Vehicle Feature Point Trajectory Clustering and Vehicle Behavior Analysis in Complex Traffic Scenes

Xuan Wang(✉), Jindong Zhao, Yingjie Wang, Jun Lv, and Weiqing Yan

Yantai University, Yantai 264005, China
xuanwang_91@126.com

Abstract. Video-based analysis technology has a wide range of applications in intelligent transportation system (ITS). Vehicle segmentation and behavior analysis has become an important research area in traffic video analysis. To solve the problem of 2D video detection technology in actual traffic video scenes, a bottom-up analysis method is employed to study the related technical problems. Firstly, M-BRISK descriptor algorithm is proposed for describing local feature points, which based on the method of original BRISK. Secondly, a 3D feature analysis method based on rigid motion constraints for vehicle trajectory is proposed. With the result of camera calibration and the preset back-projection plane, the 2D trajectory points can be back-projected to the 3D space, and the back projection data of the 2D image can be reconstructed in 3D space. Thirdly, similarity measure method is proposed for achieving the trajectory clustering. The experimental results show that the proposed method not only accelerates the speed of clustering method, but also improves the accuracy of trajectory clustering at some extent. Moreover, the vehicle motion information contained in the trajectory data can be analyzed to recognize vehicle behavior. All of these provide an important data foundation for vehicle abnormal behavior detection and the identification of traffic status levels in traffic scenes.

Keywords: Vehicle segmentation · Feature point detection · Trajectory clustering · Behavior analysis

1 Introduction

Vehicle motion segmentation and vehicle behavior analysis are important research areas in complex traffic video. In the past few decades, with the increasing coverage of traffic video surveillance, a large number of research scholars have been attracted to the key technology research of traffic video analysis. Nowadays, road monitoring equipment has been spread all over traffic junctions and road sections, and video surveillance has become the most direct and effective way to monitor the real-time operation of road traffic, as shown in Fig. 1. With

B. Li et al. (Eds.): IoTaaS 2019, LNICST 316, pp. 182–205, 2020.
https://doi.org/10.1007/978-3-030-44751-9_17

the development of computer hardware devices, many video analysis methods [2,7,11,12,25,28,29] for vehicle detection can achieve the requirement of real-time detection.

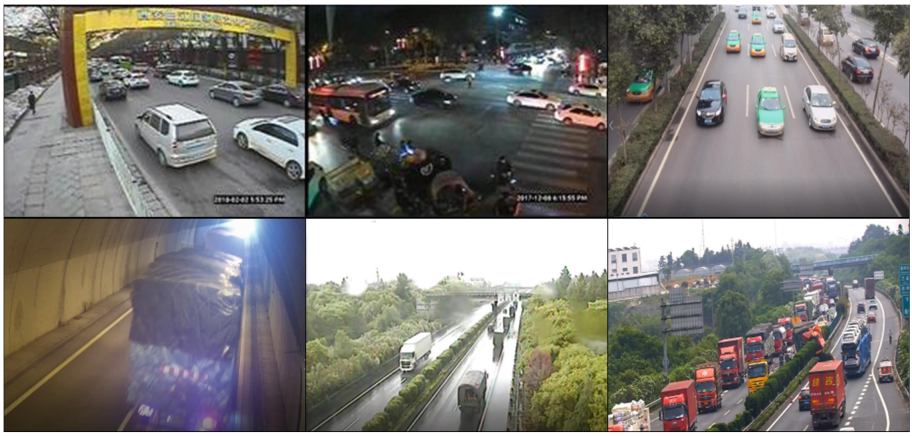


Fig. 1. Traffic video scene.

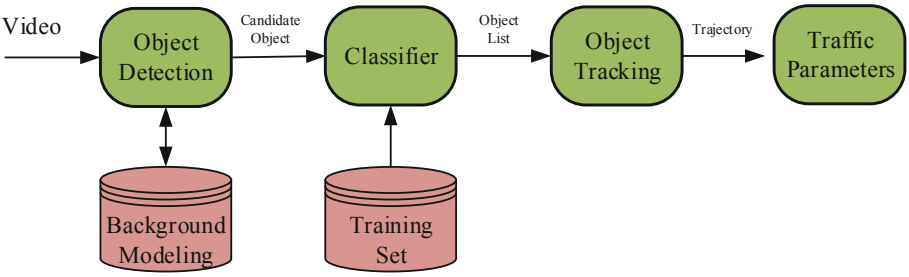


Fig. 2. Top-down traffic video analysis system model.

There are many key technologies in the intelligent development of traffic monitoring systems. These methods are used to understand how traffic video analysis works. In general, it can be summarized into two categories: the top-down approach and the bottom-up approach. The specific process is shown in Figs. 2 and 3.

The top-down approach has obvious advantages in high-definition video and smooth traffic environments. If the vehicle targets are not occluded, these methods have high detection and tracking accuracy using the appropriate classifier, but these methods have high computational complexity and low operating efficiency. They may be difficult to meet the requirement of real-time. In addition, the actual traffic scene is difficult to predict, the mutual occlusion of the vehicle

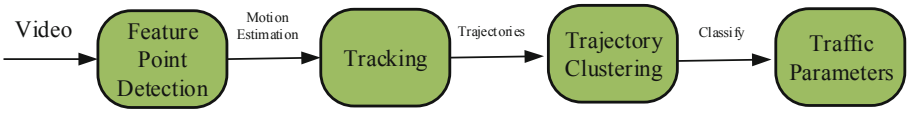


Fig. 3. Bottom-up traffic video analysis system model.

and other various environmental factors greatly affect the robustness of these methods. The bottom-up approach uses a partial-to-integral analysis method based on the feature point detection of the vehicle target, and gradually completes the object segmentation process even if the vehicle target is partially blocked during the motion. That is because the other local feature points still able to be detected, then the tracking task can be completed. After that, the behavior analysis method is needed to be performed. Moreover, the detection algorithm based on the feature point is highly efficient, and can better meet the real-time requirement in practical applications.

In the bottom-up system framework, researchers are working on various local features such as edges, corners, parts, and spots. In recent years, the local feature descriptors have a great development. The typical algorithm is SIFT proposed by Lowe [16] in 2004. The algorithm uses the gradient information around the feature point to describe it and use the image pyramid to solve the scale problem. Thus, SIFT feature descriptor has good scale invariance and rotation invariance. Subsequently, many researchers proposed some improved algorithm, such as PCA-SIFT [32], GLOH [17], SURF [3], DAISY [27] and so on.

After expressing some apparent features of the moving objects effectively, it is necessary to use a similarity measure algorithm to perform feature matching on the video sequence to complete the object tracking process [9]. The common similarity measures are including Euclidean distance, Gaussian distance, Block distance, Hamming distance, Chessboard distance, Manhattan distance, Weighted distance, Chebyshev distance, Barth Charlie coefficient, Hausdorff distance, etc. And the simplest should be the Euclidean distance. In the object tracking process, if we directly search and match the video scene globally to determine its optimal matching position, it will inevitably have to deal with a lot of redundant information, and also greatly increase the computing amount of the computer and reduce its computing speed. Therefore, it is of great significance to use a specific search algorithm to estimate and calculate the position of the object in the next moment to narrow the scope of searching.

