A Digital Video Stabilization System Based on Reliable SIFT Feature Matching and Adaptive Low-Pass Filtering

Jun Yu[1], Chang-Wei Luo[1], Chen Jiang[1,2], Rui Li[1,2], Ling-Yan Li[1], and Zeng-Fu Wang[1,2(✉)]

[1] Department of Automation, University of Science and Technology of China, Hefei 230026, China
{harrtjun,zfwang}@ustc.edu.cn
[2] Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031, China

Abstract. A real-time digital video stabilization system is proposed to remove unwanted camera shakes and jitters. Firstly, SIFT algorithm is improved to extract and match features between the reference frame and current frame reliably, and then global motion parameters are obtained based on the geometric constraint consistency between feature matches through random sample consensus algorithm. Secondly, multiple evaluation criteria are fused by an adaptive low-pass filter to smooth global motion for obtaining correction vector, which is used to compensate the current frame. Finally, stabilized video is obtained after each frame is completed by combining the texture synthesis method and the spatio-temporal information of video. The objective experiments demonstrate the system can increase the average peak signal-to-noise ratio of jittered videos around 6.12 dB, The subjective experiments demonstrate the system can increase the identification ability and perceptive comfort on video content.

Keywords: Global motion estimation · Motion filtering and compensation · Video completion

1 Introduction

Videos retrieved from hand-held video cameras are affected by unwanted camera shakes and jitters, resulting in video quality loss [1]. Digital video stabilization techniques have gained consensus, for they permit to obtain high quality and stable video footages by making use only of information drawn from footage images and do not need any additional knowledge about camera physical motion [2][3].

There are three stages for digital video stabilization: global motion estimation [4], motion filtering and compensation [5], video completion [6].

Global motion estimation can be performed by global intensity alignment approaches [7-10] or feature-based approaches [11-13]. Feature-based methods are generally faster than global intensity alignment approaches, while they are more prone to local effects. A good survey on global motion estimation can be found in [4].

After estimating the global motion, motion filtering is removing the annoying irregular jitter to recognize intentional movement. It can be performed by DFT filtered

H. Zha et al. (Eds.): CCCV 2015, Part II, CCIS 547, pp. 180–189, 2015.
DOI: 10.1007/978-3-662-48570-5_18

frame position smoothing [10], Kalman filtering [14] and motion vector integration [15] according to real system constraints [16][17]. After motion filtering, motion compensation is applied to spatially displace image frames by correction vector from the filtering result.

The goal of video completion is filling in missing image areas in a video [18]. It can be performed by mosaicing [19], sampling spatio-temporal volume patches [20], multi-layers segmenting [21][22] and local motion estimation of missing image areas [23][24]. The texture synthesis method [25][26] searches the similar texture patch to replace the unknown part in the missing image area. Good result can be obtained if enough similar information are available.

In this paper, a digital video stabilization system is proposed (Fig. 1). Our work has following advantages: 1) To increase the reliability of invariant feature transform (SIFT) algorithm, non-maximum suppression is used to obtain evenly distributed feature points, and multi-objective optimization is used to improve the feature matching accuracy. 2) Feature matches are used to estimate global motion by random sample consensus (RANSAC) fitting. 3) An adaptive low-pass filter with adaptive length according to the variation of global motion is constructed, thus over stabilization and under stabilization are prevented effectively. 4) Multiple evaluation criteria are fused to increase the robustness of motion filtering and compensation. 5) The performance of image texture synthesis method [25] is promoted with the plentiful spatio-tempral information of video to conduct the video completion.

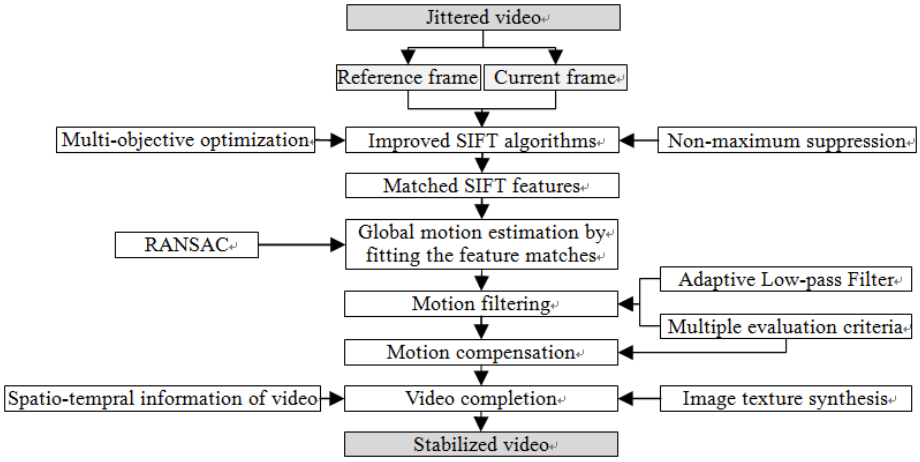


Fig. 1. Framework.

