Asian Journal of Information Technology 5 (9): 968-975, 2006

Fast Global Motion Estimation for Image Sequences

Z.E. Baarir and F. Charif
Lesia Laboratory of Research, Electronic Department, University of Biskra, Algeria

Abstract: Optical flow estimation is important in the area of computer vision. This study presents a fast modified Horn and Schunck study for robust boundary preserving estimation of optical flow. Variational approaches have addressed this topic and proposed methods that account for velocity boundaries at the cost of significant computational complexity, which makes them inadequate for current real-time applications. The proposed method is derived from the benchmark algorithm of Horn and Schunck and Simoncelli's matched-pair 5 tap filters, such that it produces robust, fast and exact detection of motion boundaries and it is very simple to implement. Experimental results using synthetic and real optical flow image sequences are presented to demonstrate the effectiveness of our method in comparison to selected methods.

Key words: Horn Schunck algorithm, optical flow estimation, simoncelli's filters, synthetic, real image sequences

INTRODUCTION

Optical Flow (OF) is the displacement of each image pixel in an image sequence. Image motion estimation is a fundamental issue in low-level vision and is used in many applications such as robot navigation, object tracking, image coding, image segmentation and motion compensation. Recently, it is also used in medical imagery in measuring blood flow and heart-wall motion and in the measurement of minute amounts of growth in corn seedlings. A great number of approaches for OF estimation have been proposed in the literature, including differential, correlation-based, energy-based and phase-based methods[1]. The benchmark of differential methods is that of Horn and Schunck[2]. It is simple, it can be implemented in real time and it is a fast method that produces good estimates except at motion boundaries. Several approaches have addressed this topic and proposed methods that account for velocity boundaries at the cost of significant computational complexity, which makes them inadequate for current real-time applications[3].

Another major concern is the approximation errors that occur when the gradient-based approach is adopted. These errors are due to inaccurate numerical approximation of partial derivatives, as well as temporal and spatial aliasing during sampling of the image brightness function. In order to solve the above-mentioned error problem, Barron[1] suggested a spatiotemporal pre-smoothing with a spatiotemporal Gaussian filter with standard deviation of 1.5 to the target image are performed, first. Then, a four-point central differences for differentiation (with mask coefficients 1/12(-1,8,0,-8,1))(MHS).The total number of images required is determined by the size of the derivative kernel and the level of smoothing , given by the standard deviation of the Gaussian, this required a total of 15 frames to compute flow. In this study we propose two possible ways to solve these problems:

- The first is to adapt the Horn and Schunck algorithm so that it produces robust boundary preserving estimates, while retaining its simplicity and speed of execution. We do so by generalizing the smoothing filter in the Horn and Schunck algorithm so as to vary with position and conform to local variations of velocity[3-6].
- The second is to estimate the spatial and temporal derivatives of images by Simoncelli's matched-pair 5 tap filters after smoothing by a simple averaging kernel (1/4,1/2,1/4)[6,7].

Horn and schunck formulation: In our study, the Horn and schunk's differential method has been employed[2], which is mainly based on optimising the energy function that is a function of an image constraint and a smoothness constraint:

$$\iint\left[\left(E_t + uE_x + vE_y\right)^2 + \lambda^2(u_x^2 + u_y^2 + v_x^2 + v_y^2)\right]dx.dy \qquad (1)$$

Where: E = E (x,y,t) denotes image brightness function at time t, (u,v) designates optical velocity ,(u_x ,u_y) its spatial derivatives, (E_x ,E_y ,E_t) the spatiotemporal image brightness derivatives and λ is the weighting parameter between the two constraints. The choice of a fixed value

Corresponding Author: Z.E. Baarir, Lesia Laboratory of Research, Electronic Department, University of Biskra, Algeria

for this parameter remains to be a very crucial problem in such motion estimation algorithms. Iterative equations are used to minimize (1) and obtain image velocity at each image location:

$$u_i^{n+1} = \bar{u}_i^n - \alpha E_x$$
$$v_i^{n+1} = \bar{v}_i^n - \alpha E_y \quad (2)$$