One common way is to predict the location of the moving object in the next frame and find the best matching position in the vicinity area, such as Kalman filtering [10], extended Kalman filtering [30], and particle filtering. Another way is to continually optimize the direction of searching to speed up the process of searching and matching, such as Mean Shift [5] and Camshift [6].

Behavioral understanding of the moving object can be achieved through the trajectory pattern analysis. In the process of discriminating the trajectory mode, trajectory feature extraction and learning method of trajectory pattern are two

important steps, which have an important influence on the realization of the trajectory behavior recognition. In terms of trajectory feature selection, Buzan [8] uses the method of calculating the longest common subsequence to realize the clustering and retrieval of motion trajectories; based on the Euclidean distance between trajectories, Hu et al. [31] used multi-level clustering method to perform equal-dimensional processing on vehicle trajectories in order to solve the classification problem of trajectories; There are two main methods for learning the behavior trajectory: neural network-based learning method and unsupervised clustering-based learning method. Johnson [13] and Sumpter [26] used a self-organizing feature map (SOM) neural network approach to modeling the spatial pattern of motion trajectories. Hu et al. [31] used the fuzzy SOM method to learn the motion trajectory and behavior pattern of the target to realize the detection and discrimination of abnormal events.

The bottom-up video analysis method can solve the problem of vehicle segmentation in complex traffic scenes, and it has higher operational efficiency. Therefore, based on the design ideas of this kind of video analysis method, this paper conducts related research, taking the image local feature points as the research object, using the tracking matching algorithm to obtain the 2D motion trajectory of the vehicle feature points, and analyzing the clustering problem between the trajectories based on the rigid motion constraint. The main contribution of this paper are as following:

- A feature extraction algorithm based on BRISK is proposed for complex traffic scenes. In terms of feature detection, we uses the adaptive FAST algorithm to detect the feature points in the scale space. In terms of feature description, we constructs a hybrid binary feature descriptor based on BRISK. The method can not only guarantee calculation rate, but also extract and locate the feature points effectively.
- A 3D feature analysis method of vehicle trajectory based on rigid motion constraints is proposed. Camera calibration is used to build the back projection data of the image in 3D space. Combined with the idea of back projection, the relative height between different trajectories is obtained based on the rigid motion constraint. Then, the estimated values of relevant traffic information of feature points corresponding to each trajectory in 3D space are further obtained.
- Using the extracted 3D information estimation of feature point trajectory to construct a new similarity measure between trajectories, and applying it to the framework of spectral clustering algorithm to realize the vehicle feature point trajectory clustering in 3D space.
- Based on the 3D information of vehicle feature point trajectory and its clustering results, the behavior model and semantic analysis of vehicle trajectories in traffic scene are carried out, and the traffic prevalence of actual roads is analyzed.

The rest of this paper is organized as follows. An overview of the system is presented in Sect. 2. Section 3 describes the method of feature extraction. 3D

feature reconstruction is presented in Sect. 4. Section 5 displays the vehicle trajectory clustering method. Vehicle behavior analysis is given in Sect. 6. Experimental results are reported in Sect. 7 and finally Sect. 8 draws the conclusion.

2 Overview of the System

According to the specific research process of vehicle trajectory extraction and behavior analysis, the overall technical framework of this paper is shown in Fig. 4. The research content is mainly divided into three points: vehicle feature point trajectory extraction, trajectory feature extraction and cluster analysis, and vehicle behavior analysis. The vehicle feature point trajectory extraction is the basis of the latter link. It mainly studies the feature point detection and stable tracking of the vehicle target in the video sequence, and then obtains the trajectory data of the vehicle feature point in the 2D image plane. The feature extraction and cluster analysis of the trajectory are based on the camera calibration of the real traffic scene. The motion characteristics of the vehicle trajectory in 3D space are obtained by rigid motion constraint analysis, and the similarity measure is constructed to realize the clustering segmentation. Vehicle behavior analysis is based on the 3D trajectory to obtain the traffic parameters in the actual traffic scene, using the prior knowledge to semantically express the vehicle trajectory, providing a data foundation for further analysis of the individual vehicle behavior and the traffic flow behavior.

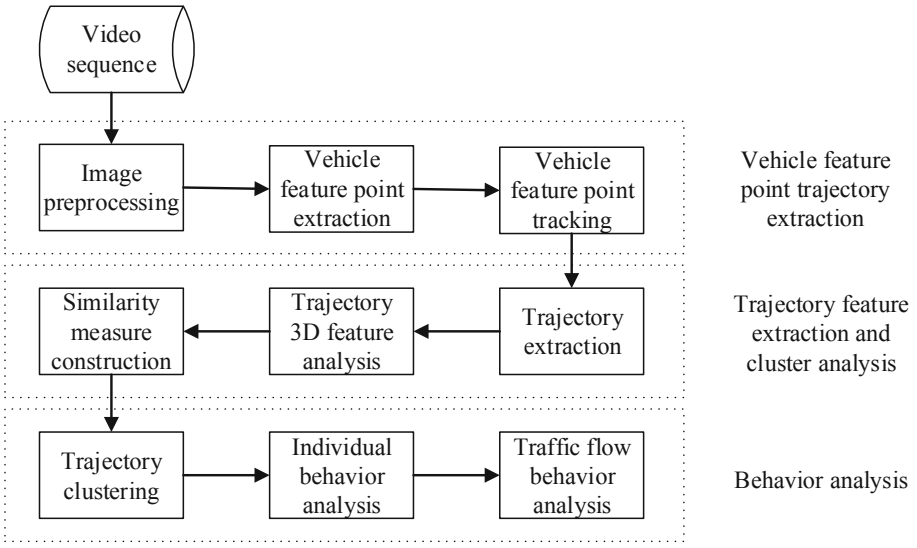


Fig. 4. Overview of the system framework.

3 Feature Extraction

In recent years, researchers have proposed several methods for binary feature description for real-time applications, such as BRIEFF [4], ORB [24], BRISK [15] and FREAK [1]. In fact, achieving high quality features and maintaining low computational costs is very challenging. This paper proposes a feature point detection algorithm based on the improved BRISK algorithm for complex traffic scenes. In the aspect of feature point detection, it uses the adaptive FAST algorithm mask to detect the feature points of the scale space. In the aspect of feature point description, it constructs a hybrid binary structure feature descriptor based on BRISK algorithm.

3.1 Feature Point Detector

FAST (Features from accelerated segment test) [22,23] is a corner detection algorithm proposed by Edward Rosten and Tom Drummond. The most outstanding advantage of this algorithm is that the computational efficiency is very high. Its computational speed is as fast as its name, and it is more efficient than other mainstream algorithms (such as SIFT, SUSAN, Harris). And if the machine learning method is applied to the FAST algorithm, it can show better results. The FAST corner detection algorithm is often used for video processing research due to its speed advantage. The principle of the FAST corner point is: if a pixel point and a specific number of pixels in its surrounding area are located in a different area, the pixel point is called a corner point. That is to say, some attributes are irrelevant. In the case of grayscale images, the gray value of the point is smaller or larger than the gray value of the point in its surrounding area, then the pixel may be a corner, as shown in Fig. 5.