2 Image Matching

SIFT algorithm has been shown excellent performance in image matching [27]. It can be divided into three stages: feature detection, feature description and feature matching. However, there are two shortcomings of SIFT algorithm, namely non-evenly distribution of the feature points and non-adaptive feature matching strategy.

A large range, evenly distribution of feature points is a key factor ensuring the quality of image matching. In the feature detection stage of SIFT algorithm, the feature points are determined by the comparison of the extreme of the 26 surrounding pixels; thus the extreme value detected represents 27 pixels of the feature points, which are likely to fall into the local extremes. The spatial distribution of the detected feature points tend to be concentrated within a certain range, and the feature points may reflect only one or a few objects characteristics of the image. Whereas the required feature points should be able to reflect the overall characteristics of image, not just some local characteristics. In order to obtain a more uniform distribution of feature points, a large detection range should be considered. The reason is as follows. The greater the detection range, the greater the range of the local extremes represented by feature points. When the feature points are treated as local extremes in a wider range, the distances between them are more distant and a more uniform distribution of feature points will be obtained. Therefore, a detecting feature method with the non-maximal suppression [28] is used. There are $(2\times r+1)\times 2-1+18$ pixels used to detect extremes with the radius of r at the current scale and 18 pixels at the adjacent scales, not only 26 pixels. In the original SIFT algorithm, feature detection is to compare feature points with 8 pixel at the current scale and 18 pixels at the neighbors and scales, while the used detecting feature method [28] is to compare feature points with 48 pixels at the current scale and 18 pixels at the neighbors and scales. Although the number of feature points detected by the used method is reduced, the features are distributed on a wider scope.

Because the SIFT feature points are disorder, and are not described regularly, such as corner, straight line, edge. So the normal image matching technology, such as relevant matching, is hard to achieve high accuracy. So the multi-objective optimization theory [29] is introduced into SIFT feature matching to reduce the error matching ratio, which consider Euclidean distance between correlation coefficient and feature point as the objective function and the confidence degree is taken as the constraint. The optimization purpose is to select the most satisfactory scheme from many alternative ones according to a more than one objective. The advantage of multi-objective optimization is that we can regulate the trade-off problem among multi-objectives to make them realize optimization at the same time under some certain constraint conditions. The details refer to [29].

As a result, the result of image matching is a list of keypoints pairs that can be easily used as the input of feature-based motion estimation stage.

3 Global Motion Estimation

Based on the perspective projection imaging model, the global motion, associating feature $\left(x_i, y_i\right)^T$ in frame I_n with feature $\left(x_j, y_j\right)^T$ in frame I_{n+1}, is described by:

$$\begin{aligned} x_j &= \left(a_1 x_i + a_2 y_i + a_3\right) / \left(a_7 x_i + a_8 y_i + 1\right) \\ y_j &= \left(a_4 x_i + a_5 y_i + a_6\right) / \left(a_7 x_i + a_8 y_i + 1\right) \end{aligned} \tag{1}$$

where $(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)$ are the parameters to be solved.

The whole set of feature matches probably includes wrong matches or correct matches that indeed belong to self-moving objects in the filmed scene. Here RANSAC [30] is used to deal with this problem. Firstly, six couples of features are selected randomly from the feature set, and a solution is obtained from them. Then a subset of feature set is obtained by $\boldsymbol{S}_1^* = \left\{ \left((x_i, y_i)^T ; (x_j, y_j)^T \right) \middle\| \begin{matrix} x_j - (a_1 x_i + a_2 y_i + a_3)/(a_7 x_i + a_8 y_i + 1) \\ y_j - (a_4 x_i + a_5 y_i + a_6)/(a_7 x_i + a_8 y_i + 1) \end{matrix} \right\| \le T \right\}$, T is a given threshold. Secondly, above process is repeated K times [13], and the subset with the most elements is selected. Finally, LM method is applied on the selected subset to obtain the final solution.

