

Research Article

An Effective Algorithm for Video-Based Parking and Drop Event Detection

Gang Li , Huansheng Song , and Zheng Liao

School of Information Engineering, Chang'an University, Xi'an 710064, Shaanxi, China

Correspondence should be addressed to Gang Li; 1723915372@qq.com

Received 13 November 2018; Revised 11 February 2019; Accepted 18 March 2019; Published 7 April 2019

Guest Editor: Ke Deng

Copyright © 2019 Gang Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Real-time and accurate detection of parking and dropping events on the road is important for the avoidance of traffic accidents. The existing algorithms for detection require accurate modeling of the background, and most of them use the characteristics of two-dimensional images such as area to distinguish the type of the target. However, these algorithms significantly depend on the background and are lack of accuracy on the type of distinction. Therefore, this paper proposes an algorithm for detecting parking and dropping objects that uses real three-dimensional information to distinguish the type of target. Firstly, an abnormal region is initially defined based on status change, when there is an object that did not exist before in the traffic scene. Secondly, the preliminary determination of the abnormal area is bidirectionally tracked to determine the area of parking and dropping objects, and the eight-neighbor seed filling algorithm is used to segment the parking and the dropping object area. Finally, a three-view recognition method based on inverse projection is proposed to distinguish the parking and dropping objects. The method is based on the matching of the three-dimensional structure of the vehicle body. In addition, the three-dimensional wireframe of the vehicle extracted by the back-projection can be used to match the structural model of the vehicle, and the vehicle model can be further identified. The 3D wireframe of the established vehicle is efficient and can meet the needs of real-time applications. And, based on experimental data collected in tunnels, highways, urban expressways, and rural roads, the proposed algorithm is verified. The results show that the algorithm can effectively detect the parking and dropping objects within different environment, with low miss and false detection rate.

1. Introduction

With the increasing demands of traffic transportation in modern life, such as express delivery and logistics, the number of motor vehicles in the city continues to rise. The increase in the number of motor vehicles has caused numerous problems, such as parking and dropping incidents that reduce road traffic efficiency [1]. Therefore, accurately detecting the parking and dropping events on the road in real time is a key factor to ensure a safety-of-life traffic system [2].

Parking and throwing objects are static targets in traffic scenes. The detection algorithms for such targets in intelligent-traffic-incident-detection systems developed at home and abroad are mainly divided into two steps: target area detection and target type differentiation.

There are two methods for target area detection: tracking method and nontracking method.

The tracking method detects the stationary target by analyzing the characteristics of the foreground target trajectory. For example, Bevilacqua et al. [3] firstly obtain the foreground target by background difference; secondly, use the optical flow method to track the target; finally, analyze the displacement of the target center position. If the displacement of the target center position has been moving within a small range for a certain period of time, it is considered that a parking event has occurred. This method is simple to implement but there is a problem if the parking is detected during the parking period. Bing-Fei Wu et al. [4] proposed a tunnel event detection system. First, the background extraction is performed; then the foreground target is obtained by subtracting the background; finally, the target is tracked. Considering that the movement of the parked vehicle is small, the average distance of the vehicle is calculated using the trajectory line to determine the parking

event. Guler et al. [5] used the object tracker and the scene description layer (background) to detect the stopped vehicle. The basic idea of the algorithm is to use the object tracker to track the target in the scene. When a pixel is detected as a still pixel, it is compared with the scene description layer. When the two are similar, the pixel is a static target. The likelihood will decrease and vice versa. Although the method uses the samples in the i-LIDS for testing, the accuracy can be met, including the parking lot scene and the legacy package scene. But there are still two shortcomings: on the one hand, the rapidly changing background is considered a parking event. On the other hand, when the scene description layer itself contains a static vehicle, the area where the vehicle leaves is mistakenly detected as the area where the parking event occurs. In a traffic scene with a large traffic flow, it is difficult to extract a background image that does not contain any vehicle. Zhang Beibei et al. [6] used the particle filter algorithm combined with the OSTU threshold to detect the stationary target. The biggest problem of this method is the selection of the threshold and the false positive caused by the void after segmentation. Akhawaji R et al. [7] used a mixed Gaussian model to model the background and use Kalman filtering to detect stationary targets in the forbidden area. The algorithm is sensitive to illumination and is easy to lose targets when traffic is heavy. He T et al. [8] used special GPS points for map matching and track indexing, then simulates normal trajectories, extracts features, and uses distributed testing to detect illegal parking.

The nontracking method mainly relies on background modeling and analysis of foreground pixel time series features to detect stationary target regions. For example, Fatih Porikli et al. [9] used the double background method to detect the parking litter. This method does not use any tracking technology to detect abnormal events only by subtracting the background. The basic idea is as follows: Firstly, two mixed time Gaussian models are used to establish two backgrounds with different time constants (short background and long background), and the online Bayesian update mechanism is used to update the two backgrounds in real time. Among them, the short background describes the most recent target from motion to still, and the long background describes the real background of the scene. When there are foreground pixels in the scene, compare them with the two backgrounds, respectively. When the pixel is very similar to the short background and has a large difference from the long background, the pixel is considered to be a static target pixel. If a pixel is continuous it has been a static target pixel for a period of time, and the pixel is marked as an abnormal pixel. This method has a certain real-time performance, but the anti-interference is poor, and the time constant when establishing a long background and a short background is difficult to determine. Stauffer et al. proposed a parking detection algorithm that uses the difference method to achieve target extraction. The algorithm has strong real-time performance. However, its biggest shortcoming is that when the target type is identified, the interference caused by other factors (such as pedestrians, bicycles, etc.) is not filtered out, resulting in an increase in the false positive rate. Zhao Min et al. [10] showed that, in order to avoid the false positive rate which is too high, firstly,

the hybrid Gaussian model is used to obtain the suspected moving foreground target in the background extraction and update; then, the steady state change of the foreground target is analyzed to detect the stationary target. There are two major shortcomings of this method. First, the amount of calculation is large. Second, when the training time is too short and the background model is not fully trained, the foreground target is estimated as the background, which affects the detection of static targets.

Static targets in traffic scenes are mainly divided into parking and throwing objects, so the distinction between static targets is the distinction between parking and throwing objects. The current algorithm mainly uses the two-dimensional features of the target to identify it. For example, Wang Dianhai and Hu Hongyu [11] identify the difference between the vehicle target and the target area of the target in the foreground. However, due to the angle of the camera, etc., the target of the throwing object detected in the close scene is not much different from the target area of the vehicle detected in the distant view. Based on the detection of the target area, Mu Chunyang [12] used Hough ellipse fitting, wheel circularity and compact feature extraction to identify the parking events.

Today's machine learning [13] applications on images do have very good performance, but the traditional methods we use still have advantages: (1) Today, when machine learning is prevalent, it does not mean abandoning traditional methods. In-depth study of traditional methods is something we have been doing and is valuable. (2) In the future, we hope to continue to develop in the direction of embeddedness. The expensive cost of machine learning limits the popularity of the methods we propose in the paper. Therefore, we have chosen a method based on three-dimensional information to classify vehicles.

Through the survey of research on video-based parking and dropping objects, the key issues found in most algorithms for detecting parked and discarded objects are mainly in two aspects. The first issue is how to detect the target, which is the core part of the algorithm. Tracking and nontracking methods are generally used to detect the target area. Both methods need to extract and update the background. However, under complicated traffic scenes such as low visibility, large traffic flow, and intense lighting changes, it is difficult to extract an ideal background image. The second one is how to distinguish the target type. When distinguishing the target type, the two-dimensional feature of the target is often used, but the imaging process of the camera is a process of dimensional reduction. In this process, the target will undergo significant scale changes and geometric deformation. Therefore, these methods of using image features to identify targets have significant limitations. In view of these two shortcomings, it is of great theoretical and practical value to study algorithms for detecting parking and dropping objects that do not rely on background and use real three-dimensional information for target recognition.

In this paper, the above problems are studied and a new method is proposed. The real-time video collected by the camera is used as the data source. The image analysis and processing program is used to realize the automatic detection

and feedback of parking and litter events. It is mainly divided into the following three steps. Firstly, based on the state evolution, the initial determination of the abnormal region in the image is carried out. Secondly, the two-way tracking and eight-neighbor seed filling algorithm is used to segment the parking and the drop area in the image. Finally, the three-dimensional information is used to distinguish the target.

2. Preliminary Determination of Abnormal Regions in Images

The detection of abnormal regions is the core of the whole algorithm, so choosing an appropriate detective algorithm is the first problem to consider. Current algorithms for detection are too dependent on the background and computationally intensive. We will use status change to detect abnormal regions, which can effectively avoid the above two shortcomings.

The abnormal area refers to the image area where the steady state changes. The basic idea of the algorithm is as follows: Firstly, the image is preprocessed to highlight useful information, which lays a foundation for improving the accuracy of subsequent detection. Secondly, the detection of abnormal regions is carried out based on the status change, and solutions are given for some of the shortcomings. Finally, an improved algorithm for detection is proposed, and the results before and after the improvement are compared and analyzed.

2.1. Image Preprocessing. When the video image is captured, the camera will be affected by factors such as illumination changes and noise pollution, which will affect the captured image. In order not to affect the result of the algorithm, the image will be preprocessed by image enhancement, edge extraction, and median filtering.

2.1.1. Image Enhancement. In the case of night or smog, the contrast and color of the collected traffic video images will be degraded, and a lot of useful information will be covered, which is very unfavorable for the subsequent algorithm. Among the algorithms for image enhancement, the gray-level section transforms based on gray level transformation which has been widely used due to its advantages such as simplicity and diverse transformation functions [14–16]. In this paper, the original image is processed by a three-stage linear transformation, as shown in Figure 1. The expression as follows:

$$g(x, y) = \begin{cases} c & 0 \leq f(x, y) \leq a \\ \frac{d-c}{b-a} [f(x, y) - a] + c & a \leq f(x, y) \leq b \\ d & b \leq f(x, y) \leq M \end{cases} \quad (1)$$

In Figure 1, $f(x, y)$ represents the original gray value at (x, y) , $g(x, y)$ represents the gray value at (x, y) after image enhancement, and M is the maximum gray value of 255. And Figure 2 shows the original image and the enhanced image.

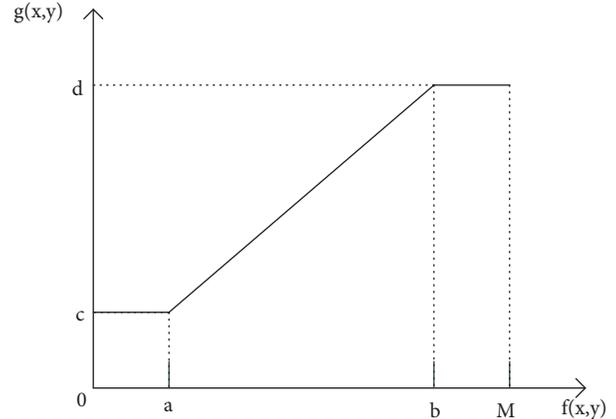


FIGURE 1: Linear transformation of gray scale.

Comparing the two images, it can be seen that the contrast of the enhanced image is significantly improved.

2.1.2. Edge Extraction. The edge is the most basic feature of the image, which is invariant to changes in light, so in order to reduce the effect of light on the detection result, edge extraction is applied after image enhancement. There are many classic operators for edge detection [17], after considering the detection effect, real-time performance, and arithmetic speed. We employ a simplified first-order differential operator that uses local differences to find the edges of the image. Its expression is as follows:

$$E(x, y) = \max(|f(x, y) - f(x, y + 1)|, |f(x, y) - f(x + 1, y)|) \quad (2)$$

$f(x, y)$ represents the original gray picture and $E(x, y)$ represents the edge enhancement gray picture. The experimental results of edge enhancement of the video image of Xi'an South Second Ring by the operator are shown in Figure 4(c).

2.1.3. Median Filtering. In the process of acquisition, transmission, and storage, any process may introduce noise, and the presence of noise seriously affects the result of edge enhancement. Therefore, it is necessary to denoise the detected edge image. The median filter [18] also filters out noise while maintaining the edges of the image. The basic principle is as follows: select a symmetrical area centering on each pixel in the image, and sort all the pixel values in the area, taking the middle pixel value as the pixel value of the current point. Figure 3 is a result of the horizontal straight type window filtering of Figure 4(c), from which it can be found that the image quality is significantly improved, and the edge of the image is well preserved while filtering noise.