Where:

$$\alpha = (E_x \bar{u}_i^n + E_y \bar{v}_i^n + E_t)/(\lambda^2 + E_x^2 + E_y^2)$$

Where i denotes image position, u_i^o and v_i^o denote initial velocity estimates at i (typically zero), n is the iteration number and $(\bar{ui}, \bar{vi})$ is the neighbourhood averages of (u_i, v_i)at I. The Horn and Schunck technique is widely used to estimate OF , but it also takes a considerable number of iterations to converge .The modified Horn and Schunck algorithm presented in this study adds one step to the classical algorithm: a momentum term which considers past velocity is added to (2):

$$u_i^{n+1} = \bar{u}_i^n - \alpha E_x + \mu.(u_i^n - u_i^{n-1})$$
$$v_i^{n+1} = \bar{v}_i^n - \alpha E_y + \mu.(v_i^n - v_i^{n-1}) \quad (3)$$

The range of μ is between 0 and 1.

Differentiation: Horn and Schunck differential technique compute velocity from Spatio-temporal derivatives of Image intensity or filtered versions of the image and the performance of this approach weighs heavily on the choice of temporal filter[8,1,9-11,4-6,3,2,7]. In this study we present tow techniques to estimate the spatial and temporal derivatives of images.

Horn and schunck.'s derivatives (Hs): The original method of Horn and Schuck (HS) described in[1,2] use just 2 images (E(t) and E(t+1)) to estimate intensity derivatives Precisely, we use the following masks:

$$M_x = \frac{1}{4}\begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix} M_y = \frac{1}{4}\begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} M_t = \frac{1}{4}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

And then the partial derivatives are calculated by:

$$E_x = M_x * (E(t) + E(t+1))$$
$$E_y = M_y * (E(t) + E(t+1))$$
$$E_t = M_t * (E(t+1) - E(t)) \quad (4)$$

Simoncelli.'s derivatives (SHS): In this study we estimate the spatial and temporal derivatives of images by Simoncelli's matched-pair 5 tap filters after smoothing by a simple averaging kernel (1/4,1/2,1/4) with superior accuracy to the Gaussian 1.5 filter and requires 7 frames[9]. Simoncelli balanced/matched filters[7] combine presmoothing (p_5) and differentiation (d_5) ;The prefiltering kernel's coefficients were (0.036, 0.249, 0.431, 0.249, 0.036), while the differential kernel's coefficients were (-0.108,-0.283, 0.0, 0.283, 0.108). For example, Ey is computed by applying p_5 in the t dimension , then p_5 to those results in the x dimension and finally d_5 to those results in the y dimension .E_x and E_t are computed in a similar manner.

Estimate neighborhood average: The estimate neighborhood average $(\bar{ui}, \bar{vi})$ at i is computed according to a smoothing filter (ψ) adapted to local variations of velocity . It will be used in iteration (3), as[4-6,3]:

$$\bar{u}_i^n = \Psi(\{u_j^n\} : j \in \eta_i)$$
$$\bar{v}_i^n = \Psi(\{v_j^n\} : j \in \eta_i) \quad (5)$$

Where η_i is the set of neighbors of i. The performance of this study weighs heavily on the choice of smoothing filters ψ. We explore therefore several smoothing filters.

Intensity-weighted average: In this study, $\psi = \psi_I$ produces Intensity-weighted average and it can be computed as follows:

$$\Psi_I = \sum_{j \in \eta_i} \frac{\frac{1}{1+|E_j - E_i|}}{\sum_{j=1}^{\delta} \frac{1}{1+|E_j - E_i|}} u_j \quad (6)$$

This is an isotropic diffusion in all digital directions allowed by the form of η^i, eight directions in the case of an 8-neighborhood[3]. A similar formula is used for v.

