Robust Digital Image Stabilization Based on Spatial-location-invariant Criterion

Xiaojiang Peng, Junzhou Chen and Jiashu Zhang
College of Information Science and Technology,
Southwest Jiaotong University, Chengdu, China
Email: xiaojiang_p@yahoo.com, {jzchen, jszhang}@swjtu.edu.cn

***Abstract*—In this paper, a robust digital image stabilization (DIS) algorithm based on a spatial-location-invariant criterion is proposed to estimate the local motion vectors in jitter videos. In the algorithm, the feature points are detected by sub-region Harris detector and matched by cross diamond hexagon search algorithm, and then in order to obtain accurate local motion vectors, the presented criterion called spatial-location-invariant is used to remove those points that error-matching or in moving objects. With these vectors, the global motion vector is computed by least-squares algorithm in similarity motion model. Finally the real motion model is applied to motion compensation. The experimental results show that the proposed DIS technique can deal with arbitrary rotation, translation, and is robust to foreground moving objects.**

I. INTRODUCTION

Recently, video content analysis is receiving more and more attention in the video surveillance system, visual navigation system, TV-guided system and etc. In some applications, however, unwanted video vibrations would lead to degraded performances of analysis algorithms. Digital image stabilizer can be used as a front-end system in most of the mentioned applications, which is to eliminate jitter caused by the undesired camera shake.

Generally, digital image stabilization algorithm is composed of three main processes: motion estimation (ME), motion compensation (MC) and image compensation (IC), as show in Fig.1. Usually, the ME unit includes local motion estimation and global motion estimation, and the purpose of this unit is to estimate the global motion vector caused by unwanted camera motion. With the correct global motion vector (GMV), the MC unit then warps frames to create a more visual stable image sequence. In order to obtain a full frame, the IC unit is used to restore the undefined areas.

Among the above three processes, the ME module is the most important role which is similar to image registration essentially. The challenges of ME in DIS may include: 1) the presence of foreground moving objects, 2) the existence of translation, rotation and scaling transformation at the same time. Several methods have been proposed to estimate the motion parameters such as: 1) global motion estimation using projection algorithm [1] and phase correlation-based algorithm [2]; 2) motion estimation based on blocks matching [3-4][13]; 3) fast motion estimation based on bit-plane and gray-coded bit-plane matching [5-6]; 4) motion estimation based on log-polar transformation [7]; 5) motion estimation based on

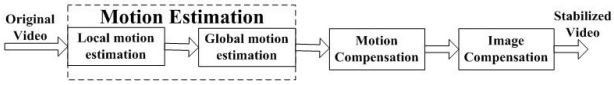


Fig. 1. Framework of DIS.

SIFT features matching algorithm [8-9]; 6) motion estimation based on optical flow technique [10]. The first three methods estimate only the translational movement and produce poor performance when the image fluctuation contains rotational motion dominantly. The latter three algorithms can provide good performance in estimating translational, rotational and scaling motion, but are time-consuming. Additionally, all of them are less robust to illumination and foreground moving objects in real application. Recently, in order to improving long casual video captured by amateur, Liu applied subspace constraints on feature trajectories [11] and content-preserving warps [12], and a very perfect performance was reached in most videos recorded by hand-held devices. But it was not real-time system, just an off-line software.

In this paper, to stabilize the unstable video with moving objects, a robust local motion estimation method based on Harris feature points and spatial-location-invariant criterion is devised. Then the real motion model parameters are generated from similarity motion model parameters.

The remaining part of this paper is organized as follows. Section II gives our detail solution to local motion estimation based on the spatial-location-invariant criterion. The scheme of our global motion estimation and compensation is described in section III. Section IV presents some experimental results and Section V concludes the paper.

II. ROBUST LOCAL MOTION ESTIMATION BASED ON SPATIAL-LOCATION-INVARIANT CRITERION

The problem we are addressing in our project is that of using DIS algorithm to stabilize the vibratory surveillance video. To extract local motion vectors fast and accurately, our earlier work has attempted to apply feature blocks matching like [4], bit plane matching like [5], optical flow like [10] and etc. All of them were proved to be not robust enough to the presence of illumination change, foreground moving objects and rotation. Then we have turn to develop a robust algorithm using Harris feature points based on a spatial-location-invariant criterion.

978-1-61284-972-0/11/$26.00 ©2011 IEEE

A. Harris Feature Points

The Harris detector [13] has been proved one of the most successful algorithms in corner and edge detection. It is rotationally invariant improved from Moravec's corner detector.