2.2. Abnormal Region Determination Based on Status Change. Usually the gray value of each pixel in the image does not change for a long time. Only when the foreground target passes from the pixel area, the gray value of the point will be changed, and when the foreground target passes, the change of the pixel gray value is greater than the change caused by the

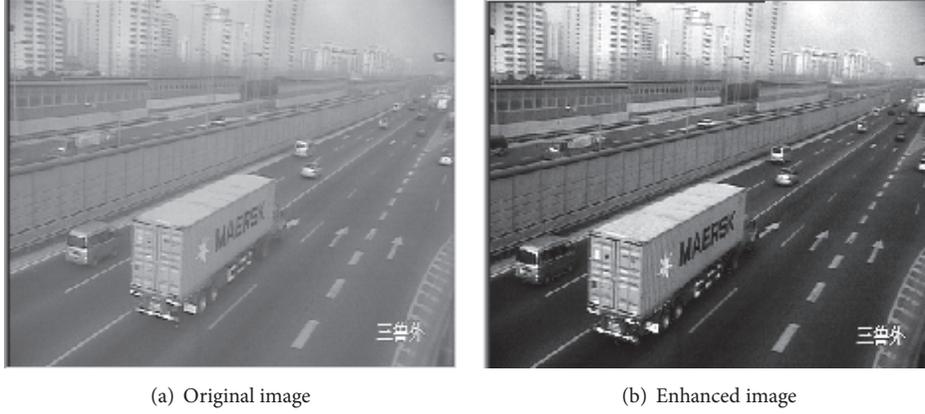


FIGURE 2: Original image and enhanced image.



FIGURE 3: Median filter effect.

environmental influence. Thus, when it is detected that there is a pixel point in the image where the gray value changes greatly, which means that the status change happen, it is determined that the foreground object exists. The detected change in the gray value may be caused by the moving target passing through the detected area or the target entering the detected area and stopping. Therefore, in order to correctly detect the parking and the dropping object, it is necessary to remove the gray value caused by the moving target passing through the detected area.

When the pixel gray value suddenly changes and returns to the initial gray value in a short time, it indicates that the moving target passes through the area, but does not stop. And when the pixel gray value suddenly changes and remains stable for a while, indicating that the pixel point is occupied by the foreground target, it is highly probable that a parking or dropping event occurs. Since a single pixel contains too little information and does not take into account the influence of surrounding pixels, the block is used as the basic processing unit.

Figure 5 shows the texture's change of a block in two situations, where the moving object passes through the detected area and the moving object enters the detected area and stops.

As can be seen from Figure 5, when the moving object passes through the detected area, the gray values of the two stable states are not much different. However, when the moving object stops in the detected area, the gray values of the two stable states are greatly different. In order to get rid of the dependence on the background, preliminary determination is made on the abnormal region in the image after

comparing the difference between the two stable states of the block.

2.2.1. Basic Concept

(1) *Image Segmentation*. The video image has a dimension of 720×288 and is divided into blocks of dimension 8×6 , so the image is divided into 90×48 blocks. Block's coordinates are shown in Figure 6.

(2) *SAD Value (Sum of Absolute Difference)*. SAD represents the sum of the absolute values of the pixel gray differences at corresponding positions between blocks. In this paper, SAD is mainly used for two aspects: one is the similarity matching between the current frame and the template frame; the other is the similarity matching between the current state and the reference state. Therefore, the SAD calculation formula of any block at time t is as follows:

$$SADT_t(m, n) = \sum_{y=0}^{h-1} \sum_{x=0}^{w-1} |E_t(x, y) - T_t(x, y)| \quad (3)$$

$$SADS_t(m, n) = \sum_{y=0}^{h-1} \sum_{x=0}^{w-1} |SR_t(x, y) - SV_t(x, y)| \quad (4)$$

$SADT_t(m, n)$ represents the SAD value between the current frame and the template frame at block coordinates (m, n) . $SADS_t(m, n)$ represents the SAD value between the current state and the reference state at block coordinates (m, n) . $E_t(x, y)$ and $T_t(x, y)$ represent the gray values of the pixel coordinates (x, y) in the current frame and the template frame. $SR_t(x, y)$ and $SV_t(x, y)$ represent gray values of pixel coordinates (x, y) in the reference state and the current state. $x=0, 1, \dots, w-1$. $y=0, 1, \dots, h-1$. $m=0, 1, \dots, C-1$. $n=0, 1, \dots, R$.

(3) *Definition of Exception Block*. The detected area in this article is the entire lane in the image. For any small block in the detected area, there only are two states: a road state and an abnormal state; the latter one represents parking or dropping. If a small piece maintains a state for a while, the small piece is considered to be in a steady state. The abnormal block in the

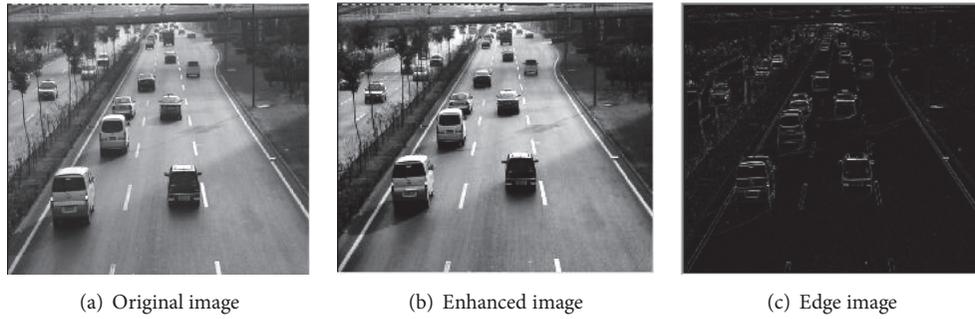


FIGURE 4: Image preprocessing.

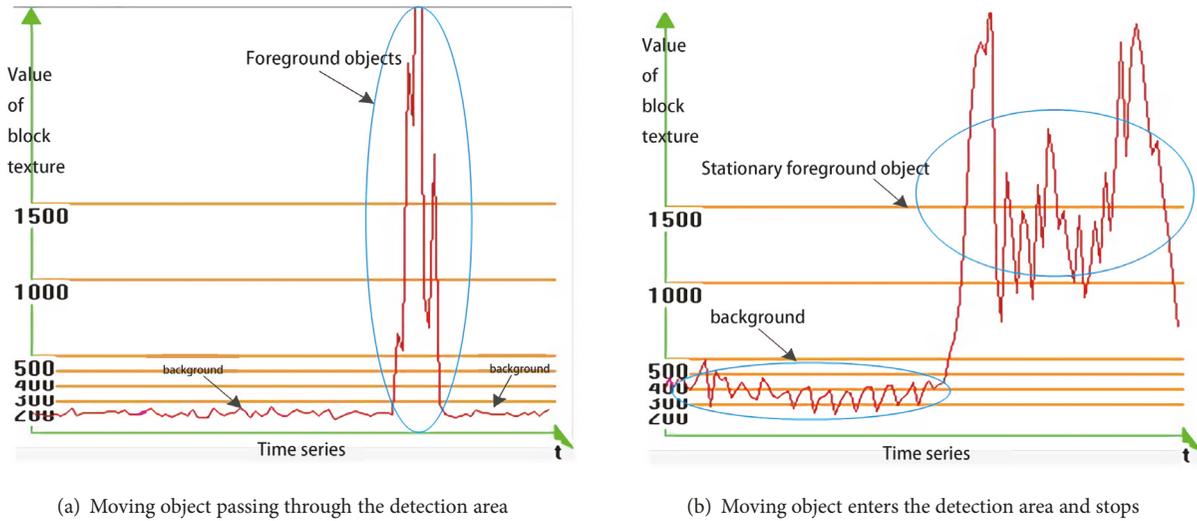


FIGURE 5: Texture changes of blocks in the detection area.



FIGURE 6: Block coordinate.

text refers to an image block that changes from a road stable state to an abnormal stable state.

2.2.2. Algorithm Description and Experimental Results. Based on the basic concepts, this section will introduce the core of the algorithm. According to the above method of dividing

the image, a counter $C(m,n)$ with an initial value of 0 and an abnormality flag $D1(m,n)$ and $D2(m,n)$ with an initial value of false are set for each block in the image, the (m,n) is the block coordinates. The abnormality flag $D1(m,n)$ is used to mark whether the block meet the state change, and $D2(m,n)$ is used to mark whether the block meet the forward trajectory but has no backward trajectory after bidirectional tracking of the block. First, the first frame image in the video is assigned to the template frame, and then starting from the second frame image, each image block is detected according to the following steps.

Firstly, calculate $SADT_t(m,n)$ of the block in the current frame and the corresponding block in the template frame according to formula (3). If $SADT_t(m,n) < Th_{sad}$, then the data of the current frame of the block matches the template frame's data, and the counter $C(m,n)$ is incremented by 1. Otherwise the counter reset and the data of the corresponding block in the template are updated with the current frame's data of the block.

Secondly, when the value of the counter $C(m,n)$ reaches the threshold Th_a , the gray value of the block in the current frame is saved. And if the threshold Th_a is reached for the first time, the gray value is saved to $R_t(m,n)$; otherwise it is saved to $V_t(m,n)$.

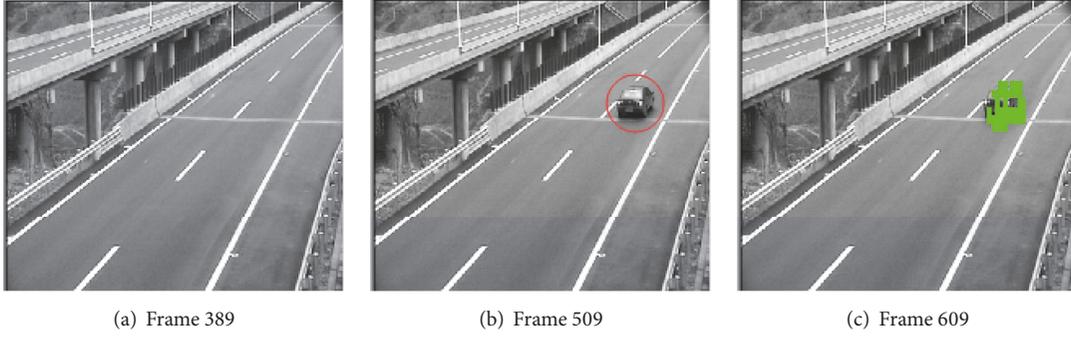


FIGURE 7: Test results of parking on Chongqing Expressway.

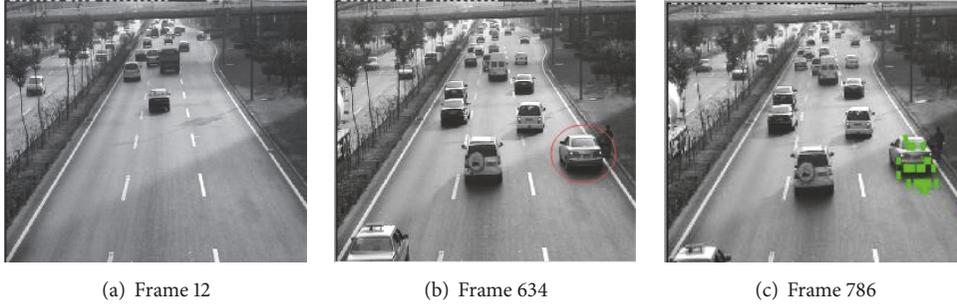


FIGURE 8: Xi'an South Second Ring Road parking test results.

Thirdly, when the value of the counter $C(m,n)$ reaches the threshold Th_b ($Th_b > Th_a$), save the gray value of the block in the current frame to $V_t(m,n)$, and calculate $SADS_t(m,n)$ according to formula (4). If $SADS_t(m,n) < Th_c$, it means that the status has not changed. Otherwise, set $DI(m,n)$ to True, indicating that the block is an exception block. The reference state is updated with the current state regardless of whether the state changes or not; that is, $R_t(m,n) = V_t(m,n)$.

In order to verify the effectiveness of the above algorithm, the experiment was carried out in four scenarios: Chongqing Expressway, Xi'an South Second Ring Road, Xi'an South Second Ring Bus Lane, and Shanghai Fuxing Tunnel. The experimental results are as follows.

In Figures 7–11, (a) shows that no parking or dropping event, (b) the red circle in the figure indicates an abnormal event, and (c) the green image block in the figure indicates the detected abnormal region.

As can be seen from Figures 7–11, the algorithm can effectively detect abnormal blocks in the image. But through Figures 12–14, it is found that the algorithm is too sensitive to illumination, image noise, etc.

2.2.3. Existing Problems and Solutions. In order to eliminate false alarm, the algorithm are analyzed and found to have the following defects.

(1) *Moving Target Texture.* Some false alarms due to image noise contain less edge information. For example, in the edge-enhanced graph of the block (6, 30) in Figure 15, the gray scale of the block is enlarged to find that the texture change is smooth, and the gray scale is mainly concentrated between 0

and 13, and the total gray value of the block is low. The edge information of moving objects such as vehicles and dropping objects is more obvious. The gray-scale distribution of the block (12, 51) in Figure 16 is more dispersed, and the total gray value is significantly higher than the block (6, 30) shown in Figure 15(a).

Therefore, it can be judged whether it is an abnormal block according to the total gray value in the image block, and the calculation formula is as formula (5).