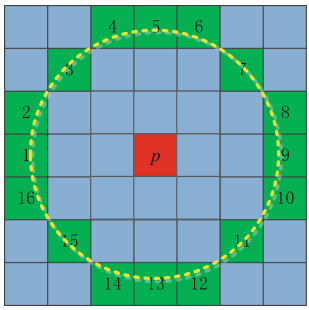


Fig. 5. FAST algorithm principle diagram.

3.2 Feature Point Descriptor

The FAST algorithm only performs feature point detection in the image, but does not further describe the feature points, so it can not apply the feature points

to the process of image matching and tracking. Therefore, researchers have proposed many feature descriptors based on the feature points of FAST detection, such as ORB, BRISK, FREAK and so on. Based on the BRISK algorithm, a hybrid binary descriptor is proposed in this paper. The image pyramid is constructed by using the Brisk algorithm in the scale space. Then, the information of the BRISK algorithm is enriched by the information of the local downsampling. This method improve the robustness of the original BRISK.

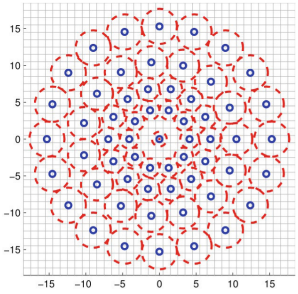


Fig. 6. Sampling mode of BRISK.

The sampling mode of BRISK is shown in Fig. 6. It can be found that the BRISK algorithm only considers the intensity relationship between the sampling points, that is, only the pairwise intensity comparison between the sampling point positions is considered. The local information of the sampling point is lost, which makes the algorithm unstable. Therefore, the basic idea of this paper is to construct feature descriptors based on the local information of the sampling points and the information between the pairs of sampling points to improve the robustness of the original BRISK.

Let $\mathbf{\Pi}$ be a set of all N sample point positions, for each sample position $\mathbf{p}_i^\alpha = (x_i, y_i) \in \mathbf{P}$, uniform sampling of four points $S(\mathbf{p}_i^\alpha) = \{s_{i,k}^\alpha, k = 1, 2, 3, 4\}$ is performed on a circle of radius R centered on $\mathbf{p}_i^\alpha$, where α is the local main direction, this paper used the Intensity Centroid [21] algorithm to calculate the main direction of the feature points. According to the LBP operator [19], local information can be encoded by the gray relationship between the sampling position $\mathbf{p}_i^\alpha$ and each local sampling point $s_{i,k}^\alpha$. However, this encoding is sensitive to the center point $\mathbf{p}_i^\alpha$, so it was not applied to binary descriptions. In order to encode local information robustly, this paper uses the gray relation between local sample points $s_{i,k}^\alpha$ for encoding.

Assuming $I(\mathbf{p}_i^\alpha, \sigma)$ is the smoothed gray value of point $\mathbf{p}_i$, and σ is the Gaussian filter variance. For each rotated sample position $\mathbf{p}_i^\alpha$, the paired gray values of the local sample points $s_{i,k}^\alpha \in S(\mathbf{p}_i^\alpha)$ are compared. A local gradient binary descriptor is constructed by combining all test results into a binary string, each bit b corresponding to:

$$b = \begin{cases} 1, & I(s_{i,k}^{\alpha}, \sigma_i) > I(s_{i,t}^{\alpha}, \sigma_i) \\ 0, & otherwise \end{cases} \quad \begin{array}{r} \forall \mathbf{p}_i^{\alpha} \in \mathbf{P} \wedge s_{i,k}^{\alpha}, s_{i,t}^{\alpha} \in S(\mathbf{p}_i^{\alpha}) \\ \wedge k, t = 1, 2, 3, 4 \wedge k \neq t \end{array} \tag{1}$$

Since the local sampling position of each sample point has four points, the dimension of this feature descriptor is $N \times C_4^2 = 6N$ bits. It should be noted that the gray scale comparison between local sample points $s_{i,k}^{\alpha}$ is closely related to the local gradient operator, because they both consider the gray difference between the local sample pairs.

The feature descriptor of the above construction encodes the local information of the sampling point into a binary string. We further supplements it with the global information of the sample points, which is encoded by the gray intensity comparison between the sample points. Use set $\mathbf{A}$ to represent the all combined results of the sample point pairs:

$$\mathbf{A} = \{(\mathbf{p}_i^{\alpha}, \mathbf{p}_j^{\alpha}) | \mathbf{p}_i^{\alpha}, \mathbf{p}_j^{\alpha} \in \mathbf{P} \wedge i \neq j\} \tag{2}$$

Furthermore, a subset $\mathbf{B}$ in which it has M pairs of sample points is selected from $\mathbf{A}$, so that each bit b of the binary descriptor is constructed by:

$$b = \begin{cases} 1, & I(\mathbf{p}_j^{\alpha}, \sigma_j) > I(\mathbf{p}_i^{\alpha}, \sigma_i) \\ 0, & otherwise \end{cases} \quad \forall (\mathbf{p}_i^{\alpha}, \mathbf{p}_j^{\alpha}) \in \mathbf{B} \tag{3}$$

In this part, we construct the feature descriptor in a manner consistent with the original BRISK. The same is that the short-range pairs of sample points are used to construct the feature descriptor. The difference is that the M-sample point pairs of the shortest distance are only a supplementary part of the previous local gradient-based binary feature descriptor. The mixed BRISK descriptor (M-BRISK) is constructed by the above two steps of binary string.

4 3D Feature Reconstruction

4.1 Inverse Projection Transformation

The imaging process of the camera is a description of the loss of information in 3D real space, and this process is irreversible. At present, most of the methods for image detection, tracking and behavior analysis are based on 2D image plane. However, due to the perspective transformation of the camera imaging, the geometrical and motion characteristics inherently of the objects are no longer exist in the 2D image plane. For example, some geometric features such as symmetry, parallelism, vertical and circular will be changed due to perspective projection transformation; the same moving object has obvious scale changes at different positions in the video sequence; and vehicles with uniform motion in 3D world space are performed non-uniform motion in the 2D video sequence, and so on. All of the above situations will make the related algorithms based on 2D image

facing great difficulties. In order to solve the segmentation problem of vehicles in complex traffic scenes, it is necessary to extract the 3D information of the vehicle. This paper proposes a method based on rigid motion constraints for vehicle 3D trajectory feature analysis.