The Harris matrix M at (x_i, y_i) is define as:

$$\mathbf{M} = \begin{bmatrix} (\frac{\partial I_{(x_i,y_i)}}{\partial x})^2 & \frac{\partial I_{(x_i,y_i)}}{\partial x}\frac{\partial I_{(x_i,y_i)}}{\partial y} \\ \frac{\partial I_{(x_i,y_i)}}{\partial x}\frac{\partial I_{(x_i,y_i)}}{\partial y} & (\frac{\partial I_{(x_i,y_i)}}{\partial y})^2 \end{bmatrix} \otimes w \quad (1)$$

Where $\frac{\partial I_{(x_i,y_i)}}{\partial x}$ and $\frac{\partial I_{(x_i,y_i)}}{\partial y}$ are the gradients in horizontal and vertical direction at (x_i, y_i) which usually computed by $[-1; 0; 1]$ and $[-1; 0; 1]^T$ gradient filter or 3×3 Sobel masks, w is a weight value in the Gaussian window. And then the response function is given as:

$$\mathbf{R}(x_i, y_i) = Det(M) - kTr^2(M) \quad (2)$$

Harris and Stephens have proved if the value of response function in (x_i, y_i) is a local maximum and larger than a threshold, then (x_i, y_i) is a feature points. In practice, we can adjust the threshold to get the wanted numbers of feature points.

B. Sub-region Harris Detector

In our video, the foreground moving objects usually have many feature points which would degrade the performance of our scheme by using a global threshold Harris detector. Considering that the moving objects are unlikely to fill the whole frame, we have divided the frame into four sub regions and applied the Harris Detector with different thresholds to them respectively. By doing this we have achieved scattered feature points instead of dense ones.

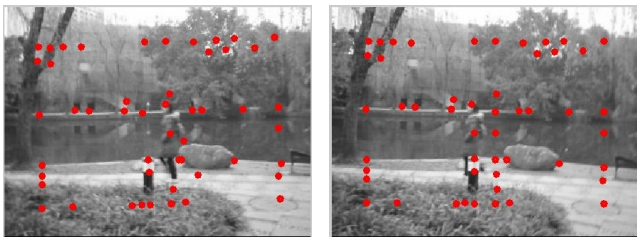
C. Feature Points Match by Cross Diamond Hexagon Search Algorithm

After detecting the Harris feature points in current frame, we need to get the corresponding points in the reference frame for the purpose of obtaining local motion vectors (LMVs). In the beginning, we have extracted feature points in the two frames and let every feature points in current frame match the feature points in the reference one by adopting the criterion of minimizing the sum of absolute difference (SAD). However, that too many different Harris feature points in the two frames because of rotation would lead to sharply reduced pairs of matching points, and the process is time-consuming.

Then we have followed the motion estimation in video coding technique. Full search algorithm can catch accurate corresponding points but the large calculation is intolerant. Cross diamond hexagon search (CDHS) algorithm is proved an effective fast search method to get best match [14]. The matching result by using CDHS algorithm is shown in Fig.2.

D. Spatial-location-invariant Criterion and Checking

Note that the error-matching points and the matching points at moving objects are unavoidable sometimes as shown in Fig.2. Obviously, if we compute the GMV using these matching points would lead to inaccurate result. Considering this, Xu



(a) (b)

Fig. 2. Matching points by CDHS algorithm with SAD criterion. (a) Reference frame. (b) Current frame.

Lidong [4] used a recursive least squares (RLS) after getting the inaccurate result. However, it is very time-consuming. In our video, scaling factor is nearby one and then for rigid body, the distance between two points is constant after affine transformation. This is called spatial-distance-invariant criterion. The mathematical specification of this is given by:

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} = \sqrt{(x_i^{'} - x_j^{'})^2 + (y_i^{'} - y_j^{'})^2} \quad (3)$$

Where (x_i, y_i), (x_j, y_j), $(x_i^{'}, y_i^{'})$, $(x_i^{'}, y_i^{'})$ are the position of two points in source image and transformed image.

