International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2763
Issue 04, Volume 3 (April 2016) www.ijirae.com

Implementation of a Robust Video Stabilization System on Raspberry Pi

M Chethan Kumar Naidu*
E&I Dept, BIT Bangalore-04
mchethannaidu01@gmail.com

Lokesha H
CSIR-NAL,Bangalore-17
loki_hlmg@nal.res.in

Ashwathappa P
E&I Dept, BIT Bangalore-04
pashwathappa@gmail.com

Abstract— Micro Air Vehicles (MAV) are now a days are very popularly and are used for wide range of applications such a surveillance, exploration etc. MAVs are generally fitted with a digital camera on board it and the videos captured by it generally involve undesired jitter. This paper mainly focuses on stabilizing the unstable videos; raspberry pi 2 is used to process these videos. This video stabilization mainly involves estimation of the camera motion path and smoothing motion path.

Keywords— ***Micro Air Vehicle, Video Stabilization, Motion Estimation, Harris Stephen Corner Detection, Random Sample Consensus.***

I. INTRODUCTION

This The MAVs are a type of unmanned aerial vehicle (UAV), its application ranges from surveillance, search and rescue, sensing, exploration, among others. The videos captured by an MAVs involves unstable camera motions which are caused by various reasons such as vibrations of onboard motors, weather conditions etc, so as an end result it produces unstable videos. The unstable video results in visual fatigue, and affects the further analysis of the image sequence. So in this sense it becomes detrimental that to reduce this undesired jitter, video stabilization becomes critical. Video stabilization is the process of removing undesired jitters and camera shakes in an unstable video which are caused due to unstable camera motion. Video stabilization also improves the quality of the video.

Generally there are two types of stabilization techniques one is the mechanical image stabilization where one uses external devices such as shock absorbers, gyroscopes etc., to stabilize unstable videos. The other technique is known as digital video stabilization where it is uses feature tracking method to obtain camera motion path and by using suitable motion model and motion compensation we can stabilize the videos. The later technique that is the digital video stabilization is a 2D process. Furthermore this is a robust stabilization technique where it does not require any additional information like details of the capturing device, captured place etc.

Video stabilization generally consists of motion estimation where we use feature tracking method Harris-Stephens corner detection to identify the camera motion path, next process is motion smoothing which is used to remove undesired camera motions by using an appropriate motion model such as homography camera motion path is smoothened. When performing video stabilization on the respective frames, due motion correction the frames may go out of bounds which creates black regions near the edges of the frames to avoid we go for cropping method. The general block diagram of video stabilization is shown in fig 1.

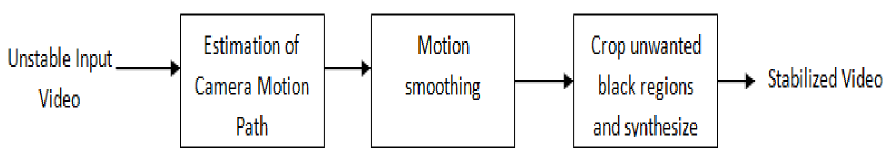

Figure 1 General video stabilization block diagram

II. PROPOSED METHOD

The proposed video stabilization algorithm flowchart is shown in Figure 2. First process is to obtain the video parameter information. Then we build the motion estimators to estimate the camera motion path. Using this information build the motion models to stabilize the videos. The stabilized video contains undesired black regions near edges of the frames to remove it we crop the frame.

A. Feature tracking and Feature Matching

Here the main goal is to identify motion distortion between two consecutive frames. Harris-Stephens corner detection is used to identify the features in each frame. The Harris-Stephens corner detection is one of the fastest feature tracking techniques. The corners detected on the first frame are shown in Figure 3.

After the key feature points from each frame are obtained, the correspondences between the points that are identified previously need to be picked up. The matching degree between the points that existed in frame A and B is needed to found for each point. The Sum of Squared Differences (SSD) can be used to measure the matching degree between points. Each point in frame A is compared with the corresponding point in frame B to find the lowest matching cost.