Camera calibration is the important part for obtaining the 3D parameters of objects based on video/image. However, the trajectory clustering and behavior analysis under the monocular camera is based on the vehicle feature point trajectory in 3D space. It is an important precondition for the subsequent algorithm to obtain the transformation between the 2D image and the 3D space. The working process of the camera model is:

$$\lambda p = K \begin{bmatrix} R & T \end{bmatrix} P_W = H P_W \tag{4}$$

where, $H = K \begin{bmatrix} R & T \end{bmatrix}$, $p = [u, v, 1]^T$, $P_W = [X_W, Y_W, Z_W, 1]^T$, λ is the scale factor, K is the camera internal parameters, R and t compose the external parameter matrix of the camera. The internal and external parameters can be calculated accurately by the recovery method of vanishing points [14].

Camera imaging is a perspective projection process from 3D space to 2D image. Conversely, the transformation process of mapping 2D image to 3D space is called inverse perspective mapping (IPM). In order to obtain reconstructed images with perspective effects through back-projection transformation, researchers can only use existing constraints and prior knowledge to make certain reasoning and estimation [20]. A common method is to first use the transformation relationship and the constraints to achieve the location mapping, and then fill the data. In order to describe this inverse transformation process more clearly, the mapping process is marked as:

$$p_I = F \cdot P_W \tag{5}$$

where F represents a transformation matrix of the 3D world coordinate system and the 2D image coordinate system. If a certain image coordinate p_I and one of its coordinate value in the 3D world coordinate system are known, for instance, if the actual height Z_W of the coordinate point is known, the specific position P_W in the 3D world which corresponding to p_I can be obtained. The process is expressed as:

$$P_W = F^{-1} \cdot (P_I \bigoplus Z_W) \tag{6}$$

In terms of mathematical theory, if the pixel points in the 2D image plane are back-projected into the 3D world coordinate system directly, unique solution can not be obtained owing to the uncertain scale parameter. However, if a certain dimensional coordinate parameter in the 3D space is determined, the 3D coordinate corresponding to the 2D pixel coordinate can be obtained uniquely. Therefore, we can preset a back-projection plane in 3D space, that is, to determine information of a certain dimension, so that the data of inverse projection transformation can be obtained on the back-projection plane.

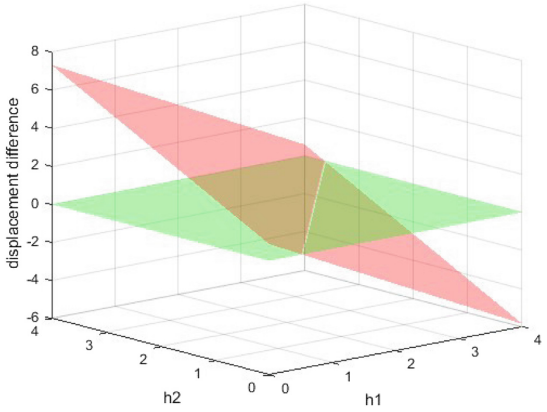


Fig. 7. Rigid motion constraint.

4.2 Rigid Motion Constraint

This section simulates the motion trajectories of feature points on a rigid body in 3D space, and they are back-projected onto several back-projection planes paralleling to the road surface. In the actual scene, the height range of the vehicles have a certain limitation. Generally, the height of vehicles is no more than 4 m. Therefore, the 3D information of the vehicle trajectory points can be obtained indirectly. Specifically, we use the enumeration method to test the trajectory height information to reconstruct the trajectory information in 3D space, and use the rigid motion constraints to calculate the height relationship between different reconstructed 3D trajectories.

$$D(P_{fi}^{h_1}, P_{fj}^{h_2}) = D(P_{Fi}^{h_1}, P_{1i}^{h_1}) - D(P_{Fj}^{h_2} - P_{1j}^{h_2}) \quad h_1, h_2 = 0, \ldots, 4 \tag{7}$$

where, $D(P_{Fi}^{h_1}, P_{1i}^{h_1})$ is the displacement of the feature point p_i during F frame in 3D space, $P_{fi}^{h_1}$ is reconstructed 3D trajectory from the 2D trajectory point p_{fi} with the height information h_1, and $D(P_{fi}^{h_1}, P_{fj}^{h_2})$ is the displacement difference of the two reconstructed 3D trajectories. Figure 7 shows the relationship between the displacement differences of two trajectories in an ideal case (without tracking error) using the height enumeration method. It can be found that the heights of the two trajectories and their displacement are on the same plane:

$$ah_1 + bh_2 + c = Diff \tag{8}$$

If the two trajectories belong to the same car, the displacement value is the same, that is, the $Diff$ is zero. Therefore, the height relationship of the two trajectories can be obtained:

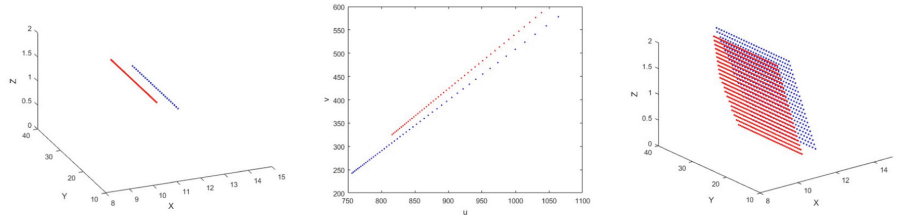
$$ah_1 + bh_2 + c = 0 \tag{9}$$

4.3 3D Information Reconstruction

Based on the idea of back-projection transformation, we set the vehicle moving direction (ie, the road direction) as the Y_W direction, and constructed multiple back-projection planes parallel to the Y direction. Then, we reconstructed the 3D trajectory information from the 2D trajectories with the known enumeration value of Z_W. According to Eq. 4, (X_W, Y_W) can be calculated as follows:

$$\begin{aligned} Y_W &= \frac{A - B(H_{31}u - H_{11})}{(H_{32}u - H_{12})(H_{31}v - H_{21}) - (H_{32}v - H_{22})(H_{31}u - H_{11})} \\ X_W &= \frac{A - Y_W(H_{32}u - H_{12})(H_{31}v - H_{21})}{(H_{31}u - H_{11})(H_{31}u - H_{21})} \end{aligned} \tag{10}$$

where, $A = (Z_W H_{13} + H_{14} - v(Z_W H_{33} + H_{34}))(H_{31}v - H_{21})$, $B = Z_W H_{23} + H_{24} - v(Z_W H_{33} + H_{34})$. Thereby, it is possible to recover the 3D trajectory of the vehicle target at different height planes. As shown in Fig. 8(a), we simulated the trajectory of the same vehicle in an ideal state. The 2D image projection in the calibration scene is shown in Fig. 8(b). It can be found that the feature point trajectories of different heights have different pixel displacement and pixel speed. Therefore, the 3D trajectory can be estimated by constructing the different projection planes of different heights as shown in Fig. 8(c). Using these trajectory information and the motion characteristics of the rigid objects, this paper attempts to analyze the real position information and a series of 3D features of vehicles in 3D space, such as the actual speed, acceleration, displacement, driving direction, etc.



(a) (b) (c)

Fig. 8. 3D information reconstruction.

