Multimed Tools Appl (2015) 74:6745–6767
DOI 10.1007/s11042-014-1928-z

Video stabilization with moving object detecting and tracking for aerial video surveillance

Ahlem Walha · Ali Wali · Adel M. Alimi

Published online: 5 April 2014
© Springer Science+Business Media New York 2014

Abstract Aerial surveillance system provides a large amount of data compared with traditional surveillance system. But, it usually suffers from undesired motion of cameras, which presents new challenges. These challenges must be overcome before such video can be widely used. In this paper, we present a novel video stabilization and moving object detection system based on camera motion estimation. We use local feature extraction and matching to estimate global motion and we demonstrate that Scale Invariant Feature Transform (SIFT) keypoints are suitable for the stabilization task. After estimating the global camera motion parameters using affine transformation, we detect moving object by Kalman filtering. For motion smoothing, we use a median filter to retain the desired motion. Finally, motion compensation is carried out to obtain a stabilized video sequence. A number of aerial video examples demonstrate the effectiveness of our proposed system. We use the software Virtual Dub with the Deshaker-Plugin for test purposes. For objective evaluation, we use Interframe Transformation Fidelity for video stabilization tasks and Detection Ratio for moving object detection task.

Keywords Moving object detection · Aerial surveillance · Scale invariant feature transform (SIFT) · Digital video stabilization

A. Walha (✉) · A. Wali · A. M. Alimi
REGIM: REsearch Groups on Intelligent Machines, National Engineering School of Sfax (ENIS),
University of Sfax, BP 1173, Sfax 3038, Tunisia
e-mail: walha.ahlem@ieee.org

A. Wali
e-mail: ali.wali@ieee.org

A. M. Alimi
e-mail: adel.alimi@ieee.org

1 Introduction

Aerial surveillance has a growing importance nowadays. It is an effective way to provide large amounts of video data for a variety of applications including search and rescue, military operations, commercial applications, counter terrorism, and border patrol. Objects in aerial videos need to be detected and labeled in order to be used in other automated video processing, such as event detection, summarization, indexing and high level aerial video understanding.

A key task of video surveillance is to identify and track all moving objects in the scene and to generate exactly one track per object. This may involve detecting the moving objects and tracking them while they are visible. In aerial surveillance, this problem is very difficult. The challenges of moving object detection in mobile platform include camera motion, small object appearances of only few pixels in the image, changing object background, object aggregation, panning, and noise. Therefore, video stabilization has become essential in mobile surveillance systems. Also it is the first step in many aerial applications.

In literature, various moving object detection systems in aerial video surveillance are reported. These systems usually applied video stabilization as a pre-processing step to analyze aerial video. But by using this method we can lose slowly moving object.

Our contribution in this paper is that we propose to integrate the moving object detection into the stabilization algorithm and demonstrate that detection after stabilization doesn't work well. We also demonstrate that Scale Invariant Feature Transform (SIFT) as features are robust for video stabilization and moving object detection purposes. Evaluation of commonly used feature detectors and descriptors showed that SIFT performs better on a wide range of test sequences. By using SIFT point extraction and matching, we can locate regions of the image where a residual motion occurs. In this paper is that we applied Kalman filtering on this moving region and not on the whole image in order to estimate the motion of the region.

The paper is organized as follows. In Section 2 we provide a summary of the related work in the area of video stabilization and moving object detection. In Section 3, we present the challenge of aerial video surveillance. Section 4 gives the complete system framework of our proposed approach by the SIFT feature extraction and matching process and how it is adopted to the stabilization and motion detection problem. In Section 5 our detailed experimental evaluation which includes comparisons with existing methods are outlined. Finally, Section 6 concludes the paper along with future research directions.

2 Related work

In general, two modules are necessary for digital video stabilization : global motion estimation module and motion compensation module. A perfect motion correction need an accurate global motion estimation. In literature, there are many methods that aim to estimate global motion accurately. In [10], a method for global motion estimation is introduced by calculating the motions of four sub-images located at the corners of the image. This method is proved to be efficient and accurate, but it is of limited applicability because of the assumption that foreground objects are more likely located at the center of the image and hence less likely to be cropped in these four local images at the corners. In [27], a method based on circular blocks matching is proposed in order to estimate local motion . The global motion parameters is generated by repeated least square. In [5], global motion is estimated by extracting and tracking corner features. However, these features are not robust when image transformations such as scaling and rotation are present. Hence, SIFT [18], which is considered to be invariant to image scaling and rotation, is being widely used in the latest methods for global motion estimation [2]. In [21], SIFT based on particle filter approach is introduced. The authors used particle filter for an accurate estimation of undesired motion of the camera.

To separate desired motion from undesired ones several techniques are used such as Motion Vector Integration [9], Frame Position Smoothing [30], Gaussian filtering [12], Kalman filtering [10] and extended Kalman filtering. In many video stabilization systems motion estimation parameters are determined by Kalman filtering [10]. But, executing this algorithm on the whole frame is inefficient because Kalman filtering is unable to handle nonlinear models and non-Gaussian noise. Yang et al. [29] propose a combination method between particle filter and SIFT algorithm [24] to estimate the global motion parameters. The authors claim that this algorithm is very efficient because the process of generation of proposal density using SIFT algorithm highly reduces the number of unnecessary particles (samples for motion estimation). But Yang et al.'s method suffers from the problem of foreground interference, that is, inaccurate estimation of the global camera motion resulting from a moving foreground object. Also the use of particle filter need an extra computational load for computing movement and correction for each particle. In addition it is unable to remove outlier.

For moving objects detection and tracking in surveillance video, relative works have been done in literature. However, they mostly tackle stationary camera scenarios [25]. Recently, there has been an increasing interest in studying motion from aerial video [22]. Most of these recent works use the temporal information obtained from video feed either for tracking and detection of moving objects [13] or for enhancement of the detection performance of stationary objects [20]. Lin et al. [14] proposed an ego-motion estimation and background/foreground classification method. The authors built their model focusing on the motion vectors obtained by using the SURF algorithm to extract the feature points and their correspondence between frames. There may be some problems caused by the fact that the feature points selected by the algorithm may be lost. As a result, the model built based on the feature motion vectors may fail to get the moving object. In [26], the authors obtain the motion model of the background by computing the optical flow between two adjacent frames in order to get motion information for each pixel. Cuntoor [7] used Histogram of oriented Gradients, Histogram of oriented optical Flow and Haar features to classify the motion segmentation into person vs. other and vehicle vs. other. Rudol et al. [20] detected stationary and moving humans in thermal imagery by using Haar features with an AdaBoost. But the detection is only valid if a person is detected in a number of consecutive frames. COCOA [1] system is a 3-staged framework. It is capable of performing motion compensation, moving object detection and tracking on aerial videos. In this Work, motion compensation is achieved using direct frame to frame registration which is followed by an object detection algorithm that relies on frame differencing and background modeling. Finally, moving blobs are tracked as long as the objects remain in the field of view of the aerial camera. The detection approach presented in [16] is also based on assumption that potential detection region has to be present over a number of consecutive video frames. First, the track of a potential human is build by extracting points of interest and matching them, in this case the potential person signature associated with this track is classified using standard template matching. In [28], the authors proposed a detection approach of moving vehicles by using a Bayesian framework to estimate the optical flow. But this method didn't achieve good performance under the condition of camera vibration and noise interference.

Yang et al. [31] introduce an aerial video surveillance system to detect moving and static vehicles. For moving vehicles, extracted and matched feature point (SIFT and Kanada-Lucas-Tomasi(KLT)) are classified into three categories: background, moving vehicles with forward direction and moving vehicles with backward direction. For static vehicles, road region is extracted by a method based on edge detection algorithm in the first place.

Multiple Hypothesis Tracker [8] for moving target tracking on aerial video is not a good choice because many objects are very close together and generate too many data association hypotheses. Recently, a new class of filters, called the Probability Hypothesis Density [6], has been introduced for radar applications and has been used in video tracking.

All these systems solve the problem of moving object detection after the step of stabilization [23]. In this paper, we present a new system for video stabilization and moving object detection in aerial video with hybridization of the two processes. Our problem is defined as considering a particular pixel position. The pixel value is changing over time without a certain pattern in aerial video because the background is changing all the time.

