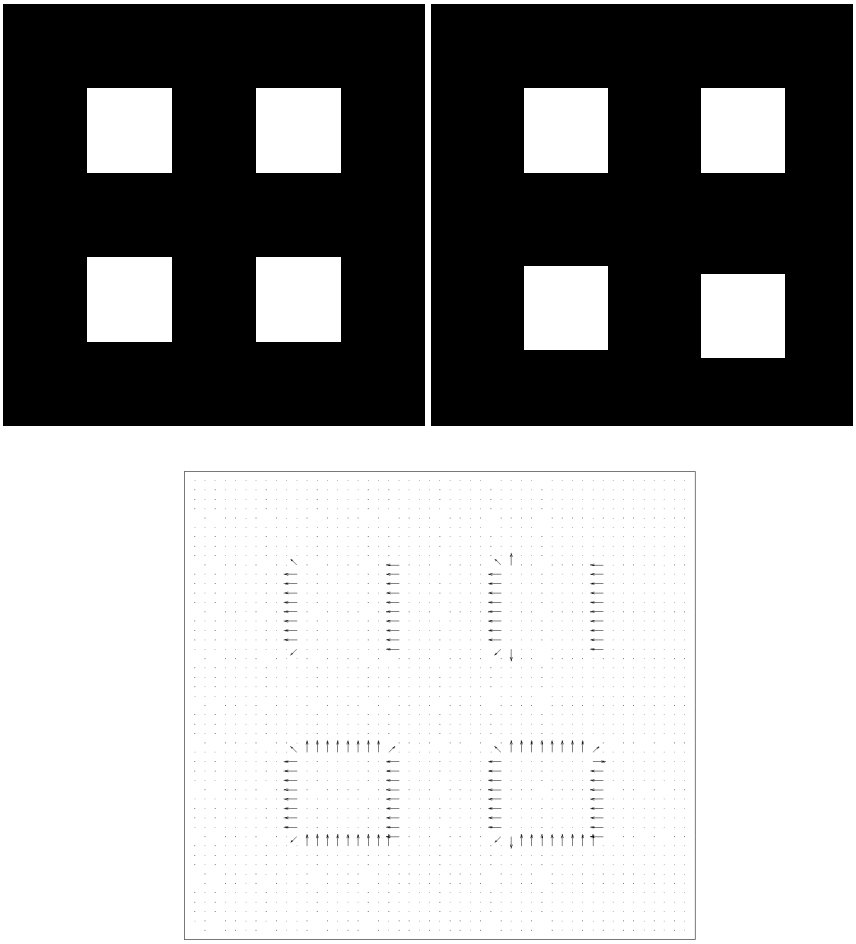


Fig. 1 Test example for computing displacements via pseudo–inverse. Top left: Test image with four objects. Top right: image with shifted objects. The top objects are shifted to the right by one and two pixels, respectively. The bottom objects are shifted diagonally with a displacement of length $\sqrt{2}$ and $\sqrt{8}$. Bottom: displacement calculated using the Moore–Penrose pseudo–inverse, see (5).

In order to avoid computation of the pseudo–inverse, we additionally regularize the probably singular matrix $(\nabla \mathbf{u})^T \nabla \mathbf{u}$ by the regular, symmetric, and strictly positive definite matrix $(\varepsilon I + (\nabla \mathbf{u})^T \nabla \mathbf{u})$ with some $\varepsilon > 0$. Therefore, in the sequel, we consider minimizing of the regularized functional

$$\begin{aligned} &\mathcal{F}^{\varepsilon}_{\mathbf{u}^{(0)}}(\mathbf{u}) \\ &:= \frac{1}{2} \int_{\Omega} (\mathbf{u} - \mathbf{u}^{(0)})^T (\varepsilon I + (\nabla \mathbf{u})^T \nabla \mathbf{u})^{-1} (\mathbf{u} - \mathbf{u}^{(0)}) \\ &\qquad + \alpha \, |\nabla \mathbf{u}| \, d\mathbf{x} \, . \end{aligned} \tag{8}$$

For this functional, the existence theory within the classical framework of the *Calculus of Variations* [7,8] is not applicable. In addition, for a theoretical analysis a further complication is due to the fact that minimization has to be considered over the space of multichannel functions with components of finite total variation. In order to implement the minimization of $\mathcal{F}^{\varepsilon}_{\mathbf{u}^{(0)}}$ numerically, quasi-convexification techniques would be most efficient. This approach requires the analytical calculation of the quasi-convex envelope of the function

$$(\mathbf{x}, \xi, \nu) \to \frac{1}{2} (\xi - v(\mathbf{x}))^T (\varepsilon I + \nu^T \nu)^{-1} (\xi - v(\mathbf{x})) + \alpha \, |\nu|$$

with respect to ν. However, for $M > 1$ or $n > 1$, the quasi-convex envelope function is not known so far, and thus efficient numerical minimization based on this approach is not at hand. For $M = 1$ (that is for gray valued data) and for $n = 1$ the quasi-convexification equals the convexification and can be calculated analytically.

