Determining Optical Flow using a Modified Horn and Schunck's Algorithm

Rafael A. B. de Queiroz, Gilson A. Giraldi, Pablo J. Blanco, Raúl A. Feijóo
National Laboratory for Scientific Computing - LNCC
Petropólis-RJ, Brazil
bonfimraf@gmail.com

Abstract—The computation of the optical flow field can be performed through the minimization of some energy functional that consists of two terms: a data term, that requires brightness constancy of patterns in the image sequence, and a regularization term to guarantee piecewise smoothness and to avoid ill-posed problems. In this paper, we propose a new regularization term based on the symmetric gradient of the flow. The new algorithm is discussed in terms of invariance and a numerical scheme is developed based on Horn and Schunck's technique. In the numerical results, we compare our method with the traditional Horn and Schunck's algorithm to show the potential of our formulation when considering efficiency and precision.

Keywords - optical flow; regularization term.

I. INTRODUCTION

According to Horn and Schunck, Optical Flow (OF) is the distribution of apparent velocities of movement of brightness patterns in an image [5]. The robust computation of this flow is strongly needed for many applications in computer vision and medical image analysis [3], [4]. There are a great deal of papers about OF calculation. For instance, Barron et al. [2] summarize the major algorithms and McCane et al. [6] evaluate the performance of seven OF algorithms using synthetic and real image sequences.

For the OF estimation, the Horn and Schunck's algorithm is one of the most used due to its simplicity and efficiency, which justifies and motivates the study reported in this work. We focus our attention on the smoothness constraint (regularizer) of this algorithm [5]. It computes the regularization term as the sum of the square magnitudes of the gradients of the OF velocity components [1], [5]. The mathematical foundations behind regularization theory in computer vision and OF is considered in other works [7], [8].

The focus of the present paper is to examine the performance and efficiency of Horn and Schunck's algorithm when the smoothness constraint is based on the symmetric gradient. Up to the best of our knowledge, it is a new proposal in OF computation. We provide the description of the new algorithm and, in the numerical experiments, we compare the classical and modified algorithms using synthetic data in the experimental tests.

The paper is organized as follows. In Section II, the algorithm of Horn and Schunck is presented. The proposed algorithm is described in Section III. Two numerical tests are shown in Section IV. Finally, Section V contains our conclusions and future directions of this work.

II. HORN AND SCHUNCK'S ALGORITHM

A sequence of 2D images is mathematically described as a function $I(x, y, t)$, where I is the image intensity at time t and at position (x, y). The algorithm of Horn and Schunck computes the velocity $\vec{v} = (u, v)$ for each pixel of the image by minimizing the functional given by

$$F = \int_{\Omega} [\alpha^2 E_c{}^2 + E_b^2] d\Omega, \tag{1}$$

where α is a regularization parameter; E_b^2 and E_c^2 are, respectively, the data and regularization terms given by

$$E_b^2 = \left(I_x u + I_y v + I_t\right)^2, \tag{2}$$

$$E_c^2 = (u_x)^2 + \left(u_y\right)^2 + (v_x)^2 + \left(v_y\right)^2 = \left\|\nabla \vec{v}\right\|_2^2. \tag{3}$$

In (2), $I_x = \dfrac{\partial I}{\partial x}$, $I_y = \dfrac{\partial I}{\partial y}$ and $I_t = \dfrac{\partial I}{\partial t}$. In (3), $u_x = \dfrac{\partial u}{\partial x}$, $u_y = \dfrac{\partial u}{\partial y}$, $v_x = \dfrac{\partial v}{\partial x}$ and $v_y = \dfrac{\partial v}{\partial y}$. By minimizing the functional (1), we obtain the Euler's equations

$$\begin{cases} uI_x^2 + v(I_x I_y) = \alpha^2 \nabla^2 u - I_x I_t, \\ vI_y^2 + u(I_x I_y) = \alpha^2 \nabla^2 v - I_y I_t. \end{cases} \tag{4}$$