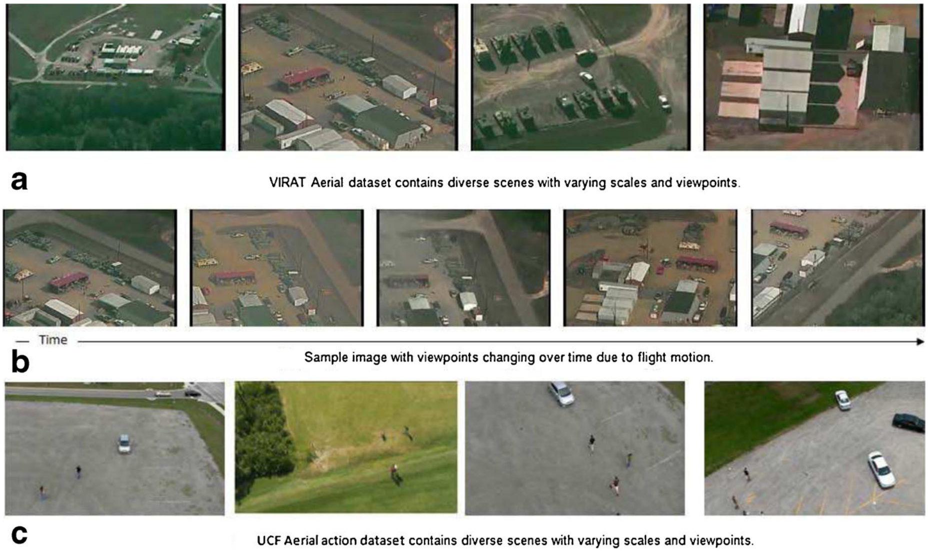
3 Aerial video characteristics

When an UAV flies at hundreds of meters it provides a large region of surveillance , but they usually produce noisy and blurring images. Moving sensor in UAV surveillance systems, capture videos at low resolution and low frame-rate, which presents a challenge to motion detection, foreground segmentation, tracking and other related algorithms. Scale and view variations and few pixels on region of interest are among these challenges. So the per-processing is necessary to attain a better detection effect. Characteristics of the video itself, such as unmodeled gain control, rolling shutter, compression, pixel noise and contrast adjustment may further impede motion detection algorithms due to violation of brightness constancy and geometric models. Unlike the fixed background in traditional surveillance system, the background in the UAV surveillance platform changes frequently because of the high speed of the aircraft. To identify traffic status and incidents, the complex background should be filtered, traffic features should be detected, both of which should be accomplished. So it has a very high demand of the performance of the algorithm of moving object detection. In this context, our tests video are obtained using a UAV equipped with a camera. Videos are recorded at a flying altitude of over 400 feet. Figure 1 show this data set.

4 Overview of the proposed system

Background modeling could not be applied to mobile surveillance system when background is, moving fast. At the same time, camera vibration and noise enormously affect the accuracy of detection. SIFT feature extraction can identify keypoints in order to be tracked over multiple frames of video and they are invariant to image translation, rotation and scale. For this reason, in this paper, we use the feature point tracking method to acquire a serial of features, which are then classified into three categories: undesired motion, moving object and static object. Our system consists of three sub-systems:

- **Global motion estimation** using SIFT feature point extraction and matching to eliminate camera vibration and noise.
- **Moving object detection** using RANSC and Kalman filtering.
- **Motion compensation** using affine transformation



VIRAT Aerial dataset contains diverse scenes with varying scales and viewpoints.

Sample image with viewpoints changing over time due to flight motion.

UCF Aerial action dataset contains diverse scenes with varying scales and viewpoints.

Fig. 1 Aerial dataset: **a** Diversity of VIRAT scenes **b** Sample images containing person activities over time **c** Diversity of UCF scenes

The flowchart of our method is illustrated in Fig. 2 and the details are explained as follows.

4.1 Local feature extracting and matching

SIFT algorithm presents a great adaptability to detect and describe keypoint feature because of its invariability to scale changes, rotation changes and blur [15]. This algorithm operates through four steps: detection of extrema in scale-space, localization and filtering of keypoints, assignment of orientation and the generation of descriptors. The SIFT algorithm establishes difference of Gaussian (DOG) in order to identify the locations of candidate keypoints in different scales spaces. The function of DOG can be defined as follows:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \qquad (1)$$

Where $L(x, y, \sigma)$ is an approximation to the scale normalized Laplacian of Gaussian, $I(x, y)$ is an input image, σ is the scale factor, $*$ is the convolution operator and $G(x, y, \sigma)$ is a variable scale Gaussian, which is defined as

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} exp^{-(x^2+y^2)/2\sigma^2} \qquad (2)$$

To locate stable keypoint, the scale-space extrema is found by computing the difference-of-Gaussian between the two images. To detect the local maxima and minima of $G(x, y, \sigma)$, each pixel in the DoG images is compared to its 8 neighbors at the same scale, plus the 9 corresponding neighbors at neighboring scales. If a pixel is a local maximum or minimum, it is selected as a candidate key-point.

To reject law contrast points, which are sensitive to noise and poorly localized along the edge, we use second order Taylor expansion of the scale space function $D(x, y, \sigma)$ at

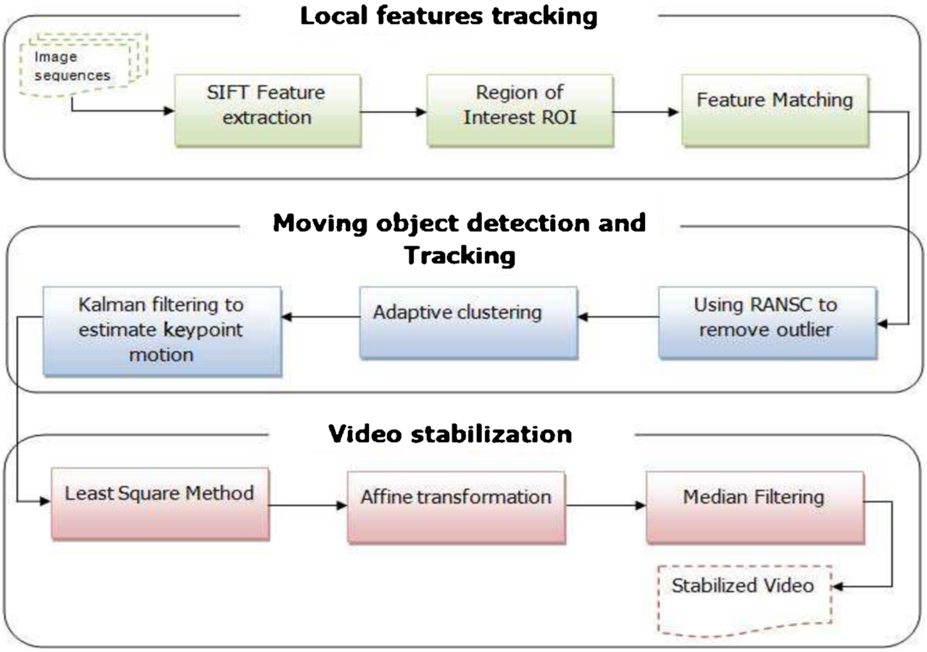


Fig. 2 Overview of the proposed system

sample point X_0, which becomes:

$$D(X) = D(X_0) + \left(\frac{\partial(D(X_0))}{\partial X}\right)^T X + \frac{1}{2}X^T \frac{\partial^2(D(X_0))}{\partial^2 X} X \tag{3}$$

$X = (x, y, \sigma)^T$ denotes the offset from the sample point. The location of extremum $\hat{x}$ can be calculated by differentiating equation (3) with respect to X and equating to zero, yielding

$$\hat{X} = -\left(\frac{\partial^2 D}{\partial X^2}\right)^{-1} \frac{\partial D}{\partial X\cdot} \tag{4}$$

So the function value at the extremum $D(\hat{x})$, can be written as equation (5) and then be compared with a threshold D_0 When $|D(\hat{x})| < D_0$, the point should be rejected.

$$D(\hat{X}) = D + \frac{1}{2}\frac{\partial D^T}{\partial X}\hat{x} \tag{5}$$

For stability, it is not sufficient to reject keypoints with low contrast. The difference of Gaussian function will have a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise. So it is possible to filter out the feature points along the edge by using a $2 * 2$ Hessian matrix, giving

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{6}$$

Assuming α, β are the eigenvalues of the matrix H, which satisfy $\alpha > \beta$. So

$$\frac{Tr^2(H)}{Det(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(\gamma\beta+\beta)^2}{\gamma\beta^2} = \frac{(\gamma+1)^2}{\gamma} \tag{7}$$

Thus, if $\frac{Tr^2(H)}{Det(H)} > \frac{(\gamma_0+1)^2}{\gamma_0}$ the point should be rejected as edge point where γ_0 is a threshold. green This threshold is the difference between the 2 biggest gradients. It is not difficult to find that the greater γ_0 is the more feature points we will get.

