A MULTIGRID METHOD FOR THE COMPUTATION OF THE OPTICAL FLOW USING A CURVATURE BASED REGULARIZER

K. RUHNAU*, R. WIENANDS*, AND H. KÖSTLER†

Abstract. Optical flow techniques are used to compute an approximate motion field in an image sequence. We apply a variational model for the optical flow introducing a curvature based regularizer that requires the solution of a fourth order system of partial differential equations with jumping coefficients. A geometric multigrid solver for that problem is presented which is composed of collective Gauss-Seidel relaxation and standard geometric transfer operators. Galerkin based coarse grid operators are applied for an efficient treatment of jumping coefficients. Finally, some results on convergence rates, timings and visual quality of the approximated motion field for synthetic and real world images are shown.

1. Introduction. Optical flow is commonly defined to be the motion of brightness patterns in a sequence of images. It was introduced by Horn and Schunck [21], where they proposed a differential method to compute the optical flow from pairs of images using a brightness constancy assumption and an additional smoothness constraint on the magnitude of the gradient of the velocity field in order to regularize the problem. Since then optical flow has been studied intensively and many extensions to that simple variational approach e.g. considering different regularizing terms were investigated [20, 27, 14, 32, 18, 33, 34, 5, 6] and applied to various applications ranging from robotics to video compression. Another applications is Particle Image Velocimetry (PIV), where optical flow provides approximate motion of fluid flows. Here, it is necessary to incorporate physically more meaningful regularizers to be able to impose e.g. a incompressibility condition of the velocity field. Suter [29] introduced therefore a smoothness constraint on the divergence and curl of the velocity field, that was used intensively in the following [16, 9, 24, 8].

The contribution of this work is twofold. First, we investigate the properties of a curvature based regularizer, well-known in image registration that is related to optical flow [12, 26], and that is a special case of a second order div-curl based regularizer [16]. The intention of this approach is to let affine motion unpenalized while higher order motions are still used to enforce smoothness. We present the variational optical flow approach for this regularizer in Section 2, combine it with the simple Horn and Schunck and extend it to an isotropic version to deal with discontinuities in the velocity field. Another reason why we want to use a higher order regularizer is, that for some applications additional information from features or landmarks is given for the optical flow computation [37]. Here, the higher order regularizer is required to avoid singularities in the solution [11, 13].

Besides accuracy of the approximate motion field obtained by optical flow, an important goal is to achieve real time or close to real time performance in many applications, which makes an efficient numerical solution of the underlying system of partial differential equations (PDE) mandatory. First attempts to use multilevel techniques to speed up optical flow computations are due to Glazer [15] and Terzopoulos [30]. After that, several multigrid based solvers were proposed for different optical flow regularizers (see, e.g., [10, 2, 23, 22, 7, 6]). Our second contribution is a geometric multigrid method presented in Section 3 that is used to efficiently solve the fourth order system of partial differential equations derived from our variational optical flow approach. Especially the smoothing method is investigated in some detail.

In Section 4 optical flow results using the curvature regularizer both for synthetic and real world images are found and compared to the classical regularizer of Horn and Schunck. We end this paper with an outlook for future developments based on curvature regularization.

*University of Cologne, Mathematical Institute, Cologne, Germany
†University of Erlangen-Nuremberg, Department of Computer Science 10, Erlangen, Germany

2. Optical Flow model. The variational approach to compute the motion field as proposed by Horn and Schunck [21] uses two assumptions. The first is, that a moving object in the image does not change its gray values, that is e.g. changes of illumination are neglected. For an image sequence $I : \Omega \times \mathbb{T} \to \mathbb{R}\,, \Omega \subset \mathbb{R}^2$ describing the gray value intensities for each point $\boldsymbol{x} = (x, y)$ in the regular image domain Ω at time $t \in \mathbb{T} = [0, t_{max}], t_{max} \in \mathbb{N}$ this so-called brightness constancy assumption reads

$$\frac{dI}{dt} = 0 \ . \tag{2.1}$$

Therefore the movement of a gray value at a point (x, y, t) is

$$I(\boldsymbol{x}, t) = I(x + dx, y + dy, t + dt) \ . \tag{2.2}$$

Taylor expansion of $I(x+dx, y+dy, t+dt)$ around (x, y, t) neglecting higher order terms and using (2.2) gives

$$I_x u + I_y v + I_t \approx 0$$

with the partial image derivatives $\frac{\partial I}{\partial x} = I_x$, $\frac{\partial I}{\partial y} = I_y$, $\frac{\partial I}{\partial t} = I_t$ and the optical flow velocity vector $\boldsymbol{u} = (u, v)^T\,, u := \frac{dx}{dt}\,, v := \frac{dy}{dt}$. This brightness constancy assumption is used throughout this paper, but by itself results in an ill-posed, under-determined problem. Therefore additional regularization is required. Horn and Schunck proposed as second assumption a smoothness constraint

$$S_1(\boldsymbol{u}) = \|\nabla u\|^2 + \|\nabla v\|^2$$

and combined both in an energy functional

$$E_1(\boldsymbol{u}) := \int_\Omega (I_x u + I_y v + I_t)^2 + \alpha S_1(\boldsymbol{u}) dx \tag{2.3}$$

that is to be minimized. $\alpha \in \mathbb{R}^+$ represents a weighting parameter. The curvature based regularizer penalizes second derivatives instead and can be written as

$$S_2(\boldsymbol{u}) = (\Delta u)^2 + (\Delta v)^2 \ .$$

As already mentioned, it is a special case of the div-curl based regularizer [16]

