MULTIGRID COMPUTATION OF ROTATIONALLY INVARIANT NON-LINEAR OPTICAL FLOW

Chris Alvino, Allen Tannenbaum, Anthony Yezzi

Georgia Institute of Technology
School of Electrical and Computer Engineering
Atlanta, GA, USA

Cecilia Curry

Centers for Disease Control and Prevention
Atlanta, GA, USA

ABSTRACT

In supplement to an earlier paper, we present an altered cost functional for the computation of an edge-preserving optical flow that is invariant to rotation. In addition, we explain how the solutions to the resulting non-linear partial differential equations may be computed more efficiently with non-linear multigrid techniques. We prove the rotational invariance of this functional and report computation times on a real image sequence.

1. INTRODUCTION

Optical flow is used to extract motion information from a sequence of images. Optical flow offers a valuable quantification for motion such as for object tracking, time-to-impact calculation, or for determining fluid flow in an image. Optical flow fields representing large scale motion, i.e., those that vary smoothly over an image, are well captured by the optical flow method proposed by Horn and Schunck [1]. Horn and Schunck proposed a regularizer that heavily penalizes changes in the optical flow field. The result is that the optical flow functional presented by Horn and Schunck does not preserve the occluding edges of moving objects well because the functional globally enforces a smoothly varying flow field.

Authors have proposed different regularizers in order to capture certain desirable properties of optical flow fields. Hildreth presented a method to ensure smoothness along prespecified contours [2]. Nagel proposed an oriented smoothness constraint that constrains the optical flow field in directions along which optical flow cannot be determined from gray value changes [3].

Kumar et al. recommend the use of regularizer whose calculation requires solving a non-linear PDE for the purpose of preserving edges [4]. It has occurred to the authors that the functional presented in this paper is not invariant to rotations in the image. A regularization functional that is not invariant to rotation will produce optical flow fields that are dependent on orientation of the image with respect to the x- and y-axis. Unlike the linear partial differential equations (PDEs) resulting from Horn and Schunck's functional, the partial differential equations resulting in [4] are non-linear, and difficult to compute with traditional gradient descent methods.

Gradient descent methods, while simple to implement, are slow to approximate the coarse scale features of a function. Multigrid relaxation methods, which address this issue, have been of wide interest in the mathematical and image processing communities [5, 6, 7, 8]. Multigrid relaxation methods take advantage of the coarse scale structure of the solution function by solving on a refined, coarser grid, where the computational cost is low [6]. Multigrid techniques therefore compute a coarse scale approximation to the solution quickly. Optical flow fields typically have significant coarse scale structure and thus are well suited to multigrid computation. Glazer applied multigrid techniques to Horn and Schunck's optical flow functional [9]. Battiti et al. have introduced an adaptive multiscale approach in the flavor of multigrid for computing Horn and Schunck optical flow [10]. Enkelmann applied multigrid techniques to the optical flow constraint presented by Nagel [11].

In this paper, we present an altered functional that is similar to that presented in [4], but is rotationally invariant. We present proof of the rotational invariance of this new functional and the improved computational performance. We also compute an optical flow fields that minimize this functional with non-linear multigrid techniques. The increase in computational speed is significant.

2. OPTICAL FLOW FUNCTIONAL

Let $I : \mathbb{R}^3 \rightarrow \mathbb{R}$ represent a time varying image, i.e., $I(x, y, t)$ is the intensity value at spatial location (x, y) and at time t. Let $u(x, y)$ and $v(x, y)$ be the horizontal and vertical components of the optical flow, respectively. Under the assumptions of brightness constancy and smoothly varying intensity patterns, Horn and Schunck [1] motivate the use of an

0-7803-9134-9/05/$20.00 ©2005 IEEE

optical flow constraint,

$$I_x u + I_y v + I_t = 0\,, \tag{1}$$

where I_x, I_y, and I_t denote the partial derivatives of the image function with respect the subscripted variable. Optical flow functions u and v which solve Eq. (1) capture motion in image intensity in directions orthogonal to isobrightness contours.

