Research Article

Vehicle tracking fusing the prior information of Kalman filter under occlusion conditions

Zhe Li[1] · Yangzhou Chen[1] · Zhuo Yin[1]

Abstract

The occlusion in traffic scenarios have posed great challenges in vehicle tracking. Establishing effective and robust vehicle tracking algorithms under occlusion conditions is a necessary issue for many traffic applications. This paper presents a novel approach of vehicle tracking by fusing the prior information of Kalman filter. The prior information of Kalman filter is used for background update, precise morphological operation and occlusion judgment. The fusion of observation and prior information is adopted to segment vehicles under occlusion. It effectively improves the robustness of the vehicle tracking algorithm in the case of occlusion. A novel vehicle description method is proposed to express vehicle shape more accurately and thus reduces the misjudgement of occlusion. Experiments show that the proposed algorithm can accurately track vehicles even under occlusion.

Keywords Vehicle tracking · Vehicle occlusion · Kalman filter · Information fusion

1 Introduction

Video surveillance equipment has been widely applied to urban traffic especially for key sections and intersections because of low hardware demand, cheaper and cheaper cost, installing and maintaining easily, etc. Detection and tracking of vehicles through video monitoring equipment has being become a fundamental research issues. However, video-based vehicle detection and tracking is less robust under certain conditions and circumstances, such as susceptibility to occlusion, illumination changing, bad weather, and night. Vehicle occlusion in urban traffic scenarios is very common, so it is a fairly challenging task to effectively detecting and tracking vehicles under occlusion conditions.

In the literature, target tracking algorithms under occlusion conditions are mainly divided into five categories, [1]: (1) center weighted regions matching [2, 3], (2) subblock matching [4–9], (3) trajectory prediction [2, 4, 6, 8, 10], (4) Bayesian theory [4, 10, 11], (5) multiple algorithm fusion [2–11]. Wang et al. [2] use the Mean-Shift algorithm for vehicle tracking in the absence of occlusion; when occlusion occurs, the GM(1,1) model based on historical vehicle position information is used to predict the vehicle position at the next moment. However, they do not utilize the current detection value. The algorithm uses the Bhattacharyya coefficient to determine whether the occlusion occurs or not, but it greatly increases the computation. In [3], the target block in the tracking window is divided into some sub-blocks whose confidence will be evaluated. When the occlusion occurs, the Mean-Shift tracking algorithm is performed for each sub-block, and the sub-block with the highest confidence is used as the final position of the target. The confidence of each sub-block is decided by the Bhattacharyya coefficient and background discrimination. In [4], SIFT features of the target are extracted when the target is occluded, and feature template matching is conducted in the next frame to locate the vehicle position. Liang et al. [5] divide the target into several partial image blocks and weight each local image block. If a partial

✉ Yangzhou Chen, yzchen@bjut.edu.com | [1]Beijing Key Lacoratory of Traffic Engineering and College of Aritificial Intelligence and Automation, Beijing University of Technology, Beijing 100124, China.

SN Applied Sciences (2019) 1:822 | https://doi.org/10.1007/s42452-019-0852-2

Received: 19 March 2019 / Accepted: 27 June 2019 / Published online: 4 July 2019

image block is occluded, the corresponding weight is small, thereby reducing the appearance change of the target due to occlusion. However, the proposed algorithm has poor real-time performance, and the tracking effect is poor when multiple factors change simultaneously. Ghasemi et al. [6] adopt two methods to solve the occlusion problem, (1) match the target foreground with the vehicle template and locate the vehicle accurately, if the similarity is greater than a threshold; (2) use Kalman filter to predict vehicle position, if the similarity is less than this threshold. Wu et al. [7] propose the RDHOGPF feature to solve the problem of partial occlusion of vehicles. Krieger et al. [8] propose the DRIFT feature to solve the problem of partial occlusion of vehicles and for the case of complete occlusion they use kalman filter to predict vehicle position. Qing et al. [9] use the taillights to adjust the bounding box so as to accurately locate the position of the vehicle. However, this algorithm is easy to be affected by the background and has poor stability when vehicles have a same color. Qin et al. [10] use particle filter to predict the vehicle position and use the predicted position as the input of KCF algorithm, so as to solve the vehicle tracking problem in the case of occlusion. Aeschliman et al. [11] use historical images to replace the target regions of the current frame with the background image of the corresponding regions by maximizing the post estimation probability of the target, so as to establish a more accurate background model and improve the vehicle tracking effect under occlusion. When occlusion occurs, many works [12–15] use geometric methods to find the concave points in the occlusion region, so as to divide the block containing several vehicles. These methods take a large amount of calculation and requires excessive dilation operation. It is easy to combine the originally unconnected vehicles into a block, resulting in extra occlusion.