After feature extraction process, it is necessary to match feature point between two successive frames.

For this process, we are investigating the matching process as proposed by Lowe [15].This process is based on finding a match between two consecutive image features using Euclidean distance. The Euclidian distance between SIFT descriptors is employed to determine the initial corresponding feature point pairs in different images. We used RANSAC to filter outliers that come from the imprecision of the SIFT model.

4.2 Moving object detection

RANSAC is a robust estimator [11] where it was used to derive a usable model from a set of data.We used RANSC to find the dominant motion without being influenced by the noise motion produced by moving object. The distance parameter t is determined by a statistical theory. Firstly, assume that the distribution of effective point under transformation model according to the distance is known, we calculate the distance threshold t such us the probability of effective point in point set is α. Suppose the distribution satisfies the zero mean and variance σ of the Gaussian distribution, we can compute the value t. In this case, the square distance between points is d^2, which is the square sum of Gaussian variant, meets the χ_m^2 (Chi-square Distribution) that has m degrees of freedom. Based on the integral property of Chi-square Distribution, the probability of random variable that obeys the Chi-square Distribution is lower than the integral upper limit, the formal is as follows

$$F_m(k^2) = \int_0^{k_2} \chi_m^2(\xi)d\xi < k^2 \tag{8}$$

the distance threshold t can be calculated by

$$t^2 = F_m^{-1}(\alpha)\sigma^2 \tag{9}$$

Then, we can classify the point set into effective point and invalid point.

In our case, the distance threshold for deciding outliers for RANSAC is $d = 0.005$.

$$\begin{cases} if \quad d^2 < t^2 \quad then \quad effectivepoint \\ else \quad invalidpoint \end{cases} \tag{10}$$

The second parameter for RANSAC is the number of iterations, which N is chosen high enough to ensure that the probability p (usually set to 0.99) which is at least one of the sets of random samples does not include an outlier. In our case maximum number of iterations is 1 000. Let u present the probability that any selected data point is an inlier and $v = 1 - u$ is the probability of observing an outliers. N iteration of the minimum of denoted points are required, where

$$1 - p = (1 - u^m)^N \tag{11}$$

And thus with some manipulation,

$$N = \frac{log(1-p)}{log[1-(1-v)^m]} \tag{12}$$

The consume time of RANSAC can be calculated as follows:

$$T = N(T_G + MT_E) \tag{13}$$

where T_G is the time spent on generating a hypothesis, T_E is the time spent on evaluating the hypothesis for each data, M is the number of whole data.

Let $P = \{p_1, ..., p_n\}$ be a set of points in R^d. The Voronoi cell associated to a point p_i, denoted by $V(p_i)$, is the region of space that is closer from $p_i$than from other points in P:

$$V(p_i) = \{p \in R^d : \forall \neq i, \parallel p - p_i \parallel \leq \parallel p - p_j \parallel\} \tag{14}$$

where $V(p_i)$ is the intersection of $n - 1$ half-spaces bounded by the bisector planes of segments $[p_i p_j]$, $j \neq i$. Therefore, $V(p_i)$ is a convex polytope, possibly unbounded. The Voronoi diagram of P, denoted by V or (P), is the partition of space induced by the Voronoi cells $V(p_i)$. Result is shown in Fig. 3.

After keypoint feature matching, we obtain a whole number of local motion vectors. The sets of vector contain two types of information: real motion of the camera and motion related for moving objects in the scene. To distinguish between these two motions, we assume that the velocity of moving objects is very large compared to other motions.

In this step, an adaptive clustering is employed. A threshold is fixed to select moving objects. This threshold is the average distance between all matching points. It varies from one frame to another. By comparing this threshold to distance between two correlated keypoints we can detect moving object. In our system, we detected moving object for each frame. Therefore tracking becomes difficult. Because false detection and the presence of objects that enter and leave the scene can modify the number of detected object in

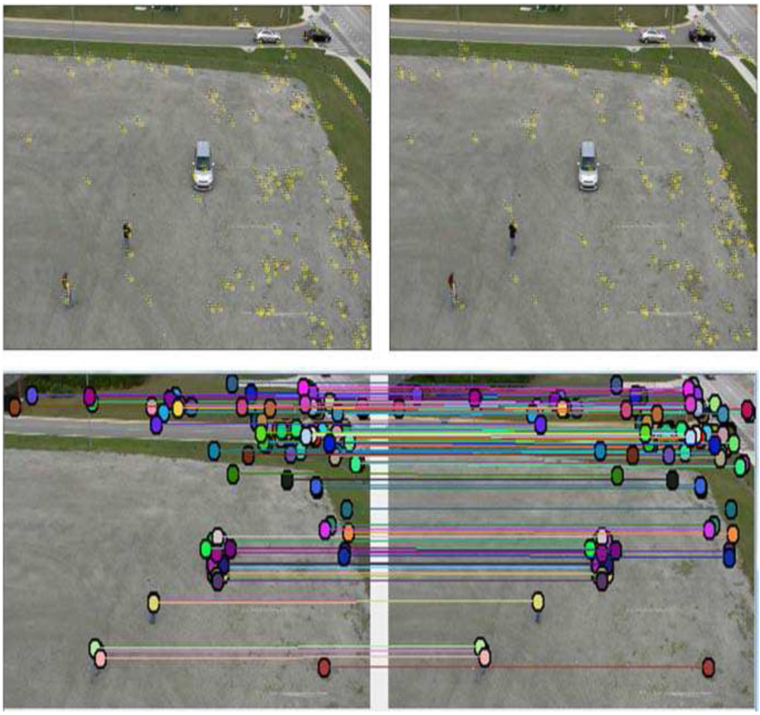


Fig. 3 SIFT point extraction and matching for two consecutive frame

consecutive frames. To resolve this problem, we used Kalman filtering for each keypoint selected as moving object. In general, Kalman filter is used for filtering a noisy dynamic system. It estimates the new states of the system and then corrects it by the measurements. In our case, the motion can be described as shown in (15) and (16).

$$\mathbf{x_k} = \begin{pmatrix} x_k \\ \dot{x}_k \end{pmatrix} = \begin{pmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} + w_{k-1} \tag{15}$$

$$z_k = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \mathbf{x_k} + v_k \tag{16}$$

Considering x_k as the position coordinate in one direction, z_k as the measured position, w_{k-1} is the process noise and v_k is the measurement noise. We assume zero to the acceleration because we do not have information about the control of the motion, also we modeled the change in velocity by the process noise. As it is shown in (15), we do not include the acceleration in the process equation, and the effect of the acceleration noise is described by the velocity noise. Result is shown in Fig. 4

4.3 Video stabilization

We adopt four parameters 2D affine motion model to describe geometric transformation between two consecutive frames. If *P(x,y,1)* is the point in frame *n*, and *P'(x',y',1)* is the same point in the successive frame, then the transformation from *P* to *P'* can be represented as shown in (17).

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} S\cos\theta & -S\sin\theta & T_x \\ S\sin\theta & S\cos\theta & T_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{17}$$

The affine matrix can describe accurately pure rotation, panning, and small translations of the camera in a scene with small relative depth variations and zooming effects. *S* is the

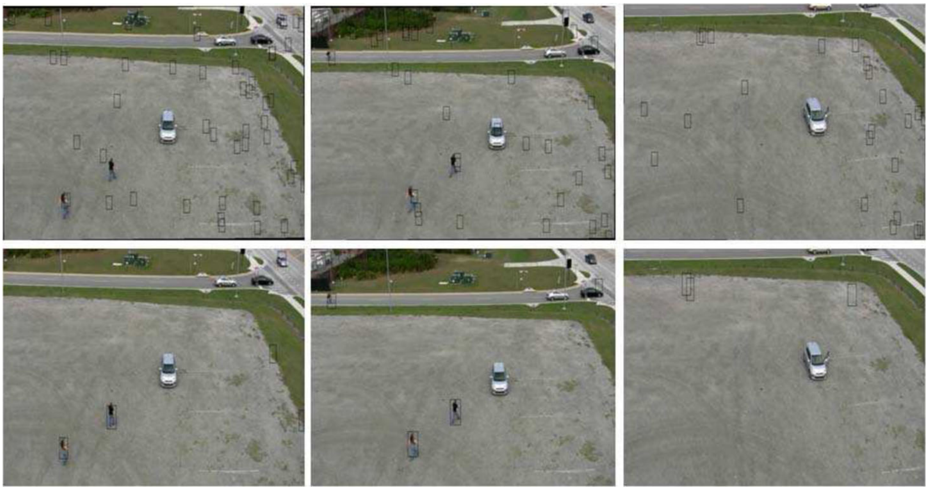


Fig. 4 Kalman filtering

scale, θ is the rotation and T_x and T_y are the translations. This has only four free parameters compared to the full affine six transformations: one scale factor, one angle, and two translations. The linear Least Squares Method on a set of redundant equations is a good choice to solve this problem. It results in robust parameter estimation. Least square method is used to reduce the error of the image processing and easier to implement fast, accurate, and robust motion estimation.