However, as is well known in the optical flow literature [1, 12, 2], the solution to Eq. (1) is not well-posed since optical flow can not be determined along isobrightness contours. To resolve this ambiguity, authors have presented a regularization in the form of a term that penalizes the departure from smoothness of the optical flow functions, u and v. [1, 4]. Horn and Schunck defined departure from smoothness as,

$$e_{\text{HS}}(u,v) = \iint_{\text{Image}} u_x^2 + u_y^2 + v_x^2 + v_y^2 \, dx\, dy\,, \tag{2}$$

where subscripted spatial variables again denote partial derivatives. More compactly, we can denote $e_{\text{HS}}(u,v)$ as $e_{\text{HS}}(\mathbf{u})$, where $\mathbf{u} = \begin{pmatrix} u(x,y) \\ v(x,y) \end{pmatrix}$.

By casting the optical flow constraint in Eq. (1) in terms of a functional,

$$e_{\text{c}}(u,v) = \iint_{\text{Image}} \left(I_x u + I_y v + I_t\right)^2 \, dx\, dy\,, \tag{3}$$

the problem of solving for optical flow becomes well-posed and we are able to find the u and v that minimize the combined functional, $\lambda e_{\text{c}} + e_{\text{HS}}$. Note that λ is a constant parameter that controls the penalty tradeoff between the optical flow term and the smoothness term.

The smoothness term, e_{HS}, presented by Horn and Schunck has the advantage of being well-behaved numerically since the Euler-Lagrange equations for the functional $\lambda e_{\text{c}} + e_{\text{HS}}$ produce the coupled pair of second-order elliptic partial differential equations (PDEs) given by,

$$\Delta u = \lambda(I_x u + I_y v + I_t) I_x\,, \tag{4}$$

$$\Delta v = \lambda(I_x u + I_y v + I_t) I_y\,, \tag{5}$$

where $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ denotes the Laplacian operator.

However, a disadvantage of Horn and Schunck's smoothness term is that is does not allow for the sharp changes in optical flow that occur near edges of moving objects. Thus, while Horn and Schunck's method is useful for optical flow fields which vary smoothly over the entire image, it also has the undesirable effect of smoothing the boundaries of moving objects.

To combat this problem, Kumar et al. [4] present an edge-preserving smoothness term, defined as,

$$e_{\text{K}}(u,v) = \iint_{\text{Image}} \sqrt{u_x^2 + u_y^2} + \sqrt{v_x^2 + v_y^2}\, dx\, dy\,, \tag{6}$$

as a replacement for Horn and Schunck's smoothness term.

We present an optical flow smoothness term that has similar edge-preservation properties while maintaining the rotational invariance of Horn and Schunck's smoothness term. We propose the use of,

$$e_{\text{RI}}(u,v) = \iint_{\text{Image}} \sqrt{u_x^2 + u_y^2 + v_x^2 + v_y^2}\, dx\, dy\,, \tag{7}$$

In the following discussion, we will define the function, $L = \sqrt{u_x^2 + u_y^2 + v_x^2 + v_y^2}$, i.e., L is the integrand of Eq. (7). As a result from the calculus of variations, the proposed error functional, $\lambda e_{\text{c}} + e_{\text{RI}}$, is minimized by the solution to the non-linear partial differential equations given by,

$$-\frac{\partial}{\partial x}\frac{u_x}{L} - \frac{\partial}{\partial y}\frac{u_y}{L} + \lambda(I_x u + I_y v + I_t) I_x = 0\,, \tag{8}$$

$$-\frac{\partial}{\partial x}\frac{v_x}{L} - \frac{\partial}{\partial y}\frac{v_y}{L} + \lambda(I_x u + I_y v + I_t) I_y = 0\,. \tag{9}$$

The computation of Horn and Schunck's optical flow is very fast in comparison to the computation of the optical flow resulting from regularizer in (7). This is due to the parabolic nature of the error surface resulting from Horn and Schunck's regularizer as compared to the cone-like error surface resulting from the regularizer in (7). Thus, the need for increased computational speed is greater when using the regularizer in (7). We suggest the use of non-linear multigrid methods to solve these partial differential equations.

