

# Optical Flow Estimation using Fourier Mellin Transform

Huy Tho Ho

School of EEE, Adelaide University, Australia

huy.ho@adelaide.anu.edu.au

Roland Goecke

RSISE, Australian National University, Australia

roland.goecke@anu.edu.au

## Abstract

*In this paper, we propose a novel method of computing the optical flow using the Fourier Mellin Transform (FMT). Each image in a sequence is divided into a regular grid of patches and the optical flow is estimated by calculating the phase correlation of each pair of co-sited patches using the FMT. By applying the FMT in calculating the phase correlation, we are able to estimate not only the pure translation, as limited in the case of the basic phase correlation techniques, but also the scale and rotation motion of image patches, i.e. full similarity transforms. Moreover, the motion parameters of each patch can be estimated to sub-pixel accuracy based on a recently proposed algorithm that uses a 2D sinc function in fitting the data from the phase correlation output. We also improve the estimation of the optical flow by presenting a method of smoothing the field by using a vector weighted average filter. Finally, experimental results, using publicly available data sets are presented, demonstrating the accuracy and improvements of our method over previous optical flow methods.*

## 1. Introduction

Without doubt, the measurement of optical flow is one of the fundamental problems in computer vision. It is the process of approximating the movement of brightness patterns in an image sequence and, thus, provides useful information for the determination of the 3D structure of the environment and the relative motion between the camera and the objects in the image [1]. There are many techniques for computing optical flow that have been proposed including *Gradient-based techniques* [10, 14] that are based on spatio-temporal derivatives of image intensity; *Variational techniques* [3, 7] that extend the gradient-based techniques by using different data or smoothness constraints [21]; *Block matching-based techniques* [1, 12] that assume all pixels in a block undergo the same motion; *Energy-based techniques* [9] that are based on the output energy of velocity-tuned filters [5]; *Bayesian techniques* [11] that utilize probability smoothness constraints in the form of a Gibbs random field; and

*Phase correlation techniques* [20] that compute the flow by applying the phase correlation to the images locally.

In addition to its applications in image registration, phase correlation has become one of the most common methods used for estimating the optical flow of image sequences. Its most important advantage is its immunity to illumination changes and moving shadows because the phase correlation function is normalized by the Fourier magnitude [2]. Furthermore, as the phase correlation method is performed in the frequency domain, it is much more computationally efficient than the block matching method [20]. However, the main disadvantage of this method is that it can only estimate displacements assuming a pure translation model within the image or block [16]. Our work extends the phase correlation technique by using the Fourier Mellin Transform (FMT) instead of the Fourier Transform to analyze not only the pure translation but also the scale and rotation motions of an image or image block. Thus, the proposed method can handle a full rigid motion model that the ordinary phase correlation could not. As a result, a more accurate estimation of the optical flow can be achieved. We also estimate the optical flow to sub-pixel accuracy based on a recently proposed algorithm that uses a 2D sinc function in fitting the phase correlation outputs [2]. Furthermore, a vector weighted average filter that uses phase correlation scores from the FMT as weighting coefficients is proposed in order to smooth the optical flow. Finally, the proposed approach is tested on a variety of image sequences including both synthetic and real scenes taken from publicly available data sets [4, 5, 15]. The results are compared with the ground truth as well as results by Lucas-Kanade's method [14], due to the popularity of this algorithm. We also compare our results with the multi-scale version of the anisotropic diffusion method of Proesmans *et al.* [18], which was highly regarded in the evaluations performed in [15], and with results of the 2D Combined Local Global (2D-CLG) method proposed by Bruhn *et al.* [7].

The remainder of the paper is organized as follows. Section 2 describes the FMT and how it can handle the translation as well scale and rotation motion of images or image patches. The application of FMT in calculating the opti-

cal flow of an image sequence is described in Section 3. In Section 4, we discuss how to improve the accuracy of our method by estimating sub-pixel accuracy and smoothing the flow field. Section 5 discusses a number of implementation issues. Next, in Section 6, we show a number of experimental results to demonstrate the effectiveness of our approach. Finally, the conclusions are given in Section 7.

## 2. Fourier Mellin Transform

The use of the Fourier Mellin Transform for rigid image registration was proposed in [19], that is to match images that are translated, rotated and scaled with respect to one another. Let  $F_1(\xi, \eta)$  and  $F_2(\xi, \eta)$  be the Fourier transforms of images  $f_1(x, y)$  and  $f_2(x, y)$ , respectively. If  $f_2$  differs from  $f_1$  only by a displacement  $(x_0, y_0)$  then

$$f_2(x, y) = f_1(x - x_0, y - y_0) \quad , \quad (1)$$

or in frequency domain, using the Fourier shift theorem [20]

$$F_2(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} \times F_1(\xi, \eta) \quad . \quad (2)$$

The cross-power spectrum is then defined as

$$C(\xi, \eta) = \frac{F_1(\xi, \eta)F_2^*(\xi, \eta)}{|F_1(\xi, \eta)F_2(\xi, \eta)|} = e^{j2\pi(\xi x_0 + \eta y_0)} \quad (3)$$

where  $F^*$  is the complex conjugate of  $F$ . The Fourier shift theorem guarantees that the phase of the cross-power spectrum is equivalent to the phase difference between the images. The inverse of (3) results in

$$c(x, y) = \delta(x - x_0, y - y_0) \quad (4)$$

which is approximately zero everywhere except at the optimal registration point.

