International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 10, October 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

REAL TIME VIDEO STABILIZATION USING MATLAB

Sarang Shirole[1], Prof. A. B. Deshmukh[2]

[1,2]*Department of Electronics and Telecommunication, SKNCOE, SPPU Pune, (India)*

ABSTRACT

Video processing is broadly used in numerous areas such as space, health, auto industry, city planning and military where exact image frames are required. For example, in a surgical operation where cameras are used, the operator needs real time video which is stable to recognize the exact position of the problem. In a military application where object tracking is the task, successive frames should be stable in the spatial domain, so that tracking algorithm can work appropriately. In today's world, there are many applications which are working on video. Particularly, video from camera's which are placed on moving platform. But due to uneven surface or camera vibration, captured image tends to have undesired jitters, shakes and blurs. This results into unlikable viewing experience and also affect on image processing application such as surveillance or some military application also. Digital video stabilization is an essential video improvement technology which is focusing on removing unnecessary camera vibrations from image sequences. In the proposed method, first feature points are detected with the Features from Accelerated Segment Test (FAST) corner detection algorithm and then these feature points are extracted using efficient Fast Retina Key point (FREAK) descriptor. Then based on hamming distance feature points are matched between consecutive frames. Next, the matched point pairs are fitted to the affine transformation model using a M-estimator SAmple Consensus (MSAC) algorithm, which is a variant of the RANSAC algorithm based approach to estimate inter-frame motion strongly. Then the calculated results are used to find out the cumulative motion parameters between the conjugative frames, and the translational components are smoothened. Experimental results have shown that the proposed system can deal with standard stored and real time image/video input including arbitrate translation and rotation noise and can produce full-frame stabilized output providing a better viewing experience. Experimental results have shown that proposed MATLAB algorithm can deal with stored standard videos, real time images and videos.

Keywords: Feature detection, Feature extraction, MATLAB, motion estimation, video stabilization.

I. INTRODUCTION

Over the past few years video cameras placed on moving platform, handheld devices and mobile platforms such as mobile robots, UAV etc have become more and more popular in the consumer market due to a remarkable decrease in the cost of such devices. Also, Mobile robots are used in many applications such as search and rescue tasks. But due to cameras mounted on moving platform, results into undesired jitters making captured images blurred which affects the viewing experience and video processing applications such as video encoding and video surveillance. Camera motion and platform vibrations can be difficult to avoid when using mobile robots, handheld cameras, which will generate unstable images. Video stabilization is, therefore, becoming an essential technique in mobile robot technology, advanced digital cameras and camcorders.

Camera motion estimation is a vital step in video stabilization. Stabilization methods utilize the fact that camera motion causes the affine transform of the frames [3]. Therefore, the first step of video stabilization is, to identify the global affine transformation. However, it is very important to have correct motion estimation techniques. Also the robustness of the estimation is significant due to the fact that an incorrect estimate will results into a sudden jitter in the image sequence. In particular, the estimation should be performed consistently in the presence of outliers, blurring, or illumination variance.

In many cases, user intentionally moves camera to get desired videos. Hence, it is important to preserve the intentional camera motion while removing the undesired motion due to an unsteady and vibrating platform. Combination of camera motion estimation with motion separation determines the undesired motion, which is compensated and thus stable image sequences are produced.

II. RELATED WORK

Video stabilization techniques have attracted great interests in recent years. With handheld cameras, camera motion and platform vibrations are difficult to be avoided, which generates unstable image sequences. Therefore, video stabilization becomes an indispensable technique in advanced digital cameras and camcorders.

There are mainly three stabilization techniques. The electronic video stabilization, optical video stabilization and digital video stabilization. In electronic video stabilizer (EVS) [1], motion sensors are used to detect the camera movement for the motion compensation and hence stabilizing the image sequence. Whereas an optical video stabilizer (OVS) [2], stabilizes the recorded images by varying the optical path to the sensor. Digital image processing techniques are used to remove the undesired motion effects to generate a compensated image sequence. This method is called Digital video stabilization (DVS) [3]. The DVS system is generally composed of two processes: motion estimation and motion compensation. Motion estimation is to estimate the reliable camera motion through three processing steps on input video sequences:

1) Evaluate local motion vectors
2) Estimate the global motion vector including the intentional and the unintentional motion vectors
3) Estimate the unintentional motion. Motion compensation is to compensate the unintentional motion.