2.1. Rotational Invariance

Consider an optical flow field, $\mathbf{u}$, and a rotation by θ, $\mathbf{Ru}$, where,

$$\mathbf{R} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}\,, \tag{10}$$

is a two dimensional rotation matrix. Then the rotated optical flow field is given by $\mathbf{u}_r = \mathbf{Ru}$.

We claim that the functional in Eq. (6) is not invariant to rotation and the functional in Eq. (7) is invariant to rotation. To show that the cost functional in Eq. (7) is invariant to rotation, we substitute the rotated optical flow into the cost functional. We will denote this as $e_{\text{RI}}(\mathbf{u}_r)$. This substitution yields that the integrand of $e_{\text{RI}}(\mathbf{u}_r)$ is,

$$\sqrt{\begin{array}{c}(u_x\cos\theta - v_x\sin\theta)^2 + (u_y\cos\theta - v_y\sin\theta)^2 \\ +(u_x\sin\theta + v_x\cos\theta)^2 + (u_y\sin\theta + v_y\cos\theta)^2\end{array}} \tag{11}$$

$$= \sqrt{\begin{array}{c}u_x^2\cos^2\theta + v_x^2\sin^2\theta + u_y^2\cos^2\theta + v_y^2\sin^2\theta \\ +u_x^2\sin^2\theta + v_x^2\cos^2\theta + u_y^2\sin^2\theta + v_y^2\cos^2\theta\end{array}} \tag{12}$$

$$= \sqrt{u_x^2 + u_y^2 + v_x^2 + v_y^2}\,, \tag{13}$$

which is identical to the integrand of the functional without rotation of the optical flow. A similar computation with the

function from Eq. (6) results in the integrand,

$$\sqrt{\begin{array}{c}\left(u_x^2\cos^2\theta + v_x^2\sin^2\theta - 2u_xv_x\cos\theta\sin\theta\right)\\ +\left(u_y^2\cos^2\theta + v_y^2\sin^2\theta - 2u_yv_y\cos\theta\sin\theta\right)\end{array}} \quad (14)$$
$$+ \sqrt{\begin{array}{c}\left(u_x^2\sin^2\theta + v_x^2\cos^2\theta + 2u_xv_x\cos\theta\sin\theta\right)\\ +\left(u_y^2\sin^2\theta + v_y^2\cos^2\theta + 2u_yv_y\cos\theta\sin\theta\right)\end{array}},$$

which is only necessarily equal to the original functional when θ is an integer multiple of 90 degrees.

3. COMPUTATION

3.1. Gradient Descent

As a result from the calculus of variations, the left hand side of Eqs. (8) and (9), act as infinite dimensional gradients for the functions u and v. That is by augmenting the functions u and v with an artificial evolution parameter, s, we obtain the evolutions,

$$\frac{\partial u}{\partial s} = -\frac{\partial}{\partial x}\frac{u_x}{L} - \frac{\partial}{\partial y}\frac{u_y}{L} + \lambda(I_xu + I_y + I_t)I_x \quad (15)$$
$$\frac{\partial v}{\partial s} = -\frac{\partial}{\partial x}\frac{v_x}{L} - \frac{\partial}{\partial y}\frac{v_y}{L} + \lambda(I_xu + I_y + I_t)I_y \quad (16)$$

We numerically implement this gradient descent using the forward Euler scheme, i.e., at each iteration, by replacing u and v by corrected versions as follows,

$$u^{n+1} = u^n - \gamma\widehat{\frac{\partial u^n}{\partial s}}, \quad (17)$$
$$v^{n+1} = v^n - \gamma\widehat{\frac{\partial v^n}{\partial s}}. \quad (18)$$

Note that the superscript denotes the iteration, γ is a time step parameter, and $\widehat{\frac{\partial u^n}{\partial s}}$ and $\widehat{\frac{\partial v^n}{\partial s}}$ denote numerical approximations to the gradient at step n. We compute these with central difference approximations in the center of the image and use Neumann boundary conditions, i.e., that normal derivatives at the image boundaries are zero.

3.2. Multigrid

We solve for the optical flow components u and v using multigrid relaxation methods. Since Eqs. (8) and (9) are nonlinear partial differential equations, it is necessary to use an algorithm that handles non-linear operators, known as the Full Approximation Storage (FAS) algorithm.