$$S_{2'}(\boldsymbol{u}) = \alpha_1 \|\nabla \mathrm{div} \boldsymbol{u}\|^2 + \alpha_2 \|\nabla \mathrm{curl} \boldsymbol{u}\|^2 \ ,$$

where $\alpha_1 = \alpha_2 = 1$. Note that affine motions are in the kernel of the curvature based regularizer. The corresponding energy functional $E_2(\boldsymbol{u})$ is obtained by simply replacing S_1 by S_2 in (2.3). The resulting Euler-Lagrange equations constituting a necessary condition for a minimum of $E_k(\boldsymbol{u}), k \in \{1, 2\}$ are

$$-\alpha \Delta^k u + I_x(I_x u + I_y v + I_t) = 0 \tag{2.4a}$$
$$-\alpha \Delta^k v + I_y(I_x u + I_y v + I_t) = 0 \tag{2.4b}$$

with natural Neumann boundary conditions. This system of second resp. fourth order PDEs is discretized by finite differences and solved by a geometric multigrid method, described in the next section.

3. Multigrid solver. In this Section we describe the multigrid treatment of the resulting linear system corresponding to the discrete version of (2.4). We assume that the reader is familiar with the main multigrid principles (see, e.g., [3, 17, 4, 31, 35]) and restrict the considerations to a short discussion of the involved multigrid components that characterize the complete multigrid algorithm.

3.1. Discrete system. The biharmonic operator which appears in (2.4) for $k = 2$ is known to lead to poor multigrid performance. Therefore it is a common approach to split up the biharmonic operator into a system of two Poisson-type equations [31]. Employing this idea, (2.4) with $k = 2$ can be transformed into the following system using additional unknown functions w^1 and w^2:

$$0 = -\Delta u(\boldsymbol{x}) + w^1(\boldsymbol{x}) \tag{3.1a}$$
$$0 = -\Delta v(\boldsymbol{x}) + w^2(\boldsymbol{x}) \tag{3.1b}$$
$$0 = -\alpha\Delta w^1(\boldsymbol{x}) + I_x\left(\boldsymbol{x},t\right)(I_x(\boldsymbol{x},t)u(\boldsymbol{x}) + I_y(\boldsymbol{x},t)v(\boldsymbol{x}) + I_t(\boldsymbol{x},t)) \tag{3.1c}$$
$$0 = -\alpha\Delta w^2(\boldsymbol{x}) + I_y(\boldsymbol{x},t)\left(I_x(\boldsymbol{x},t)u(\boldsymbol{x}) + I_y(\boldsymbol{x},t)v(\boldsymbol{x}) + I_t(\boldsymbol{x},t)\right) \tag{3.1d}$$

with $\boldsymbol{x} \in \Omega$. Note that for $w^1 \equiv u$ and $w^2 \equiv v$ equations (3.1c) and (3.1d) are identical to (2.4) with $k = 1$, that is equivalent to the optical flow system based on the original second order regularizer of Horn and Schunck [21]. To allow for a linear combination of the classical second order and our fourth order system, we replace the Laplacian in equations (3.1a) and (3.1b) by $-((1-\beta)\Delta + \beta)$ with $\beta \in [0,1]$. For $\beta = 0$ we obtain the fourth order system from (2.4) with $k = 2$ whereas for $\beta = 1$ a system similar to the original Horn and Schunck second order system results because (3.1a) and (3.1b) exchange into $-u + w^1 \equiv 0$ and $-v + w^2 \equiv 0$, respectively. The difference is, that we now have two different borders for the data term and for the regularizer, whereas the original system had one border for both data term and regularizer. Finally the discrete system under consideration for a fixed time t reads

$$\boldsymbol{L}_h(\boldsymbol{x}) \begin{pmatrix} u_h(\boldsymbol{x}) \\ v_h(\boldsymbol{x}) \\ w_h^1(\boldsymbol{x}) \\ w_h^2(\boldsymbol{x}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -I_x(\boldsymbol{x},t)I_t(\boldsymbol{x},t) \\ -I_y(\boldsymbol{x},t)I_t(\boldsymbol{x},t) \end{pmatrix} \qquad \text{with} \qquad \boldsymbol{x} \in \Omega_h \ , \tag{3.2}$$

$$\boldsymbol{L}_h(\boldsymbol{x}) = \begin{pmatrix} -(1-\beta)\Delta_h - \beta & 0 & 1 & 0 \\ 0 & -(1-\beta)\Delta_h - \beta & 0 & 1 \\ I_x^2(\boldsymbol{x},t) & I_x(\boldsymbol{x},t)I_y(\boldsymbol{x},t) & -\alpha\Delta_h & 0 \\ I_x(\boldsymbol{x},t)I_y(\boldsymbol{x},t) & I_y^2(\boldsymbol{x},t) & 0 & -\alpha\Delta_h \end{pmatrix} ,$$

and discrete functions u_h, v_h, w_h^1, and w_h^2. Here, Ω_h denotes the discrete image domain (i.e. each $\boldsymbol{x} \in \Omega_h$ refers to a pixel) and Δ_h the standard five-point central discretization of the Laplacian (compare with, e.g., [31]). The determinant of the discrete system is given by

$$\begin{aligned} \det\left(\boldsymbol{L}_h\right) = &\left(\alpha^2\beta^2 - 2\alpha^2\beta + \alpha^2\right)\Delta_h^4 + \left(2\alpha^2\beta - 2\alpha^2\beta^2\right)\Delta_h^3 \\ &+ \left(\alpha^2\beta^2 + (\alpha - \alpha\beta)\left(I_x^2 + I_y^2\right)\right)\Delta_h^2 + \alpha\beta\left(I_x^2 + I_y^2\right)\Delta_h \end{aligned}$$