(2) *Impact of Nearby Vehicles on Test Results*

$$GS(m, n) = \sum_{y=0}^{h-1} \sum_{x=0}^{w-1} E(x, y) \quad (5)$$

By conducting experiments on road sections with different traffic densities, it can be found that the time required to detect anomalies on road sections with less traffic density (number of image frames) is shorter than high traffic density. As shown in Figures 7–11, it takes about 100 frames to detect anomalies in a high-speed scene in Chongqing. And it takes about 150 frames to detect anomalies in the scene of the South Second Ring Road in Xi'an.

Figure 17(a) shows the real-time video image of the South Second Ring Road in Xi'an, where the traffic density is large. Taking the red block (67, 16) as an example, Figure 17(b) shows the curve of ST and counter C during the t time. And the two curves show that there are 8 cars passing by in the period and interfering with the results of the test. This makes the counter unable to reach the threshold λ , which means that the steady state cannot be reached.

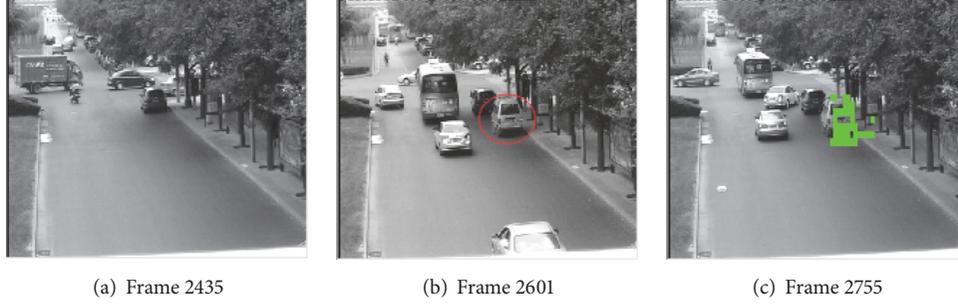


FIGURE 9: Xi'an South Second Ring Busway parking detection results.

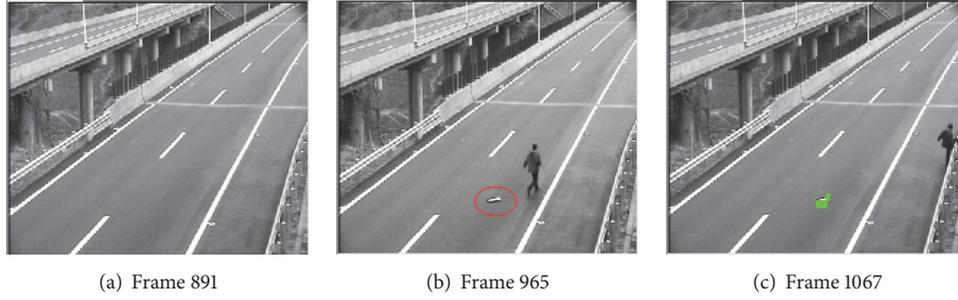


FIGURE 10: Detection results of the droppings on Chongqing Expressway.

From Figure 17(b), it can be found that the counter C of the block $(67, 16)$ reaches the higher value δ three times during the t period, as shown at three points A, B, and C. And during these three periods, the counter accumulation is due to the fact that the vehicle matches the template, so it is possible to consider connecting the counters so that the block reaches a steady state more quickly. The state in which the counter reaches δ is referred to as the potential steady state. Therefore, the difference between adjacent potential stable states in the time series can be considered, and if the difference is small, the counter is accumulated.

(3) *The Effect of Slow Changes in Light on Experimental Results.* The abnormal block in Figures 12(c) and 13(c) is a false alarm due to a slow change in illumination. Figure 18 is a graph showing the ST and counter C curves of a false alarm block (12, 29) in Figure 13(c).

It can be seen from Figure 18(b) that the ST of the block changes smoothly; even at the point A (red ellipse), the ST value when the counter is cleared due to exceeding the threshold Th_{sad} is not much different from the previous ST value. As can be seen from Figure 19(b), the ST will suddenly change when the counter is cleared caused by the vehicle passing by. In this case, consider the relationship between the ST change of the E frame before the counter is cleared (in view of the calculation amount, E generally takes 200) and the current ST, and determine whether it is a false alarm caused by illumination.

In this paper, the variance of historical ST values is used to measure the change. The specific calculation formula is as follows:

$$\mu(m, n) = \frac{1}{E} \sum_{t=0}^E \text{SADT}_t(m, n) \quad (6)$$

$$\sigma(m, n) = \frac{1}{E} \sum_{t=0}^E |\mu(m, n) - \text{SADT}_t(m, n)| \quad (7)$$

$\mu(m, n)$ represents the mean of the SADT of the block (m, n) before the E frame and $\delta(m, n)$ represents the variance of the SADT. If the current $\text{SADT}(m, n)$ of the block (m, n) exceeds the threshold Th_{sad} , the change is slower than the $\text{SADT}(m, n)$ of the previous E frame, and the $\text{SADT}_t(m, n)$, $\mu(m, n)$, and $\delta(m, n)$ of the block (m, n) meet the formula (8). The false alarm is considered to be caused by a slow change in illumination, and the counter is directly cleared. Otherwise, it is caused by the passing of the vehicle, and the counter is accumulated.

$$|\text{SADT}_t(m, n) - \mu(m, n)| \leq \alpha * \delta(m, n) + \text{Base} \quad (8)$$

$\alpha > 0$ in formula (8), the value of which changes according to the scene, Base is the base value, generally $\alpha = 3$, and $\text{Base} = 20$.

2.2.4. Improved Algorithm and Experimental Results. Based on the algorithm proposed in Section 2.2.2, this section comprehensively considers the edge information and the change of the history ST to improve the algorithm. $\text{StateFlag1}(m, n)$ and $\text{StateFlag2}(m, n)$ whose initial values are both false indicate whether the block has a potential stable state and a stable state. Use $\text{PotentialState}(m, n)$ to save the potential stability of the block. The specific steps are as follows.

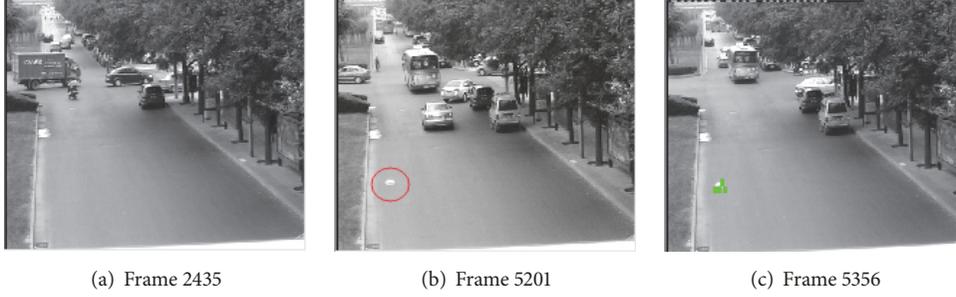


FIGURE 11: Results of detecting the dropping in the bus lane of South Second Ring Road in Xi'an.

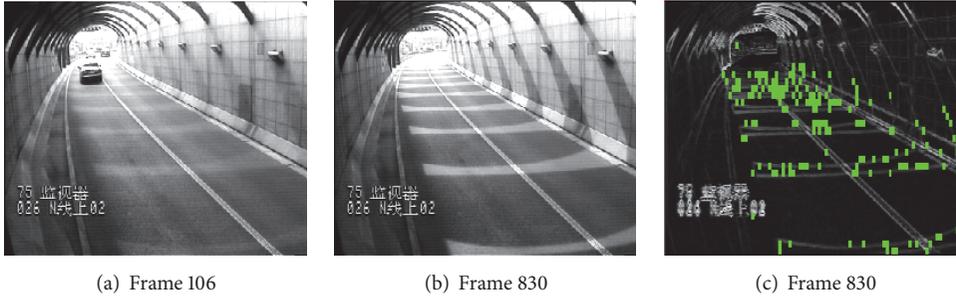


FIGURE 12: Shanghai Fuxing Road Tunnel Exit.

The first frame in the video is assigned to the template frame, then from the second frame, each image block is detected as follows.

Step 1. Calculate $SADT_t(m, n)$ of the block in the current frame and the corresponding block in the template frame according to formula (3), and record the $SADT_t(m, n)$.

Step 2. If $SADT_t(m, n) < Th_{sad}$, then the current frame data of the block matches the template frame data, and the counter $C(m, n)$ is incremented by one. Otherwise, the mean and variance of the historical $SADT$ of the block are calculated by equations (6) and (7). If the formula (8) can be met, the counter is cleared and the data of the corresponding block in the template is updated with the current frame data of the block. If the counter reaches the threshold δ ($\delta < Th_a$) and $StateFlag1(m, n) = False$, the gray value of the current block is saved to the potential stable state $PotentialState(m, n)$. After the value of the counter $C(m, n)$ is saved to $PeekC(m, n)$, the counter is cleared, and $StateFlag1(m, n)$ is set to True. If the counter $C(m, n)$ reaches the threshold δ and $StateFlag1(m, n) = True$, the SAD between the current gray level of the block and the potential steady state $PotentialState(m, n)$ is calculated. If $SAD < Th_{sad}$, connect the counter, the counter peak value is assigned to the current counter, $C(m, n) = PeekC(m, n)$.

Step 3. If the value of the counter $C(m, n)$ reaches the given threshold Th_a and $StateFlag2(m, n) = False$, it means that the counter reaches the threshold Th_a for the first time, then the gray value is stored in $SR_t(m, n)$ and $StateFlag2(m, n)$ is set to True. Otherwise it is saved in $SV_t(m, n)$.

Step 4. When the value of the counter $C(m, n)$ reaches a given threshold Th_b ($Th_b > Th_a$), $SADT_t(m, n)$ is calculated according to (4). If $SADT_t(m, n) < Th_c$, this indicates that the steady state has not changed; otherwise go to Step 5. The reference state is updated with the current state regardless of whether the state changes or not, that is, $SR_t(m, n) = SV_t(m, n)$.

Step 5. Calculate the sum of the pixel values of the current block using (5). If $GS(m, n) < Th_d$, the block's exception flag $DI(m, n)$ is set to false; otherwise it is set to True, indicating that the block is an exception block.

In order to verify the improved algorithm, the results are compared. As shown in Figures 19, 20, and 21, the improved algorithm has significantly reduced false alarm and is more adaptable to complex environments. As can be seen from Figure 22, the original algorithm can detect the abnormality at the 786th frame, and the improved algorithm can detect the abnormality at the 755th frame. Therefore, the improved algorithm has a significant improvement in detection speed.

3. Segmentation of Parking and Dropping Areas in the Image

The anomalous area detected in the second section inevitably includes the effects caused by illumination changes, noise, etc. Therefore, this section first performs bidirectional tracking on the selected abnormal area, which is determined to be caused by a parking or a parachute event. Then, the eight-neighbor seed filling algorithm is used to analyze the final abnormal region to segment the parking and the dropping area.

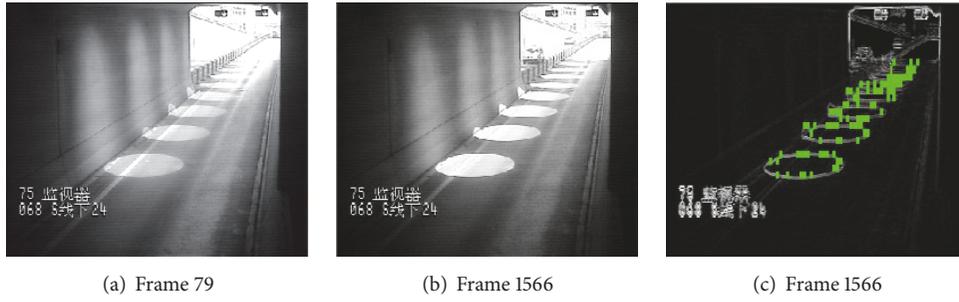


FIGURE 13: Shanghai Fuxing Road Tunnel Entrance.

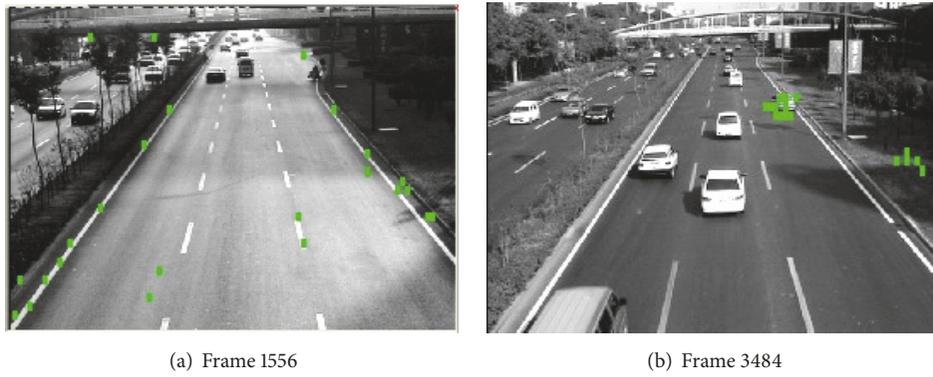


FIGURE 14: Xi'an South Second Ring Road.

