Automation in Construction 72 (2016) 294–308

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Detecting and tracking vehicles in traffic by unmanned aerial vehicles

Liang Wang, Fangliang Chen, Huiming Yin *

Department of Civil Engineering and Engineering Mechanics, Columbia University, 610 Seeley W. Mudd 500 West 120th Street, New York, NY 10027, USA

ARTICLE INFO

Article history:
Received 17 September 2015
Received in revised form 31 March 2016
Accepted 5 May 2016
Available online 25 May 2016

Keywords:
Unmanned aerial vehicle (UAV)
Vehicle detection
Vehicle tracking
Traffic data
Optical flow
Scale invariant feature transform (SIFT)

ABSTRACT

Using unmanned aerial vehicles (UAV) as devices for traffic data collection exhibits many advantages in collecting traffic information. This paper introduces a new vehicle detecting and tracking system based on image data collected by UAV. This system uses consecutive frames to generate vehicle's dynamic information, such as positions and velocities. Four major modules have been developed: image registration, image feature extraction, vehicle shape detecting, and vehicle tracking. Some unique features have been introduced into this system to customize the vehicle and traffic flow and to jointly use them in multiple consecutive images to increase the system accuracy of detecting and tracking vehicles. Field tests demonstrate that the present system exhibits high accuracy in traffic information acquisition at different UAV altitudes with different view scopes, which can be used in future traffic monitoring and control in metropolitan areas.

1. Introduction

With the ongoing growth of our metropolitan road network, it is indispensable to have a comprehensive monitoring system for the complex transportation. However, there are many limitations based on the current monitoring systems. Firstly, the range of traditional road traffic monitoring is restricted to the sensor's distributions, such as induction loops, radar sensors and traffic cameras. According to the sparse distribution of the current traffic monitory system, there are many blind regions on a city road network. In certain cases such as emergency mitigation, it is required to temporarily supervise the detailed traffic situations at the "hotspots", such as the regions of traffic incidents, sources and/or destinations of traffic flow, and the emergency locations with damage of ground infrastructure [1], etc. Secondly, most traffic sensors are designed to collect traffic information on a fixed road section or with a limited road length. As a result, it is convenient to obtain traffic data at lane's level, including each lane's average speed, density and flow, but it is hard to obtain traffic data at vehicle's level based on these discretely distributed sensors, such as vehicle's trajectory data.

The vehicle's level data is the fundamental data for both intelligent transportation systems (ITS) and transportation management [2]. Therefore, a monitor method designed for traffic data at vehicle's level is of significance in transportation engineering. On the other hand, in the research of driving behaviors, a detailed and accurate vehicle trajectory data is also necessary. Driving behavior models capture drivers' tactical maneuvering decisions in different traffic conditions, which are essential components in microscopic traffic simulation systems. Due to the limited availability of detailed trajectory data, most models have not been validated rigorously [3]. Data availability has posed a significant obstacle to the advancement of driving behavior modeling. Therefore, a system for detecting and tracking vehicles based on UAVs can on one hand compensate the disadvantage in the existing transportation monitoring system, while on the other hand it can also fulfill the data requirements in the research of driving behaviors modeling.

As a useful and powerful aerial robot, UAVs have been playing important roles on data and image acquisition. For example, they have been widely used in the research of agriculture, geology, hydrology, cinematography, etc. [4–9]. Compared with traditional transportation sensors located on the ground or low angle cameras, UAVs exhibit many advantages, such as low cost, easy to deploy, high mobility, large view scope, uniform scale, etc. UAVs can record a load with different lengths by adjusting flying altitude to fulfill different research requirements. Compared with low angle cameras, the video recorded by UAVs has less influence on the block of vehicles in a lane, and could measure a vehicle's position more accurately from the top view [10]. However, UAVs are rarely applied in the transportation monitoring. One of the main reasons is the lacking of an effective and robust method to detect and track vehicles in UAV's image data.

Usually, the traffic data captured by UAV contains much complex information than those by traditional monitoring system. UAV's videos include not only the traditional data such as the traffic flow average speed, density and flow, but also each vehicle's level data, such as vehicle's trajectory data, lane change data and car following data on the road. In addition, the data from a frame of a UAV's video contains multiple vehicles and the frame frequency of the UAV's video is very high, thus the data size from the UAV's video is very large. Moreover, data from the UAV's

* Corresponding author. Tel.: +1 212 851 1648; fax: +1 212 854 6267.
E-mail addresses: lw2451@columbia.edu (L. Wang), fangliang.chen@columbia.edu (F. Chen), yin@civil.columbia.edu (H. Yin).

http://dx.doi.org/10.1016/j.autcon.2016.05.008
0926-5805/

video contain the fundamental information of transportation research and management, and play an important role in several other fields of transportation science and engineering, including safety studies and capacity analysis [3,11,12]. Considering such features, the data collection, reduction and analysis can be considered as an important component in the big data analysis in transportation. It can also be extended to other civil engineering applications.

Compared with traditional traffic surveillance systems, detecting and tracking vehicles through the images captured by a UAV has specific challenges. First of all, the camera of a UAV surveillance platform changes frequently because the camera in a UAV may rotate, shift and roll during video recording. In addition, sudden shakes might also happen due to wind fluctuations, which can cause negative effects in the vehicle tracking. On the other hand, in driver behavior research models, such as car following and lane change models, each car's accurate trajectory data is needed. Missing car data and tracking error could affect the accuracy of the model parameters settings. Therefore, a high resolution of images is crucial for accurately calculating vehicle speed and lateral position of vehicles in the process of vehicle detecting and tracking.

Some approaches to vehicle detecting and tracking based on a UAV system have been proposed in the literature [13]. Based on the recognition methods applied in the researches, a vehicle recognition method can be categorized into optical flow and feature extraction-matching methods. The optical flow method has many advantages in tracking and detecting moving objects in consecutive frames, such as autonomous robot navigation and surveillance of large facilities. Optical flow can capture the moving objects in a video, but the movement is the sum of the motion of both the camera and the vehicles. It is essential to identify and separate the camera's motion from the vehicle's motion. Rodríguez-Canosa, et al. [8] developed a real-time method to detect and track moving objects from UAVs. This method introduces an artificial optical flow by estimating the camera's motion, compares it with the real optical flow directly calculated from the video, and then calculate the motions of objects. Xu et al. [14] introduced a vehicle detecting and tracking method for low angle camera video. The cellular neural network was used to subtract the background and to refine the detection results with optical flow. Frarnebäck and Nordberg [15] constructed a polynomial expansion to approximate the movement between two consecutive frames at the pixel level. Based on the optical flow field constructed, the vehicle's motion could be calculated.

The feature extraction-matching method has been widely used in photogrammetry and computer vision. The working process consists of extracting features of interest from two or more images of the same object and matching these features in adjacent images [16]. In the photo-based research, generally these methods search the vehicle-like features in one or two photos, and then refine the detection results through classification and representation based on the predefined database. The vehicle-like features include a vehicle's edges, shapes, feature points, colors, gradients, etc. Zhao and Nevatia [17] figured out the important car features from a captured photo based on human experience in psychological tests, and then used the boundary of the car, front windshield and the shadow as car features in car recognition. Kaâniche et al. [18] presented a vision system for traffic surveillance with a fixed-wing UAV. The method analyzed the corner and edge information in a frame, and the Dempster–Shafer theory was used in the process of verification to increase the accuracy of vehicle detection. Kim and Malik [19] introduced a new vehicle detection method, which combines photos of multiple cameras and generates a 3D-model of vehicles. This vehicle detection and description algorithm was based on a probabilistic line feature grouping, and it could increase the computing speed and reliability. Gleason et al. [20], introduced a multiple features extraction and classification method for vehicle detection. The vehicle features, such as histogram of oriented gradients, edge orientation, and color were applied to increase the detecting accuracy. Leifloff [21] presented an approach for vehicle detection from optical satellite images, where an improved Haar-like feature was used in the method. Vehicle queues were detected using a line features extraction technique in the analysis. Tuermeret al. [22] applied the features of histogram of oriented gradients, Haar-like features and local binary patterns in the vehicle detection, where a sophisticated blob detector was used for vehicle detection. Leitloff et al. [23] developed a vehicle detection method that relies on an extended set of Haar-like feature operators. The support vector machine was used in the process of classification and refine vehicle detection results.

In general, because of the limited information available in one photo, the photo-based vehicle recognition method could only obtain a vehicle's static information, like position or gap between vehicles, but it is impossible to get the dynamic information, such as speed and acceleration of vehicles. However, the video based feature extraction-matching method focuses on the relationship and connection among the matched feature points, rather than their characteristics. Therefore, the video based feature extraction-matching method exhibits many advantages for vehicle tracking compared with the photo-based vehicle recognition method. Cao et al. [24] introduced a new framework of multi-motion layer analysis to detect and track moving vehicles in a UAV's video. The Kanade–Lucas–Tomasi (KLT) feature was selected in the vehicle motion and background motion layers. The new method is more effective and robust in the application. Cao et al. [25] applied a histogram orientation gradient (HOG) feature on vehicle detection. All HOG features are combined to establish the final feature vector to train a linear SVM classifier for vehicle classification. Lingua et al. [16] analyzed the advantages of a scale invariant feature transform (SIFT) operator for the feature extraction-matching method in the UAV systems, and developed an auto-adaptive version of the SIFT operator used in the UAV's photogrammetry field. Many researchers have also used the feature extraction-matching method for vehicle tracking. Some studies used the vehicle image as a feature [23,26] or the matched feature points [27–29] to track a vehicle's motion on the road.

Overall, detecting and tracking vehicles in traffic by UAV videos and photos has been attracting increasing attentions among transportation research community. However, many problems have not been solved yet. Firstly, the accuracy of vehicle recognition is low. Normally, the detecting accuracy of the existing technology is lower than 90% [20,22,23,30], and the driver's detail trajectory data cannot be obtained [31,32]. Moreover, traffic information, such as road, traffic flow and driver behavior features, have not been included in these methods [13,27,33]. Overall, a well-developed vehicle detecting and tracking method for actual transportation application has not been developed yet.

This paper introduces a new method attempting to address these problems. It combines many features in different optical methods into an integrated system, which consists of four modules as follows: image registration, image feature extraction, vehicle shape detecting, and vehicle tracking. Complementary advantages of different optical methods considerably improve the accuracy by this method. In addition, vehicle and traffic flow features and the corresponding specifications have been implemented in the system. In what follows, Section 2 will introduce the experiments with the UAV; Section 3 will present the methodology of vehicle detection and tracking in traffic. Section 4 will demonstrate the numerical results based on the experimental data for the vehicle detection and tracking. Finally, some conclusions of this work are provided in Section 5.

2. Experiments

A UAV traffic monitoring system has been set up to study traffic information, which consists of a quadrocopter, a camera mount, an image transfer system, and a camera as shown in Fig. 1a, while the camera mount and the camera are enlarged in Fig. 1b.