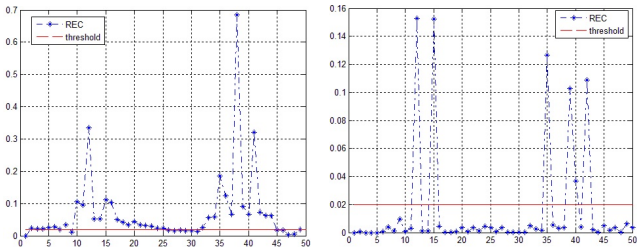
But in the video sequences, various noises will cause an error, and the error become large when the distance is long between two points due to the scale factor. Here we have used the relative error to measure the change of distance after transformation. The relative error between the distances is defined as:

$$\varepsilon = \frac{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - \sqrt{(x_i^{'} - x_j^{'})^2 + (y_i^{'} - y_j^{'})^2}}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \quad (4)$$

Then we can set a threshold to judge whether the two points which generating the distance in current frame are accurate matching or not.

Before applying this criterion to check the error-matching or foreground points, we must find an accurate matching point which is not the feature point at moving objects. Here we have used an iteration method to get it. We have connected a point with the other points in current frame and computed their relative errors forming a relative error curves (REC). If more than a half of the relative errors are larger than the given threshold as shown in Fig.3 (a), we delete this point and move to the next point to do the same operation; else we regard this point as a correct matching point as shown in Fig.3 (b). In the correct-matching point's REC, we delete those points which caused large relative error and save the others to compute the global motion vector. In Fig.3 (b), we would delete the twelfth, fifth points or LMVs and etc.

Sometimes, another issue we would face to is that if we only to checking by the above method, those error-matching points which happen to be on the circumferences of the correct-matching points will not be deleted. For example, in Fig.4, point P and $P^{'}$ cannot be deleted when $P1$ and $P1^{'}$ is the correct-matching point pair we have found. Experiments



(a) (b)

Fig. 3. The relative error curves (REC). (a) REC for error-matching point. (b) REC for correct-matching point.

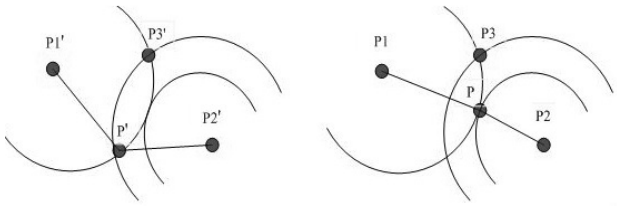
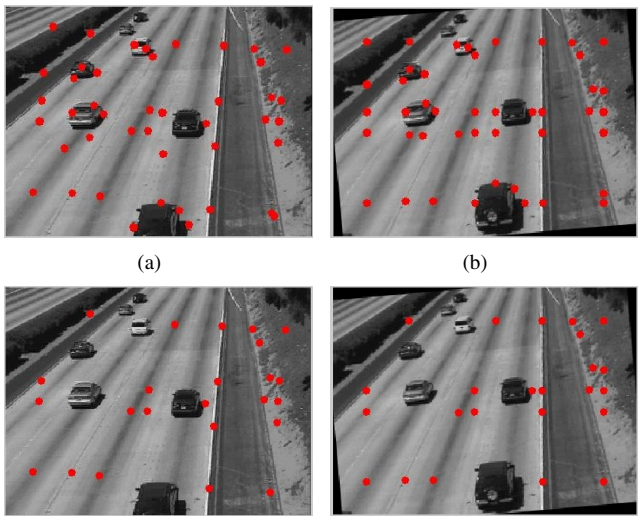


Fig. 4. An exception of spatial-distance-invariant criterion.



(a) (b) (c) (d)

Fig. 5. A representative artificial sample of our checking criterion. (a) and (b) Matching points in reference and current frame. (c) and (d) The correct-matching points in reference and current frame after checking.

demonstrate that these error-matching points are mainly caused by the fast search method, motion blur and noise. Considering that the error-matching point cannot be on the circumferences of two correct-matching points meanwhile, thus, we have chosen two correct-matching points to check in the current frame.This method is named spatial-location-invariant criterion which represents the topological structure among three points is invariant. As in Fig.4, we can use $P2$ and $P2'$ to check again. By doing this, we can delete arbitrary number of existing error-matching points or foreground moving points. A representative artificial sample is shown in Fig.5. The top two images display the matching points by CDHS algorithm. And the bottom two images show the checking result. As we see, the error-matching points and the matching points in the moving vehicle have been deleted.

III. GLOBAL MOTION ESTIMATION AND COMPENSATION

A. Real Motion Model and Similarity Motion Model

From the correct LMVs, we can choose a motion model to estimate the GMV. To get good performance in our projection, we have considered all the parameter including scaling change, rotation and translation. The real motion model can be defined as:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \gamma \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \end{bmatrix} \tag{5}$$

Where (x_0, y_0) is the rotation center,γ is the scaling factor, (x, y) and (x', y') is the coordinate in the current frame and reference frame respectively, d_x and d_y are horizontal and vertical translation parameters respectively.