4 Motion Filtering and Compensation

An adaptive low-pass filter and multiple evaluation criteria are applied in the motion filtering. The following low-pass filter is used:

$$h(t) = \begin{cases} \sin(2\pi t/N - 1)/(2\pi t/N - 1) & -(N-1)/2 \le t \le (N-1)/2 \\ 0 & other \end{cases} \tag{2}$$

where N is the length of filter. To make N be adjusted according to the variation of global motion parameters, N is initialized as 5 manually after experiments, then following indices are computed:

Cumulative variation of gloabal motion parameters:

$$S = \sum_{i=1}^{N} \left| \boldsymbol{M}_i - \boldsymbol{M}_{ave} \right|, \boldsymbol{M}_{ave} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{M}_i \tag{3}$$

Max variation of gloabal motion parameters:

$$\rho = \max \left\{ \left| \boldsymbol{M}_i - \boldsymbol{M}_{ave} \right|, i = 1, 2, \cdots, N \right\} / S \tag{4}$$

Smoothness of gloabal motion parameters:

$$\lambda = S / \boldsymbol{M}_{ave} \tag{5}$$

Then a max threshold of λ (*threshold 1*) and a min threshold of ρ (*threshold 2*) are determined manually after experiments. Finally, N is adjusted online during stabilization process: if λ is smaller than *threshold 1*, and ρ is smaller than *threshold 2*, N is increased. Otherwise, N is decreased.

Firstly, the estimated global motion parameters $\boldsymbol{M}$ is chosen as one criterion to evaluate the video jitter. The adaptive low-pass filter is applied on $\boldsymbol{M}$, and the smoothing components are set as the motion filtering result. However, we found the motion filtering result of $\boldsymbol{M}$ is not satisfying when the jitter has very frequent tiny

rotation component. The reason is: when the rotation component is very tiny, the filtering result is almost same to the original value, thus the compensation effect is very limited, and the human visual system still feel jittery when watching the compensated result. To alleviate this problem, the Euclidean distance of matched keypoints (EDMK) between adjacent frames is used as the second criterion, and the adaptive low-pass filter is also used to smooth the x component and y component of EDMK. Finally, the average of the filtering results by both criteria is set as the final result.

After motion filtering, the motion compensation is conducted as follow:

Firstly, the correction vector for the first criterion is obtained by computing the difference between original parameters $\boldsymbol{M}$ and filtering parameters $\hat{\boldsymbol{M}}$. Then the motion compensation is applied according to the equation (1). The only difference is (x, y) is the pixel position.

Secondly, the correction vector for the second criterion is obtained by computing the difference between original EDMK and filtering EDMK. Then the motion compensation is applied by displacing the pixel according to the correction vector.

Finally, the coordinates of pixels are set as the average coordinates of the pixels compensated by the first evaluation criterion and the pixels compensated by the second evaluation criterion.

5 Video Completion

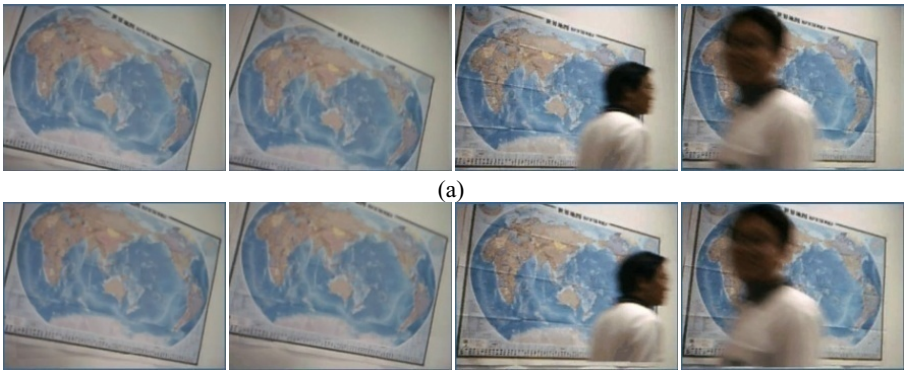
Good result can be obtained by the texture synthesis method [25][26] if enough similar information are available. However, it is hard to obtain satisfying result only by this method because the similar information in the single image is usually not enough. The texture synthesis method can be improved if the plentiful interframe information of video is introduced. The way of combining them is: the most similar texture patch of an original texture patch $\boldsymbol{A}$ in the current frame is searched in the adjacent frames by the texture synthesis method. If it is found, and the found texture patch is $\boldsymbol{B}$ in the adjacent frames, the neighbor texture patch of $\boldsymbol{B}$ will have the high priority to be the most similar texture patch of the neighbor texture patch of $\boldsymbol{A}$ during searching. If it is not found, it is searched in the current frame by the texture synthesis method.

