Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2015, Article ID 357191, 12 pages
http://dx.doi.org/10.1155/2015/357191

Research Article

Hybrid Video Stabilization for Mobile Vehicle Detection on SURF in Aerial Surveillance

Gao Chunxian,[1] Zeng Zhe,[2] and Liu Hui[1]

[1]*Department of Communication Engineering, Xiamen University, Xiamen 361005, China*
[2]*College of Geo-Resources and Information, China University of Petroleum, Qingdao 266555, China*

Correspondence should be addressed to Liu Hui; liky781219@163.com

Received 24 September 2014; Revised 4 December 2014; Accepted 5 December 2014

Academic Editor: Muhammad Naveed Iqbal

Detection of moving vehicles in aerial video sequences is of great importance with many promising applications in surveillance, intelligence transportation, or public service applications such as emergency evacuation and policy security. However, vehicle detection is a challenging task due to global camera motion, low resolution of vehicles, and low contrast between vehicles and background. In this paper, we present a hybrid method to efficiently detect moving vehicle in aerial videos. Firstly, local feature extraction and matching were performed to estimate the global motion. It was demonstrated that the Speeded Up Robust Feature (SURF) key points were more suitable for the stabilization task. Then, a list of dynamic pixels was obtained and grouped for different moving vehicles by comparing the different optical flow normal. To enhance the precision of detection, some preprocessing methods were applied to the surveillance system, such as road extraction and other features. A quantitative evaluation on real video sequences indicated that the proposed method improved the detection performance significantly.

1. Introduction

In recent years, analysis of aerial videos has become an important topic [1] with various applications such as intelligence, surveillance, and reconnaissance (ISR), intelligence transportation, and military fields [2, 3]. As an excellent supplement of ground-plane surveillance system, airborne surveillance is more suitable for monitoring fast-moving targets and covers larger area [4]. Mobile vehicles in aerial videos need to be detected for event observation, summarization, indexing, and high level aerial video understanding [5]. This paper is focused on vehicle detection from a low altitude aerial platform (about 120 m above ground).

Detection of objects has traditionally been a very important research topic in classical computer vision [6, 7]. However, there are still some challenges related to detection with low resolution aerial videos. Firstly, vehicles in aerial video have small size and low resolution. Lack of color, low contrast between vehicles and backgrounds, and small and variable vehicle sizes (400~550 pixels) make the appearance and size of vehicle not very distinct to arouse correspondence. On the other hand, frame and background modeling usually assume static background and consistent global illumination. However, in practice, changes of background and global illumination are common in aerial videos due to motion of the global camera. Moreover, UAV video analysis requires real-time processing. Therefore, fast and robust detection algorithm is strongly desired. So far, detection of moving vehicle is still a big challenge.

In this work, a vehicle detection method was proposed based on the method of VSAM by Cohen and Medioni [8]. The similarity and difference of these two methods were discussed in detail. We used Speeded Up Robust Feature (SURF) for video stabilization and demonstrated its validity. The scene context such as road in mobile vehicle detection was introduced, and good results were obtained. Also, complementary features such as shape were used to achieve well-performed detection.

This paper is organized as follows. Section 2 enumerates related work on vehicle detection from aerial videos. Section 3 describes the details about the proposed approach. Section 4 presents our experimental results. Conclusions of this work are summarized in Section 5.

2. Related Work

In the literature, some approaches have been proposed to deal with vehicle detection in airborne videos. However, they mostly tackle stationary camera scenarios [9–11]. Recently, there has been an increasing interest in studying the mobile vehicle detection from moving cameras [12]. Background subtraction technique is one of the most successful approaches to extract moving objects [13, 14]. However, they have limitation that they are only applicable with the stationary cameras in fixed fields of view. Detection of moving objects with moving cameras has been researched to overcome this limitation.

As for moving object detection in video captured by moving camera, the most typical method for detecting moving objects with mobile cameras is the extension of background subtraction method [15, 16]. In these methods, panoramic background models are constructed by applying various image registration techniques [17] to input frames and the position of current frame in panoramas if found by image matching algorithms. Then, moving objects are segmented in a similar way to the fixed camera case. Cucchiara et al. [15] built background mosaic considering internal parameters of cameras. However, camera internal parameters are not always available. Shastry and Schowengerdt [18] proposed a frame-by-frame video registration technique using a feature tracker to automatically determine control-point correspondences. This converts the spatiotemporal video into temporal information, thereby correcting for airborne platform motion and attitude errors. However, digital elevation map (DEM) is not always available. In this work different types of motion model are used, none consider registration error by parallax effect.

The second method to detect moving objects with moving camera is optical flow [2, 19, 20]. The main concept proposed in [2] is to create an artificial optical flow field by estimating the camera motion between two subsequent video frames. Then, this artificial flow is compared with the real optical flow directly calculated from the video feed. Finally, a list of dynamic pixels is obtained and then grouped into dynamic objects. Yalcin et al. [19] propose a Bayesian framework for detecting and segmenting moving objects from the background, based on statistical analysis of optic flow. In [20] 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. The methods of optic flow need calculation of the field of optic flow first which is sensitive to noise and cannot get a precise result; meanwhile, it is not proper to detect real-time moving vehicles.

Recently, appearance feature based classification is used widely in vehicle detection [3, 4]. Shi et al. [3] proposed a moving vehicle detection method based on a cascade of support vector machine (SVM) classifiers. Shape and histogram of orientated gradient (HOG) features are fused to training SVM for classifying vehicles and nonvehicles. Cheng et al. [4] proposed a pixelwise feature classification method for vehicle detection using dynamic Bayesian network (DBN). These approaches are promising. However, the effectiveness of methods depends on the selected feature. For example, color feature of each pixel in [4] is extracted by new color transformation in [21]. However, the new color transformation only considers the difference between vehicle color and road color and does not take similar color among vehicle color, building color, and road color (Figures 9(a2) and 9(b1)). Moreover, the fact that a number of positive and negative training samples need to be collected to train the SVM for vehicle classification is another concern.

In this paper, we designed a new vehicle detection framework that preserves the advantages of the existing works and avoids their drawbacks. The modules of the proposed system framework are illustrated in Figure 1. It is two-stage object detection: initial vehicle detection and refined vehicle detection with scene context and complementary features. The whole framework can be roughly divided into three parts, which are video stabilization, initial vehicle detection, and refined vehicle detection. Video stabilization is used to eliminate camera vibration and noise with SURF feature extraction. Initial vehicle detection is used to find the candidate motion region with optical flow normal. Performing background color removal can not only reduce false alarms and speed up the detection process but also facilitate the road extraction. The initial vehicle detections are refined by using the road context and complementary features such as size of the candidate region. The whole process is proceeding online and iteratively.

3. Hybrid Method for Moving Vehicle Detection

Here, we elaborate each module of the proposed system framework in detail. We compensated the ego motion of airborne vehicle by SURF [22] feature point based image alignment on consecutive frames and then applied an optical flow normal method to detect the pixels with motion. Pixels with high optical flow normal value were grouped as candidates of mobile vehicles. Meanwhile, the features such as size were used to improve the detection accuracy.

