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

Rafael Alves Bonfim de Queiroz,
Gilson Antonio Giraldi,
Pablo Javier Blanco,
Raúl Antonino Feijóo
National Laboratory for Scientific Computing – LNCC

***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.**

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 OF is strongly need for many applications in computer vision and medical image analysis [3], [4]. There are a lot of papers about OF computation. 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 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 several works (see [7], [8] and references therein).

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 by using synthetic data.

The paper is organized as follows. In Section II, the original algorithm of Horn and Schunck is described. The proposed method is presented in Section III. The numerical tests are shown in Section IV. Finally, Section V contains our conclusions and discusses 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 total derivative of change of brightness is given by

$$\frac{DI}{Dt} = \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t}, \tag{1}$$

where $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ can be computed directly from a pair of images $I(x,y,t)$ and $I(x,y,t+\delta t)$; $\frac{dx}{dt}$ and $\frac{dy}{dt}$ are the u and v components of velocity, respectively. Horn and Schunck suppose that, when the pattern moves, the intensity $I(x,y,t)$ of a particular point in the pattern is conserved, so that $\frac{DI}{Dt} = 0$ in expression (1). Considering this constraint, $I_x = \frac{\partial I}{\partial x}$, $I_y = \frac{\partial I}{\partial y}$ and $I_t = \frac{\partial I}{\partial t}$, equation (1) can be rewritten as:

$$\frac{DI}{Dt} = I_x u + I_y v + I_t = 0. \tag{2}$$

We must observe that this is a single equation with two unknowns variables u and v. Besides, no smoothness constraints and prior knowledge about the solution have been considered yet. All these problems can be addressed by introducing a regularization term [1], [7] and a variational formulation for the OF problem. Therefore, Horn and Schunck [5] compute u and v for each pixel of the image by minimizing the functional.

$$E^2 = \int_\Omega \left[\alpha^2 E_c^2 + E_b^2\right] d\Omega, \tag{3}$$

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

$$\begin{aligned} E_b^2 &= (I_x u + I_y v + I_t)^2, & (4)\\ E_c^2 &= (u_x)^2 + (u_y)^2 + (v_x)^2 + (v_y)^2 = \parallel \nabla \mathbf{v} \parallel_2^2, & (5)\end{aligned}$$

where $\mathbf{v} = (u,v)$. By minimizing the functional (3), we obtain the Euler equations:

$$\begin{aligned} I_x^2 u + I_x I_y v &= \alpha^2 \nabla^2 u - I_x I_t \\ I_x I_y u + I_y^2 v &= \alpha^2 \nabla^2 v - I_y I_t \end{aligned} \tag{6}$$

The Laplacian $\nabla^2 u$ and $\nabla^2 v$ in equation (6) are approximated by [5]:

$$\begin{aligned} \nabla^2 u &= 3(\overline{u} - u), \\ \nabla^2 v &= 3(\overline{v} - v), \end{aligned} \tag{7}$$

where $\overline{u}$ and $\overline{v}$ are local average values between the pixels:

$$\begin{aligned} \overline{u}_{i,j,k} &= \frac{1}{6}\{u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k}\} \\ &+ \frac{1}{12}\{u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k}\}, \\ \overline{v}_{i,j,k} &= \frac{1}{6}\{v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k}\} \\ &+ \frac{1}{12}\{v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k}\}. \end{aligned} \tag{8}$$

By replacing expression (7) in equation (6), we can obtain the system

$$\begin{aligned}(3\alpha^2+I_x^2)u+I_xI_yv &= 3\alpha^2\overline{u}-I_xI_t,\\ I_xI_yu+(3\alpha^2+I_y^2)v &= 3\alpha^2\overline{v}-I_yI_t.\end{aligned} \tag{9}$$

whose solution is given by

$$\begin{aligned}u &= \frac{(3\alpha^2+I_y^2)\overline{u}-I_xI_y\overline{v}-I_xI_t}{3\alpha^2+I_x^2+I_y^2},\\ v &= \frac{(3\alpha^2+I_x^2)\overline{v}-I_xI_y\overline{u}-I_yI_t}{3\alpha^2+I_x^2+I_y^2}.\end{aligned} \tag{10}$$

