An adapted Lucas-Kanade's method for optical flow estimation in catadioptric images

A. Radgui, Cedric Demonceaux, El Mustapha Mouaddib, D. Aboutajdine, M. Rziza

HAL Id: inria-00325390
https://inria.hal.science/inria-00325390

Submitted on 29 Sep 2008

An adapted Lucas-Kanade's method for optical flow estimation in catadioptric images [*]

A.Radgui[1], C.Demonceaux[2], E.Mouaddib[2], D.Aboutajdine[1], and M.Rziza[1]

[1] GSCM-LRIT, Université Mohammed V-Agdal ,
B.P 1014, Rabat, Maroc
[2] M.I.S. Université de Picardie Jules
Verne, 33, rue Saint Leu 80039 Amiens Cedex 1.

Abstract. The optical flow estimation is one of important problem in computer vision. Differential techniques were used successfully to compute the optical flow in perspective images. Lucas-Kanade is one of the most popular differential method that solve the problem of optical flow by given constrain that motion is locally constant. Even if this method works well for the perspective images, this supposition is less appropriate in the omnidirectional images due to its distortion. In this paper, we propose to use new constraint based on motion model defined for paracatadioptric images. This new constraint will be combined with an adapted neighborhood windows witch are adequate to catadioptric images. We will show in this work that these two hypothesis allows to compute efficiently optical flow from omnidirectional image sequences.

1   Introduction

Optical flow is the projection on image of the 3D scene motion. It is usually called the optical flow field or the image velocity field. Optical flow was used by robotics researchers in many tasks such as : object detection and tracking, movement detection, robot navigation and visual odometry. For traditional cameras, performance analysis of a number of most popular optical flow techniques can be found in [1]. The basic assumption for the optical flow calculation is the conservation of pixel intensity. Different optical flow algorithms solve the problem of optical flow by given additional conditions. The Lucas-Kanade method [2] is still one of the most popular versions of two-frame differential methods for motion estimation. This method supposes the flow locally constant. Even if this method works well for the perspective images, this supposition is less appropriate in the omnidirectional images because of its distortion.
This problem was recently treated in omnidirectional vision. The authors tried to adapt the existing method to distorted catadioptric images. A method to estimate optical flow on the omnidirectional images was also proposed on [3] using wavelet approach basing on a brightness change constraint equation. Gluckman and Nayar [4] adapted the known ego motion estimation technique to a motion

estimation method using spherical projection of the optical flow field. The transformation between spherical projection model and catadioptric camera model represent a mapping to the sphere witch is performed by determining of Jacobian. Daniilidis and al. [5] or Tosic and al [6] proposed to use the spherical equivalence [7] to define a new space of optical flow processing taking account of the sensor geometry.

We propose to use new constraint based on motion model defined for paracatadioptric images and which is appropriate to the geometry of our image formation. This new constraint will be applied in adapted neighborhood windows more adequate to catadioptric images. We will show in this work that these two hypothesis, and compared to Lucas-Kanade method, allows to compute effectively optical flow from omnidirectional image sequences. The paper has the following structure. In section 2, we will present the motion model derived from camera model geometry for paracatadioptric sensors. Section 3 will describe our approach to compute optical flow using motion model and appropriate neighbourhood. In section four the experimental results will be given and comparative measurement will be discussed.

2   Optical flow estimation

Optical flow algorithms estimate the deformations between two images. The basic assumption for the optical flow calculation is that pixel intensity is conserved. It is assumed that the intensity, or color, of the objects has not changed significantly between the two images. Based on this idea, we have the following assumption :

$$I(x, y, t + 1) = I(x + V_x, y + V_y, t), \tag{1}$$

where $\overrightarrow{V} = (V_x, V_y)$ is the vector of velocity. Then, by derivation, we obtain the well known optical flow constraint equation :

$$\overrightarrow{\nabla} I . \overrightarrow{V} + \frac{\partial I}{\partial t} = 0, \tag{2}$$

where, $\overrightarrow{\nabla} I = (\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y})$ is the gradient of the image.
The gradient constraint equation given in (2) is in two unknowns and cannot be solved. This is known as the aperture problem of the optical flow. To solve this aperture problem, Lucas and Kanade [2] compute the optical flow on a point (u, v) considering that the motion is constant in a fixed neighborhood $\vartheta_{u,v}$ of this point. Let us note $\Theta = (a, b)$ the vector parameters of motion, the Lucas-Kanade's method consists to search the velocity vector $\overrightarrow{V}$ at the point (u, v) as the solution of :

$$\overrightarrow{V}(u, v) = \arg \min_{\Theta} \sum_{x,y \in \vartheta_{u,v}} [\frac{\partial I}{\partial x}.a + \frac{\partial I}{\partial y}.b + \frac{\partial I}{\partial t}]^2. \tag{3}$$