A. Numerical Scheme

In (4), $\nabla^2 u$ and $\nabla^2 v$ are approximated by [5]

$$\nabla^2 u = 3(\bar{u}_{i,j,k} - u_{i,j,k}), \qquad \nabla^2 v = 3(\bar{v}_{i,j,k} - v_{i,j,k}), \tag{5}$$

where $\bar{u}_{i,j,k}$ and $\bar{v}_{i,j,k}$ are local average values between the pixels estimated by [5]

$$\bar{u}_{i,j,k} = \frac{1}{6}\left(u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k}\right) + \frac{1}{12}\left(u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k}\right), \quad (6)$$

$$\bar{v}_{i,j,k} = \frac{1}{6}\left(v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k}\right) + \frac{1}{12}\left(v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k}\right). \quad (7)$$

By replacing (5) in (4), we can obtain the system

$$\begin{cases} u(3\alpha^2 + I_x^2) + v(I_x I_y) = 3\alpha^2 \bar{u} - I_x I_t, \\ v(3\alpha^2 + I_y^2) + u(I_x I_y) = 3\alpha^2 \bar{v} - I_y I_t. \end{cases}$$

It is iteratively solved through the procedure [5]

$$u^{n+1} = \frac{\bar{u}^n(3\alpha^2 + I_x^2 + I_y^2) - I_x(I_x\bar{u}^n + I_y\bar{v}^n + I_t)}{3\alpha^2 + I_x^2 + I_y^2}$$

$$v^{n+1} = \frac{\bar{v}^n(3\alpha^2 + I_x^2 + I_y^2) - I_y(I_x\bar{u}^n + I_y\bar{v}^n + I_t)}{3\alpha^2 + I_x^2 + I_y^2}$$

where n is the current iteration of the numerical procedure; $\bar{u}^n$ and $\bar{v}^n$ are estimated by (6)-(7). I_x, I_y and I_t are, respectively, approximated by [5]

$$I_x \approx 0.25(I_{i,j+1,k} - I_{i,j,k} + I_{i+1,j+1,k} - I_{i+1,j,k}) + 0.25(I_{i,j+1,k+1} - I_{i,j,k+1} + I_{i+1,j+1,k+1} - I_{i+1,j,k+1}), \quad (8)$$

$$I_y \approx 0.25(I_{i+1,j,k} - I_{i,j,k} + I_{i+1,j+1,k} - I_{i,j+1,k}) + 0.25(I_{i+1,j,k+1} - I_{i,j,k+1} + I_{i+1,j+1,k+1} - I_{i,j+1,k+1}), \quad (9)$$

$$I_t \approx 0.25(I_{i,j,k+1} - I_{i,j,k} + I_{i+1,j,k+1} - I_{i+1,j,k}) + 0.25(I_{i,j+1,k+1} - I_{i,j+1,k} + I_{i+1,j+1,k+1} - I_{i+1,j+1,k}), \quad (10)$$

where the *j* index corresponds to the *x* direction in the image, the *i* index to the *y* direction, the *k* index lies in the time direction.

III. MODIFIED HORN AND SCHUNCK'S ALGORITHM

In this section, the proposed method is presented. It computes the velocity $\vec{v} = (u, v)$ for each pixel of the image by minimizing the functional given by

$$F = \int_\Omega [\alpha^2 G_s{}^2 + E_b^2] d\Omega, \quad (11)$$

where $G_s^2 = \left\|\nabla^s \vec{v}\right\|_2^2$, $\nabla^s \vec{v} = \dfrac{\nabla\vec{v} + (\nabla\vec{v})^T}{2}$ is called symmetric gradient and E_b^2 is computed by (2). By minimizing the functional (11), we obtain the Euler's equations

$$\begin{cases} uI_x^2 + v(I_x I_y) = \alpha^2\left[\nabla^2 u + \dfrac{1}{2}\dfrac{\partial}{\partial y}\left(\dfrac{\partial v}{\partial x} - \dfrac{\partial u}{\partial y}\right)\right] - I_x I_t, \\ vI_y^2 + u(I_x I_y) = \alpha^2\left[\nabla^2 v + \dfrac{1}{2}\dfrac{\partial}{\partial x}\left(\dfrac{\partial u}{\partial y} - \dfrac{\partial v}{\partial x}\right)\right] - I_y I_t. \end{cases} \quad (12)$$

