IFERP

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE)
Vol 3, Issue 7, July 2016

Video Stabilization using Block Based SIFT Triangulation without Accumulative Global Motion Estimation

[1] Asst. Prof. G. Balachandaran [2] G. Pavithra, [3]V. Ranjani [4]R. Niveditha
[1][2] Department of Electronics and Communication Engineering,
Jeppiaar Engineering College, Chennai.

Abstract: **In Feature Based Motion estimation, the most time consuming process is the key point extraction (features) and their similarity measurement. In our paper, we use SIFT based feature extraction in adjacent frames. Based on similarity measurement, few features are selected. The selected features are used to calculate the motion vectors or Global motion between two frames. The computation complexity is higher if we use the whole image to extract points. The complexity of key point extraction is proportional to the area of the images (frames) and the complexity of similarity measurement is proportional to the square of the area. In our paper, we propose block based feature extraction. Using blocks for feature matching, the number of key points is reduced and hence accuracy might be decreased. We use four blocks in the four corners of the frame and use SIFT to extract key points and measure similarity. Now we have several reliable matches from each block, yet interference of moving objects may cause error in global motion estimation. This error can be removed by applying inter block feature matching. Randomly select three points (each from different block) and calculate the area formed by the triangle in both the images. This matching will give us features which are not part of local motion caused by moving objects. Now we have the best matched features obtained from SIFT. Feature matching is very important and crucial step in motion estimation. Mismatch is always inevitable in motion estimation for feature based methods no matter how powerful the feature point locator is. To solve this problem, we propose a novel idea of forming a triangle based on three best features. The pixel values of the triangle formed in two images is accumulated and matched. This matching ensures the removal of faulty features. Though theoretically this may sound good, practically this might not work well if the images are blurred. To avoid this motion de-blurring of frames can be done. Now we have best matched features, using that we calculate the global motion vectors. These global motion vectors are smoothed by using filter. The smoothened motion vectors are used to stabilize the video.**

I. INTRODUCTION

A hand held camera acquires image sequences that are affected by both desired and undesired motion. The latter motions are often irregular and unsmooth. In case of hand held camera, the intention may have been to pan the camera smoothly, but the hand holding the camera may have been unsteady. Videos captured by hand leads visible frame to frame jitter and perceived as “shaky”, which is not enjoyable to watch. The jerky image sequences also have negative impact on image sequence encoding efficiency which decreases the quality of image sequences. Video stabilization is, therefore becoming an essential technique that is used in advanced digital cameras and camcorders. It is defined as the elimination of undesired motions to remove shaking from hand held cameras. It is an important video enhancement technology which aims at removing shaky motion from videos. The implementation of video stabilization algorithms has to be cheap and use less memory. Digital Image Stabilization (DIS) meets these demands as it uses the image stream for stabilization and therefore does not need any additional equipment Digital Image Stabilization (DIS) system can be divided into three modules Motion estimation, detection of unwanted movements and motion compensation. The main goal is to compensate the unwanted shaking movements without affecting moving objects. By calculating a smooth motion close to the actual motion of the device by employing image processing, much of this shaking in videos can be reduced.

This research work describes SIFT feature matching to estimate the inter frame motion to account for Global Motion Estimation and Motion smoothing is performed to smooth the GMV used for stabilizing the video. While the motion estimation is achieved by selecting the reliable features from four blocks and triangulation method is adapted to find the best feature matches and smoothing is done by adaptive fuzzy filtering and kalman filtering. Feature matching algorithm is used in our thesis as it produces accurate results and less computational load. The developed algorithm provides a fast and robust stabilization and alters real time performance. Results show that our video stabilization using SIFT feature matching is superior in stabilizing the shaky videos in terms of accuracy and efficiency.