The transformation from a frame to the corresponding motion compensated frame is directly computed using this affine transformation matrices. Then, we need to compensate the current frame to obtain stable images. Compensation of the images cannot be calculated directly from the parameters calculated in (17), since undesired motion of the sensors and normal motion of the UAV should be separated. Finally original frames must be warped to obtain the compensated frames.

This is achieved by using median filter to estimate motion for the current frame. Specifically, the proposed method calculates the spatial correlation between the current motion vector and its neighboring motion vectors. If the current motion vector is not correlated with its neighboring motion vectors, we decide that the current motion vector is an outlier, and correct the motion vector into a new motion vector generated by the median filter. Hence, we can remove undesired motion in the frame caused by outliers, and improve the image quality of the frame. So it is a straightforward process to place the new frame back in line with the estimated motion.

5 Experimental results

To illustrate the effectiveness of our system we conducted our test on two different databases: VIRAT Aerial Video Data Set[1] and UCF Aerial Action Data Set.[2]

The resolution of aerial videos in VIRAT Aerial Video Data Set are at 640x480 with 30Hz frame rate recorded from a high altitude with low resolution and tough conditions. Also UCF Aerial Action Data Set contains collection of video and represents a diverse pool of actions featured at different heights and aerial viewpoints. Multiple instances of each action were recorded at different flying altitudes which ranged from 400-450 feet and were performed by different actors. The videos were taken in 60 frames per second with the resolution of 1920 x 1080 pixels.

The challenges posed by these two datasets include characteristics of aerial videos such as different shapes and sizes, scale changes, shadows, cloth variations, variety of scenes, and different scenarios. Figure 1 summarizes the 2 datasets adopted to evaluate our system.

5.1 Comparison of feature detection methods

In this test, we compare six famous feature detection methods:

- **BRIFF** *(Binary Robust Independent Elementary Features)*: introduced by Calonder et al. [4] as an efficient descriptor for keypoints. This descriptor is built by simple binary tests on a subset of the pixels surrounding the keypoint center.

[1] http://www.viratdata.org/

[2] http://crcv.ucf.edu/data/UCF_Aerial_Action.php

- **RIFF:** ***(Rotation-Invariant, Fast Feature)*****:** introduced by Takacs et al. [17],based on a HOG computed at a circular support area and used an annular binning to achieve orientation invariance.
- **LAZY**.
- **SURF:** ***(Speeded Up Robust Features)*****:** introduced by Bay et al. [3] and based on Fast-Hessian Detector to find Keypoints. this descriptor mainly focuses on reducing computational time.
- **ORB:** ***(Oriented FAST and Rotated BRIEF)*****:** introduced by Rublee et al. [19] ORB detect key points by adding a fast and accurate orientation component, and uses the rotated BRIEF descriptor.

The performance of these methods are compared for rotation, blur and illumination changes: All the experiments use repeatability measurement and the number of correct matches for the evaluation measurements.

We use the same image dataset, which includes the general deformations, such as view changes, illumination changes and rotation as shown in Fig. 5. All the experiments

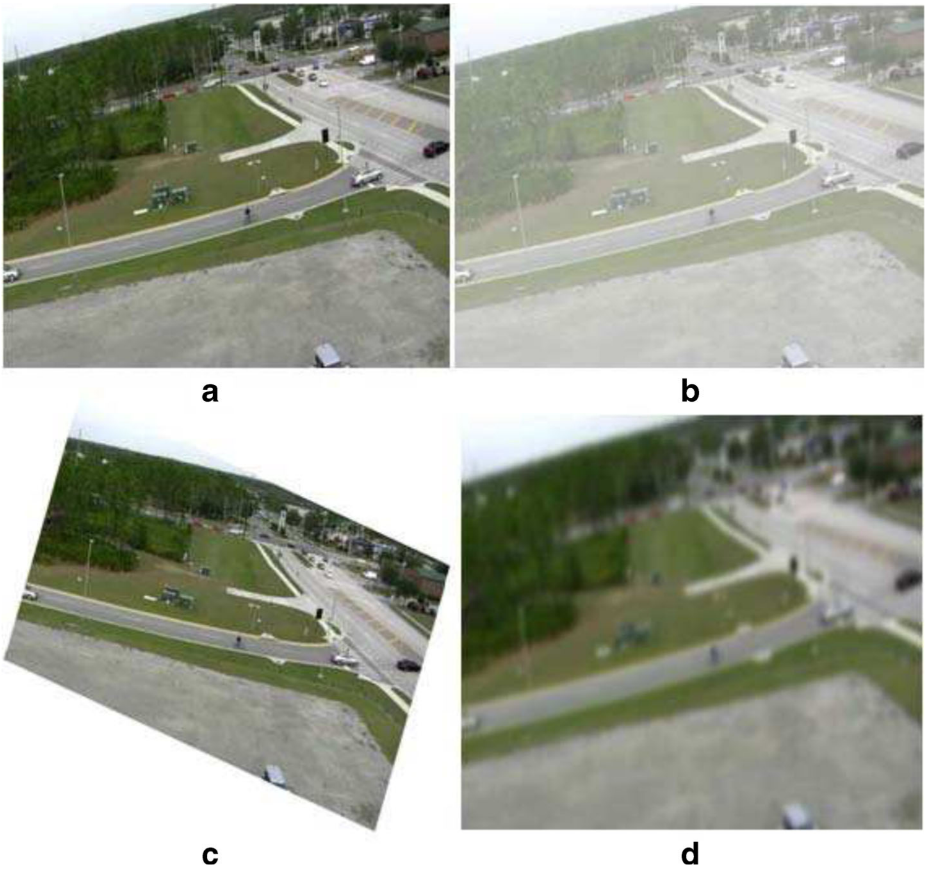


Fig. 5 Part of test images. **a** is the original image, **b** is the illumination changed image, **c** is the rotation images and **d** is the blurred images

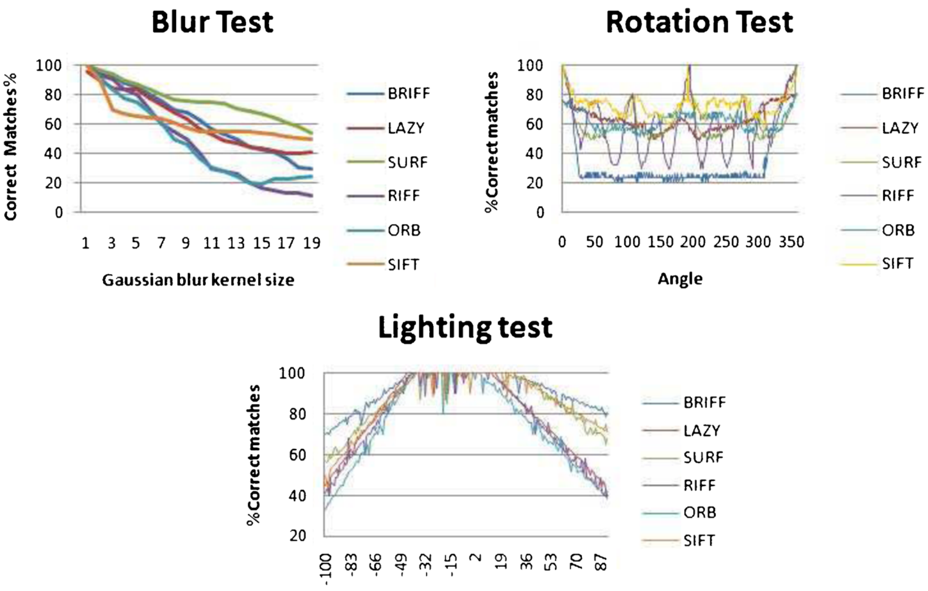


Fig. 6 Blur, rotation and lighting test

work on PC AMD 3200+, 2.0G, and 2.0 GO RAM, with Windows 7 as an operating system.