Based on the existing literature, this paper proposes a vehicle tracking method fusing the prior information of Kalman filter, and improves vehicle tracking effect in the aspects of background update, morphological operation, occlusion judgment and vehicle description. As the accurate and real-time background model update is closely related to the vehicle tracking effect, this paper takes the image regions outside the target in each frame image as the background regions to carry out the background update in real time or at intervals, so as to obtain a real-time and more accurate background image. Using the prediction value of the vehicles of the Kalman filter, accurate morphological operation is carried out on the binary image, and the occlusion caused by the morphological operation is effectively avoided. Due to different occlusion causes and results, occlusion problems are divided into two categories in this paper: (1) occlusion between vehicles; (2) occlusion between a vehicle and a roadside obstruction. The first category of occlusion is the merging of vehicles as a block. For the first category of occlusion, the segmentation operation is adopted. The second category of occlusion shows that the vehicle is divided into several blocks by a roadside obstruction. For the second category of occlusion, merging operation is adopted. This paper identifies occlusion by fusing the prior information of Kalman filter. The judgment condition of the first category of occlusion is that a detection value contains several prediction values. The judgment condition of the second category of occlusion is that a prediction value contains several detection values. Different from previous literatures, this paper fuses the prediction value and detection value of the current frame in the segmentation operation, instead of only using the prediction value, so as to get a more accurate estimation of vehicle position. In the proposed algorithm, a novel vehicle description method is adopted. In the literature, the minimum external rectangle parallel to the coordinate axis is used to describe the vehicle (as shown in Fig. 9). This description cannot accurately express the target region, so it is difficult to judge whether the occlusion occurs through the rectangular box, and it is unable to conduct accurate image manipulation. For this reason, the minimum external rectangle is adopted for vehicle description (as shown in Fig. 10), which makes the description of vehicle form (such as shape and direction) more accurate.

1.1 Fusion algorithm under occlusion conditions

The proposed fusion algorithm includes two parts: vehicle detection and vehicle tracking. The algorithm process is shown in Fig. 1.

In order to solve the problems in existing vehicle detection algorithms based on background difference method, this paper proposes corresponding improvements in two aspects. The first is to reduce the possibility of occlusion. The second is to solve the problem of the failure of tracking under occlusion.

1.2 Vehicle detection

In high-point video, when the two cars are close to each other or have occlusions, the morphological operation often process them into a connected block. By fusing prior information of Kalman filter of vehicle moving, the occurrence of this situation can be effectively reduced or even avoided, thereby improving the accuracy of vehicle detection and reducing tracking loss rate.

Due to fusing the prior information, the whole vehicle detection algorithm is divided into two parts: global

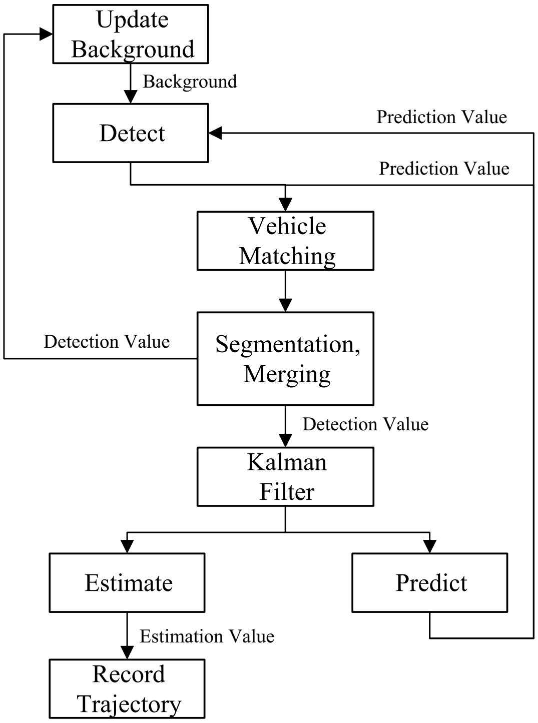


Fig. 1 Algorithm flow chart

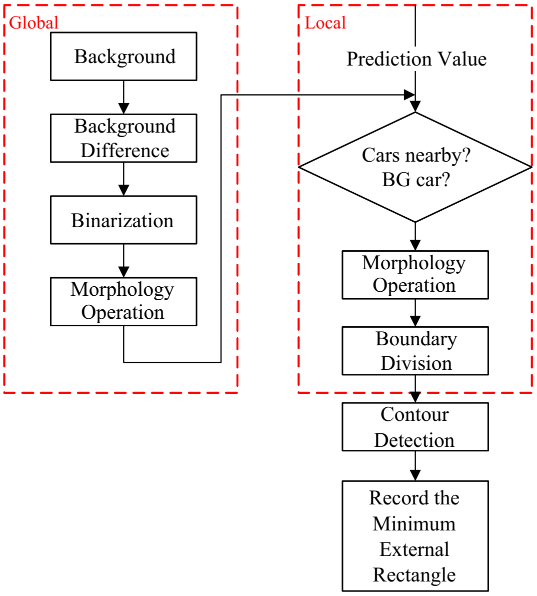


Fig. 2 Vehicle detection flow chart

operation and local operation, as shown in Fig. 2. The global operations are for initialization and the local operations are to reduce the occurrence of occlusion.

1.3 Global operations

The purpose of the global operation is to preprocess the image and remove noise as much as possible without expanding the vehicle region.

First, it is necessary to create a background image. The accuracy of the background image has direct effect in detecting vehicles. Considering the continuity of a video, the background image changes continuously. Therefore, in the initialization process, several consecutive video images are used to cut and splice a real background image. The positions and the sizes of vehicles can be obtained by vehicle detection in the current frame. At this point, the regions outside the vehicle are taken as new background image to update the corresponding regions of the original background image (as shown in Fig. 3).