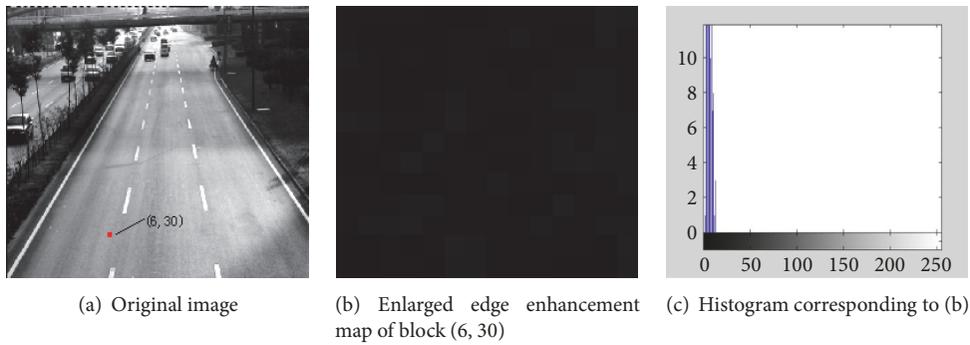


FIGURE 15: Texture analysis of the false alarm block.

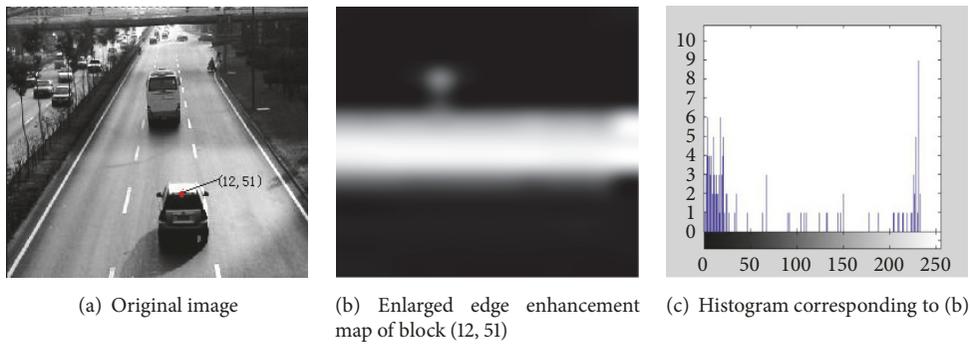


FIGURE 16: Texture analysis of anomalous blocks.

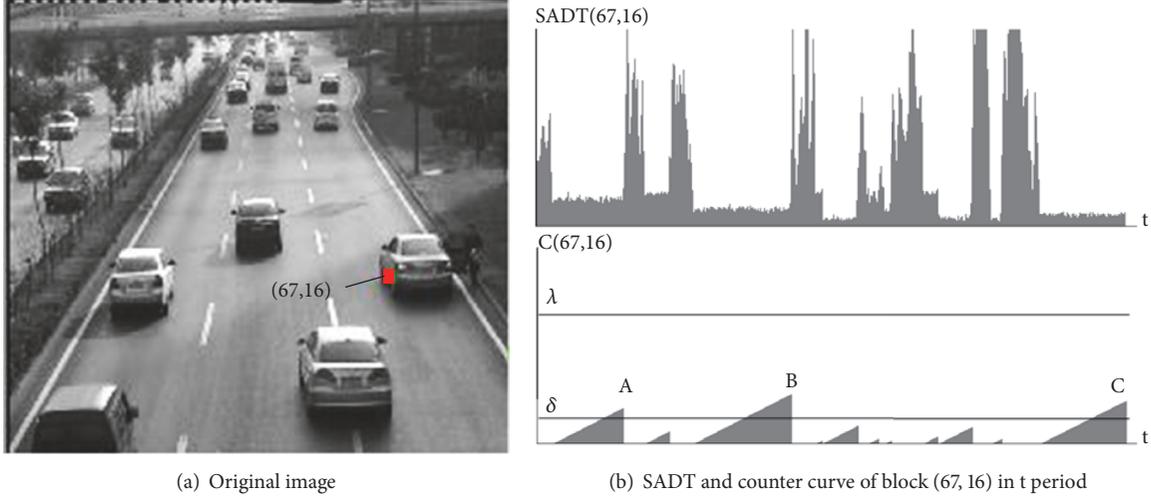


FIGURE 17: Effect of passing vehicles on detection.

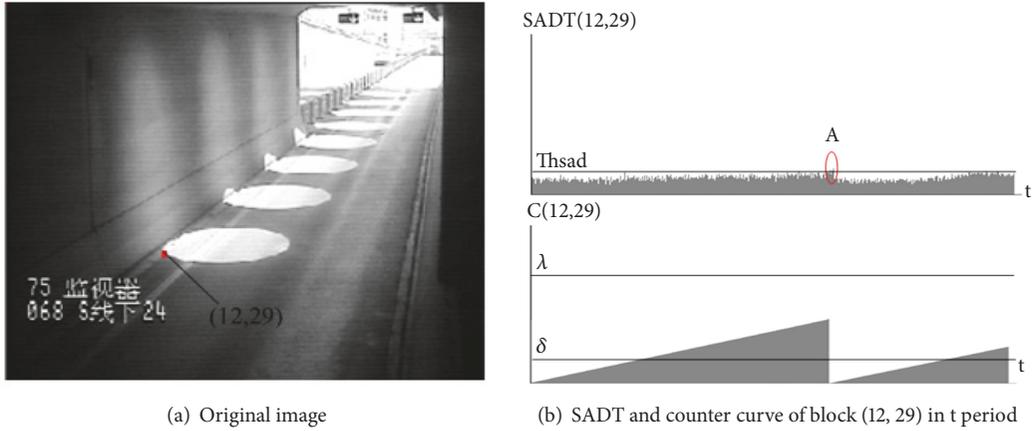


FIGURE 18: Effect of illumination on detection

3.1. *Two-Way Tracking to Determine Parking and Dropping Areas.* Parking and dropping objects are from the state of motion to the state of rest and have the characteristics of a forward trajectory without a backward trajectory. However, the false alarm caused by the shadow does not have such a feature, so the two-way tracking of the detected abnormal region can further reduce false alarm and improve accuracy.

3.1.1. *Corner Extraction and Corner Matching.* The pixel points in the image where the brightness changes drastically and the pixel points of the maximum value of the curvature on the edge curve of the image are called corner points, which contain many pieces of important information [19]. In this paper, the *Moravec* [20] corner extraction algorithm for obtaining corner by calculating the gray-scale variance is used. This method does not depend on other local features of the object and has fast speed and real-time performance.

Figure 23 shows the results of detecting corners.

After acquiring the corner point of the abnormal block, in order to track the abnormal block, it is necessary to find the position of the corner point in the next time series

to match the corner points. The matching process is the process of finding the location with the greatest similarity to a given template in the search area. Commonly used matching methods [21] include block matching method, pixel recursive method, phase correlation method, and spatial feature method. Compared with other methods, the block matching method [22] has the characteristics of small calculation and easy implementation. Therefore, the block matching method is used to obtain the motion trajectory of the abnormal block.

The block matching method needs to use the matching criterion to measure the matching rate between two small blocks, so it is necessary to select the matching criteria. And the matching criteria directly affect the tracking accuracy and the calculation amount. Common matching criteria [23] are mean absolute error (MAD), mean square error function (MSE), normalized cross-correlation function (NCCF), and sum of absolute error (SAD).

Among them, the MSE criterion and the NCCF criterion require more multiplication operations, the time complexity is higher, and it is rarely used. The calculation amount of SAD is smaller than that of MAD, and the implementation

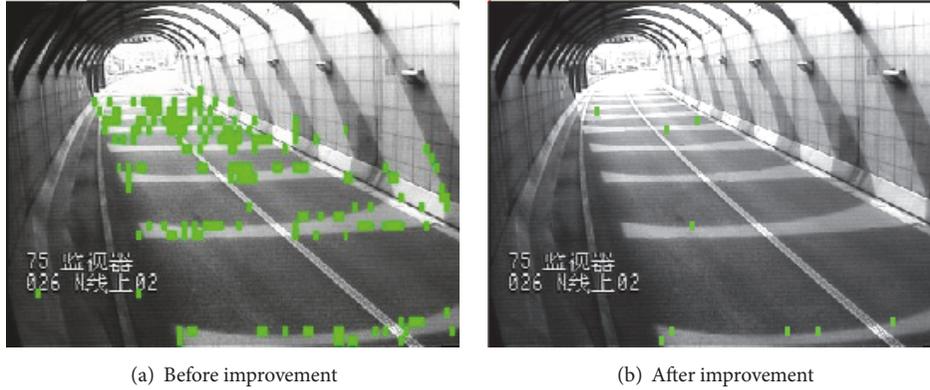


FIGURE 19: Comparison of the effects before and after the improvement of the algorithm for the exit of Shanghai Fuxing Road Tunnel.

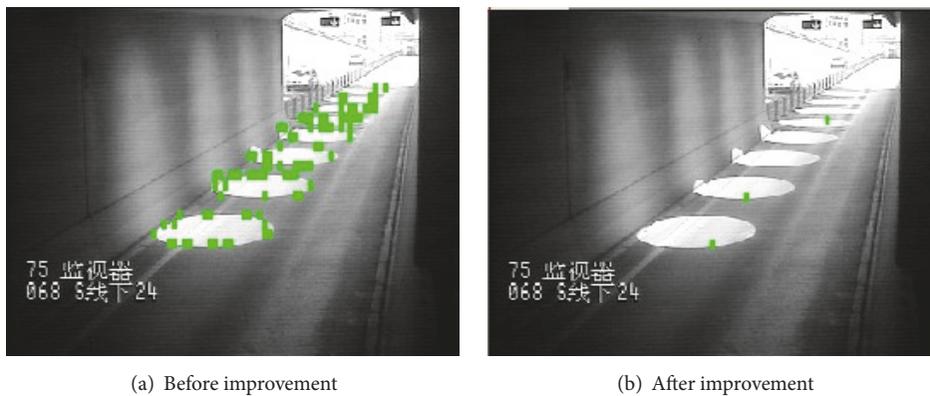


FIGURE 20: Comparison of the effects before and after the improvement of the algorithm for the entrance of Shanghai Fuxing Road Tunnel.

is simple, convenient, and widely used. Therefore, this paper uses SAD as the selection criterion for the optimal matching point.

3.1.2. Two-Way Tracking Algorithm. First, in the preliminary abnormal block, the corner point $P_0(x, y)$ is obtained by the Moravec algorithm.

Second, read the m -frame image closest to the preliminary abnormal block, and carry out backward tracking with P_0 as the initial point. Find the best matching point P_{N-1} according to the SAD criterion in the previous frame with P_0 as the initial point. Centering on P_{N-1} , continue to find the best matching point in the $N-2$ frame. And so on, if the target tracking trajectory Tracker $(P_N, P_{N-1}, \dots, P_{N-m})$ can be obtained, go to (3) for forward tracking; otherwise the exception block will not be processed.

Third, forward tracking with P_0 as the initial point. Find the best matching point P_{N+1} according to the SAD criterion in the next frame with P_0 as the initial point. Focusing on P_{N+1} , continue to find the best matching point in the $N+2$ frame. In this way, if the target tracking trajectory Tracker (P_N, P_{N+1}, \dots) cannot be obtained, the abnormality flag $D2(m, n)$ of the abnormal block is set to True.

As can be seen from Figure 24, the parking or dropping object after the two-way tracking must meet the

characteristics of having a backward trajectory without a forward trajectory.

3.1.3. Analysis of Two-Way Tracking Experiment Results. The comparison of the detected results before and after the two-way tracking is shown in Figure 25. It can be seen that after two-way tracking, false alarm due to shadows and the like have been completely eliminated.

3.2. Segmentation of Parking and Dropping Area Based on Connected Domain Analysis. The parking and dropping areas after two-way tracking contain many abnormal image blocks, which constitute the actual target area. Therefore, in order to further determine the position and size of the target area and segment the target area, the connected domain analysis of the image is performed in units of image blocks.

3.2.1. Common Connected Domain Analysis Algorithm. An image area that is adjacent in position and has the same pixel value is generally referred to as an image connected domain. The process of finding and marking all connected domains in an image is called connected domain analysis. There are many methods for connected domain analysis. Here are two of the most commonly used algorithms: two-pass scanning and seed filling. A method of finding all

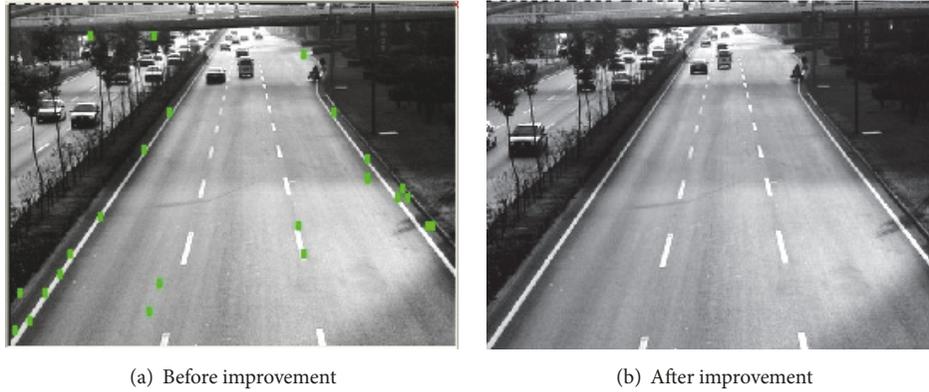


FIGURE 21: Comparison of the effects before and after the improvement of the algorithm for Xi'an South Second Ring Road.