Although the constant model and the fixed neighborhood give good results for optical flow estimation in perspective cameras, this assumptions are never

verified in the omnidirectional images due to the distortions. In the paper, we propose to use new constraint based on motion model defined for paracatadioptric images and which is appropriate to the geometry of our image formation. In the next part, we will define a new constraint for the optical flow around the point (u, v). Then we will define a new neighborhood adapted to our omnidirectional images.

## 3 Motion model in paracatadioptric images

We consider omnidirectional sensor composed on parabolic mirror and orthographic camera (Fig. 1). The 3D point $P(X, Y, Z)$ is projected on the mirror as point $p_m(x_m, y_m, z_m)$ and on the image plan in $p(x, y)$.

According to the model projection used, the image projection p(x,y) of a 3D point P(X,Y,Z) can be defined by :

$$
\begin{cases}
x = \alpha_x \frac{h\ X}{\sqrt{X^2 + Y^2 + Z^2} - Z} + x_0 \\
y = \alpha_y \frac{h\ Y}{\sqrt{X^2 + Y^2 + Z^2} - Z} + y_0
\end{cases}
\tag{4}
$$

where α_x (respectively α_y) is a scale in x direction (respectively y direction) between world coordinates and image coordinates. (x_0, y_0) are coordinates of a principal point witch correspond to image coordinates of the projection of camera origin on the retina and h is the mirror parameter.

Note that in the case of Lucas-Kanade method, the motion model constant around the neighborhood is valid only if the camera moves belong to the fronto-parallel plane. We will consider the equivalent hypothesis : the camera moves on a plane which is perpendicular to its optical axis. For example, it corresponds to the case of omnidirectional camera embedded on a mobile robot.
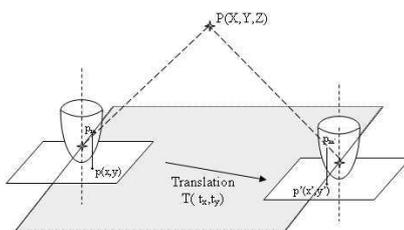
Fig. 1. Example of the camera motion on a plane perpendicular to its optical axis

The motion of the camera is equivalent to an inverse motion of the scene $\overrightarrow{T} = (t_x, t_y)$ of 3D point. The projection of translated 3D point in image is p'(x',y') and can be defined by:

$$\begin{cases} x' = \alpha_x \dfrac{h\ (X+t_x)}{\sqrt{(X+t_x)^2+(Y+t_y)^2+Z^2}-Z} + x_0 \\ y' = \alpha_y \dfrac{h\ (Y+t_y)}{\sqrt{(X+t_x)^2+(Y+t_y)^2+Z^2}-Z} + y_0 \end{cases} \tag{5}$$

To simplify this modelisation, we consider that the components of translation $\overrightarrow{T} = (t_x, t_y)$ are small and can be neglected compared to the 3D word coordinates of point P. Moreoever, the scale in x direction and y direction will be considered equal to simplify equations $\alpha_x = \alpha_y = \alpha$. Using this hypothesis, equation (5) can be simplified as:

$$\begin{cases} x' \simeq x + \dfrac{\alpha h\ t_x}{\sqrt{X^2+Y^2+Z^2}-Z} \\ y' \simeq y + \dfrac{\alpha h\ t_y}{\sqrt{X^2+Y^2+Z^2}-Z} \end{cases} \tag{6}$$