omitting the dependency on space and time for convenience. For the special cases $\beta = 0$ and $\beta = 1$ we obtain

$$\det\left(\boldsymbol{L}_h\right) = \alpha^2\Delta_h^4 + \alpha\left(I_x^2 + I_y^2\right)\Delta_h^2 \qquad \text{and} \qquad \det\left(\boldsymbol{L}_h\right) = \alpha^2\Delta_h^2 + \alpha\left(I_x^2 + I_y^2\right)\Delta_h \ ,$$

respectively. The principle part of $\det\left(\boldsymbol{L}_h\right)$ is Δ_h^m with $m = 4$ for $\beta \in [0,1)$ and $m = 2$ for $\beta = 1$ due to $\alpha \neq 0$. Hence four boundary conditions for $\beta \neq 1$ are required and two boundary conditions for $\beta = 1$ (see, e.g., [3, 31]). In principle one can use pure Dirichlet or pure Neumann boundary

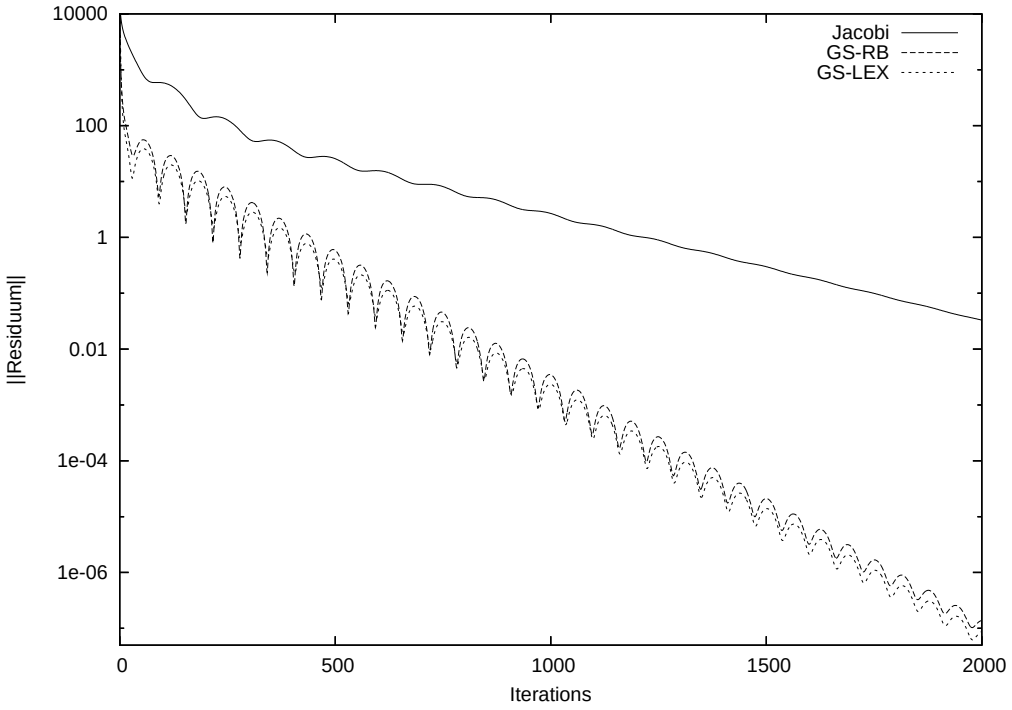


FIGURE 3.1. *Residuum improvement of relaxations.*

conditions for each of the four unknown functions. For $\beta = 1$, only two out of the four boundary conditions are linearly independent. For the experiments presented in Section 4 we used pure homogeneous Neumann boundary conditions.

3.2. Smoothing method. The straight-forward generalization of a scalar smoothing method to a system of PDEs like (3.2) is a collective relaxation method. This relaxation method sweeps over all grid points $\boldsymbol{x} \in \Omega_h$ in a certain order, for example, in a lexicographic or a red-black manner. At each grid point the four difference equations are solved simultaneously, i.e., the corresponding variables $u_h(\boldsymbol{x})$, $v_h(\boldsymbol{x})$, $w_h^1(\boldsymbol{x})$ and $w_h^2(\boldsymbol{x})$ are updated simultaneously. This means that a (4×4)-system has to be solved at each grid point.

First of all we have to note that the large sparse matrix which corresponds to the discrete system (3.2) is neither symmetric nor diagonally dominant. Furthermore, it is not an M-Matrix due to positive off-diagonal entries. As a consequence, most of the classical convergence criteria for standard iterative methods like Jacobi or Gauss-Seidel relaxation do not apply and it has to be expected that these methods might diverge for certain parameter choices. In our numerical tests for collective lexicographic or red-black Gauss-Seidel relaxation (abbreviated by GS-LEX and GS-RB, respectively) we always observed an overall convergence, although for certain combinations of α, β, I_x, I_y there were single relaxation steps with an increasing residuum. An example of such a convergence history is shown in Fig. 3.1 for collective Jacobi, GS-LEX and GS-RB relaxation.

However, if a relaxation method is applied within a multigrid algorithm then we are mainly interested in its smoothing properties. That is, the relaxation is aimed at a sufficient reduction of the high-frequency components of the error between the exact solution and the current approximation. A quantitative measure of its efficiency represents the smoothing factor μ_{loc} obtained by local Fourier analysis. μ_{loc} is defined as the worst asymptotic error reduction by one relaxation

TABLE 3.1
Smoothing factors for GS-LEX and GS-RB, $\alpha = 1500$.