Second, background difference method is used for video images (as shown in Fig. 4), and binarization is performed (as shown in Fig. 5).

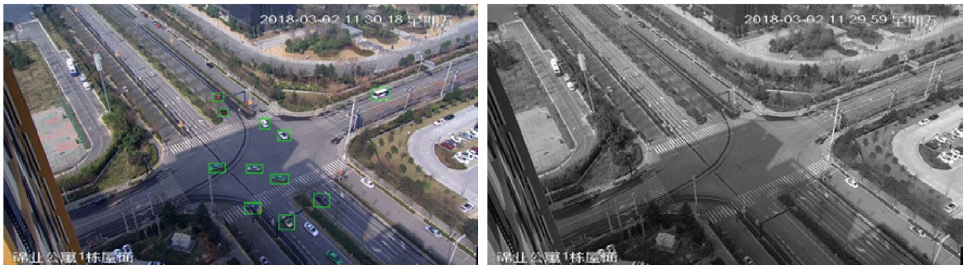
Finally, a morphology opening operation is performed on the whole image (as shown in Fig. 6). The all structure elements of morphology operation in this paper are rectangles of 3×3.

The above operations can remove the noise and make the vehicle regions highlighted. However, large holes may exist in vehicle regions, which is not conducive to vehicle contour detection. If the whole image continues to be dilated, the noise point will be expanded into a noise block, which will cause great interference to the detection, and excessive dilation operations will make originally separated vehicles be connected, which will lead to false rejection and false detection. Therefore, the priori information of Kalman filter is fused to process the vehicle regions more precisely.

The computational complexity of global operations is $w_g \times h_g \times t_g$. Here, w_g is the width of the whole image, h_g is the height of the whole image, and t_g is the times of morphological operations. Note that other operations, such as binaryzation, are also spend time. But they are not time-consuming bottlenecks. In computational complexity, we are only talking about time bottlenecks in the algorithm, specifically, morphological operations.

1.4 Local operations

When processing the current t-th frame image, the estimation values of Kalman filter and state transfer equations of the $(t-1)$-th frame image are used to obtain the predicted values of the t-th frame image (the predicted positions of the vehicles in the t-th frame image). The predicted values are considered as the prior information of Kalman filter. On this basis, through further image processing operations on



(a) The green rectangles of the vehicle regions. **(b)** The background image.

Fig. 3 Background image update

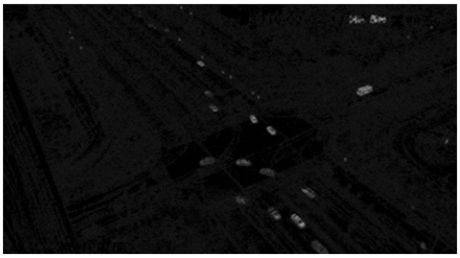


Fig. 4 Background difference image

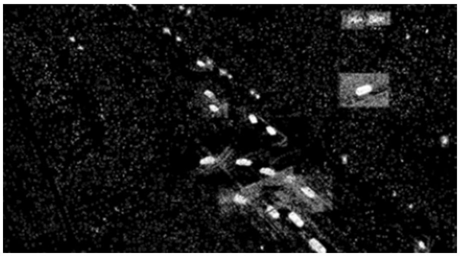


Fig. 5 Binarization image

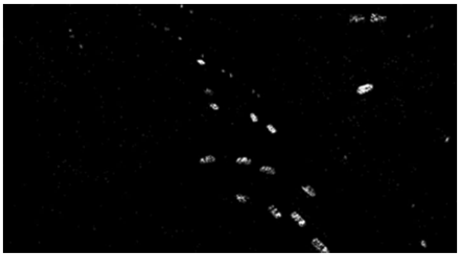


Fig. 6 Global morphology opening operation

the vehicle regions, more accurate positioning of vehicles can be achieved.

Taking a vehicle detected as an example, the first step is to determine whether there are other vehicles around the vehicle and whether the vehicle is regard as a background (BG for short) vehicle whose color is close to the background's. Different morphological operations are used in different cases (as shown in Table 1). The numbers in the table indicate the times of corresponding morphological operations. Taking the case of "Cars nearby, not BG car" as an example, two erosion operations and one dilation operation are performed on the vehicle regions orderly. Local morphological operations can not only reduce the occurrence of occlusion, but also effectively solve the problem of false rejection of background vehicles.

Table 1 Morphological operation in different cases

	Cars nearby, BG car	Cars nearby, not BG car	No cars nearby, BG car	No cars nearby, not BG car
Erode	0	2	0	0
Dilate	2	1	4	2

The binarization image has only two colors: black and white. The black indicates background and the white indicates foreground. In order to further eliminate vehicle occlusion, prior information is used to process the vehicles exactly. Suppose that the current t-th frame image is being processed. The vehicle positions are unknown before contour detection. But the prediction values in (t – 1)-th frame image can be used to replace the unknown vehicle positions in t-th frame image. With this knowedge (prior information of Kalman filter), we can judge whether a vehicles is near another one. If there are other vehicles near a vehicle, take the boundary line of the vehicle near other vehicles (as shown in Fig. 7) as the dividing line of the two vehicles. The red line in Fig. 7, which belongs to the prediction value, is the boundary of the target car and the nearby car.