Digital video stabilization system is classified in two main categories. Direct method or block based method and feature based method. In block based algorithm image is divided into blocks and these blocks are matches with reference frames to obtain local motion vector. From this local motion vector global motion is estimated. Generally, these algorithms use fewer computations, but results' accuracy is moderated. While feature based approaches required high computations, but provides accurate results.

III. MOTIVATION

A various video stabilization techniques have been developed in recent years [5] [6] [7]. These techniques generally produce better stabilization results, but very few techniques can achieve real-time performance for practical video stabilization in portable applications because of their high computational complexity. Some of the techniques can achieve good performance but they are really slow. U-SURF [8] is a Speeded Up Robust Feature which used in video stabilization. It achieves good speed but it can only deal with up-down distortion, limiting its general applications. Like this there are various techniques available, but they are only applicable to videos with translational jitter, which is insufficient in most applications where rotation and scaling must be considered. The correction range is also fixed due to a limited search area. So we need a system which deals with above problems. The proposed system in this paper differs from the above ones in few respects. 1) The feature-based method has been used to estimate the affine parameters between consecutive frames, which can deal with not only translational but also rotational and scaling jitter in the video. 2) The proposed system can handle large-scale shake over an unlimited search area.

IV. PROBLEM DEFINITION

Implement a robust real time video stabilization system and compare the results with other stabilization methods.

V. SYSTEM OVERVIEW

The real time images and stored videos with translational and rotational jitter provides to the stabilization system. The system executes the stabilization algorithm and stabilizes each frame. The proposed stabilization algorithm consists of five major procedures: feature detection, feature description, feature matching, motion estimation, and motion filtering.

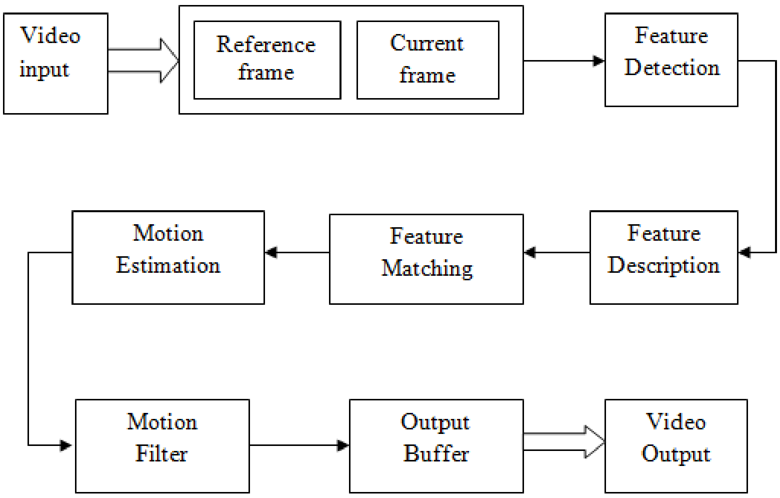


Fig 1: Block diagram of video stabilization system

Video input: The first block is input block which support different kind image format and video format like .avi, .mp4, .jpg based on operating system installed in your computer [9].

Reference and Current frame: The first frame of video is considered as reference frame which is ideally stable. Second frame of the video is considered as current frame which can contain noise or undesired motion effect. This procedure is in loop where each frame is current frame and their previous frame is the reference frame.

Feature detection: In this step, the corners of each frame are detected using Features from accelerated segment test (FAST) corner detector. The FAST detector determines whether a pixel is a corner by comparing its intensity value with those of surrounding pixels in a circular ring [3].

Feature description: once the corner points are detected, it is necessary to get descriptor for each corner point, for that we have a novel key point descriptor motivated by the human visual system and more particularly the retina, named as Fast Retina Key point (FREAK). By efficiently comparing image intensities over a retinal sampling pattern, a cascade of binary strings is calculated [11].