The system (10) can be iteratively solved through the scheme:

$$\begin{aligned}u^{k+1} &= \overline{u}^k-\frac{I_x\left(I_x\overline{u}^k+I_y\overline{v}^k+I_t\right)}{3\alpha^2+I_x^2+I_y^2},\\ v^{k+1} &= \overline{v}^k-\frac{I_y\left(I_x\overline{u}^k+I_y\overline{v}^k+I_t\right)}{3\alpha^2+I_x^2+I_y^2},\end{aligned} \tag{11}$$

where k indicates the current iteration of the algorithm, and the partial derivatives I_x, I_y and I_t are computed by a forward finite difference method [5]:

$$\begin{aligned}I_x &= 0.25(I_{i,j+1,k}-I_{i,j,k}+I_{i+1,j+1,k}-I_{i+1,j,k}\\ &+ I_{i,j+1,k+1}-I_{i,j,k+1}+I_{i+1,j+1,k+1}-I_{i+1,j,k+1}),\\ I_y &= 0.25(I_{i+1,j,k}-I_{i,j,k}+I_{i+1,j+1,k}-I_{i,j+1,k}\\ &+ I_{i+1,j,k+1}-I_{i,j,k+1}+I_{i+1,j+1,k+1}-I_{i,j+1,k+1}),\\ I_t &= 0.25(I_{i,j,k+1}-I_{i,j,k}+I_{i+1,j,k+1}-I_{i+1,j,k}\\ &+ I_{i,j+1,k+1}-I_{i,j+1,k}+I_{i+1,j+1,k+1}-I_{i+1,j+1,k}).\end{aligned} \tag{12}$$

III. Modified Horn and Schunck's algorithm

This section presents our method which is a modified Horn and Schunck's algorithm. We propose to compute the velocity $\mathbf{v}=(u,v)$ for each pixel of the image by minimizing the functional:

$$\mathcal{F}=\int_\Omega\left[\alpha^2G_s^2+E_b^2\right]d\Omega, \tag{13}$$

where E_b^2 is given in expression (4) and:

$$G_s^2=\parallel\nabla^s\mathbf{v}\parallel_2^2=\left\|\frac{(\nabla\mathbf{v})+(\nabla\mathbf{v})^T}{2}\right\|_2^2, \tag{14}$$

The expression $\nabla^s\mathbf{v}=\frac{(\nabla\mathbf{v})+(\nabla\mathbf{v})^T}{2}$, is called symmetric gradient. By minimizing the functional (13), we obtain the Euler equations:

$$\begin{aligned}I_x^2u+I_xI_yv &= \alpha^2\left[\nabla^2u-\frac{1}{2}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)\right]-I_xI_t,\\ I_xI_yu+I_y^2v &= \alpha^2\left[\nabla^2v+\frac{1}{2}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)\right]-I_yI_t.\end{aligned} \tag{15}$$

Before considering the numerical issues, we shall discuss some aspects of the proposed method. Firstly, G_s^2 remains invariant when replacing $\mathbf{v}$ by $\mathbf{w}=\mathbf{v}+(\beta y+C_1,-\beta x+C_2)$ because $\nabla^s\mathbf{v}=\nabla^s\mathbf{w}$. Also, a simple algebra shows that $\parallel\nabla^s\mathbf{v}\parallel_2^2=\parallel\nabla\mathbf{v}\parallel_2^2-0.5(u_y-v_x)^2$. Moreover, $(u_y-v_x)^2=\|\nabla\times\mathbf{v}\|_2^2$, where $\nabla\times\mathbf{v}$ is an alternative way to denote curl($\mathbf{v}$). Therefore, G_s^2 is also rotation invariant regularizer [8] because if we replace $\mathbf{v}$ by $\mathbf{w}=\mathbf{A}\mathbf{v}$, with $\mathbf{A}$ a rotation (orthogonal) matrix then we can show that:
$\|\nabla^s\mathbf{w}\|_2^2=\|\nabla\mathbf{w}\|_2^2-0.5\|\nabla\times(A\mathbf{v})\|_2^2=\|\nabla\mathbf{v}\|_2^2-0.5\|\nabla\times\mathbf{v}\|_2^2$.