Velocity-weighted average: In this study, $\psi = \psi_V$ produces velocity-weighted average and it can be computed as follows:

$$\Psi_V = \sum_{j \in \eta_i} \frac{(\frac{1}{1+|u_j - u_i|})^{\beta}}{\sum_{j=1}^{\delta} (\frac{1}{1+|u_j - u_i|})^{\beta}} u_j \quad (7)$$

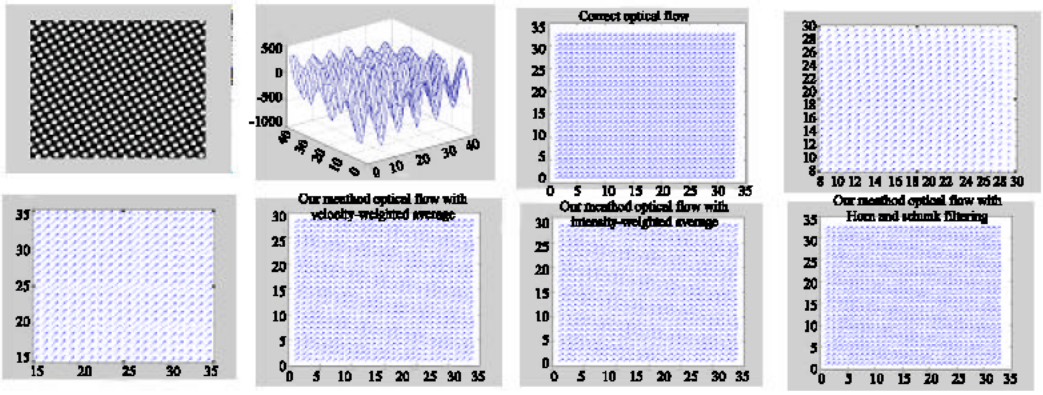

Fig. 1: Optical flow results of different techniques on the sinusoidal 1 sequence. From upper left to lower right, frame 1in the sequence, correlation between two frames (1 and 2), the correct optical flow , HS optical flow, MHS optical flow, Our method optical flow with Ψ_V, Our method optical flow with Ψ_I, Our method optical flow with Ψ_H

Where $\beta > 1$ is introduced to account for the possibly small range of velocity values[3]. A similar formula is used for v.

Horn andschunk's filtering: In this subsection, $\psi = \psi_H$ is Horn and schunk's filtering and it can be computed according to the mask:

$$M_H = \begin{pmatrix} 1/12 & 1/6 & 1/12 \\ 1/6 & 0 & 1/6 \\ 1/12 & 1/6 & 1/12 \end{pmatrix}$$

And then the estimate neighborhood average is calculated by:

$$\Psi_H = u * M_H \qquad (8)$$

A similar formula is used for v.

Proposed hornandschunck algorithm: The proposed algorithm is outlined as follow:

```
Begin
For each pixel (i,j) do
-calculate the values Ex(i,j),Ey(i,j)and Et(i,j) using (HS) or (MHS)
 or (SHS) derivatives.
-initialize the values u (i,j) and v(i,j) to zero
  End {for}
    -choose a suitable weighting value λ and μ.
     While not converged studydo
       For each pixel (i,j)  studydo
      -compute ū_i, v̄_i  (withψ_I or ψ_V or ψ^H).
      - update u (i,j), v(i,j) using (3)
   End {for}
 End {while}
End
```

Experimental results: This section shows some of the results obtained using the method described in this study. Comparisons with some of the commonly cited techniques are also made (Horn and Schunck, Barron et al (MHS), Uras *et al*. Anandan, Lucas and Kanade and Nagel[1]). A variety of images have been tested, including synthetic image and different types of real images.

Synthetic image sequence: To test the algorithms, we computed optical flows on synthetic image sequence with the true displacements known at every pixel location. We measured the average differences between correct velocity $\vec{v}_c$ and an estimate $\vec{v}_e$ using the angular distance introduced by Barron and al with 100% density[1]:

$$\Phi_E = \arccos(\vec{v}_c . \vec{v}_e) \qquad (9)$$

The average errors and standard deviations of Φ_E were calculated by neglecting a 20-pixel-wide boundary. Figure 1 shows the results of different techniques on the image Sinusoidal sequence with velocity V = (1.585,0.863) pixels/frame. The first two images in the top row are frame 1in the sequence and the correlation between two frames (1 and 2) from the Sinusoidal1 sequence. The third Image in the top row is the correct optical flow field. From Fig. 2, we can see that it is difficult to choose a fixed value of $0.1 \le \lambda \le 0.4$ for all the sequences. However, a range of λ seems to yield to best compromise for the tests that were performed on test image. The optimal λ value of the smoothness constraint was set to 0.19 for all the experiments conducted in this study.