The quadrocopter used in the experiments of this paper is the DJI Phantom 2. It includes motors, battery, electronic parts, and the connection port for the camera mount. The core control part of the UAV is the flight control unit, which is a lightweight multi-rotor control platform

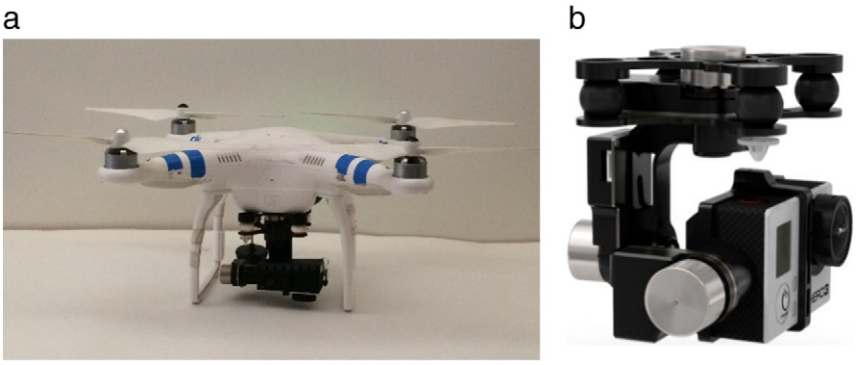


Fig. 1. The UAV traffic monitoring system used in this experiment: (a) the whole setup and (b) the camera mount and the camera.

specially designed for this lightweight UAV. The flight control unit combines the main controller (MC), a gyro-accelerometer, and a barometric altimeter. The basic information of the machine is listed in the Table 1.

The second device is the camera mount. A 3-axis H3-3D gimbal (Fig. 1b) powdered by a built-in inertial measurement unit (IMU) and special servo module was applied, which can compensate movement of the quadrocopter and provide stable quality visual video during the data collection. The camera used in the experiment is a GoPro 3 in silver version. The camera parameter settings in the experiments are listed in Table 2.

The third part of the UAV used in experiment is the image transfer system. It transmits real time video and UAVs' flight data back to the controller. The information displayed on the controller's screen is shown in Fig. 2a. In addition, the more important real-time flight data is superimposed on the video. The information includes power voltage, channel, distance between aircraft and home points, height, control mode, fail-safe mode, pitch attitude, roll attitude, flight velocity, GPS satellite, video input, vertical velocity, attitude line and aircraft nose direction. During the test, the VAV's position has significant influence on the results, therefore the height and aircraft nose direction are key to the experiment. In order to control the UAV more precisely, reference lines are added on the screen. The video monitor is attached with the quadrocopter's remote controller, and the working image is shown in Fig. 2b.

Considering the capability restrictions of image transfer system, the video was stored in the camera's SD card, and only the low-resolution images were obtained in the monitor during the experiment. Because of this, the vehicle tracking and detection method cannot be operated in real-time with current experiment devices. However, with a more powerful image transfer system, the video might be directly transferred into a computer. Therefore, the vehicles' real-time detecting and tracking could be possible using the method introduced in this paper.

3. Vehicle detecting and tracking methodology

In this section, the whole procedures of the method for vehicle detection and tracking are presented. The workflow chart of the system is illustrated in Fig. 3, which consists of four major modules: image registration, image feature extraction, vehicle shape detecting and vehicle tracking. Each part will be elaborated subsequently.

Two factors of this method are highlighted. First, three image features are simultaneously used, including edge, optical flow and the local feature point which are elaborated in Section 3.2. These features work together to increase the accuracy of vehicle recognition. On the other hand, vehicle detecting and vehicle tracking are analyzed as two independent modules in this method. It is convenient to make specific adjustments based on their different objectives. For instance, vehicle detection aims to accurately identify each vehicle's shape in a single frame, while vehicle tracking focuses on each vehicle's motion between consecutive frames. Therefore, each module of the present method is based on the combination of computer vision and transportation. The properties of traffic flow and driver behavior have also been taken into account in this method.

3.1. Image registration

Because of the instability of the UAV flying in the air, the vehicle motion recorded in the video is actually the sum of the real motion of the vehicle and that of the camera. In order to obtain each vehicle's trajectory data, the camera motion should be separated and eliminated from the vehicle motion in the video. This section describes a sequential and automatic framework aiming at the precise registration of a road area. In the computer vision science, these processes of transforming different sets of frames into one coordinate system are called as image registration, which includes transformation models and matching point selection [34,35]. In Section 3.1.1 three types of transformation

Table 1
Parameters of the quadrocopter in the experiments.

Parameter	Range
Operating temperature	−10°C~50°C
Power consumption	3.12 W
Take-off weight	<1000 g
Hovering accuracy (GPS Mode)	Vertical: ±0.8 m, horizontal: ±2.5 m
Max yaw angular velocity	200°/s
Max ascent/descent speed	Ascent: 6 m/s, descent: 2 m/s
Max flight velocity	10 m/s
Diagonal distance	350 mm
Weight with battery	800 g
Max flight time without load	25 min

Table 2
Camera parameters in the experiments.

Parameter	Range
Resolution	1920 × 1080
Frequency	30 fps
Optical zoom	1×
Horizontal degree of field of view	94.4
Vertical degree of field of view	55.0
Diagonal degree of field of view	107.1
Dimensions	177 mm × 100 mm × 100 mm
Weight	590 g

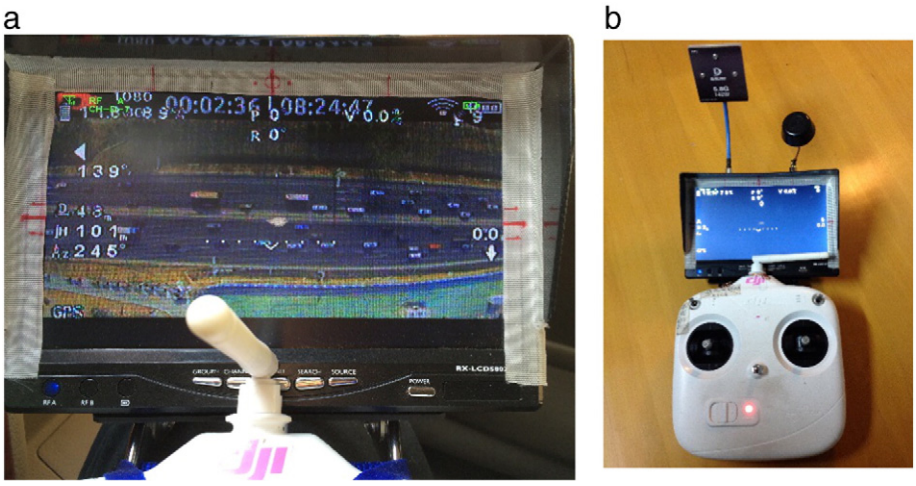


Fig. 2. Image capture and transfer system: (a) image taken during the experiment and (b) the remote controller of the quadrocopter and video monitor.

and corresponding applications are introduced. The methods of matching point selection in road environment are discussed in Section 3.1.2.

3.1.1. Transformation models

Normally, the general transformation between two images could be expressed in the following equation [36]:

$$X_{a,p} = TX_{b,p} \tag{1}$$

where $X_{a,p}$ and $X_{b,p}$ are the image coordinates of the point p in the frame a and frame b; T is the transformation matrix, which includes the information of rotation matrix from a to b, translation vector from a to b, and camera's intrinsic parameter matrices, etc.

Because an optical imaging system strongly relies on the weather and illumination conditions during the experiment [33], there is no uniform transformation model could be applied for all UAV videos and all traffic situations, thus the matrix T in Eq. (1) has different expressions correspondingly. Theoretically, a complex T is more compatible for complex situations, but has higher probability to fail, while a simple T could only be used in a specific situation, but is relatively stable in the image registration [33].

For an arbitrary object, the relationship of a fixed point in an image and a reference image is following a fundamental matrix [34,37]. However, if a road surface in a UAV's video is planar, and the scene height differences are relatively small compared to the UAV's flight height, the image transformation process could be assumed as a homography transformation problem [33]. The homography transformation matrix can be expressed as Eq. (2) [33,38].

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

where h_1 to h_8 are the parameters needed to be determined. At least four corresponding points are required to determine the homography matrix, while three of them must be non-collinear [38].

Because of the camera mount in a UAV system, some camera movements along certain specific directions could be well controlled by

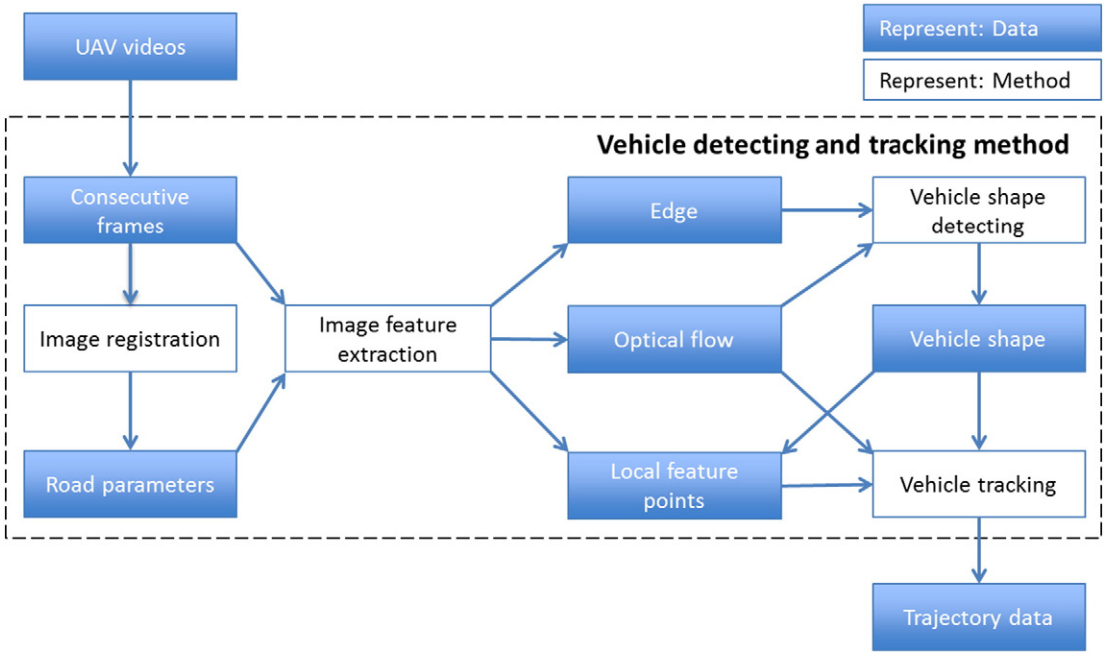


Fig. 3. Workflow chart for vehicle detection and tracking in a UAV system.