The real motion model can be simplified to the similarity motion model:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix} \tag{6}$$

From (5) and (6), we get a, b, c, d such that

$$a = \gamma \cos\theta \tag{7}$$

$$b = \gamma \sin\theta \tag{8}$$

$$c = (1 - a)x_0 + by_0 + d_x \tag{9}$$

$$d = (1 - a)y_0 + bx_0 + d_y \tag{10}$$

Here we see, that c and d are not only relevant to translation as described in [4], but also the rotation. Combining n points, we can get a linear system of equations as follows:

$$\begin{bmatrix} x_1 & -y_1 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & -y_n & 1 & 0 \\ y_1 & x_1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ y_n & x_n & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} x_1' \\ \vdots \\ x_n' \\ y_1' \\ \vdots \\ y_n' \end{bmatrix} \tag{11}$$

The equivalent form is like this:

$$\mathbf{Av} = \mathbf{b}' \tag{12}$$

Because all the point pairs are accurate matched, we can solve the linear system by applying least-square method. The solution is simply as:

$$\mathbf{v} = (\mathbf{A^T A})^{-1} \mathbf{A^T b}' \tag{13}$$

B. Compensation

Obviously, the parameters in similarity model (a, b, c, d) are not enough to compute the parameters in real motion model $(\gamma, \theta, x_0, y_0, d_x, d_y)$. Face to this situation, Xu Lidong [4] has neglected the rotation center and simplified and to the translation parameters, Hwy Kuen Kwak [15] has used a

TABLE I
PARAMETERS OF THE MOTION MODEL

Parameters	a	b	c	d	$\theta(^\circ)$	γ
Values	0.99	-0.04	-4.41	7.23	-2.33	1.00

smart method to estimate the rotation center and angle. But in our scheme, we directly compute the parameters through (7) to (10) on the premise that the rotation center is the image center.

In process of image compensation, we fill the undefined areas using reference frame in order to reduce time.

IV. EXPERIMENTAL RESULTS

A. Two Consecutive Images

To show the effectiveness of our DIS solution, simulation results are illustrated for two consecutive images. The reference and current frame (320×240) are the top-left and top-right image respectively shown in Fig.6. We stabilized the current frame to the reference frame using our method. And the parameters of similarity model are indicated in Table I. According to (7) (8), we can compute the scaling factor and rotation angle by:

$$\theta = arctan\frac{b}{a} \tag{14}$$

Their results are shown in Table I as well. Assuming the rotational center is the center of the current frame, and then we can get (d_x, d_y) is equal to $(0, 1)$ after round off from (9) and (10). The computed vector of translation is very different from (c, d), and there would be a serious error if we warp the current frame by (c, d) [4].

The stabilization result is shown in the middle of the figure 6. The bottom-left image in the figure is the difference image between the two original successive frames and the bottom-right image shows the difference between the stabilized frame and the reference frame.

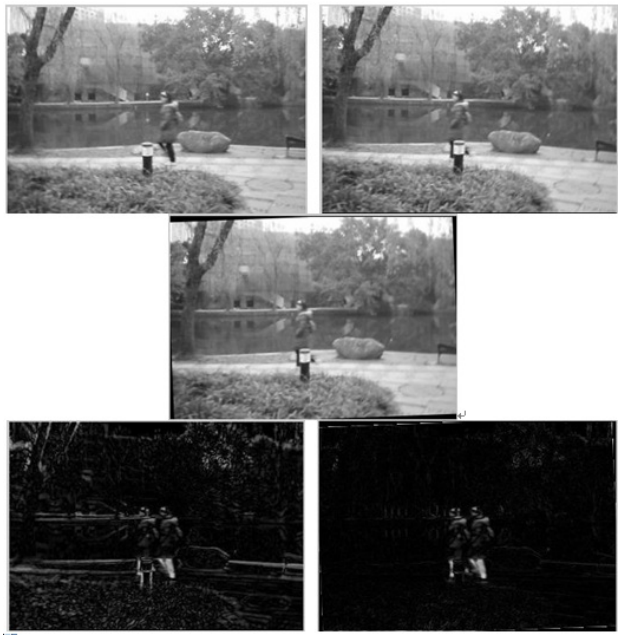


Fig. 6. Stabilization results for two real consecutive frames.