Three qualities and one performance tests were done for each kind of descriptor.

- **Rotation test** this test shows how the feature descriptor depends on feature orientation.
- **Blur test** this test shows how the feature descriptor is robust against blur.
- **Lighting test** this test shows how the feature descriptor is robust against lighting.
- **Performance test** is a measurement of description extraction time.

For the rotation test case, it's the rotation of the source image around its center for 360 degrees. Blur test uses Gaussian blur with several steps and the lighting test changes the overall picture brightness. The metric for all quality tests is the percentage of correct matches between the source image and the transformed one.

As shown in Fig. 6, SIFT is represented by the orange line that detects more matches which are stable to rotation. In blur test, we simulate the motion blur which can occur if camera moves suddenly. All descriptors demonstrate good results in this test. The more blur size is applied, the less percent of correct matches is obtained. In this case, SIFT shows its best performance here. In lighting test, the transformed images differ only in overall

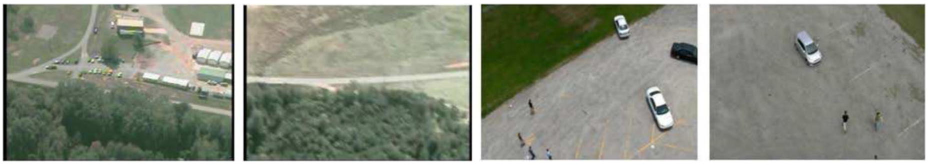


Fig. 7 Original image

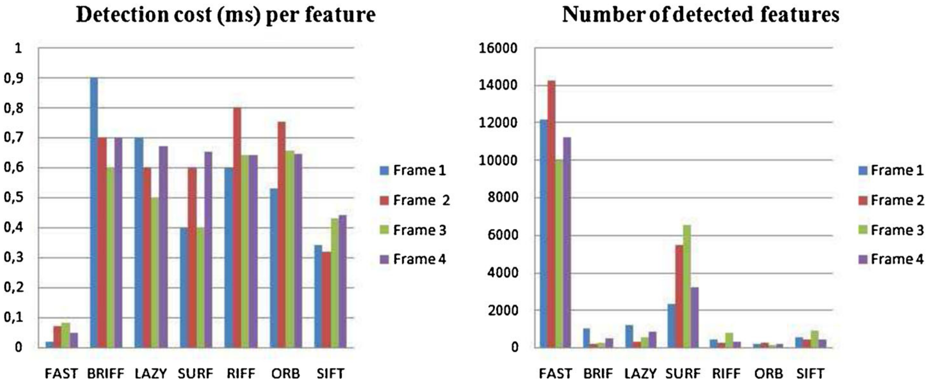


Fig. 8 Performance test

image brightness. All kinds of descriptors work well in this case. The major reason is that all descriptors extracted are normalized. We can consider that SIFT present an acceptable result.

To verify the effectiveness of SIFT, four images are taken for the experimental data as shown in Fig. 7.

Regarding computational efficiency, in Fig. 8 , SURF requires approximately 0.4 ms to detect a feature. The run-time of FAST is under 0.1 ms per feature while that of SIFT comes to 0.5 ms per feature and other methods compute each feature in around 0.6 ms. SURF is faster than SIFT in detecting features, since the SURF method uses a fast-Hessian detector on the basis of an integral image. However, all methods cannot track motion in real time (25 frames per second). This is because feature-based motion recovery methods are time consuming in terms of detecting points and finding their correspondences. However, we can use the GPU (graphics processing unit) to accelerate our implementations and make it real time.

The result of this experiment shows that SIFT matching is a robust descriptor for error localization. SIFT present their stability in most situations although they are slow. SURF is the fastest one with good performance the same as SIFT. ORB show their advantages in rotation and illumination changes.

5.2 Stabilization evaluation

Our first experiment consists of comparing our system to Deshaker.[3] Figure 9a shows five frames of the unstable input sequence corresponding to frames 2, 3, 5, 18 and 21 are taken from UCF Aerial action Data Set. In this video sequence, the scene contains one moving object. Figure 9b shows the stabilization result of Deshaker system and Fig. 9c shows the stabilization result of our system. Another result with many moving objects in the scene is shown in Fig. 10. This scene is challenging because of the fast moving objects.

Deshaker calculates motion vector using matching algorithm. Based on motion vector, panning and rotation are calculated and compensation motions are performed. The stable results are marked by the red lines, which shows the stabilized value and produced video sequence. We find that our stabilization system is working well especially in the case where

[3] http://www.guthspot.se/video/deshaker.htm

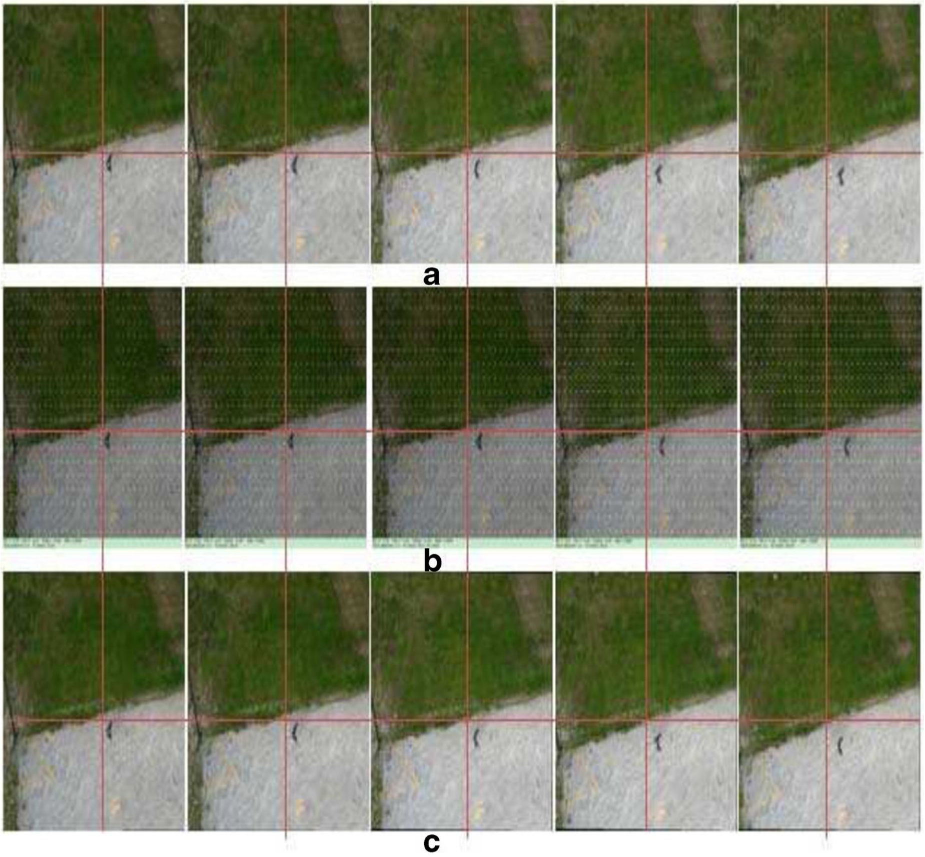


Fig. 9 Input frame from UCF Aerial dataset with one moving object in the scene. **b** Stabilized frame by deshaker stabilizer. **c** Stabilizer frame by our system

video sequences include moving objects. Due to the accuracy of detected keypoints and the use of adaptive RANSAC to remove outliers, our system gives good results compared to Deshaker .

Next, we use a sequence from VIRAT Aerial Video. Figure 11a shows five frames of the unstable input sequence. Figure 11b shows the stabilization results using Deshaker. Figure 11c shows the stabilization results using our proposed system. The challenges presented by this sequence include low image resolution, changing weather conditions, and crowded backgrounds. We use this sequence to illustrate the robustness of our system to non uniform depths which are present in this sequence.