The image after local processing is shown in Fig. 8. At this point, the prior information is fully utilized to further process the vehicle regions, and the dividing line is effectively used to separate different vehicles. These two steps greatly reduce the impact of vehicle occlusion. Then the general contour detection algorithm can be used to extract the positions and the scales of the vehicles.

The computational complexity of local operations is $\sum_{i=1}^{n} w_i \times h_i \times t_i$. Here, w_i is the width of the i-th vehicle in

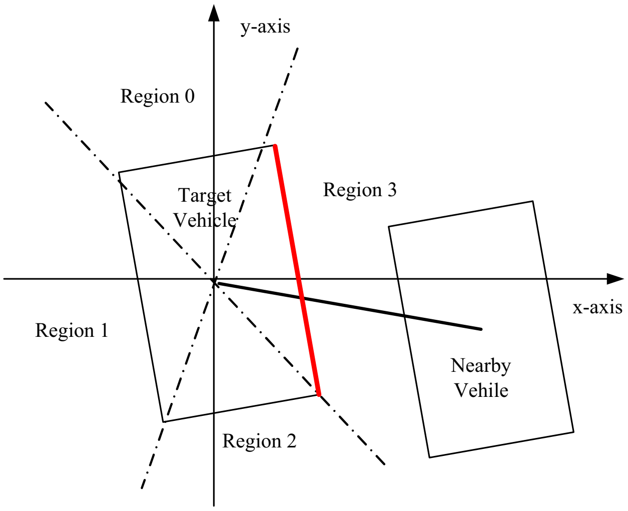


Fig. 7 A schematic diagram of the boundary of cars

Fig. 8 Local morphological operation

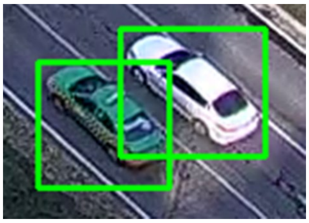


Fig. 9 The minimum external rectangle parallel to the coordinate axis

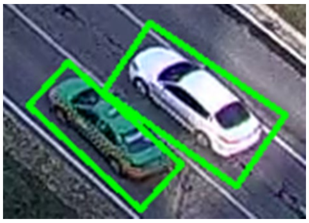


Fig. 10 The minimum external rectangle

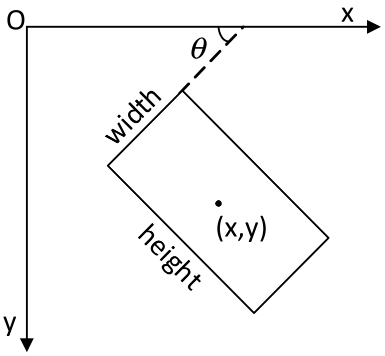


Fig. 11 The diagram of box

the current frame image, h_i is the height of the i-th vehicle in the current frame image, t_i is the times of morphological operations of the i-th vehicle, and n is the numbers of vehicles in the current frame image.

1.5 Vehicle description method

In previous studies, the minimum external rectangle parallel to the coordinate axis is usually used for vehicle description (as shown in Fig. 9). The description method is expressed as the following rectangle:

$$\text{rect} = \{\text{x}, \text{y}, \text{width}, \text{height}\} \tag{1}$$

Here, (x, y) indicates the upper-left coordinates of the rectangle; (width, height) indicates the width and the height of the rect. Although this description method can accurately describe the vehicle position, it cannot accurately describe the scale and the direction of the vehicle, so it cannot accurately determine whether occlusion occurs. In Fig. 9, for example, there is no occlusion between the two vehicles, but their detection values (green rectangular) overlap.

This imprecise description method is not conducive to the subsequent operations of the algorithm. In order to describe the scale and the direction of the vehicle more accurately, we propose a novel vehicle description, which is the minimum external rectangle of the vehicle (as shown in Fig. 10). The description is expressed as the following rectangle:

$$\text{box} = \{\text{x}, \text{y}, \text{width}, \text{height}, \theta\} \tag{2}$$

Here (x, y) indicates the coordinates of the center of box. (width, height) indicates the width and the height of the box, where when the horizontal axis rotates counterclockwise around the origin of coordinates, the edge of the box first parallel to the horizontal axis is the width of the box. θ indicates the angle between the horizontal axis and the width of the box (acute angle). Figure 11 accounts for these parameters in details.

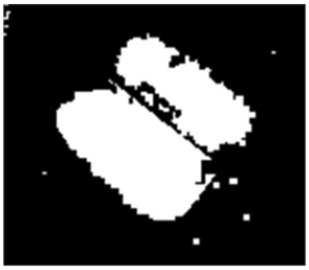


Fig. 12 The binary image of the occlusion occurring between vehicles

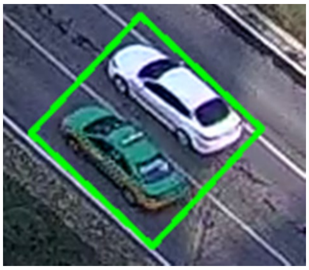


Fig. 13 The video image of the occlusion occurring between vehicles

1.6 Operations of segmentation and merging

There are two common cases of occlusion in vehicle detection: (1) occlusion between vehicles; (2) occlusion between a vehicle and a roadside obstruction. These two cases correspond to the segmentation and the merging operations respectively.