Before considering the numerical issues, we shall discuss some aspects of the proposed method. Firstly, G_s^2 remains invariant when replacing $\vec{v}$ by $\vec{w} = \vec{v} + (\beta y + C_1, -\beta x + C_2)$ because $\nabla^s \vec{v} = \nabla^s \vec{w}$. Also, a simple algebra shows that $\left\|\nabla^s \vec{v}\right\|_2^2 = \left\|\nabla \vec{v}\right\|_2^2 - 0.5(u_y - v_x)^2$. Moreover, $(u_y - v_x)^2 = \left\|\nabla \times \vec{v}\right\|_2^2$. Therefore, G_s^2 is also rotation invariant regularizer [8] because if we replace $\vec{v}$ by $\vec{w} = A\vec{v}$ with A a rotation (orthogonal) matrix then we can show that

$\left\|\nabla^s \vec{w}\right\|_2^2 = \left\|\nabla \vec{w}\right\|_2^2 - 0.5\left\|\nabla \times (A\vec{v})\right\|_2^2$, i.e.,

$\left\|\nabla^s \vec{w}\right\|_2^2 = \left\|\nabla \vec{v}\right\|_2^2 - 0.5\left\|\nabla \times \vec{v}\right\|_2^2$.

A. Numerical Scheme

In (12), $\nabla^2 u$ and $\nabla^2 v$ are approximated by (5). The cross-partial derivatives in (12) are computed by

$$\begin{cases} \dfrac{\partial}{\partial y}\left(\dfrac{\partial v}{\partial x} - \dfrac{\partial u}{\partial y}\right) \approx 2(\Phi_u + u), \\ \dfrac{\partial}{\partial x}\left(\dfrac{\partial u}{\partial y} - \dfrac{\partial v}{\partial x}\right) \approx 2(\Phi_v + v), \end{cases} \quad (13)$$

where the difference operators Φ_u and Φ_v are determined by

$$\Phi_u = -\frac{1}{2}\left(u_{i+1,j,k} + u_{i-1,j,k}\right) + \frac{1}{8}\left(v_{i+1,j+1,k} - v_{i-1,j+1,k} - v_{i+1,j-1,k} + v_{i-1,j-1,k}\right), \quad (14)$$

$$\Phi_v = -\frac{1}{2}\left(v_{i,j+1,k} + v_{i,j-1,k}\right) + \frac{1}{8}\left(u_{i+1,j+1,k} - u_{i-1,j+1,k} - u_{i+1,j-1,k} + u_{i-1,j-1,}\right). \quad (15)$$

By replacing (5) and (13) in (12), we can obtain the system

$$\begin{cases} u(2\alpha^2 + I_x^2) + v(I_x I_y) = 3\alpha^2 \bar{u} + \alpha^2\Phi_u - I_x I_t, \\ v(2\alpha^2 + I_y^2) + u(I_x I_y) = 3\alpha^2 \bar{v} + \alpha^2\Phi_v - I_y I_t. \end{cases}$$

It is iteratively solved by

$$u^{n+1} = \frac{(3\bar{u}^n + \Phi_u^n)(I_y^2 + 2\alpha^2) - (3\bar{v}^n + \Phi_v^n)I_x I_y - 2I_x I_t}{4\alpha^2 + 2I_x^2 + 2I_y^2},$$

$$v^{n+1} = \frac{(3\bar{v}^n + \Phi_v^n)(I_x^2 + 2\alpha^2) - (3\bar{u}^n + \Phi_u^n) I_x I_y - 2 I_y I_t}{4\alpha^2 + 2I_x^2 + 2I_y^2},$$

where $\bar{u}^n$ and $\bar{v}^n$ are estimated by (6)-(7), Φ_u^n and Φ_v^n are calculated through (14)-(15). I_x, I_y and I_t are approximated by (8)-(10).

IV. EXPERIMENTAL TESTS

In order to demonstrate the efficiency and robustness of the proposed algorithm, two numerical experiments will be carried out and discussed. We compare the performance of the developed model with the original algorithm of Horn and Schunck using two error metrics. The first one, namely $E(\theta)$, is the mean of the angular error θ computed by $\theta = \cos^{-1}(\vec{v} \cdot \hat{v})$, where $\hat{v}$ is the correct motion vector and $\vec{v}$ is the estimated OF vector.

The second error metrics is the mean-squared-error (*MSE*) defined by:

$$MSE = \frac{1}{2(M \times N)} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(\vec{v}_{i,j} - \hat{v}_{i,j}\right)^2,$$

where $M \times N$ is the spatial resolution of the image.