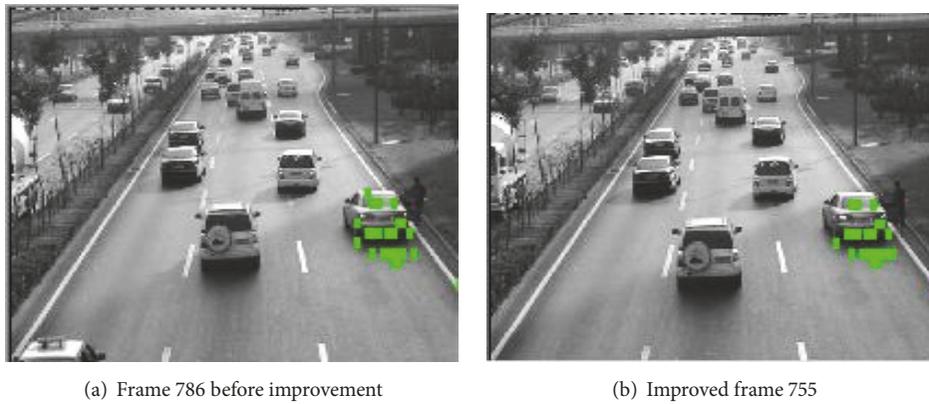


FIGURE 22: Comparison of the effect of detecting parking.

connected domains in an image and marking them by repeatedly scanning the images twice is called a two-pass scanning method [24]. Seed filling is often used to fill graphics. A foreground pixel is selected as a seed, and then all foreground pixels adjacent to the seed position and having the same pixel value as the seed are referred to as connected domains.

The above are two basic connected domain analysis methods. In view of the fact that the two-pass scanning method requires two scans of the image, and a large amount of space is needed to store the equal relationship between the markers, the seed filling method is used to analyze the connected region of the abnormal region.

3.2.2. Algorithm Description and Experimental Results. After the abnormal block detection, the abnormality $D1(m, n)$ and $D2(m, n)$ of each image block exist the following three cases:

The first one is $D1(m, n) = \text{False}$ and $D2(m, n) = \text{False}$, which means that the image block has neither state change nor track feature.

The second is $D1(m, n) = \text{True}$ and $D2(m, n) = \text{False}$, indicating that the image block has a state change but does not meet the track characteristics.

The third type is $D1(m, n) = \text{True}$ and $D2(m, n) = \text{True}$, indicating that the image block has both state changes and trajectory characteristics.

The image block of the second case is an abnormal area determined initially, and the image block of the third case is an abnormal area caused by the occurrence of a parking or a dropping event.

(1) *Algorithm Description.* Firstly, scan the image in rows in units of image blocks. If an abnormal block is scanned, a new connected region is considered to appear.

- (a) The current scanned exception block is seeded and marked. And the upper, lower, left and right boundaries of the connected area are set to be the boundary of the abnormal block. Then, the eight image blocks adjacent to the abnormal block are sequentially scanned, and if there are abnormal blocks, all the abnormal blocks are pushed onto the stack.
- (b) Popping the top block of the stack, giving the same mark, and updating the four boundaries of the connected domain according to the positional relationship between the position of the abnormal block and the upper, lower, left and right boundaries. Then, the eight image blocks adjacent to the abnormal block are sequentially scanned. If there are abnormal blocks, all the abnormal blocks are pushed onto the stack.
- (c) Repeat (b) until the stack is empty. Then a connected area with four known boundaries in the image is

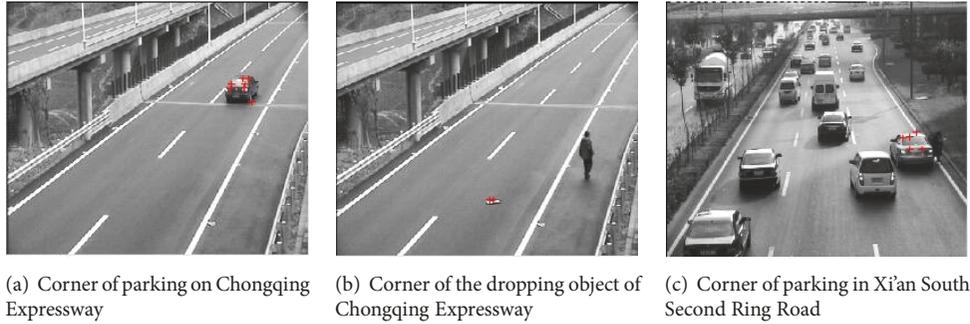


FIGURE 23: Detected corner points.

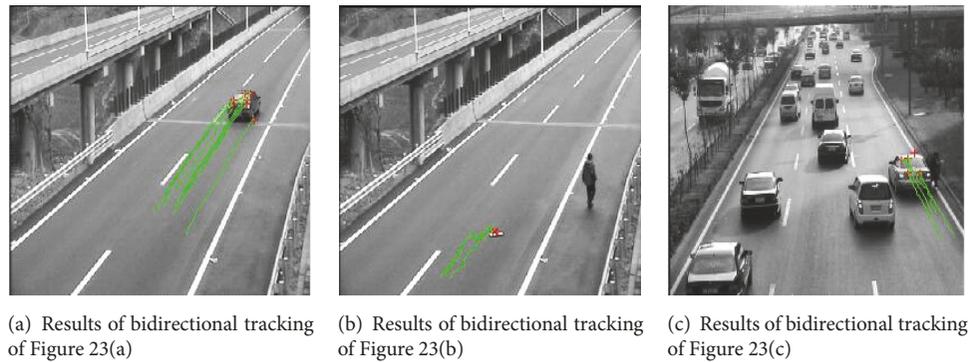


FIGURE 24: The result of bidirectional tracking of the corner points in Figure 24.

found, and all the exception blocks in the connected area are marked as True.

Second, repeat step one until the end of the scan.

After the scan, all connected domains in the image can be found. All the exception blocks in each connected domain have the same mark, and four boundaries of each connected domain can be derived. By considering each connected domain as a target area, you can determine the location and size of each target area, as shown in Figure 26.

(2) *Analysis of Results.* As can be seen from Figure 26, the abnormal region obtained by the above connected domain analysis algorithm is smaller than the actual abnormal region, and there is a case where the same abnormal target is divided into a plurality of abnormal targets, which seriously affects subsequent processing.

3.2.3. Existing Problems and Solutions. The above method of connected domain analysis has defects, mainly reflected in the following two aspects:

- (1) Due to the interference of illumination, vehicles, pedestrians, etc., the timing of each image block in the abnormal target reaching an abnormal state may be different, as shown in Figure 27.

In Figure 27, A is an abnormal target, and the gray value of each block is different. The larger the gray value, the less time it takes for the block to reach the abnormal state.

In view of this situation, it is necessary to save the abnormality flag of the detected abnormal block for a while (usually 2~3 seconds). If an image block with an abnormality flag $D1(m, n)=D2(m, n)=True$ appears around the block during this time, the block is considered to meet the adjacent condition.

- (2) When using the two-way tracking method to reduce false positives caused by shadows and other disturbances, some image blocks in the abnormal target are also removed. Thereby causing the originally connected area to become a nonconnected area, and one target becomes multiple targets. Therefore, when carrying out the connected domain analysis, all the image blocks with the abnormality flag $D1(m, n)=True$ and adjacent to the seed are pushed onto the stack. (The initial seed must be an exception block).

3.2.4. Improved Algorithm and Experimental Results. According to the two shortcomings proposed in Section 3.2.3, the connected domain analysis method is improved. The specific steps are as follows.

Firstly, scan the image in rows in units of image blocks. If an abnormal block is scanned, a new connected region is considered to appear.

- (a) The current scanned exception block is seeded and marked. And the upper, lower, left and right boundaries of the connected area are set to be the boundary of the abnormal block. Then, the eight image

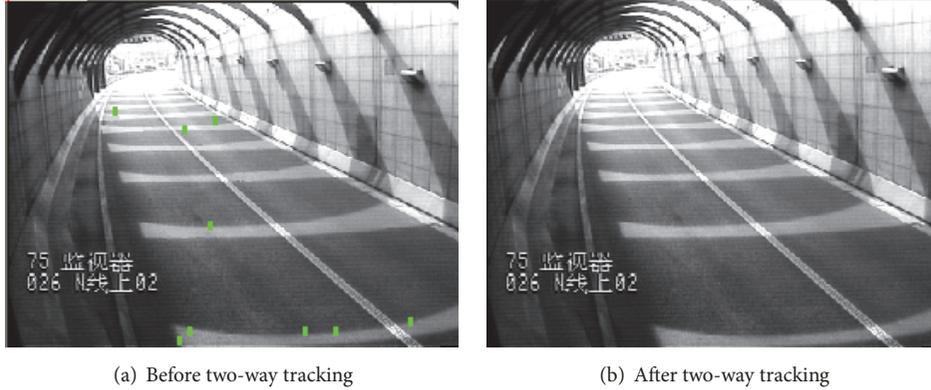


FIGURE 25: Comparison of abnormal blocks detected before and after bidirectional tracking.

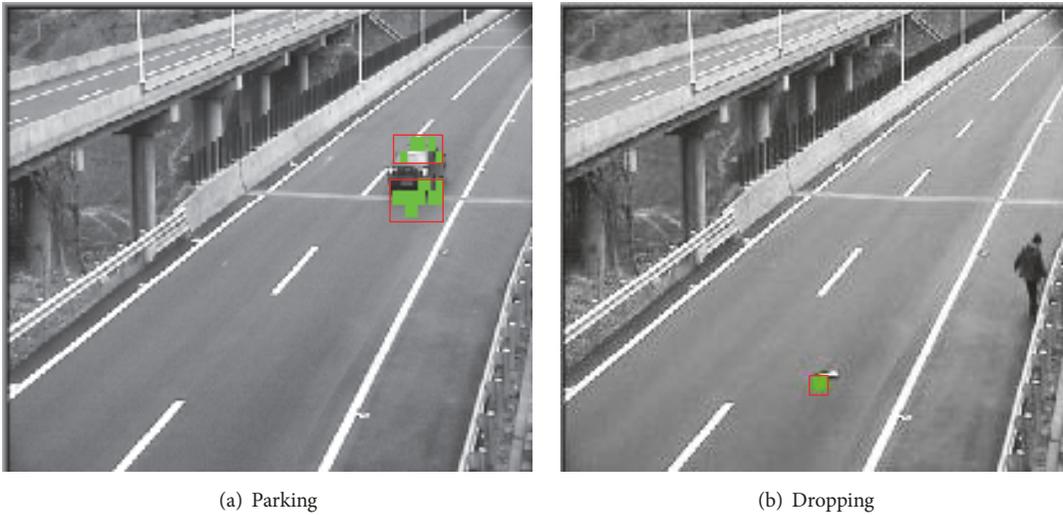


FIGURE 26: Results of connected domain analysis.

blocks adjacent to the abnormal block are sequentially scanned, and If there are image blocks with abnormal markers $D1(m, n)=True$ in these eight image blocks, all the blocks are pushed onto the stack.

- (b) Popping the top block of the stack, giving the same mark, and updating the four boundaries of the connected domain according to the positional relationship between the position of the abnormal block and the upper, lower, left and right boundaries. Then, the eight image blocks adjacent to the abnormal block are sequentially scanned. If there are image blocks with abnormal markers $D1(m, n)=True$ in these eight image blocks, all the blocks are pushed onto the stack.
- (c) Repeat (b) until the stack is empty. Then a connected area with four known boundaries in the image is found, and all the exception blocks in the connected area are marked as True.

Secondly, Repeat step one until the end of the scan.

After the scan is finished, all the connected domains in the image can be acquired; likewise, each connected domain

is considered as a target region, and the experimental result is shown in Figure 28.

By comparing Figure 26 with Figure 28, it can be found that the anomalous region obtained by the improved connected domain analysis method is more accurate.

4. Separation of Parking and Dropping Objects

The work done in the previous section only defines the parking and dropping area in the image and does not distinguish the target type, that is, whether it is parking or dropping. The traditional methods of differentiation like area, rate of change of the direction of motion, and average speed have limitations. This section describes how to differentiate between targets using 3D information.

4.1. Calibration. The use of a camera to capture images or video will result in the loss of target information, which will cause a series of problems such as geometric deformation and scale change when processing images in traffic scenes. If you can use the image to restore objects in the space and



FIGURE 27: Status difference of different blocks in the abnormal area.