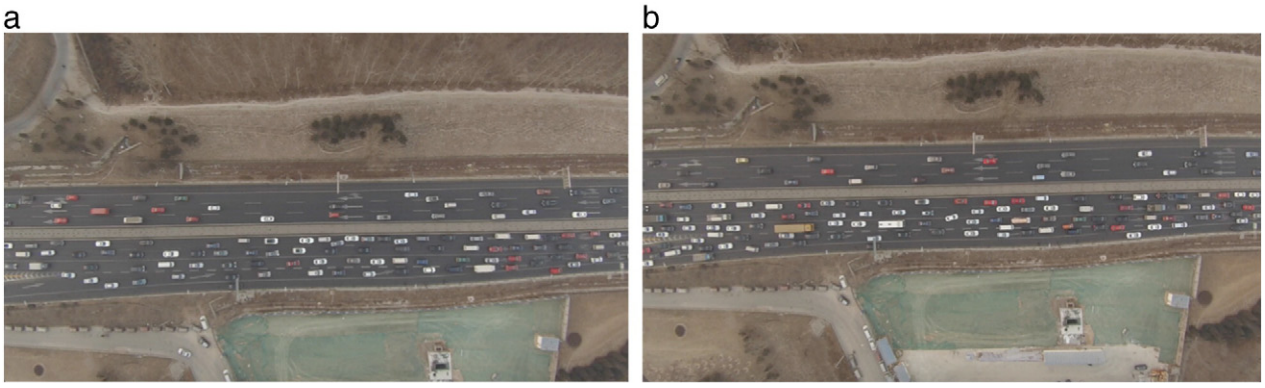


Fig. 4. Two frames in a UAV experiments: (a) the 120th frame, (b) the 1530th frame.

adjusting the operational parameters. As shown in Fig. 4, the camera's rotation along the road lateral direction is very small, but its main movement is on the road lateral direction. In that situation, the lines along the longitudinal direction of the road are always parallel and follow the linear transformation requirements. Although the lines in the frame are not always parallel on the lateral direction, considering the road length is much longer than the width in the video, and the vehicle's width is relatively small, we still can assume that this is an affine transformation. Accordingly, the transformation model can be expressed by Eq. (3) [36].

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & t_x \\ a_3 & a_4 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3}$$

where the 3×3 matrix is the affine transformation matrix. The parameters $a_1$to a_4 define the scale, rotation, and sheering effects of the combination of any linear transforms, and t_x and t_y are translation parameters on the x and y directions. In Eq. (3), 6 parameters are needed to be determined.

If the wind during the experiment is very small and the whole UAV system is well optimized, the camera mount could isolate the UAV's movement and accurately maintain the camera always vertically facing to the ground (Fig. 5). In that situation, the transformation model could be considered as a scale–rotation transformation, which can be mathematically expressed in Eq. (4) [36].

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s \cdot cos\gamma & s \cdot (-sin\gamma) & t_x \\ s \cdot sin\gamma & s \cdot cos\gamma & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{4}$$

where the transformation matrix has four unknown parameters. Besides t_x and t_y, s represents the scale changes, and γ is the rotation angle of the camera. Two matching points are required to determine the matrix.

For a real UAV video, the transformation model selection is based on the quality of the video. Table 3 lists the three most possible image transformation models could be used in the image registration. Considering the field testing environment and the stability of a UAV's video, the scale– rotation matrix has been implemented in our simulation in the following part of this paper.

3.1.2. Matching point selection

Normally, the altitude of a UAV is higher than 100 m, and the region of a road is only a small part of the whole view scope of a UAV's camera. As a result, the scene covers a large amount of unnecessary information, which might cause errors in the process of image registration [33]. On the other hand, repeated patterns are also a severe problem that may lead to the failure of the algorithm. As there are several similar road elements, such as landmarks and lights, the road area of a freeway severely suffers from such a problem. For these reasons, traditional image registration methods, such as SIFT-based matching [39] and Kanade–Lucas–Tomasi (KLT) based matching [40], may have difficulties in parameter determination [33].

To improve the accuracy and robustness of image registration for the UAV's video, two methods are proposed in the matching point selection. First, the road features are selected to calculate the camera motion. To reduce the error caused by the repeated pattern, the elements of a road are used as matching points, including road width, road direction angle θ and lane mark. However, the relatively higher objects on the road, such as signal boards, are not recommended as matching points, because the height of signal board is usually higher than 10 m, which do not accurately satisfy the plane assumption of homography transformation that all objects in the image should be in the same plane. For example, the two signal boards in Fig. 6b, which have been marked in green boxes, are not proper matching points.

Furthermore, the color form of the video is recommended to transform into HSV, which represents the hue, saturation and value. Compared with other color forms, HSV easily captures the road features,

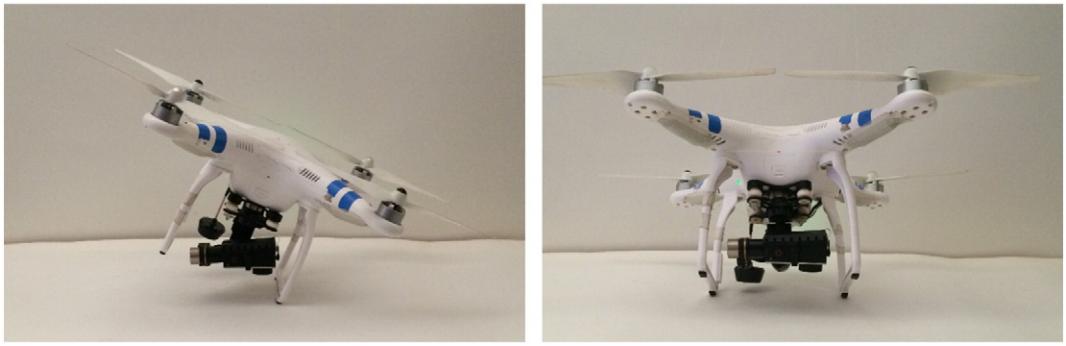


Fig. 5. The camera mount isolate of the quadrocopter.

Table 3
Transformation models in the image registration.

Transformation model	Minimum matching points	Requirements
Homography	4	The observed road in a plane or the height of the observed object is relatively small compare with UAV's altitude.
Affine	3	The camera rotation along the road lateral direction is small.
Scale–rotation	2	The camera could always vertically face to the ground.

because the road materials usually absorb light while lane marks reflect light clearly. The road area shows much lower value of V than others. As a result, by setting a suitable threshold for V, the road information can be easily captured. Fig. 6a illustrates an example of image registration; whereas Fig. 6b shows the result of setting the threshold of V as 0.35, which is set according to the illumination and the camera parameters.

3.2. Image feature extraction

After the image registration, the next step is to extract the image features such as edge, optical flow and local feature point. The image features will be used in the modules of vehicle detection and vehicle tracking in Sections 3.3 and 3.4. In this section, the concepts of three image features are briefly introduced. Considering the characteristics of traffic flow to be investigated, some specific modifications are made in the process of image feature extraction.

3.2.1. Edge

Edge detection has been carried out by a set of mathematical methods, which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, exhibits discontinuities. Such points are typically organized into a set of curved line segments termed edges [41]. In this method, the operator of the Prewitt edge detection [35] is applied. Mathematically, the Prewitt operator uses two 3 × 3 kernels that are convolved with the original image to calculate approximations of the derivatives—one for horizontal changes, and the other for vertical, which could be expressed by Eq. (5) [35].

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} * A \text{ and } G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} * A \tag{5}$$

where * denotes the two-dimensional convolution operation; A is an a grayscale digital image, which is converted from the original UAV's video; and G_x and G_y are two images which contain the horizontal and vertical derivative approximations at each point.

In this method, we only analyze the direction vertically to the longitudinal direction of the road. Therefore, the lateral direction edge used in the method can be calculated by

$$G_{lat} = G_x \sin\theta + G_y \cos\theta \tag{6}$$

where θ is the angle from the x-coordinate to the road longitudinal direction, in which the value is obtained in the process of image registration in Section 3.1.

Then based on each pixel's $G_{lat}(i,j)$, it is possible to generate the edge binary image with a proper threshold δ, where i and j are the pixel indexes at the x- and y-axes separately. The value of δ is affected by experimental environments and operational parameters of a UAV's camera. Each pixel's value in the binary image can be calculated by Eq. (7) and a frame of the resulted edge is shown in the Fig. 7.

$$F_{edge}(i,j) = \begin{cases} 1 & G_{lat}(i,j) \geq \delta \\ 0 & G_{lat}(i,j) < \delta \end{cases} \tag{7}$$

Although Fig. 7 only shows the vehicles' lateral boundary and the vehicle shape is not very clear, the vehicle shape could be easily recognized together with the optical flow result which will be introduced in the following section.

3.2.2. Optical flow

Optical flow is the pattern of any apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene [42]. For a 2D case with an extra time dimension being considered, a voxel at location (x,y,t) with intensity $I(x,y,t)$ can move by Δx, Δy and Δt between two consecutive image frames, and the brightness is similar. Assuming the movement is small, the image constraint at $I(x,y,t)$ with Taylor series can be expressed by Eq. (8) [42].

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + O\left(\Delta x^2, \Delta y^2, \Delta t^2\right) \tag{8}$$

Eq. (8) can be rewritten into

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0 \tag{9}$$

where V_x, V_y are the x and y components of the velocity.

Eq. (9) is known as the optical flow equation [40]. Because there are two unknown variables (V_x and V_y), an additional equation is required, given by some other constraints. Different optical flow methods introduce different extra conditions for that.

The Lucas–Kanade operator [40] is applied in this method. It assumes that the flow is essentially constant in a local neighborhood of

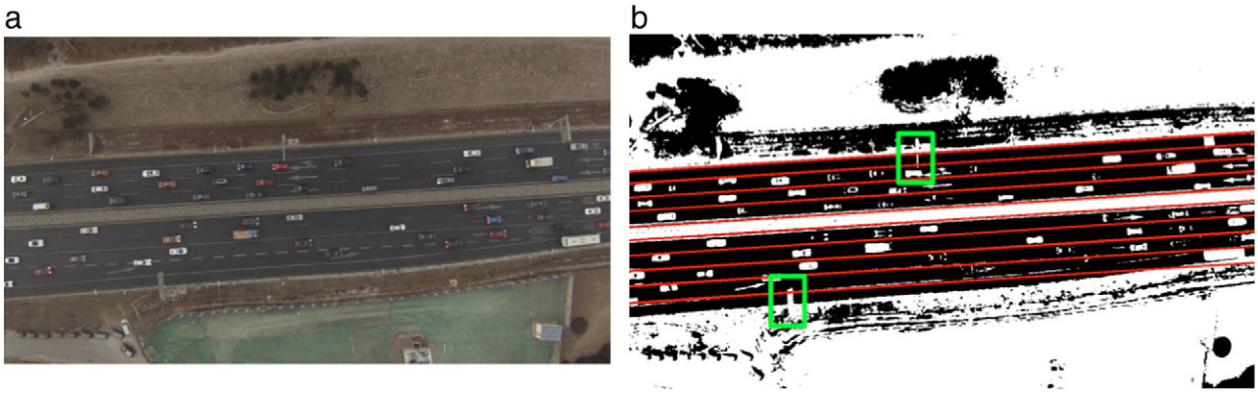


Fig. 6. Traffic flow in UVA's frame: (a) the original UVA's frame and (b) the BW figure by setting the threshold of V = 0.35.