II EXPERIMENT SETUP

The experimental setup for our algorithm is that first we download the SIFT demo programme [23] which is made available free for research purpose only. This SIFT demo version consists of compiled binary programme for finding SIFT invariant features that runs in windows and tested under Matlab version 7 which does not need any image processing tool box. The executable file for finding the SIFT features is “siftWin32.exe” which runs in a directory with two scripts 1 is for finding SIFT features and the other is for performing matching is available in the directory. We use Matlab version 7 to develop our experimentation. Now after the having the preliminary conditions needed, the set by procedure of our experimentation is given as

Feature Extraction:

A feature is something that can be tracked in an image; it may be a point or a corner. Using SIFT, feature extraction is nothing but extracting the key-points of the image in our method. Feature extraction is done by dividing the image into four different blocks in an image rather than considering a full image at a time. These boxes are located slightly to the corner of the image as the main idea is that all the global motion occurs in these areas while local motion or object motion appears in the centre. By using blocks is to decrease the number of key-points, as large number of key-points are extracted by using SIFT which rapidly increases the computational complexity which is a main drawback in SIFT. At least of 5 to 10 correct features in an image are enough for us to know the inter-frame motion between the successive frames. The key point extraction is proportional to the area and similarity measurement is proportional to the square of the area. Key-points are extracted from all the four different blocks using SIFT, to use it for similarity measurements. Number of key points is reduced to more than half by the usage of blocks for more exact matching and thus reducing the chance of interference of moving object. The feature extraction using SIFT is done.

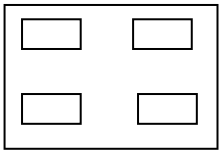

Figure1: Feature extraction from four blocks of image

III FEATURE MATCHING:

The basic idea of Feature matching is to find matching key-point from the neighbor frames. The best candidate match for each key-point is found by identifying its nearest neighbor of the key-points from the adjacent frames. SIFT implements Best-Bin-First algorithm (BBF) to search for nearest neighbor explained in . The nearest neighbor is the key-point with minimum distance for the invariant descriptor. These are the only points that have the chance of having matching descriptors. An effective measure can be obtained by comparing the distance of the closest neighbor to that of second closest neighbor. This method performs well as the correct matches need to have closest neighbor, to achieve reliable matching. For other false matches sometimes we will have the similar distance, and then we will discard the second closest match as providing an estimate of false match.In each block, we select a key-point from previous frame and find the nearest neighbor and second nearest neighbor key-point from the next frame. Calculate the distance between key-point from previous frame to that of next frame nearest neighbor and second nearest neighbor. If the distance ratio to the nearest neighbor is less than threshold value then we will select the nearest neighbor as best match to the key-point of the previous frame, else we will check the distance for the second nearest neighbor. Suppose X is a key-point from previous frame and Y is the nearest neighbor, Z is the second nearest neighbor in the next frame. The probability that Y is X’s best match if the distance ratio between X and Y is less than the threshold. If not then we will check the distance ratio of X and Z i.e. second nearest neighbor. The keypoints with more than the threshold value will be discarded to obtain reliable matching. i.e. $r(X) = r(Y)/r(Z)<$Threshold. The distance ratio of point X to point Y will be less than the threshold value. By following this process we will select the best key-points from all the blocks for matching.

Now after having the matched key points obtained from the nearest neighbor method using SIFT, we need to select the key points that helps in calculating in GMV as some features tends to represent the moving objects in the image. We only consider the key points that represents the background in the image to know the inter-frame motion as the moving objects consists of velocity which leads to errors in calculating our model parameters. To overcome the errors in calculating GMV, we use distance invariance to select the optimal matching points. Using this we will avoid the mismatch caused due to interference of moving objects. The basic idea is to randomly select three key-points from three different blocks to form a triangle. These selected key-points are the best matches obtained from our similarity measurement using SIFT. We calculate the area of triangle formed by the key-points and comparing to the area formed by key-points of same blocks in the next frame. If the area of triangle formed from the previous frame is matched to the area formed from the next frame, then we consider these key-points as best match pairs otherwise discard as mismatch may occur.