4 Semi-Group Solution Concept

In the following we recall the *convex semi-group solution concept* [3]: Let $\mathcal{R} : H \to \mathbb{R} \cup \{\infty\}$ be a convex

functional on a Hilbert space H, and let $\mathbf{u}_\alpha$ be a minimizer of the variational regularization functional

$$\mathcal{G}_{\mathbf{u}^{(0)}}(\mathbf{u}) := \frac{1}{2}\left\|\mathbf{u}-\mathbf{u}^{(0)}\right\|_H^2 + \alpha\mathcal{R}(\mathbf{u})\ .$$

Then, for a smooth initial function $\mathbf{u}^{(0)}$ and $\alpha \to 0$, $(\mathbf{u}_\alpha - \mathbf{u}^{(0)})/\alpha$ converges to an element of the subgradient $\partial\mathcal{R}(\mathbf{u}^{(0)})$ of $\mathcal{R}$. Choosing $\mathbf{u}^{(k)} \in \operatorname{argmin}\mathcal{G}_{\mathbf{u}^{(k-1)}}$, iterative minimization of $\mathcal{G}_{\mathbf{u}^{(k)}}$ yields an approximation of the solution of the flow

$$\partial_t\mathbf{u} \in \partial\mathcal{R}(\mathbf{u}) \text{ at scale } t = k\alpha\ .$$

In other words, variational regularization approximates a diffusion filtering scale space, which is the associated gradient flow equation. For convex semi-groups the solutions of diffusion filtering and variational methods are comparable and look rather similar [20].

We expect a similar behavior for the non-convex functional $\mathcal{F}^\varepsilon_{\mathbf{u}^{(0)}}$ and derive the according flow equation, which is the gradient flow associated with (8). We use the abbreviations

$$A^\varepsilon(\mathbf{u}) := \left(\varepsilon I + (\nabla\mathbf{u})^T\nabla\mathbf{u}\right)^{-1}\ ,$$

and

$$S^\varepsilon_{(k-1)}(\mathbf{u}) := \frac{1}{2}\int_\Omega (\mathbf{u}-\mathbf{u}^{(k-1)})^T A^\varepsilon(\mathbf{u})(\mathbf{u}-\mathbf{u}^{(k-1)})\,d\mathbf{x}\ .$$

The *directional derivative* of $S^\varepsilon_{(k-1)}$ at $\mathbf{u}$ in direction ϕ (provided it exists) satisfies

$$\begin{aligned}\partial_\tau S^\varepsilon_{(k-1)}(\mathbf{u}+\tau\phi) &= \int_\Omega \phi^T A^\varepsilon(\mathbf{u})(\mathbf{u}-\mathbf{u}^{(k-1)})\,d\mathbf{x} \\ &\quad + \frac{1}{2}\int_\Omega (\mathbf{u}-\mathbf{u}^{(k-1)})^T\partial_\phi A^\varepsilon(\mathbf{u})\,(\mathbf{u}-\mathbf{u}^{(k-1)})\,d\mathbf{x}\,,\end{aligned} \tag{9}$$

where

$$\partial_\phi A^\varepsilon(\mathbf{u}) := \lim_{\tau\to 0}\frac{A^\varepsilon(\mathbf{u}+\tau\phi)-A^\varepsilon(\mathbf{u})}{\tau}.$$

In a similar, formal manner, the directional derivative of $\mathcal{R}(\mathbf{u}) = \int_\Omega |\nabla\mathbf{u}|$ at $\mathbf{u}$ in direction ϕ can be derived and reads as

$$\partial_\tau\mathcal{R}(\mathbf{u}+\tau\phi) = \int_\Omega \nabla\phi^T\frac{\nabla\mathbf{u}}{|\nabla\mathbf{u}|}\,d\mathbf{x}. \tag{10}$$

Note, that by the right hand side of (10) actually the subdifferential of the total variation semi-norm evaluated in direction ϕ is meant.

Then, using (9) and (10), with $\mathbf{v}^{(k)} := \mathbf{u}^{(k)} - \mathbf{u}^{(k-1)}$, gives the optimality condition for the minimizer $\mathbf{u}^{(k)}$ of $\mathcal{F}^\varepsilon_{\mathbf{u}^{(k-1)}}$:

$$\begin{aligned}&\int_\Omega \phi^T A^\varepsilon(\mathbf{u}^{(k)})\frac{\mathbf{v}^{(k)}}{\alpha}\,d\mathbf{x} \\ &\quad + \frac{1}{2}\int_\Omega \frac{(\mathbf{v}^{(k)})^T}{\alpha}\partial_\phi A^\varepsilon(\mathbf{u}^{(k)})\,\mathbf{v}^{(k)}\,d\mathbf{x} \\ &= \int_\Omega \nabla\phi^T\frac{\nabla\mathbf{u}^{(k)}}{\left|\nabla\mathbf{u}^{(k)}\right|}\,d\mathbf{x} \qquad \text{for all } \phi\ .\end{aligned} \tag{11}$$

Let $t > 0$ be fixed and $k = \lfloor t/\alpha \rfloor$, then, as in the convex case, we can expect that $\mathbf{v}^{(k)}/\alpha$ converges to $\partial_t\mathbf{u}(t)$ for $\alpha \to 0$. From that, in turn, it follows that $\mathbf{v}^{(k)} \to 0$. This together with (11) then shows that

$$\int_\Omega \phi^T A^\varepsilon(\mathbf{u}(t))\partial_t\mathbf{u}(t)\,d\mathbf{x} = -\int_\Omega \nabla\phi^T\frac{\nabla\mathbf{u}(t)}{|\nabla\mathbf{u}(t)|}\,d\mathbf{x}. \tag{12}$$

Applying, in a formal way, Green's formula and the fundamental lemma, from (12) the strong formulation is derived and reads as

$$A^\varepsilon(\mathbf{u}(t))\partial_t\mathbf{u}(t) = \nabla\cdot\left(\frac{\nabla\mathbf{u}(t)}{|\nabla\mathbf{u}(t)|}\right)\ , \tag{13}$$

where $\mathbf{u}(t)$ satisfies natural (Neumann) boundary conditions.