Thus, the expression of the motion can then be simplified as:

$$\begin{cases} V_x = \dfrac{\alpha h\ t_x}{\sqrt{X^2+Y^2+Z^2}-Z} \simeq t_x \dfrac{(x-x_0)}{X} \\ V_y = \dfrac{\alpha h\ t_y}{\sqrt{X^2+Y^2+Z^2}-Z} \simeq t_y \dfrac{(y-y_0)}{Y} \end{cases} \tag{7}$$

Then, using the inverse projection equations:

$$\begin{cases} X = \dfrac{2\ Z\ h\ \alpha(x_0-x)}{(x-x_0)^2+(y-y_0)^2-\alpha^2 h^2} \\ Y = \dfrac{2\ Z\ h\ \alpha(y_0-y)}{(x-x_0)^2+(y-y_0)^2-\alpha^2 h^2}, \end{cases} \tag{8}$$

we finally obtain, the equation of motion model by the following equations:

$$\begin{cases} V_x \simeq a(x-x_0)^2 + a(y-y_0)^2 + c \\ V_y \simeq b(x-x_0)^2 + b(y-y_0)^2 + d \end{cases} \tag{9}$$

where:

$$a = \frac{t_x}{2Zh\alpha}$$

$$b = \frac{t_y}{2Zh\alpha}$$

$$c = \frac{\alpha\ h\ t_x}{2Z}$$

$$d = \frac{\alpha\ h\ t_y}{2Z}.$$

Following to the motion model (eq. (9)), the translation of 3D point in space describes a circle in image plane and is depending on the position of point in image. The parameters of the motion depend on Z, h, α, and translation (t_x, t_y).

This motion model will be used to constraint the motion around the point (u, v) by searching $\Theta = (a, b, c, d)$ solution of:

$$\overrightarrow{V}(u,v) = \arg\min_{\Theta} \sum_{x,y \in \vartheta_{u,v}} [\frac{\partial I}{\partial x}.(a[(x-x_0)^2+(y-y_0)^2]+c)+\frac{\partial I}{\partial y}.(b[(x-x_0)^2+(y-y_0)^2]+d)+\frac{\partial I}{\partial t}]^2$$

(10)

The system of equations can be represented as:

$$\begin{cases} I_{x_1}g_1 a + I_{y_1}g_1 b + I_{x_1}c + I_{y_1}d = -I_{t_1} \\ \quad ..... \\ I_{x_i}g_i a + I_{y_i}g_i b + I_{x_i}c + I_{y_i}d = -I_{t_i} \quad (x_i, y_i) \in \vartheta_{u,v} \\ \quad ..... \\ I_{x_n}g_n a + I_{y_n}g_n b + I_{x_n}c + I_{y_n}d = -I_{t_n} \end{cases}$$

(11)

where:

$$g_i = (x_i - x_0)^2 + (y_i - y_0)^2$$

and $I_p = \frac{\partial I}{\partial p}$.

With this there are more than four equations for the four unknowns and thus the system is over-determined. To solve the over-determined system of equations, the least squares method is used. Therefore, the solution of (10) is given by:

$$\Theta = \begin{pmatrix} \sum I_{x_i}^2 \cdot g_i^2 & \sum I_{x_i} \cdot I_{y_i} \cdot g_i^2 & \sum I_{x_i}^2 \cdot g_i & \sum I_{x_i} \cdot I_{y_i} g_i \\ \sum I_{x_i} \cdot I_{y_i}^2 \cdot g_i^2 & \sum I_{y_i}^2 \cdot g_i^2 & \sum I_{x_i} \cdot I_{y_i} \cdot g_i & \sum I_{y_i}^2 \cdot g_i \\ \sum I_{x_i}^2 \cdot g_i & \sum I_{x_i} \cdot I_{y_i} \cdot g_i & \sum I_{x_i}^2 & \sum I_{x_i} \cdot I_{y_i} \\ \sum I_{x_i} \cdot I_{y_i} \cdot g_i & \sum I_{x_i}^2 \cdot g_i & \sum I_{x_i} \cdot I_{y_i} & \sum I_{y_i}^2 \end{pmatrix}^{-1} \begin{pmatrix} -\sum I_{x_i} \cdot g_i \cdot I_{t_i} \\ -\sum I_{y_i} \cdot g_i \cdot I_{t_i} \\ -\sum I_{x_i} \cdot I_{t_i} \\ -\sum I_{y_i} \cdot I_{t_i} \end{pmatrix}$$