If  $f_1$  and  $f_2$  are related by a translation  $(x_0, y_0)$  and a rotation  $\theta_0$  then

$$f_2(x, y) = f_1(x \cos \theta_0 + y \sin \theta_0 - x_0, \quad (5)$$

$$-x \sin \theta_0 + y \cos \theta_0 - y_0) \quad . \quad (6)$$

Using the Fourier translation property and the Fourier rotation property [19], we have

$$F_2(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} \times F_1(\xi \cos \theta_0 + \eta \sin \theta_0, -\xi \sin \theta_0 + \eta \cos \theta_0). \quad (7)$$

Let  $M_1$  and  $M_2$  be the magnitudes of  $F_1$  and  $F_2$ , respectively. They are related by

$$M_2(\xi, \eta) = M_1(\xi \cos \theta_0 + \eta \sin \theta_0, -\xi \sin \theta_0 + \eta \cos \theta_0) \quad (8)$$

To recover the rotation, the Fourier magnitude spectra are transformed to polar representation

$$M_1(\rho, \theta) = M_2(\rho, \theta - \theta_0) \quad (9)$$

where  $\rho$  and  $\theta$  are the radius and angle in the polar coordinate system, respectively. Then, (3) can be applied to find  $\theta_0$ .

If  $f_1$  is a translated, rotated and scaled version of  $f_2$ , the Fourier magnitude spectra are transformed to log-polar representations and related by

$$M_2(\rho, \theta) = M_1(\rho/s, \theta - \theta_0) \quad (10)$$

$$M_2(\log \rho, \theta) = M_1(\log \rho - \log s, \theta - \theta_0) \quad (11)$$

i.e.,

$$M_2(\xi, \theta) = M_1(\xi - d, \theta - \theta_0) \quad (12)$$

where  $s$  is the scaling factor,  $\xi = \log \rho$  and  $d = \log s$ .

The phase correlation in (3) is first applied to obtain the scale  $s$  and angle  $\theta_0$ , before the translation is found with another phase correlation on the scaled and rotated image.

Fig. 1 shows an example of frame 10 in the *Venus* sequence [4], together with its frequency log-polar and windowed frequency log-polar plots, respectively. More details on applying a window to an image before calculating its transform are discussed in Section 5.

## 3. Applying Fourier Mellin Transform to Optical Flow

Assume that we have two blocks  $B_1$  and  $B_2$  of size  $M \times N$  pixels located at the same position  $\mathbf{p} = (x, y)$  in two images  $f_1$  and  $f_2$ , respectively. If  $B_2$  is related to  $B_1$  by a translation  $(x_0, y_0)$ , scale  $s$  and rotation  $\theta_0$ , by applying the Fourier Mellin Transform as described in Section 2 to these two blocks, we can get the motion vector  $\mathbf{v}_{\mathbf{p}} = (v_x, v_y)$ .  $\mathbf{v}_{\mathbf{p}}$  can be computed as

$$v_x = \frac{N}{2} \{1 - (\cos \theta_0 + \sin \theta_0)s\} - x_0 \quad (13)$$

$$v_y = \frac{M}{2} \{1 - (\cos \theta_0 + \sin \theta_0)s\} - y_0 \quad (14)$$

The above calculation can be repeated for every pixel in the input images. However, we are also able to divide the input images into a (regular) grid and only perform the motion vector calculation on the nodes of the grid. The dense flow field can be obtained later by interpolation based on the smoothness constraint of the optical flow [8].

## 4. Improving the Accuracy of the Optical Flow

### 4.1. Sub-pixel Accuracy

Sub-pixel accuracy can be considered as a key performance issue in motion estimation. We employ a recently proposed method in [2] to obtain high accuracy sub-pixel motion estimates for our algorithm. We can increase the accuracy of the rotation and scale parameters by using high resolution for the log polar coordinates in Section 2. Thus,

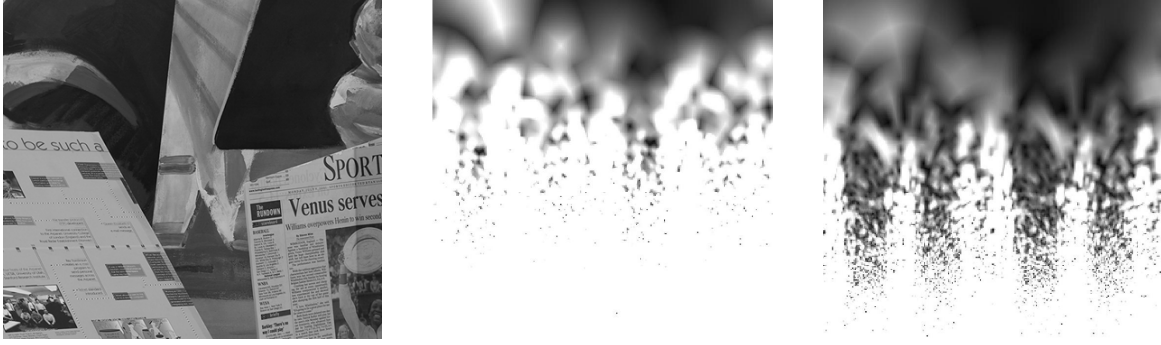


Figure 1: An example of a frame in the *Venus* sequence and its frequency log-polar and windowed frequency log-polar plots.

we only need to concentrate on getting the sub-pixel accuracy for the translation parameter.