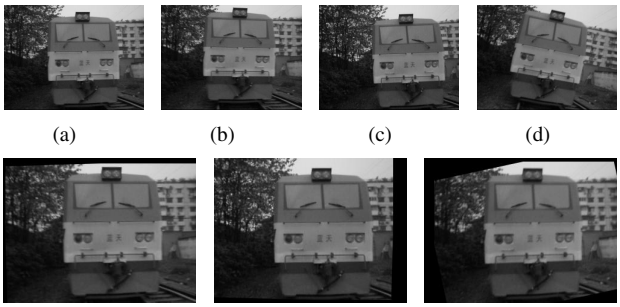


Fig. 7. The 1st video stabilization result with short focal length camera and no moving objects. (a) Reference frame. (b), (c) and (d) The 7th, 72th and 88th frame. (e), (f) and (g) The stabilized 7th, 72th, 88th frame.

B. Video Sequence

Our video is stabilized by the proposed method. To evaluate the performance, the rotational, translational and scaling range of stabilization is considered, and the fidelity between current frame and reference frame using the peak signal-to-noise ratio (PSNR) [16] is evaluated as well. The PSNR between consecutive frame I_1 and I_0 is defined as:

$$PSNR(I_1, I_0) = 10log_{10}\frac{255^2}{MSE(I_1, I_0)} \tag{15}$$

Where the mean squared error (MSE) is to measure the average departure per pixel from the desired stabilized result.

To illustrate the validity of our method in different scenes, some video frames with and without moving objects are show in Fig.8 and Fig.7. Both videos suffered more or less translation and rotation, and the camera focal length in Fig.8 is longer than that in Fig.7. Purely for visual purpose, the stabilized frames in Fig.7 are not compensated. After half hundred consecutive frames in both videos are processed by the proposed method, the scaling factor, rotational angle and translation relative to the reference frame are shown in Fig.9 and the PSNRs are shown in Fig.10. As above mentioned, the scaling factors in our video are nearby one, and the estimated result in Fig.9 (a) agrees with the practice. The results in Fig.9 (b) (c) (d) demonstrate that the presented method is able to process large rotation and translation because of the global search using CDHS algorithm. In Fig.10, the frames stabilized exhibit much higher PSNR than the original frames, and we can improve the video quality even if terrible video captured by long focal length camera with PSNR less than 12 dB.

V. CONCLUSION

DIS is a very useful technique for improving video quality. In this paper, an effective and robust digital image stabilization method based on Harris feature points is proposed. In the method, sub-region Harris detector is used to scatter the feature points. Then cross diamond hexagon search algorithm

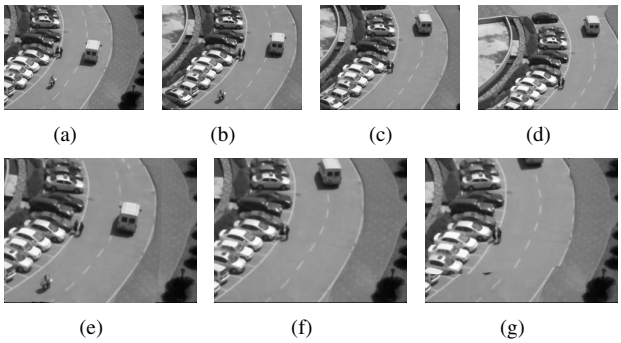


Fig. 8. The 2nd video stabilization result with long focal length camera and moving objects. (a) Reference frame. (b), (c) and (d) The 8th, 36th and 50th frame. (e), (f) and (g) The stabilized 8th, 36th, 50th frame.

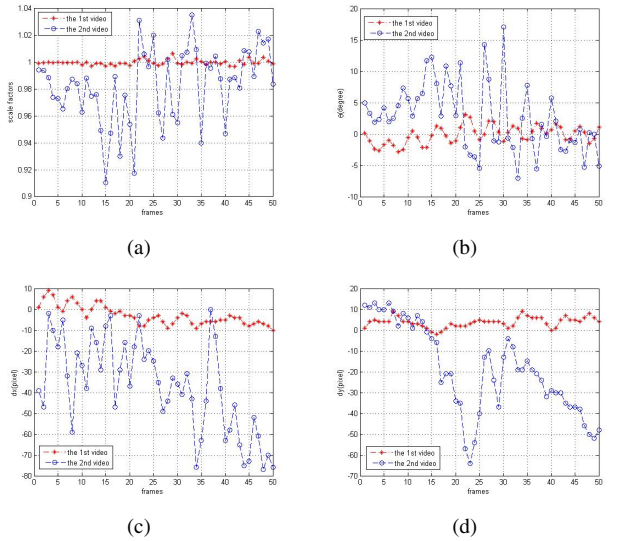


Fig. 9. Estimated parameters for the unstable real video. (a), (b), (c) and (d) Estimated scaling factors, rotational angles, horizontal translation and vertical translation.