(12)

4 Adapted neighborhood

It is known that rectangular windows usually used for cameras with perspective projections are not appropriate for catadioptric cameras with a non linear projection. Some authors, like in [8][9] and [10], have proposed to adapted neighborhood taking account of nature of the images. We propose adapted neighborhood window using the shape of the parabolic mirror. Each point in image presented with Cartesian coordinates $p = (u, v)$ is projected to the mirror point represented in spherical coordinates as $p_m = (r, \theta, \varphi)$ where θ denotes the elevation, φ the azimuth and r the distance of the point from the effective view point (Fig. 2). The desired neighborhood is obtained by selecting image points in interval fixed on a range of elevation and azimuth to determine the mirror point nearest the image point $x = (x, y)$. Let $p_{mi} = (r_i, \theta_i, \varphi_i)$ be the spherical coordinates of the projection image point $p_i = (x_i, y_i)$ in the mirror. The adapted neighborhood $\vartheta_{u,v}$ is defined as follows:

$$(x_i, y_i)\epsilon \, \vartheta_{u,v} \Leftrightarrow |\theta - \theta_i| < d\theta \text{ and } |\varphi - \varphi_i| < d\varphi.$$

(13)

As seen in (Fig. 6) and compared to fixed neighborhood, the adapted neighborhood will be bigger if the point is close to the mirror's periphery and smaller if the point is close to the image center. This is a proof that the neighborhood is now adapted to the resolution of these images since their size changes.
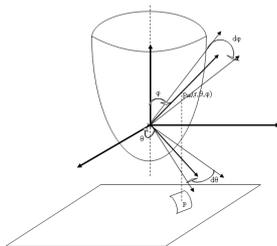
Fig. 2. Adapted neighborhood for catadioptric images

5 Experiments and Results

To show the amelioration given by our proposed method, we will compare it to classical Lucas-kanade's one in real and synthetic sequences. The two methods are applied to the synthetic sequences for which 2d motion fields are known. To compare these two methods we use two different performance measures. The first one is the angular error between the real velocity V_r and an estimated V_e [1]. It is the most popular measure of performance used to evaluate the optical flow estimation method and it is given by:

$$Ae = \arccos(\overrightarrow{V_r}.\overrightarrow{V_e}) \tag{14}$$

Although the angular error is prevalent, it is insufficient due to its bias for large flow witch are penalized less than errors in small flows. For this reason, we use also the normalized magnitude of the vector difference between the estimated flow vectors and the real ones [11] to compare the two methods. The magnitude errors Me (Eq. 15) allows to take account for small and large flow by fixing significance threshold. Following to the equation bellow, error score larger than 1 indicate that estimate flows are larger than the real flows in the scene. Error score 0 indicate perfect estimation where the error score 1 indicate zero estimation in V_e. Here we use $T = 0.5$.

$$Me = \begin{cases} \frac{\|V_r - V_e\|}{\|V_r\|} & if \ \|V_r\| >= T, \\ |\frac{\|V_e\| - T}{T}| & if \ \|V_r\| < T \ and \ \|V_e\| >= T, \\ 0 & if \ \|V_r\| < T \ and \ \|V_e\| < T, \end{cases} \tag{15}$$

5.1 Synthetic sequences

Due to the use of partial derivatives on the images, the images should be textured. We consider three different movement applied to camera in synthetic case to show the effectiveness of our contribution (see Fig. 3). The first sequence results from translation of 1cm along X-axis and 2cm along Y-axis. The (Fig. 4 showed the optical flow obtained with Lucas-Kanade method and that obtained with our new model constrained method with adapted neighborhood. The second sequence (Fig. 5)results from translation of 2cm along X-axis and 4cm along Y-axis. In (Fig. 6) we show the optical flow from image sequence obtained with rotation of camera around the Z-axis using $angle = 1°$. The last image sequence (Fig. 7) is obtained from translation of the camera with 4cm along Z-axis. Note that this last sequence is not appropriate to our motion model since the camera don't move on a plane perpendicular to it's optical axis. For all sequences, the intrinsic parameters are fixed on $\alpha = 100$ and $h = 2.3$. The windows size used in Lucas-kanade method is $15 * 15$ and equivalently the adapted neighborhood windows is arranged to have the same size for the point close to the image center. For this reason, the value of $d\theta$ and $d\varphi$ are fixed on: $d\theta = \Pi/25$, $d\varphi = \Pi/50$.