Sub-pixel accuracy can be achieved by using bilinear interpolation, for example. However, such methods are computationally expensive and require the storage of a significant amount of data. In [2], a method of calculating sub-pixel accuracy by fitting a modified *sinc* function to the peaks of the phase correlation was proposed:

$$esinc(x) = \exp(-x^2) \frac{\sin \pi x}{\pi x} . \quad (15)$$

The function *esinc* is parameterized with regard to magnitude ( $A$ ), scale ( $B$ ), and shift changes ( $C$ ), i.e.  $Aesinc(B(x - C))$ . Let us just consider the fitting for the one-dimensional case (x-axis). Let  $c(x_m)$  be the maximum peak of the correlation surface and  $c(x_{m-1})$ ,  $c(x_{m+1})$  be the nearest neighbors on either side of  $c(x_m)$ . The unknown parameters  $A_x$ ,  $B_x$  and  $C_x$  can then be approximated by solving numerically the following least-squares minimization problem, for example, using a gradient descent method:

$$(A_x, B_x, C_x) = \underset{A_x, B_x, C_x}{\operatorname{argmin}} \sum_{x_i=x_{m-1}, x_m, x_{m+1}} \left[ c(x_i) - A_x \exp(-(B_x(x_i - C_x))^2) \frac{\sin(\pi(x_i - C_x))}{\pi(x_i - C_x)} \right]^2 . \quad (16)$$

We then compute  $A_y$ ,  $B_y$  and  $C_y$  in a similar way.

## 4.2. Smoothing the Optical Flow

There may be a few mismatches in the correlation output that can be considered as noise on the estimated optical flow. Therefore, we need a method of smoothing the flow to filter out these outliers. A vector weighted average filter that uses the phase correlation scores from the FMT computation is proposed as a solution to this problem.

Let  $\mathbf{v}_{\mathbf{p}_i} = (v_{x_i}, v_{y_i})$  be the estimated velocity for a pixel  $\mathbf{p}_i = (x_i, y_i)$ . Then, the output smoothed velocity at  $\mathbf{p}_i$  is

$\mathbf{v}'_{\mathbf{p}_i} = (v'_{x_i}, v'_{y_i})$ , computed as

$$\mathbf{v}'_{\mathbf{p}_i} = \frac{\sum_{\mathbf{p}_j \in N_i} (c_{\mathbf{p}_j} \mathbf{v}_{\mathbf{p}_j})}{\sum_{\mathbf{p}_j \in N_i} c_{\mathbf{p}_j}} , j \neq i \quad (17)$$

where  $\mathbf{p}_j$  is a pixel in  $N_i$ , the 8-pixel neighborhood of the pixel  $\mathbf{p}_i$ .  $c_{\mathbf{p}_j}$  is the peak in the correlation surface when calculating the translation for vector  $\mathbf{v}'_{\mathbf{p}_j}$ . In other words, the velocity at a pixel will be replaced by the weighted average of the velocities of its 8 neighbors with the weighting coefficients being the correlation scores.

## 5. Implementation

For the implementation of the algorithm in practice, we have to take into account a number of issues:

**Boundary effects:** The shift in the spatial domain must be cyclic in order to obtain a perfect impulse [16, 20]. Since the 2D Discrete Fourier Transform assumes periodicity in both directions, spurious peaks may be caused by discontinuities from left to right boundaries, and from top to bottom. In order to limit this effect, we can apply a 2D weighting window  $w(x, y)$  to each  $M \times N$  block. The window may be a Gaussian-like window or a raised-cosine window. In our implementation, we use the raised-cosine window or Hanning window to reduce the boundary effect [16]. The 2D Hanning window  $w(n_1, n_2)$  of size  $M \times N$  can be easily computed as an outer product of the two 1D Hanning windows  $w(n_1)w^T(n_2)$  where  $0 \leq n_1 \leq M$ ,  $0 \leq n_2 \leq N$  and  $w^T$  is the transpose of  $w$  [17]. The 1D Hanning window of size  $N$  is defined as

$$w(n) = 0.5 \left\{ 1 - \cos \left( \frac{2\pi n}{N-1} \right) \right\} \quad (18)$$

where  $0 \leq n \leq N$  [17].

**Spectral leakage:** The impulse will degenerate into a peak if the displacement vector does not correspond to an integer multiple of frequency [20]. Thus, we need to search not only the highest peak but also the surrounding peaks for the

optimal match.

**Range of displacement:** Due to the periodicity of the 2D DFT with the block size  $M \times N$ , only displacements  $(x_0, y_0)$  satisfy  $-\frac{N}{2} \leq x_0 \leq \frac{N}{2}$  and  $-\frac{M}{2} \leq y_0 \leq \frac{M}{2}$  can be detected [20].

**Block size:** It is one of the most important parameters in this algorithm as well as in any other phase correlation or block matching based motion estimation method. The block size should be large enough to capture large displacement vectors and small enough so that these vectors remain constant within the block. In our experiments, we see that a block size of  $32 \times 32$  provides the best results for an image size of around  $300 \times 500$ . More generally, this problem can be addressed by using hierarchical methods [1].

