Robust and efficient multigrid techniques for the optical flow problem using different regularizers

I. Christadler[*], H. Köstler[†], U. Rüde [‡]

12th August 2005

Abstract

Optical flow and the related non-rigid image registration both lead to a variational minimization problem that requires robust and efficient numerical solvers due to the often non-smooth input data and the large number of unknowns in real applications.

Existing multigrid algorithms highly depend on smooth input data. To ensure their efficiency often pre-smoothed images are used although this leads to difficulties particularly with small objects. In order to overcome these problems we supplied the basic multigrid algorithm with Galerkin coarsening and matrix-dependent transfer operators. Further improvements can be obtained by developing suitable line-wise or block smoothers and by using iterant recombination. Our synthetic and real world results demonstrate that we can relax the restrictions on the smoothness of the image data without loosing the fast multigrid convergence rate.

The visual quality of the results can be improved by using application dependent regularizers. We compare isotropic, anisotropic, and spatio-temporal diffusion based regularizers and show their additional requirements on the solver.

1 Introduction

Optical flow is one method to compute a dense approximate motion field $\mathbf{u}$ out of two or more subsequent images in a video sequence $I = (I_i | i \in \mathbb{N})$ describing the grey value intensities for every point in the image domain Ω. Optical flow, as described in this section, was introduced by Horn and Schunk [1] and since then it has been studied intensively (e.g. [2],[3]). The goal of this paper is to make existing fast multigrid based solvers for the optical flow (e.g. [4],[5],[6]) more robust and extend them to more advanced regularizers. For medical applications in 3D it is also necessary to parallelize the code due to the large

[*]Leibniz-Rechenzentrum, Germany, Iris.Christadler@lrz-muenchen.de
[†]University of Erlangen-Nuremberg, Germany, Harald.Koestler@informatik.uni-erlangen.de
[‡]University of Erlangen-Nuremberg, Germany, Ulrich.Ruede@informatik.uni-erlangen.de

amount of memory consumption. For the sake of simplicity we restrict ourselves to two grey value images $I_1 = R$ and $I_2 = T$ in 2D. The basic assumption for the optical flow field $\mathbf{u} = (u, v)$ is that a moving object in the image does not change its grey values. This means that we neglect changes in the illumination. Mathematically this assumption can be written as $T(\mathbf{x}) = R(\mathbf{x}+\mathbf{u})$, with $\mathbf{x} = (x_1, x_2)$. This equation cannot be fulfilled exactly for most real world examples. Therefore using the image derivatives R_x, R_y in space and R_t in time we introduce a difference measure $\Phi(I) = (R_x(\mathbf{x})u + R_y(\mathbf{x})v + R_t(\mathbf{x}))^2$ (note that this is a linearized version of $\Phi(I) = (T(\mathbf{x}) - R(\mathbf{x} + \mathbf{u}))^2)$ that has to be minimized. We call Φ the data term. Since this single condition leads to a under-determined system of equations one has to use a second one called regularizer Ψ. A simple choice is $\Psi(.) = \|\nabla \mathbf{u}\|^2$ which forces a smooth solution, i.e. the vector field $\mathbf{u}$ varies only smoothly between neighboring points (in space). In the variational formulation these two assumptions lead to the following minimization problem

$$\min_{\mathbf{u}} \int_{\Omega} (T(\mathbf{x}) - R(\mathbf{x} + \mathbf{u}))^2 + \alpha \|\nabla \mathbf{u}\|^2 \, d\Omega \tag{1}$$

over the domain Ω. With the parameter $\alpha \in \mathbb{R}_+$ we can change the weighting between data term and regularizer. Using the variational calculus we find that a solution for problem (1) has to fulfill the Euler-Lagrange equations

$$\begin{array}{rcll} -\alpha\Delta\mathbf{u} + \nabla R((\nabla R)^T\mathbf{u} + R_t) & = & 0 & \text{in } \Omega \\ \mathbf{u} & = & 0 & \text{on } \partial\Omega \end{array} . \tag{2}$$

The Dirichlet boundary conditions can also be replaced by Neumann boundary conditions. In order to solve this system of PDEs we first discretize it using finite differences on the rectangular regular grid Ω_h with mesh-size h. Δ_h denotes e.g. in 2D the usual 5-point stencil for the Laplacian, in 3D the usual 7-point stencil ([7]). For the discrete image derivatives we use the method suggested by Horn and Schunk ([1], for 3D see [8]). After the discretization one has to solve a system of linear equations $A\mathbf{u} = f$ with sparse system matrix A (cf. [9])

$$\begin{pmatrix} (R_x)^2 + \alpha\Delta u & R_x R_y \\ R_x R_y & (R_y)^2 + \alpha\Delta v \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -R_x R_t \\ -R_x R_t \end{pmatrix} .$$

2 Multigrid

Multigrid is a very efficient technique for solving sparse linear systems ([7]). In the following we use the notation from [10] and rewrite the system from the last section as

$$A_h u^h = f^h \;, \quad \sum_{j \in \Omega^h} a_{ij}^h u_j^h = f_i^h \; (i \in \Omega^h) \tag{3}$$

where A_h is a symmetric and positive definite matrix. The usual multigrid efficiency is achieved through the combination of two iterations, the smoother, and the coarse grid correction.