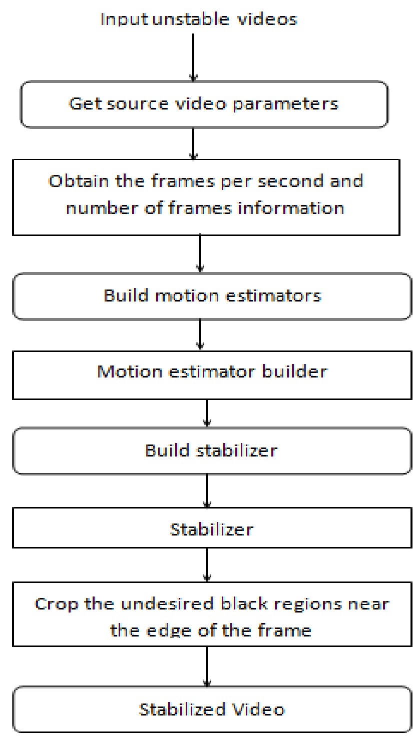

Fig 2. Proposed Video Stabilization Flowchart

The feature matching is to relate features of one frame to features of the second frame. The feature matching of the 1st and 2nd frames of the sample video is shown in figure 4. The same process is continued for all the frames of the video.

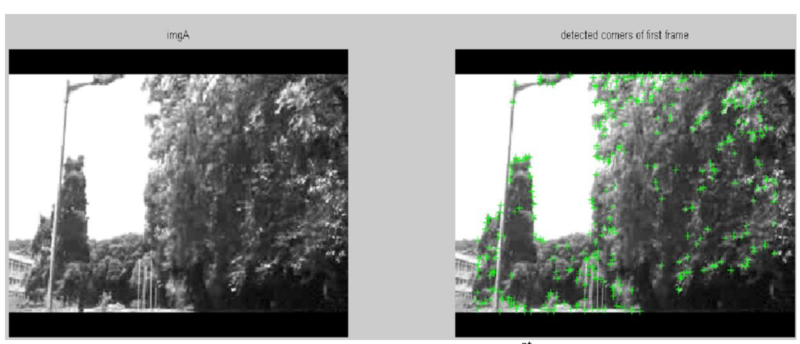

Fig 3 Corners Detected on the 1st Frame

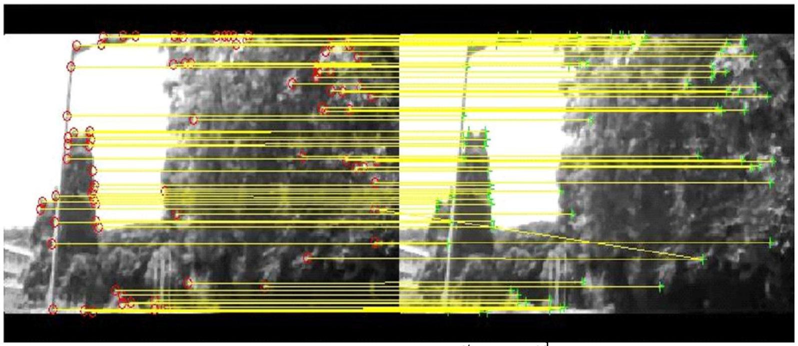

Fig 4 Feature Matching of 1st and 2nd Frame

B. Estimation of Motion Path.

In order to estimate the motion between two consecutive frames an affine transform that makes the descriptors from the first set of points match most closely with the descriptors from the second set is calculated. The affine transform will be a 3-by-3 matrix of the form:

$$\begin{bmatrix} a\_1 & a\_2 & 0 \\ a\_3 & a\_4 & 0 \\ T_x & T_y & 1 \end{bmatrix} \quad (1)$$

Where *a* define scale, rotation and sheering effects of the transform, while *Tx* and *Ty* are the translations. Suppose Px(x,y) and Px'(x',y') are the pixel locations of corresponding points in consecutive video frames, the relation between these two locations can be expressed as follows:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H.\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

$$H = \begin{vmatrix} S.\cos(\theta) & -S.\sin(\theta) & 0 \\ S.\sin(\theta) & S.\cos(\theta) & 0 \\ T_x & T_y & 1 \end{vmatrix} \quad (3)$$

Where the parameter S represents scaling Theta represents rotation and T_x, T_y represents translation of two successive frames.