## 6. Experiments and Discussion

The proposed method was tested on a broad range of image sequences including both synthetic and real scenes. The images were taken from publicly available data sets [4, 5, 15] in order to guarantee the objectiveness of our evaluations. Many optical flow methods have been proposed in the literature. It would be beyond the scope of this paper to compare our FMT method to all of them. We compared our approach with the well-known algorithm of Lucas-Kanade [14]. The Matlab implementations of this algorithm were obtained from the Piotr’s Image and Video Processing Toolbox (<http://vision.ucsd.edu/~pdollar/toolbox/doc/>). The multi-scale version of the anisotropic diffusion method of Proesmans *et al.* [18], which produced the best overall results in the evaluation by McCane *et al.* [15], was also used in the comparison. The C implementation of this algorithm was downloaded from the website [http://www.cs.otago.ac.nz/gpxpriv/vision\\_optflow.html](http://www.cs.otago.ac.nz/gpxpriv/vision_optflow.html). The parameters of these algorithms were not changed. In addition, we also include the results from our implementation of the 2D Combined Local Global (2D-CLG) method proposed by Bruhn *et al.* [7]. The experimental results shows that our approach performs better than this highly regarded algorithm in most of the cases.

### 6.1. Error Metrics

The first measure of performance that we use in the comparison is the average angular error (*AAE*) [5]. This is the most common measure of performance for optical flow [4]. Let  $\mathbf{v}_0 = (u_0, v_0)$  be the correct velocity and  $\mathbf{v}_1 = (u_1, v_1)$  be the estimated velocity. The angular error (*AE*) between these two vectors is

$$\psi_E = \arccos(\vec{\mathbf{v}}_0 \cdot \vec{\mathbf{v}}_1) \quad (19)$$

where  $\vec{\mathbf{v}}_0, \vec{\mathbf{v}}_1$  are the 3D normalized representations of  $\mathbf{v}_0, \mathbf{v}_1$ , respectively and defined as

$$\vec{\mathbf{v}}_0 = \frac{1}{\sqrt{u_0^2 + v_0^2 + 1}}(u_0, v_0, 1) \quad (20)$$

$$\vec{\mathbf{v}}_1 = \frac{1}{\sqrt{u_1^2 + v_1^2 + 1}}(u_1, v_1, 1) \quad (21)$$

The *AAE* is then obtained by calculating the average of all the angular errors between correct and estimated velocities in the optical flow. However, it can be seen from (19) that errors in regions of large flows are penalized less in *AE* than errors in regions of small flows [4]. However, one needs to be cautious when using the *AAE* metric as estimates with the same error magnitude may result in significantly different angular error values [22].

Another error metric is the normalized magnitude of the vector difference between the correct and estimated flow vectors [15]. The magnitude of the correct velocity is used as the normalization factor. The magnitude of difference error is defined as

$$E_M = \begin{cases} \frac{\|\mathbf{v}_0 - \mathbf{v}_1\|}{\|\mathbf{v}_0\|} & \text{if } \|\mathbf{v}_0\| \geq T \\ \left| \frac{\|\mathbf{v}_1\| - T}{T} \right| & \text{if } \|\mathbf{v}_0\| < T \text{ and } \|\mathbf{v}_1\| \geq T \\ 0 & \text{if } \|\mathbf{v}_0\| < T \text{ and } \|\mathbf{v}_1\| < T \end{cases} \quad (22)$$

where  $T$  is a significant threshold. The algorithm is not expected to reliably produce accurate flow vectors in areas where the actual flow magnitude is less than  $T$  [15]. We used  $T = 0.5$  in all of our experiments. The average magnitude of difference error (*AME*) is then calculated as the average of the normalized magnitude of difference errors.

The percentage of computed flow vectors within a given error tolerance is also shown in the form of a cumulative error histogram. For better comparison with the McCane *et al.* [15] study, the average angular error is also divided into bins of  $18^\circ$  wide and the average magnitude of difference error is divided into bins of 0.2 wide similar to their study. The best algorithm is at the top of the graph as the ideal curve follows the left and top sides of the histogram.

### 6.2. Synthetic Data

The synthetic *street* sequence [15] is used for both qualitative and quantitative evaluation. The sequence and its ground truth flow are both available from the web site <http://www.cs.otago.ac.nz/research/vision/>. Figure 2a shows frame 14 of the *street* sequence. This is a complex sequence given the movement of the camera from left to right and the movements of the two cars in opposite directions with different velocities. Thus, the sequence is a good example to test our approach and compare with other methods. The ground truth flow and our result are shown in Figures

2b and 2c, respectively. The optical flow results obtained from Bruhn *et al.*, Lucas-Kanade, and Proesmans *et al.* are also shown in Figures 2d, 2e, 2f, respectively. Note that the optical flows have been resized and scaled in order to view them more easily. It can be seen clearly from the figures that our approach coincides well with the ground truth flow field.

The quantitative comparison in Table 1 confirms the qualitative results. The proposed approach outperforms the Bruhn *et al.*, Lucas-Kanade, and Proesmans *et al.* in both the AAE and AME error metrics.

### 6.3. Real Data

Results on real image sequences from the Middlebury data sets [4] (<http://vision.middlebury.edu/flow/data/>) are also reported in order to show that the proposed approach also works well on real-world data. According to Baker *et al.* [4], these sequences are much more complex than the data sets used in [5]. They are claimed to contain all the components that make the optical flow ambiguous and difficult including aperture problem, textureless regions, motion discontinuities, occlusions, large motions, small objects, non-rigid motion, mixed pixels, changes in illumination, motion blur, non-Lambertian reflectance, and camera noise [4]. Thus, these sequences provide more meaningful comparisons between different optical flow algorithms.