β	0	0.4	1
GS-LEX	0.49973	0.49980	0.49970
GS-RB	0.25003	0.25009	0.25000

step of all high-frequency error components. For details on local Fourier analysis we refer to the literature [3, 31, 36]. Dealing with operators based on variable coefficients prevents a direct application of local Fourier analysis. However, the analysis can be applied to the locally frozen operator at a fixed grid point $\boldsymbol{\xi}$. Replacing the variable $\boldsymbol{x}$ by a constant $\boldsymbol{\xi}$, one obtains an operator $\boldsymbol{L}_h(\boldsymbol{\xi})$ with constant frozen coefficients. In case of smoothly varying coefficients the smoothing factor for $\boldsymbol{L}_h(\boldsymbol{x})$ can be bounded by the maximum over the smoothing factors for the locally frozen operator, i.e.,

$$\mu_{\text{loc}}\left(\boldsymbol{L}_h(\boldsymbol{x})\right) = \max_{\boldsymbol{\xi}\in\Omega_h} \mu_{\text{loc}}\left(\boldsymbol{L}_h(\boldsymbol{\xi})\right). \tag{3.3}$$

As a popular test case we consider frame 8 of the Yosemite sequence shown in Fig. 4.2. Table 3.1 presents the corresponding smoothing factors calculated via (3.3) for GS-LEX and GS-RB with varying β. α is fixed at 1500 which turned out to be a proper choice w.r.t. the average angular error, compare with Section 4.2 below. Obviously there is hardly any influence of the parameter β on the resulting smoothing factor. We always observe nearly the same smoothing factors as they are well-known for the Poisson equation (i.e. $\mu = 0.5$ for GS-LEX and $\mu = 0.25$ for GS-RB). Systematic tests show that the same statement is also valid for the parameter α. As a consequence we can expect to obtain the typical multigrid efficiency as long as the coarse grid corrections works properly, compare with Section 3.3. The situation is considerably more complicated if we apply decoupled relaxations (compare with [31]) which will be discussed elsewhere.

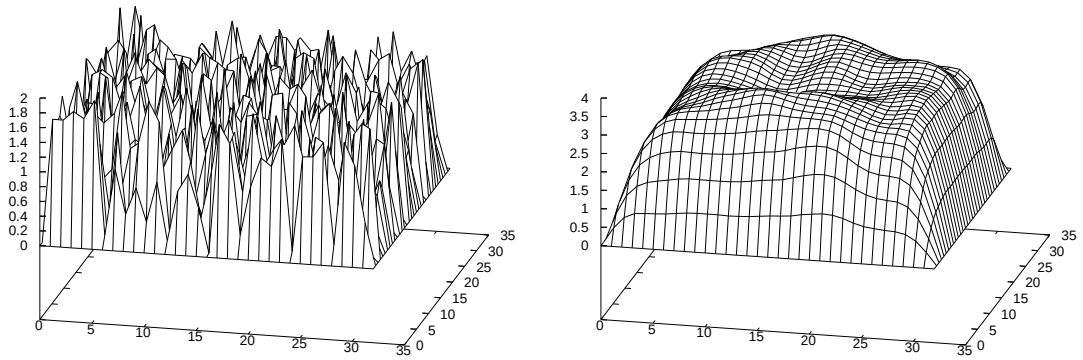
Note that I_x and I_y are not varying smoothly over the image domain Ω_h for this test case. Instead we have moderate jumps in the coefficients. As a consequence, the smoothing factors from Table 3.1 are not justified rigorously. However from practical experience, they can be considered as heuristic but reliable estimates for the actual smoothing properties especially since we only have moderate jumps.

To back up the theoretical results from smoothing analysis we also tested the smoothing effect of the collective relaxations numerically. The smoothing effect of GS-LEX can be clearly seen from Fig. 3.2. Here, the initial (random) error on a 33×33 grid (a scaled down version of frame 8 from the Yosemite sequence) and the error after five collective GS-LEX steps of the first component u of the optical flow velocity vector is shown.

Summarizing, there is sufficient evidence that collective Jacobi, GS-LEX and GS-RB relaxation are reasonable *smoothing* methods even though they might diverge for single relaxation steps as stand alone solvers.

3.3. Remaining multigrid components. Next to the collective GS relaxation standard multigrid components are applied. To handle the jumping coefficients in I_x and I_y we use Galerkin coarse grid operators. Since there are only moderate jumps it is not necessary to consider operator-dependent transfers but we can stay with straight-forward geometric transfers like full-weighting and bilinear interpolation. The numerical performance of the resulting multigrid algorithm is discussed in the next Section.

4. Experimental results. All experiments for different combinations of α and β (see below) were conducted with five $V(2,2)$ cycles using three levels. The algorithmic details are given in the previous section. If not stated otherwise, all experiments used collective GS-LEX as smoother.



Initial Error Error after five GS-LEX steps

FIGURE 3.2. *Error smoothing of GS iteration for a scaled down version of frame 8 from the Yosemite sequence.*

TABLE 4.1
Convergence rates for $\alpha = 1500$ *in the Yosemite sequence.*

	GS-LEX			GS-RB		
Cycle	$\beta = 0$	$\beta = 0.4$	$\beta = 1$	$\beta = 0$	$\beta = 0.4$	$\beta = 1$
1	0.053	0.051	0.048	0.091	0.090	0.074
2	0.054	0.042	0.045	0.070	0.055	0.044
3	0.096	0.065	0.148	0.115	0.069	0.127
4	0.124	0.086	0.196	0.156	0.093	0.181
5	0.131	0.093	0.232	0.172	0.110	0.233

The input images are smoothed with a discrete Gaussian filter mask (standard deviation $\sigma = 1.2$) in order to be able to compute the image derivatives.