Normally, for problems defined on a regular grid, direct discretization is used for the computation of the coarse grid operator A_H. Since we have to deal with strongly jumping coefficients a multigrid method with direct discretization would converge slowly or even diverge. This is especially a problem for highly textured images or small choices of α. Therefore, we construct A_H based on the Galerkin principle, i.e. it is defined as the Galerkin operator

$$A_H = \mathcal{I}_h^H A_h \mathcal{I}_H^h \ ,$$

with the prolongation $\mathcal{I}_H^h : \Omega^H \mapsto \Omega^h$ and the restriction $\mathcal{I}_h^H : \Omega^h \mapsto \Omega^H$, which are both full rank linear mappings. The Galerkin-based coarse-grid corrections minimize the energy norm of the error with respect to all variations in the range of the prolongation ([10], p. 431).

Often, full weighting is chosen as restriction $\mathcal{I}_h^H$ and bilinear interpolation as interpolation operator $\mathcal{I}_H^h$. Since we can have highly jumping coefficients we used matrix-dependent transfer operators instead ([11], [12]).

We applied a coupled Gauss-Seidel or a damped Jacobi relaxations as smoother and to handle difficulties with jumping coefficients also line or block versions of both.

Another special technique used to improve the multigrid solver is iterant recombination ([13]), where two or more old iterants from previous cycles are used to determine the optimal coefficients of the linear combination of the latest approximations.

3 Advanced regularizers

To improve the quality of the computed optical flow field we used a number of advanced regularizers instead of the comparatively easy to discretize and to compute smooth regularizer by Horn and Schunk. All shown regularizers are based on the work of Weickert and Schnörr. These regularizers have been discretized anew to fit into our multigrid framework. A detailed overview of the regularizers can be found in [3]. The exact step-by-step discretization, setting up the system of equations, the incorporation of these regularizers into our multigrid context, and many results are given in [14]. A first impression from the benefits of advanced regularizers is shown in figure 1. The vanished .jpg artifacts in the background of the rightmost image are the most outstanding improvements besides the sharper and more recognizable silhouette of the man.

Isotropic image-driven regularizer The first idea for a more advanced regularizer is based on the assumption that edges in an image often belong to the outline of an object and therefore are likely to separate two adjacent motion vectors. The natural conclusion is to build a regularizer that smoothes little in the vicinity of image edges so that motion boundaries are not blurred. We use the regularizer

$$\Psi(.) := a(\|\nabla R\|^2)(\|\nabla \mathbf{u}\|^2)\,, \text{ with}$$
$$a(\|\nabla R\|^2) = 1 + (m-1)/(1 + \sqrt{R_x^2 + R_y^2 + R_t^2})\,, \quad m \geq 1\,.$$

Figure 1: The norm of the computed optical flow field for a man clapping his hands. *From left to right:* Horn-Schunk, isotropic image-driven, and flow-driven regularizer.

Anisotropic image-driven regularizer The isotropic regularizer can easily be extended to an anisotropic one which permits smoothing along edges but not perpendicular to them. We experimented with the regularizer

$$\Psi(.) = \nabla\mathbf{u}^T D(\nabla R)\nabla\mathbf{u}\,, \text{ with}$$
$$D(\nabla R) = \frac{1}{\|\nabla R\|^2 + \lambda^2}(\nabla R^{\perp}\nabla R^{\perp T} + \lambda^2\,\mathrm{Id}\,)\,, \quad \lambda > 1\,.$$

Flow-driven regularizer An even more advanced regularizer with appealing results is the flow-driven one. This regularizer is able to distinguish between image edges and real edges in the motion flow. It includes the so far computed optical flow vector field to get a better approximation in a following step. Unfortunately, the regularizer $\Psi(.) = \chi(\|\nabla\mathbf{u}\|^2)$ leads to a nonlinear system of equations

$$-\operatorname{div}(\chi'(\|\nabla\mathbf{u}\|^2)\nabla\mathbf{u}) + \nabla R((\nabla R)^T\mathbf{u} + R_t) = 0\,, \text{ with}$$
$$\chi'(s^2) = \varepsilon + (1-\varepsilon)/(2\sqrt{1+s^2/\lambda^2})\,, \quad \varepsilon > 0\,, \lambda \geq 1\,,$$

which cannot be solved straightforward with our multigrid algorithm. We used “Additive Operator Splitting” (AOS) [15] instead.