For the test cases, we stop the iterative scheme when $|F^{n+1} - F^n| / |F^n| < 10^{-3}$, where F is the value of the functional (1) or (11) in iteration n of the algorithms. Fig. 1 illustrates the first frames of the synthetic sequence in cases 1 and 2, which has 80×80 pixels.

(a) Test case 1 (b) Test case 2

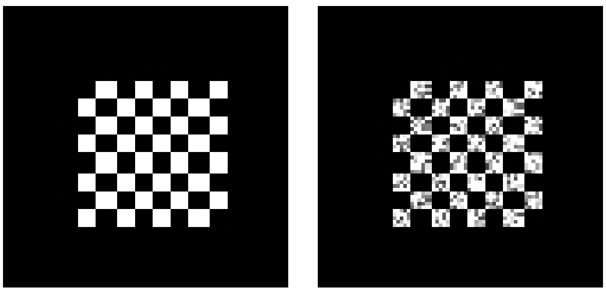

Figure 1. First frames of the synthetic sequence in the test cases.

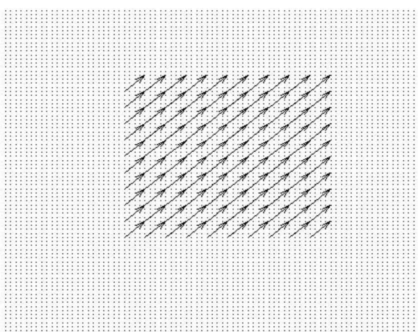

Figure 2. Real OF for cases 1 and 2.

(a) **Test case 1** - A texture is moving with the constant speed $\vec{v} = (1,1)$ pixel/frame (see Fig.1a). Considering $\alpha = 0.2$ (regularization parameter), the OF estimated by both the original algorithm of Horn and Schunck and the proposed model are pictured in Figs. 3 and 4, respectively. As one can see from these figures, the algorithms provided reasonable results when compared with the real OF (see Fig. 2). Tables I and II show the convergence rate and errors of the algorithms. It can be seen that the performance of the new model is quite similar to the original algorithm when the α regularization parameter increase.

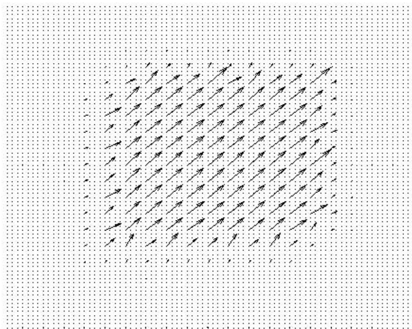

Figure 3. OF estimated in case 1 by the original Horn and Schunck algorithm.

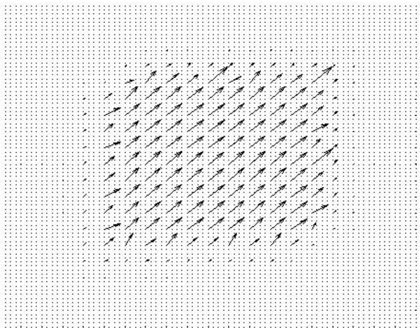

Figure 4. OF estimated in case 1 by the proposed algorithm.

TABLE I. CASE 1 - PERFORMANCE OF HORN AND SCHUNCK´S ALGORITHM

α	Iteration number	$E(\theta)$	***MSE***
0.1	19	1.573	0.027
0.2	26	1.522	0.028
0.4	50	1.572	0.034
0.8	107	1.571	0.040
1.0	134	1.571	0.042

TABLE II. CASE 1 - PERFORMANCE OF THE MODIFIED ALGORITHM

α	Iteration number	$E(\theta)$	***MSE***
0.1	20	1.573	0.029
0.2	25	1.572	0.029
0.4	43	1.571	0.034
0.8	87	1.571	0.041
1.0	110	1.571	0.042

(b) **Test case 2** - We add multiplicative noise to the image I in Fig. 1a, using equation $J = I + nI$, where n is uniformly distributed random noise with mean 0 and variance 0.2. The first frame of this case is shown in Fig.1b. Figs. 5 and 6 present the OF computed by the original algorithm of Horn and Schunck and proposed model, respectively, for $\alpha = 0.4$. It can be seen from these figures that both algorithms properly compute the OF when compared with the real OF (see Fig. 2). This observation is supported by the errors shown in Tables III and IV. Also, the number of iterations reported in these tables show that the convergence rate of the modified algorithm was a little superior to the one observed by the original method.