For synthetic image sequences, and using the known real optical flow, we calculate the mean of angular errors obtained from each sequences and also the standard deviation estimated from the Lucas-Kanade method and that from our approach. The results obtained on the errors values are indicated in (tab.1). It is seen that both, angular errors and standard deviation, are inferior in the adapted Lucas-Kanade's method. We also present the results in form of cumulative histogram graphs that give the percentage of pixels less than some values of angular and magnitude errors for two methods. This form of statistics results are used recently [12] [13] [11] to estimate the performance of algorithms. The best algorithm follows the top and the left sides of the histogram. It can be seen that it correspond to our proposed algorithm (see Fig. 8, 9, 10). We also show the vector velocity field in (Fig. 4, 5, 6, 7) for different synthetic image sequences. Although the translation along Z-axis is not appropriate to our motion model, the velocity field result on our adapted method seems more suitable than that on the Lucas-Kanade's method. Excepted this sequence, it is seen effectively that our constrained model with adapted neighborhood is very adequate to omnidirectional images due to his modified size comparing to the fixed one. It can then estimate the large movement in the periphery of image and solve the problem of temporal aliasing.

5.2 Real sequences

We have also used sequences of real omnidirectional images. The sequence is obtained using a camera witch is mounted on the mobile robot and moves along the Y-axis. In this case, the textured scene is more adequate with the uses of spatial derivatives in the images. Unfortunately in this case we don't have the ground truth for real sequence, we have use the error map to compare the results. The result optical flow is showed in (Fig. 11). Using a zoom on example of selected

area in image it can be seen that error map in adapted Lucas-Kanade's method is better. The error maps are obtained with motion compensation for the two compared method in the images.

6   conclusion

In perspective images, the Lucas-Kanade method is often used to compute optical flow from where cames our motivation to adapt it in omnidirectional case. In this paper, We have derived new constrains taking account of the nature of images to compute velocity vector field in catadioptric images. Using the projection model of our image formation, we have presented a motion model to replace the constant one in classical method. We have also adapted the neighborhood to catadioptric images. The adapted method is tested in synthetic and real images and is compared to the classical one. Using the derived motion model, and due to the modified size of adapted neighborhood in omnidirectional images, our algorithm can estimate the vector flows successfully in all different areas in image even with different resolution. However, the adapted neighborhood stage is expensive in term of computation cost in the proposed iterative algorithm while the phase of optical flow estimation does not differ from the traditional one. We will extend this work to other estimator of optical flow and we will generalize it to other models of catadioptric cameras.

References

1. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. International Journal of Computer Vision **12** (1994) 43–77
2. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81). (April 1981) 674–679
3. Demonceaux, C., Akkouche, D.K.: Optical flow estimation in omnidirectional images using wavelet approach. Conference on Computer Vision and Pattern Recognition Workshop: cvprw **7** (2003) 76
4. Gluckman, J., Nayar, S.: Ego-motion and omnidirectional cameras. In: ICCV '98: Proceedings of the Sixth International Conference on Computer Vision. (1998) 999
5. Daniilidis, K., Makadia, A., Blow, T.: Image processing in catadioptric planes: spatiotemporal derivatives and optical flow computation. In IEEE Workshop on omnidirectional Vision (2002) 3–12
6. Tosic, I., Bogdanova, I., Frossard, P., Vandergheynst, P.: Multiresolution Motion Estimation for Omnidirectional Images. In: In Proc. of European Signal Prossecing Conference. (2005)
7. Geyer, C., Daniilidis, K.: Catadioptric projective geometry. In: International Journal of Computer Vision. Volume 43. (2001) 223–243
8. Demonceaux, C., Vasseur, P.: Markov random fields for catadioptric image processing. Pattern Recogn. Lett. **27**(16) (2006) 1957–1967
9. Svoboda, T., Pajdla, T.: Matching in catadioptric images with appropriate windows, and outliers removal. In Proc. of the 9th International Conference on Computer Analysis of Images and Patterns **2124** (2001) 733–740