Figures 3 and 5 show the ground truth optical flows and the results of different techniques including the proposed FMT approach on the two real sequences *Venus* and *RubberWhale*, respectively. In both cases, the optical flows obtained using the FMT method are the closest to the ground truth. The superior performance of our method is confirmed by the quantitative results in Table 1 and the cumulative histograms in Figures 4 and 6, where it outperformed the other methods both in terms of average angular error and magnitude of difference error. In Table 1, we also include the results of all four algorithms on two other sequences from the Middlebury’s database [4] including the *Dimetrodon* and *Hydrangea* sequences. In these two sequences, our approach still provides the best results compared to all other algorithms. Furthermore, the AAEs of the proposed FMT method on the *Venus* and *Dimetrodon* sequences ( $5.51^\circ$  and  $7.33^\circ$ , respectively) are lower than for the Black-Anandan method [6] ( $7.64^\circ$  and  $9.26^\circ$ , respectively) and the Zitnick method [23] ( $11.42^\circ$  and  $30.10^\circ$ , respectively). The AAEs of these state-of-the-art algorithms are quoted from the Middlebury vision web site [4].

Fig. 4 and 6 show the cumulative histograms of the angular error and magnitude of difference error for all four algorithms in the two sequences *Venus* and *RubberWhale* respectively. It can be seen from the graphs that our approach performs better than the other three methods. In the FMT results for the *Venus* sequence, more than 90% of the

angular errors are less than  $18^\circ$  and nearly 90% of the magnitude of difference errors are less than 0.2. The methods of Proesmans *et al.* and Bruhn *et al.* have quite similar performance in both the angular and magnitude of difference error metrics in these two above sequences. However, their results are still not comparable to our approach. The method of Lucas-Kanade does not provide good results in the *Venus* sequence. There are only 35% of the angular errors are less than  $18^\circ$  and only 10% of the magnitude of difference errors are less than 0.2 in the Lucas-Kanade. However, its performance is quite similar to the performance of the methods of Proesmans *et al.* and Bruhn *et al.* in the *RubberWhale* sequence.

## 7. Conclusion

We have presented a method of estimating the optical flow of an image sequence using the Fourier Mellin Transform. Our approach overcomes the limitations of previous phase correlation techniques by converting the Fourier transforms of image patches into log polar coordinates and thus, being able to estimate not only the translation but also the scale and rotation motion of the patches, i.e. full similarity transforms. Furthermore, the optical flow is estimated to sub-pixel accuracy using a recently proposed algorithm that fits an *esinc* function to the correlation surface. The estimated flow is further smoothed by a vector weighted average filter based on the phase correlation scores of the FMT in order to reject outliers and further improve the accuracy. Results comparing this approach to other optical flow algorithms show its effectiveness. In the future, we plan to investigate the integration of a search for local keypoints, such as the keypoints of the Scale Invariant Feature Transform (SIFT) [13], into our framework as a pre-processing step in order to limit the computationally more expensive optical flow calculations to distinctive features of the image sequence. This could help to reduce the computational cost and to improve the accuracy of the algorithm.

## Acknowledgements

This work was done at the Research School of Information Sciences and Engineering (RSISE), the Australian National University. We would like to thank Jason Saragih for providing the implementation of the 2D-CLG method used in the comparison. We also would like to thank Prof. Michael J. Black for providing the image sequences from the Middlebury’s database.

## References

- [1] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *IJCV*, 2(3):283–310, 1989. 1, 4