In the following, again for notational convenience, we avoid the dependence of $\mathbf{u}$ with respect to t as well. Multiplying both sides of (13) by $A_\varepsilon(\mathbf{u})^{-1}$, we get

$$\partial_t\mathbf{u} = (\varepsilon I + (\nabla\mathbf{u})^T\nabla\mathbf{u})\ \nabla\cdot\left(\frac{\nabla\mathbf{u}}{|\nabla\mathbf{u}|}\right). \tag{14}$$

Moreover, the initial condition associated with the flow is $u(0) := u^{(0)}$. Now, letting $\varepsilon \to 0$, which only seems to make sense mathematically if $M \le 2$, we obtain the evolutionary partial differential equation

$$\partial_t\mathbf{u} = ((\nabla\mathbf{u})^T\nabla\mathbf{u})\ \nabla\cdot\left(\frac{\nabla\mathbf{u}}{|\nabla\mathbf{u}|}\right). \tag{15}$$

Remark 2 For scalar data ($M = 1$) the equation (15) reads as

$$\partial_t u = |\nabla u|^2\ \nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right). \tag{16}$$

One recognizes that (16) differs from the Mean Curvature Flow equation

$$\partial_t u = |\nabla u|\ \nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right) \tag{17}$$

only by the leading factor.

Let $p \geq 0$. In the following we consider a generalized functional of (7), which is defined by

$$\mathcal{F}^{0,p}(\mathbf{u}) := \frac{1}{2}\int_\Omega (\mathbf{u}-\mathbf{u}^{(0)})^T \left(((\nabla \mathbf{u})^T \nabla \mathbf{u})^p\right)^\dagger (\mathbf{u}-\mathbf{u}^{(0)}) + \alpha\,|\nabla \mathbf{u}|\; d\mathbf{x} \tag{18}$$

We note that the power of a matrix is defined via spectral decomposition. Of particular interest is the case $p = 1/2$, because then the functional (18) is invariant under affine rescaling of the image intensity. Moreover, the semi-group approach (see also [10] for the scalar case) results in the gradient flow

$$\partial_t \mathbf{u} = ((\nabla \mathbf{u})^T \nabla \mathbf{u})^{1/2}\; \nabla \cdot \left(\frac{\nabla \mathbf{u}}{|\nabla \mathbf{u}|}\right), \tag{19}$$

which, in the scalar case, is exactly the *Mean Curvature Flow* (MCF) equation. For scalar, radial-symmetric monotonous data an analytical comparison of the solutions of (18) and the MCF equation has been given in [9].

5 Mean Curvature Dependent Optical Flow

Here, we concentrate on the case $M = 1$, i.e., on scalar valued images. We recall (3), which states that the optical flow $\mathbf{\Phi} - \mathrm{Id}$ according to $u(t)$ satisfies

$$\mathbf{\Phi} - \mathrm{Id} = (\nabla u)^{T\,\dagger}(-\partial_t u)\,.$$

Now, let u be the solution of (16). Taking into account that the pseudo–inverse of $(\nabla u)^T$ is $\nabla u / |\nabla u|^2$, it follows from (16) that

$$\begin{aligned}\mathbf{\Phi} - \mathrm{Id} &= -\frac{\partial_t u}{|\nabla u|^2}\nabla u \\ &= -|\nabla u| \underbrace{\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right)}_{=:\kappa} \frac{\nabla u}{|\nabla u|}\,.\end{aligned} \tag{20}$$

This shows that the optical flow according to the solution of (16) develops according to the mean curvature κ in normal direction to the level lines of u. To illustrate this fact numerically, we calculate the optical flow displacement from two subsequent time frames of the solution of (16) (see Fig. 2 top row) using (20). The result is depicted in Fig. 2, bottom row.

For comparison, for the Mean Curvature Flow equation (17) the level set of u evolve according to mean curvature.

6 Interpolation of Multi Channel Data

The evolution equations (14) and (19) (with regularization of the pseudo–inverse) can be used for interpolating discrete multi-channel data by restricting $\mathbf{u}$ to satisfy interpolation constraints. In this section we concentrate on the PDE (14).

The problem of interpolating multi-channel data has already been studied in the literature before, see for example [1,21,18,19]. The difference between the approaches in [21,18,19] and ours are the different partial differential equations used for filtering: [21,18,19] use *anisotropic diffusion*, whereas the PDE (14) generalizes the *Mean Curvature Flow* equation (recall that the equation (15) is the asymptotic limit).

To begin with, we recall the interpolation constraints proposed in [11,16]. For the simplicity of notation we restrict ourself to multi-channel data defined on a two-dimensional rectangular domain

$$\Omega := \left(\frac{1}{2}, N_x + \frac{1}{2}\right) \times \left(\frac{1}{2}, N_y + \frac{1}{2}\right),$$

where $N_x, N_y \in \mathbb{N}$. The domain is partitioned into cells ('pixels')

$$Q_{i,j} := \left(i - \frac{1}{2}, i + \frac{1}{2}\right) \times \left(j - \frac{1}{2}, j + \frac{1}{2}\right)$$

for $(i,j) = (1,1), (1,2), \ldots, (N_x, N_y)$. Let G be a kernel function G defined on $\mathbb{R}^2$ and compactly supported in $[-\frac{1}{2}, \frac{1}{2}]^2$. Let $\mathbf{Z} := (z_{m,i,j})$ be a tensor, which denotes sampled data of a function $G * \mathbf{u} : \mathbb{R}^2 \to \mathbb{R}^M$ at the positions (i,j). Here $*$ denotes the convolution operator. In particular:

$$z_{m,i,j} := (G * u_m)(i,j), \tag{21}$$

for $(m,i,j) = (1,1,1), (1,1,2) \ldots, (M, N_x, N_y)$. Examples for kernel functions typically used in literature are listed in [18].