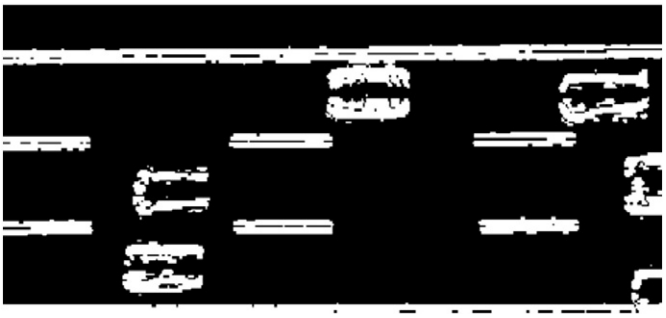


Fig. 7. Edge feature in the lateral direction of the road.

the pixel under consideration, and solves the basic optical flow equations for all the pixels in that neighborhood by using the least squares criterion. The optical flow equation for pixel p can be assumed to hold for all pixels within a region Ω centered at p. Namely, the local image flow vector (V_x, V_y) can be obtained by calculating the minimum value of the following weighted item [40].

$$\sum_{x,y\in\Omega} W^2(x)\left(I_x V_x + I_y V_y + I_t\right)^2 \tag{10}$$

where W is an $n\times n$ diagonal matrix containing the weights $W_{ii}=\omega_i$ to be assigned to the equation of pixel q_i. Usually there is more weight to the pixels that are closer to the central pixel p. The Lucas–Kanade method obtains a compromise solution by the least square criterion [40]:

$$A^T W^2 A V = A^T W^2 b \tag{11}$$

where $A=(\Delta I(q_1),\ldots,\Delta I(q_n))^T$, $W=diag(W(q_1),\ldots,W(q_n))$, $b=-\left(\frac{\partial I(q_1)}{\partial t},\ldots,\frac{\partial I(q_n)}{\partial t}\right)^T$ and V is obtained as Eq. (12) [40]:

$$V=\left(A^T W^2 A\right)^{-1} A^T W^2 b$$

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i I_x(q_i)^2 & \sum_i I_x(q_i) I_y(q_i) \\ \sum_i I_y(q_i) I_x(q_i) & \sum_i I_y(q_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(q_i) I_t(q_i) \\ -\sum_i I_y(q_i) I_t(q_i) \end{bmatrix} \tag{12}$$

Each vehicle's speed includes two directions: longitudinal and lateral. Normally, the latter can be assumed to be zero unless the driver wants to make a lane change. In that case, the lateral speed should be still very small compared with the longitudinal speed. In a UAV's video, most movements in the lateral direction belong to the camera motion. To minimize such errors, only the vehicle's velocity along the road longitudinal direction is considered, which can be expressed by

$$V_{lon} = V_x \cos\theta + V_y \sin\theta \tag{13}$$

Although it is possible to obtain each pixel's velocity in Eq. (13), because of the instability of the optical flow at pixel level, it is hard to directly obtain each vehicle's speed accurately from the result of optical flow [43]. Therefore, in this method, the optical flow is used to detect the shape of each vehicle, and the value of V_{lon} will be transferred into a binary value according to a proper threshold in Eq. (14). Fig. 8 is the final binary figure for one frame.

$$F_{OF}(i,j) = \begin{cases} 1 & V_{lon}(i,j) \geq \tau \text{ and } V_{lon} \leq \sigma \\ 0 & otherwise \end{cases} \tag{14}$$

where τ and σ are the maximum and minimum rational speeds, which relate to the UAV's altitude and the stability after the image registration. Higher altitude leads to smaller τ and σ, while the lower stability requires bigger σ to eliminate the camera motion.

3.2.3. Local feature point

The scale-invariant feature transform (SIFT) is an algorithm in computer vision to detect and describe local features in images. This algorithm was published by David Lowe in 1999 [44]. The SIFT has been widely used in many applications, such as object recognition, image stitching, 3D modeling, gesture recognition, video tracking, individual identification of wildlife and match moving [16], etc. The SIFT is selected as the feature detector and matching method in this method due to its advantages over other methods. The local feature points are invariant to image scaling, translation, and rotation, and partially invariant to illumination changes and affine or 3D projection [44]. The stability of local feature could compensate the negative effects caused by the movement and vibration of the UAV's camera in the air. Furthermore, because each vehicle in a video will change its position as proceeding forward, from the perspective of the camera, the angle of a vehicle will also accordingly change. This scenerio is illustrated in Fig. 9, in which the image of a vehicle has a transparent change as it proceeds. The vehicle tracking accuracy is challenged by the obvious angle change of a vehicle in the video. However, the SIFT local feature points are immune to image rotation and remain invariant in this situation. Also, the UAV's camera setting might slightly change during video recording, such as brightness and white balance. As SIFT doesn't rely on the absolute values of brightness and colors, it can handle these differences in the process of matching.

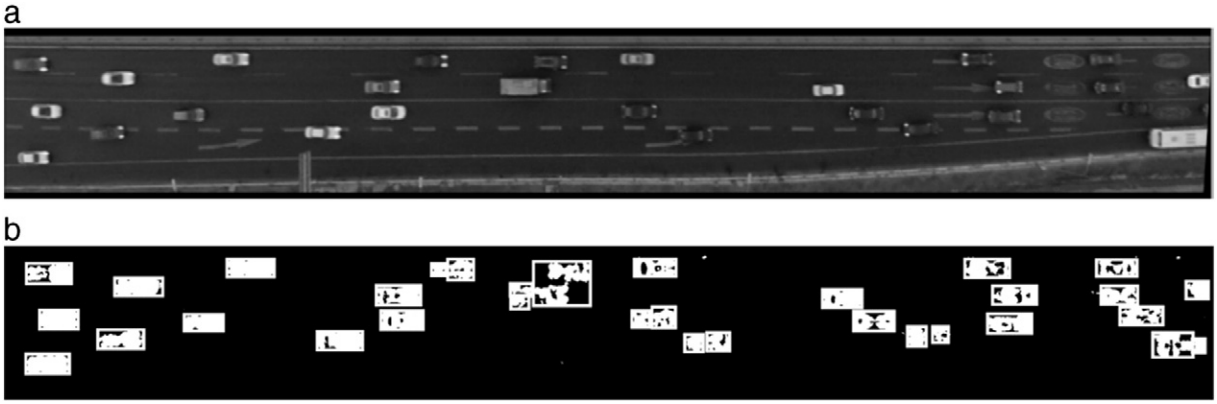


Fig. 8. Optical flow in the longitudinal direction of the road. (a) The original grayscale frame and (b) the optical flow result.

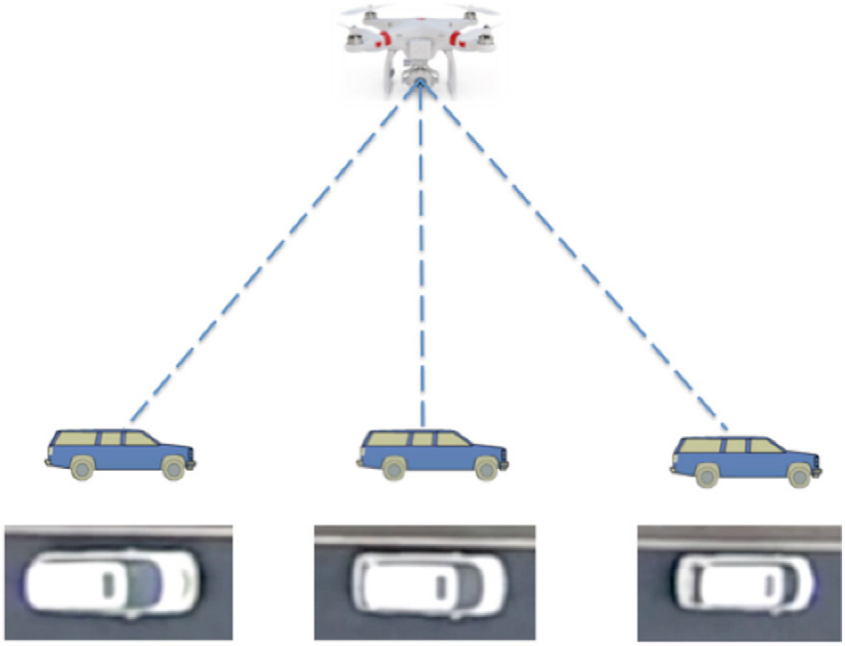


Fig. 9. Different relevant angles between vehicle and UAV.

In conclusion, three image features are used in the method, which includes edge, optical flow and local feature point. The input, required orientation, calculation range and applications of each feature are summarized in Table 4.

3.3. Vehicle shape detection

In this module, the method detects the boundary of a vehicle appeared in the vehicle shape detection region. To solve the potential problems in vehicle shape detection, both features of optical flow and edge are used.

3.3.1. Vehicle shape detection region

Considering the vehicle is a rigid body, each vehicle's shape will not change after determined in the following processors. As a result, vehicle shape detection can be conducted in a region of interest, which is called as the vehicle shape detection region in this method.

There are two requirements for the vehicle shape detection. The first is that the length of the region should be longer than that of the longest vehicle during the experiment; whereas the second is that the vehicle shape detection region should be an area where vehicles stay in the center of the lane in the shape detection region if no lane change is considered. In this method, the features of image edge are used to refine the result of vehicle shape. A region of a vehicle edge with less lane mark interrupted could increase the accuracy of vehicle shape detecting. As shown in Fig. 10, the left side part of red line is the vehicle shape detection region in this test, we choose the left (upstream) 300 pixels width as shape detection region.

3.3.2. Vehicle shape detection with optical flow and edge

From the image feature result of optical flow obtained in Section 3.2.2, it is possible to identify the rough vehicle shape information. However, the result of the optical flow has some potential errors in the process of vehicle shape recognition.

Firstly, the similar brightness inside a vehicle implies that the matrix A in Eq. (11) is singular, where A is defined as the difference of intensity around the subject point. When pixels intensity is similar to the insider of a vehicle's region, matrix A will be close to zero and be singular. The velocity at this point, $V = (A^T W^2 A)^{-1} A^T W^2 b$, is considered as zero [43]. Consequently, when the middle area of a long vehicle has few recognized features from the top view, the optical flow method could recognize the long vehicle as two separated parts. Thus it is possible that one vehicle will be recognized as two short vehicles. Another error is when two vehicles are close to each other, the optical result cannot accurately identify the boundaries of the two close vehicles. For example, two cars move on two adjacent lanes with a similar speed, and the distance between the two vehicles is small. Due to the vehicle shadow and the camera angle, the gap between two vehicles is invisible in the result of optical flow. This method will mistakenly recognize two cars as one. These two optical flow potential errors are shown in top in Fig. 11b.