The modified algorithm was run several times with the momentum parameter μ set to different values. The algorithm was considered to have converged when the average error became less than 0.1°. These results are plotted in Fig. 3, it is seen that for a momentum equal to

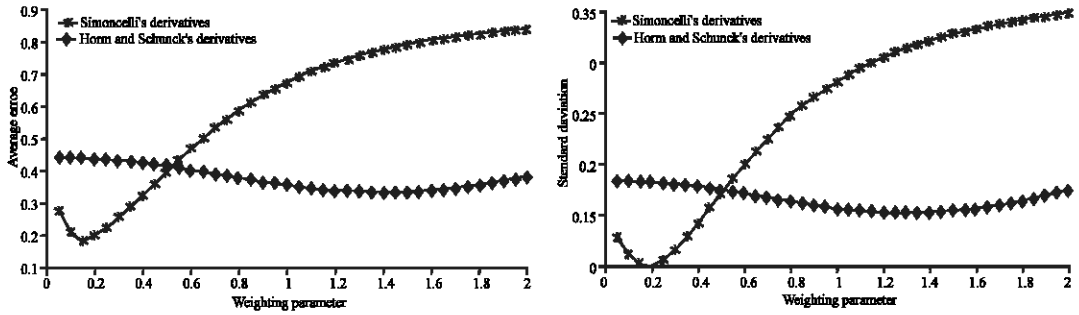

Fig. 2: Sinusoidal1: Impact of various filter and different values of λ (the smoothness constraint) on motion estimation with Ψ_V: (a) average error, (b) standard deviation

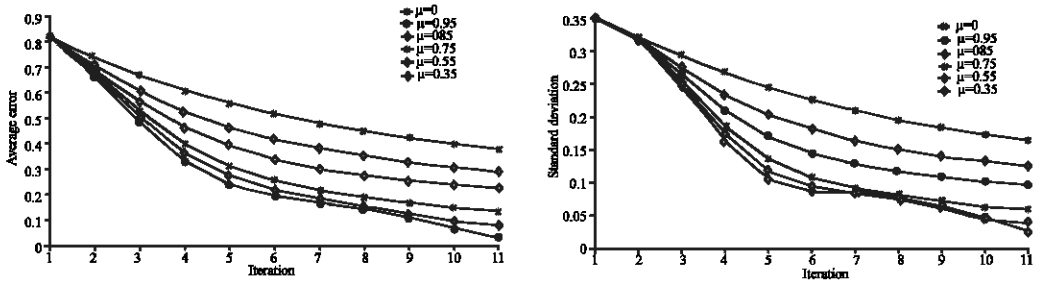

Fig. 3: Sinusoidal1: Impact of various momentum parameter with Ψ_V: (a) average error, (b) standard deviation

Table 1: Summary of sinusoid 2D velocity results

Technique	Average error	Standard deviation	Frames used
HS [2]	4.19°	0.50°	2
MHS [2]	2.55°	0.59°	15
Uras et al[2]	2.59°	0.71°	15
Anandan [2]	30.80°	5.45°	2
Lucas and kanade[2]	2.74°	0.16°	15
Nagel[2]	2.55°	0.93°	15
SHS with Ψ_I	0.0204°	0.0160°	7
SHS with Ψ_H	0.0208°	0.0178°	7
SHS with Ψ_V	0.0197°	0.0168°	7

0 and 0.35 the algorithm had not converged after more than 300 iterations. For larger momentums such as 0.95and 0.85, the algorithm was performed better and convergence was reached in a smaller number of iterations (11 iterations). When the momentum was decreased to 0.75 and 0.55 , the behavior became more and more volatile and the algorithm took more iterations to converge (30 iterations). Hence, it became clear for this particular example that the optimal momentum must be in the vicinity of 0.9. The average errors ,standard deviations and the number of image frames used of the computed flows for above mentioned methods applied to Sinusoidal1 are shown in Table 1, which are generally very good , but our method give smaller e rrors and is performed better than the original Horn and Schunck method (HS))) and the modified (HS) method by Barron and *al* (MHS). So that, the velocity-weighted average is the best technique for all the tree methods mentioned in this study, as shown in Fig. 4 (histograms of errors).