3.1. SURF Based Video Stabilization. Registration is the process of establishing correspondences between images, so that the images are in a common reference frame. Aerial images are achieved with a moving airborne platform, and large camera motion exists between consecutive frames; thus sequence stabilization is essential for motion detection. Global camera motion is eliminated or reduced by the process of image registration. For registration, descriptors such as SURF or SIFT (scale invariant feature transform) [23] can be used. In particular, SURF features were exploited due to its efficiency.

3.1.1. SURF Feature Extracting and Matching. The selection of features for motion estimation is very important, since unstable features may produce unreliable estimations with variations in rotation, scaling, or illumination. SURF is a robust image interest point detector, first presented by Bay et al. [22]. SURF descriptor is similar to the gradient information extracted by SIFT. SURF algorithm includes two main parts: the feature point detection and feature point

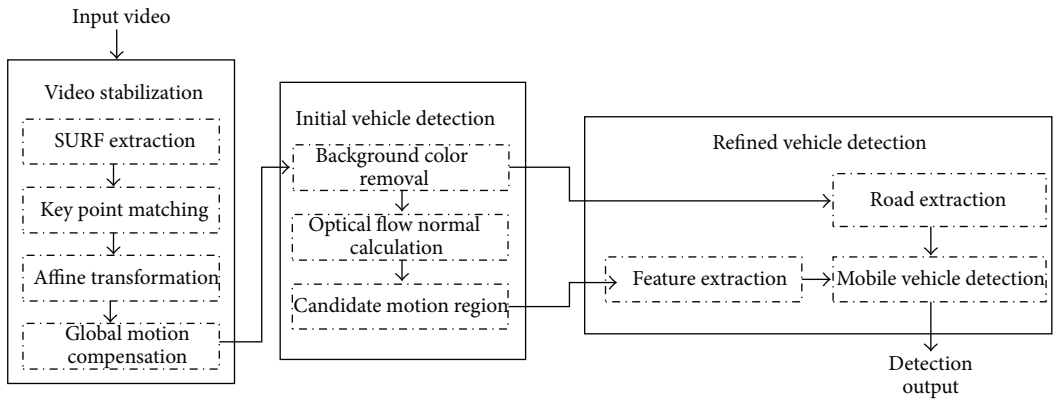


Figure 1: Overview of the proposed method.

description. But in the whole process, using fast Hessian matrix to detect feature points and introducing the integral image and the box filter to compute approximations of the Laplacian of Gaussians improve the efficiency of the algorithm. SURF has similar performance to SIFT; however, it is faster. An example of SURF is shown in Figures 4(a) and 4(b).

3.1.2. Feature Point Detection. The integral images allow for fast computation of box filters. The entry of an integral image $I_\Sigma(X)$ at a location $X = (x, y)$ represents the sum of all pixels in the input image I with a rectangular region formed by the origin and X:

$$I_\Sigma(X) = \sum_{i=0}^{i\le x}\sum_{j=0}^{j\le y} I(i,j). \tag{1}$$

Once the integral image has been computed, it takes three additions to calculate the sum of the intensities. $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$, and $D(x_4, y_4)$ are assumed to be four points, respectively, of the rectangular area shown in Figure 2. Hence, the sum of all pixels in the black rectangular area can be expressed by $I_\Sigma(A)+I_\Sigma(D)-(I_\Sigma(C)+I_\Sigma(B))$. The calculation time is independent of its size. This is important in SURF algorithm.

Then SURF uses the Hessian matrix to detect feature points, for a point $X = (x, y)$ in the image marked in the scale σ on Hessian matrix is defined as

$$H(X,\sigma) = \begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix}. \tag{2}$$

In formula (2), $L_{xx}(X,\sigma)$ means the convolution result of the point in the image and the Gaussian filter second order partial derivative $\partial G(X,\sigma)/\partial x^2$, and the calculation methods in $L_{xy}(X,\sigma)$ and $L_{yy}(X,\sigma)$ are similar.

In order to reduce the workload of calculation, SURF uses the box filters to replace, respectively, L_{xx}, L_{xy}, and L_{yy} with the convolution of the original input images D_{xx}, D_{xy}, and D_{yy}. The calculations are shown in Figure 3 and formula (3).

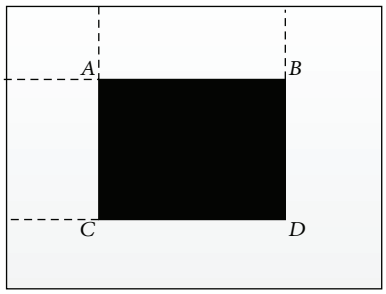


Figure 2: Integral image.

In Figure 3, the weight of black pixel is −2 and white pixel is 1. The formula of D_{xx}, D_{xy}, and D_{yy} calculations using integral image is shown as follows:

$$\begin{aligned} D_{xx} &= (r-2\times b, c-l-b, r+2\times b, c+l+b) \\ &\quad -3\times(r-2\times b, c-b, r+2\times b, c+b) \\ D_{yy} &= (r-l-b, c-2\times b, r+l+b, c+2\times b) \\ &\quad -3\times(r-b, c-2\times b, r+b, c+2\times b) \\ D_{xy} = D_{yx} &= (r-l, c-l, r-1, c-1) \\ &\quad +(r+1, c+1, r+l, c+l) \\ &\quad -(r-l, c+1, r-1, c+l) \\ &\quad -(r+1, c-l, r+l, c-1). \end{aligned} \tag{3}$$

In formula (3), (r, c) are the row and column of the pixel in image, respectively, l is 1/3 of the size of box filter, and $B = [l/2]$ ([] is operation of rounding).

The formula H_{approx}, which is the approximation for the Hessian matrix Gaussian calculation determinant matrix, can be illustrated as follows:

$$\det\left(H_{\text{approx}}\right) = D_{xx}D_{yy} - \left(0.9D_{xy}\right)^2. \tag{4}$$