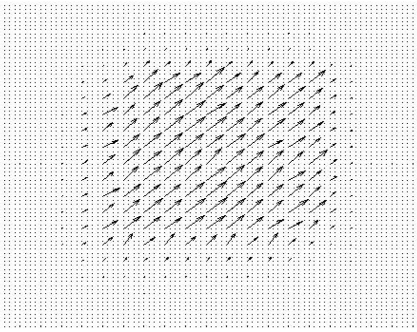

Figure 5. OF estimated in case 2 by the original Horn and Schunck algorithm.

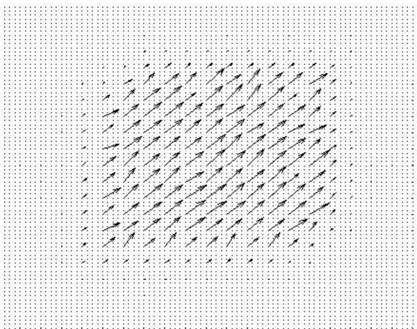

Figure 6. OF computed in case 2 by the proposed algorithm.

TABLE III. CASE 2 - PERFORMANCE OF HORN AND SCHUNCK´S ALGORITHM

α	Iteration number	$E(\theta)$	*MSE*
0.1	21	1.577	0.035
0.2	25	1.568	0.036
0.4	48	1.570	0.034
0.8	102	1.572	0.042
1.0	122	1.569	0.051

TABLE IV. CASE 2 - PERFORMANCE OF THE MODIFIED ALGORITHM

α	Iteration number	$E(\theta)$	*MSE*
0.1	20	1.565	0.040
0.2	24	1.572	0.038
0.4	44	1.575	0.035
0.8	85	1.571	0.042
1.0	106	1.569	0.046

V. CONCLUSION

The performance of the modified version of Horn and Schunck's algorithm has been evaluated on synthetic image sequences and compared with that of the classical algorithm. In the numerical experiments, the precision of the new model was practically similar to the original technique. Moreover, the tests indicate that the modified Horn and Schunck's algorithm has convergence rate a little superior to the original model. Further directions for this work are to explore connections with diffusion filters and the effects of the invariance observed in Section III for real world image sequences [8]. Also, more experimental tests with different movements will be achieved in order to allow more comparisons between the algorithms.

ACKNOWLEDGMENT

We gratefully acknowledge the support provided by CNPq (Grant 141474/2009-2). Also, we would like to acknowledge the reviewers for their valuable comments and suggestions that helped to improve the quality of the manuscript.

REFERENCES

[1] F. Bartolini, A. Piva and R. Piva, “Enhancement of the Horn and Schunck optic flow algorithm by means of median filters,” Proceedings 13th International Conference on Digital Signal Processing DSP97, pp. 503-506, 1997.

[2] J.L. Barron, D.J. Fleet and S.S. Beauchemin, “Performance of optical flow techniques,” The International Journal of Computer Vision, vol. 12, pp. 43-77, 1994.

[3] J. Konrad and E. Dubois, “Bayesian estimation of motion vector fields,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 14, pp. 910-927, 1992.

[4] M.A. Gutierrez, S.S. Furuie, M.S., Rebelo and J.C. Meneghetti, “Automatic algorithms to analyze and quantify cardiac left ventricle parameters by means of spect,” ESPC/SPI-B285-v-Medical Imaging, vol V, 2005.

[5] B.K.P. Horn and B.G. Schunck, “Determining optical flow,” Artificial Intelligence, vol. 17, pp. 185-203, 1981.

[6] B. McCane, K. Novins, D. Crannitch and B. Galvin,, “On benchmarking optical flow,” Computer Vision and Image Understanding, vol. 84, pp. 126-143, 2001.

[7] T.A. Poggio, V. Torre and C. Koch, “Computational vision and regularization theory,” Nature, pp. 314-319, 1985.

[8] J. Weickert and C. Schnorr, “A theoretical framework for convex regularizers in PDE-based computation of image motion,” International Journal of Computer Vision, vol. 45, pp. 245-264,2001.