The three techniques (Ψ_I, Ψ_E and Ψ_H) have nearly the same convergence; however the velocity-weighed average Ψ_V is more convergent than the other two , as shown in Fig. 5. Furthermore, our method offers a faster convergence speed, typically around 10-30 iterations are required, but the number of iterations for Horn and Schunck's approach (HS)[1]and Barron and al.'s approach (MHS)[1] were both set to 500.All the programs apart from the author's were obtained from the ftp site at ftp://csd.uwo.ca/pub/vision. Our algorithms are implemented using MATLAB.7.

Real image sequences: Three real image sequences have also been tested and good results have been obtained. Figures 6-8 show the experimental results of the flow fields produced by our method on the three real image sequences: Hamburg Taxi, Miss America and SRI sequence [2.10]. Fig 6, 7 and 8a show three static frames grabbed from three standard real image sequences, Fig. 6a, shows the six frame of the 'Miss

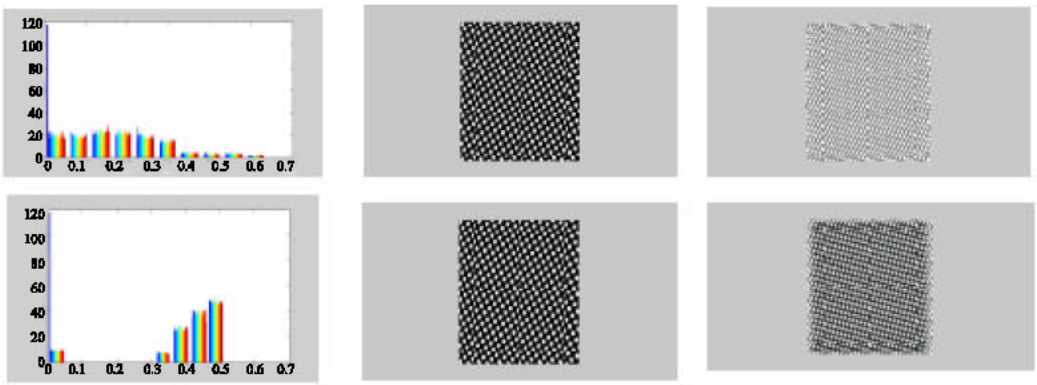

Fig. 4: Sinusoidal1: From upper left to lower right, error histograms for our method with Ψ_V and μ =0.95, interpolated, error image, Horn and Schunck algorithm with μ =0, interpolated and error image