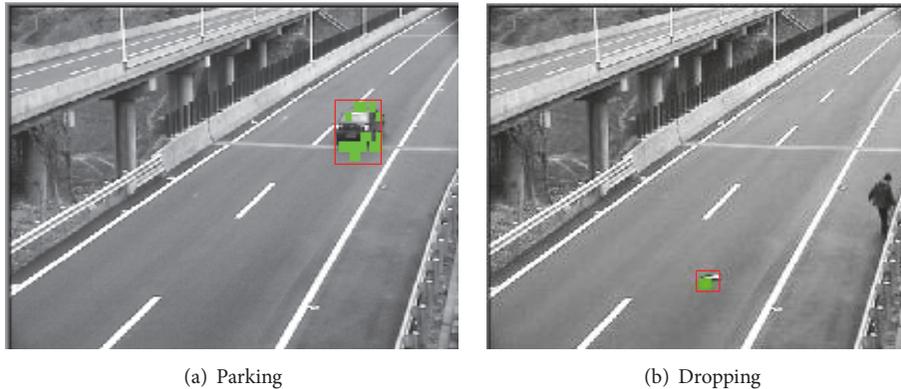


FIGURE 28: Improved results.

use the properties of the object itself, these problems will be completely eliminated. In such an environment, camera calibration technology was born. The main purpose of this technology is to define the internal and external parameters of the camera under a specific imaging model and then establish the relationship between image pixel coordinates and world coordinates.

The camera imaging process can be described by its imaging model. In this paper, we use a linear imaging model for calibration, which can make calculations easier.

With the deepening of calibration technology research, many scholars have proposed a representative calibration method [25]. In 1981, Martins [26] proposed an algorithm for camera calibration using two planes. This method simplifies the process of solving based on meeting the calibration accuracy. In 1986, R.Y.Tsai [27] established the Tsai camera model and proposed a two-step calibration method. This method simplifies the process of solving on the basis of meeting the calibration accuracy. In 1998, Zhang Zhengyou [28] proposed a method for camera calibration using 2D planar targets. This method only needs to make a calibration plate and then shoot the calibration plate at different angles. By detecting the feature corners in each photo, you can estimate the camera's internal and external parameters. The method has the advantages of low cost, high precision, and the like, and has been widely applied. In 2014, Zheng Yuan [29] proposed a camera calibration method based on

vanishing point. Its principle is as follows: as long as the vanishing point formed in the image by the parallel lines in three directions in the space, and the height of the camera in the space or a known distance on the road surface or a height perpendicular to the road surface, you can find the internal and external parameters of the camera.

Consider the existence of a large number of parallel markings in traffic scenes, and the country has uniform standards for the dimensions of these markings. Therefore, the use of these marking lines can not only easily find the vanishing point in three directions, but also easily find a line segment of a known distance on the image. Therefore, in the paper, employ the method of camera calibration based on vanishing point proposed by Zheng Yuan et al.

Since the Direct Linear Transformation (DLT) [30] requires six known points as inputs, the determination of these six points requires special markers [31]. If such a marker cannot be found in the scene to be calibrated [32], the calibration cannot be completed. No known points are used as input, and calibration can be done in scenes where the DLT cannot be calibrated using only some of the markings on the road surface and known camera parameters. Calibration is usually performed using the vanishing point in the image.

4.2. Parking and Dropping Object Distinguishing Method Based on Image Inverse Perspective Mapping. The image captured by the camera is a three-space to two-dimensional

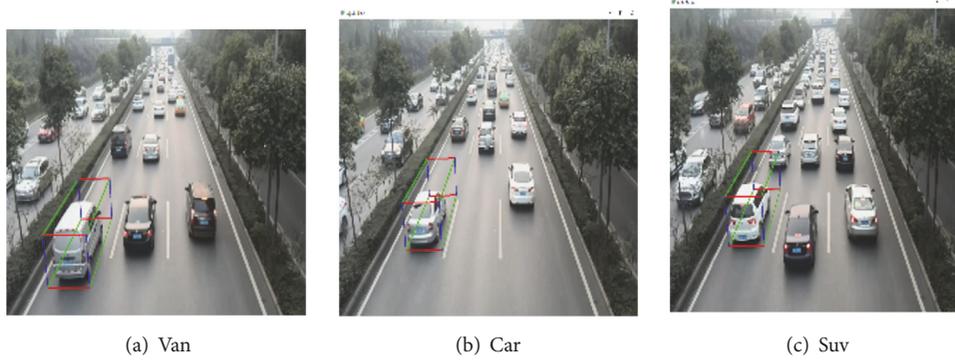


FIGURE 29: A rough three-dimensional model frame.

projection, mathematically known as the perspective mapping. According to the camera imaging model, the closer to the camera, the larger the object. And the image during projection has been deformed and physical size information has been lost.

In this paper, the inverse perspective mapping method [33] is used to establish the mapping relationship between the two-dimensional image plane and the three-dimensional space, so as to eliminate the deformation and recover the physical size of the object [34]. Points in 3D space mapped to image are unique, but there are many possibilities for inverse perspective mapping from 2D images to 3D space [35]. Points in one image correspond to points in multiple 3D spaces.

If a plane is determined in advance in three-dimensional space, the points on the plane are one-to-one correspondence with those on the two-dimensional image, and the plane is called a back-projection plane. In this way, the data in the two-dimensional image can be mapped onto the back-projection plane to obtain a map, which contains the information of the back-projection plane. And the closer the back-projection plane is to the surface of the object [36], the more accurate the three-dimensional information of the object is expressed.

We set the back-projection surface in the anomalous area and get the true size of the abnormal target through the corresponding back-projection map. If the size is close to the size of the real vehicle [37], the target is considered to be parking; otherwise it is a dropping object.

4.2.1. Algorithm Implementation

Step 1. In the abnormal pixel area, the circumscribed rectangular area of the vehicle in the image is located, and the three-dimensional model frame of the detected vehicle is roughly determined according to the circumscribed rectangle, as shown in Figure 29.

Step 2. Extract an edge of the circumscribed rectangular area of the vehicle, and construct the inverse projection of the bottom surface of the three-dimensional model frame on the obtained edge map. On the edge map, the linear segment extraction method proposed in this paper (in Section 4.2.3) is employed to extract the edge lines of the front and side of



FIGURE 30: Extraction results of the chassis line.

the vehicle chassis, and record the intersection point P of the two lines. The extraction results are shown in Figure 30.

Step 3. The area is defined by the plane that contains the bottom line (the red one in Figures 31 and 32) and is perpendicular to the road surface and the intersection of the three-dimensional model frame, which is the inverse projection of the front of the vehicle. Then according to the principle of the inverse projection, the reverse projection image can be defined as shown in Figures 33, 34, 37, and 38. In the same way, as shown in Figures 35 and 36, it is the left view.

Step 4. The line extraction algorithm described in Section 4.2.3 is used for the left view and the front view obtained in the Step 3, and the results are shown in Figures 39–44. Since the back-projected image can reflect the actual physical size of the object, the actual height of the vehicle can be obtained directly from the topmost line in the front view. And assume that the height of the vehicle is H .

Step 5. By the intersection P defined in Step 2 and the vehicle height H defined in Step 4, the position of the reverse projection top view can be determined, as shown in Figures 37, 38, and 45. Similarly, the result of the straight line detection of the top view can be obtained, as shown in Figures 43, 44, and 46.

Step 6. The three-dimensional information of the target is defined based on the result of the line extraction on the three

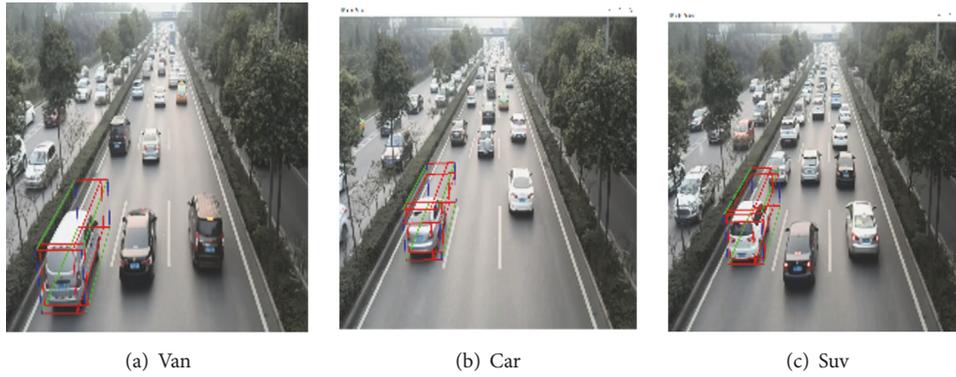


FIGURE 31: Location of the back-projection plane of the front and left views.



FIGURE 32: Location of the back-projection plane of the front and left views of truck.



FIGURE 33: The inverse projection of the main view of van.



FIGURE 34: The inverse projection of the main view of truck.



FIGURE 35: The inverse projection of the left view of van.

views. The straight line distance between the left and right sides of the top view defines the width of the object. The distance between the leftmost and rightmost of all lines in the left view defines the length of the object. The height of the object is defined from the main view in Step 4. The result of the detection of the final three-dimensional size of the white vehicle is shown in Table 1.

Table 1 is the estimation process of the three-dimensional size of the vehicle target. If any of the above steps are not performed correctly, then the abnormal area is considered to be dropping rather than parking [38–40]. For example, the number of straight lines detected on the plane of the reverse projection is too small or the target does not have a chassis edge. If all the steps are completed, then it can be determined whether the target is a vehicle target or a throw according to the three-dimensional information.

4.2.2. Construction Method of Reverse Projection. There are three inverse projective planes involved in this paper, which are the tail (or head), side and top of the vehicle, as shown in Figure 48. The inverse projective plane is divided into a grid according to a certain resolution. Since the position of the inverse projective plane has been determined, then the adverse projection relationship between the projection area and the adverse projection area in the image is determined, a one-to-one mapping relationship between coordinates of the three-dimensional space and the pixel coordinates of the image can be established [41]. The specific construction process is mapping each pixel point in the projection area

TABLE 1: Test results for white vehicle size.

	Object width	Object length	Object height
Van	188cm	405cm	197cm
Car	175cm	405cm	147cm
SUV	169cm	390cm	168cm
Truck	254cm	1843cm	321cm



FIGURE 36: The inverse projection of the left view of truck.

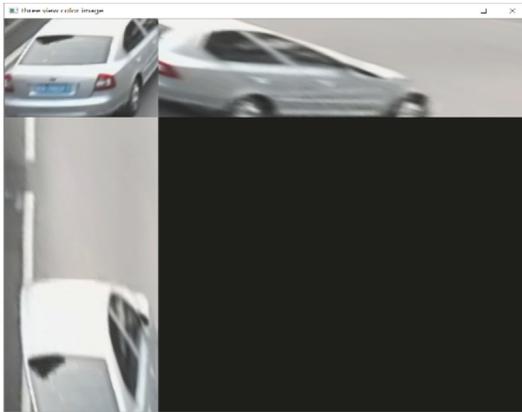


FIGURE 37: Three-view color image of car.

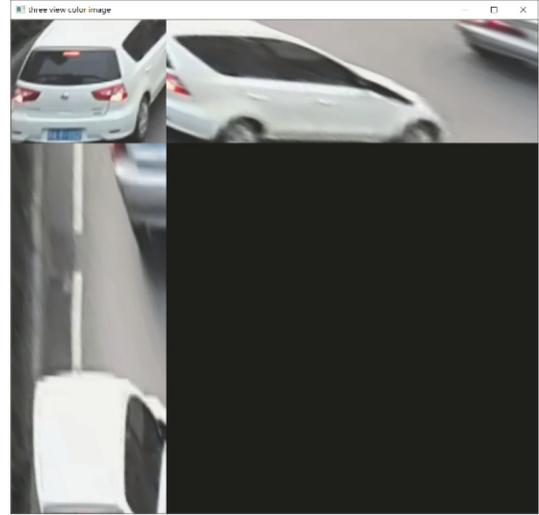


FIGURE 38: Three-view color image of suv.

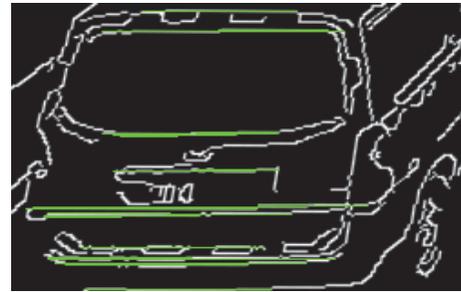


FIGURE 39: Extraction result of the line in the main view of van.