From the above, the specific algorithm for 3D feature extraction of vehicle trajectories in traffic scenes is as follows:

- Calibrate the camera in the traffic scene and obtain the transformation matrix H;
- Use enumeration method to set the back projection planes of different heights in the range of 0–4 m, and back-project the 2D trajectories on to the different back projection planes;

- Calculate the velocity $V(i,h)$ of each feature point trajectory on different inverse projection planes, where i represents the i-th trajectory and h represents the height of the back projection plane;
- Use the K-mean algorithm to achieve the clustering analysis, and classify the trajectory data by velocity difference;
- Perform the histogram statistics on $V(i,h)$ of each set of trajectories, and use the velocity interval with the highest frequency as an estimated value of the real velocity in 3D space;
- Calculate height information of each cluster by using the spatial relationship and the estimated velocity;
- Reconstruct the position information of the feature points in the 3D space by combining the height information of each cluster.

5 Trajectory Clustering

5.1 Clustering Algorithm

The similarity measure is an important basis for data mining techniques such as data classification, clustering and abnormal behavior recognition. This paper constructs a new similar measure relationship between the trajectories using 3D information of feature point trajectories and applies it to the spectral clustering algorithm.

The specific implementation steps of the algorithm are as follows:

- Construct a similarity matrix W between the trajectories according to the similarity measure between the trajectory data;
- Calculate its normalized Laplacian matrix $L = D^{-\frac{1}{2}}(D-W)D^{-\frac{1}{2}}$;
- Calculate the eigenvalues $\{\lambda_i, i = 1,2,\cdots,n\}$ and eigenvectors $\{E_i, i = 1,2,\cdots,n\}$ of L;
- Calculating an indication feature vector Q_i corresponding to E_i;
- Perform the K-means clustering using the feature vectors corresponding to the first k minimum eigenvalues of Q.

5.2 Similarity Measure

Based on the 3D feature analysis of the vehicle feature point trajectory, we performed 3D feature extraction for each feature point trajectory, and constructed an attribute feature vector $F = (H, V, X, Y)$ that can represent each trajectory information, where H represents the relative height between the trajectory and the reference trajectory, V indicates the 3D velocity of the trajectory reconstructed by the trajectory set T, and (X, Y) represents the 3D coordinate of the trajectory point at a certain time. F covers not only the feature information inherent of each trajectory, but also the relative positional relationship between the trajectories. Therefore, we use the trajectory set T to extract the eigenvector F corresponding to each trajectory, and combine the Gaussian similarity calculation model to construct a new similarity measure S:

$$S(T_i, T_j) = \exp(-\frac{d^2(F_{T_i}, F_{T_j})}{2\sigma^2}) \tag{11}$$

where, $d(F_{T_i}, F_{T_j})$ is the Euclidean distance of the attribute feature vector extracted by any two trajectories in the trajectory set T, F_{T_i} is a 1×4 feature vector which includes four parameters of the trajectory, and σ is the scale factor.

Since the research object of this paper is the vehicles, the distribution of feature points on the same vehicle is limited. It means that the X coordinate range of feature point trajectory in the same vehicle can not exceed the width of the vehicle itself. Using this property, the similarity matrix W between vehicle trajectory sets is further constructed by:

$$W(T_i, T_j) = \begin{cases} \exp(-\frac{d^2(F_{T_i}, F_{T_j})}{2\sigma^2}) & d(X_{T_i^f}, X_{T_j^f}) \leq \xi \\ 0 & otherwise \end{cases} \tag{12}$$

where, ξ is a threshold parameter according to the actual outer contour size standard of the road vehicle, but since the reconstructed trajectory 3D information is an estimated value, there is a certain error between estimated value and the real value.

In the case of a given set of trajectories, the construction process of the similar matrix is as follows:

- Calculate the 2D velocity $v = \{v_1, v_2, \ldots, v_n\}$ of each 2D trajectory of the trajectory set, and select the trajectory with the minimum 2D velocity v_p as the reference trajectory T_p;
- Calculate the relative height between each trajectory and the reference trajectory.
- Use the enumeration method to construct different heights of the back projection planes in the range of 0–4 m to recover each trajectory in 3D space;
- Calculate an estimated velocity of each trajectory in 3D space and the spatial position of the feature point at the current frame;
- Construct the attribute feature vector $F_{T_i} = (H_i, V_i, X_i, Y_i)$ of each trajectory;
- Calculate the similarity matrix W between the trajectory data set T using the Eq. (12);

6 Vehicle Behavior Analysis

6.1 Vehicle Individual Behavior Analysis

In this section, the 3D information of the vehicle trajectory is used to further analyze the behavior pattern of the individual vehicle in traffic scene, so as to detect the abnormal behavior of the vehicle. Many related threshold information are contained in this section, and for a determined traffic scene, the associated threshold information is the same. These threshold information is dependent on theoretical calculations and empirical values.

Over-Speed and Low-Speed Driving. According to China's Road Traffic Safety Law, the highway sections should identify the limits of their driving speed clearly. For example, the maximum speed of vehicles on the highway cannot exceed 120 km/h, and the minimum speed cannot be lower than 60 km/h.

Therefore, the speed limit value of the road section can be obtained for the determined road section of highway. If $V_i > V_\alpha$, the vehicle is judged to be over-speed; if $V_i < V_\beta$, the vehicle is determined to be low-speed, where V_i is the estimated value of the real speed of the i-th vehicle, V_α and V_β are the maximum speed and the minimum speed of the road section respectively.

Retrograde. The camera has a fixed installation position and angle in the traffic scene. Firstly, we determine the correct driving direction of the road manually based on the driving direction of the vehicle in the traffic video. Then, we use camera calibration technology to obtain the 3D position information of the direction marking line and its direction vector. As shown in Fig. 9, one direction vector can be set in the two-lane road section, and two or more correct direction vectors should be set according to the actual situation.

Fig. 9. Setting of the correct driving direction.

According to the 3D trajectory information of the vehicle feature point, the motion vector (X_i^f, Y_i^f) of each frame can be determined, and the information of the X direction is used to select the correct driving direction of the road for retrograde event discrimination. If (X_R, Y_R) indicates the correct direction of the road, the direction angle of the vehicle can be obtained:

$$\bar{\theta} = \frac{1}{m}\sum_{i=0}^{m} |\arccos(\frac{X_R X_i^f + Y_R Y_i^f}{\sqrt{X_R^2 + Y_R^2}\sqrt{(X_i^f)^2 + (Y_i^f)^2}})| \tag{13}$$

$$IsRetrograde = \begin{cases} true & \bar{\theta} \geq \alpha \\ false & \bar{\theta} < \alpha \end{cases} \tag{14}$$

where, (X_i^f, Y_i^f) is the direction of motion of the i-th trajectory in the same category at the f frame, and α is the empirical threshold. In the actual application

process, the fault tolerance of the algorithm needs to be considered. The vehicle behavior cannot be judged according to the data at a certain moment. Instead, it should be counted whether the direction angle of the vehicle motion satisfies the retrograde condition for a period of time. This paper counts the number l that the vehicle direction angle is greater than the empirical threshold α for a period of time. If $l > \beta$, we consider the vehicle as a retrograde vehicle.