We rewrite (21) as follows: Let $G_{i,j} := G(\cdot - (i,j))$, then for each $(m,i,j) = (1,1,1), \ldots, (M, N_x, N_y)$

$$z_{m,i,j} = \langle G_{i,j}, u_m \rangle_{L^2(\Omega)}\,,$$

where

$$\langle u, v \rangle_{L^2(\Omega)} = \int_\Omega u \cdot v$$

is the L^2 inner product.

We say that an multi-channel function

$$\mathbf{u} = (u_1, \ldots, u_M)^T$$

satisfies the interpolation constraints for some discrete data $\mathbf{Z} = (z_{m,i,j})$, if

$$\langle G_{i,j}, u_m \rangle_{L^2(\Omega)} = z_{m,i,j}.$$

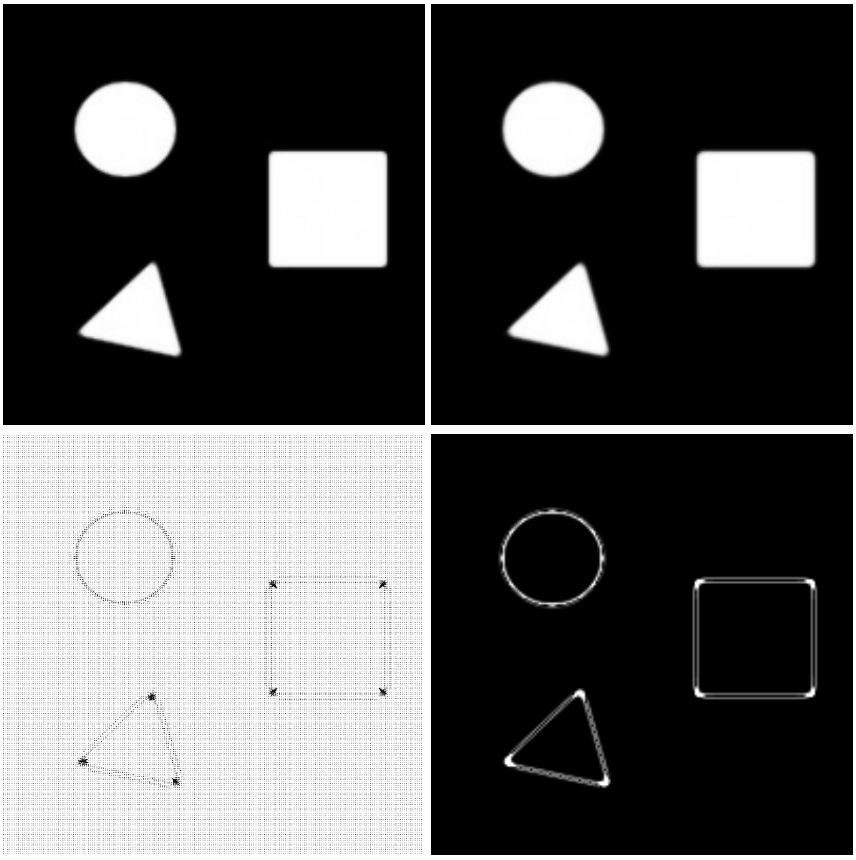


Fig. 2 Test example for computing displacements of the gradient flow (16). Top: solution of the gradient flow at $t_1 = 20$ (left) and $t_2 = 30$ (right). Bottom left: displacement field calculated via (20) (vectors were upscaled for accentuation). The figure shows that the displacement field is pronounced at regions of high curvature. Bottom right: absolute value of the displacement field.

The set of functions satisfying the interpolation constraints for data $\mathbf{Z}$ is denoted by $\mathcal{U}_{\mathbf{Z},G}$.

Example 1 We consider for G the two-dimensional δ distribution, i.e.,

$$G(x,y) = \delta(x)\delta(y).$$

Then $z_{m,i,j} = u_m((i,j))$. The *nearest neighbor (componentwise, piecewise constant) interpolation* reads as

$$u_m^{(0)}|_{Q_{i,j}} = z_{m,i,j}, \quad (m,i,j) = (1,1,1), \ldots, (M, N_x, N_y)\ .$$

Here, $\mathbf{u}^{(0)} = \mathbf{u} \circ \mathbf{\Phi}$ is satisfied for

$$\mathbf{\Phi}(x,y)|_{Q_{i,j}} := (i,j).$$

In particular $\mathbf{u}$ can be interpreted as a distortion of $\mathbf{u}^{(0)}$ by a local sampling displacement $\mathbf{\Phi}$.