In the first case, when there is occlusion between vehicles (that is, one vehicle is too close to another, resulting in the two vehicles connected), it is shown in the binary image that multiple vehicles are connected together, resulting in only one target detected (as shown in Figs. 12, 13). This case occurs when a detection value contains multiple prediction values (as shown in Fig. 14). It is assumed that the detection value is accurate when the vehicle enters the monitoring range. It means one detection value accurately includes one vehicle. When the first case occurs, several prediction values should be used to divide the detection value into several detection values by using the prediction values, so that only one vehicle is included in each detection value. Then the problem of occlusion between vehicles can be solved. The segmentation method is to delete the wrong detection value and regard the prediction values as the new detection values.

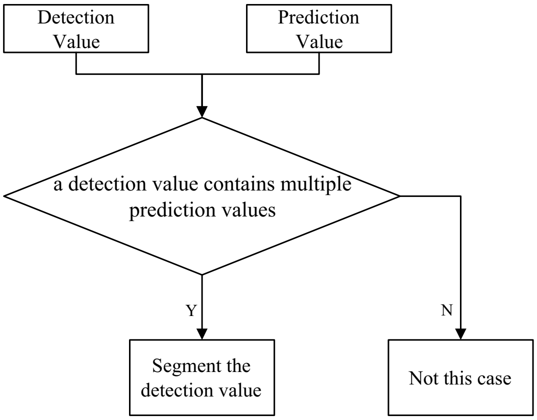

Detection Value
Prediction Value
a detection value contains multiple prediction values
Y
N
Segment the detection value
Not this case

Fig. 14 Segmentation operation flow chart

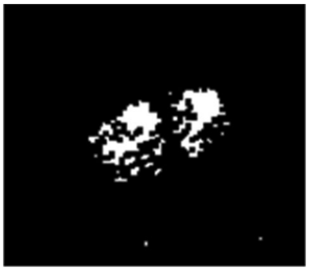


Fig. 15 The binary image of a vehicle occluded by a roadside obstruction

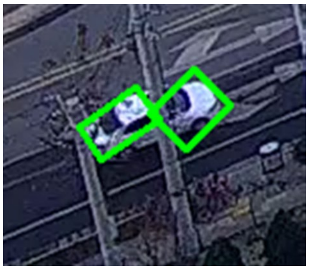


Fig. 16 The video image of a vehicle occluded by a roadside obstruction

In the second case, when the vehicle is occluded by a roadside obstruction (such as power poles, trees, etc.), it is shown in the binary image that the vehicle is divided into several parts by the obstruction, resulting in multiple targets detected (as shown in Figs. 15, 16). This case occurs when a prediction value contains multiple detection values (as shown in Fig. 17). When the second case occurs, the prediction value should be used to combine several detection values into one detection value, so that the detection value can completely contain a vehicle. Then the problem of occlusion between a vehicle and a roadside obstruction can be solved. The merging method is to delete the wrong detection values and regard the prediction value as the new detection value.

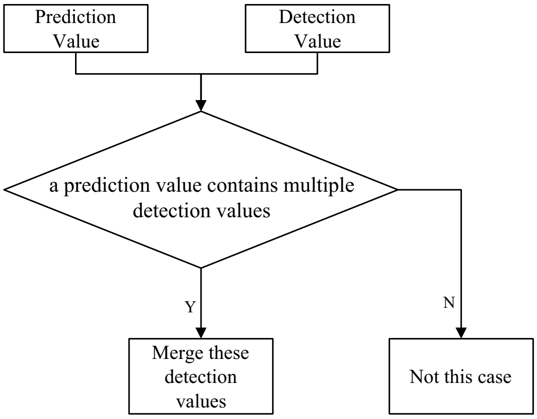

Prediction Value
Detection Value
a prediction value contains multiple detection values
Y
N
Merge these detection values
Not this case

Fig. 17 Merging operation flow chart

1.7 Adaptive adjustment of Kalman filter parameters

Consider a linear model of vehicle moving whose state transfer equation is:

$$x^+ = Ax + Bw \tag{3}$$

The measurement equation is:

$$y = Cx + v \tag{4}$$

Here, $x = (p_x, p_y, v_x, v_y)^T \in \mathbb{R}^4$ indicates the true vehicle position (groundtruth, p_x and p_y indicate the true coordinate of x-axis and y-axis respectively, v_x and v_y indicate the true velocity of x-axis and y-axis respectively). $x^+ \in \mathbb{R}^4$ indicates the groundtruth of the next time. $y = (\tilde{p}_x, \tilde{p}_y)^T \in \mathbb{R}^2$ indicates the detected vehicle position in the video (detection, $\tilde{p}_x$ and $\tilde{p}_y$ indicate the detected coordinate of x-axis and y-axis respectively). $A \in \mathbb{R}^{4\times 4}$ indicates state transition matrix, $B \in \mathbb{R}^{4\times 4}$ indicates process noise coupling matrix, and $C \in \mathbb{R}^{2\times 4}$ indicates measurement matrix. $w \in \mathbb{R}^4$ expresses process noise with $w \sim \mathcal{N}(0, Q)$, and $v \in \mathbb{R}^2$ is measurement noise with $v \sim \mathcal{N}(0, R)$. $Q \in \mathbb{R}^{4\times 4}$ indicates process noise covariance matrix, and $R \in \mathbb{R}^{2\times 2}$ indicates measurement noise covariance matrix.