A. Numerical Scheme

Substituting (7) in (15), we obtain the system:

$$\begin{aligned}u(3\alpha^2+I_x^2)+v(I_xI_y) &= 3\alpha^2\overline{u}-\frac{\alpha^2}{2}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)-I_xI_t\\ u(I_xI_y)+v(3\alpha^2+I_y^2) &= 3\alpha^2\overline{v}+\frac{\alpha^2}{2}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)-I_yI_t\end{aligned} \tag{16}$$

The cross-partial derivatives are approximated by:

$$\begin{aligned}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right) &\simeq -2(\Phi_u+u)\\ \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right) &\simeq 2(\Phi_v+v)\end{aligned} \tag{17}$$

where the difference operators Φ_u and Φ_v are given by:

$$\begin{aligned}&\Phi_u=-\frac{1}{2}(u_{i,j+1,k}+u_{i,j-1,k})\\ +\ &\frac{1}{8}(v_{i+1,j+1,k}-v_{i-1,j+1,k}-v_{i+1,j-1,k}+v_{i-1,j-1,k}),\\ &\Phi_v=-\frac{1}{2}(v_{i+1,j,k}+v_{i-1,j,k})\\ +\ &\frac{1}{8}(u_{i+1,j+1,k}-u_{i-1,j+1,k}-u_{i+1,j-1,k}+u_{i-1,j-1,k}).\end{aligned} \tag{18}$$

Using equation (17) we can rewrite the system (16) as:

$$\begin{aligned}u(2\alpha^2+I_x^2)+v(I_xI_y) &= 3\alpha^2\overline{u}+\alpha^2\Phi_u-I_xI_t,\\ u(I_xI_y)+v(2\alpha^2+I_y^2) &= 3\alpha^2\overline{v}+\alpha^2\Phi_v-I_yI_t.\end{aligned} \tag{19}$$

The solution of the system (19) is given by:

$$\begin{aligned}u &= \frac{(3\overline{u}+\Phi_u)(I_y^2+2\alpha^2)-(3\overline{v}+\Phi_v)I_xI_y-2I_xI_t}{4\alpha^2+2I_x^2+2I_y^2},\\ v &= \frac{(3\overline{v}+\Phi_v)(I_x^2+2\alpha^2)-(3\overline{u}+\Phi_u)I_xI_y-2I_yI_t}{4\alpha^2+2I_x^2+2I_y^2},\end{aligned} \tag{20}$$

which can be iteratively solved through the scheme:

$$\begin{aligned}u^{k+1} &= \frac{(3\overline{u}^k+\Phi_u^k)(I_y^2+2\alpha^2)-(3\overline{v}^k+\Phi_v^k)I_xI_y-2I_xI_t}{4\alpha^2+2I_x^2+2I_y^2},\\ v^{k+1} &= \frac{(3\overline{v}^k+\Phi_v^k)(I_x^2+2\alpha^2)-(3\overline{u}^k+\Phi_u^k)I_xI_y-2I_yI_t}{4\alpha^2+2I_x^2+2I_y^2},\end{aligned} \tag{21}$$

where k is the current iteration of the numerical procedure; $\overline{u}^k$ and $\overline{v}^k$ are estimated by equation (8); I_x, I_y and I_t are computed by (12); Φ_u^k and Φ_v^k are calculated through expression (18).