Now let $\mathbf{u}^{(0)} \in \mathcal{U}_{\mathbf{Z},G}$ be arbitrary. The nearest neighbor interpolation in Example 1 motivates the assumption that, for a sampled function $\mathbf{u}$, there exists $\mathbf{\Phi}$ such that $\mathbf{u}^{(0)} = \mathbf{u} \circ \mathbf{\Phi}$. Recalling the concepts presented in Section 2 we consider the functional defined in (8) restricted to the set $\mathcal{U}_{\mathbf{Z},G}$ in order to reconstruct $\mathbf{u}$ from given data $\mathbf{u}^{(0)}$. In turn, we restrict the flow equation (14) to $\mathcal{U}_{\mathbf{Z},G}$:

$$\partial_t \mathbf{u} = P_{\mathcal{U}_{0,G}} \left((\varepsilon I + \nabla \mathbf{u}^T \nabla \mathbf{u}) \nabla \cdot \left(\frac{\nabla \mathbf{u}}{|\nabla \mathbf{u}|} \right) \right), \tag{22}$$

where

$$P_{\mathcal{U}_{0,G}}(v) = v - \|G\|_{L^2(\mathbb{R}^2)}^{-2} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \langle G_{i,j}, v \rangle_{L^2(\Omega)}\, G_{i,j}$$

is applied on each component separately. Note that the assumption $\mathbf{u}^{(0)} \in \mathcal{U}_{\mathbf{Z},G}$ together with $\partial_t \mathbf{u} \in \mathcal{U}_{0,G}$ asserts that the solution $\mathbf{u}(t)$ stays in $\mathcal{U}_{\mathbf{Z},G}$ for all $t \geq 0$. At this point we remark that there is no analytical theory guaranteeing the well posedness of (22).

Since the PDE (22) comprises a projection, for a the numerical solution a time-explicit scheme with sufficiently small step size Δt is required.

7 Numerical Results for Image Interpolation

We compare our method consisting in numerically solving (22) to two standard interpolation methods, namely

nearest neighbor and cubic interpolation, as well as to established, sophisticated interpolation methods proposed by Tschumperlé & Deriche [22] and by Roussos & Maragos [19]. The method of Tschumperlé & Deriche is implemented in the GREYCstoration software (see http://cimg.sourceforge.net/greycstoration/), test results for the method of Roussos & Maragos are available from the site http://cvsp.cs.ntua.gr/∼tassos/PDEinterp/ssvm07res/.

In our method, the kernel function has to be chosen appropriately. We use

$$G(x,y) := \frac{1}{\int_{[-1/2,1/2]^2} g_\sigma(x,y)\,dx\,dy}\chi_{[-1/2,1/2]^2}\; g_\sigma(x,y),$$

where g_σ is the two-dimensional isotropic Gaussian of standard deviation σ. A value of 20 is used for the variance σ^2. The initial data $\mathbf{u}^{(0)}$ are obtained from a sinc-interpolation satisfying the interpolation constraints introduced in Section 6.

For evaluating the different methods, we use the three test images shown in Fig. 3. For each image, a low and a high resolution version is available, where the low resolution image is obtained from the high resolution image via low-pass filtering (convolution with a bi-cubic spline) and down-sampling by a factor of four. The test images were obtained from http://cvsp.cs.ntua.gr/∼tassos/PDEinterp/ssvm07res/.

The methods mentioned above are used to up-sample the low resolution image by a factor of four.

Our method is applied with 100 time steps, $\Delta t = 0.03$, $\varepsilon = 0.05$ and $\sigma^2 = 20$ for the first and 100 time steps, $\Delta t = 0.05$, $\varepsilon = 0.01$ and $\sigma^2 = 20$ for the second test image, respectively. Except as noted otherwise, we use the PDE (14), which corresponds to the choice $p = 1$ in (18).

For GREYCstoration (version 2.9) we use the option '-resize' together with the aimed size of the high resolution image and parameters '-anchor true', '-iter 3' and '-dt 10'. For the remaining parameters the default values are used. The results of Roussos' method were obtained from the web site mentioned above.

Let us consider the results of up-sampling the first test image. In order to highlight the differences between the methods, we compare only details of the resulting images, see Fig. 4.

The results with nearest neighbor and cubic interpolation are shown in Fig. 4, top right and middle left, respectively. Both results are unsatisfactory and confirm, what is well known from the literature, that by nearest neighbor interpolation the up-sampled images look blocky and cubic interpolation produces blurry images. The result of GREYCstoration with interpolation constraints (Fig. 4, middle row right) also appears blurry, but compared to cubic interpolation better reconstruct the edges in the image. The method proposed by Roussos & Maragos as well as our method (see Fig. 4, bottom row) produce sharp and well reconstructed edges.

In section 4 we also investigated the energy functional (18) for the case $p = 1/2$ and the corresponding PDE (19). With our method adapted to this case, the results are slightly smoother than in the case of $p = 1$, but the difference is merely visible. For this reason the results for $p = 1/2$ are not depicted here.

In order to further investigate the differences between the PDE based methods, we zoom into two regions of the second test image, one region containing an edge (see Fig. 5) and one region with texture (see Fig. 6).

Fig. 5 shows the edge region after applying the methods proposed by Tschumperlé with interpolation constraints (top row, second left), Roussos (top row, second right) and our method (top row, right). For comparison we have plotted also the detail of the original image (top row, left). One can see that by Tschumperlé's method the edges appear blurry and irregular. This seems to be an effect of the interpolation constraints, because when Tschumperlé's method is applied without constraints, strong anisotropic diffusion orthogonal to the image gradient enhances the edges. By the method of Roussos the edge is reconstructed in a sharp way, but overshoots appear. Our method is also able to reconstruct the edge sharply but with little overshoots. Concerning the gray mark at the parrot's beak, we observe that Tschumperlé's method reconstructs the shape of the mark better than the other methods do.