4.1. Convergence rates. For constant coefficients I_x and I_y one obtains the typical multigrid convergence factors similar as for the Poisson equation which can be nicely predicted by local Fourier analysis. For jumping coefficients a slight deterioration of the convergence rate can be observed. Table 4.1 lists some results, which show an intresting coupling. The best convergence rates are achieved when the combination of α and β is optimal with respect to the quality of the solution. Fig. 4.1 shows an AAE (see Section 4.2) plot over β for $\alpha = 1500$. There the best quality with respect to AAE is obtained for $\beta \approx 0.4$. On the other hand the best convergence rates for $\alpha = 1500$ are also obtained for $\beta \approx 0.4$.

4.2. Quality of optical flow. In general, it is very hard to quantify the quality of the optical flow velocity field. For synthetic image sequences, often a ground truth motion field (see [25] for details) is used to measure the quality of a computed optical flow field by the Average Angular Error (AAE). It is calculated via (cf. [6])

$$AAE(\boldsymbol{u}_c, \boldsymbol{u}_e) = \frac{1}{|\Omega|} \int_\Omega \arccos\left(\frac{\boldsymbol{u}_c^T \boldsymbol{u}_e}{|\boldsymbol{u}_c||\boldsymbol{u}_e|}\right) dx \ , \tag{4.1}$$

where $\boldsymbol{u}_c = (u_c, v_c, 1)$ is the ground truth and $\boldsymbol{u}_e = (u_e, v_e, 1)$ the estimated optical flow vector.

Most real world image sequences do not not offer a ground truth motion field, therefore in this case the quality of the optical flow is often measured visually by plotting the vector field and rate it compared to the expected result. For example how smooth is the vector field inside objects or how does the vector field look at edges from different movements (objects moving over a static

background). The first requirement is usually a strength of the original second order regularizer. It only allows small changes of near vectors and smoothes edges out.

In the following we use two sequences, one synthetic and one real world [1], to evaluate our optical flow model.

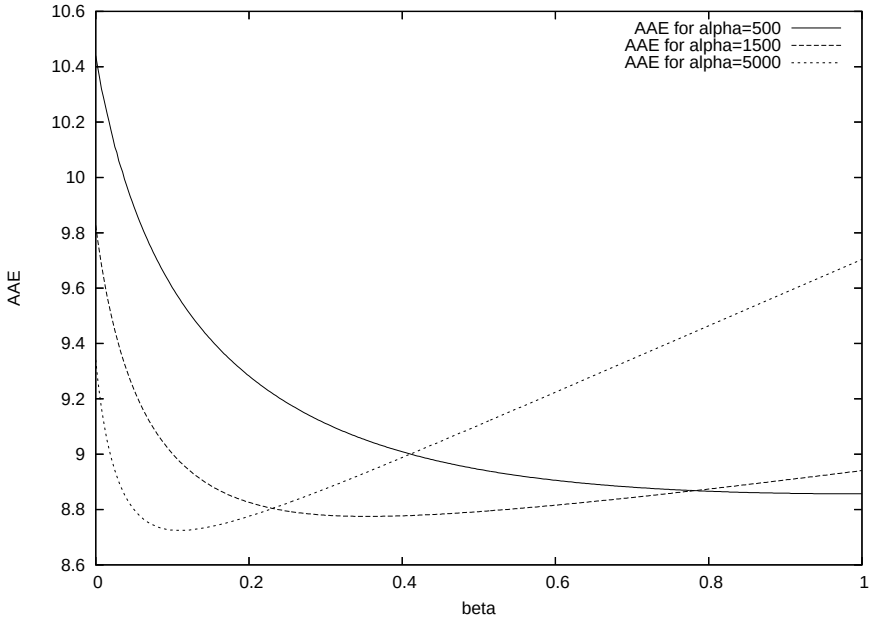


FIGURE 4.1. *AAE plot of the calculated optical flow between pictures 8 and 9 from the Yosemite sequence for* $\alpha = 500$, $\alpha = 1500$ *and* $\alpha = 5000$.

4.2.1. Yosemite sequence. The Yosemite sequence with clouds, created by Lynn Quam [19], is a more complex test case (see Fig. 4.2). It consists of 15 frames of size 316×252 and depicts a flight through the Yosemite national park. In this sequence translational (clouds) and divergent motion (flight) is present. Additionally we have varying illumination in the region of the clouds, thus our constant brightness assumption is not fulfilled there.

All tests were obtained with frames 8 and 9 of the Yosemite sequence. First we consider in Fig. 4.1 the AAE for $\alpha = 500, 1500, 5000$ and varying β. $\alpha = 500$ was chosen because it is the optimal value (w.r.t. a minimal AAE) for the second order system. Using $\beta \neq 1$ does not improve the AAE for the Yosemite sequence for $\alpha = 500$. Considering $\alpha = 1500$ and $\alpha = 5000$ shows, that the pure fourth order system ($\beta = 0$) does not perform better compared to the optimal AAE obtained for the second order system ($\beta = 1$, $\alpha = 500$) either. However, when β is varied to combine the fourth and second order system, the resulting AAE for $\alpha = 1500, 5000$ outperforms both pure systems.