We used Peak Signal-to-Noise Ratio (PSNR), an error measure, to compare the quality of the video stabilization with Deshaker and COCOA System [1] . $PSNR$ between frame n and frame $n + 1$ is defined as

$$PSNR(n) = 10 \log_{10} \frac{I_{MAX}}{MSE(n)} \tag{18}$$

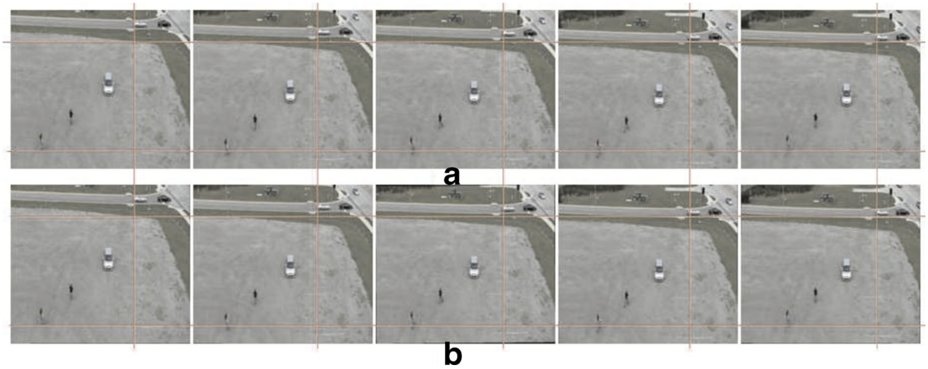


Fig. 10 Result for our stabilized method with one moving object in the scene. **a** Stabilized frame by deshaker stabilizer. **b** Stabilizer frame by our system

$$MSE(n) = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I_N(x,y) - I_{n+1}(x,y)]^2 \qquad (19)$$

Where $MSE(n)$ Mean-Square-Error between frames, I_{MAX} is the maximum intensity value of a pixel and N and M are frame dimensions. The $PSNR$ value for each frame of the

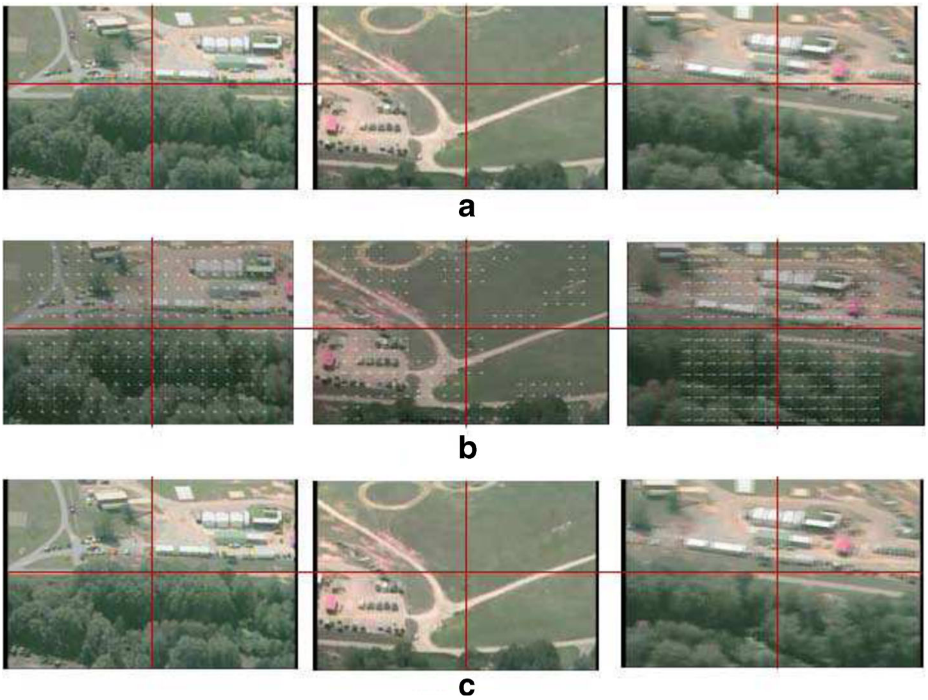


Fig. 11 **a** Input frame from Virat Aerial dataset. **b** Stabilized frame by deshaker stabilizer. **c** Stabilizer frame by our system

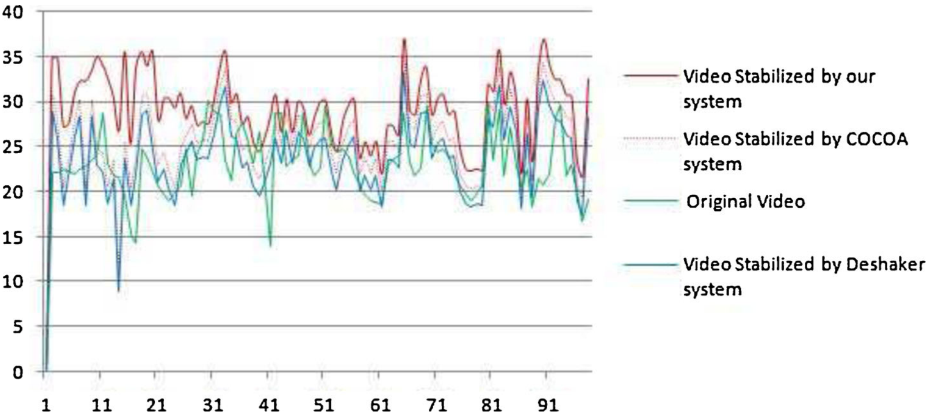


Fig. 12 Graph of the Peak Signal-to-Noise ratio of the original video and the stabilized video by Deshaker, COCOA System and our system

original video and our stabilized version are shown in the graph in Fig. 12 Higher PNSR between two stabilized frames represents good quality of stabilized video.

We find that our stabilization system is working well especially in the case where video sequences include moving objects. Due to the accuracy of detected keypoints and the use of adaptive RANSAC to remove outliers, our system gives a good results compared to COCOA system.

The measurement of Interframe Transformation Fidelity (ITF) is defined as

$$ITF = \frac{1}{N_{frame} - 1} \sum_{k=1}^{N_{frame}-1} PSNR(k) \tag{20}$$

ITF (Inteframe Transformation Fidelity) is defined as the average of the PSNR between two consecutive frames. In general, this average is used to obtain a rough estimation of the quality of the stabilized video in a single value. Like PSNR, upper ITF values represent super quality video stabilization. ITF values for five tested sequences are shown in Table 1. This evaluation illustrates that the ITF of our stabilized videos is superior to the ITF of the original videos.

The ITF of our stabilized videos increases, which is fairly acceptable. We also observe that our proposed system is better than Deshaker system and COCOA System for all tested sequences.

Table 1 ITF comparison

Sequence	Original ITF	Stabilized ITF		
		Proposed system	Deshaker system	COCOA System
Sequence 1	18.321	20.123	19.127	19.321
Sequence 2	17.876	19.542	19.210	19.452
Sequence 3	16.963	18.432	17.385	18.032
Sequence 4	17.856	19.874	18.985	19.432
Sequence 5	19.998	20.653	20.128	20.763

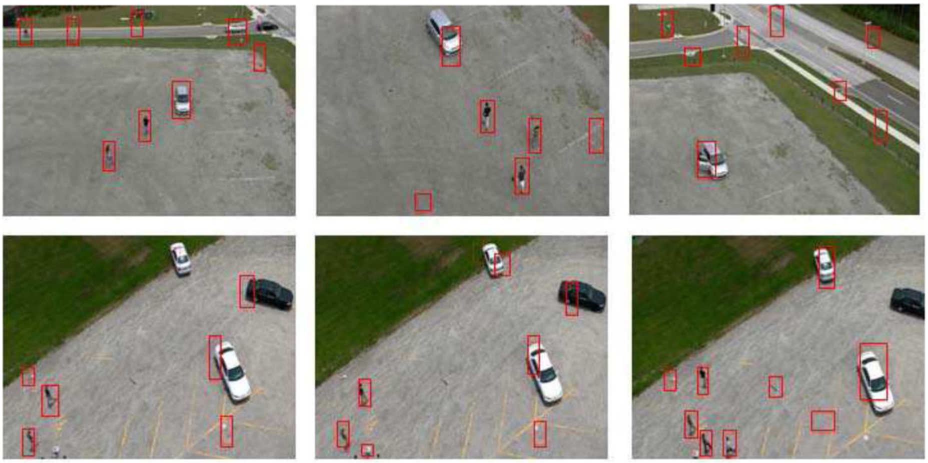


Fig. 13 Moving object detection after stabilization

5.3 Moving object detection evaluation

To evaluate the performance of moving object detection task, our tests were run on a number of real aerial video sequences with various contents. Aerial video include cars, buildings and people moving around a large open area. The test sequence obtained by UCF aerial data set consists of 1000 frames with a resolution of 960 × 540 pixels. Along the sequence, 43 moving objects appear including several splits and different situations.