IV. Experimental Results

In order to demonstrate the efficiency and robustness of the proposed algorithm, three numerical experiments will be carried out and discussed. We compare the performance of our 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:

$$\theta = \cos^{-1}(\mathbf{v} \cdot \hat{\mathbf{v}}),$$

where $\mathbf{v}$ is the correct motion vector and $\hat{\mathbf{v}}$ is the estimated OF vector. The second error metrics is the mean-squared-error (MSE) defined by:

$$\text{MSE} = \frac{1}{2(N \times M)} \sum_{i=1}^{N} \sum_{j=1}^{M} \|\mathbf{v}_{i,j} - \hat{\mathbf{v}}_{i,j}\|^2, \tag{22}$$

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

For the test cases, we stop the iterative scheme when the difference between the values of the functional (equations (3) or (13)), calculated on two consecutive iterations, is less than 10^{-3}.

- **Case 1** – Translation. A texture is moving with the constant speed $\mathbf{v} = (1, 1)$ pixel/frame. Figure 1a presents the first frame of this case, which has 80×80 pixels. Considering $\alpha = 0.4$ (regularization parameter), the OF estimated by both the original algorithm of Horn and Schunck and the proposed model are pictured in Figures 3 and 4, respectively. As one can see from these figures, the algorithms provided reasonable results when compared with the real OF (see Figure 2). In these figures, it can also be observed that the modified algorithm have a lower diffusion velocity field (it preserves the discontinuities), although some vectors are more distorted near the edge of the texture.

  Tables I and II show the convergence rate and errors of the algorithms. It can be seen that the results of our model are satisfactory when compared with those of the original algorithm.

α	iteration number	$E(\theta)$	MSE
0.05	17	1.5733	0.0268
0.10	27	1.5728	0.0303
0.20	54	1.5720	0.0426
0.40	126	1.5714	0.0648
0.80	245	1.5710	0.0718

TABLE I

Case 1 – Performance of Horn and Schunck's algorithm.

α	iteration number	$E(\theta)$	MSE
0.05	18	1.5729	0.0292
0.10	25	1.5725	0.0310
0.20	32	1.5720	0.0333
0.40	50	1.5715	0.0380
0.80	212	1.5710	0.0773

TABLE II

Case 1 – Performance of the algorithm proposed in this paper.

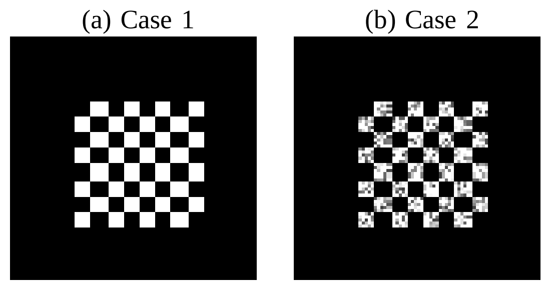

Fig. 1. First frames of the synthetic sequence in the cases 1 and 2.

- **Case 2** – Adding noise. We add multiplicative noise to the image I in Figure 1a, using the equation $J = I + nI$, where n is uniformly distributed random noise with mean 0 and variance 0.2. Figure 1b shows the first frame of this case. Figures 5 and 6 present the OF computed by the original algorithm of Horn and Schunck and model proposed, respectively, for $\alpha = 0.4$. From these figures, it can be noted that the modified algorithm have a lower diffusion velocity field as in case 1, although some vectors are more distorted near the edge of the texture. It can also be seen from these figures that both algorithms properly compute the OF when compared with the real OF (see Figure 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 proposed algorithm was superior to the one observed by the original method.

α	iteration number	$E(\theta)$	MSE
0.05	16	1.5758	0.0427
0.10	26	1.5705	0.0375
0.20	53	1.5746	0.0429
0.40	116	1.5711	0.0548
0.80	235	1.5706	0.0621

TABLE III

Case 2 – Performance of Horn and Schunck's algorithm.