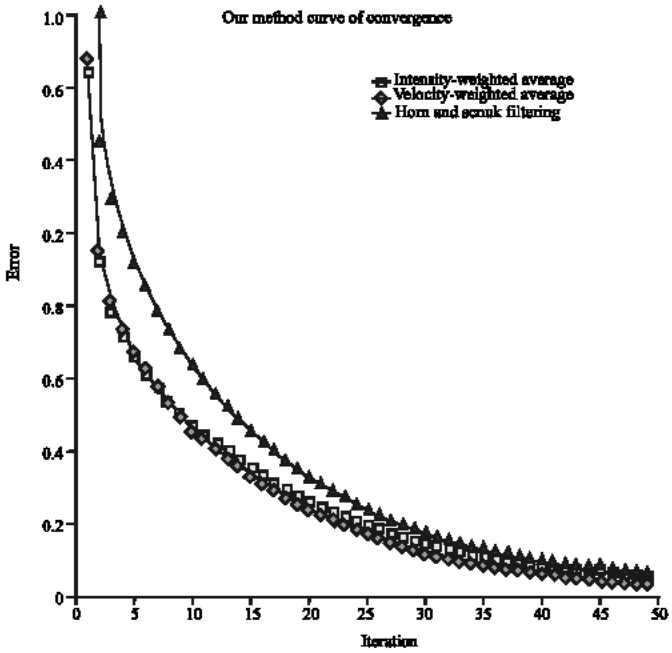

Fig. 5: Our method curves of convergence

America' image sequence, Fig .7a shows a frame of the SRI image sequence (in this sequence , the camera translates parallel to the ground plane. This sequence is very challenging because of poor resolution, considerable occlusion and low contrast. The maximum velocity is approximately 2 pixel/frame)and Fig. 8a shows a frame of the Hamburg Taxi image sequence (this sequence contains three principal moving objects : •)the taxi turning around the corner(1 pixel/frame); •) a car in the lower left, driving from right (3 pixels/frame); •) a van in lower right driving right to left(3 pixels/frame)) .Since it is very difficult to determine the true of for real images .Usually only qualitative testing is performed. The validity of motion estimates for these real sequences can be verified by using these estimates in a motion- compensated interpolation .For this task, a spline interpolation is used. The impact of the proposed method using various filters on motion-compensated prediction and prediction error is shown in Fig. 6. From this Figure, we can see that our method-according to the Bayesian criteria method[12]-is more accurate and precise; furthermore, it preserves velocity boundaries. Note Bayesian criteria method is 8.5617d Bpenalty in the peak prediction error as compared with our method (our method with different filters (Ψ_I, Ψ_E and Ψ^H) is virtually identical). Figure 7 (b-d) and Fig. 8b show the computed flow fields

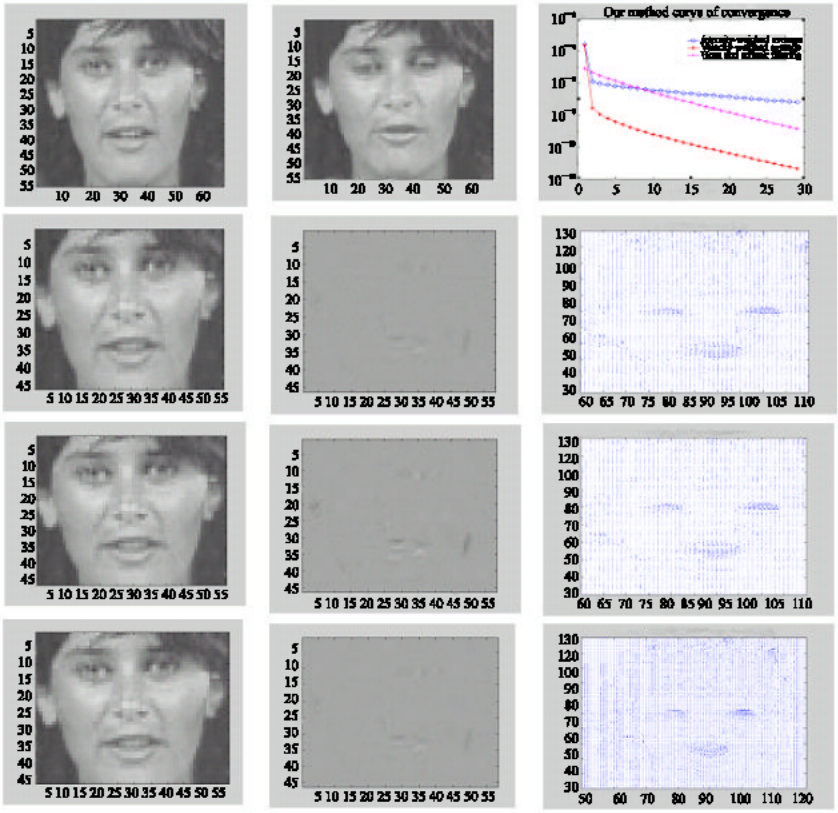

Fig. 6: Comparison of our method: From upper left to lower right, frame 6 and 8 of Miss America, convergence curve, interpolated and Error image as well as optical flow using Ψ_h, Ψ_i, Ψ_v, respectively