5.3.1 Moving object detection after stabilization

To prove that moving object detection after stabilization doesn't work well, we test our subsystem moving object detection using stabilized aerial video. Figure 13 shows the results under different conditions in the video. The moving object is identified with a red rectangle. From the results, we can see that moving object can be successfully detected with different backgrounds. But we find a failure in the detection process. To evaluate the performance of this method, we used Detection Ratio(*DR*) and False Alarm Ratio *(FAR)*. In (21) and (22)

Table 2 Quantitative analysis of detection & tracking task after stabilization

Video stream	Moving object	Detection	DR	FAR
Video 1	2	4	1	0.2
Video 2	4	7	1	0.5
Video 3	3	7	1	0.6
Video 4	5	12	1	0.8
Video 5	7	14	1	0.34

TP is true positives of moving objects, *FP* is false positives of moving objects and *FN* is false negatives (not detected). Results are shown in Table 2.

$$DR = TP/(TP + FN) \tag{21}$$

$$FAR = FP/(TP + FP) \tag{22}$$

As shown in Table 2, the FAR is high because in the process of stabilization, we lose a lot of information related to motion vector of mobile objects. These motion vector are useful to distinguish between moving object and static object. Results of our proposed system are illustrated in the next subsection. These results are fairly improved.

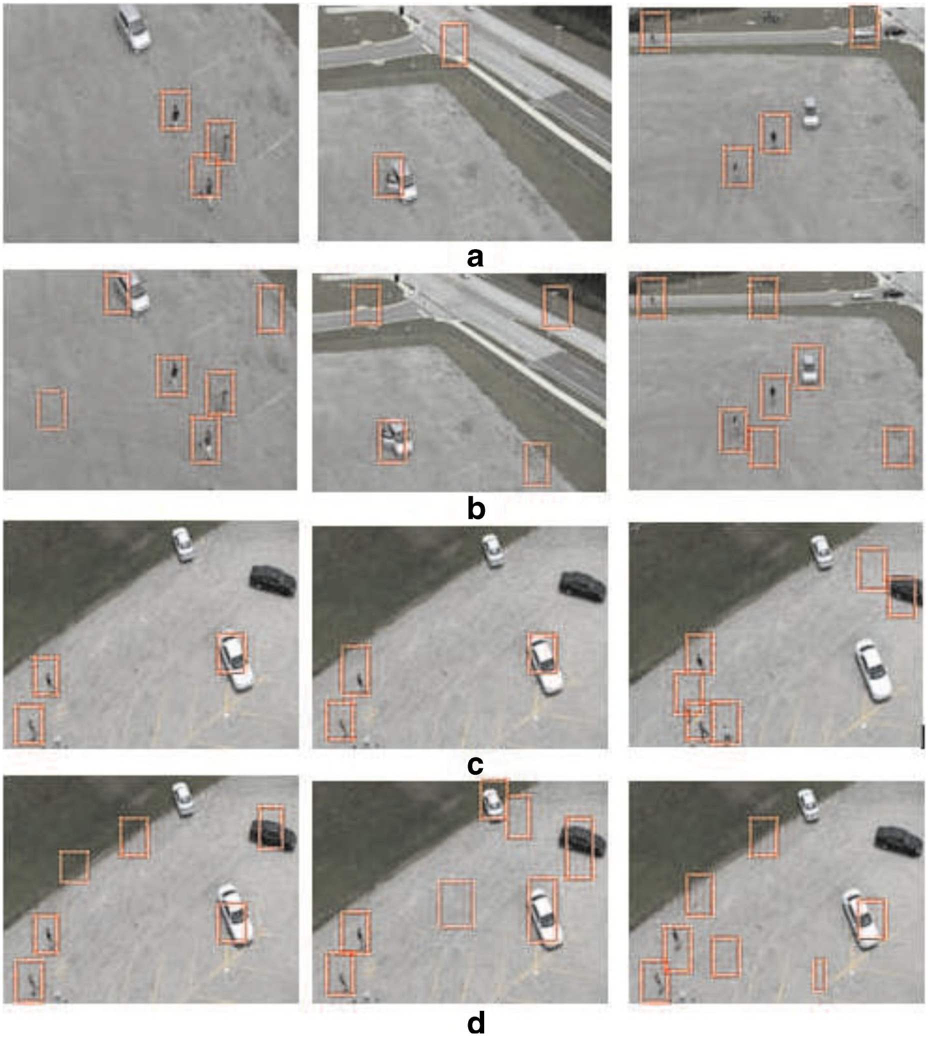


Fig. 14 Detection result of moving object detection. **a, c** Moving object detection by our system. **b, d** Moving object detection by COCOA system

Table 3 Quantitative analysis of detection & tracking task in the process of stabilization

Video stream	Moving object	Detection	DR	FAR
Our system				
Video 1	2	3	1	0.33
Video 2	4	5	1	0.16
Video 3	3	6	1	0.56
Video 4	5	5	1	0.70
Video 5	7	10	1	0.23
COCOA system				
Video 1	2	7	0.65	0.53
Video 2	4	9	0.78	0.89
Video 3	3	9	0.87	0.91
Video 4	5	7	0.98	0.50
Video 5	7	13	0.87	0.54

5.3.2 Moving object detection in the process of stabilization

In order to illustrate the performance of our system, we present the results in two figures: in Fig. 14, we show some detection results issued from our system and COCOA System in the presence of the challenges of background component changes, illumination changes and noise. Our tests were run on the typical video presented in the previous subsection.

For the quantitative analysis of our results we used two metrics: DR and FAR. Table 3 illustrates the performance of our system. Our system has the highest rates of DR and the lowest rate in FAR.

6 Conclusions and future works

In this paper, we propose to integrate the moving object detection into the stabilization algorithm and to demonstrate that detection after stabilization doesn't work well. We also demonstrates that SIFT as features are robust for video stabilization and moving object detection purposes. We uses this feature points to perform the feature extraction and matching process for camera motion estimation. Evaluation of commonly used SIFT keypoints demonstrates their effectiveness on a wide range of test sequences. By using SIFT point extraction and matching, we can locate regions of the image where a residual motion occurs. Another contribution in this paper is that we applied Kalman filtering on this moving region and not on the whole image in order to estimate the motion of the region. Our future work will focus on the following aspects to improve our method:

- To increase the accuracy of matching point, color information can be involved for a robust point matching strategy. This will help the affine transform estimation;
- More local and global features, such as object contour and geometrical relationship, can be applied to trade of noise and significant image distortion. A different descriptor for feature point has to be constructed for this purpose. However, for above mentioned improvement, we have to balance between the processing speed and algorithm complexity and robustness.