Figure 4.2 shows image details of the resulting velocity fields for the Yosemite sequence. The right half of this detail includes the high mountain from the middle of the pictures. The mountains are moving from right to left, while the clouds region is moving (pure horizontally) from left to right. For $\beta = 1$ one can see the usual behavior of the original Horn and Schunck regularizer, which tries to produce a smooth solution even over the mountain crest. Our fourth order system performs much better in this regard, as the region of influence is notably smaller for example at the right crossover. While producing a better AAE, $\beta = 0.4$ mixes in the smoothing behavior of the

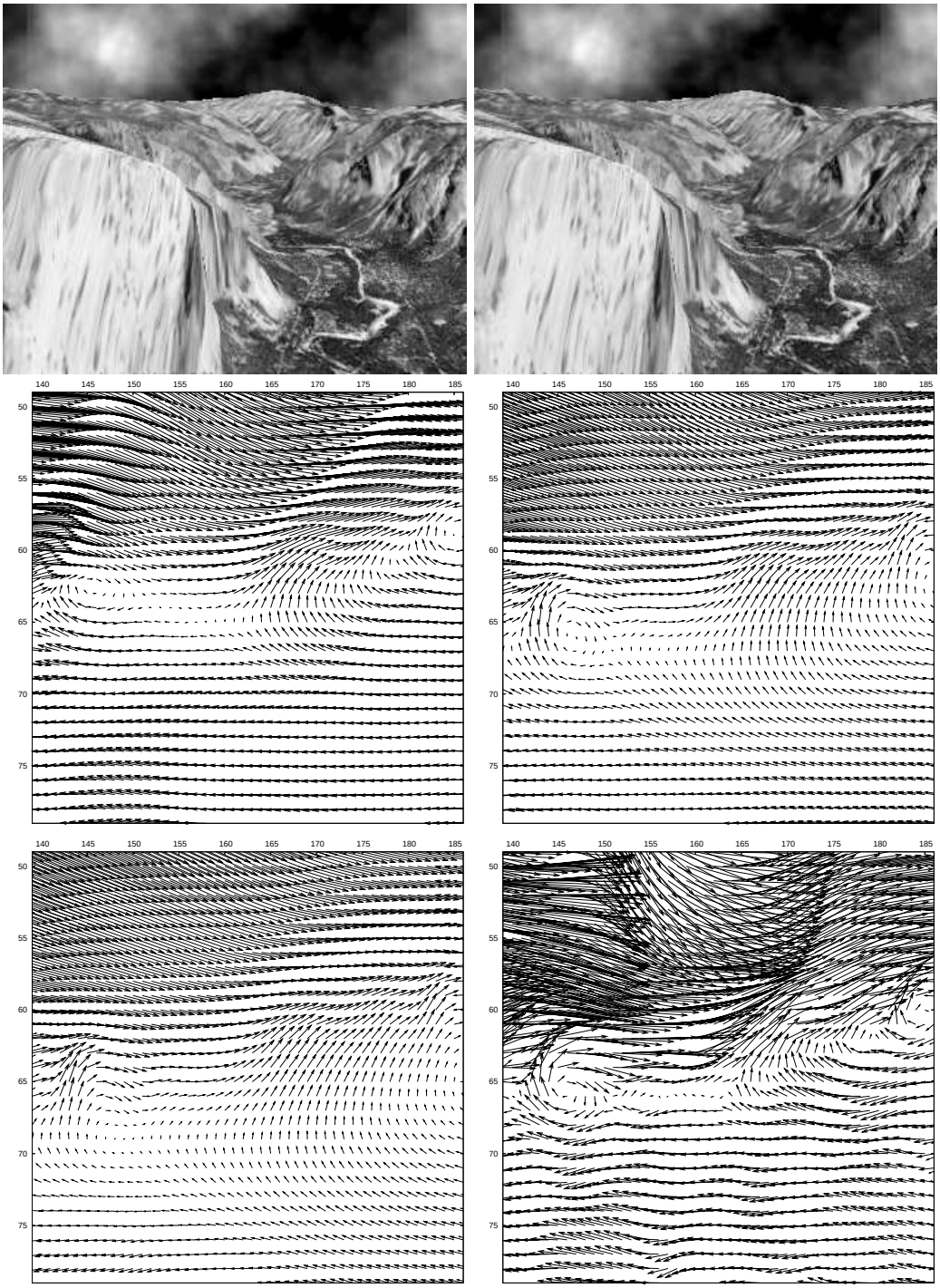


FIGURE 4.2. *First line: Frames 8 and 9 from Yosemite sequence. Second line: A detail from the optical flow located left from the highest mountain in the middle of the image. It was calculated with* $\alpha = 1500$ *and (from left to right)* $\beta = 0$ *and* $\beta = 0.4$. *Third line: The same detail for* $\beta = 1$ *and on the right for* $\beta = 0$ *using* S_3.

second order system and leads to a solution that is too smooth. $\beta = 0.4$ performs better at other regions of the image. One can also observe that all methods fail to calculate the pure horizontal flow in the clouds region. That is due to the fact, that the brightness varies here and thus fails to meet the constant brightness assumption of the data term.

To improve the static weighting of the regularizer, which produces an equally smooth solution throughout the picture, we introduce a diffusivity function $g : \mathbb{R} \to \mathbb{R}^+$ given by, for instance, the regularized total variation diffusivity [28]

$$g(s^2) = \frac{1}{2\sqrt{s^2 + \epsilon^2}} \ , \tag{4.2}$$

where $\epsilon = 10^{-3}$ is used to avoid unbounded diffusivities. This leads to an isotropic curvature based regularizer

$$S_3(\boldsymbol{u}) = g(|\nabla I|^2) \left((\Delta u)^2 + (\Delta v)^2 \right) \ .$$

Fig. 4.2 contains the results from the same image detail of the Yosemite sequence. What one can see here is a very sharp edge from the mountain crest and a smooth solution directly below. This visual result can be further improved if, e.g., ∇I is smoothed before the system is set up.

4.2.2. Hamburg Taxi sequence. As a real image sequence we take the famous Hamburg Taxi sequence of size 256×190 shown in Fig. 4.3. For the multigrid solver the image sequence was scaled up to 256×192 to allow for three multigrid levels. In this street scene, originally provided by the University of Hamburg and available from `http://i21www.ira.uka.de/image_sequences`, there are four moving objects; the taxi turning the corner, a car in the lower left driving from left to right, a van in the lower right entering the picture and driving right to left, and a pedestrian in the upper left.