α	iteration number	$E(\theta)$	MSE
0.05	16	1.5663	0.0530
0.10	20	1.5738	0.0406
0.20	28	1.5728	0.0395
0.40	52	1.5710	0.0378
0.80	198	1.5681	0.0632

TABLE IV

Case 2 – Performance of our algorithm.

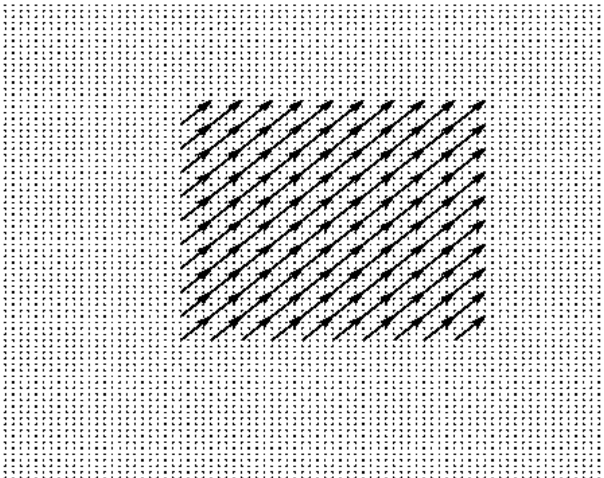

Fig. 2. Real OF for the cases 1 and 2.

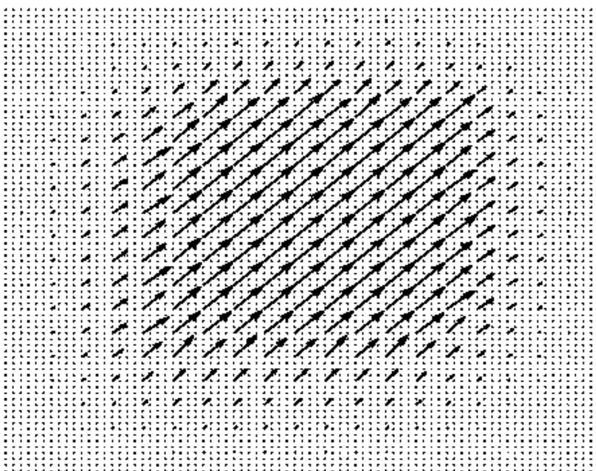

Fig. 3. OF estimated in the case 1 by the original Horn and Schunck algorithm.

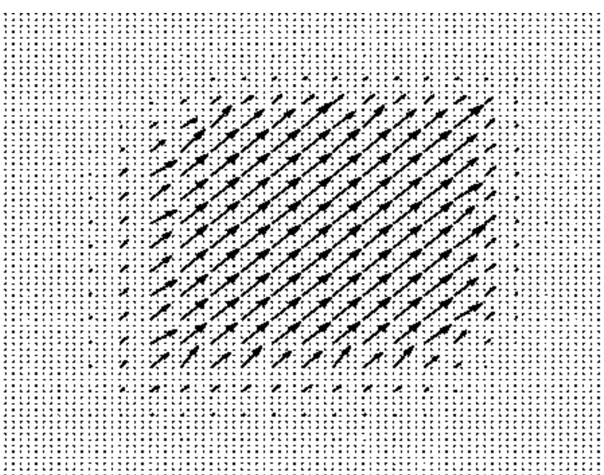

Fig. 4. OF estimated in the case 1 by the algorithm proposed in this paper.