6 Experiments

Experiments are conducted using a workstation with AMD Athlon (tm) II X4 640 3.01G, memory 2G, NVIDIA GT200 and CUDA 1.3.

Two jittered videos are captured [31]. The first is the video without moving object, and has 2476 frames, while the second is the video with moving object, and has 3124 frames. The GPU+CPU framework [32] is used to achieve the real-time ability. Because the global motion estimation, motion filtering and compensation need large computation, they are implemented in GPU, while other parts are implemented in CPU. In addition, the GPU implement of SIFT [33] is used to accelerate the process of feature extraction.

Fig. 2 shows the video stabilization results on the captured videos.



(a)

(b)

Fig. 2. (a) The frames before video stabilization. (b) The frames after video stabilization.

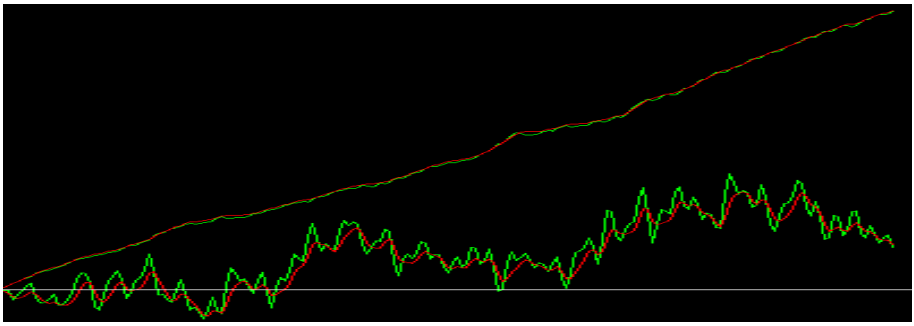


Fig. 3. Green curves are a_3, a_6 of the original video, red curves are a_3, a_6 of the stabilized video.

Fig. 3 is the motion filtering results of $\boldsymbol{M}$ by the adaptive low-pass filter. It shows the adaptive smoothing effect of the filter.

An index, peak signal-to-noise ratio (*PSNR*) between the reference frame $\boldsymbol{S}_0$ and current frame $\boldsymbol{S}_1$, is defined to evaluate the stabilization quality:

$$PSNR(\boldsymbol{S}_1, \boldsymbol{S}_0) = 10 \cdot \log_{10}^{255^2 / MSE(\boldsymbol{S}_1, \boldsymbol{S}_0)} \tag{6}$$

where *MSE* is the mean square error of pixel value between two images. This index reflects the coherence between two images. The large the index, the better the video stabilization result.

Table 1 show the average *PSNR* on captured videos. As can be seen from it, the average *PSNR* is increased by the proposed video stabilization method around 6.12 dB, and the real-time ability is also achieved. Therefore, jittered video is stabilized by the proposed method nicely in real-time.

Table 1. Qualitative evaluation result of video stabilization.

	Average *PSNR* of original videos	Average *PSNR* of stabilized videos	Average time each frame takes
Captured video	25.35	31.47	0.045s

6.1 Improved SIFT Vs. Original SIFT

To evaluate the performance of the improved SIFT algorithm, experiment is conducted on different pairs of images from a standard LEAR image database [32]. Table 2 shows that the matching correct rate of improved SIFT is outperform that of original SIFT. From it, we can see the effectiveness of the improvements on the distribution of feature points and the feature matching strategy.

Table 2. Comparison of the matching correct rate between improved SIFT and original SIFT.

	Original SIFT	Improved SIFT
LEAR image database	84.56%	89.67%

6.2 Using Single Evaluation Criterion Vs. Fusing Multiple Evaluation Criteria

The effect of fusing multiple evaluation criteria is verified on a video clip, in which some very frequent tiny rotation component is added. From Table 3, we can see the superiority of fusing multiple evaluation criteria.

Table 3. Evaluation between single evaluation criterion and multiple evaluation criteria.