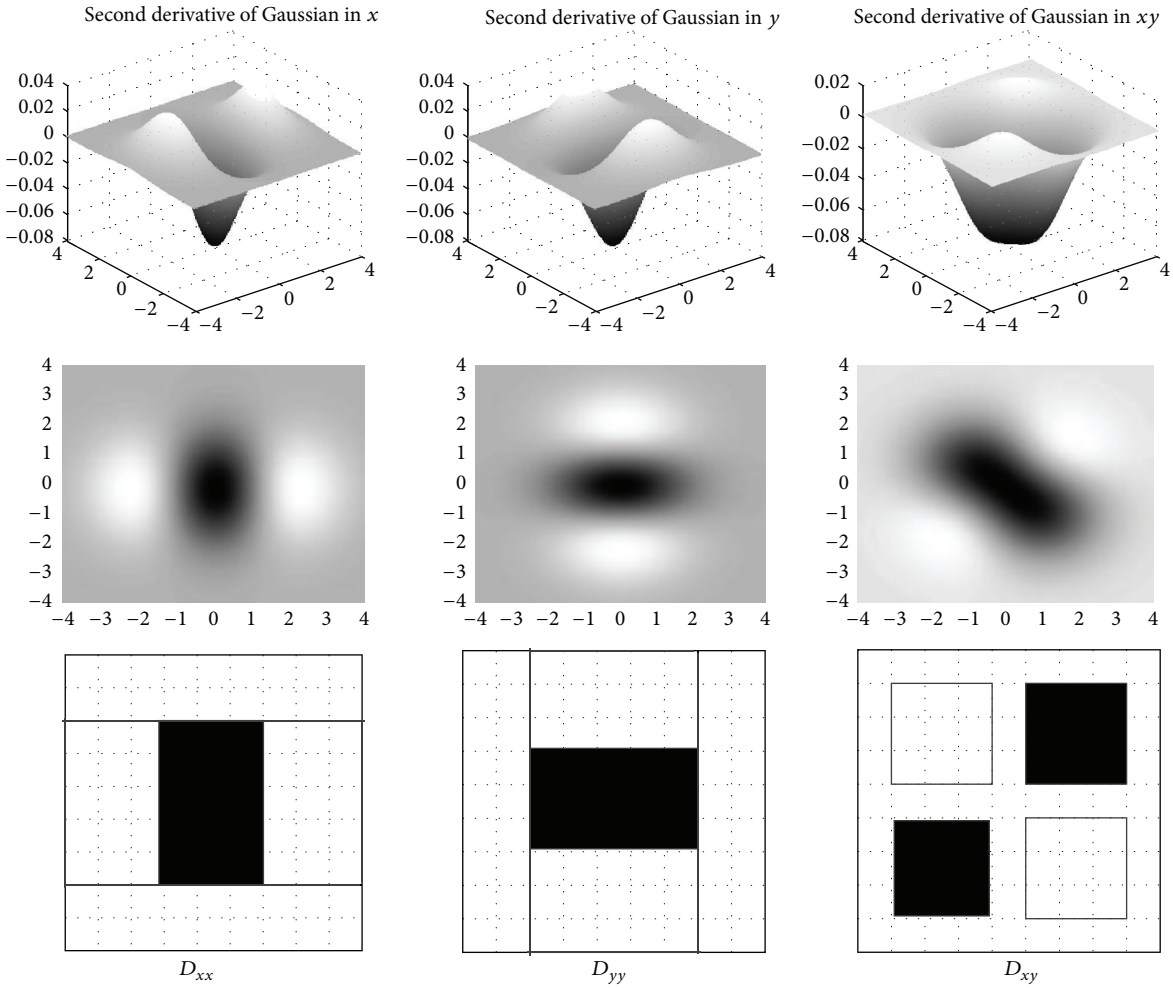


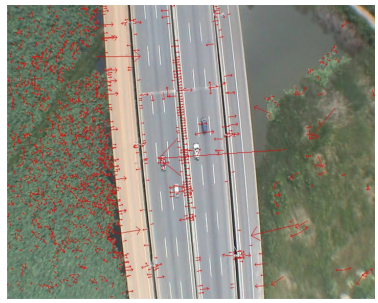
Figure 3: Box filter of 9×9.

By using a nonmaxima suppression method in the neighborhood, the image feature points can be found in different scales.

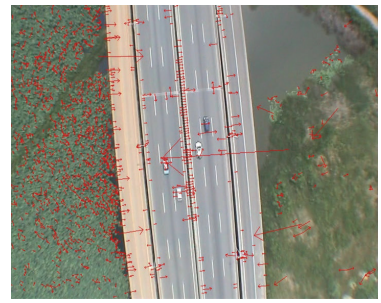
3.1.3. Feature Point Description. In order to be invariant to image rotation, a dominant orientation for each key point is identified first in feature point description. For a key point, Haar wavelet responses in x and y directions are calculated within a circular neighborhood of radius $6s$ around it, where s is the corresponding scale of the detected key point. The Haar wavelet responses can be computed using Haar wavelet filters and integral images. The wavelet responses are then weighted with a Gaussian ($\sigma = 2s$) centered at the key point. The dominant orientation can be estimated by rotating a sliding fan-shaped window of size $\pi/3$. At each position, the horizontal and vertical responses within the sliding window are summed and used to form a new vector. The longest such vector over all windows is assigned as the orientation of the key point.

Then, SURF descriptor is generated in a $20s$ square region centered at the key point and oriented along its dominant orientation. The region is divided into 4×4 square subregions. For each subregion, Haar wavelet responses d_x in horizontal direction and d_y in vertical direction are computed from 5×5 sample points. Then the wavelet responses d_x and d_y are weighted with a Gaussian ($\sigma = 3.3s$) centered at the key point. The responses and their absolute values are summed up over each subregion and form a 4D feature vector ($\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|$). Thus for each key point, this results in a descriptor vector of length $4 \times 4 \times 4 = 64$. Finally, the SURF descriptor is normalized to make it invariant to illumination changes.

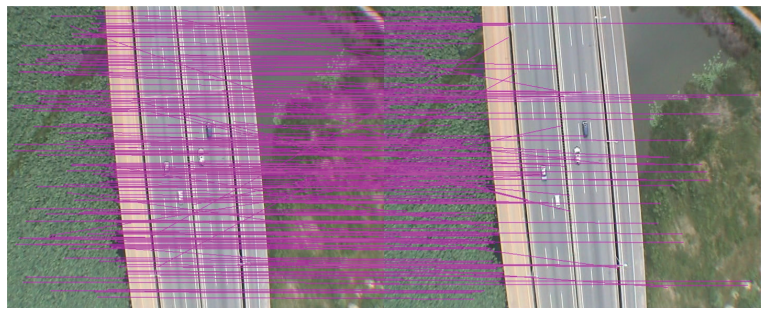
After feature extraction process, it is necessary to match feature point between two successive frames. For this process, we are investigating the matching process as proposed by Lowe [23]. This process is based on finding a match between two consecutive image features using Euclidean distance. The Euclidian distance between SURF 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 SURF model. The example is shown in Figures 4(c) and 4(d).



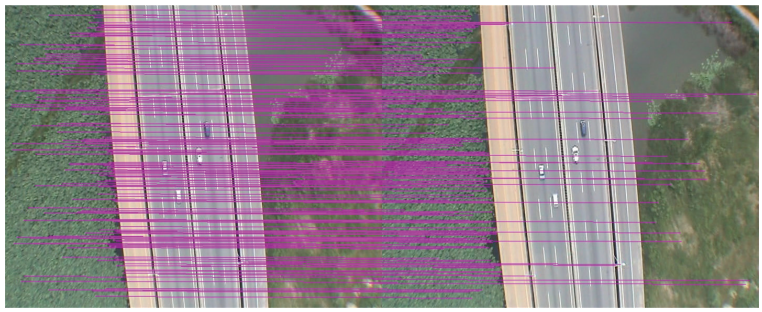
(a) SURF features of 1st frame



(b) SURF features of 2nd frame



(c) Features matching between frame 1 and frame 2



(d) Features matching between frame 1 and frame 2 after RANSAC

(e) Stabilized frame 2

Figure 4: Image registration with SURF key points and image warping.