C. Motion Smoothing

To obtain a stabilized video, the relative change acquired in above step is utilized to remunerate the present casing; this is likewise last stride for video adjustment. For a given arrangement of video casings Fi, i=0,1,… ,the above method is utilized to calculate the mutilation between all edges Fi and Fi+1 as relative changes, Hi. Thus the cumulative distortion of a frame *i* relative to the first frame will be the product of all the preceding inter-frame transform as:

$$H_i^{cummulative} = \prod_{j=0}^{i-1} H_i \quad (4)$$

To obtain a stabilized video, at every stride, the change H between two progressive edges is computed and joined with combined change Hcumulative. This portrays the camera movement from now on after the principal outline. This total change is utilized to twist the progressive edges in a video.

D. Crop Window

Due to motion correction of the video stabilization process there will be cases where the frames may go out of bounds of outside the boundary of the frame. Because of this effect some places inside the frame has black portions, which when viewed will result in unpleasant viewing experience. So therefore we generate a crop window such that it is always within the bounds of the frame and also this crop window will also be transformed according to motion models.

III. Hardware and software Analysis

A. Hardware Analysis

Raspberry Pi 2 model is the revised version of previous Pi models. It centralizes the embedded system module. Raspberry Pi 2 model consists of 4 input and output ports. More USB provides the overcurrent behavior and better hotplug. Micro SD version is changed to nicer push-push. Power consumption is reduced by using switches. Hence, power consumption is very less compared to other version i.e. 0.5W -1W. It has 1GB of Random Access Memory (RAM). It has a Quad-core ARM Cortex A7 CPU Processor with clock speed of 900MHz.ARMv7 allows the processor to run LINUX, snappy UBUNTU, and higher windows version i.e. Windows 10.Raspberry Pi processor requires its own Operating System (OS). Raspbian or NOOBS are the different version of OS which can be implemented.

B. Software Analysis

Software analysis deals with 2 different stages of analysis of software. First, let us discuss about the OS used. Raspbian is the OS which is used in Raspberry Pi 2, the OS is dumped in the SD card and then SD card is inserted into Pi board. After inserting the card, connect the necessary hardware components and provide power supply. Therefore, OS will be installed and it starts booting. Once the booting is finished it asks to login using default username and password. Once the board is setup, check out with network setting and network configuration to access internet. So that it will work with real time application.

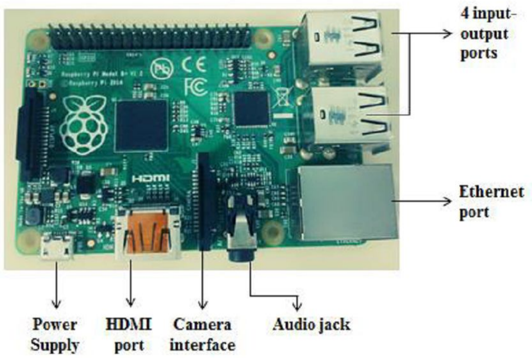

Fig. 5 Raspberry pi 2 Board

Secondly, software analysis deals with the Open CV platform which supports different programming languages like c/c++, python, Java, android etc. It also helps to access MATLAB modules. We are using the Open CV 3.0.0 version. Open CV helps the Python coding much simpler by providing predefined modules which saves the processing time and memory.

IV. Results and discussion

The most important part of this video stabilization process is the estimation strong features on each frames, for this Harris-Stephens corner detected are very important to identify the actual camera motion path. Next it is very important to match the features from one frame to its consecutive frame. Here it is very important to identify strong correspondences between the two frames because there are many wrong point correspondences also, to identify the strong correspondences we employ Random Sample Consensus (RANSAC). The matching of features of two overlapped frames are shown in Fig 6.