The differences in the behavior of the methods can also be recognized when applying the Sobel-operator to the interpolated images: The thickness of the edges in the result of the Sobel-operator indicates the blurriness of the reconstructed edge. We see that the proposed method produces sharper edges than the method by Roussos and more regular edges than the method by Tschumperlé. The overshoots introduced by Roussos' method can also be observed in the outcome of the Sobel-operator. They are far stronger than the overshoots produced by our method.

Now we investigate the effect of the interpolation methods on textures. Fig. 6, top left, shows a textured region of the original image. The results of the methods proposed by Tschumperlé (with interpolation constraints) and Roussos are given in Fig. 6, top right and bottom left, respectively. The result of the proposed method is shown in Fig. 6, bottom right. One observes a certain blurriness in the results by Tschumperlé's method. As for the result before, we point out that incorporating the interpolation constraints seems to have

Fig. 3 Three test images. Each test image is available in a low and a high resolution version with a factor of four between both resolution.

Fig. 4 Up-sampling by a factor of four, Detail of the first test image. *top left:* original high resolution image, *top right:* nearest neighbor interpolation, *middle left:* cubic interpolation, *middle right:* interpolation using GREYCstoration, *bottom left:* interpolation method proposed by Roussos et. al, *bottom right:* proposed interpolation method

a strong effect on the result. When applying GREYCstoration without imposing constraints, the results are much more influenced by the anisotropic diffusion and the edges and the texture are accentuated. In the result of the interpolation method proposed by Roussos, we see a strong effect of the anisotropic diffusion on the texture, so that the result is more visually appealing than the other results. Nevertheless, a comparison with the original image shows that original and reconstructed texture differ significantly. In particular the orientations of the short stripes in the face of the parrot are different. Note that the anisotropic diffusion induced by the direction of the texture also affects the pupil of the parrot. On the result of our method we remark that the reconstruction of the texture is quite conservative, i.e., we stay near the initial guess. The

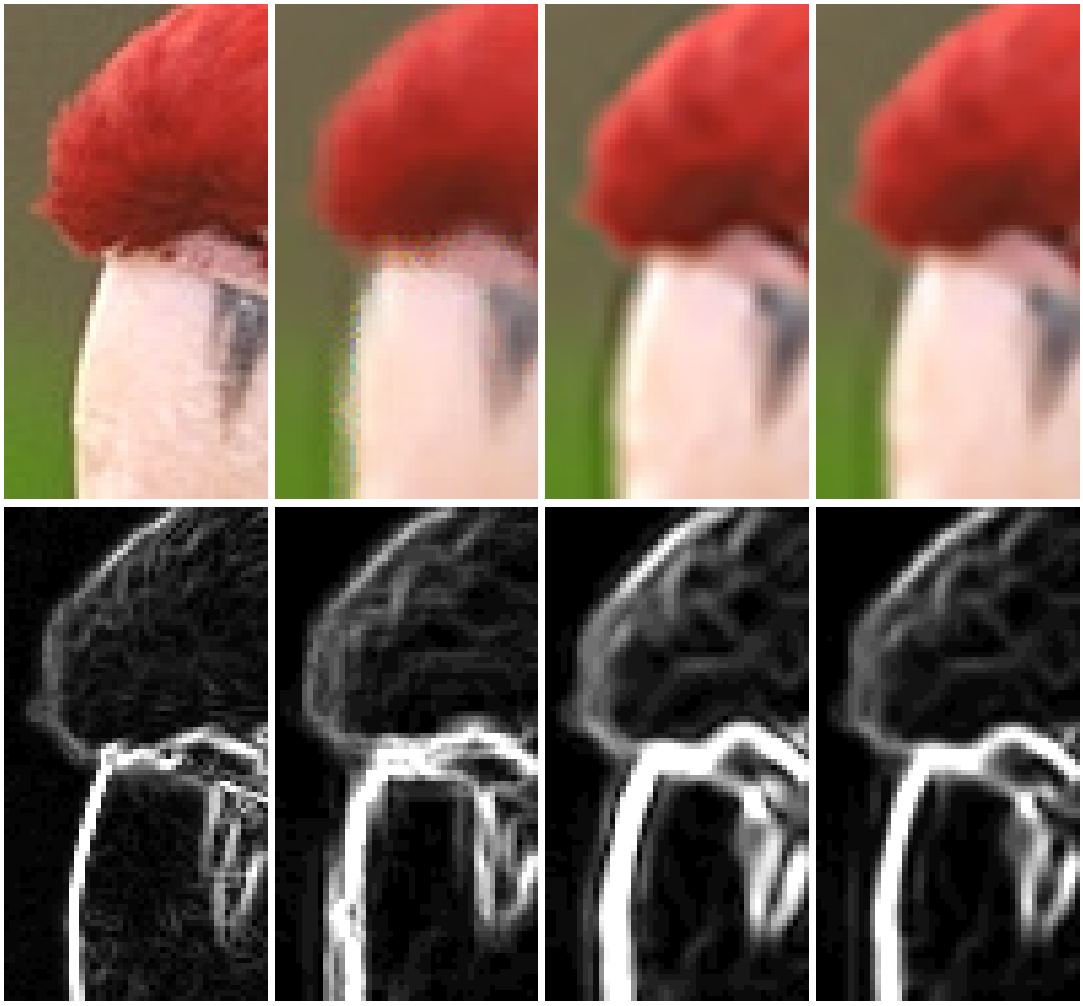


Fig. 5 Detail of an edge in the original and interpolated images (*top row*, using GREYCstoration with interpolation constraints, Roussos' method, and the proposed method) and subsequently applied Sobel-operator (*bottom row*).