3.1.4. Motion Detection and Compensation. The temporally and spatially changing video can be modeled as a function $I_t(x, y)$, where (x, y) is the spatial location of a pixel and t is the temporal locator index, within the sequence. The function I can be thought of as representing the pixel intensity at location (x, y) and time t. Thus, this function satisfied the following property:

$$I_{t+\tau}(x, y) = I_t(x - \varepsilon, y - \eta). \tag{5}$$

This means that an image taken at time $t + \tau$ is considered to be shifted from the earlier image by $d = (\varepsilon, \eta)$, called the displacement in time τ. If the pixel is obscured by noise, or if there is an abnormal intensity change due to light reflection by objects, (5) can be redefined as

$$I_{t+\tau}(x, y) = I_t(x - \varepsilon, y - \eta) + n_t(x, y). \tag{6}$$

Using feature matching, we can get the geometric transformation between $I_{t+\tau}$ and I_t. Indeed, let $\Gamma_{t,t+\tau}$ denote the warping t of the image to the reference frame $t + \tau$. And the stabilized image sequence is defined by $I^s_{t+\tau} = I_t(\Gamma_{t,t+\tau})$. The parameter estimation of the geometric transform is done by the minimum mean square error criteria:

$$E = D\sum_t \left(I_t(x, y) - I_t\left(\Gamma_{t+\tau,t}\right)\right)^2. \tag{7}$$

Generally, the geometric transformation between two images can be described by a 2D or 3D homograph model. We adopted four parameters 2D affine motion model to describe geometric transformation between two consecutive frames. If $P(x, y)$ is the point in frame t, and $P'(x', y')$ is the same point in the successive frame, then the transformation from P to P' can be represented as shown in the following:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} S\cos\theta & -S\sin\theta & T_x \\ S\sin\theta & S\cos\theta & T_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{8}$$

or in the form of $Y = AX$. 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 scaling factor, θ is the rotation, and T_x and T_y are the translations in the horizontal and vertical direction, respectively. Corresponding pairs of feature points were used to determine the transform matrix in (1) from two consecutive image frames. Since four unknowns exist in (8), at least three pairs are needed to determine a unique solution. Nevertheless, more matches can be added under least-square criteria to ensure results are more robust:

$$A = \left[X^T X\right]^{-1} X^T Y = \Gamma_{t,t+1}. \tag{9}$$

Then we can compensate the current frame to obtain stable images. Compensation of the video is calculated directly using warping operation. The example is shown in Figure 4(e).

3.2. Vehicle Detection. After removing the undesired motion of camera, the first step of mobile vehicle detection was the initial vehicle detection, which produces the vehicle candidates, including many false alarms.

3.2.1. Normal Flow. The reference frame and the warped one do not, in general, have the same metric since in most cases, the mapping function $\Gamma_{t,t+\tau}$ is not a translation but a 2D affine transform. This change in metric can be incorporated into the optical flow equation associated with the image sequence I_t, in order to detect more accurately candidate mobile vehicle region. From the image brightness constancy assumption [24, 25], the gradient constraint equation selected by Horn and Schunck [24] is

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0, \tag{10}$$

where u and v are the optical flow velocity components and $\partial I/\partial x$, $\partial I/\partial y$, and $\partial I/\partial t$ are the spatial gradients and temporal gradient of image intensity. Equation (10) is written in matrix form:

$$\begin{gathered}\begin{bmatrix} I_x & I_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} = -I_t \Longrightarrow \begin{bmatrix} I_x \\ I_y \end{bmatrix}\begin{bmatrix} I_x & I_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} I_x \\ I_y \end{bmatrix} I_t \\ \nabla I \left(\nabla I\right)^T w = -\nabla I I_t \left(w = \begin{bmatrix} u \\ v \end{bmatrix}, \nabla I = \begin{bmatrix} I_x \\ I_y \end{bmatrix}\right).\end{gathered} \tag{11}$$

The optical flow associated with the image sequence $I_{t+\tau}$ is

$$\nabla I^s_{t+\tau}\left(\nabla I^s_{t+\tau}\right)^T w = -\nabla I^s_{t+\tau}\frac{\partial I^s_{t+\tau}}{\partial t}. \tag{12}$$

Expanding the previous equation we obtain

$$\nabla I_t\left(\Gamma_{t,t+\tau}\right)\left(\nabla I_t\left(\Gamma_{t,t+\tau}\right)\right)^T w = -\nabla I_t\left(\Gamma_{t,t+\tau}\right)\frac{\partial I_t\left(\Gamma_{t,t+\tau}\right)}{\partial t}. \tag{13}$$

According to composite function derivation rules

$$\nabla I_t\left(\Gamma_{t,t+\tau}\right) = \nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right). \tag{14}$$

Expanding (13) we obtain

$$\begin{aligned}&\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\left(\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\right)^T w \\ &\quad = -\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\frac{\left(I_t\left(\Gamma_{t,t+\tau}\right) - I\left(t\right)\right)}{\tau} \\ &\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\left(\nabla I_t\left(\Gamma_{t,t+\tau}\right)\right)^T\left(\nabla\Gamma_{t,t+\tau}\right)^T w \\ &\quad = -\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\frac{\left(I_t\left(\Gamma_{t,t+\tau}\right) - I_t\right)}{\tau}.\end{aligned} \tag{15}$$

And therefore, the normal flow $w_\perp$ is characterized by

$$w_\perp = \frac{-\nabla\Gamma_{t,t+\tau}\nabla I_t\left(\Gamma_{t,t+\tau}\right)\left(\left(I_t\left(\Gamma_{t,t+\tau}\right) - I_t\right)/\tau\right)}{\left\|\nabla\Gamma_{t+\tau,t}\nabla I_t\left(\Gamma_{t+\tau,t}\right)\right\|\left\|\nabla\Gamma_{t+\tau,t}\nabla I_t\left(\Gamma_{t+\tau,t}\right)\right\|}. \tag{16}$$

Although $w_\perp$ does not always characterize image motion, due to the aperture problem, it allows accurate detecting of moving points. The amplitude of $w_\perp$ is larger near moving regions and becomes null near stationary regions. The relation of normal flow and optical flow is shown in Figure 5(a) and the candidate mobile region detection is shown in Figure 5(b).

3.2.2. Context Extraction. Context is especially useful in aerial video analysis, because most of the vehicles move in special area. And road is an effective context information for robust mobile vehicle detection. Many estimate the road network using the scene classification, which needs complicated training and many issues are prepared in advance. Based on human knowledge in general, we can make the following brief description of the road.

(i) Road has constant width along all its length.

(ii) Road always is vertical or horizontal in the airborne videos.

(iii) There are two distinct parallel edges of the road.

(iv) Road is always a connected region area.

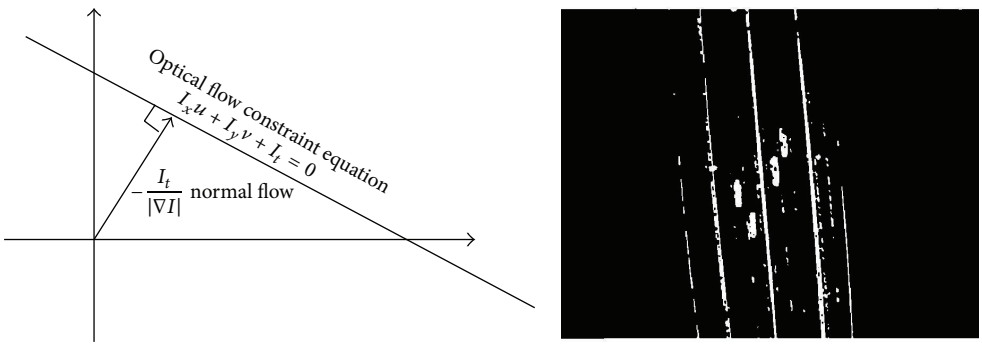
Based on above assumption, we use Canny Edge detection and Hough Transform to extract the road area. The results are shown in Figure 6.