In order to quantify the two potential errors in optical flow, the edge feature (Fig. 11c) is used as a supplement to the optical flow method. The pseudo code of the framework is listed in Table 5. The code from line 4 to 6 can solve the first potential error, and that from line 7 to 8 can solve the second potential error. The result of vehicle shape detection is shown in Fig. 11d. After one vehicle shape is confirmed, the system will move to the next step of vehicle tracking.

3.4. Vehicle tracking

To improve the accuracy and decrease the error rate, both features of local feature point and optical flow are used in the vehicle tracking module. Occlusion is also considered in this method. The workflow of the vehicle tracking algorithm is illustrated in Fig. 12. The local feature points

Table 4
The summary of three types of image features.

Features	Input	Orientation	Range	Application
Edge	One frame	Vertical to the road longitudinal direction	On the vehicle shape detection region	Vehicle shape detection
Optical flow	Consecutive frames	Parallel to the road longitudinal direction	All the road region	Vehicle shape detection and vehicle tracking
Local feature point	Consecutive frames	No direction	Inside detected shape of each vehicle	Vehicle tracking

Fig. 10. The result of vehicle shape detection.

are matched in two consecutive frames in each vehicle shape, and the matched points are selected in vehicle tracking, which will be introduced in Section 3.4.1. There are some traffic signs or bridge that may block a vehicle's shape. To avoid these occlusion's negative effects, this method will check each matched local feature point, and identify the occlusion point. Then the recognized occlusion local feature point will be stored into a database, which will be helpful to recognize the occlusion point in the following steps. The occlusion case will be discussed in Section 3.4.2. Because the matched local feature points are not reliable in the process of vehicle tracking, as a backup method, optical flow could also help to track vehicle. The relevant introductions are presented in Section 3.4.3. In Section 3.4.4, the validation of each vehicle's tracking result will be demonstrated.

3.4.1. Update speed with matched local feature points

In the step of vehicle tracking, we use the matched local feature points to detect the movement of the vehicle in consecutive frames. As shown in Fig. 13a, the blue local feature points belong to the frame at time n, and the red local feature points belong to the frame at time $n+\Delta t$, where Δt is the time between two consecutive frames. The matched local feature pair is considered as the same point on the vehicle. As a result, the distance between the matched local feature pairs is considered as the distance of vehicle movement during Δt. In Fig. 13b, two consecutive frames are shown in cyan–red figure.

3.4.2. Occlusion with traffic signs

In some situations, part of the vehicle will be occluded by the traffic signs in the video. Then the matched local feature points on the vehicle will be clustered into two groups. One group remains moving in the original speed and the other group is static, because these matched local feature points are from the traffic signs. Several steps are taken to solve this problem. In the first place, these matched local feature points of the traffic signs are removed and the matched local feature points left are used to track the vehicle. If no matched local feature points are left, the vehicle's previous speed would be used to track the vehicle. Then, the matched local feature points of the occlusion will be stored in the occlusion point database. If the same occlusion point appears in the following frames, the computer will easily recognize it and eliminate the negative influences of the occlusion point directly.

3.4.3. Track vehicle through optical flow

As is known, the SIFT method is a powerful tool in the research of feature point. However, in the process of finding the matched local feature points, there exist many restrictions and requirements [16,45]. If the resolution of the image is not enough or the altitude of UAV is too high, the pixels for each vehicle will be fairly limited. As a consequence, it will be difficult to identify matched local feature points in the region of detected vehicle shape. Thus, it is impossible to track vehicle based on

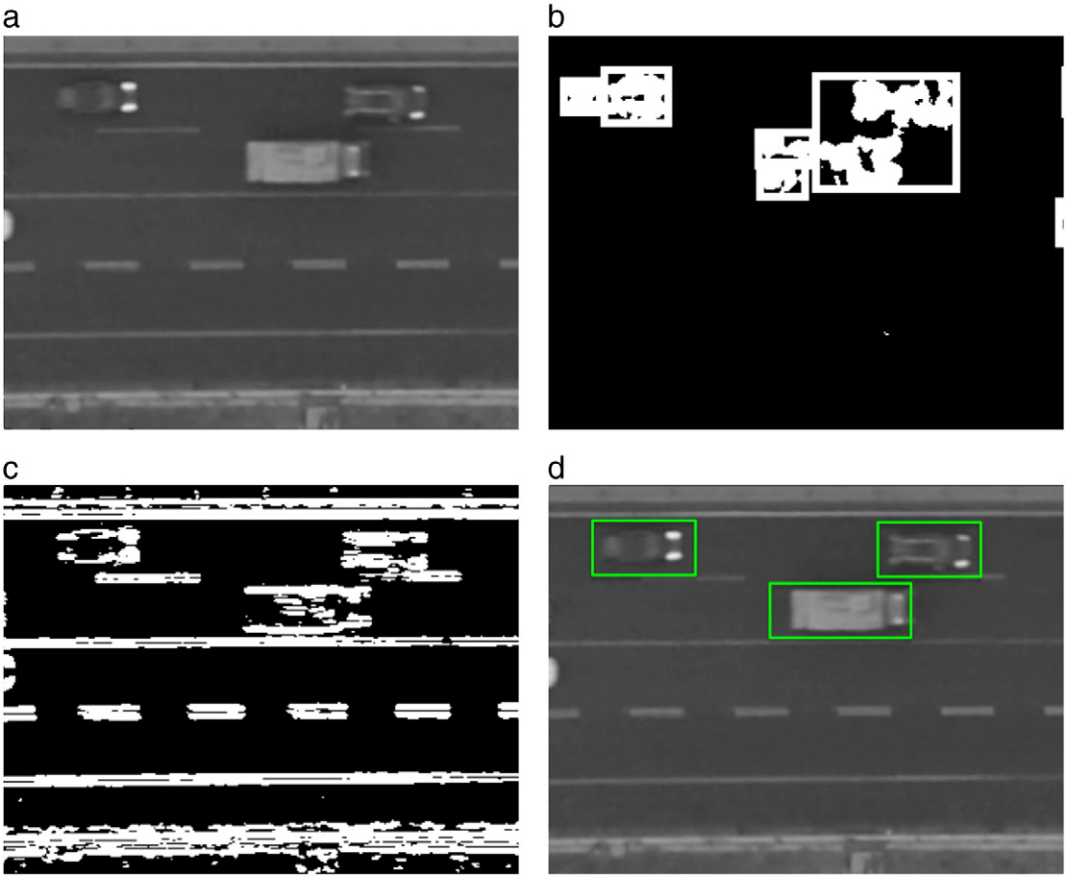


Fig. 11. (a) The original figure; (b) the optical flow result; (c) the edge feature result; and (d) the final detection result.

Table 5
Framework of the vehicle shape detection with optical flow and edge.

Line	Code
0	Input: optical flow feature in road longitudinal direction and edge feature in road lateral direction
1	For I = first frame to last image
2	Recognize vehicle shape set $\{B_j\}$ based on result of optical flow in each frame
3	For j = all vehicle shape B_j
4	For k = all vehicle shape B_k close to B_j, where $B_k \in \{B_j\}$
5	If B_j and B_k share two continuous edge lines
6	Link the areas of B_j and B_k as one vehicle.
7	If the width of B_j is bigger than the threshold λ (in the calculation $\lambda = 4m$)
8	Separate B_j according to the result of edge, and recalculate the vehicle shapes
9	Output: the confirmed vehicle shape with a unique ID

the SIFT method. To deal with such problem, the optical flow will be included in vehicle tracking.

As shown in Table 6, the vehicle optical flow results at time t + Δt are demonstrated as a black block, the blue box is the vehicle position at time t, and the dashed box is the estimated vehicle position at time t + Δt. By this way, each vehicle shape at time t is marked with 9 points (Table 6). There are 5 points on both longitudinal and lateral directions. In the longitudinal direction, if one optical flow block covers 3 of the 5 points, this block is considered as part of the car and this optical flow block could be used in tracking this vehicle. The same principle is applied in the lateral direction. If the vehicle speed is low or the video frequency is high, one optical flow block will be considered as part of the car only when it covers at least 4 of the 5 longitudinal points.

This is a relatively conservative principle to minimize the detecting error in the vehicle tracking process. However, because only the optical flow block occupied 3 of the 5 points could be used in vehicle tracking, the vehicle trajectory result is not very smooth compared with updating speed with local feature points. Considering that optical flow is only a backup method in vehicle tracking process and to avoid tracking failure, this rule is reasonable.

3.4.4. Vehicle speed verification

In order to test whether the tracking results are reasonable, the movement of each vehicle is required to satisfy certain basic criteria. If the acceleration value of a vehicle is larger than the threshold, the vehicle tracking result is considered as a failure. In this situation, previous step's speed will be used. The vehicle movement limitations used are based on the parameters of the Gipps car following model. The maximum acceleration is $3m/s^2$ and the maximum deceleration is $-3.5m/s^2$ [46].

4. Results and discussion

To demonstrate the present vehicle detecting and tracking system, field tests were conducted and the modeling results are compared with the actual observations. The location of the field tests was selected as the north part of the 5th Ring Road in Beijing, China. The testing road