	Motocross	Parrots	House
Nearest n.	127.24	40.67	69.53
Cubic	114.66	33.46	64.38
Lanczos	119.10	34.99	66.15
proposed $p = 1/2$	108.89	31.58	62.68
proposed $p = 1$	108.04	31.54	62.72
Tschumperlé	120.45	37,35	67.89
Roussos	109.04	31.60	62.74

Table 1 Difference between the interpolated image and the high-resolution counterpart, measured in the TV-norm for the different test images and interpolation methods.

blockyness is slightly reduced by the evolution process. Taking a look at the eye of the parrot, the relation of our method to Mean Curvature Flow can be observed: The pupil is reconstructed as a perfectly circular shape.

We also provide a quantitative error measure to compare the proposed method with standard and state-of-the art interpolation methods.

To this end we choose the three different test images shown in Figure 3 and seven different interpolation methods: nearest neighbor interpolation, cubic interpolation, Lanczos interpolation, the methods proposed by Tschumperlé and by Roussos and the proposed method. For the latter we also applied the modified version with $p = 1/2$.

We measure the difference between an interpolated image $\mathbf{u}$ and the corresponding high-resolution image $\mathbf{u}^*$ with respect to the (discrete) TV-norm,

$$|\mathbf{v}|_{TV} := \sum_{i,j} |v_{i,j}| + \sum_{i,j} |v_{i+1,j} - v_{i,j}| + |v_{i,j+1} - v_{i,j}|\,,$$

see Table 1. We have chosen the TV-norm for the following reasons:

– Using an L^p-norm of $\mathbf{u} - \mathbf{u}^*$ is inappropriate, since this error measure is biased by the interpolation constraint.

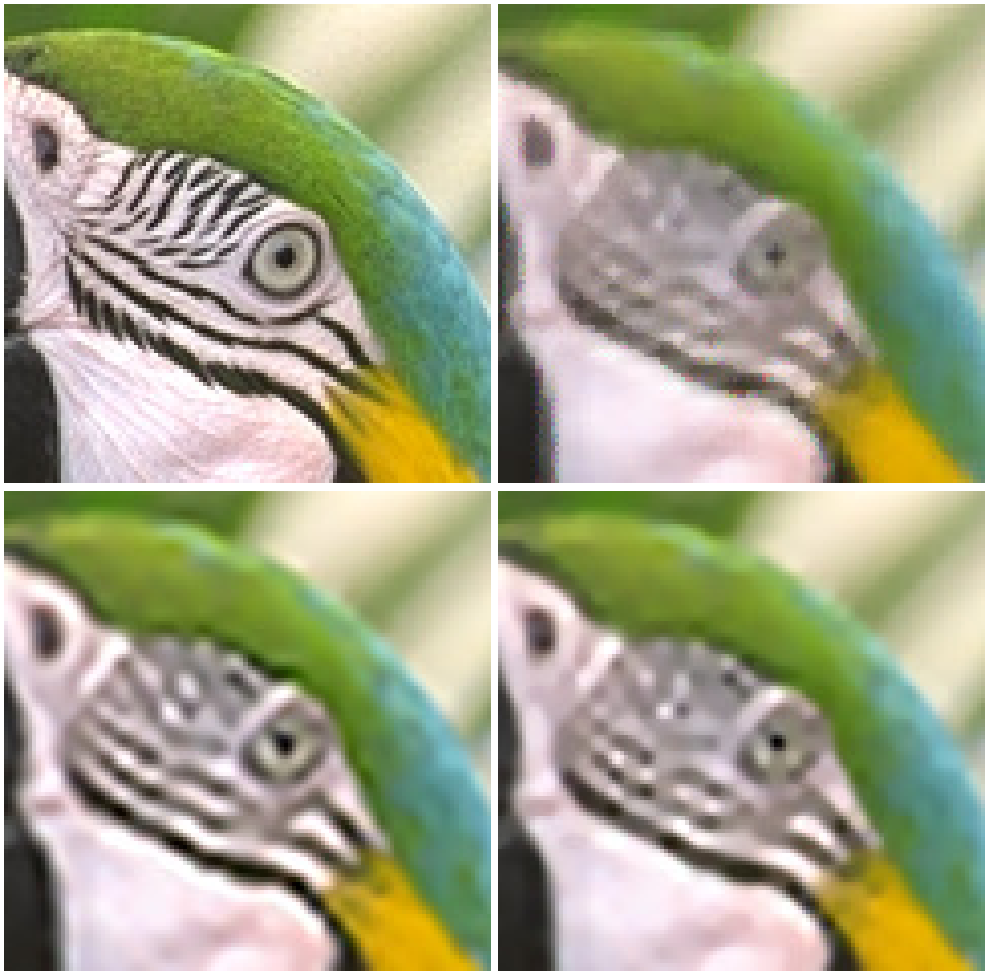


Fig. 6 A texture detail of the original (*top left*) and interpolated images using GREYCstoration (*top right*), Roussos' method (*bottom left*) and the proposed method (*bottom right*).