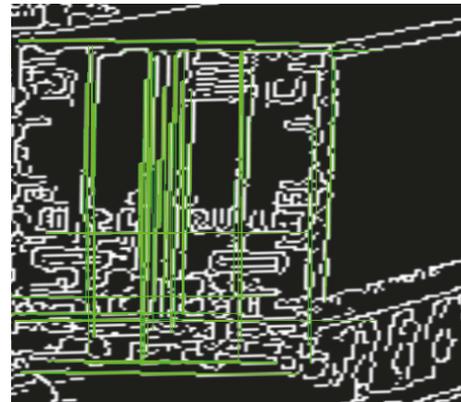


FIGURE 40: Extraction result of the line in the main view of truck.

on the image to a corresponding pixel point in the reverse projection image, that is, copying information in each small grid on the inverse projective plane into the reverse projection image. As shown in Figure 48, m denotes a small grid in the inverse projection plane, p denotes that m maps to a pixel point on the image, and mp denotes a pixel point corresponding to the grid m on the reverse projection image. Therefore, the inverse projection is the process of mapping pixel points in an image to pixel points of a reverse projection image.

As can be seen from the Figure 48, an inverse projective plane is constructed close to the target surface, and the corresponding reverse projection image is a copy of the target surface. The problem of geometric deformation of the target surface in the image due to camera perspective is eliminated [42], and the true characteristics of the target surface are well reflected, and the actual physical size of the object in the image can be reflected as realistically as possible.

4.2.3. A Method for Detecting Straight Line Based on the Reverse Projection Image. In this application, it is necessary to detect the linear segments of the vehicle body. And the smooth design of the modern vehicle manufacturing process makes the original distinct straight line segment on the

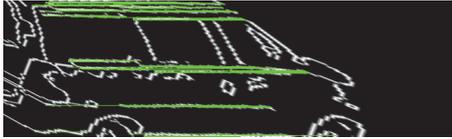


FIGURE 41: Extraction result of the line in the left view of van.

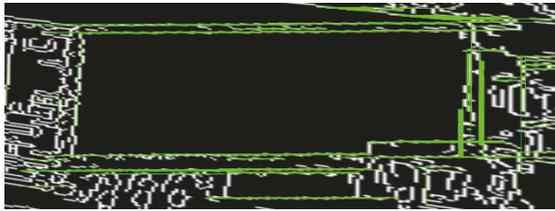


FIGURE 42: Extraction result of the line in the left view of truck.

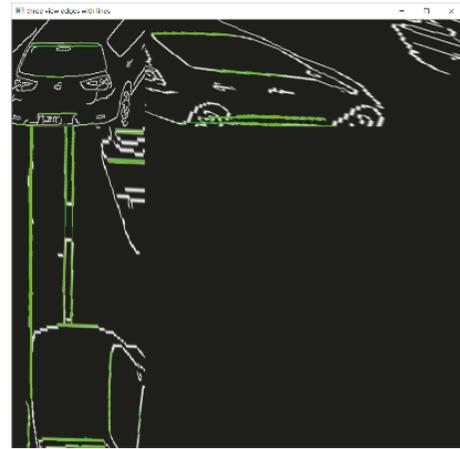


FIGURE 44: Extraction result of the line of SUV.

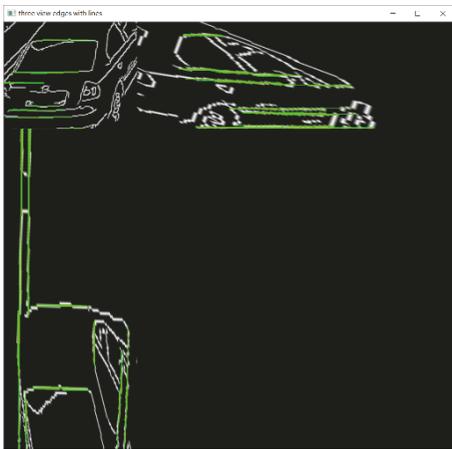


FIGURE 43: Extraction result of the line of car.



FIGURE 45: Inverse projection of the top view of van.

contour of the vehicle smooth and inconspicuous. Therefore, the conventional method of extracting a straight line cannot link the partial broken edge and the curved segments with less curvature, and the detected straight line is prone to breakage.

This paper designs a method of edge coding based on the reverse projection image. If it is the edge of the image, it is encoded as 1, not the edge of the image is encoded as -1. Calculate and get the line segment with the largest sum of the directions in which the extracted line is located. An example of the encoding of an edge is shown in Figure 47.

The advantage of this is that, without the use of time-consuming and complex algorithms such as Hough transform, the straight line is corrected using a back-projection image, so that the algorithm for detecting the line only needs to consider vertical or horizontal lines. The complexity of the algorithm is reduced and the accuracy of the detected line can be greatly improved, and the line with small curvature and local fracture is well inclusive. The algorithm flow is shown in Figure 49.

4.3. *Experimental Results and Analysis.* The algorithm proposed in the paper is implemented in VC6.0 environment [43], and the effectiveness of the algorithm is verified under

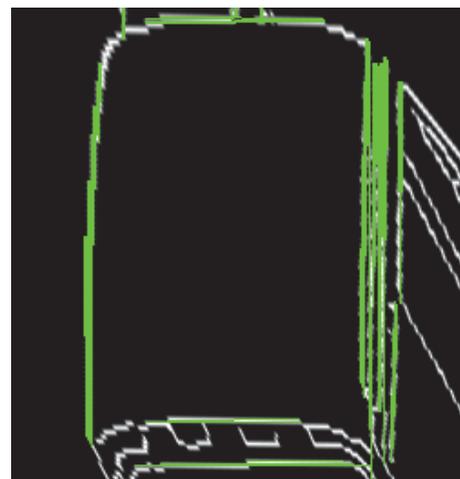


FIGURE 46: Extraction result of the line in the top view of van.

-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
1	1	-1	-1	-1	-1	-1	-1	-1	-1	-1	1	1
-1	-1	1	1	1	-1	1	1	1	-1	1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

FIGURE 47: Edge coding.

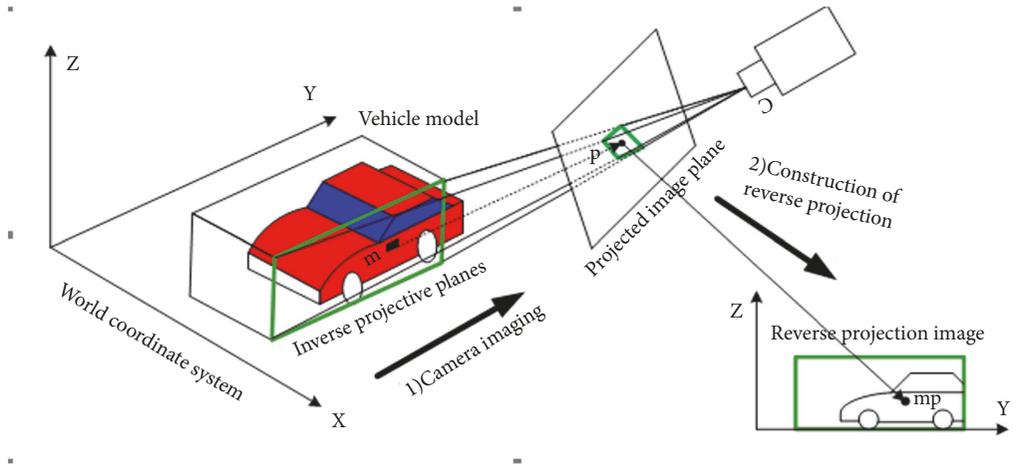


FIGURE 48: Construction of the inverse projection.

TABLE 2: Parking detection results in different scenarios.

scenarios	Number of parking events	Total number of alarms	The total number of correct alarms	Recognition rate %	Missed detection rate %	False detection rate %	Average time required for testing /ms
Xi'an South Second Ring	38	39	37	97.3	2.63	5.12	4300
Beijing Yanqing Road Section	52	54	49	94.2	5.77	9.25	4900
Chongqing Expressway	43	45	42	97.7	2.32	4.44	4100
Shanghai Outer Ring Road	60	63	58	96.7	3.33	7.93	4800
Shanghai Fuxing Road Tunnel	26	28	25	96.1	3.84	10.0	5000

different traffic scenarios. The algorithm proposed in the paper is implemented in VC6.0 environment, and the effectiveness of the algorithm is verified under different traffic scenarios. Due to the limited length of the article, five representative scenes are taken as examples, including Xi'an South Second Ring Road, Shanghai Fuxing Road Tunnel, Shanghai Overseas Ring Road, Beijing Yanqing Road Section, and Chongqing Expressway. Collect real-time video in these scenarios and use the algorithm [44, 45] designed in this paper to detect parking and dropping events. The test results are shown in Tables 2 and 3. The results of the parking test are shown in Table 2. The results of the dropping test are shown in Table 3.

It can be seen from Tables 2 and 3 that the algorithm can detect parking and dropping events more accurately in poor quality video scenes such as urban roads with high traffic flow and high-speed roads or tunnels with fast traffic speeds. Among them, the recognition rate of parking can reach more than 94%, the recognition rate of dropping objects can reach more than 92%, and the abnormality can be alarmed within 5s. Moreover, the missed detection rate and false detection rate of the parking event can be controlled below 10%, the missed detection rate of the dropping can be controlled below 10%, and the false detection rate is controlled below 20%.

Therefore, the algorithm proposed in the paper has better detection accuracy [46].

5. Conclusion

The existing algorithms for detecting parking and dropping objects generally have two shortcomings: significant dependence on background and inaccurate distinction between parking and dropping objects.

In view of the above deficiencies, we study on two aspects: the detection of stationary targets and the differentiation of target types. First, based on the method of the status change, when there is an object that did not exist before in the traffic scene, the abnormal region in the scene is preliminarily defined, and then the initially determined abnormal region is bidirectionally tracked to further determine the target from the motion to the stationary in the scene. Finally, the eight-neighbor seed filling method is used to segment the target area. Therefore, the dependence on the background is reduced. Only tracking the target area where the state changes is needed, which significantly reduces the amount of computation. Second, a method of using the three-dimensional information of the target to distinguish the target type is proposed. Firstly, use the difference between

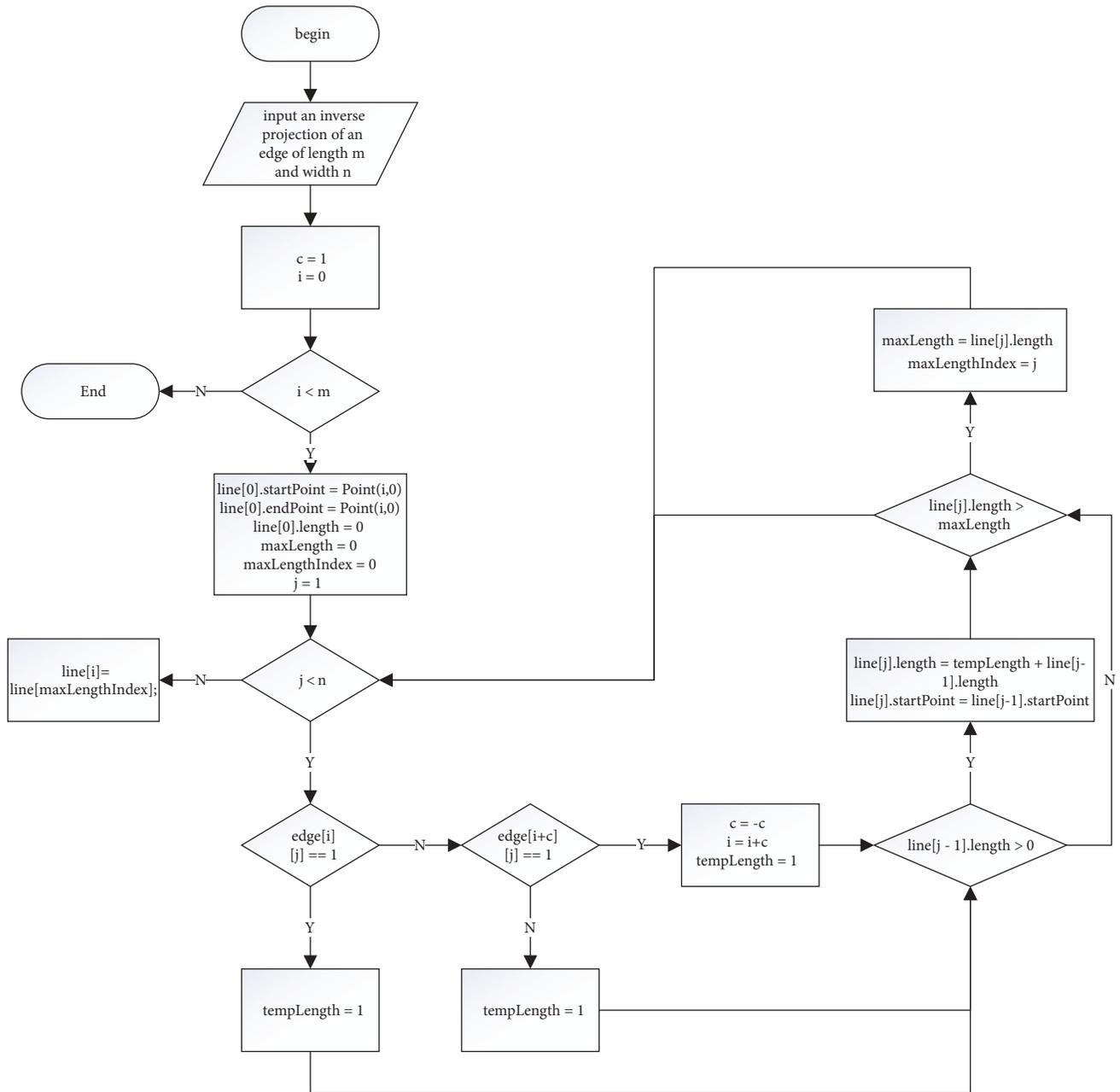


FIGURE 49: Algorithm for detecting straight lines based on the inverse projection.

the projections of the feature points to determine the relative height between the feature points, and use the height to distinguish the parking and dropping objects. Secondly, establish a 3D wireframe model of common vehicle model. The parking and the dropping objects are distinguished by the matching of the projection of the wireframe model on the two-dimensional image and the area of the target area. Thirdly, by establishing the inverse projection planes of different heights, the three-dimensional information of the length, width, and height of the target is obtained, and the parking and the dropping objects are distinguished by the size of the vehicle which is known. This method of distinguishing target types using three-dimensional information

can not only accurately distinguish the parking and dropping objects, but also roughly classify the models of stationary vehicles.