10. Ieng, S., Benosman, R., Devars, J.: An efficient dynamic multi-angular feature points matcher for catadioptric views. Proceedings of Omnivis 2003, Workshop on Omnidirectional Vision and Camera Networks, USA, Madison Wisconsin (2003)
11. McCane, B., Novins, K., Crannitch, D., Galvin, B.: On benchmarking optical flow. Comput. Vis. Image Underst. **84**(1) (2001) 126–143
12. Baker, S., Roth, S., Scharstein, D., Black, M., Lewis, J., Szeliski, R.: A database and evaluation methodology for optical flow. International Conference on Computer Vision : ICCV07 (2007) 1–8
13. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision **47**(1-3) (2002) 7–42
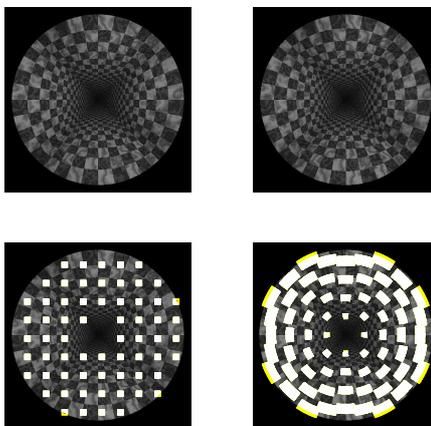
Fig. 3. Top: Example of synthetic images. Bottom left: The fixed and classical neighborhood , bottom right: Adapted neighborhood

| Sequences			Angular error	Standard deviation
sequence 1	Lucas-Kanade		$12.86°$	0.27
	Our approach		$11.37°$	0.21
Sequence 2	Lucas-Kanade		$24.76°$	0.43
	Our approach		$22.44°$	0.35
Sequence 3	Lucas-Kanade		$16.99°$	0.32
	Our approach		$15.26°$	0.24