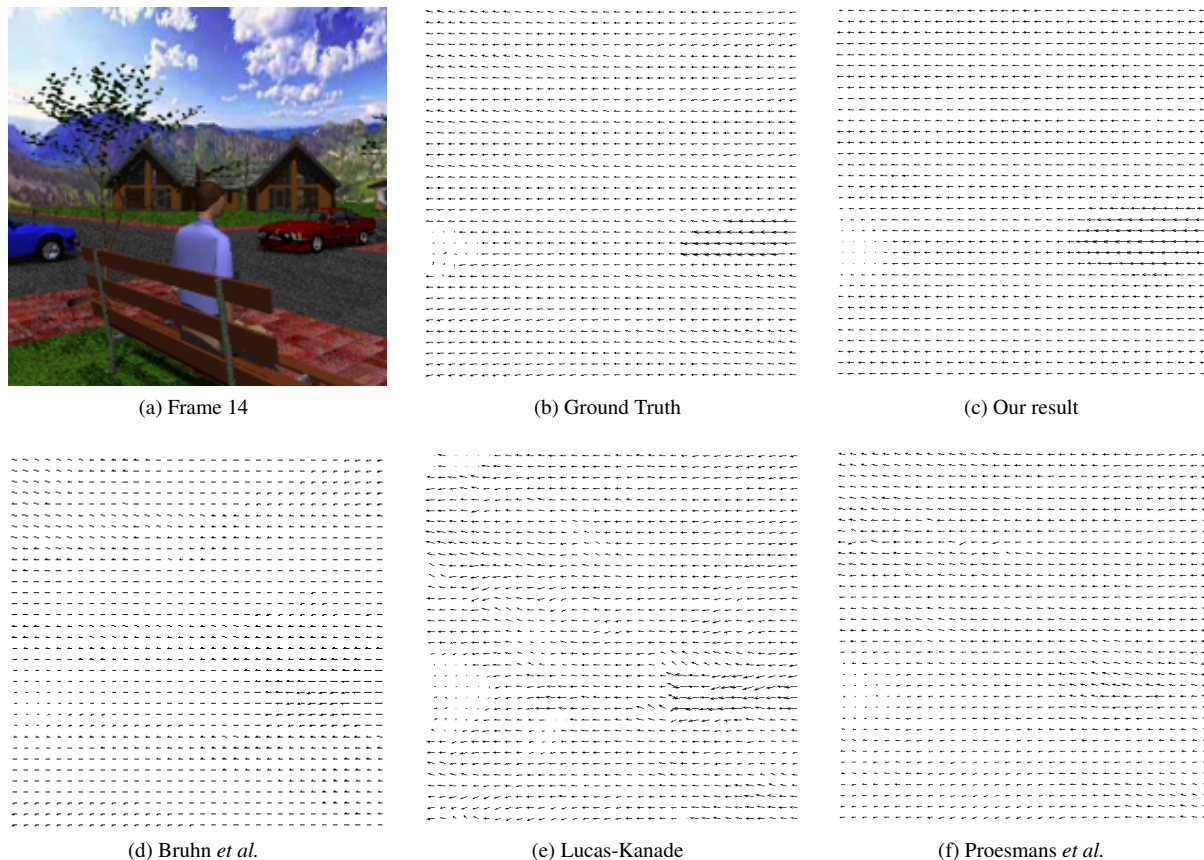


Figure 2: Experimental results for the *Street* sequence.

Method	Street		Venus		RubberWhale		Dimetrodon		Hydrangea	
	<i>AAE</i>	<i>AME</i>	<i>AAE</i>	<i>AME</i>	<i>AAE</i>	<i>AME</i>	<i>AAE</i>	<i>AME</i>	<i>AAE</i>	<i>AME</i>
Lucas-Kanade	6.45°	0.18	41.27°	0.74	18.69°	0.46	37.14°	0.66	29.85°	0.81
Proesman <i>et al.</i>	6.31°	0.17	18.25°	0.45	17.43°	0.38	22.23°	0.50	21.49°	0.82
2D-CLG	4.75°	0.15	19.38°	0.45	16.75.°	0.37	18.59°	0.40	27.87°	0.81
FMT (proposed)	4.66°	0.14	5.51°	0.14	10.07°	0.26	7.33°	0.18	11.83°	0.56

Table 1: Comparison of *AAE* and *AME* error metrics for different methods on synthetic and real image sequences

- [2] V. Argyriou and T. Vlachos. A study of sub-pixel motion estimation using phase correlation. In *Proc. BMVC*, volume 1, pages 387–396, 2006. 1, 2, 3
- [3] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM J. App. Maths.*, 60(1):156–182, 1999. 1
- [4] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *Proc. ICCV*, 2007. 1, 2, 4, 5
- [5] J. Barron, D. Fleet, and S. Beauchemin. Performance of Optical Flow. *IJCV*, 12(1):43–77, 1994. 1, 4, 5
- [6] M. Black and P. Anandan. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *CVIU*, 63(1):75–104, 1996. 5
- [7] A. Bruhn, J. Weickert, and C. Schnorr. Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods. *IJCV*, 61(3):211–231, 2005. 1, 4
- [8] C. Fuh and P. Maragos. Region-based optical flow estimation. In *Proc. CVPR*, pages 130–135, 1989. 2
- [9] D. Heeger. Optical flow using spatiotemporal filters. *IJCV*, 1(4):279–302, 1988. 1
- [10] B. Horn and B. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17(1–3):185–203, 1981. 1
- [11] J. Konrad and E. Dubois. Comparison of stochastic and deterministic solution methods in Bayesian estimation of 2D motion. *Img. and Vis. Comp.*, 9(4):215–228, 1991. 1
- [12] B. Liu and A. Zaccarin. New fast algorithms for the estimation of black motion vectors. *IEEE Trans Circ. and Syst. Vid.*