We recast Eqs. (8) and (9) in the following fashion,

$$-\frac{\partial}{\partial x}\frac{u_x}{L} - \frac{\partial}{\partial y}\frac{u_y}{L} + \lambda(I_xu + I_yv)I_x = -\lambda I_tI_x \quad (19)$$
$$-\frac{\partial}{\partial x}\frac{v_x}{L} - \frac{\partial}{\partial y}\frac{v_y}{L} + \lambda(I_xu + I_yv)I_y = -\lambda I_tI_x \quad (20)$$

which we can write compactly as $\mathcal{L}(u,v) = f$, whereby the left hand side of the equation is a non-linear operator on the optical flow functions u and v. It is nonlinear due to the appearance of the L in the denominator.

The algorithm originates with an initial guess of the optical flow functions, $\tilde{u}$ and $\tilde{v}$. The multigrid method then seeks corrections, $\hat{u}$ and $\hat{v}$, to the initial guess functions such that,

$$\mathcal{L}(\tilde{u}+\hat{u}, \tilde{v}+\hat{v}) = f. \quad (21)$$

In order to find $\hat{u}$ and $\hat{v}$, we write,

$$\mathcal{L}(\tilde{u}+\hat{u}, \tilde{v}+\hat{v}) - \mathcal{L}(\tilde{u},\tilde{v}) = f - \mathcal{L}(\tilde{u},\tilde{v}) = -r. \quad (22)$$

where r is known as the *residual*. By decimating r, $\tilde{u}$, and $\tilde{v}$ to a coarser grid, thus obtaining r_c, $\tilde{u}_c$, and $\tilde{v}_c$, and by also identifying a coarse grid version of the operator $\mathcal{L}$ as $\mathcal{L}_c$, we can solve for coarse grid versions of the corrections $\hat{u}$ and $\hat{v}$ by solving,

$$\mathcal{L}_c(\hat{u}_c, \hat{v}_c) = \mathcal{L}_c(\tilde{u}_c, \tilde{v}_c) - r, \quad (23)$$

for $\hat{u}_c$ and $\hat{v}_c$.

Eq. (23) is solved for $\hat{u}_c$ and $\hat{v}_c$ on the coarsified grid, where the computational cost is lower than on the original grid, using the gradient descent technique described in Section 3.1. Once $\hat{u}_c$ and $\hat{v}_c$ are obtained, the corrections $\hat{u}$ and $\hat{v}$ are computed by,

$$\hat{u} = \mathcal{U}(\hat{u}_c - \tilde{u}_c) \quad (24)$$
$$\hat{v} = \mathcal{U}(\hat{v}_c - \tilde{v}_c) \quad (25)$$

where $\mathcal{U}$ denotes an upsampling operator. The initial guess functions are then updated by, $\tilde{u} \leftarrow \tilde{u} + \hat{u}$ and $\tilde{v} \leftarrow \tilde{u} + \hat{v}$, and finally, gradient descent is used on the resulting functions, $\tilde{u}$ and $\tilde{v}$, in order to obtain the fine scale features of the solution.

The above mentioned algorithm is, in fact, a two grid solution method. However, note that while Eq. (23) can be solved using gradient descent or a similar relaxation technique, it is also in a form similar to Eq. (21) and thus can be solved by the same coarse grid correction process on an even coarser grid where the computational cost is even lower. This process can be carried out until the computational cost of solving with gradient descent on the coarsest grid is negligible. More details about the implementation of multigrid algorithms can be found in [13].

4. SIMULATIONS

The left side of Fig. (1) shows a frame from the well-known Hamburg Taxi sequence in which two cars, one with high intensity and one with low intensity, are moving in the image, while the rest of the scene remains stationary. The circular

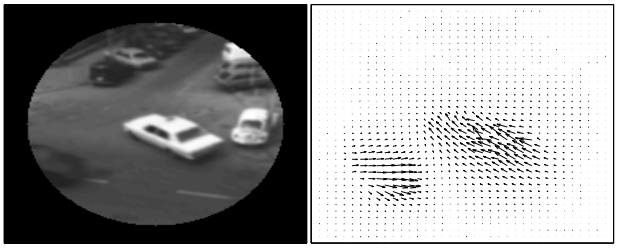

Fig. 1. Left: frame from Hamburg Taxi sequence. Right: proposed optical of Hamburg Taxi sequence.