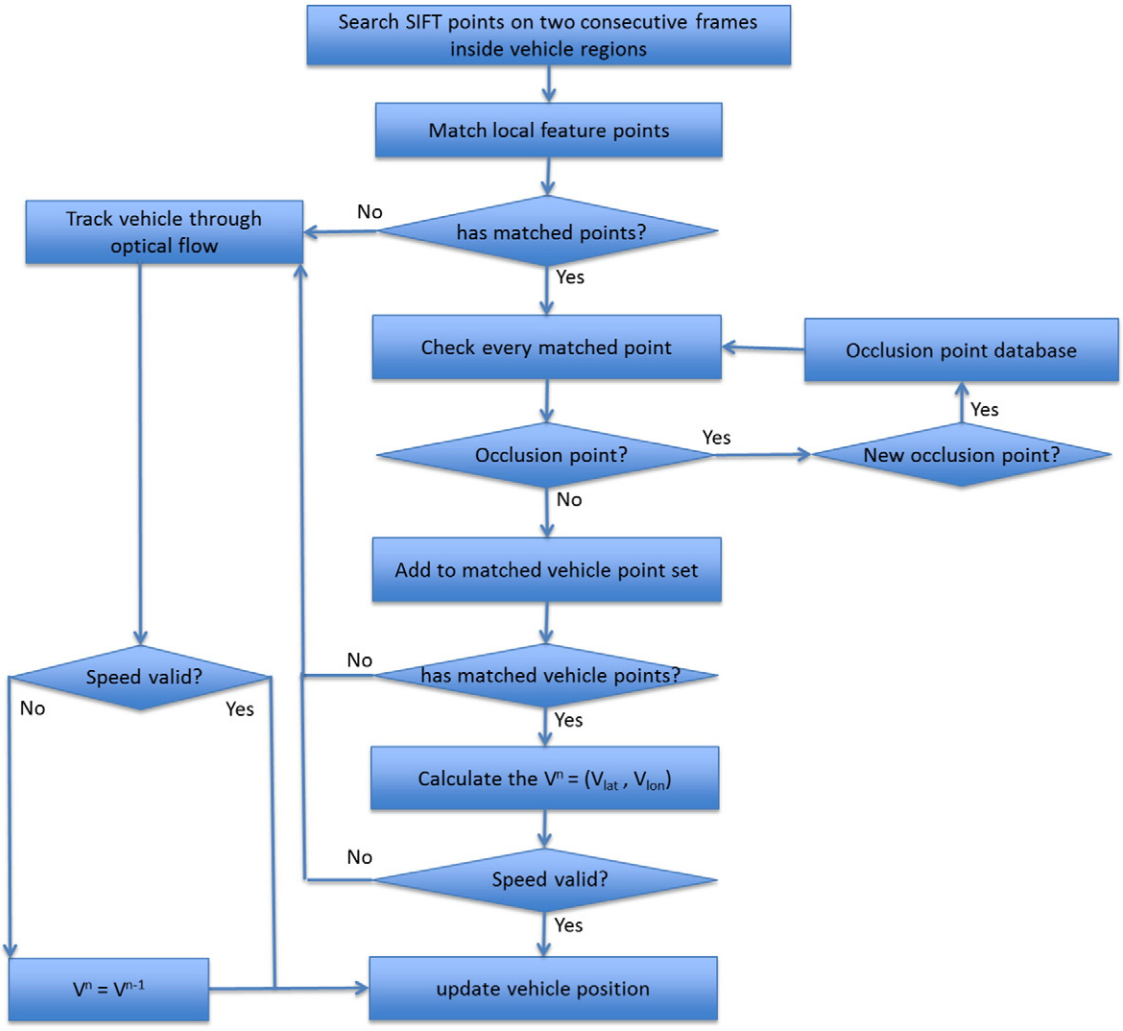


Fig. 12. Workflow of vehicle tracking.

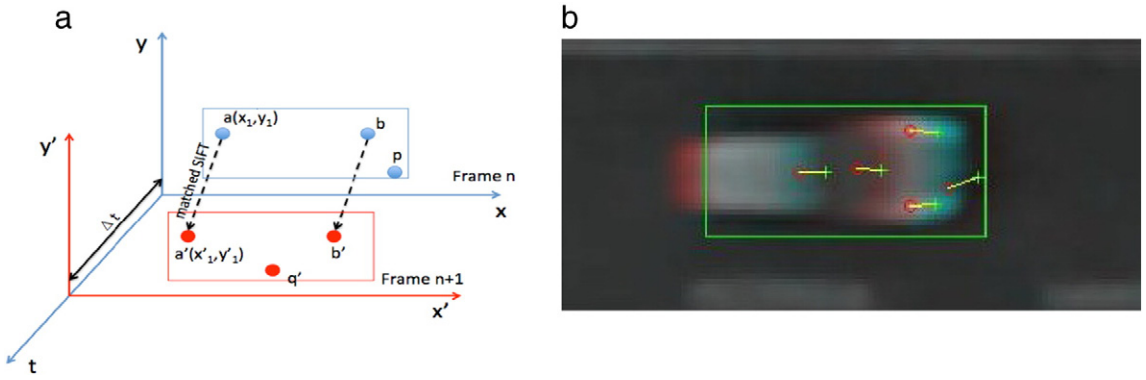


Fig. 13. (a) The demonstration of vehicle tracking based on matched local feature points. (b) One example of track vehicle with matched local feature points.

has two direction traffic flows, which is an arterial road and the traffic flow is large. The side stretching from west to east has four lanes with an additional merging lane and a shoulder lane. The other side of the road has four lanes with a shoulder lane. This road section includes complex traffic marks on the lane, such as the number of speed limit and direction sign. Near the merging lane, there is a traffic signal board. The image of the experiment road is shown in Fig. 14.

From 2014 Dec to 2015 May, the road was monitored by 66 times and the total video length is over 850 min. Limited by the battery capacity, each test was about 13 min. Except for the time of taking off, landing and adjustment, the length of each valuable video on vehicle detecting and tracking is about 10 min. Considering the conditions of traffic flow, weather and sunshine, the tests were chosen at the morning time from 6:30 a.m.–7:20 a.m.

Considering its convenience and portability, a MacBook Pro laptop was selected in the tests. The processor is 2.7 Ghz Intel Core i7, memory is 16 GB 1600 Mhz DDR3, graphics is Intel HD graphics 4000 1536 MB. The method is programmed in MATLAB. The basic image feature extraction methods used in the program are listed in the following Table 7.

The running time depends on the traffic situation of the observed road section and the frame frequency of the video. Considering the road section length is 228 m (UAV's altitude is 120 m), about 25 vehicles on each frame (only consider on direction traffic flow); and video frame frequency is 30 Hz, and the resolution is 1920 × 1080. In such conditions, a 10 min UAV video requires 142 min of the image processing, which equals to 0.47 s/frame.

4.1. Vehicle detection results

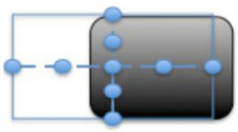
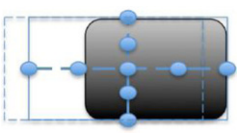
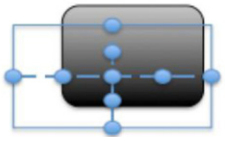
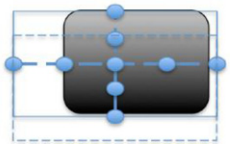
To evaluate the vehicle detection and tracking algorithms, the values for correctness, completeness and quality are calculated for different scenes from several campaigns. They are defined in Eqs. (15–17) from reference [23], with true positives being the number of vehicles detected, false positives as the number of non-vehicle detections and false negatives the number of vehicles missed.

$$\text{Correctness} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{15}$$

$$\text{Completeness} = \frac{\text{true positives}}{\text{true positives} + \text{false negetives}} \tag{16}$$

$$\text{Quality} = \frac{\text{true positives}}{\text{true positives} + \text{false negetives} + \text{false positives}} \tag{17}$$

Table 6
The rule of vehicle tracking based on optical flow.

Original situation	Adjustment result
	
	

The vehicle detection results are listed in Table 8 and are validated manually. It is worthy to note that through the camera horizontal degree of field of view is 94.4° (Table 2), however, after the correcting the image deformation, the observed road section length may reduce. From Table 8, the results of correctness are 100% for all cases with different altitudes, which means every recognized vehicle is correct. In transportation research, the correctness is more important than completeness in vehicle shape detection, because one false positive failure (consider a non-vehicle as a vehicle) will directly cause the failure of vehicle tracking in the following steps (as vehicle tracking is depending on the vehicle detecting result). Furthermore, this false positive detected "vehicle" (this is not a vehicle in the video) is like a "ghost" vehicle, and will generate unpredictable errors, which could cause difficulties in the traffic flow analysis.

There are two main reasons that may cause false negative failures. On one hand, when vehicle is crossing the lane mark in the vehicle shape detection region, the lane mark will have negative effects in vehicle detecting. Especially in vehicle lateral edge figure, the computer cannot tell the vehicle shape with lane mark edge. On the other hand, if a vehicle shape is very different from the typical vehicle, it will cause false negative failure. For this reason, a predefined vehicle width threshold is set in the vehicle shape detecting. If a vehicle is over that limitation, the vehicle will not be recognized. For example, in the 120 m-altitude video, the vehicle recognized width limitation is 30 pixels to 45 pixels.

Overall, vehicle detecting could provide acceptable accuracy when the UAV's altitude within the range of 100 m–120 m. But when the UAV's altitude increases to 150 m, the accuracy decreases to 96.1%. Because each vehicle's occupied pixels in video is reduced (in 150 m-altitude video, each vehicle occupies only about 150 pixels), correspondingly the optical flow and lateral edge are not accurate enough for vehicle detection. Theoretically, a higher resolution camera could increase the accuracy of detection results in higher altitude.

Fig. 14. The experiment location (altitude is 150 m).

4.2. Vehicle tracking results

Because this method separate the vehicle detection and tracking into two different processes, the vehicle tracking results will be demonstrated individually. However, there's no exact method or standard to evaluate the result of vehicle tracking. Indirectly, the concept of "lost vehicle" [48–50] is introduced in macroscopic traffic simulation, which is defined as a vehicle moving out of the lane, or superimposed on another one, or staying in a position for an irrationally long time. The number of "lost vehicles" can be an indicator to evaluate the vehicle tracking algorithms. The results of vehicle tracking at different altitudes are listed in Table 9. The vehicle tracking accuracy will decrease when the UAV altitude increase. But the presented method could still generally keep a higher accuracy in tracking each vehicle from 100 m to 150 m altitude.

For each vehicle tracking result, both the vehicle longitudinal and the lateral positions are recorded in an equal time interval sequence. Fig. 15 is a typical vehicle tracking results. This vehicle moves from the second lane to the third lane and takes about 20 s through the research region (200 m). It is clear that during the lane change the vehicle decreases its speed for a proper gap to enter. The average speed at the second lane is about 14 m/s, and the average speed at destination lane (third lane) is about 12 m/s. However, at gathering vehicle positions, whatever the technique is used, the observed points happen to be dispersed in the neighborhoods of the actual points and the path drawn on the observed positions inevitably fluctuates around the true path followed by the vehicle [51]. Therefore there are two types of limitations of the raw experiment results. Firstly, the vehicle's trajectory position and speed vibrate greatly, and two consecutive frames have great differences (Fig. 15). Secondly, the measurement error is more obvious in the vehicle lateral direction. Compared with a vehicle's longitudinal speed, the vehicle's lateral speed is relatively very small. As shown in Fig. 15(b), the maximum lateral speed is only 10% of the maximum longitudinal speed, which means that the same measurement error will occupy higher percentage in vehicle lateral speed. On the other hand, vehicle lateral position is hard to predict. A driver may change lateral speed and direction at any time, which means that it is hard to set a limitation to eliminate unreasonable lateral movement.

In general, the raw vehicle trajectory data could describe a vehicle's main behavior, such as speed change, lane change, etc. However the raw trajectory data still has some limitations and disadvantages for microscopic transportation research with higher accuracy requirements. A proper assessment and optimize method for raw vehicle trajectory data is thus highly demanded in the transportation analysis.

4.3. Applications and limitations

The presented method provides a convenient and reliable way to acquire each vehicle's information. The trajectory data could be used in most transportation research and traffic data analysis. This method can be directly used in actual traffic monitoring and control in dense metropolitan areas for disaster mitigation and emergency management [1]. As a useful aerial robot, UAV is playing an important role in the transportation monitoring system for the ongoing growth of city road network. The high quality trajectory data based on this method will provide important information for city transportation management, planning and construction.

Table 7
The main image feature extraction method applied in the code.

Image feature	Method	Code
Edge	Prewitt edge detection	MATLAB
Optical flow	Lucas–Kanade operator	MATLAB
Local feature point	SIFT	VLFeat [47]

Table 8
Vehicle detection results.