3.2.3. Complementary Features. Initial vehicle detection produces candidate mobile vehicle regions, including many false alarms, shown in Figure 7. We use shape (size) [3] of the candidate motion regions to improve the detection performance. Size feature is a four-dimensional vector, which is represented as (17), where l and w denote the length and width of the object, respectively:

$$f = \left\{l, w, \frac{l}{w}, l * w\right\}. \tag{17}$$

4. Experiment Results and Analysis

We tested our method with three surveillance videos. The first two were got from our own hardware platform, shown in Figure 9(a), named 2.avi and gs.avi, respectively. The other



(a) Relation between normal flow and optical flow (b) Candidate mobile regions

Figure 5: Candidate mobile regions.

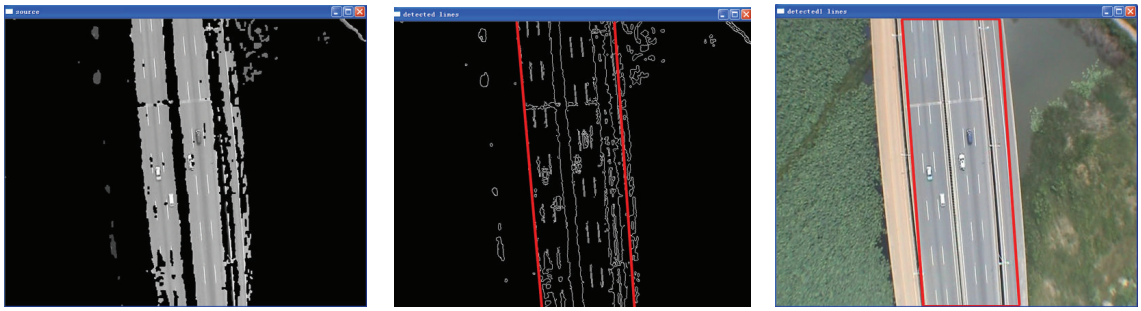


Figure 6: Road extraction results.

Figure 7: Initial vehicle detection results.

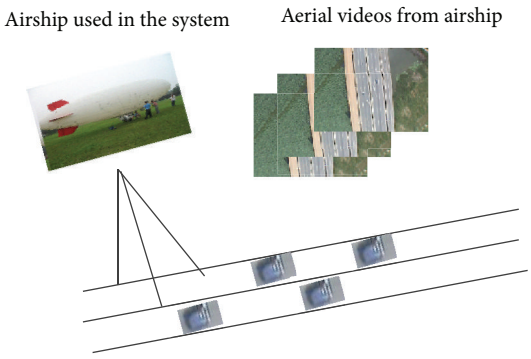


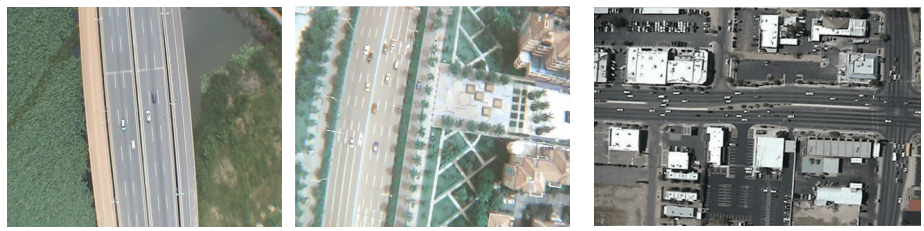
Figure 8: Hardware used in the experiments.

is from the Shastry and Schowengerdt's paper [18], shown in Figure 9(b), named TucsonBlvd_origin.avi. The first two were taken in 25 frames per second with resolution of 720 × 576 pixels on the airship of 120 m height from the ground, where the speed of airship is 30 Km/h, shown in Figure 8.

From the vehicle numbers and background complexity in Figure 9, vehicles contained in (a1) are the least. And the background is simple, which includes no buildings; therefore it cannot cause visual error. The vehicles increase in the (a2), and the background includes buildings, which cause visual error. The most complex video is (b), which not only includes more vehicles and buildings but also has lowest resolution. And experiments' results show that different videos have different detection performance.

The hardware platform of the simulation is CPU 2.1 GHz and RAMS 2 G. The software used in the experiments is opencv1.0 and VC++ 6.0.

4.1. Image Stabilization Comparison between SURF and SIFT. Our first experiment consists of comparing our video stabilization system to [5]. This system is based on SIFT feature extraction. We demonstrated the Speeded Up Robust Feature (SURF) key points are more suitable for the stabilization task. Figure 10 shows five frames of the unstable input sequence corresponding to 1, 2, 5, 10, and 15, taken from 2.avi.

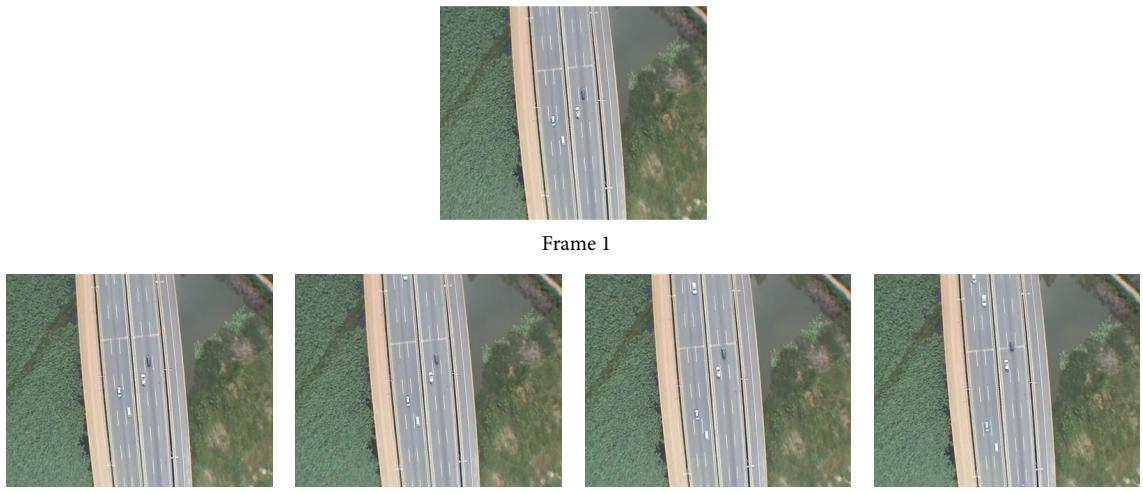


(a1) 2.avi (a2) gs.avi (b1) TucsonBlvd_origin.avi

(a) Aerial videos from Figure 8

(b) Aerial videos from Shastry and Schowengerdt paper [18]

Figure 9: Snapshots of the experimental videos.



Frame 1

Frame 2 Frame 5 Frame 10 Frame 15

Figure 10: Frame used in the image stabilization comparison.