In the model (3) and (4), $A \in \mathbb{R}^{4\times 4}$, $B \in \mathbb{R}^{4\times 4}$ and $C \in \mathbb{R}^{2\times 4}$ are decided by the system and always definite. $Q \in \mathbb{R}^{4\times 4}$ and $R \in \mathbb{R}^{2\times 2}$ cannot be obtained directly, so they are always set definite values which can make better effect of experiments. In our experiments, the matrices have the following forms

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$Q = \mathrm{diag}(10^{-7}, 10^{-7}, 10^{-7}, 10^{-7}), \quad R = \mathrm{diag}(10^{-4}, 10^{-4}).$$

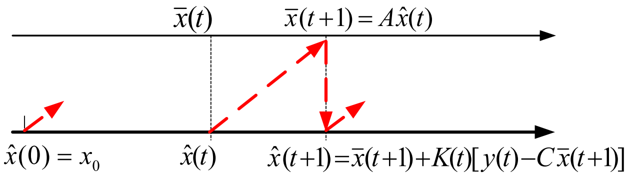

$\bar{x}(t)$
$\bar{x}(t+1) = A\hat{x}(t)$
$\hat{x}(0) = x_0$
$\hat{x}(t)$
$\hat{x}(t+1)=\bar{x}(t+1)+K(t)[y(t)-C\bar{x}(t+1)]$

Fig. 18 Evolution of Kalman filter

The model of Kalman filter consists of two parts. The first part is the time update equation:

$$\begin{aligned} \bar{x}^+ &= A\hat{x}, & \hat{x}(0) &= x_0 \\ \bar{P}^+ &= APA^T + BQB^T, & P(0) &= P_0 \end{aligned} \tag{5}$$

The second part is the measurement update equation:

$$\begin{aligned} \hat{x} &= \bar{x} + K(y - C\bar{x}) \\ K &= \bar{P}C^T(C\bar{P}C^T + R)^{-1} \\ P &= (I - KC)\bar{P} \end{aligned} \tag{6}$$

Here, $\bar{x} \in \mathbb{R}^4$ indicates prior state estimation of current time t, $\hat{x} \in \mathbb{R}^4$ indicates posterior state estimation of time t, $\bar{P} \in \mathbb{R}^{4\times 4}$ indicates prior estimation error covariance matrix, $P \in \mathbb{R}^{n\times n}$ indicates posterior estimation error covariance matrix, and $K \in \mathbb{R}^{n\times p}$ indicates Kalman filter gain matrix. In addition, the superscript $-$ indicates prior estimate and $\wedge$ indicates posterior estimate. Superscript $+$ indicates one step forward operator (the next time), and subscript 0 indicates the initial value, i.e., $x_0 = (p_x^0, p_y^0, 0, 0)^T$ (p_x^0 and p_y^0 indicate the position of a vehicle first detected), $P_0 = \mathrm{diag}(1, 1, 1, 1)$.

The evolution process of Kalman filter is shown in Fig. 18.

In the model of Kalman filter, the process noise covariance Q reflects the degree of trust in the prediction value. The smaller Q, the higher degree of trust in the prediction value. The measurement noise covariance matrix R reflects the degree of trust in the measurement value. The smaller R, the higher degree of trust in the measurement value.

To reduce the influence of detection value error, adaptive adjustment of noise covariance matrix is used. When occlusion occurs, the algorithm considers that the reliability of the detection value is low. So R should be increased and Q should be decreased. Therefore, the influence of detection value error on the estimation value is reduced (Fig. 19).

2 Experimental results

This paper selects the high-point video of the intersection of Jinye Road and Zhangbaqi Road in Xi'an as experimental video data (as shown in Fig. 20), which includes a large number of occlusions of all two cases. The experimental video has a scale of 1920*1080 and a frame rate of 25.