Feature matching: The matching cost we use between points is the Hamming distance since FREAK descriptors are binary. Points in frame A and frame B are matched putatively. Note that there is no uniqueness constraint, so points from frame B can correspond to multiple points in frame A. Matched point pairs between adjacent frames are detected based on the similarity of the detected corners. For each detected corner in the current frame, the candidate match with the minimum distance in the previous frame can be found. In information theory, the Hamming distance between two equal length strings is the total number of positions by which the subsequent symbols are different. In other words, it is minimum number of substitutions necessary to change one string into another, or the minimum number of errors that could have transformed one string into the other. For binary strings A and B the Hamming distance is equal to the number of one's (population count) in A XOR B [12].

Motion Estimation: Many of the point correspondences obtained in the previous step are incorrect. But we can still derive a robust estimate of the geometric transform between the two images using the M-estimator SAmple Consensus (MSAC) algorithm [3], which is a variant of the RANSAC algorithm. This algorithm, when given a set of point correspondences, will search for the valid inlier correspondences. From these it will then derive the affine transform that makes the inliers from the first set of points match most closely with the inliers from the second set. This affine transform will be a 3-by-3 matrix of the form [13]:

$$\begin{bmatrix} a\_1 & a\_3 & 0; \\ a\_2 & a\_4 & 0; \\ t\_x & t\_y & 1 \end{bmatrix}$$

The parameters *a* define scale, rotation, and shearing effects of the transform, while the parameters *t* are translation parameters.

Motion filtering: In this step, image is wrapped based on affine transformation calculated in the previous step [3].

VI. EXPERIMENTAL RESULTS

Proposed algorithm is tested on a laptop with an Intel i5 processor with 2.4 GHz frequency and 4 GB (3.97 GB usable) RAM. Machine runs on 64 bit operating system. The code for proposed algorithm is run on MATLAB R2016a. Proposed algorithm is tested for different types of videos and real time images with different types of scenarios. Some original frames of them are shown in Fig. 3. And their stabilized frames are shown in fig. 4. Their properties listed below in table I.

Table I. Properties of test video

Test video	Resolution	Frame Rate (fps)	Type
a	640x480	59	.avi
b	1280x720	29	.mp4
c	640x480	30	.avi

IJARSE
ISSN: 2319-8354

The Peak Signal-to-Noise Ratio (PSNR) is used to evaluate the effectiveness of the stabilization method. The PSNR between consecutive frames is defined as:

$$PSNR(I_1, I_0) = 10 \log \frac{{I_{max}}^2}{MSE(I_1, I_0)}$$

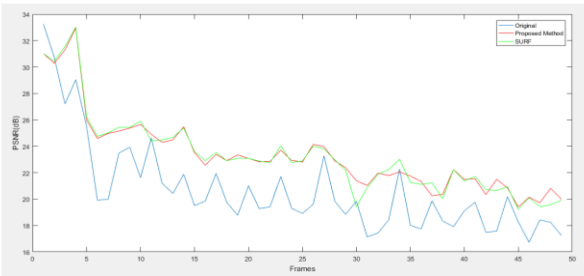
where *Imax* is the maximum intensity value of a pixel and *MSE* ( ) is the mean square error between consecutive frames, which is defined as:

$$MSE(I_1, I_0) = \frac{1}{(N \times M)} \sum_{i=1}^{N} \sum_{j=1}^{M} (I_1(i,j) - I_0(i,j))^2$$

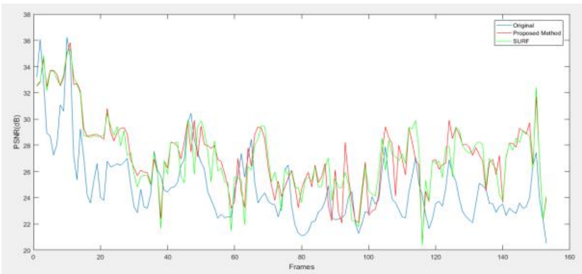
where *N* and *M* represent the frame dimensions. PSNR measures the similarity between frames. Because the stabilized sequence has more continuous frames than the original jittered one, the PSNR of the stabilized sequence should be higher than that of the original sequence.