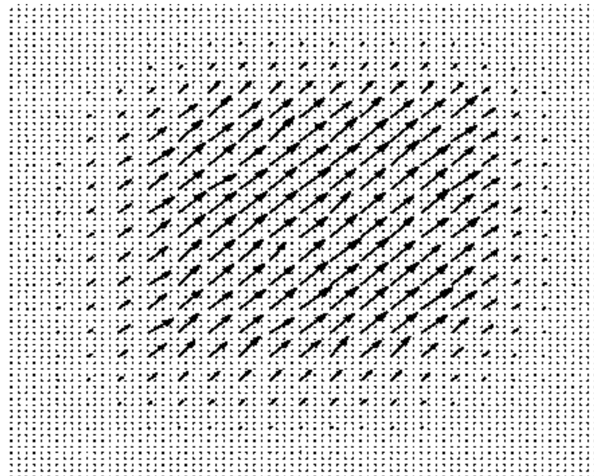

Fig. 5. OF computed in the case 2 by the original Horn and Schunck algorithm.

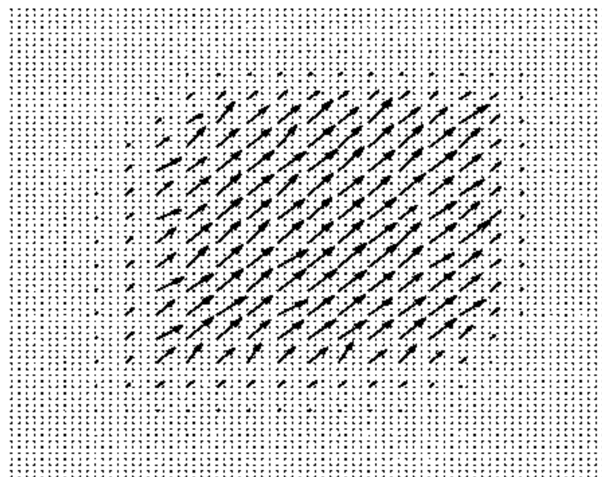

Fig. 6. OF computed in the case 2 by our algorithm.

- **Case 3** – Intravascular Ultrasound (IVUS). Figure 7 shows the IVUS image which is rotated with angular velocity of -1 degree/frame. It is an interesting test case, because IVUS image sequences have much noise and non-trivial patterns.
  In this case, images of 186×192 pixels are used. Tables V and VI show that the performance of the OF algorithms is quite similar, indicating that these techniques are of the same order.
  The OF determined by the algorithm of Horn and Schunck and model proposed is presented in Figures 9 and 10, respectively, for $\alpha = 0.4$. It is clear from these figures that the algorithms yield practically analogous results and such OF solutions agree with the real OF (see Figure 8).

α	iteration number	$E(\theta)$	MSE
0.05	28	0.6544	0.2766
0.10	37	0.6500	0.2653
0.20	54	0.6436	0.2502
0.40	82	0.6375	0.2333
0.80	126	0.6307	0.2186

TABLE V
CASE 3 – PERFORMANCE OF HORN AND SCHUNCK'S ALGORITHM.

α	iteration number	$E(\theta)$	MSE
0.05	28	0.6574	0.2845
0.10	36	0.6533	0.2756
0.20	52	0.6473	0.2633
0.40	78	0.6404	0.2493
0.80	120	0.6327	0.2353

TABLE VI
CASE 3 – PERFORMANCE OF THE OUR ALGORITHM.

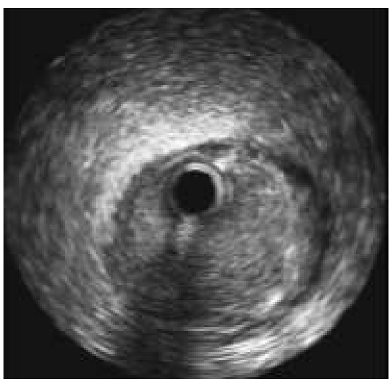

Fig. 7. First frame of the synthetic sequence in the case 3.