Further the output video quality is also measured based on the proposed methods. This is evaluated based on SVD based grayscale Image value and graphical measurement. Singular value decomposition (SVD) is developed as a new measurement that can express the quality of distorted images either graphically that is in 2D measurement or numerically as a scalar measurement, both near and above the visual threshold. The experiments here utilized SVD based measurement that outperformed the normally used PSNR.

$$M - SVD = \frac{\sum_{i=1}^{\left(\frac{k}{n}\right)x\left(\frac{k}{n}\right)} |D_i \; - D_{mid}|}{\left(\frac{k}{n}\right)x\left(\frac{k}{n}\right)} \quad (5)$$

Where Dmid represents the midpoint of the sorted DiS k is the image size n is the block size M–SVD is the measurement of Singular value decomposition. The results of various sample videos are shown in Table 1.

An example for the output quality for Vid1 based on Equation 1 with k=8, n=1, Di and Dmid represented by 256×256 matrix attained M-SVD of ≈ 22.20. Hence, the numerical quality obtained from the three sample videos are tabulated in Table 1. As visualize in Figure 6, it can be seen that Vid3 obtained the best quality based on the calculated value that is 40.50% followed by Vid2 with 39.21% and Vid1 22.20%**.** This resembled that Vid1 has great distortion whilst Vid3 is least distorted.

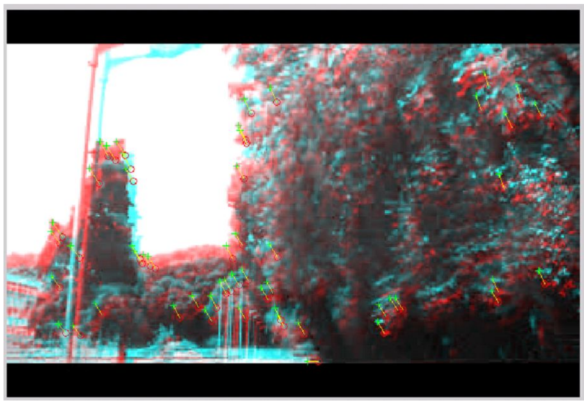

Fig. 6 Feature Matching between two overlapped frames

TABLE 1: SAMPLE VIDEO AND QUALITY RESULTS

SAMPLE 1ST FRAME	SAMPLE INPUTS	NUMBER OF FRAMES	QUALITY VALUE (M -SVD)
	Vid1	749	29.70
	Vid2	382	25.40
	Vid3	90	44.26
	Vid4	130	35.94

V. CONCLUSION

The unstable videos are stabilized to a satisfactory level and also the quality of the video is also increased. The video stabilization is processed on the raspberry pi in real time. Now the future developments could be to find a better feature detectors and motion models to improve the efficiency of the system. Further we employ cropping information in the frames which some of the viewers may dislike, to overcome this problem we can look at motion inpainting technique.

ACKNOWLEDGMENT

The authors would like to acknowledge to CSIR- National Aerospace Limited, Bangalore Institute of Technology for their support rendered during the tenure of this project work.

REFERENCES

[1] C. Harris and M.J. Stephens, “A combined corner and edge detector”, Proc of Alvey Vision Conference, pp 147-152, 1988.
[2] A C. Morimoto, R. Chellappa. “Evaluation of Image stabilization algorithms,” ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, vol. 5, pp. 2789-2792, 1998.
[3] Rodrigo José, Ortiz Cayón “Online video stabilization for UAV” AI & R Lab, Dipartimento di Elettronica e Informazione,2012-13.
[4] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3d video stabilization . In ACM SIGGRAPH 2009. 225, 226, 230, 231
[5] Y. Wang, Z. Hou and K. Leman, “Key point-Based Near-Duplicate Images Detection Using Affine Invariant Feature and Color Matching”. ICASSP 2011, Prague, Czech Republic, May 2011.
[6] Sheng-Che Hsu, Sheng-Fu Liang, Kang-Wei Fan, and Chin-Teng Li, A Robust In-car digital Image stabilization Technique”, IEEE Trans. on Systems Man and Cybernetics, Vol.37, No.2, March 2007.
[7] opencv-python-tutorials-readthedocs.org
[8] www.raspberrypi.org