Next, we compute the global motion vector $\Gamma_{1,n}$, shown in Table 1. Table 1 shows that the airplane moves in vertical direction mostly and the accuracy of vector $\Gamma_{1,n}$ is almost the same in two video stabilization methods.

Figures 11(a) and 11(b) show the stabilization result of SIFT and SURF, respectively. Subjectively, our video stabilization system has the same results compared to SIFT.

Then, we used Peak Signal-to-Noise Ratio (PSNR), an error measure, to evaluate the quality of the video stabilization. PSNR between frame 1 and stabilized frame n is defined as

$$\mathrm{PSNR}\left(I_n^s, I_1\right) = 10\log_{10}\frac{255}{\mathrm{MSE}\left(I_n^s, I_1\right)}, \tag{18}$$

where $\mathrm{MSE}(I_n^s, I_1)$, mean square error, between frames N and M is frame dimensions:

$$\mathrm{MSE}\left(I_n^s, I_1\right) = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}\left[I_n^s(x,y) - I_1(x,y)\right]^2. \tag{19}$$

We found that our stabilization system using SURF feature is working well compared to the stabilization system using SIFT feature in Figure 12. For the parallax effect of wrapping operation and multiple moving vehicles, the PSNR is low. So in the mobile vehicle detection, we use the normal optical flow.

Objectively, our video stabilization system has the better results compared to SIFT from Table 1 and Figure 12.

Last, we compare the performance of the two video stabilization methods, shown in Figure 13.

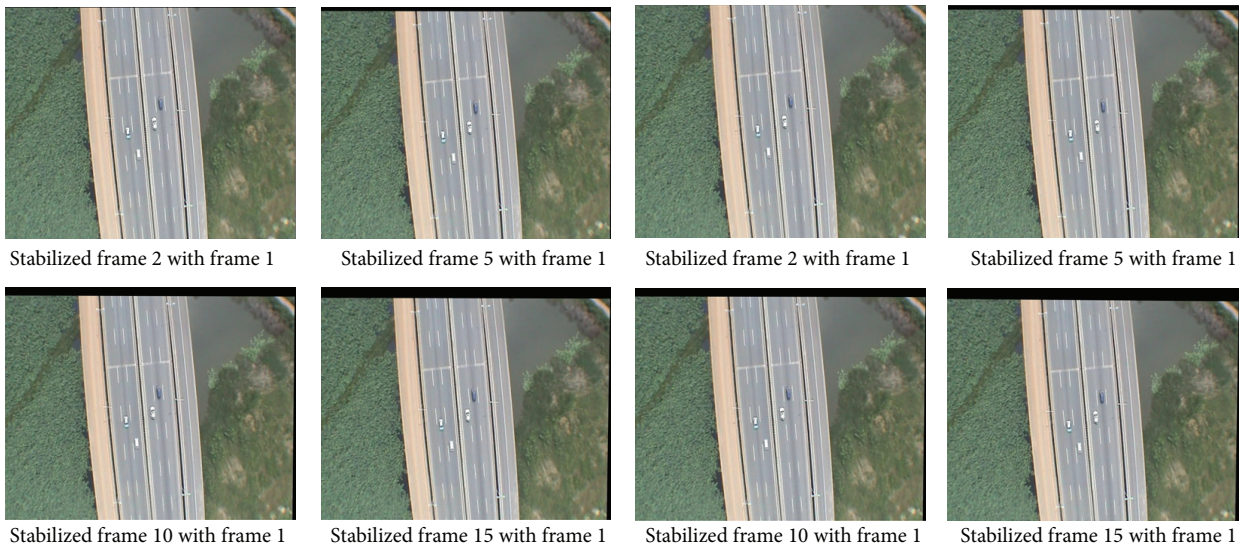
Through the experiments, the image stabilization accuracy is the same in subjective and objective evaluation. And the efficiency of image stabilization on SURF is better than on SIFT. We find that our stabilization system is working well.

4.2. Mobile Vehicle Detection Comparison between Proposed Method and Existing Methods. To evaluate the performance of mobile vehicle detection, our tests were run on a number of real aerial video sequences with various contents. Aerial video includes cars and buildings. Figure 14 shows the results under different conditions in video. The mobile vehicle 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 (20),

Table 1: Global motion parameter comparison of SURF and SIFT using 2D affine transform.

	SURF	SIFT
$\Gamma_{2,1}$	$\Gamma_{2,1} = \begin{bmatrix} 1.00103 & 0.00021 & -0.74070 \\ 0.00119 & 1.00090 & 1.94483 \\ 0 & 0 & 1 \end{bmatrix}$	$\Gamma_{2,1} = \begin{bmatrix} 1.00057 & 0.00054 & -1.12613 \\ 0.00068 & 1.00107 & 2.23921 \\ 0 & 0 & 1 \end{bmatrix}$
$\Gamma_{5,1}$	$\Gamma_{5,1} = \begin{bmatrix} 0.99944 & 0.00130 & -2.297314 \\ 0.00085 & 1.00163 & 11.80027 \\ 0 & 0 & 1 \end{bmatrix}$	$\Gamma_{5,1} = \begin{bmatrix} 1.00365 & 0.00149 & -2.81091 \\ 0.00211 & 1.00420 & 11.32544 \\ 0 & 0 & 1 \end{bmatrix}$
$\Gamma_{10,1}$	$\Gamma_{10,1} = \begin{bmatrix} 1.00553 & 0.00380 & -6.69503 \\ 0.00187 & 1.01297 & 21.52460 \\ 0 & 0 & 1 \end{bmatrix}$	$\Gamma_{10,1} = \begin{bmatrix} 1.00819 & 0.00384 & -6.90752 \\ 0.00384 & 1.01145 & 21.79191 \\ 0 & 0 & 1 \end{bmatrix}$
$\Gamma_{15,1}$	$\Gamma_{15,1} = \begin{bmatrix} 1.00926 & 0.00735 & -8.09136 \\ 0.00206 & 1.01958 & 27.66451 \\ 0 & 0 & 1 \end{bmatrix}$	$\Gamma_{15,1} = \begin{bmatrix} 1.01004 & 0.00795 & -8.25470 \\ 0.00226 & 1.01917 & 27.77338 \\ 0 & 0 & 1 \end{bmatrix}$



(a) Stabilization using SIFT features

(b) Stabilization using SURF features

Figure 11: The 1st frame wrapped with the 2nd frame, 5th frame, 10th frame, and 15th frame, respectively, using SIFT features (a) and SURF features (b).

TP is true positives of mobile vehicles, FP is false positives of mobile vehicles, and FN is false negatives (not detected). Results are shown in Table 2. And Figure 15 shows vehicle detection results comparison of 2.avi by using GMM, LK, and proposed method:

$$\text{DR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \qquad \text{FAR} = \frac{\text{FP}}{\text{TP} + \text{FP}}. \tag{20}$$

For the quantitative analysis of our results we used two metrics: DR and FAR. Table 2 and Figures 14 and 15 illustrate the performance of our system. Because the resolution and complexity of videos are different, the detection performance is different. Our system has the highest rates of DR and the lowest rate in FAR.