Table 1. Mean of angular error and standard deviation for the three synthetic sequences verifying our motion model
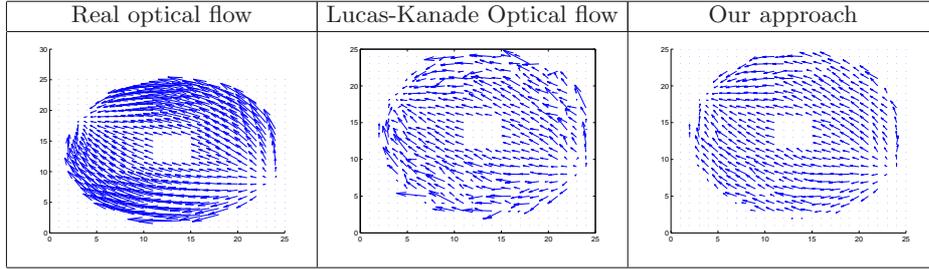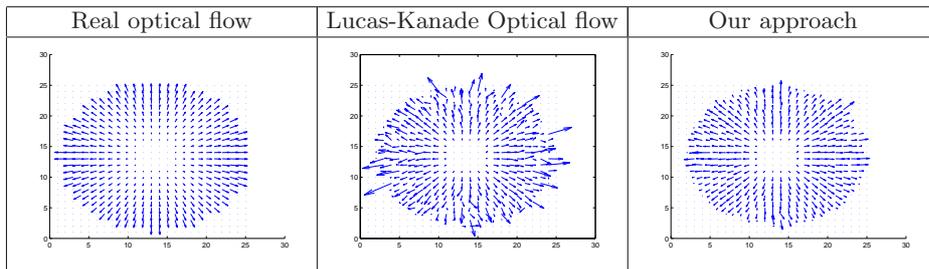
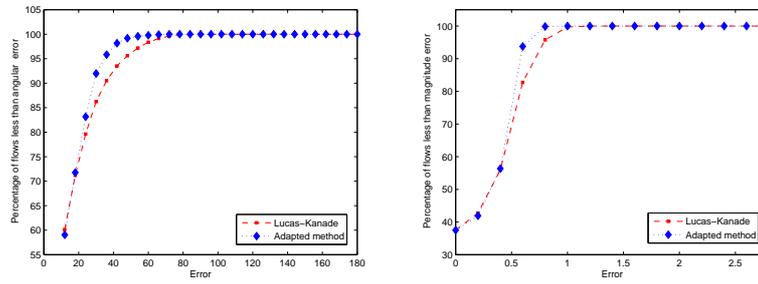**Fig. 4.** Optical flow with translation $tx = 2cm$ $ty = 1cm$
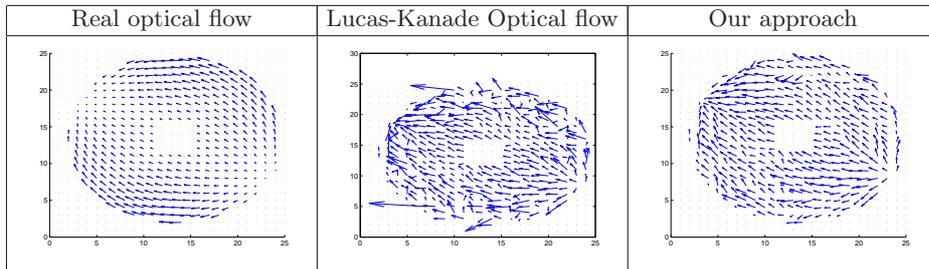
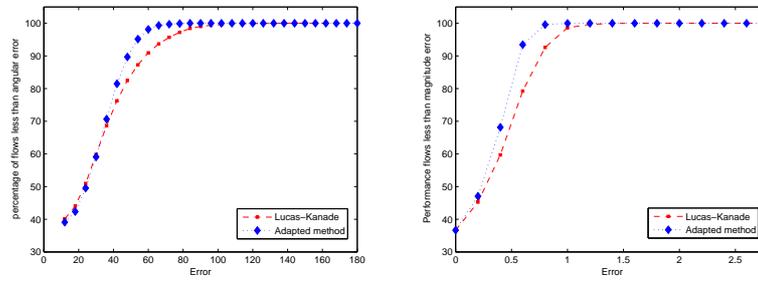**Fig. 5.** Optical flow with translation $tx = 4cm$ $ty = 2cm$
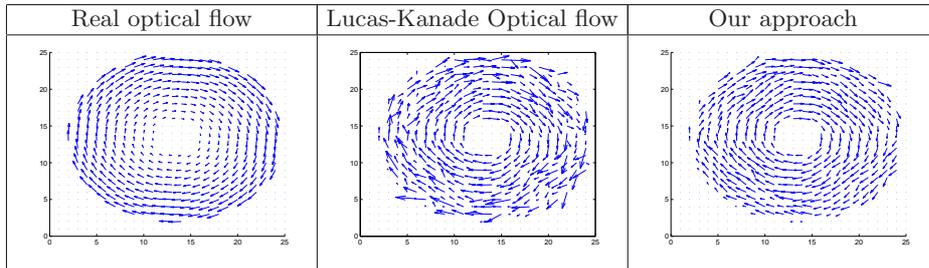
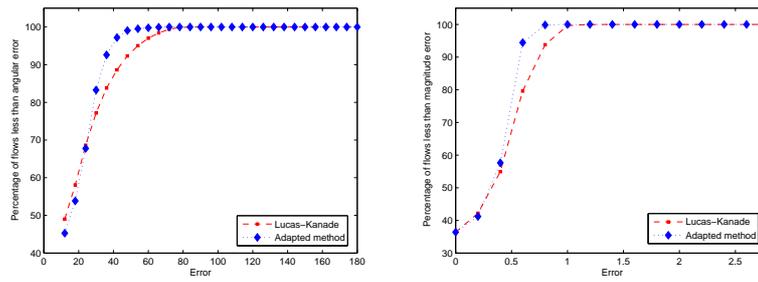**Fig. 6.** Optical flow with rotation around Z-axis of 1

**Fig. 7.** Optical flow with translation $tz = 2cm$

Fig. 8. Relative performance of the two algorithms with sequence :translation $tx = 2cm$ $ty = 1cm$. Left: percentage on angle error. Right: magnitude of difference performance.