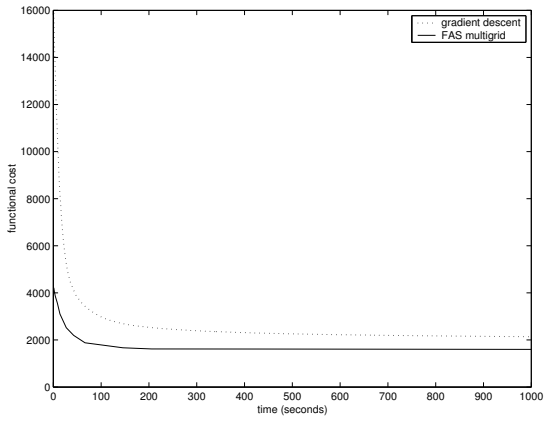

Fig. 2. Functional cost versus computation time for the gradient descent and FAS multigrid algorithms.

window was introduced to show the edge preserving nature of the proposed regularizer.

The right side of Fig. (1) shows the proposed optical flow field on a circular window of the Hamburg Taxi sequence. This image was the result of the FAS multigrid algorithm explained above using 5 different grids. The original image was 189 pixels by 189 pixels and each successively coarser grid was downsampled by a factor of 2. Thus, the final grid was 5 pixels by 5 pixels. Note the edge preserving nature of the optical flow field around the circular window and the properly captured lack of motion between the two cars.

Fig. (2) shows the value of the proposed functional, $\lambda e_{\mathrm{C}} + e_{\mathrm{RI}}$, for the gradient descent algorithm versus time and the value of the same functional for the FAS multigrid algorithm versus time for this frame of the Hamburg Taxi sequence. Note that the FAS multigrid algorithm nears complete convergence within 200 seconds. All computations were performed on a 1.8 GHz computer. The gradient descent algorithm requires several hours (not shown) to approach convergence under the same conditions due to the cone-like error surface mentioned above.

5. REFERENCES

[1] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[2] E. C. Hildreth, "Computations underlying the measurement of visual motion," *Artificial Intelligence*, vol. 23, pp. 309–354, 1984.

[3] H. H. Nagel, "Constraints for the estimation of displacement vector fields from image sequences," in *Int. Joint Conference on Artificial Intelligence*, Karlsruhe, FRG, August 1983, pp. 945–951.

[4] A. Kumar, A. R. Tannenbaum, and G. J. Balas, "Optical flow: A curve evolution approach," *IEEE Transactions on Image Processing*, vol. 5, no. 4, pp. 598–610, 1996.

[5] A. Brandt, "Multi-level adaptive solutions to boundary value problems," *Math. Comput.*, vol. 31, pp. 333–390, 1977.

[6] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, SIAM, second edition, 2000.

[7] D. Terzopoulos, "Image analysis using multigrid relaxation methods," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 129–139, 1986.

[8] S. Acton, "Multigrid anisotropic diffusion," *IEEE Trans. Image Proc.*, vol. 7, no. 3, pp. 280–291, March 1998.

[9] F. Glazer, *Multi-Resolution Image Processing and Analysis*, chapter Multilevel relaxation in low-level computer vision, pp. 312–330, Springer-Verlag, New York, 1984.

[10] R. Battiti, E. Amaldi, and C. Koch, "Computing optical flow across multiple scales: An adaptive corarse-to-fine strategy," *International Journal of Computer Vision*, vol. 6, no. 2, pp. 133–145, 1991.

[11] W. Enkelmann, "Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences," *Computer Vision, Graphics, and Image Processing*, vol. 43, pp. 150–177, 1988.

[12] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM Computing Surveys*, vol. 27, no. 3, pp. 433–467, 1995.

[13] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, Cambridge University Press, second edition, 1992.