5. Conclusion

In this paper, we present a hybrid method to detect mobile vehicle efficiently in aerial videos. We also demonstrate that SURF as features are robust for video stabilization and mobile vehicle detection purpose compared with SIFT. A quantitative evaluation on real video sequences demonstrates that the proposed method improves the detection performance. Our future work will focus on the following aspects to improve our method.

(i) To increase the accuracy of the mobile vehicle, more local and global features, such as color information and gradient distribution, can be applied in the methods.

(ii) We have to balance between the processing speed and algorithm complexity and robustness.

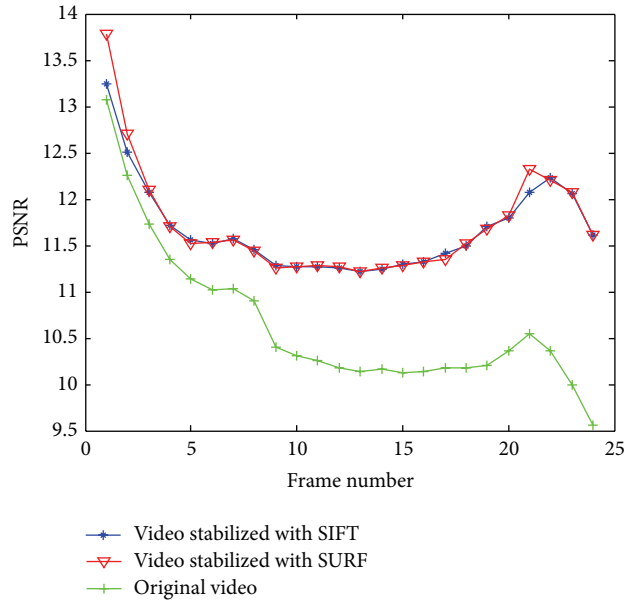


Figure 12: Graph of the Peak Signal-to-Noise Ratio of the original video and the stabilized video.

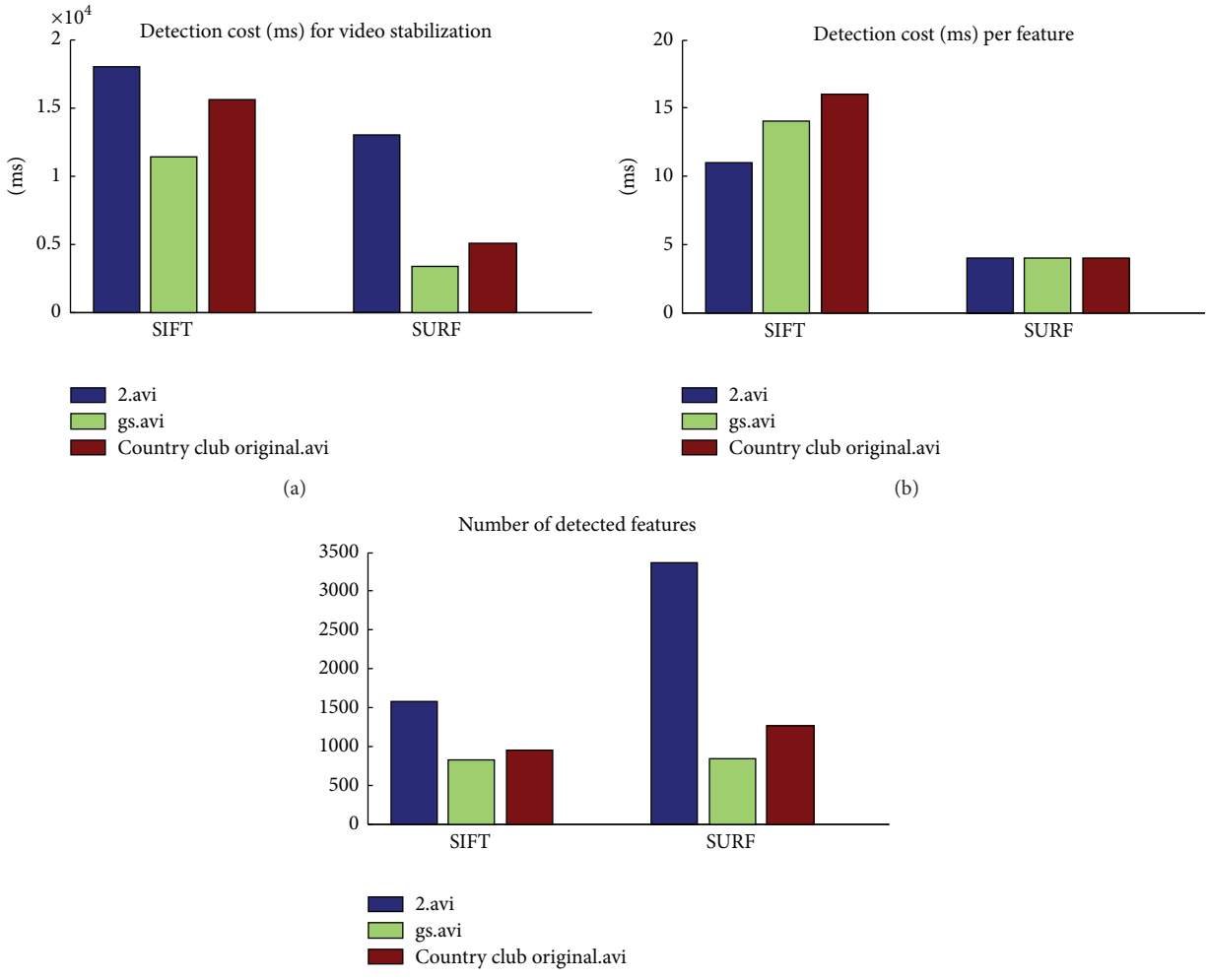


Figure 13: Performance test.

Table 2: Quantitative analysis of detection with other methods.

	DR (%)			FAR (%)		
	2.avi	gs.avi	tucsonBlvd_original.avi	2.avi	gs.avi	tucsonBlvd_original.avi
Our method	94	58	24	6	18	33
GMM method	29	26	1	45	75	79
LK method	20	9	1	91	80	93