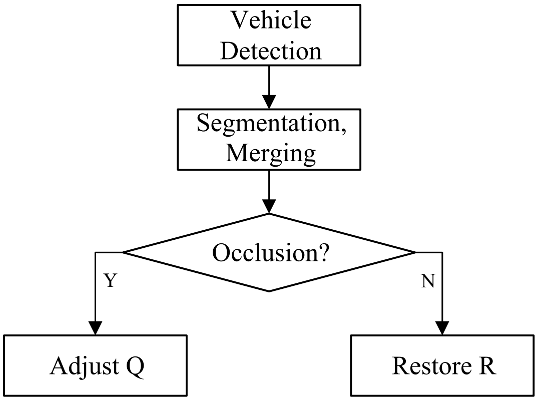


Fig. 19 The block diagram of adaptive adjustment of Kalman filter parameter

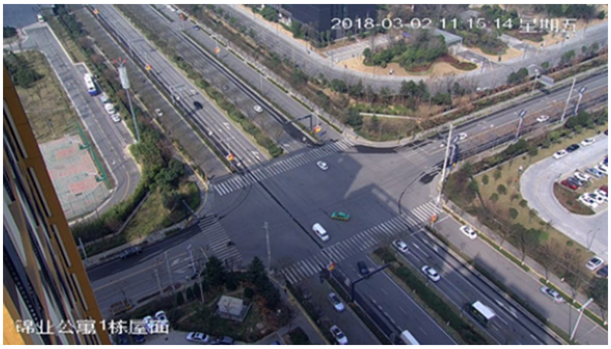


Fig. 20 The high-point video of the selected intersection

Figure 21 shows the processed results of part video sequence. The columns 1 to 3 indicate the processed results of frames 584, 690 and 1030 in different stages. Line 1 indicates the background difference image; Line 2 indicates the binary image; Line 3 indicates the image of global morphology operation; Line 4 indicates the image of local morphology operation. Line 5 indicates the detection result. Line 6 indicates the tracking result. In the frame 584, several vehicles are clustered in a small area that is prone to occlusion between vehicles. From lines 1–4 of the images in the first column, it can be seen that the contours of the vehicles are more and more clear and the boundaries between the vehicles are more and more obvious after the processing of the algorithm. In the frames 690 and 1030, the vehicle is divided into two parts by a roadside obstruction. After the merging operation, the final tracking result is still one target.

To test the generalization ability of the proposed algorithm, we also test the algorithm at another intersection which is shown at Fig. 22. The order of the images is left to right, top to bottom. The first is the result of detection. The second is the background difference image. The third is the binary image. The fourth is the image of global morphology operation. The fifth is the image of local morphology operation.

The vehicle trajectories extracted by the algorithm in this paper are compared with that of manual calibration to verify the effectiveness of the algorithm. To objectively describe the performance of the algorithm, coverage rate is defined to describe the accuracy of the vehicle trajectories. If the overlap rate of corresponding vehicle bounding box between the tracked vehicle trajectory and the manually calibrated vehicle trajectory is greater than 90% (taking the tracked vehicle trajectory as the benchmark), the sample point is considered successful in tracking, and otherwise the tracking fails. The coverage rate of each trajectory is the ratio of the number of sample points tracked successfully to the total number of sample points. Figure 23 shows the coverage rates of 20 vehicle trajectories whose average coverage rate is 95%. The blue circle represents the proposed method, the red star represents the traditional background difference method (BDM for short) and the green diamond represents the result of YOLO v3 which uses deep neural networks. For the proposed method, the tracking effect of most vehicles is satisfactory, but a small number of vehicles are lost due to the long time of continuous false rejection in the tracking process. For the BDM, most vehicles' tracking effect is terrible. The reason is that occlusion occurs frequently in the scenario and the BDM lacks of the operations in the case of occlusion. So the trajectories are always incomplete when using the BDM. And another reason is the minimum external rectangle parallel to the coordinate axis leads to the low overlap rate which cannot always reach 90%. So in the occlusion scenario, the proposed method can perform better than the BDM. For the YOLO v3, overall, the performance is the best. But in some cases, such as the 2-th and 15-th vehicles, its performance is not as good as the proposed algorithm. The reason is that deep learning may miss the vehicles when their appearance changes dramatically. However, the background difference algorithm does not have this problem bacause this algorithm don't care about the appearance of targets.

The computational complexity of BDM is obviously lower than that of the proposed algorithm, but BDM's performance is too bad. It is difficult to analyze the computational complexity of deep neural networks. But we can use FPS to evaluate the computational complexity of the algorithms. On the same computing platform, the FPS for YOLO v3 is about 16. And if we don't draw all the auxiliary lines which will spend a lots of time, the FPS for the proposed algorithm, is about 13–28. As the computational complexity of this algorithm is closely related to the number of vehicles, FPS is a range.