By testing in a large number of different traffic scenarios [47], the results show that the algorithm can effectively detect the parking and dropping objects based on the low miss detection rate and false detection rate and can meet the real-time requirements.

Data Availability

The measured data used to support the findings of this study have not been made available because it belongs to the local

TABLE 3: Detection results of the falling objects in different scenarios.

scenarios	Number of parking events	Total number of alarms	The total number of correct alarms	Recognition rate %	Missed detection rate %	False detection rate %	Average time required for testing /ms
Xi'an South Second Ring	14	15	13	92.8	9.09	13.3	4300
Beijing Yanqing Road Section	19	22	18	94.7	7.14	18.2	4900
Chongqing Expressway	16	18	15	93.7	6.25	16.7	4100
Shanghai Outer Ring Road	15	17	14	93.3	6.67	17.6	4800
Shanghai Fuxing Road Tunnel	13	14	12	92.3	7.69	14.3	5000

authorities of traffic control and management in Shanghai, Xi'an, and Chongqing, China.

Conflicts of Interest

The authors declare that all data availability in this article are true and reliable and there are no conflicts of interest regarding the publication of this paper.

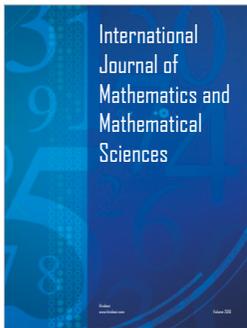
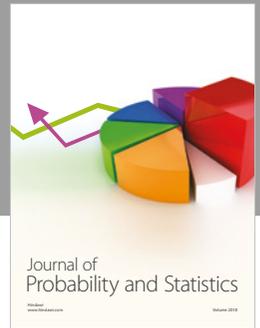
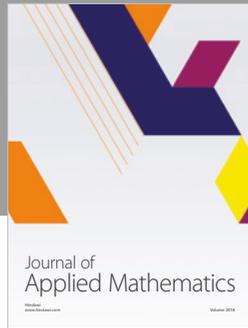
Acknowledgments

The work was funded by the Project of Shaanxi Provincial Science and Technology Program (Grant no. 2014JM8351), the Fundamental Research Funds for the Central Universities (Grants nos. 2013G1241109 and 300102248305), and the National Natural Science Foundation of China (Grant no. 61501058). Thanks are due to Liting Sun for the great work she has done.

References

- [1] M. Alam, D. Moroni, G. Pieri et al., "Real-time smart parking systems integration in distributed ITS for smart cities," *Journal of Advanced Transportation*, vol. 2018, Article ID 1485652, 13 pages, 2018.
- [2] J. Li, C. Liu, J. X. Yu, Y. Chen, T. Sellis, and J. S. Culpeper, "Personalized influential topic search via social network summarization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1820–1834, 2016.
- [3] A. Bevilacqua and S. Vaccari, "Real time detection of stopped vehicles in traffic scenes," in *Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 266–270, DBLP, London, UK, September 2007.
- [4] B.-F. Wu, C.-C. Kao, C.-C. Liu, C.-J. Fan, and C.-J. Chen, "The vision-based vehicle detection and incident detection system in Hsueh-Shan tunnel," in *Proceedings of the 2008 IEEE International Symposium on Industrial Electronics, ISIE 2008*, pp. 1394–1399, UK, July 2008.
- [5] S. Guler, J. A. Silverstein, and I. H. Pushee, "Stationary objects in multiple object tracking," in *Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2007*, pp. 248–253, UK, September 2007.
- [6] Z. Beibei and W. Qisheng, "The study of highway illegal parking real-time detection algorithm based on video," *Journal of Electronic Science and Technology*, vol. 24, no. 9, pp. 20–23, 2011.
- [7] R. Akhawaji, M. Sedky, and A.-H. Soliman, "Illegal parking detection using Gaussian mixture model and kalman filter," in *Proceedings of the 14th IEEE/ACS International Conference on Computer Systems and Applications, AICCSA 2017*, pp. 840–847, Hammamet, Tunisia, November 2017.
- [8] H. Tianfu, B. Jie, L. Ruiyuan et al., "Detecting vehicle illegal parking events using sharing bikes' trajectories," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 340–349, London, UK, August, 2018.
- [9] F. Porikli, "Detection of temporarily static regions by processing video at different frame rates," in *Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2007*, pp. 236–241, UK, September 2007.
- [10] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, 2000.
- [11] D.-H. Wang, H.-Y. Hu, Z.-H. Li, and Z.-W. Qu, "Detection and recognition algorithm of illegal parking," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 40, no. 1, pp. 42–46, 2010.
- [12] M. Chunyang, M. Xing, and Z. Panpan, "Smart detection of vehicle in illegal parking area by fusing of multi-features," in *Proceedings of the 9th International Conference on Next Generation Mobile Applications, Services and Technologies, NGMAST 2015*, pp. 388–392, UK, September 2015.
- [13] X. Xuemei, W. Chenye, S. Chen, S. Guangming, and Z. Zhao, "Real-time illegal parking detection system based on deep learning," in *Proceedings of the 2017 International Conference on Deep Learning Technologies (ICDLT '17)*, pp. 23–27, ACM, New York, NY, USA, 2017.
- [14] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering - ASME Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [15] M.-H. Lin, J. G. Carlsson, D. Ge, J. Shi, and J.-F. Tsai, "A review of piecewise linearization methods," *Mathematical Problems in Engineering*, vol. 2013, Article ID 101376, 8 pages, 2013.
- [16] X. Su, W. Fang, Q. Shen, and X. Hao, "An image enhancement method using the quantum-behaved particle swarm optimization with an adaptive strategy," *Mathematical Problems in Engineering*, vol. 2013, Article ID 824787, 14 pages, 2013.

- [17] S. Singh, S. Saurav, R. Saini et al., "Comprehensive review and comparative analysis of hardware architectures for sobel edge detector," *ISRN Electronics*, vol. 2014, Article ID 857912, 9 pages, 2014.
- [18] L. Yu, "Image noise preprocessing of interactive projection system based on switching filtering scheme," *Complexity*, vol. 2018, Article ID 1258306, 10 pages, 2018.
- [19] R.-Y. Tang and Z.-M. Zeng, "Adaptive fractional differentiation harris corner detection algorithm for vision measurement of surface roughness," *Advances in Mathematical Physics*, vol. 2014, Article ID 494237, 6 pages, 2014.
- [20] J. Khan, S. Bhuiyan, and R. Adhami, "Feature point extraction from the local frequency map of an image," *Journal of Electrical and Computer Engineering*, vol. 2012, Article ID 182309, 15 pages, 2012.
- [21] Q. Peng, L. Tu, K. Zhang, and S. Zhong, "Automated 3D scenes reconstruction using multiple stereo pairs from portable four-camera photographic measurement system," *International Journal of Optics*, vol. 2015, Article ID 471681, 9 pages, 2015.
- [22] S. Zhu, L. Li, J. Chen, and K. Belloulata, "An efficient fractal video sequences codec with multiviews," *Mathematical Problems in Engineering*, vol. 2013, Article ID 853283, 8 pages, 2013.
- [23] C. Choi and J. Jeong, "Fast motion estimation algorithm using dual bit-plane matching criteria," in *Proceedings of the 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2014*, pp. 398–401, Croatia, May 2014.
- [24] W. Wu, Z. Zhou, S. Wu, and Y. Zhang, "Automatic liver segmentation on volumetric CT images using supervoxel-based graph cuts," *Computational and Mathematical Methods in Medicine*, vol. 2016, Article ID 9093721, 14 pages, 2016.
- [25] L. Wang, S. Lin, J. Yang et al., "Dynamic traffic congestion simulation and dissipation control based on traffic flow theory model and neural network data calibration algorithm," *Complexity*, vol. 2017, Article ID 5067145, 11 pages, 2017.
- [26] H. A. Martins, J. R. Birk, and R. B. Kelley, "Camera models based on data from two calibration planes," *Computer Graphics and Image Processing*, vol. 17, no. 2, pp. 173–180, 1981.
- [27] R. Y. Tsai, "A versatile camera calibration technique for high accuracy. 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE Transactions on Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987.
- [28] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
- [29] Y. Zheng and S. Peng, "A practical roadside camera calibration method based on least squares optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 15, pp. 831–843, 2014.
- [30] Y. I. Abdel-Aziz, H. M. Karara, and M. Hauck, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry," *Photogrammetric Engineering and Remote Sensing*, vol. 81, no. 2, pp. 103–107, 2015.
- [31] L. Jianxin, W. Xinjue, D. Ke et al., "Discovering influential community over large social networks," in *Proceedings of the 33rd IEEE International Conference on Data Engineering (ICDE)*, 2017.
- [32] S. D. Ma, "A self-calibration technique for active vision systems," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 114–120, 1996.
- [33] H. Li, M. Feng, and X. Wang, "Inverse perspective mapping based urban road markings detection," in *Proceedings of the 2nd IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS '12)*, pp. 1178–1182, Hangzhou, China, November 2012.
- [34] J. Li, C. Liu, and J. X. Yu, "Context-based XML keyword query diversification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 3, pp. 660–672, 2015.
- [35] W. Wang, C. Shen, J. Zhang, and S. Paisitkriangkrai, "A two-layer night-time vehicle detector," in *Proceedings of the Digital Image Computing: Techniques and Applications (DICTA '09)*, pp. 162–167, Melbourne, Australia, December 2009.
- [36] J. Li, C. Liu, R. Zhou, and J. X. Yu, "Quasi-SLCA based keyword query processing over probabilistic XML data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 4, pp. 957–969, 2014.
- [37] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, pp. 850–855, Hong Kong, China, August 2006.
- [38] Y. Ke, F. Qin, W. Min, and G. Zhang, "Exposing image forgery by detecting consistency of shadow," *The Scientific World Journal*, vol. 2014, Article ID 364501, 9 pages, 2014.
- [39] S. Y. Chen, H. Tong, and C. Cattani, "Markov models for image labeling," *Mathematical Problems in Engineering*, vol. 2012, Article ID 814356, 18 pages, 2012.
- [40] N. Zhou, T. Yang, and S. Zhang, "An improved FCM medical image segmentation algorithm based on MMTD," *Computational and Mathematical Methods in Medicine*, vol. 2014, Article ID 690349, 8 pages, 2014.
- [41] B. Wang, T. Su, X. Jin, J. Kong, and Y. Bai, "3D reconstruction of pedestrian trajectory with moving direction learning and optimal gait recognition," *Complexity*, vol. 2018, Article ID 8735846, 10 pages, 2018.
- [42] L. Zhao, L. F. Liu, Q. Y. Wang, T. J. Li, and J. H. Zhou, "Moving target detection and active tracking with a multicamera network," *Discrete Dynamics in Nature and Society*, vol. 2014, Article ID 976574, 11 pages, 2014.
- [43] K. F. Ackermann, B. Hoffmann, L. S. Indrusiak, and M. Glesner, "Providing memory management abstraction for self-reconfigurable video processing platforms," *International Journal of Reconfigurable Computing*, vol. 2009, Article ID 851613, 15 pages, 2009.
- [44] M. Roca-Riu, J. Cao, I. Dakic, and M. Menendez, "Designing dynamic delivery parking spots in urban areas to reduce traffic disruptions," *Journal of Advanced Transportation*, vol. 2017, Article ID 6296720, 15 pages, 2017.
- [45] B. Tian, Y. Li, B. Li, and D. Wen, "Rear-view vehicle detection and tracking by combining multiple parts for complex urban surveillance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 597–606, 2014.
- [46] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, Mateo, CA, USA, 1988.
- [47] J. Zhao, M. Han, C. Li, and X. Xin, "Visibility video detection with dark channel prior on highway," *Mathematical Problems in Engineering*, vol. 2016, Article ID 7638985, 21 pages, 2016.



Hindawi

Submit your manuscripts at
www.hindawi.com