ITF (Inter-frame transformation fidelity) is the average value of PSNR, which can be used to evaluate numerically how much a sequence has been stabilized. It is defined as:

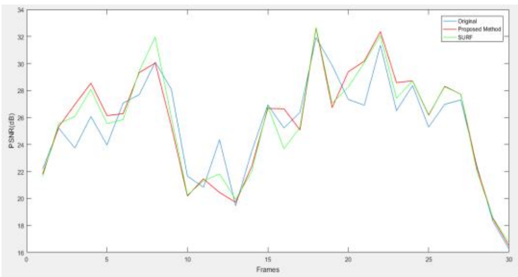
$$ITF = \frac{1}{N_{frame} - 1} \sum_{k=1}^{N_{frame}-1} PSNR(K)$$



Video (a)



Video (b)



Video (c)

Fig. 2: PSNR of original and stabilized test video sequences.

Table II, ITF of original and stabilized test video sequences

Video sequence	Original (dB)	SURF (dB)	Proposed (dB)
a	20.6908	23.3060	23.3403
b	24.8443	27.1925	27.2266
c	25.3670	25.5575	25.6732

VII. CONCLUSION

Due to cameras mounted on moving platform, results into undesired jitters making captured video blurred which affects the viewing experience and video processing applications such as video encoding and video surveillance. Hence video stabilization technique is essential in such applications. There are different video stabilization techniques available but they required more hardware and has complex computations. So we conclude that digital video stabilization technique is perfect for real time video stabilization. So in this paper a feature based stabilization method have been presented. Here we are extracting features by using FAST corner detector and FREAK feature descriptor. Then robust MSAC-based motion estimation helps to stabilize the videos with rotational jitter and arbitrate translational and also it is effective to moving objects in the foreground. To verify the effectiveness of the proposed system, we conducted the experiments on recorded video data and concluded that arbitrate translation and rotational jitters due to mobile platform are removed successfully.

VIII. FUTURE SCOPE

The proposed algorithm is implemented on MATLAB. By using parallel processor or using hardware, speed of stabilization can be increased.

REFERENCES

[1] Song, H. Zhao, W. Jing, and H. Zhu, "Robust video stabilization based on particle filtering with weighted feature points," *IEEE Trans. Consum. Electron.*, vol. 58, no. 2, pp. 570-577, 2012.

[2] https://en.wikipedia.org/wiki/Image_stabilization

[3] Jianan Li, Tingfa Xu, and Kun Zhang, "Real-Time Feature-Based Video Stabilization on FPGA". DOI 10.1109/TCSVT.2016.2515238, IEEE Transactions on Circuits and Systems for Video Technology, 2015.

[4] S. Battiato, A. R. Bruna, and G. Puglisi, "A robust block-based image/video registration approach for mobile imaging devices," IEEE Trans. Multimedia, vol. 12, no. 7, pp. 622-635, 2010.

[5] S.-K. Kim, S.-J. Kang, T.-S. Wang, and S.-J. Ko, "Feature point classification based global motion estimation for video stabilization," IEEE Trans. Consum. Electron., vol. 59, no. 1, pp. 267-272, 2013.

[6] C. Song, H. Zhao, W. Jing, and H. Zhu, "Robust video stabilization based on particle filtering with weighted feature points," *IEEE Trans. Consum. Electron.*, vol. 58, no. 2, pp. 570-577, 2012.

[7] J. Yang, D. Schonfeld, and M. Mohamed, "Robust video stabilization based on particle filter tracking of projected camera motion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, pp. 945-954, 2009.

[8] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool, "SURF: Speeded up robust features," Computer Vis. Image Understanding, vol. 110, no. 3, pp. 346– 359, 2008.

[9] https://in.mathworks.com/help/vision/ref/vision.videofilereader-system-object.html;jsessionid=24fafce827403ac5849706356c04

[10] https://en.wikipedia.org/wiki/Features_from_accelerated_segment_test

[11] Alexandre Alahi, Raphael Ortiz, Pierre Vandergheynst "FREAK: Fast Retina Keypoint" DOI 10.1109/CVPR.2012.6247715 in IEEE CONFERENCE on Computer Vision and Pattern Recognition (CVPR), 2012.