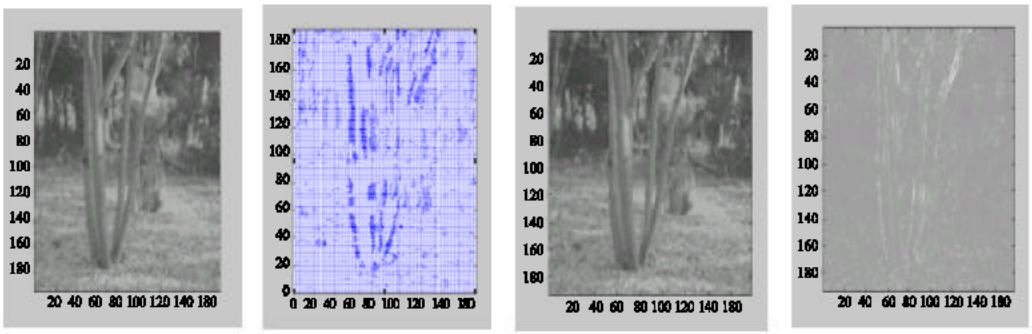

Fig. 7: Our method with velocity-based adaptive average: From upper left to lower right, frame 13 of SRI' optical flow, interpolated and error image

corresponding to Fig. 7 and 8a, respectively. The Fig. 8b-d shows the preservation of motion boundaries by Ψ_V-because it has the best performance-for the Hamburg Taxi sequence.The motion boundaries are well delineated as shown in the zoom in at the taxi ,car and a van. From this Figures, we can see that three moving objects were clearly captured and located by our method with filter Ψ_V ;moreover , our method -according to HS and MHS[1]-is more accurate and precise.

Medical image sequences: The results of two medical image sequences are presented in Fig. 9 and 10, respectively. Flow of the image frames is presented for each Image sequence.The overall results obtained from our algorithm are very interesting and reasonable.

Fig. 8: Zoom in at the car, taxi and a van: Our method with velocity-based adaptive

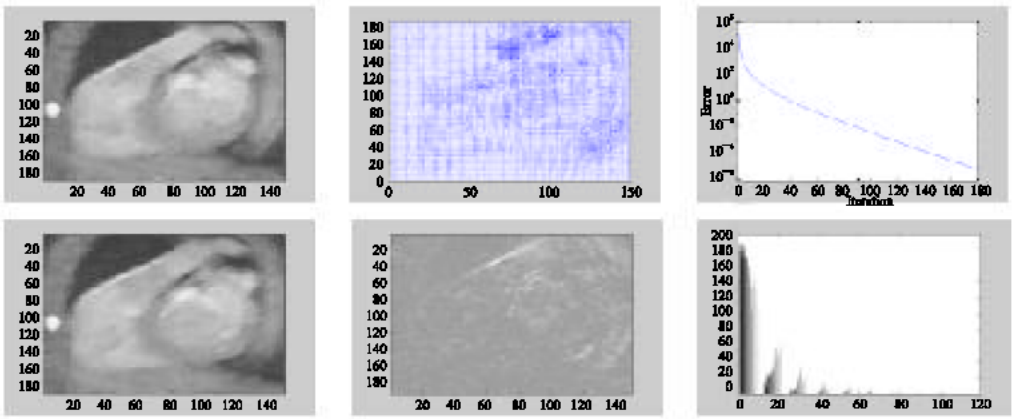

Fig. 9: Our method with velocity-based adaptive average: From upper left to lower right, frame 6 of medical sequence N°: 1, optical flow, convergence curve, interpolated, error image and error histogram