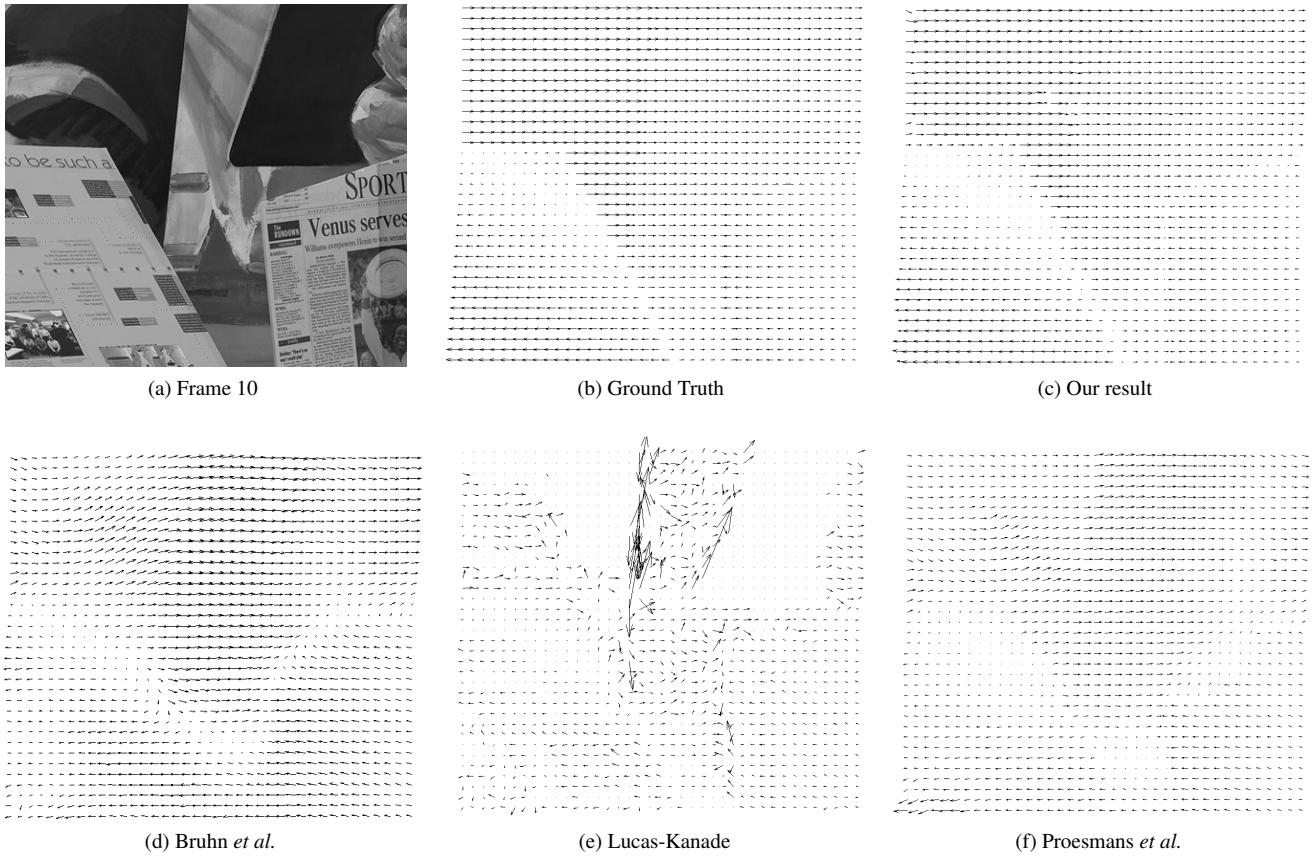


Figure 3: Experimental results for the *Venus* sequence

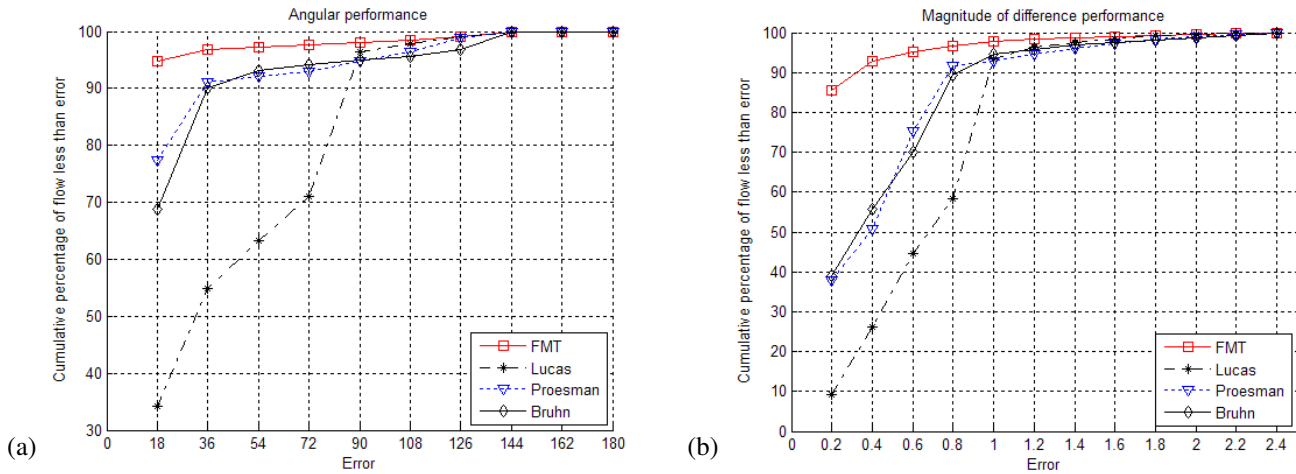


Figure 4: Performance comparison of four algorithms including the proposed FMT approach on the *Venus* sequence. Cumulative histograms for (a) average angular error and (b) average magnitude of difference error.

*Tech.*, 3(2):148–157, 1993. 1

[13] D. Lowe. Distinctive Image Features From Scale-Invariant Keypoints. *IJCV*, 60(2):91–110, 2004. 5

[14] B. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proc.*

*IJCAI 1981*, pages 674–679, 1981. 1, 4

[15] B. McCane, K. Novins, D. Crannich, and B. Galvin. On Benchmarking Optical Flow. *Comp. Vis. and Imag. Under.*, 84(1):126–143, 2001. 1, 4