Parking. If an abnormal parking event occurs, the feature point trajectory of the vehicle has obvious characteristics. It is embodied in a state in which the speed of the vehicle gradually decreases to zero, and the position information tends to be constant. The discriminating rules are as follows:

- If $V_k^f < \xi$, the counter of abnormal speed is incremented by 1.
- If $Isstop > \eta$, the vehicle has an abnormal parking event.

where, V_k^f indicates the instantaneous speed of the k-th cluster of the vehicle at the f-th frame, ξ is the minimum speed threshold and η is the speed anomaly threshold.

Abnormal Lane Change. The lane change behavior of the vehicle occurs more frequently in actual traffic, while the road is divided by solid lines (white solid line, yellow solid line, double yellow solid line). These solid lines are forbidden to be touched during the driving.

For the normal driving vehicle, its movement trend is along the direction of the lane line, that is, its motion trajectory is approximately parallel with the road marking line. However, there is a certain angle between the trajectory of the vehicle in which the lane changing behavior occurs and the road marking line. Therefore, this paper uses the following method to determine the abnormal lane change behavior of the vehicles:

- For a specific traffic road, camera calibration is performed manually to obtain the actual 3D space coordinates of the solid line marker line on the road;
- Calculate the variance in the X direction using the 3D trajectory information of the vehicle feature points in each category:

$$\bar{S} = \frac{1}{mn} \sum_{t=1}^{m} \sum_{i=1}^{n} (X_t(i) - \bar{X}_t)^2 \tag{15}$$

- If $\bar{S} > \gamma$, it is considered that the driving behavior of the vehicle is a lane change, and it is necessary to further judge whether the behavior is a violation of the rules. If $|X_t(i) - X_{Road}| < \varepsilon$, it regards the vehicle as violation of rules.

where, $X_t(i)$ represents the X coordinate of the i-th point of the t-th trajectory in 3D space, $\bar{X}_t$ is the average of the X coordinates of the t-th trajectory; X_{Road} indicates the X coordinate of the solid line marker on the road in 3D space; γ, ε are the experience threshold, which can be determined based on the specific scene.

Traffic Flow Behavior. In this section, the 3D information of the vehicle motion trajectory and the clustering results are used to calculate the traffic flow and traffic flow speed of a certain road section, which can be used to evaluate the real-time traffic status.

Traffic Flow. In order to fully consider the time series of the trajectory points during the motion, previous clustering results is combined to filter the trajectory data of the current frame, so the attribute feature extraction and cluster analysis are only carried out for the newly added trajectory data. It not only reduces the amount of calculation, but also improves the accuracy of clustering of newly trajectory data in some extent. The specific strategy can be described as:

- Set the time interval t of the clustering based on the video rate. That means cluster analysis is performed on the feature point trajectory in the current interest region every interval t frame;
- The trajectory data is filtered twice before each run of the clustering algorithm. One is to screen out trajectory data that meets a certain length; the other is to filter out the new trajectory data.
- Count the clustering results obtained each time, then the traffic flow per hour or day of the road section is obtained.

Traffic Flow Speed. In order to facilitate the measurement and calculation, this paper selects the interval average speed as the measurement index of the traffic speed. In the selected observation section, several instantaneous moments are selected at fixed time intervals, and the average value of the instantaneous speeds of all vehicles at that moment is calculated by using the vehicle feature point trajectory 3D information. The specific formula is as follows:

$$\bar{v_s} = \frac{1}{MN} \sum_{k=1}^{N} \sum_{i=1}^{M} \frac{s_k(i)}{\triangle t} \tag{16}$$

where, $\triangle t$ is the time interval of adjacent frames, $s_k(i)$ is the distance during the time interval between the current frame and the previous frame at the ith feature point of the kth vehicle.

7 Experimental Results

In this section, we evaluate the performance of the proposed system. In Sect. 7.1, the performance analysis of feature descriptor is evaluated. The trajectory clustering results on the different traffic videos are shown in Sect. 7.2. Section 7.3 performs the application results of vehicle behavior analysis.

7.1 Performance Analysis of Feature Descriptor

The performance indicators of the feature descriptors were evaluated using the recall and $1 - precision$ curves proposed in [18]. This paper compared the M-BRISK descriptor with the SURF, ORB, BRISK and FREAK descriptors. Since SURF is a classic fast descriptor, ORB, BRISK and FREAK are recently proposed binary descriptors. For fair comparison, image blocks of the same size (31×31) are set for all test descriptors and the different images of Oxford dataset are used for correlation test. The original picture of the data set is shown in Fig. 10. Each group of images has different changing factors, including fuzzy processing, rotation and scale change, perspective change, illumination change, and image compression. Figure 11 shows the experimental results of different descriptors for different impact indicator.

Fig. 10. Test images.

For all cases, the M-BRISK descriptor is better or at least comparable to the descriptors of all other tests. This is because the discrimination of the descriptor can be improved by the combination of the local features and the information between them. As can be seen from Table 1, M-BRISK runs at the same rate level as the ORB, BRISK and FREAK algorithms, and they are all much faster than the SURF. In summary, the M-BRISK algorithm can achieve higher performance with high speed, and it suitable for real-time applications.

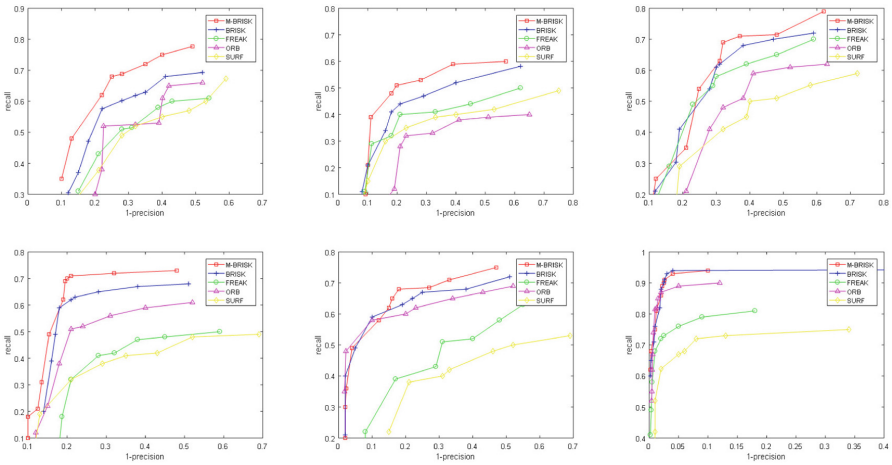


Fig. 11. Experimental results of different descriptors for each set of image pairs.

Table 1. Running time.