Fig. 9. Relative performance of the two algorithms with sequence :translation $tx = 4cm$ $ty = 2cm$. Left: percentage on angle error. Right: magnitude of difference performance

Fig. 10. Relative performance of the two algorithms with sequence :rotation around Z-axis of 1. Left: percentage on angle error. Right: magnitude of difference performance
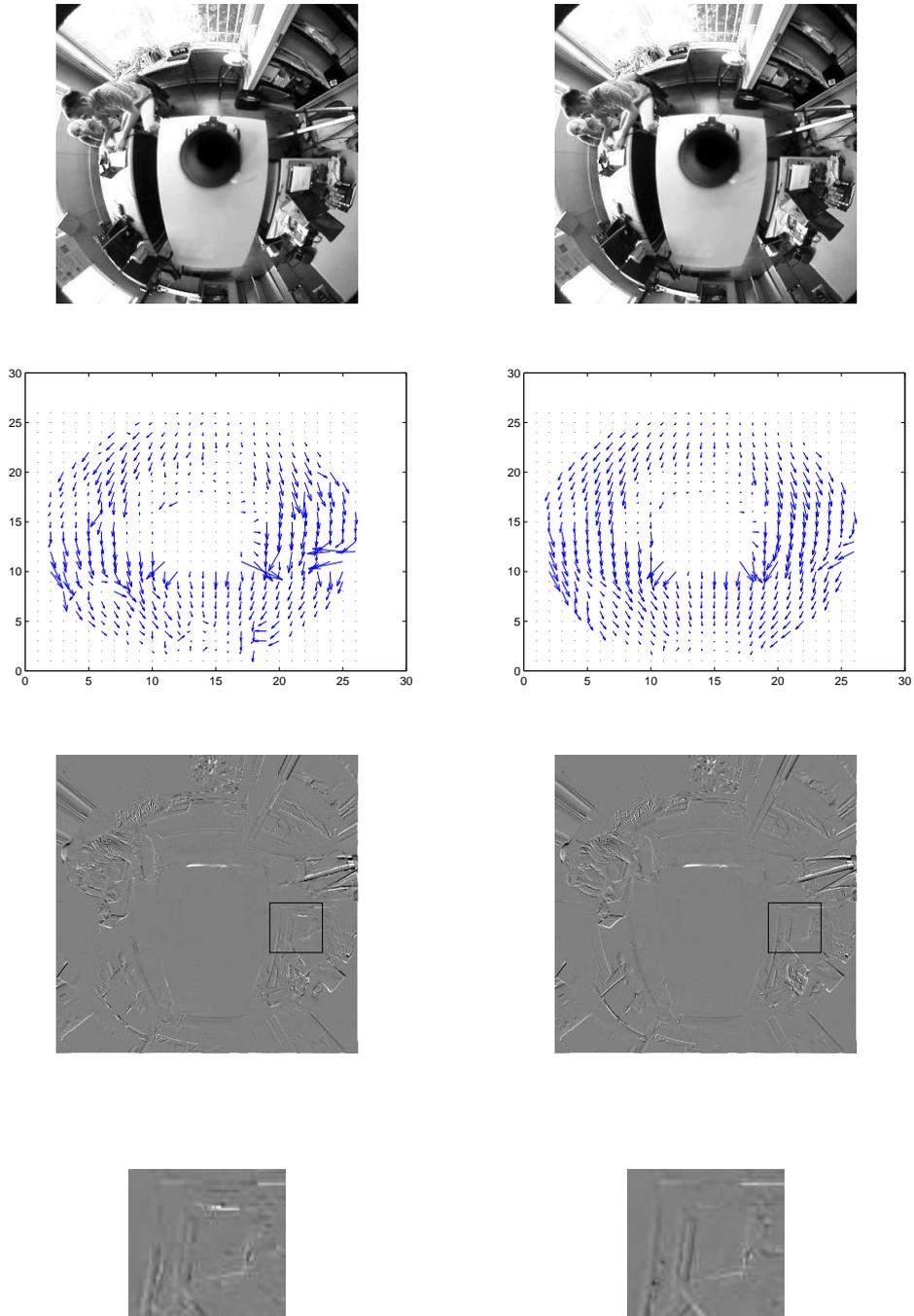
Fig. 11. Top: Example of pair in real image sequences. middle: optical flow obtained using Lucas-Kanade method (left) and our approach (right), Bottom: the error map and zoom on region indicate by selected rectangle