[16] F. Pla and M. Bober. Estimating translation/deformation mo-

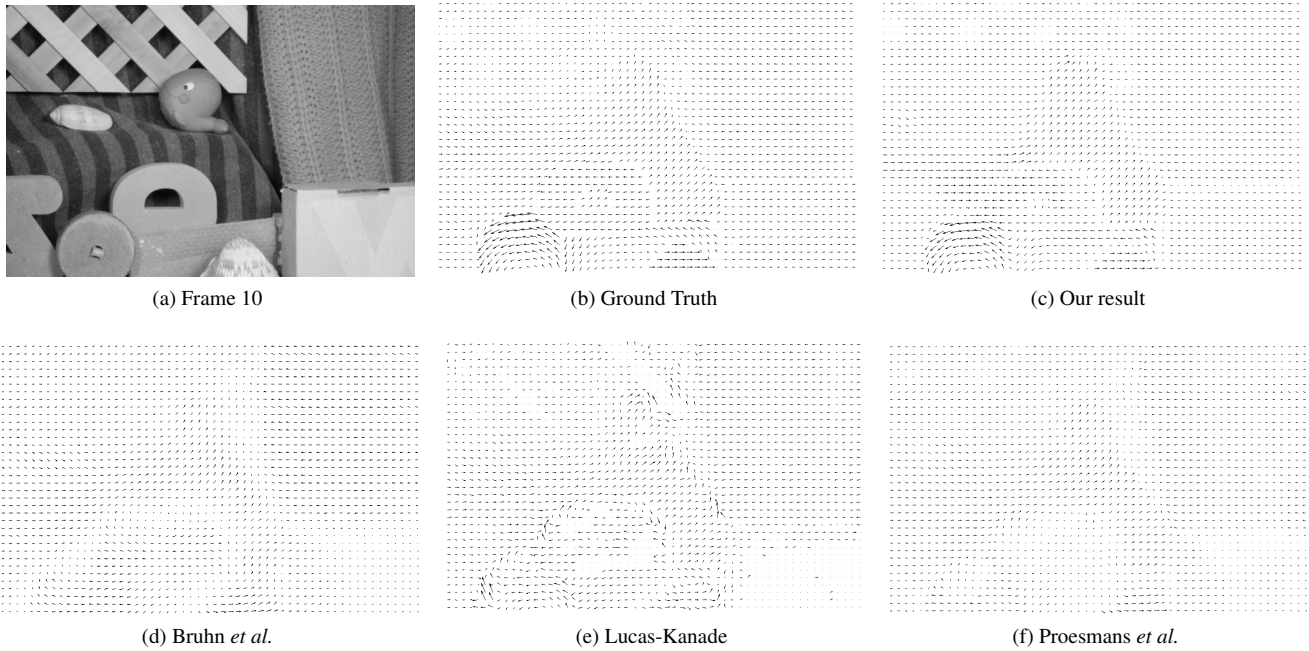


Figure 5: Experiment results for the *RubberWhale* sequence

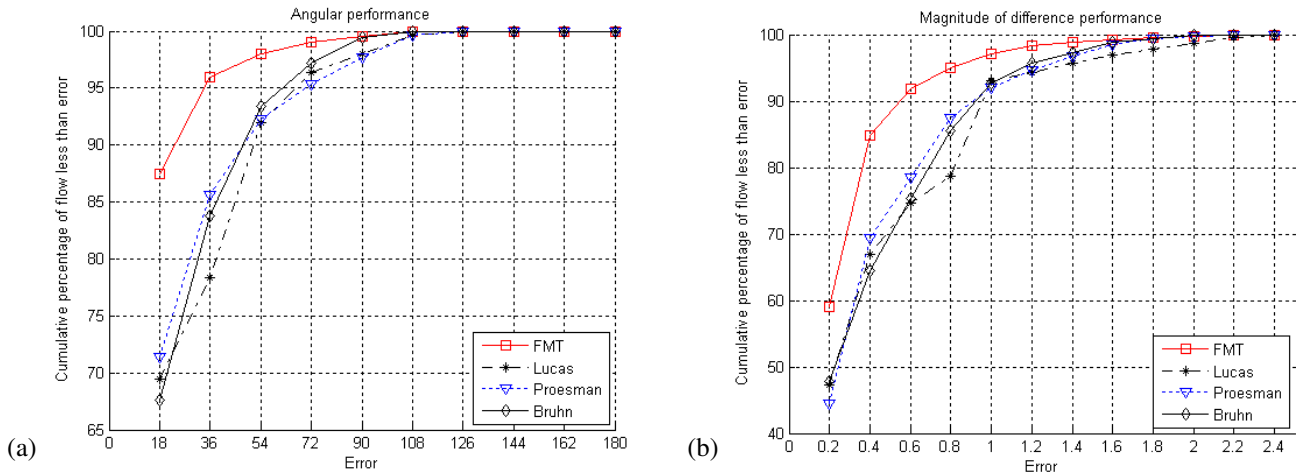


Figure 6: Performance comparison of four algorithms including the proposed FMT approach on the *RubberWhale* sequence. Cumulative histograms for (a) average angular error and (b) average magnitude of difference error.

tion through phase correlation. In *Proc. Int. Conf. Image Anal. and Proc.*, volume 1, pages 653–660, 1997. 1, 3

[17] J. Proakis and D. Manolakis. *Digital signal processing*. Prentice Hall, 2006. 3

[18] M. Proesmans, L. V. Gool, E. Pauwels, and A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. In *Proc. ECCV*, pages 295–304, 1994. 1, 4

[19] B. Reddy and B. Chatterji. An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. Image Proc.*, 5(8):1266–1271, 1996. 2

[20] A. Tekalp. *Digital video processing*. Prentice Hall, 1995. 1, 2, 3, 4

[21] J. Weickert and C. Schnorr. Variational optic flow computation with a spatio-temporal smoothness constraint. *JMIV*, 14(3):245–255, 2001. 1

[22] M. Ye, R. M. Haralick, and L. Shapiro. Estimating piecewise-smooth optical flow with global matching and graduated optimization. *IEEE Trans. PAMI*, 25(12):1625–1630, 2003. 4

[23] C. Zitnick, N. Jovic, and S. Kang. Consistent segmentation for optical flow estimation. In *Proc. ICCV*, volume 2, pages 1308–1315, 2005. 5