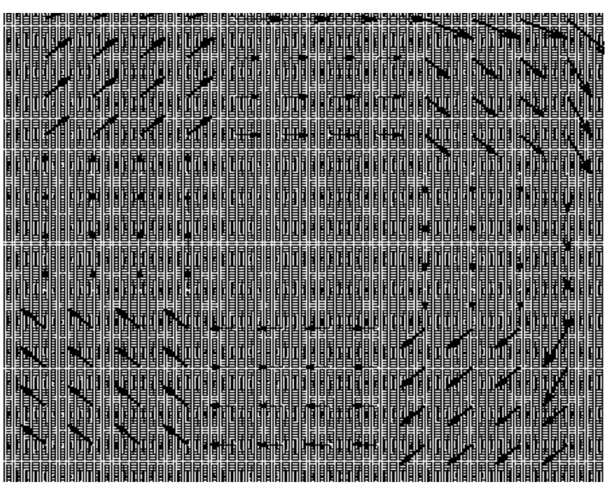

Fig. 8. Real OF for the case 3.

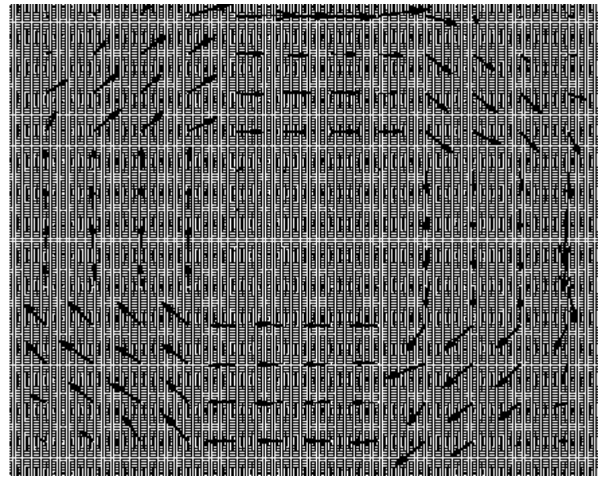

Fig. 9. OF estimated in the case 3 by the original Horn and Schunck algorithm.

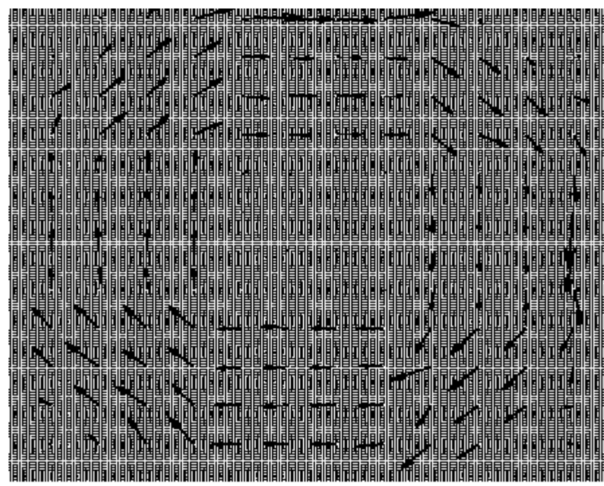

Fig. 10. OF estimated in the case 3 by our algorithm.

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 performed experiments the precision of our model was practically similar to the original algorithm. Moreover, the numerical experiments indicate that the modified Horn and Schunck's algorithm has convergence rate quite satisfactory when compared to the original model. Further directions for this work are to explore connections with diffusion filters and the effects of the invariance $\mathbf{v} \to \mathbf{v} + (\beta y + C_1, -\beta x + C_2)$, observed in section III, for real world image sequences [8].

ACKNOWLEDGEMENT

We gratefully acknowledge the support provided by CNPq (Grant 141474/2009-2).

References

[1] F. Bartolini and 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, 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, 12, 43-77, 1994.

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

[4] M.A. Gutierrez, S.S. Furuie, M.S., Rebelo, 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, ch02, 2005.

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

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

[7] T.A. Poggio, V. Torre, and C. Koch, Computational vision and regularization theory, Nature, 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, 45, 245-264, 2001.