UAV altitude (m)/Road section length(m)	100/190	120/228	150/285
Vehicle average speed (m/s)	12.4	11.2	12.0
True positives	1999	1877	1520
False positives	0	0	0
False Negatives	2	3	61
Correctness	100%	100%	100%
Completeness	99.9%	99.8%	96.1%
Quality	99.9%	99.8%	96.1%

Table 9
Vehicle tracking results.

UAV altitude (m)/Road section length (m)	100/190	120/228	150/285
Number of vehicles	1999	1877	1520
Number of "lost vehicle"	0	11	32
Error rate	0%	0.6%	2.1%

In addition, with the aid of emerging technologies in data and image acquisition, the UAVs are attracting more attentions among researchers and engineers in new applications. The present method is not limited to the vehicle monitoring in road networks. With the improvement of the accuracy, the present method can be applied in construction-related environments, search and rescue applications, structural inspection and health monitoring. For example, by introducing a road curvature instead of the static road direction in this study, the present method might be applied at different scenarios, such as a complicated overpass or open pit mines; the vehicle detecting method can be modified for different types of tracking object; if optical flow and local feature point are used in recognizing objects, the method could be applied in tracking wild animals and people.

Nevertheless, as a novel equipment emerging in traffic monitoring system, there still exist some limitations of applying UAVs in vehicle detecting and tracking:

- The UAV's behavior is sensitive to the testing environment. Based on the field experiences, the video vibration increases as the wind velocity increases. As a consequence, it might cause the image registration failure [33]. Additionally, the UAV might not properly work in bad weather conditions, such as foggy or snowy conditions. With the advance of technology, the UAV's stability and endurance might be improved considerably in the near future, and thus the effect of weather conditions and battery capacity limitations can be overcome.
- The UAV's continuous working time is limited by its battery capacity. As a traffic-monitoring device, the short continuous working time is not able to provide enough consecutive data for a long-term monitoring of a load section. The UAV used in this test is only 15 min. To the authors' knowledge, a UAV as a traffic-monitoring device is still in research level, but has not been widely used in city road today.
- The shadow will have negative effects on vehicle tracking and detecting accuracy. Intensive research is required to effectively address this issue in the computer vision [52]. To address the shadow problem, the infrared waveband camera might be an alternative way, where each vehicle can be detected and tracked by extracting its engine's heat.
- In macroscopic transportation analysis, a longer road section, say 5 km, is required [53]. However it is hard to extend the observation scope by increasing a UAV's altitude while keeping enough pixels for recognizing each vehicle. Simultaneous applications of multiple UAVs might be one of solutions.

5. Conclusions

A UAV-based vehicle detecting and tracking system has been presented with many advantages in both traffic monitoring system and driver behavior research. The method extracts vehicle trajectory data from the UAV's video at different altitudes with different view scopes. Three significant features of the method have been demonstrated:

- Three image features, including edge, optical flow and local feature point, work jointly in this method to detect and track vehicles in a UAV's video. Compared with traditional single image feature methods, the presented method considerably improves the recognition accuracy and the system robustness.
- This method is specifically designed for vehicle detection and tracking. Based on the analysis of road, traffic flow and driver behavior characteristics, some adjustments, such as transforming image's color form into HSV to obtain road features, extracting longitudinal direction feature of Optical flow for vehicle detection, vehicle shape detection region and checking vehicle speed based on the Gipps acceleration model have been introduced in the model to further improve efficiency and accuracy.
- The vehicle detection and tracking results based on the present method showed that considerable improvement of the detecting accuracy has been reached. Three road section lengths 190 m, 228 m and 285 m, were studied, all the test results are very reliable, and the vehicle detection error is lower than 3.9% and vehicle tracking error is lower than 2.1%.

As UAV is gaining more applications in the automatic control system, more exciting studies of UAV will be explored in the near future. For example, based on the image analysis and traffic flow theory, the UAV may automatically locate the traffic accident sites and transfer real-time data to the traffic management center. In addition, vehicle detecting and tracking methods for different altitudes and different accuracy should be considered. Furthermore, based on different transportation situations and observation scopes, UAVs are desired to adjust observation altitude during the operation. With those new features, the UAV can be extended to more applications.

Acknowledgements

The authors thank Ms. Fangrui Guo, Ms. Gisele Guimaraes Ribeiro, Mr. Rodolfo Felipe Kusma and Dr. Dan Hochstein for the manuscript preparation and field experiment. In addition, this work has been sponsored by University Transportation Research Center Region II – Department of Transportation through UTRC Research Initiative project.

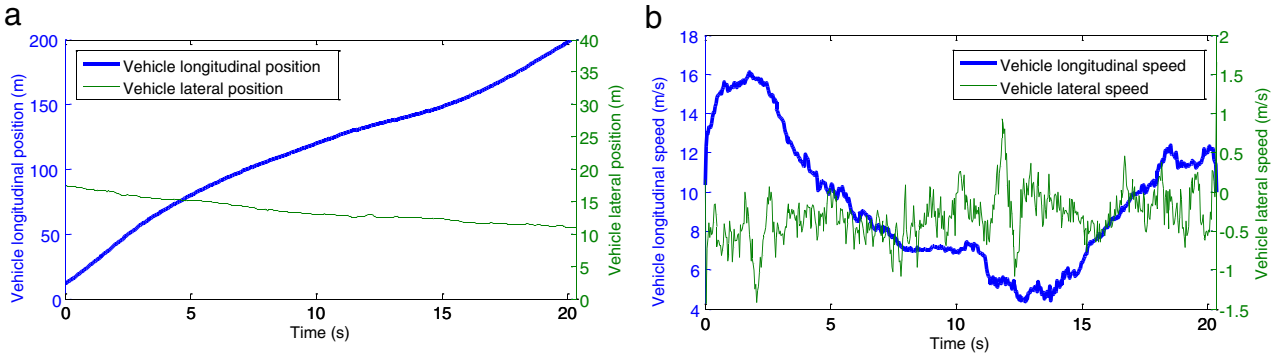


Fig. 15. A typical vehicle trajectory data; (a) vehicle position trajectory data; and (b) vehicle speed trajectory data.

References

[1] J. Leitloff, D. Rosenbaum, F. Kurz, O. Meynberg, P. Reinartz, An operational system for estimating road traffic information from aerial images, Remote Sens. 6 (11) (2014) 11315–11341, http://dx.doi.org/10.3390/rs61111315 (ISSN: 20724292).

[2] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, Y. Wang, Review of road traffic control strategies, Proc. IEEE 91 (12) (2003) 2043–2067, http://dx.doi.org/10.1109/JPROC.2003.819610 (ISSN: 0018-9219).

[3] T. Toledo, Driving behaviour: models and challenges, Transp. Rev. 27 (1) (2007) 65–84, http://dx.doi.org/10.1080/01441640600823940 (ISSN: 0144-1647).

[4] P.J. Zarco-Tejada, V. González-Dugo, J.A.J. Berni, Fluorescence, temperature and narrow-band indices acquired from a UAV platform for water stress detection using a micro-hyperspectral imager and a thermal camera, Remote Sens. Environ. 117 (2012) 322–337, http://dx.doi.org/10.1016/j.rse.2011.10.007 (ISSN: 0034-4257).

[5] A.C. Watts, V.G. Ambrosia, E.A. Hinkley, Unmanned aircraft systems in remote sensing and scientific research: classification and considerations of use, Remote Sens. 4 (6) (2012) 1671–1692, http://dx.doi.org/10.3390/rs4061671 (ISSN: 2072-4292).

[6] S.R. Herwitz, L.F. Johnson, S.E. Dunagan, R.G. Higgins, D.V. Sullivan, J. Zheng, B.M. Lobitz, J.G. Leung, B.A. Gallmeyer, M. Aoyagi, R.E. Slye, J.A. Brass, Imaging from an unmanned aerial vehicle: agricultural surveillance and decision support, Comput. Electron. Agric. 44 (1) (2004) 49–61, http://dx.doi.org/10.1016/j.compag.2004.02.006 (ISSN: 0168-1699).

[7] L. Merino, F. Caballero, J.r. Martínez-de Dios, J. Ferruz, A. Ollero, A cooperative perception system for multiple UAVs: application to automatic detection of forest fires, J. Field Rob. 23 (3–4) (2006) 165–184, http://dx.doi.org/10.1002/rob.20108 (ISSN: 1556-4967).

[8] G.R. Rodriguez-Canosa, S. Thomas, J. del Cerro, A. Barrientos, B. MacDonald, A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera, Remote Sens. 4 (4) (2012) 1090–1111, http://dx.doi.org/10.3390/rs4041090 (ISSN: 2072-4292).

[9] K. Peng, G. Cai, B.M. Chen, M. Dong, K.Y. Lum, T.H. Lee, Design and implementation of an autonomous flight control law for a UAV helicopter, Automatica 45 (10) (2009) 2333–2338, http://dx.doi.org/10.1016/j.automatica.2009.06.016 (ISSN: 0005-1098).

[10] R. Du, Z. Peng, Q. Lu, Comparison of SMS calculation methods based on NGSIM data for UAV detection, 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Vol 2, IEEE, Pp. 112–115. Aug/ 26-27/ 2012, Nanchang, Jiangxi, China, 2012, http://dx.doi.org/10.1109/IHMSC.2012.123 (ISBN: 978-1-4673-1902-7).

[11] M. Brackstone, M. McDonald, Car-following: a historical review, Transport. Res. F: Traffic Psychol. Behav. 2 (4) (1999) 181–196, http://dx.doi.org/10.1016/S1369-8478(00)00005-X (ISSN: 1369-8478).

[12] D.S. Hamdar, Driver behavior modeling, in: A. Eskandarian (Ed.), Handbook of Intelligent Vehicles, Springer, London 2012, pp. 537–558 (ISBN: 978-0-85729-084-7 978–0–85729-085-4).

[13] K. Kanistras, G. Martins, M.J. Rutherford, K.P. Valavanis, A survey of unmanned aerial vehicles (UAVs) for traffic monitoring, 2013 International Conference on Unmanned Aircraft Systems (ICUAS) 2013, pp. 221–234, http://dx.doi.org/10.1109/ICUAS.2013.6564694 (May/26–27/2013, Atlanta, GA. ISBN: 978–1–4799-0815-8).

[14] J. Xu, G. Wang, F. Sun, A novel method for detecting and tracking vehicles in traffic-image sequence, Proc. SPIE 8878, Fifth International Conference on Digital Image Processing (ICDIP 2013), 88782P (July 19, 2013), 2013, http://dx.doi.org/10.1117/12.2030637.

[15] G. Farnebäck, K. Nordberg, Motion detection in the WITAS project, Swedish Symposium on Image Analysis (SSBA): Lund, 2002 2002, pp. 99–102 (URN: urn:nbn:se:liu:diva-21607).

[16] A. Lingua, D. Marenchino, F. Nex, Performance analysis of the SIFT operator for automatic feature extraction and matching in photogrammetric applications, Sensors (14248220) 9 (5) (2009) 3745–3766, http://dx.doi.org/10.3390/s90503745 (ISSN: 14248220).