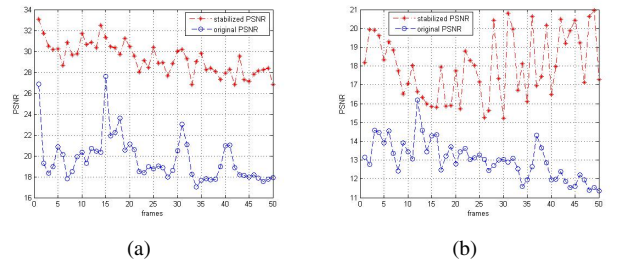


Fig. 10. PSNR comparison for real video. (a) and (b) The 1st and 2nd video.

is applied to find matching points in reference frame, and the matching points are checked by an spatial-location-invariant criterion. With the accurate local motion vectors, global motion vector is generated by LS algorithm. And motion compensation is implemented by real motion model. The experimental results shown that the proposed DIS technique can deal with large rotation, translation, and is robust to moving objects. It is believed that some robust real-time digital image stabilization algorithms would be developed with this smart spatial-location-invariant checking criterion. Limitations are very similar to Liu's [11].

REFERENCES

[1] S. B. Balakirsky and R. Chellappa, "Performance characterization of image stabilization," *Image Processing*, vol.2, pp. 413–416, 1996.

[2] Erturk and T. J. Dennis, "Image sequence stabilization based on DFT filtering," *IEE-Proceedings on Image Vision and Signal Processing*, vol. 127, pp. 95–102, 2000.

[3] F. Vella, A. Castorina and Mancuso M, "Digital image stabilization by adaptive block motion vectors filtering," *IEEE Transactions on Consumer Electronics*, vol. 48, no. 3, pp. 796–800, 2002.

[4] Xu Lidong and Lin Xinggang, "Digital image stabilization based on circular block matching," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 2, pp. 566–574, 2006.

[5] Sung-Jea Ko, et al., "Digital image stabilizing algorithms based on bit-plane matching," *IEEE Transactions on Consumer Electronics*, vol. 45, no. 3, pp. 617–622, 1998.

[6] Sung-Jea Ko, et al., "Fast Digital Image Stabilizer Based on Gray-coded Bit-plane Matching," *IEEE Transactions on Consumer Electronics*, vol. 45, no. 3, pp. 508–513, 1999.

[7] Yuan Fei, Zhang Hong and Jia Ruiming, "Digital Image Stabilization Based on Log-Polar Transform," *IEEE Fourth International Conference on Image and Graphics*, 2007.

[8] Sebastiano Battiato, et al., "SIFT Features Tracking for Video Stabilization," *IEEE 14th International Conference on Image Analysis and Processing*, 2007.

[9] Sebastiano Battiato, et al., "Fuzzy-based Motion Estimation for Video Stabilization Using SIFT Interest Points," *Proceedings of SPIE*, 2009.

[10] Jyh-Yeong Chang, et al., "Digital Image Translational and Rotational Motion Stabilization Using Optical flow Technique," *IEEE Transactions on Consumer Electronics*, vol. 48, no. 1, pp. 108–115, 2002.

[11] Feng Liu, Michael Gleicher, et al., "Subspace Video Stabilization," *ACM Transactions on Graphics (ACM SIGGRAPH 2011)*, vol. 30, 2011.

[12] Feng Liu ,Hailin Jin, et al., "Content-Preserving Warps for 3D Video Stabilization," *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)*, 2009.

[13] C. Harris and M Stephens, "A combined corner and edge detector," *Alvey Vision Conference*, 1988.

[14] Chun Ho Cheung and Lai Man Po, "Novel Cross-Diamond-Hexagonal Search Algorithms for Fast Block Motion Estimation," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 16–22, Feb. 2005.

[15] Hwy Kuen Kwak, Tae Yeon Kim and Joon Lyou, "Digital Image Stabilization subjected to Base Motions," *IEEE International Conf. on Industrial Technology*, 2006.

[16] Carlos Morimoto and Rama Chellappa, "Evaluation of image stabilization algorithms," *Proceedings of IEEE International Conf. on Acoustics, Speech and Signal Processing*, vol. 5, pp. 2789–2792, 1998.