The two image details from the optical flow are taken between the first and the second car. In the lower left corner the movement of the dark car and in the top right corner the diagonal movement of the bright car is visualized. The left vector field plot shows the optical flow for the original, second order regularizer. As one can see, the top right area includes the diagonal vectors for the bright car, but the movement of the dark car suffers from the second order smoothness. Also the background area (which does not move) between both cars is covered with a smooth solution that tries to couple the movements of both cars. The vector field from our fourth order system is visualized on the right. One can clearly see the two areas from the movement of both cars. Also the background between both cars is preserved by a relatively small flow. One can even recognize the curves from the dark car, as the flow field quickly changes outside the car object. These observations demonstrate that the fourth order system does not penalize affine motion that occurs in this sequence.

5. Conclusion and Outlook. We presented and evaluated a curvature based regularizer for optical flow computations. The arising fourth order system of PDEs were solved efficiently by a geometric multigrid solver. Here, it shows that the best results are obtained, when the weighting between regularizer and brightness constancy assumption is chosen such that the multigrid solver shows an optimal convergence rate. Next steps are the extension of the regularizer to div-curl or nonlinear curvature based regularizers, where α, β depend on the velocity field, and to higher dimensions.

REFERENCES

[1] J. Barron, D. Fleet, and S. Beauchemin, *Performance of optical flow techniques*, International Journal of Computer Vision, 12 (1994), pp. 43–77.

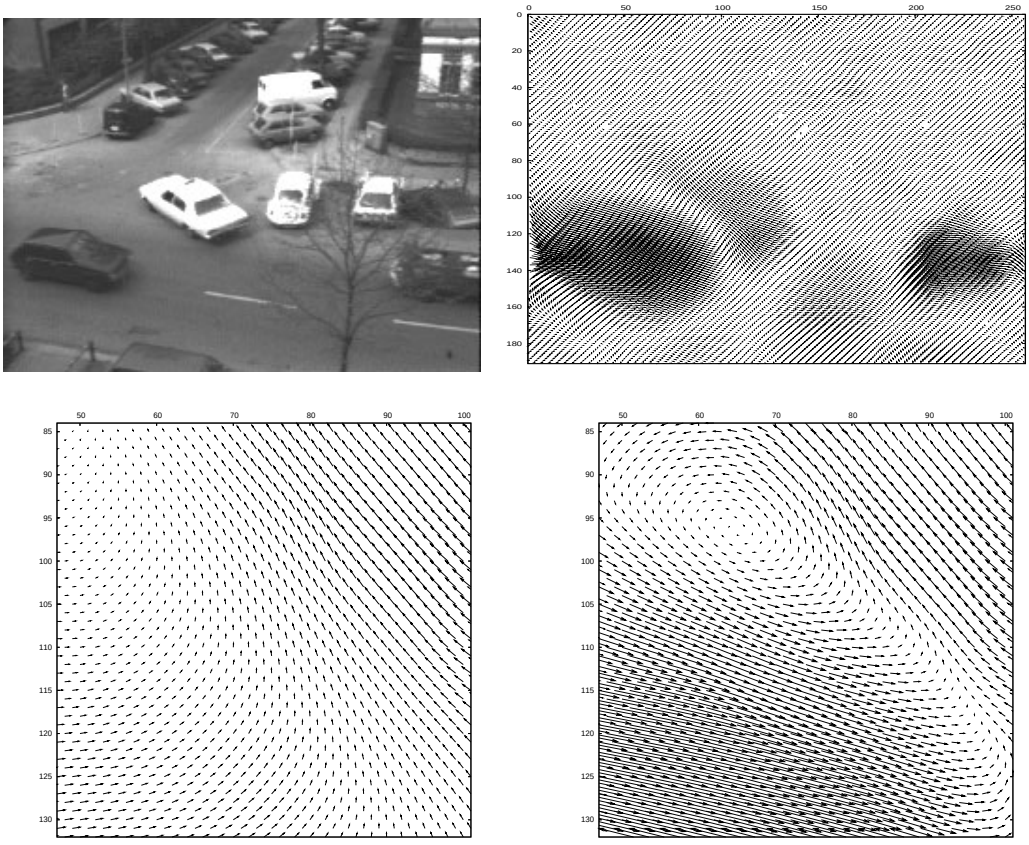


FIGURE 4.3. *First line: Image one from the Hamburg Taxi sequence and the full vector field. Second line: Two vector fields from a detail of the optical flow using alpha* $= 1500$ *and the second order regularizer* ($\beta = 1$) *on the left and our fourth order system* ($\beta = 0$) *on the right.*

[2] R. BATTITI, E. AMALDI, AND C. KOCH, *Computing optical flow across multiple scales: an adaptive coarse-to-fine strategy*, International Journal of Computer Vision, (1991), pp. 133–145.

[3] A. BRANDT, *Multigrid methods: 1984 guide with applications to fluid dynamics*, The Weizmann Institute of Science, Rehovot, Israel, (1984).

[4] W. BRIGGS, V. HENSON, AND S. MCCORMICK, *A Multigrid Tutorial*, Society for Industrial and Applied Mathematics, Philadelphia, 2nd ed., 2000.

[5] T. BROX AND J. WEICKERT, *Nonlinear matrix diffusion for optic flow estimation*, LNCS, (2002), pp. 446–453.

[6] A. BRUHN, *Variational Optic Flow Computation: Accurate Modeling and Efficient Numerics*, PhD thesis, Department of Mathematics and Computer Science, Saarland University, Saarbrücken, 2006.