[17] T. Zhao, R. Nevatia, Car detection in low resolution aerial images, Image and Vision Computing, 21(8), Computer Vision, 2001. ICCV 2001, Proceedings. Eighth IEEE International Conference, Vol 1, IEEE, Pp: 710–717. Jul/07–14/2001, Vancouver, BC, 2003, http://dx.doi.org/10.1109/ICCV.2001.937593 (ISBN: 0–7695–1143-0).

[18] K. Kaaniche, B. Champion, C. Pegard, P. Vasseur, A vision algorithm for dynamic detection of moving vehicles with a UAV, Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005 2005, pp. 1878–1883, http://dx.doi.org/10.1109/ROBOT.2005.1570387 (April/18–22/2005. ISBN: 0-7803-8914-X).

[19] Z. Kim, J. Malik, Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking, Ninth IEEE International Conference on Computer Vision, 2003. Proceedings, vol. 1 2003, pp. 524–531, http://dx.doi.org/10.1109/ICCV.2003.1238392 (Oct/13–16/2003. Nice, France. ISBN: 0–7695–1950-4).

[20] J. Gleason, A.V. Nefian, X. Bouyssounousse, T. Fong, G. Bebis, Vehicle Detection from Aerial Imagery, 2011 IEEE International Conference on Robotics and Automation (Icra) 2011, pp. 2065–2070, http://dx.doi.org/10.1109/ICRA.2011.5979853 (May/9–13/2011. Shanghai. ISSN: 1050-4729. ISBN: 978–1–61284-386-5).

[21] J. Leitloff, S. Hinz, U. Stilla, Vehicle detection in very high resolution satellite images of city areas, IEEE Trans. Geosci. Remote Sens. 48 (7) (2010) 2795–2806, http://dx.doi.org/10.1109/TGRS.2010.2043109 (ISSN: 0196-2892).

[22] S. Tuermer, F. Kurz, P. Reinartz, U. Stilla, Airborne vehicle detection in dense urban areas using HoG features and disparity maps, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 6 (6) (2013) 2327–2337, http://dx.doi.org/10.1109/JSTARS.2013.2242846 (ISSN: 1939-1404).

[23] J. Leitloff, D. Rosenbaum, F. Kurz, O. Meynberg, P. Reinartz, An operational system for estimating road traffic information from aerial images, Remote Sens. 6 (11) (2014) 11315–11341, http://dx.doi.org/10.3390/rs61111315 (ISSN: 20724292).

[24] X. Cao, J. Lan, P. Yan, X. Li, Vehicle detection and tracking in airborne videos by multi-motion layer analysis, Mach. Vis. Appl. 23 (5) (2011) 921–935, http://dx.doi.org/10.1007/s00138-011-0336-x (ISSN: 0932-8092).

[25] X. Cao, C. Wu, P. Yan, X. Li, Linear SVM classification using boosting HOG features for vehicle detection in low-altitude airborne videos, 2011 18th IEEE International Conference on Image Processing (ICIP) 2011, pp. 2421–2424, http://dx.doi.org/10.1109/ICIP.2011.6116132 (Sep/11–14/2011. Brussels. ISSN: 1522-4880. ISBN: 978–1–4577-1304-0).

[26] F. Heintz, P. Rudol, P. Doherty, From images to traffic behavior—a UAV tracking and monitoring application, Information Fusion, 2007 10th International Conference. IEEE. Jul/9–12/2007. Quebec, Que, 2007, http://dx.doi.org/10.1109/ICIF.2007.4408103 (ISBN: 978–0–662-45804-3).

[27] F. Liu, X. Liu, P. Luo, Y. Yang, D. Shi, A new method used in moving vehicle information acquisition from aerial surveillance with a UAV, Advances on Digital Television and Wireless Multimedia Communications, Springer, Berlin Heidelberg 2012, pp. 67–72, http://dx.doi.org/10.1007/978-3-642-34595-1_10 (ISBN: 978–3–642-34594-4).

[28] L. Zhang, H. Wang, L. Li, Vehicle detection and tracking in video from moving airborne platform, J. Comput. Inf. Syst. 10 (12) (2014) 4965–4972, http://dx.doi.org/10.1007/s00138-011-0336-x (ISSN: 1432-1769).

[29] W. Ren, R.W. Beard, Trajectory tracking for unmanned air vehicles with velocity and heading rate constraints, IEEE Trans. Control Syst. Technol. 12 (5) (2004) 706–716, http://dx.doi.org/10.1109/TCST.2004.826956 (ISSN: 1063-6536).

[30] D. Rosenbaum, F. Kurz, U. Thomas, S. Suri, P. Reinartz, Towards automatic near real-time traffic monitoring with an airborne wide angle camera system, Eur. Transp. Res. Rev. 1 (1) (2008) 11–21, http://dx.doi.org/10.1007/s12544-008-0002-1 (ISSN: 1866-8887).

[31] B. Coifman, M. McCord, R.G. Mishalani, M. Iswalt, Y. Ji, Roadway traffic monitoring from an unmanned aerial vehicle, Intell. Transp. Syst., IEE Proc. 153 (1) (2006) 11–20, http://dx.doi.org/10.1049/ip-its:20055014 (ISSN: 1748-0248).

[32] M., Edward Donaldormack, T. Ted, The Use of Small Unmanned Aircraft by the Washington State Department of Transportation(No. WA-RD 703.1) Washington State Department of Transportation, 2008 (2008. URL: http://depts.washington.edu/trac/bulkdisk/pdf/703.1.pdf).

[33] F.K. Nejadasl, R. Lindenbergh, Sequential and automatic image–sequence registration of road areas monitored from a hovering helicopter, Sensors 14 (9) (2014) 16630–16650, http://dx.doi.org/10.3390/s140916630.

[34] E. Vincent, R. Laganiere, Detecting planar homographies in an image pair, Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium 2001, pp. 182–187, http://dx.doi.org/10.1109/ISPA.2001.938625 (Jun/19–21/2001. Pula. ISBN: 953–96769–4-0).

[35] S.E. Umbaugh, Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIPtools, second ed. CRC Press, Inc., Boca Raton, FL, USA, 2010 (ISBN: 9781439802052 - CAT# K10112).

[36] L. Shapiro, Computer Vision and Image Processing, Academic Press, 1992 (1992. ISBN: 0-12-638660-9).

[37] P.H.S. Torr, A.W. Fitzgibbon, A. Zisserman, The problem of degeneracy in structure and motion recovery from uncalibrated image sequences, Int. J. Comput. Vis. 32 (1) (1999) 27–44, http://dx.doi.org/10.1023/A: 1008140928553 (ISSN: 0920–5691).

[38] M.,A. Alex, Multiple view geometry in computer vision, Kybernetes 30 (9/10) (2001) 1333–1341, http://dx.doi.org/10.1108/k.2001.30.9_10.1333.2.

[39] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94 (ISSN: 1573-1405).

[40] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, Proceedings of the 7th International Joint Conference on Artificial Intelligence—Volume 2, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA 1981, pp. 674–679 (URL: http://dl.acm.org/citation.cfm?id=1623264.1623280).

[41] T. Lindeberg, Edge detection and ridge detection with automatic scale selection, Int. J. Comput. Vis. 30 (2) (1998) 117–156, http://dx.doi.org/10.1023/A:1008097225773 (ISSN: 1573-1405).

[42] C.S. Royden, K.D. Moore, Use of speed cues in the detection of moving objects by moving observers, Vis. Res. 59 (2012) 17–24, http://dx.doi.org/10.1016/j.visres.2012.02.006 (ISSN: 0042-6989).

[43] F. Karimi Nejadasl, B.G.H. Gorte, S.P. Hoogendoorn, Optical flow based vehicle tracking strengthened by statistical decisions, ISPRS J. Photogramm. Remote Sens. 61 (3–4) (2006) 159–169, http://dx.doi.org/10.1016/j.isprsjprs.2006.09.007 (ISSN: 0924-2716).

[44] D.G. Lowe, Object recognition from local scale-invariant features, The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, vol. 2 1999, pp. 1150–1157, http://dx.doi.org/10.1109/ICCV.1999.790410 (Sep/20–27/1999. Kerkyra. ISBN: 0–7695–0164-8).

[45] S. Se, D. Lowe, J. Little, Vision-based mobile robot localization and mapping using scale-invariant features," IEEE International Conference on Robotics and Automation, 2001, Proceedings 2001 ICRA, vol. 2 2001, pp. 2051–2058, http://dx.doi.org/10.1109/ROBOT.2001.932909 (ISSN: 1050-4729. ISBN: 0–7803–6576-3).

[46] P.G. Gipps, A behavioural car-following model for computer simulation, Transp. Res. B Methodol. 15 (2) (1981) 105–111, http://dx.doi.org/10.1016/0191-2615(81)90037-0 (ISSN: 0191-2615).

[47] A. Vedaldi, B. Fulkerson, Vlfeat: an open and portable library of computer vision algorithms, Proceedings of the International Conference on Multimedia, ACM, New

York, NY, USA 2010, pp. 1469–1472, http://dx.doi.org/10.1145/1873951.1874249 (ISBN: 978–1–60558-933-6).
[48] P. Hidas, Modelling lane changing and merging in microscopic traffic simulation, Transp. Res. C: Emerg. Technol. 10 (5–6) (2002) 351–371, http://dx.doi.org/10.1016/S0968-090X(02)00026–8 (ISSN: 0968-090X).
[49] T. Toledo, H. Koutsopoulos, M. Ben-Akiva, Modeling integrated lane-changing behavior, Transp. Res. Rec. 1857 (1) (2003) 30–38, http://dx.doi.org/10.3141/1857-04 (ISSN: 0361–1981).
[50] Q. Yang, H.N. Koutsopoulos, A microscopic traffic simulator for evaluation of dynamic traffic management systems, Transp. Res. C: Emerg. Technol. 4 (3) (1996) 113–129, http://dx.doi.org/10.1016/S0968-090X(96)00006-X (ISSN: 0968-090X).
[51] V. Punzo, M.T. Borzacchiello, B. Ciuffo, On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (NGSIM) program data, Transp. Res. C: Emerg. Technol. 19 (6) (2011) 1243–1262, http://dx.doi.org/10.1016/j.trc.2010.12.007 (ISSN: 0968-090X).
[52] Z. Sun, G. Bebis, R. Miller, On-road vehicle detection: a review, IEEE Trans. Pattern Anal. Mach. Intell. 28 (5) (2006) 694–711, http://dx.doi.org/10.1109/TPAMI.2006.104 (ISSN: 0162-8828).
[53] J. Cao, M. Hadiuzzaman, T.Z. Qiu, D. Hu, Real-time queue estimation model development for uninterrupted freeway flow based on shockwave analysis, Can. J. Civ. Eng. 42 (3) (2015) 153–163, http://dx.doi.org/10.1139/cjce-2014-0218 (ISSN: 0315-1468).