Methods	SURF	ORB	BRISK	FREAK	M-BRISK
Running time (ms)	0.404	0.026	0.038	0.032	0.040

7.2 Trajectory Clustering Results

We collects 1000 sets of vehicle trajectory data from 20 road sections of Hangzhou Jinqu highway for clustering algorithm test, and the trajectory datasets contains different numbers of vehicle targets, including 2 vehicles, 3 vehicles, 4 vehicles, 5 vehicles, etc. Some experimental results of clustering the trajectory set are shown in Fig. 12. It can be found that even if the vehicle has a common speed or partial occlusion, the algorithm can effectively cluster the feature point trajectories belonging to different vehicles. In addition, this paper analyzes the clustering results of the 1000 sets of data based on the number of vehicle targets, and compares them with the traditional method based on 2D trajectory methods. The results are shown in Table 2.

Table 2. Clustering precision.

Number of vehicle	2	3	4	5
3D trajectory clustering accuracy (CP)	94.75%	93.54%	89.17%	87.63%
2D trajectory clustering accuracy (CP)	90.18%	85.21%	78.32%	69.67%

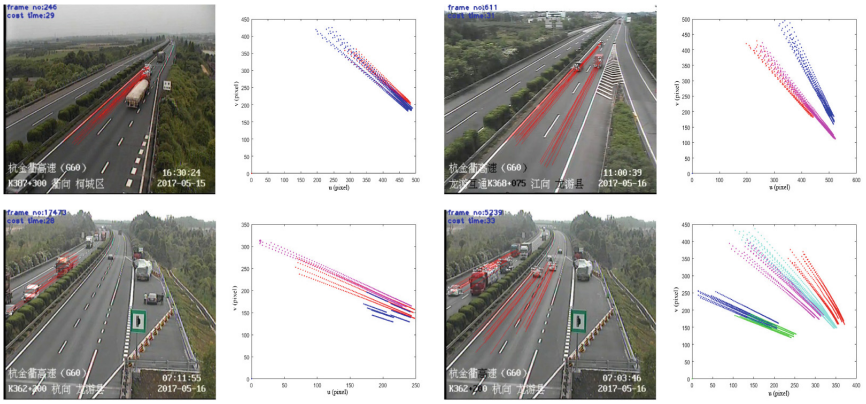
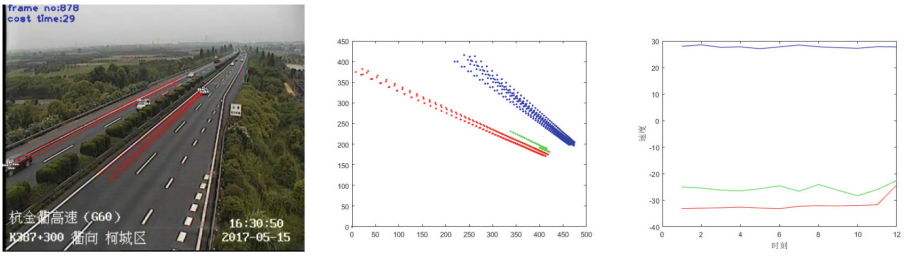


Fig. 12. Vehicle feature point trajectory clustering result.



(a) (b) (c)

Fig. 13. Vehicle real-time velocity analysis.

In order to better evaluate the clustering effect of the proposed method, we analyzed the relevant experimental results quantitatively and defined the accuracy of the clustering. The specific formula is as follows:

$$CP = \frac{1}{N} \sum_{i=1}^{N} \frac{t_i}{n_i} \times 100\% \tag{17}$$

where, N is the number of trajectory data sets with the same number of vehicles, t_i is the number of trajectory classified correctly for the i-th trajectory data set, n_i is the total number of trajectory included in the i-th trajectory data set.

7.3 Vehicle Behavior Analysis

Vehicle Individual Behavior. By reconstructing the 2D trajectory information in 3D space, the real velocity of each trajectory can be estimated, and then the real-time real velocity of the vehicle object can be estimated, so that we can draw the velocity curve of the vehicle target at each moment in order to determine whether it occurred over-speeding, low-speeding or parking events. The following is a specific experimental analysis based on specific trajectory data.

Table 3. Vehicle real-time velocity.

Objects	Real-time velocity (m/s)										
Vehicle 1	−33.2	−33.0	−32.9	−32.6	−32.9	−33.2	32.3	−32.1	−32.2	−32.0	−31.7
Vehicle 2	27.9	28.5	27.6	27.8	27.1	27.87	28.5	27.8	27.5	27.2	27.8
Vehicle 3	−28.0	−25.4	−26.2	−26.5	−25.7	−24.6	−26.6	−24.1	−26.3	−28.3	−26.1

Table 4. Direction angle of the vehicle in real time.

Objects	Direction angle (°)										
Vehicle 1	0.038	0.043	0.045	0.578	0.800	0.484	0.029	0.103	0.625	0.182	0.333
Vehicle 2	0.021	0.016	0.078	0.051	0.338	0.110	0.386	0.245	0.311	0.086	0.028
Vehicle 3	0.315	2.383	1.509	4.215	3.121	0.705	2.065	3.174	3.322	1.419	2.373

As shown in Fig. 13, Fig. 13(a) is the 2D trajectory data extracted from the vehicles of a highway section, Fig. 13(b) is the result of cluster analysis, and Fig. 13(c) is the real-time velocity curve. The partial data results of the real-time velocity at the same time are shown in Table 3, wherein the sign indicates that the running direction of the vehicles, the speed of the upstream vehicle is marked as positive, and the speed of the descending vehicle is marked as negative. Based on the estimated real-time velocity of the vehicles, it can be used as a discriminating indicator whether the vehicle has over-speed or low-speed driving. For retrograde behavior, it is necessary to observe its real-time motion vector to determine whether it has retrograde behavior, as shown in Table 4. In addition, if an abnormal parking event occurs, the velocity curve of the vehicle is as shown as Fig. 14, where its trajectory velocity will continue to approach for a period of time. For the behavior such as lane change, in addition to the direction angle, the offset in the X direction is needed to be considered. As shown in the trajectory data of Fig. 15, we can observe the driving direction angle of the real-time. As shown in Table 5, it can be found that the direction angle of the vehicle become larger and larger, and the variance of the corresponding trajectory data in the X direction is also larger than the preset. Thus, the vehicle in Fig. 15 is judged as abnormal lane change behavior.

Table 5. Direction angle of the vehicle in real time.