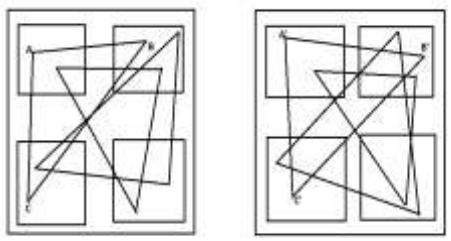

Figure.1: Triangular method

Suppose that ABC are the key-points of the triangle formed in previous frame and A'B'C' are the key-points of the triangle formed in next frame. If the area of ABC is equal to the area of A'B'C', then we consider the matched pairs (A, A') (B, B') (C, C') as best matches.

$$Area\ of\ \Delta\ ABC = Area\ of\ \Delta\ A'B'C'$$

Area of triangle used for matching the key points can be given as

$$Area = \sqrt{S(S-a)(S-b)(S-c)}$$

Where a, b and c are the distance between coordinates of the triangle and given as

$$a = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

$$b = \sqrt{(x_c - x_B)^2 + (y_C - y_B)^2}$$

$$b = \sqrt{(x_c - x_A)^2 + (y_C - y_A)^2}$$

Where (x_A, y_A) $(x_{B,}y_B)(x_{C,}y_c)$ are the coordinates of ABC respectively.

After having the matched pairs obtained from the above step we calculate the pixel difference of the matched pairs. If the pixel difference is less than the value 10 (approximately equal) then we consider these matched pairs as final best matches and discard the key points above it. The reason behind calculating the pixel difference is that the area of triangle may be equal, while the distance between the key points of the triangle formed are different which gives incorrect matching. The value 10 is selected as the images contain some blur in the frames.

$$A - A^{'} \leq 10$$

$$B - B^{'} \leq 10$$

$$C - C^{'} \leq 10$$

After calculating the pixel difference the matched pairs (A, A') (B, B') (C, C') are selected as final best matches for calculating the Global motion parameters.

IV GLOBAL MOTION ESTIMATION

The Global Motion Estimation unit produces a unique global motion vector (GMV) for each video frame that represents the camera motion during the time interval of two frames. The GMV consists of both handshake movement and intended camera movement. Handshake movement is calculated separately and eliminated in order to get the stabilized video and intentional camera motion is calculated by smoothing the global motion vector. In our method we calculate the GMV by taking the mean value of the selected matched pair in each frame in both X-direction and Y-direction

$$X = \frac{\sum_{i=1}^{n} X_i}{n} \qquad (2)$$

$$Y = \frac{\sum_{i=1}^{n} Y_i}{n} \qquad (3)$$

The flowchart of our video stabilization algorithm can be given as

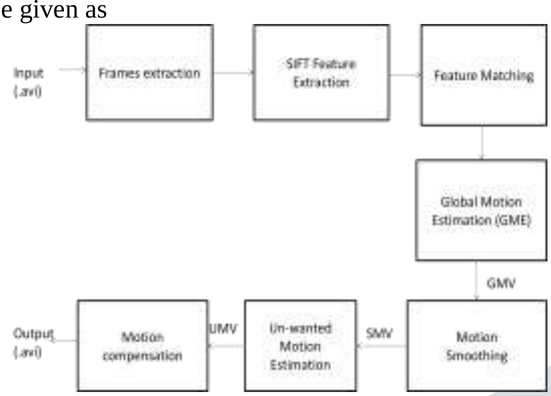

Figure.2: Flowchart of our proposed algorithm

V ADAPTIVE-FUZZY FILTERING:

Adaptive fuzzy filtering is used to smooth the GMV to generate the Smoothed Motion Vector (SMV). generates the CMV by a simple infinite impulse response (IIR) filter which is tuned adaptively by a fuzzy system. This filter automatically adjusts the coefficients depending on the unwanted camera motion and intentional camera motion. The construction of fuzzy rule base is contains 30 rules as presented in below table. To adapt the membership function of fuzzy input according to received video frames leads to a good performance for the stabilization system over difference in contents of video. From one end, a large smoothing factor prevents tracking of intentional camera motion when the acceleration is observed, so the smoothing factor should be tuned carefully.