[12] https://en.wikipedia.org/wiki/Hamming_distance

[13] https://in.mathworks.com/discovery/affine-transformation.html

[14] P. Hsuao, C. Lu, and L. Fu, "Multilayered image processing for multiscale Harris corner detection in digital," IEEE Trans. Ind. Electron., vol. 57, no. 5, pp. 1799–1805, 2010.

[15] J. Zhou, H. Hu, and D. Wan, "Video stabilization and completion using two cameras," IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 12, pp. 1879-1889, 2011.

[16] Tahiyah Nou Shene, K. Sridharan, Senior Member, IEEE, and N. Sudha, Senior Member, IEEE. "Real-Time SURF-Based Video Stabilization System for an FPGA-Driven Mobile Robot", IEEE Transactions On Industrial Electronics, 2015

[17] C. Morimoto, and R. Chellappa, "Evaluation of image stabilization algorithms," in *Proc. ICASSP*, 1998, pp. 2789-2792.

[18] G. Puglisi, and S. Battiato, "A robust image alignment algorithm for video stabilization purposes," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 10, pp. 1390-1400, 2011.

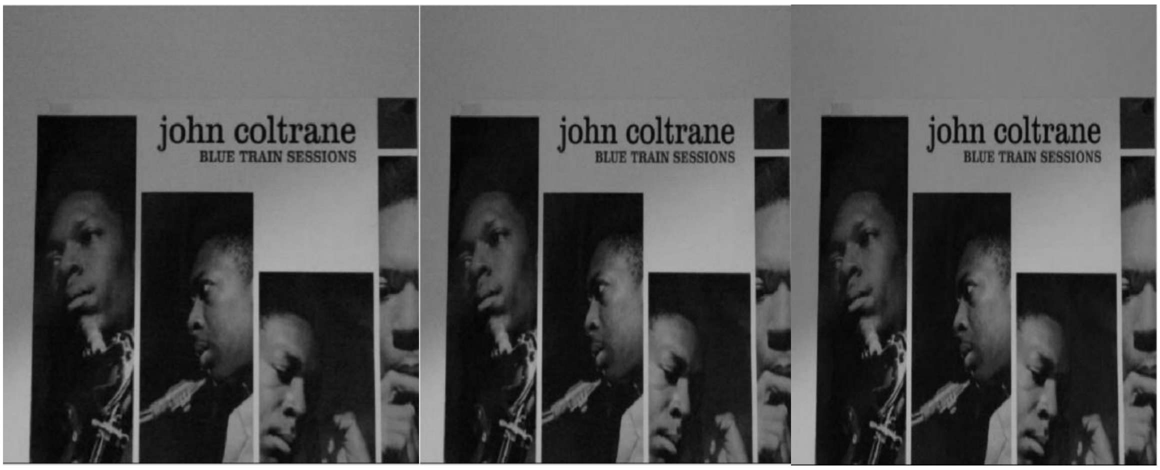
[19] C.-K. Liang, Y.-C. Peng, H.-A. Chang, C.-C. Su, and H. Chen, "The effect of digital image stabilization on coding performance [video coding]," in *Proc. ISIMP*, 2004, pp. 402-405.

[20] C. Morimoto, and R. Chellappa, "Evaluation of image stabilization algorithms," in *Proc. ICASSP*, 1998, pp. 2789-2792.

[21] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Proc. ICCV*, 2011, pp. 2564-2571.

[22] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 105-119, 2010.

[23] Sahin Isik and Kemal Ozkan, " A Comparative Evaluation of Well-known Feature Detectors and Descriptors" International Journal of Applied Mathematics, Electronics and Computers ISSN: 2147-82282 Accepted on 15th August 2014.



Video (a)

Video (b)

Video (c)

Fig .3: Frames of test video sequences

Video (a)

Video (b)

Video (c)

Fig. 4: Stabilized frames of test video sequences