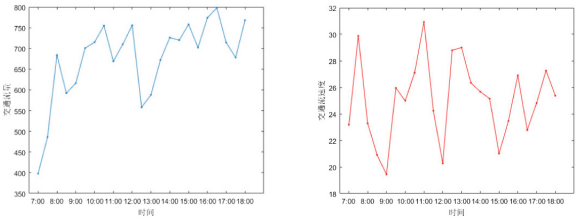
	Direction angle (°)										Variance
Trajectory 1	5.01	5.49	6.40	4.69	10.57	6.75	9.45	8.51	9.95	10.1	0.54
Trajectory 2	6.08	6.34	7.28	5.42	8.57	6.91	7.71	7.17	9.01	9.46	0.52
Trajectory 3	5.24	5.95	5.40	5.07	9.37	6.93	9.11	7.55	9.73	10.41	0.50
Trajectory 4	5.02	5.75	7.46	5.12	9.28	6.44	9.48	8.76	10.94	08.92	0.55
Trajectory 5	5.35	6.24	5.99	6.43	6.27	5.23	7.43	6.47	7.92	7.95	0.54

Traffic Flow Behavior Analysis. This section used the real-time data obtained by the proposed method to analyze the traffic flow and traffic flow speed of a highway section, which can provide the data support for the real-time traffic status. This paper took the monitoring video of Jinqu highway as the test data, and analyzed the traffic flow and traffic flow speed data of the K362 road sections at 30 min intervals from 6:30 to 18:00 on May 16, 2017. According to the obtained real-time traffic flow parameters, we drawn the real-time parameter curves and observe the time-varying rule of each traffic parameter visually, as shown in Fig. 16. It can be seen that on May 16th, the traffic volume of the K362 section of the Jinqu highway was small, and the traffic flow was large in the afternoon. Meanwhile, the traffic flow speed of the whole day is at a reasonable range. Therefore, the traffic condition of this road section is good and smooth.

(a) (b)

Fig. 14. Vehicle parking analysis.

Fig. 15. Trajectory data of lane change.



(a) (b)

Fig. 16. Traffic flow data of No. 362 section of Hangzhou Jinqu highway on May 16, 2017.

8 Conclusion

In this paper, the problems of vehicle feature point detection, trajectory extraction, rigid motion constraint, trajectory clustering and vehicle behavior analysis are studied. We constructed a mixed binary descriptor using the local gradient of the sample point position and the intensity comparison between the sample points. The algorithm has strong robustness in the face of image blur, rotation, scale, viewing angle and illumination changes, and can meet the needs of practical applications in real-time. In order to better solve the segmentation problem of moving vehicles in complex traffic scenes, this paper proposed a method based on rigid motion constraints for vehicle 3D trajectory feature analysis, and constructed a new similarity measure between trajectory sets. It is applied to the framework of the spectral clustering algorithm to realize trajectory clustering in 3D space. In addition, this paper uses the obtained 3D information of the vehicle trajectory and its clustering results to analyze the vehicle behavior in the specific traffic scenes.

Acknowledgement. This work was supported by National Natural Science Foundation of China under Grants 61801414, Natural Science Foundation of Shandong Province under Grants ZR2017QF006, the Major Science and Technology Innovation Projects in Shandong Province 2019JZZY020131, the China Postdoctoral Science Foundation under Grant 2019T120732.

References

1. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: fast retina keypoint. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
2. Barth, A., Franke, U.: Tracking oncoming and turning vehicles at intersections. In: International IEEE Conference on Intelligent Transportation Systems (2010)
3. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_32
4. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_56
5. Chang, C., Ansari, R.: Kernel particle filter for visual tracking. IEEE Sig. Process. Lett. **12**(3), 242–245 (2005)
6. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: IEEE Conference on Computer Vision and Pattern Recognition CVPR (2002)
7. Dai, Z., et al.: Video-based vehicle counting framework. IEEE Access **7**, 64460–64470 (2019)
8. Dan, B., Sclaroff, S., Kollios, G.: Extraction and clustering of motion trajectories in video. In: International Conference on Pattern Recognition (2004)
9. Fang, Y., Wu, J., Huang, B.: 2D sparse signal recovery via 2D orthogonal matching pursuit. Sci. China Inf. Sci. **55**(4), 889–897 (2012)

10. Ali, N.H., Hassan, G.M.: Kalman filter tracking. Int. J. Comput. Appl. **89**(9), 15–18 (2014)
11. Jeon, G., Anisetti, M., Lee, J., Bellandi, V., Jeong, J.: Concept of linguistic variable-based fuzzy ensemble approach: application to interlaced HDTV sequences. IEEE Trans. Fuzzy Syst. **17**(6), 1245–1258 (2009)
12. Jeon, G., Anisetti, M., Wang, L., Damiani, E.: Locally estimated heterogeneity property and its fuzzy filter application for deinterlacing. Inf. Sci. Int. J. **354**(C), 112–130 (2016)
13. Johnson, N., Hogg, D.: Learning the distribution of object trajectories for event recognition. Image Vis. Comput. **14**(8), 609–615 (1996)
14. Kanhere, N.K., Birchfield, S.T.: A taxonomy and analysis of camera calibration methods for traffic monitoring applications. IEEE Trans. Intell. Transp. Syst. **11**(2), 441–452 (2010)
15. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: binary robust invariant scalable keypoints. In: International Conference on Computer Vision (2011)
16. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
17. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2003)
18. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**, 1615–1630 (2007)
19. Ojala, T., Harwood, I.: A comparative study of texture measures with classification based on feature distributions. Pattern Recogn. **29**(1), 51–59 (1996)
20. Prasad, M., Fitzgibbon, A.W.: Single view reconstruction of curved surfaces. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2006)
21. Rosin, P.L.: Measuring Corner Properties. Elsevier (1999)
22. Rosten, E., Porter, R., Drummond, T.: Faster and better: a machine learning approach to corner detection. IEEE Trans. Pattern Anal. Mach. Intell. **32**(1), 105–119 (2008)
23. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_34
24. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: an efficient alternative to SIFT or SURF. In: 2011 International Conference on Computer Vision (2011)
25. Sivaraman, S., Trivedi, M.M.: Combining monocular and stereo-vision for real-time vehicle ranging and tracking on multilane highways. In: International IEEE Conference on Intelligent Transportation Systems (2011)
26. Sumpter, N., Bulpitt, A.: Learning spatio-temporal patterns for predicting object behaviour. Image Vis. Comput. **18**(9), 697–704 (2000)
27. Tola, E., Lepetit, V., Fua, P.: Daisy: an efficient dense descriptor applied to wide-baseline stereo. IEEE Trans. Pami **32**(5), 815–30 (2010)
28. Wang, X., Song, H., Fang, Y., Cui, H.: Novel discriminative method for illegal parking and abandoned objects. J. Adv. Comput. Intell. Intell. Inform. **22**, 907–914 (2018)
29. Wang, X., Song, H., Guan, Q., Cui, H., Zhang, Z., Liu, H.: Vehicle motion segmentation using rigid motion constraints in traffic video. Sustain. Cities Soc. **42**, 547–557 (2018)

30. Wang, Y., Papageorgiou, M.: Real-time freeway traffic state estimation based on extended Kalman filter: a general approach. Transp. Res. Part B **39**(2), 141–167 (2007)
31. Weiming, H., Dan, X., Tieniu, T., Steve, M.: Learning activity patterns using fuzzy self-organizing neural network. IEEE Trans. Syst. Man Cybern. Part B Cybern. Publ. IEEE Syst. Man Cybern. Soc. **34**(3), 1618 (2004)
32. Yan, K., Sukthankar, R.: PCA-SIFT: a more distinctive representation for local image descriptors. In: Proceedings of CVPR, vol. 2, no. 2, pp. 506–513 (2004)