Unwanted Motion Estimation:

Unwanted motion estimation (UMV) in our proposed method is obtained after obtaining the (SMV) by using the adaptive fuzzy filter. UMV can be given as

$$UMV(n) = GMV(n) - SMV(n)$$

This unwanted motion is removed from the videos with our proposed new method by using Accumulative Motion Vectors (AMV) and it is given as

$$AMV(n) = \sum_{i=1}^{N} UMV(\text{n})$$

To obtain the stabilized video, the frames are wrapped back to its original position i.e Stabilized video = offset – AMV(n)

MOS

Mean opinion score is carried out as a part of subjective analysis of our stabilized videos in this thesis work. I order to carry out subjective test in a convenient way; a MOS tool is designed with a local data base in the backend in our system. The tool is developed according to the ITU standards ITU-R BT.500-13[22] and ITU-T P.910 [21]. We use DSIS according to the ITU-R recommendations [22]. Subjective assessment is based on the human visuals system, when the subjects will grade his/her own perpetuation. The totals of 18 videos are rated by the subjects by using a grade scale of 9 points with an average age of 25. According to the recommendations by ITU-R, the number of human viewers participating in the subjective tests should not have lower than 15 years. We have performed subjective assessment on a total of 36 students with the average age of 25 from the BTH University.

6 videos using different compensation algorithms we used like adaptive filter, kalman filter and MVI are rated by the users using 9 points scale as mentioned in [21]. Figure [3-8] and [3-9] shows the design of our MOS tool which compiles of main page and the video assessment with the grading scale. The subjects given full instructions on the subjective assessment pattern and the working of MOS tool. The training sessions are shown to the subjects as per the recommendations [22] for their ease to perform the task and to avoid incorrectness in rating the video.

User Details

MOS Tool

Video Stabilization
Subjective Assessment

Figure** Error! No text of specified style in document..**3: User Interface of MOS

Figure** Error! No text of specified style in document..**4: 9 point scale

Grading for users subjective assessment of videos.Each time a reference video sequence (un-stabilized) is shown and then followed by source (stabilized) videos are shown randomly. Now subjects will rate the video with respect to the reference video by clicking on the grading scale from 1 to 9 under each video on the web-page as shown in figure [3-9]. After the grading is done for the video, new video sequence will run and process is repeated until last video sequence. The grading value is stored in the local database with their respective video names which are used to analyze the results both statistically and mathematically.

VI RESULTS

This chapter discuss about the results of video stabilization using our proposed method. These results are based on the experiment performed in the chapter 3. We evaluated the performance of our algorithm with different types of videos. Total of 16 un-stabilized videos were used for testing our algorithm and 6 videos are finalized as they are different in the camera motion than the others which are quite similar. The description of each video is given in the tabular form. The videos sequences 1 and 2 are shot from canon digital camera and the rest of the videos 3, 4, 5 and 6 are taken from the website respectively. The videos sequences that are selected for testing our algorithm consists of different resolution as CIF, QCIF, VGA, QVGA with different frame rate of 15 fps and 25 fps. All the video sequences used for the algorithm consists of .avi format.

Table** Error! No text of specified style in document..**1: Experimental Test Video Sequences

Video Type	Frame rate (fps)	Number of Frames	Resolution	First frame
CIF	15	115	352x288	
QCIF	25	122	128x96	
QVGA	15	70	320x240	
VGA	15	120	640x480	
QVGA	15	76	320x240	
VGA	25	90	640x480	

4.1 Time Consumption of Motion Estimation using our Proposed Method and Traditional Method

Table** Error! No text of specified style in document..**2: Time Consumption of Motion Estimation using our Proposed Method and Traditional Method

Video No	SIFT block (in seconds)	Traditional (in seconds)
1	30.23	70.55
2	30.23	80.34
3	40.53	110.12
4	37.16	125.13
5	44.21	156.33
6	39.49	87.21

The above table describes the time consumed in seconds for calculating the Global Motion Vectors using our proposed method with SIFT which extracts the features using four different blocks and traditional method [20] which considers the full image for extracting the features. It is clear that our method has less time consumption than the traditional method as the complexity of feature extraction is proportional to the area and the similarity measurement is proportional to the square of the area. The complexity is less than half to that of traditional method. The complexity of this process includes the key point extraction, similarity measurements of the extracted key points and final best matches obtained from the similarity measurement. The complexity of calculating the GMV using SIFT block based approach will extremely reduce the total computational complexity of the algorithm making it time saving and producing accurate matching of key points.