4 Experimental results

Our test problem consists of a point moving one pixel to the right. The true motion field for this test case would be 0 everywhere except from the position of the point, where it should be $(0,1)$. The optical flow is now expected to approximate the true motion field. The multigrid convergence rate depends on the condition number of the system matrix A_h from (3) that is small, when the condition

$$\max_{\mathbf{x}\in\Omega_h} \|\nabla R(\mathbf{x})\|^2 \approx \alpha \tag{4}$$

is fulfilled. That means that for small values of α in our test problem the convergence rate deteriorates if we use a standard multigrid method. In figure 2 we show the gain of using the techniques introduced in section 2 to improve our multigrid solver.

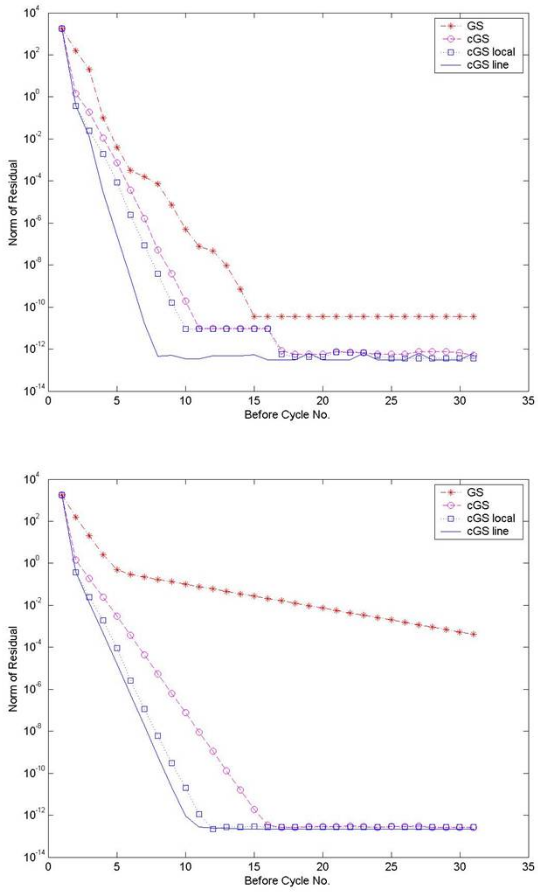

Figure 2: Residual norm history of V(2,2)-cycles for a moving point ($\alpha = 1$) using matrix-dependent transfer operators and different smoothers with (top) and without (bottom) iterant recombination .

5 Outlook

More details of the used multigrid techniques will be found in a forthcoming technical report. In our talk we will present more results from real data and explain the spatio-temporal regularizer and the possibility of including landmark based knowledge for lack of space in this article.

References

[1] Horn, B., Schunck, B.: Determining optical flow. Artificial Intelligence (1981) 185–203

[2] Battiti, R., Amaldi, E., Koch, C.: Computing optical flow across multiple scales: an adaptive coarse-to-fine strategy. International Journal of Computer Vision (1991) 133–145

[3] Weickert, J., Schnörr, C.: A Theoretical Framework for Convex Regularizers in PDE-Based Computation of Image Motion. Technical Report 13, University of Mannheim (2000)

[4] Bruhn, A., Weickert, J., et al.: Variational optic flow computation in real-time. Technical Report 89, Universität des Saarlandes (2003)

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

[6] Glazer, F.: Multilevel relaxation in low-level computer vision. Multi-resolution image processing and Analysis (A. Rosenfeld, Ed.) Springer-Verlag (1984) 312–330

[7] Briggs, W., Henson, V., McCormick, S.: A Multigrid Tutorial. Society for Industrial and Applied Mathematics, Philadelphia (2000)

[8] Köstler, H., Kalmoun, M., Rüde, U.: Parallel multigrid computation of the 3D optical flow. Technical Report 04-4, University of Erlangen-Nuremberg (2004)

[9] Kalmoun, M., Rüde, U.: A variational multigrid for computing the optical flow. Vision, Modeling and Visualization 2003 (Berlin 2003) 577–584 T. Ertl et al. (Ed.) Akademische Verlagsgesellschaft.

[10] Trottenberg, U., Oosterlee, C., Schüller, A.: Multigrid. Academic Press (2001)

[11] Alcouffe, R., Brandt, A., Dendy, J., Painter, J.: The Multi-Grid Method for the Diffusion Equation with Strongly Discontinuous Coefficients. SIAM J. Sci. Comput. **2** (1981) 430–454

[12] Knapek, S.: Matrix-Dependent Multigrid Homogenization for Diffusion Problems. SIAM J. Sci. Comput. **20** (1999) 515–533

[13] Brandt, A., Mikulinsky, V.: On recombining iterants in multigrid algorithms and problems with small islands. SIAM J. Sci. Comput. **16** (1995) 20–28

[14] Christadler, I.: Mehrgitterverfahren für die Berechnung des Optischen Flusses mit Nicht-Standardregularisierungen. Diploma thesis, University of Erlangen-Nuremberg (2004)

[15] Weickert, J., ter Haar Romeny, B.M., Viergever, M.A.: Efficient and Reliable Schemes for Nonlinear Diffusion Filtering. IEEE Transactions on Image Processing **Volume 7, Number 3** (1998) 398–410