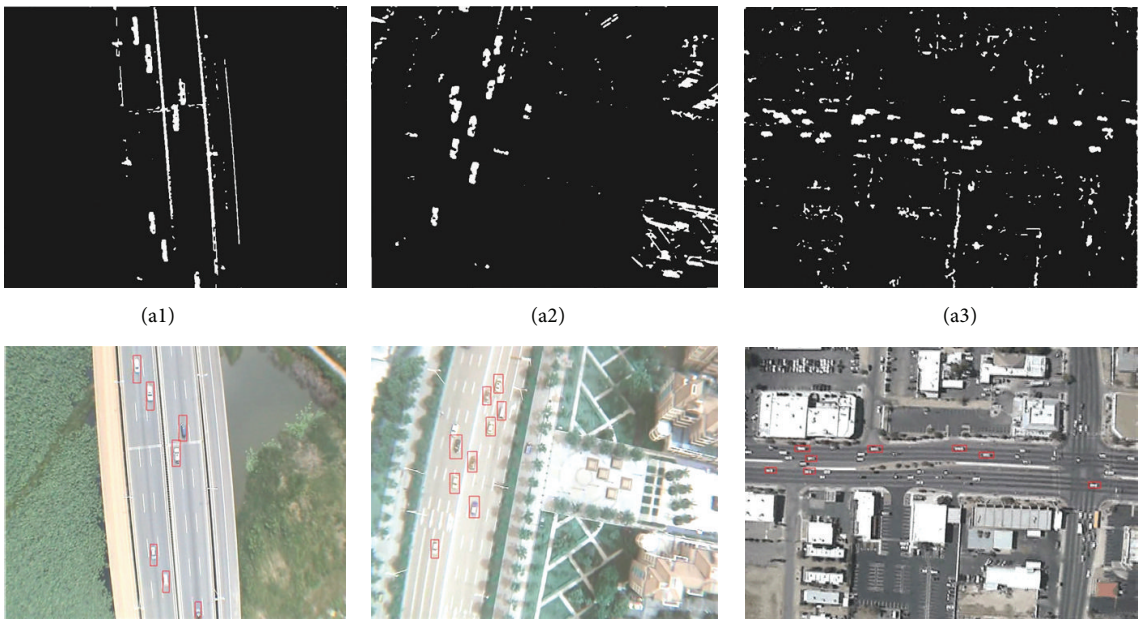


Figure 14: Vehicle detection results of the proposed method.

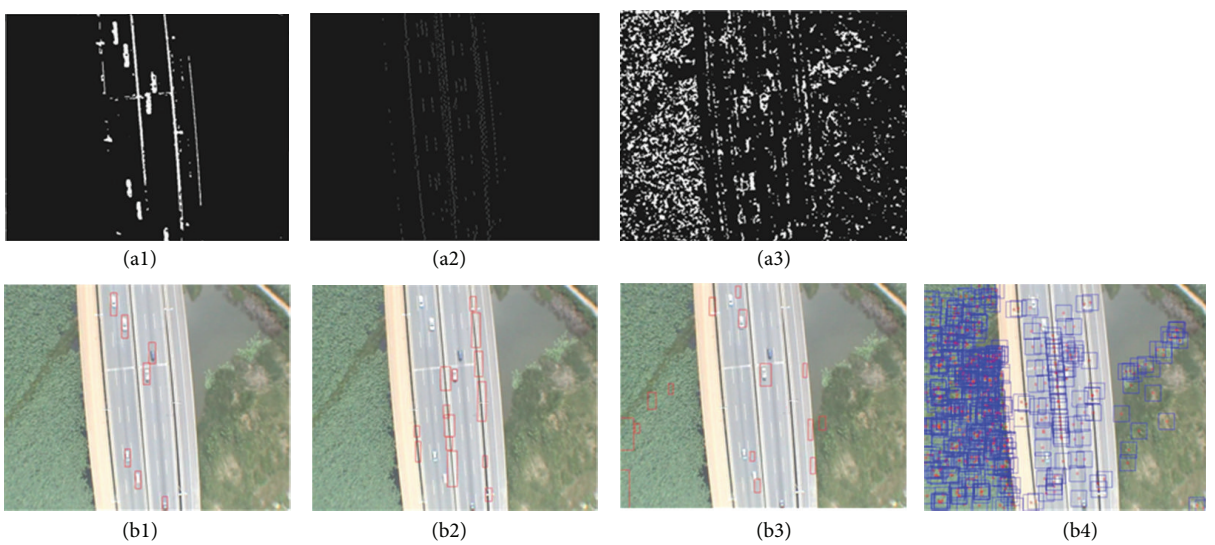


Figure 15: Vehicle detection results of GMM, LK, and proposed method.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to express their sincere thanks to the anonymous referees and editors for their time and patience devoted to the review of this paper. This work is partially supported by NSFC Grant (no. 41101355).

References

[1] R. Kumar, H. Tao, Y. Guo et al., "Aerial video surveillance and exploitation," *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1518–1538, 2001.

[2] G. R. Rodríguez-Canosa, S. Thomas, J. del Cerro, A. Barrientos, and B. MacDonald, "A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera," *Remote Sensing*, vol. 4, no. 4, pp. 1090–1111, 2012.

[3] X. Shi, H. Ling, E. Blasch, and W. Hu, "Context-driven moving vehicle detection in wide area motion imagery," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR '12)*, pp. 2512–2515, Tsukuba, Japan, November 2012.

[4] H.-Y. Cheng, C.-C. Weng, and Y.-Y. Chen, "Vehicle detection in aerial surveillance using dynamic Bayesian networks," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2152–2159, 2012.

[5] A. Walha, A. Wali, and A. M. Alimi, "Video stabilization with moving object detecting and tracking for aerial video surveillance," *Multimedia Tools and Applications*, 2014.

[6] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, article 13, 2006.

[7] I. Saleemi and M. Shah, "Multiframe many-many point correspondence for vehicle tracking in high density wide area aerial videos," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 198–219, 2013.

[8] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, Fort Collins, Colo, USA, June 1999.

[9] M. Chakroun, A. Wali, and A. M. Alimi, "Multi-agent system for moving object segmentation and tracking," in *Proceedings of the 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '11)*, pp. 424–429, September 2011.

[10] A. Wali and A. M. Alimi, "Incremental learning approach for events detection from large video dataset," in *Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS '10)*, pp. 555–560, Boston, Mass, USA, 2010.

[11] S.-C. S. Cheung and C. Kamath, "Robust background subtraction with foreground validation for urban traffic video," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 14, pp. 2330–2340, 2005.

[12] M. Teutsch and W. Kruger, "Detection, segmentation, and tracking of moving objects in UAV videos," in *Proceedings of the 9th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS '12)*, pp. 313–318, Beijing, China, September 2012.

[13] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151–1163, 2002.

[14] T. Ko, S. Soatto, and D. Estrin, "Warping background subtraction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1331–1338, San Francisco, Calif, USA, June 2010.

[15] R. Cucchiara, A. Prati, and R. Vezzani, "Advanced video surveillance with pan tilt zoom camera," in *Proceedings of the Workshop on Visual Surveillance (VS) at the 9th European Conference on Computer Vision (ECCV '06)*, Graz , Austria, May 2006.

[16] K. S. Bhat, M. Saptharishi, and P. K. Khosla, "Motion detection and segmentation using image mosaics," in *IEEE International Conference on Multimedia and Expo (ICME '00)*, pp. 1577–1580, 2000.

[17] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*, pp. 647–679, April 1981.

[18] A. C. Shastry and R. A. Schowengerdt, "Airborne video registration and traffic-flow parameter estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 4, pp. 391–405, 2005.

[19] H. Yalcin, M. Hebert, R. Collins, and M. Black, "A flow-based approach to vehicle detection and background mosaicking in airborne video," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, p. 1202, 2005.

[20] Y. Wang, Z. Zhang, and Y. Wang, "Moving object detection in aerial video," in *Proceedings of the 11th International Conference on Machine Learning and Applications*, pp. 446–450, 2012.

[21] L.-W. Tsai, J.-W. Hsieh, and K.-C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 850–864, 2007.

[22] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[23] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[24] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1–3, pp. 185–203, 1981.

[25] J. K. Kearney, W. B. Thompson, and D. L. Boley, "Optical flow estimation: an error analysis of gradient-based methods with local optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 2, pp. 229–244, 1987.