– The above discussion shows that the methods act differently mainly in regions, where the absolute value of $\nabla \mathbf{u}^*$ is high, i.e., edges and textures. A suitable error measure should therefore incorporate the gradient.
– The TV-norm is known to be a suitable norm for image processing purposes.

Table 1 shows that the proposed method achieves optimal interpolation errors with respect to the TV-norm. Only Roussos method is capable of providing comparable error rates.

8 Conclusions

We have proposed novel partial differential equations which have been designed to correct and filter for displacement errors in multi-channel data. These equations are derived via a semi-group for a non-convex energy functional. In the particular case of gray valued data, we find a partial differential equation which develops optical flow displacements according to mean curvature in normal direction of the level lines. Therefore, the paper links morphological partial differential equations (such as the mean curvature motion), non-convex variational principles, and the optical flow.

When combined with the interpolation constraints, the novel methods are very well suited for image interpolation. A comparison of the results from the proposed method and from interpolation methods from the scale space literature, in particular the GREYCstoration software of Tschumperlé and the interpolation method proposed by Roussos & Maragos, shows the competitiveness of our method.

Acknowledgements We want to thank Gerhard Dziuk (Univ. Freiburg), Peter Elbau (RICAM, Linz) and Markus Grasmair (University Innsbruck) for inspirational discussions. We thank David Tschumperlé for kindly providing GREYCstoration and Anastasios Roussos and Petros Maragos for kindly providing the test images as well as the results of their algorithm.

This work has been supported by the Austrian Science Fund (FWF) within the national research networks Industrial Geometry, project 9203-N12.

References

1. Belahmidi, A., Guichard, F.: A partial differential equation approach to image zoom. In: Proc. of the 2004 Int. Conf. on Image Processing, pp. 649–652 (2004)
2. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: [13], pp. 417–424 (2000)
3. Brézis, H.: Opérateurs maximaux monotones et semi-groupes de contractions dans les espaces de Hilbert. North-Holland Publishing Co., Amsterdam (1973). North-Holland Mathematics Studies, No. 5. Notas de Matemática (50)
4. Burger, W., Burge, M.: Digitale Bildverarbeitung. Springer (2005)
5. Chan, R., Setzer, S., Steidl, G.: Inpainting by flexible Haar wavelet shrinkage. Preprint, University of Mannheim (2008)
6. Chan, T., Kang, S., Shen, J.: Euler's elastica and curvature based inpaintings. SIAM J. Appl. Math. **63**(2), 564–592 (2002)
7. Dacorogna, B.: Weak Continuity and Weak Lower Semicontinuity of Non-Linear Functionals, *Lecture Notes in Mathematics*, vol. 922. Springer Verlag, Berlin, Heidelberg, New York (1982)
8. Dacorogna, B.: Direct Methods in the Calculus of Variations, *Applied Mathematical Sciences*, vol. 78. Springer Verlag, Berlin (1989)
9. Elbau, P., Grasmair, M., Lenzen, F., Scherzer, O.: Evolution by non-convex energy functionals. Reports of FSP S092 - "Industrial Geometry" 75, University of Innsbruck, Austria (2008). Submitted
10. Grasmair, M., Lenzen, F., Obereder, A., Scherzer, O., Fuchs, M.: A non-convex PDE scale space. In: [15], pp. 303–315 (2005). URL http://www.springerlink.com/content/g7q0j5df51majln8/
11. Guichard, F., Malgouyres, F.: Total variation based interpolation. In: Proceedings of the European Signal Processing Conference, vol. 3, pp. 1741–1744 (1998)
12. Hagen, H., Weickert, J. (eds.): Visualization and Processing of Tensor Fields. Mathematics and Visualization. Springer Verlag (2006)
13. Hoffmeyer, S. (ed.): Proceedings of the Computer Graphics Conference 2000 (SIGGRAPH-00). ACMPress, New York (2000)
14. Jähne, B.: Digitale Bildverarbeitung, 5th edn. Springer (2002)
15. Kimmel, R., Sochen, N.A., Weickert, J. (eds.): Scale Space and PDE Methods in Computer Vision, *Lecture Notes in Computer Science*, vol. 3459. Springer (2005)
16. Malgouyres, F., Guichard, F.: Edge direction preserving image zooming: a mathematical and numerical analysis. SIAM J. Numer. Anal. **39**, 1–37 (2001)
17. Nashed, M. (ed.): Generalized inverses and applications. Academic Press [Harcourt Brace Jovanovich Publishers], New York (1976)
18. Roussos, A., Maragos, P.: Vector-valued image interpolation by an anisotropic diffusion-projection pde. In: Scale Space and Variational Methods in Computer Vision, vol. 4485, pp. 104–115. Springer (2007)
19. Roussos, A., Maragos, P.: Reversible interpolation of vectorial images by an anisotropic diffusion-projection pde. In: Special Issue for the SSVM '07 conference. Springer (accepted for publication)
20. Scherzer, O., Weickert, J.: Relations between regularization and diffusion filtering. J. Math. Imaging Vision **12**(1), 43–63 (2000)
21. Tschumperlé, D.: Fast anisotropic smoothing of multi-valued images using curvature-preserving pde's. International Journal of Computer Vision (IJCV) **68**, 65–82 (2006)
22. Tschumperlé, D., Deriche, R.: Vector valued image regularization with pdes: A common framework for different applications. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005)
23. Weickert, J., Welk, M.: Tensor field interpolation with pdes. In: [12], pp. 315–325 (2006)