VII SMOOTHED MOTION VECTOR

SMV of our algorithm is calculated using adaptive fuzzy filtering. We have compared the SMV with respect to different compensation algorithms as kalman Filtering [19] and Motion Vector Integration (MVI) [20]. Figure shows the evaluation of SMV obtained by smoothing the GMV using different unstabilized test video sequences in X-direction and Y-direction. The plot consists of original GMV obtained from motion estimation which contains both the unwanted motion and wanted motion, Smoothed motion curve using adaptive fuzzy filtering, Kalman filteing and MVI. It can be clearly observed that from all the motion compensation algorithms our proposed adaptive fuzzy filtering performs well with minimum optimal compensation. The x axis is the number of frames of thee videos while the y axis depicts the displacement in pixels in all of the plots. By individually comparing SMV of both X-direction and Y-direction to adaptive fuzzy filter for the different video sequences used, it is observed that the kalman filter performance is low as it uses fixed value of filter coefficients and constant acceleration motion where as the adaptive fuzzy filter adapts the coefficients according to the size of the video. While comparing to the MVI the performance is close to the adaptive fuzzy filter and better than kalman filter in all video sequences. This is because the MVI uses sum of previous and current GMV as used by adaptive fuzzy filter but integrated with the fixed damping coefficients δ for calculating SMV. Therefore our proposed algorithm performed the minimal compensation by preserving the wanted motion than compared with the compensation algorithms in [19] and [21].

VIII CONCLUSION:

In our thesis work, we have proposed a novel approach for video stabilization based on the extraction and matching of SIFT features through video frame. We make use of feature-based motion estimation algorithm that extracts SIFT features from frames of video and then evaluation of trajectory is performed to estimate the inter frame motion, thus allowing to calculate Global Motion Vector (GMV). The intentional cameras motion is filtered using adaptive fuzzy filtering. The experiments confirm the effectiveness of the method with desired performance by panning and removing unwanted shake in different types of videos. Our algorithm would be effective for analysis of un-stabilized videos in digital cameras and mobile devices.

Future Work:

In this study, we mainly focused on stabilization of videos which are already taken from a camera or mobile device. Here we also took available resolution videos with three filters in comparison. In future, experimentation and subjective analysis can be done by using different types of filters and compare them even with high resolution videos and for the more we can even experiment improving our method on live streaming videos. And we can also compare the cost, time consuming and effectiveness of the proposed method with different tolls available in market for video stabilization.

REFERENCES

[1] L. M. Abdullah, N. Md Tahir, and M. Samad, "Video stabilization based on point feature matching technique," in *Control and System Graduate Research Colloquium (ICSGRC), 2012 IEEE*, 2012, pp. 303-307.

[2] L. Chang and L. Yangke, "Global motion estimation based on SIFT feature match for digital image stabilization," in *Computer Science and Network Technology (ICCSNT), 2011 International Conference on*, 2011, pp. 2264-2267.

[3] B. Pinto and P. R. Anurenjan, "Video stabilization using Speeded Up Robust Features," in *Communications and Signal Processing (ICCSP), 2011 International Conference on*, 2011, pp. 527-531.

[4] H. Rong, S. Rongjie, I. f. Shen, and C. Wenbin, "Video Stabilization Using Scale-Invariant Features," in *Information Visualization, 2007. IV '07. 11th International Conference*, 2007, pp. 871-877.

[5] K. L. Veon, M. H. Mahoor, and R. M. Voyles, "Video stabilization using SIFT-ME features and fuzzy clustering," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 2377-2382.

[6] J. M. Wang, H. P. Chou, S. W. Chen, and C. S. Fuh, "Video stabilization for a hand-held camera based on 3D motion model," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 2009, pp. 3477-3480 .