	Average *PSNR* of original videos	Average *PSNR* of stabilized videos	Average time each frame takes
Single evaluation criterion	18.56	23.67	0.037s
Multiple evaluation criteria	18.56	24.56	0.045s

6.3 Objective Comparison with Other Algorithm

The method in [24] is one of the state-of-the-art video stabilization methods. We have implemented it, then it and the proposed method are tested on the above video clip. We can see the proposed method is superior to the method in [24] from Table 4. This is because the proposed method fuses multiple evaluation criteria to conduct motion filtering by an adaptive low-pass filter, and the SIFT algorithm is improved to extract and match features between the reference frame and current frame reliably.

Table 4. Evaluation of several video stabilization algorithms.

	Average *PSNR* of original videos	Average *PSNR* of stabilized videos	Average time each frame takes
The proposed method	18.56	24.63	0.045s
The method in [24]	18.56	24.21	0.053s

6.4 Subjective Comparison with Other Algorithm

The problem with an objective evaluation is that the absolute truth of camera motion is not known. However, it is less problematic for the subjective evaluation since the human visual system is very sensitive to the video jitter. Therefore, user's reactions interacting with this system are evaluated.

34 users participate in the evaluation. The goal of the evaluation is to decide if the system can remove the discomfort on human visual system, and if the objects in the stabilized video can be identified easily.

In the first stage, the questionnaire is chosen for participants. Table 5 shows the constructs and questions of the survey related to the system performance. The answers to these questions are given from 'disagree' to 'agree' on a ten point scale. A Cronbach's alpha test [34] is carried out to determine if these constructs refer to the same topic. Typically, an alpha of 0.7 or greater is considered acceptable in psychological experiments. As Table 5 shows, all the alpha values obtained are greater than 0.7, indicating that the questionnaire is suitable for the evaluation in this paper.

Table 5. Cronbach's alpha results of questionnaire and mean scores after evaluation.

Construct	Question	Cronbach's alpha	Mean score of the proposed method	Mean score of the method in [24]
Smoothness	If the stabilized video is smooth and coherent.	0.743	7.79	6.73
Identification	If objects in the stabilized video can be identified easily.	0.811	7.75	6.48

In the second stage, the developed system and the method in [24] perform stabilization on captured videos, then participants compare stabilized videos with original videos. Finally, the questionnaire is filled. Table 5 shows the result of mean scores after evaluation. The maximum is 10, while the minimum is 0. For the developed system, all the scores obtained are greater than 7.5, and are higher than those of the method in [24], indicating that it has the ability to remove the discomfort on human visual system, and the objects in the stabilized video can be identified easily.

7 Conclusion

A real-time, reliable and adaptive digital video stabilization system is proposed. The SIFT algorithm is improved to match the adjacent frames robustly. Global motion parameters are obtained by RANSAC effectively. Multiple evaluation criteria are fused to conduct motion filtering by an adaptive low-pass filter. The spatio-temporal information are combined with the texture synthesis method to obtain a complete video. In future, the accuracy of motion estimation will be further improved.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. 61303150), the Open Project Program of the State Key Lab of CAD&CG (No. A1501), Zhejiang University.

References

1. Ejaz, N., Wonil, K., Soon II, K., et al.: Video stabilization by detecting intentional and unintentional camera motions. In: ICISMS, pp. 312–316. IEEE Press, New York (2012)
2. Chen, C.H., Chen, C.Y., Chen, C.H., et al.: Real-Time Video Stabilization Based on Vibration Compensation By Using Feature Block. IJICIC **7**, 5285–5298 (2011)
3. Seok-Jae, K., Tae-Shick, W., Dae-Hwan, K., et al.: Video stabilization based on motion segmentation. In: ICCE, pp. 416–417. IEEE Press, New York (2012)
4. Dung, T.V., Lertrattanapanich, S., et al.: Real time video stabilization with reduced temporal mismatch and low frame buffer. In: ICCE, pp. 61–62. IEEE Press, New York (2012)
5. Puglisi, G., Battiato, S.: A Robust Image Alignment Algorithm for Video Stabilization Purposes. TCSVT **21**, 1390–1400 (2011)
6. Puglisi, G., Battiato, S.: Robust video stabilization approach based on a voting strategy. In: ICIP, pp. 629–632. IEEE Press, New York (2011)
7. Abraham, S.C., Thomas, M.R., Basheer, R., et al.: A novel approach for video stabilization. IEEE Recent Advances in Intelligent Computational Systems **1**, 134–137 (2011)
8. Ko, S.J., Lee, S.H., Lee, K.H.: Digital image stabilizing algorithms based on bit-plane matching. TCE **44**, 617–622 (1998)
9. Ko, S.J., Lee, S.H., Jeon, S.W., Kang, E.S.: Fast digital image stabilizer based on gray-coded bit-plane matching. TCE **45**, 598–603 (1999)
10. Erturk, S., Dennis, T.J.: Image sequence stabilization based on DFT filtering. IEE Proceedings on Image Vision and Signal Processing **127**, 95–102 (2000)
11. Bosco, A., Bruna, A., Battiato, S., Bella, G.D.: Video stabilization through dynamic analysis of frames signatures. In: ICCE, pp. 312–316. IEEE Press, New York (2006)
12. Veon, K.L., Mahoor, M.H., Voyles, R.M.: Video stabilization using SIFT-ME features and fuzzy clustering. In: IEEE/RSJ ICIRS, pp. 2377–2382. IEEE Press, New York (2011)
13. Windau, J., Itti, L.: Multilayer real-time video image stabilization. In: IEEE/RSJ ICIRS, pp. 2397–2402. IEEE Press, New York (2011)
14. Erturk, S.: Image sequence stabilization based on kalman filtering of frame positions. Electronics Letters **37**, 95–102 (2001)
15. Paik, P.: An adaptative motion decision system for digital image stabilizer based on edge pattern matching. Consumer Electronics, Digest of Technical Papers (1992)
16. Auberger, S., Miro, C.: Digital video stabilization architecture for low cost devices. In: ISISPA, pp. 474–483. IEEE Press, New York (2005)