[7] I. CHRISTADLER, H. KÖSTLER, AND U. RÜDE, *Robust and efficient multigrid techniques for the optical flow problem using different regularizers*, in 18th Symposium Simulationstechnique ASIM 2005 Proceedings, F. Hülsemann, M. Kowarschik, and U. Rüde, eds., vol. 15 of Frontiers in Simulation, ASIM, SCS Publishing House, Sep 2005, pp. 532–537. also exists as technical report 05-6.

[8] T. CORPETTI, D. HEITZ, G. ARROYO, E. MÉMIN, AND A. SANTA-CRUZ, *Fluid experimental flow estimation based on an optical-flow scheme*, Experiments in Fluids, 40 (2006), pp. 80–97.

[9] T. CORPETTI, E. MÉMIN, AND P. PÉREZ, *Dense estimation of fluid flows*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (2002), pp. 365–380.

[10] W. ENKELMANN, *Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences*, Computer Vision, Graphics, and Image Processing, (1988), pp. 150–177.

[11] B. FISCHER AND J. MODERSITZKI, *Combining landmark and intensity driven registrations*, PAMM, 3 (2003), pp. 32–35.

[12] ———, *Curvature based image registration*, Journal of Mathematical Imaging and Vision, 18 (2003), pp. 81 – 85.
[13] J. Galic, I.and Weickert, M. Welk, A. Bruhn, A. G. Belyaev, and H.-P. Seidel, *Towards pde-based image compression.*, in VLSM, 2005, pp. 37–48.
[14] B. Galvin, B. McCane, K. Novins, D. Mason, and S. Mills, *Recovering motion fields: An evaluation of eight optical flow algorithms*, in British Machine Vision Conference, 1998.
[15] F. Glazer, *Multilevel relaxation in low-level computer vision*, Multi-resolution image processing and Analysis (A. Rosenfeld, Ed.) Springer-Verlag, (1984), pp. 312–330.
[16] S. Gupta and J. Prince, *Stochastic models for div-curl optical flow methods*, IEEE Signal Processing Letters, 3 (1996), pp. 32–34.
[17] W. Hackbusch, *Multi-Grid Methods and Applications*, Springer Verlag, 1985.
[18] H. W. Haussecker and D. J. Fleet, *Computing optical flow with physical models of brightness variation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 (2001), pp. 661–673.
[19] D. Heeger, *Model for the extraction of image flow*, Journal of the Optical Society of America A: Optics, Image Science, and Vision, 4 (1987), pp. 1455–1471.
[20] B. Horn, *Robot vision*, MIT Press, Cambridge, 1986.
[21] B. Horn and B. Schunck, *Determining optical flow*, Artificial Intelligence, (1981), pp. 185–203.
[22] E. Kalmoun, H. Köstler and U. Rüde, *Parallel multigrid computation of the 3D optical flow*, Tech. Rep. 04-4, IMMD10, 2004.
[23] E. Kalmoun and U. Rüde, *A variational multigrid for computing the optical flow*, Vision, Modeling and Visualization 2003, (Berlin 2003), pp. 577–584. T. Ertl, B. Girod, G. Greiner, H. Niemann, H.-P. Seidel, E. Steinbach, R. Westermann (Eds.). Akademische Verlagsgesellschaft.
[24] T. Kohlberger, E. Mémin, and C. Schnörr, *Variational dense motion estimation using the helmholtz decomposition*, in 4th Int. Conf. on Scale Space Methods in Computer Vision, L. Griffin and M. Lillholm, eds., vol. 2695, Springer LNCS, Isle of Skye, UK, 2003, pp. 432–448.
[25] B. McCane, K. Novins, D. Crannitch, and B. Galvin, *On benchmarking optical flow*, Computer Vision and Image Understanding, 84 (2001), pp. 126–143.
[26] J. Modersitzki, *Numerical methods for image registration*, Oxford University Press, Oxford, 2004.
[27] H.-H. Nagel and W. Enkelmann, *An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences*, IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI-8 (1986), pp. 565–593.
[28] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, in Proceedings of the eleventh annual international conference of the Center for Nonlinear Studies on Experimental mathematics : computational issues in nonlinear science, Amsterdam, The Netherlands, The Netherlands, 1992, Elsevier North-Holland, Inc., pp. 259–268.
[29] D. Suter, *Motion estimation and vector splines*, in CVPR94, 1994, pp. 939–942.
[30] D. Terzopoulos, *Image analysis using multigrid methods*, IEEE Transactions on Pattern Analysis and Machine Intelligence, (1986), pp. 129–139.
[31] U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, 2001.
[32] A. Verri and T. Poggio, *Motion field and optical flow: Qualitative properties*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 11 (1989), pp. 490–498.
[33] J. Weickert and C. Schnörr, *A theoretical framework for convex regularizers in pde-based computation of image motion*, International Journal of Computer Vision, 45 (2001), pp. 245—-264.
[34] J. Weickert and C. Schnörr, *Variational Optic Flow Computation with a Spatio-Temporal Smoothness Constraint*, Journal of Mathematical Imaging and Vision, 14 (2001), pp. 245–255.
[35] P. Wesseling, *Multigrid Methods*, Edwards, 2004.
[36] R. Wienands and W. Joppich, *Practical Fourier Analysis for Multigrid Methods*, vol. 5 of Numerical Insights, Chapmann and Hall/CRC, 2005. CD included.
[37] Y. Zheng, H. Köstler, N. Thürey, and U. Rüde, *Enhanced Motion Blur Calculation with Optical Flow*, in Proceedings of Vision, Modeling and Visualization, I. P. Aka GmbH, ed., RWTH Aachen, 2006, pp. 253–260.