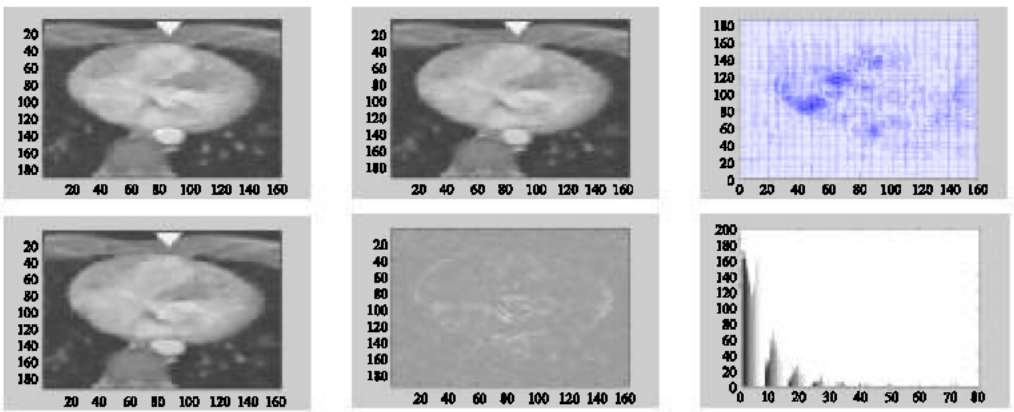

Fig. 10. Our method with velocity-based adaptive average : From upper left to lower right, frame 3 of medical sequence N°: 2, frame 4 of medical sequence N°: 2, optical flow, interpolated, error image and error histogram

CONCLUSION

In this study we have developed a fast and reliable image motion estimation method using Horn and Schunck algorithm and Simoncelli's matched-pair 5 tap filters and we have formulated and experimented with different filters (Ψ_I, Ψ_E and Ψ^H) for fast boundary preserving estimation of optical flow using synthetic or calibrated image sequence and different types of real image sequences. The modified method (SHS) lacks the capability of the original Horn and Schunck method to fill in motion estimation where gradient information is poor. On the other hand, the modified method produces better velocity estimates at the motion edge. Compared with the original Horn andSchunck method, the modified Horn and Schunck method produced a motion velocity signal of higher amplitude. Furthermore, Our method (SHS) with the velocity-weighted average has the best performance when compared to the Horn and Schunck method (HS), the modified (HS) method by Barron and al (MHS) and Bayesian criteria method[1,12]. Our method has some good features; in particular, it is very simple to implement, it is also very fast and has an exact detection of motion boundaries.

ACKNOWLEDGMENTS

The authors would like to thank Professor John Barron from the University Of Western Ontario, Canada, for providing the implementation of Simoncelli's matched-pair 5- tap filters.

REFERENCES

1. Barron, J.L., D.J. Fleet and S.S. Beauchemin, 1994. Performance of optical flow techniques, Intl. J. Computer Vision, pp: 43-77.
2. Horn, B.K.P. and B.G. Schunck, 1981. Determing optical flow, Artificial Intelligence, pp: 185-204.
3. El-Feghali, R. and A. Mitiche, 2000. Fast computation of a boundary preserving estimate of optical flow, BMCV, pp: 1-6.
4. Charif, F. and Z.E. Baarir, 2005. Relations between different methods for fast computation of a boundary preserving estimate of optical flow, Third IEEE International confirence on Signals, Systems and Devices SSD, Tunisia.
5. Charif, F. and Z.E. Baarir, 2005. Robust and fast estimation of optical flow. Conférence sur le Génie Electrique CGE04, EMP, Alger Algeria, pp: 44-55.
6. Baarir, Z.E. and F. Charif, 2005. Relations between different methods for fast motion estimation, SICE Annual Conference, Intl. Conference on Instrumentation, Control and Information Technology, Okayama Japan, pp: 152-159
7. Simoncelli, E.P., 1994. Design of multi-dimensional derivative filters, IEEE Int. Conf. Image Processing, pp: 790-793.
8. Arredondo, M.A. and K.Lebart, D. Lane, 2004. Optical flow using textures, Pattern Recognition Lett., pp: 449-457.
9. Barron, J.L. and M. Khurana, 1997. Determining optical flow for large motions using parametric models in a hierarchical framework, Vision Interface, pp: 47-56.
10. Bruhn, A. and A. Weickert, 2005. Lucas/Kanade Meets Horn/Schunck: Combining local and global optic flow methods, Intl. J. Computer Vision, pp: 211-231.
11. Burgi, P.Y., 2004. Motion estimation based on the direction of intensity gradient, Image and Vision Computing, pp: 637-653.
12. Stiller, C. and J. Konrad, 1999. Estimating motion in image sequences, IEEE Signal Processing Magazine pp: 1-34.