17. Tico, M., Vehvilainen, M.: Constraint translational and rotational motion filtering for video stabilization. In: ESPC, pp. 1474–1483. IEEE Press, New York (2005)
18. Zhiyong, H., Fazhi, H., Xiantao, C., et al.: A 2D-3D hybrid approach to video stabilization. In: ICCADCG, pp. 146–150. IEEE Press, New York (2011)
19. Litvin, A., Konrad, J., Karl, W.: Probabilistic video stabilization using kalman filtering and mosaicking. In: IS&T/SPIE SEIIVC, pp. 663–674. IEEE Press, New York (2003)
20. Wexler, Y., Shechtman, E., Irani, M.: Space-time video completion. In: CVPR, pp. 120–127. IEEE Press, New York (2004)
21. Jia, J., Wu, T., Tai, Y., Tang, C.: Video repairing: inference of foreground and background under severe occlusion. In: Proc. CVPR, pp. 364–371. IEEE Press, New York (2004)
22. Cheung, S.C.S., Zhao, J., Venkatesh M.V.: Efficient object-based video inpainting. In: ICIP, pp. 705–708. IEEE Press, New York (2006)
23. Cheung, V., et al.: Video epitomes. In: CVPR, pp. 42–49. IEEE Press, New York (2005)
24. Matsushita, Y., Ofek, E., Ge, W.N., et al.: Full-frame video stabilization with motion inpainting. TPAMI **28**, 1150–1163 (2006)
25. Criminisi, A., Perez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. TIP **13**, 1200–1212 (2004)
26. Tang, F., Ying, Y.T., Wang, J., et al.: A novel texture synthesis based algorithm for object removal in photographs. In: ACSC, pp. 248–258. IEEE Press, New York (2005)
27. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV **60**, 91–110 (2004)
28. Gao, T., et al.: Multi-Scale Image Registration Algorithm based on Improved SIFT. Journal of Multimedia **8**, 755–761 (2013)
29. Zheng, Y., et al.: Video Image Tracing Based on Improved SIFT Feature Matching Algorithm. Journal of Multimedia **9**, 130–137 (2014)
30. Hoper, P.J.: Robust statistical procedures. SIAM (1996)
31. Yu, J., Luo, C.-w., Jiang, C., Li, R., Li, L.-y., Wang, Z.-f.: Real-time robust video stabilization based on empirical mode decomposition and multiple evaluation criteria. In: Zhang, Y.-J. (ed.) ICIG 2015. LNCS, vol. 9219, pp. 125–136. Springer, Heidelberg (2015)
32. Juang, C., et al.: Speedup of implementing fuzzy neural networks with high-dimensional inputs through parallel processing on graphic processing units. TFS **19**, 717–728 (2011)
33. http://cs.unc.edu/~ccwu/siftgpu/
34. Marcosa, S., Gómez-García-Bermejob, J., Zalama, E.: A realistic, virtual head for human-computer interaction. Interacting with Computers **22**, 176–192 (2010)