References

1. Ali S, Shah M (2006) Cocoa: tracking in aerial imagery pp 62,090D–62,090D–6. doi:10.1117/12.667266
2. Battiato S, Gallo G, Puglisi G, Scellato S, Catania SSD Sift features tracking for video stabilization
3. Bay H, Tuytelaars T, Gool LV (2006) Surf: speeded up robust features. In: ECCV. pp 404–417
4. Calonder M, Lepetit V, Strecha C, Fua P (2010) Brief: binary robust independent elementary features. In: Computer vision ECCV 2010. Lecture notes in computer science, vol 6314. Springer, Berlin Heidelberg, pp 778–792
5. Censi A, Fusiello A, Roberto V (1999) Image stabilization by features tracking. In: Proceedings international conference on image analysis and processing, pp 665–667. doi:10.1109/ICIAP.1999.797671
6. Clark D, Vo BN, Bell J (2006) GM-PHD filter multitarget tracking in sonar images. In: Society of photo-optical instrumentation engineers (SPIE) conference series, society of photo-optical instrumentation engineers (SPIE) conference series, vol 6235. doi:10.1117/12.663522
7. Cuntoor N, Basharat A, Perera A, Hoogs A (2010) Track initialization in low frame rate and low resolution videos. In: 2010 20th international conference on pattern recognition (ICPR). pp 3640–3644. doi:10.1109/ICPR.2010.888
8. Daum F (1996) Multitarget-multisensor tracking: principles and techniques [book review]. IEEE Aerosp Electron Syst Mag 11(2):41. doi:10.1109/MAES.1996.484305
9. Erturk S (2001) Image sequence stabilisation: motion vector integration (mvi) versus frame position smoothing (fps). In: Proceedings of the 2nd international symposium on image and signal processing and analysis, ISPA 2001. pp 266–271. doi:10.1109/ISPA.2001.938639
10. Erturk S (2003) Digital image stabilization with sub-image phase correlation based global motion estimation. IEEE Trans Consum Electron 49(4):1320–1325. doi:10.1109/TCE.2003.1261235
11. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 24(6):381–395. doi:10.1145/358669.358692
12. Freudenberg J, Middleton R, Braslavsky J (2007) Stabilization with disturbance attenuation over a Gaussian channel. In: 2007 46th IEEE conference on decision and control. pp 3958–3963. doi:10.1109/CDC.2007.4434535
13. Huang CH, Wu YT, Kao JH, Shih MY, Chou CC (2010) A hybrid moving object detection method for aerial images, vol 1, pp 357–368
14. Lin CC, Wolf M (2010) Detecting moving objects using a camera on a moving platform. In: 2010 20th international conference on pattern recognition (ICPR). pp 460–463. doi:10.1109/ICPR.2010.121
15. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
16. Miller A, Babenko P, Hu M, Shah M (2008) Multimodal technologies for perception of humans. Springer-Verlag, Berlin, Heidelberg, pp 215–220
17. Rotation invariant fast features for large-scale recognition, vol 8499 (2012). doi:10.1117/12.945968
18. Roujol S, de Senneville BD, Hey S, Moonen CTW, Ries M (2012) Robust adaptive extended Kalman filtering for real time mr-thermometry guided hifu interventions. IEEE Trans Med Imaging 31(3):533–542
19. Rublee E, Rabaud V, Konolige K, Bradski G (2011) Orb: an efficient alternative to sift or surf. In: 2011 IEEE international conference on computer vision (ICCV). pp 2564–2571. doi:10.1109/ICCV.2011.6126544
20. Rudol P, Doherty P (2008) Human body detection and geolocalization for uav search and rescue missions using color and thermal imagery. In: Aerospace conference, 2008 IEEE. pp 1–8. doi:10.1109/AERO.2008.4526559
21. Shen Y, Guturu P, Damarla T, Buckles B, Namuduri K (2009) Video stabilization using principal component analysis and scale invariant feature transform in particle filter framework. IEEE Trans Consum Electron 55(3):1714–1721. doi:10.1109/TCE.2009.5278047

22. Teutsch M, Kruger W (2012) Detection, segmentation, and tracking of moving objects in uav videos. In: 2012 IEEE 9th international conference on advanced video and signal-based surveillance (AVSS). pp 313–318. doi:10.1109/AVSS.2012.36
23. Walha A, Wali A, Alimi AM (2013) Moving object detection system in aerial video surveillance. In: Advanced concepts for intelligent vision systems. Springer International Publishing, pp 310–320
24. Walha A, Wali A, Alimi AM (2013) Video stabilization for aerial video surveillance. AASRI Procedia 4:72–77
25. Wali A, Alimi A (2010) Incremental learning approach for events detection from large video dataset. In: 2010 7th IEEE international conference on advanced video and signal based surveillance (AVSS). pp 555–560. doi:10.1109/AVSS.2010.54
26. Wang Y, Zhang Z, Wang Y (2012) Moving object detection in aerial video. In: 2012 11th international conference on machine learning and applications (ICMLA), vol 2. pp 446–450. doi:10.1109/ICMLA.2012.206
27. Xu L, Lin X (2006) Digital image stabilization based on circular block matching. IEEE Trans Consum Electron 52(2):566–574. doi:10.1109/TCE.2006.1649681
28. Yalcin H, Black MJ, Collins R, Hebert M (2005) A flow-based approach to vehicle detection and background mosaicking in airborne video
29. Yang J, Schonfeld D, Mohamed M (2009) Robust video stabilization based on particle filter tracking of projected camera motion. IEEE Trans Circ Syst Video Tech 19(7):945–954. doi:10.1109/TCSVT.2009.2020252
30. Yang SH, Jheng FM (2006) An adaptive image stabilization technique. In: IEEE International conference on systems, man and cybernetics, 2006. SMC ’06, vol 3. pp 1968–1973. doi:10.1109/ICSMC.2006.385019
31. Yang Y, Liu F, Wang P, Luo P, Liu X (2012) Vehicle detection methods from an unmanned aerial vehicle platform. In: 2012 IEEE international conference on vehicular electronics and safety (ICVES). pp 411–415. doi:10.1109/ICVES.2012.6294294

Ahlem Walha was born on 1985 in Sfax, she is a PhD. student of Computer Science in the National Engineering School of Sfax (ENIS), since September 2011. She received the M.S. degree in 2010 from ENIS. She is currently a member of the REsearch Group on Intelligent Machines (REGIM). His research interests include Computer Vision, Video surveillance analysis. She is a graduate PhD member of IEEE.

Ali Wali Got his PhD. in Engineering Computer Systems at National school of Engineers of Sfax, in 2013. He is member of the REsearch Groups on Intelligent Machines (REGIM). His research interests include Computer Vision and Image and video analysis. These research activities are centered around Video Events Detection and Pattern Recognition. He is a Graduate member of IEEE. He was the member of the organization committee of the International Conference on Machine Intelligence ACIDCA-ICMI 2005, Third IEEE International Conference on Next Generation Networks and Services NGNS2011 and 4th International Conference on Logistics LOGISTIQUA 2011, International Conference on Advanced Logistics and Transport (ICALT' 2013).

Adel M. Alimi (S'91, M'96, SM'00). He graduated in Electrical Engineering in 1990. He obtained a PhD and then an HDR both in Electrical & Computer Engineering in 1995 and 2000 respectively. He is full Professor in Electrical Engineering at the University of Sfax since 2006. Prof. Alimi is founder and director of the REGIM-Lab. on intelligent Machines. He published more than 300 papers in international indexed journals and conferences, and 20 chapters in edited scientific books. His research interests include applications of intelligent methods (neural networks, fuzzy logic, evolutionary algorithms) to pattern recognition, robotic systems, vision systems, and industrial processes. He focuses his research on intelligent pattern recognition, learning, analysis and intelligent control of large scale complex systems. He was the advisor of 24 PhD. thesis. He is the holder of 15 Tunisian patents. He managed funds for 16 international scientific projects. Prof. Alimi served as associate editor and member of the editorial board of many international scientific journals (e.g. IEEE Trans. Fuzzy Systems, Pattern Recognition Letters, NeuroComputing, Neural Processing Letters, International Journal of Image and Graphics, Neural Computing and Applications, International Journal of Robotics and Automation, International Journal of Systems Science, etc.). He was guest editor of several special issues of international journals (e.g. Fuzzy Sets & Systems, Soft Computing, Journal of Decision Systems, Integrated Computer Aided Engineering, Systems Analysis Modelling and Simulations). He organized many International Conferences ISI' 12, NGNS' 11, ROBOCOMP' 11 & 10, LOGISTIQUA' 11, ACIDCA-ICMI' 05, SCS' 04 ACIDCA' 2000. Prof. Alimi has been awarded with the IEEE Outstanding Branch Counselor Award for the IEEE ENIS Student Branch in 2011, with the Tunisian Presidency Award for Scientific Research and Technology in 2010, with the IEEE Certificate Appreciation for contributions as Chair of the Tunisia Computational Intelligence Society Chapter in 2010 and 2009, with the IEEE Certificate of Appreciation for contributions as Chair of the Tunisia Aerospace and Electronic Systems Society Chapter in 2009, with the IEEE Certificate of Appreciation for contributions as Chair of the Tunisia Systems, Man, and Cybernetics Society Chapter in 2009, with the IEEE Outstanding Award for the establishment project of the Tunisia Section in 2008, with the International Neural Network Society (INNS) Certificate of Recognition for contribution on Neural Networks in 2008, with the Tunisian National Order of Merit, at the title of the Education and Science Sector in 2006, with the IEEE Certificate of Appreciation and Recognition of contribution towards establishing IEEE Tunisia Section in 2001 and 2000. He is the Founder and Chair of many IEEE Chapters in Tunisia section. He is IEEE CIS ECTC Education TF Chair (since 2011), IEEE Sfax Subsection Chair (since 2011), IEEE Systems, Man, and Cybernetics Society Tunisia Chapter Chair (since 2011), IEEE Computer Society Tunisia Chapter Chair (since 2010), IEEE ENIS Student Branch Counselor (since 2010), He served also as Expert evaluator for the European Agency for Research. since 2009.