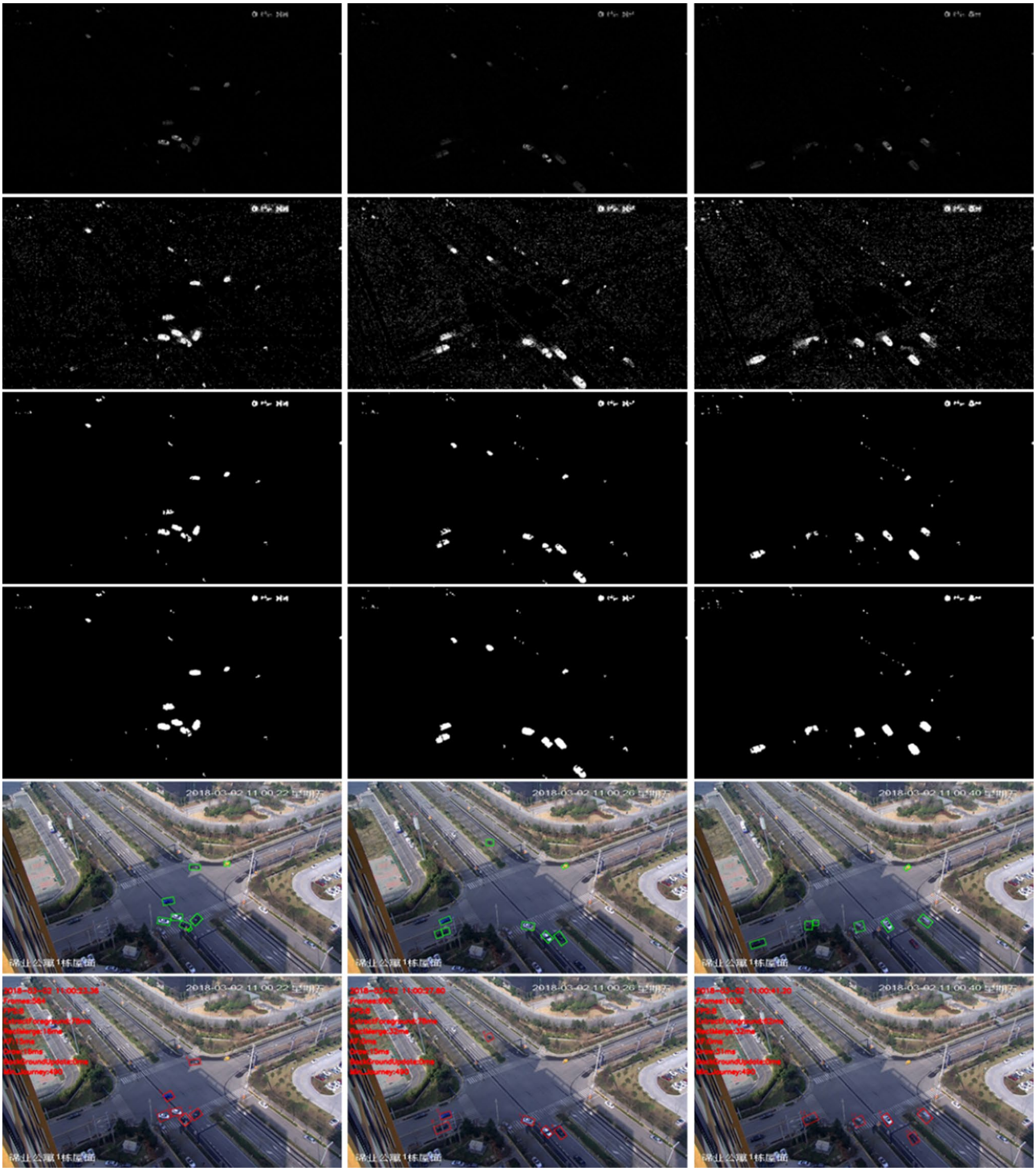


Fig. 21 Results of the part video sequence

3 Conclusion

This paper proposes a vehicle tracking method fusing the prior information of Kalman filter to solve the problem of vehicle tracking under occlusion. The method is improved in the aspects of background update, morphological operation, occlusion judgment and vehicle description. The experimental results show that the proposed tracking algorithm can effectively solve the problem of tracking in the case of occlusion between

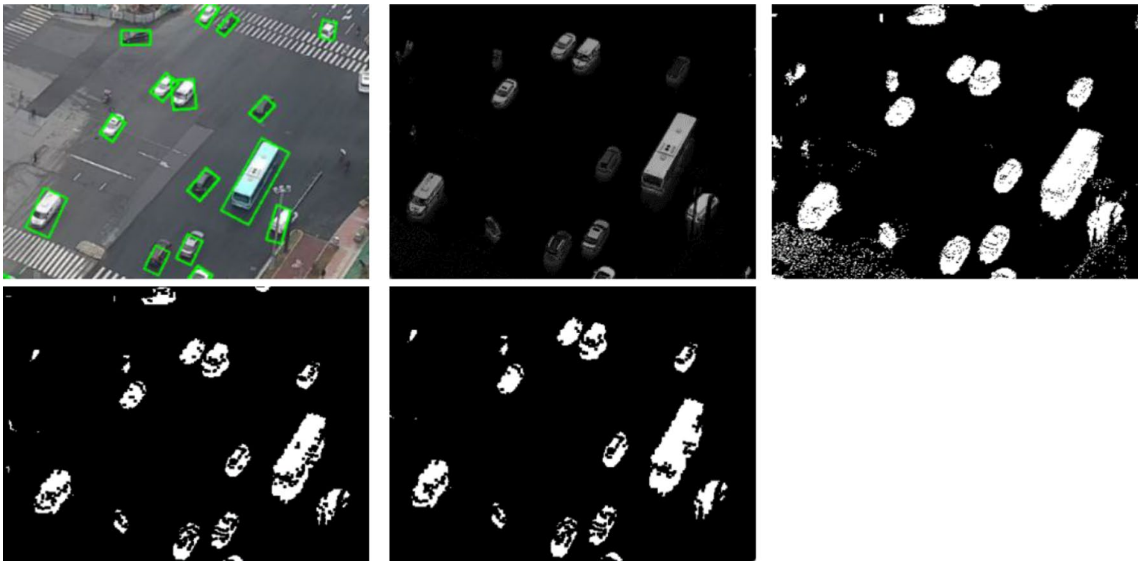


Fig. 22 The results of another intersection

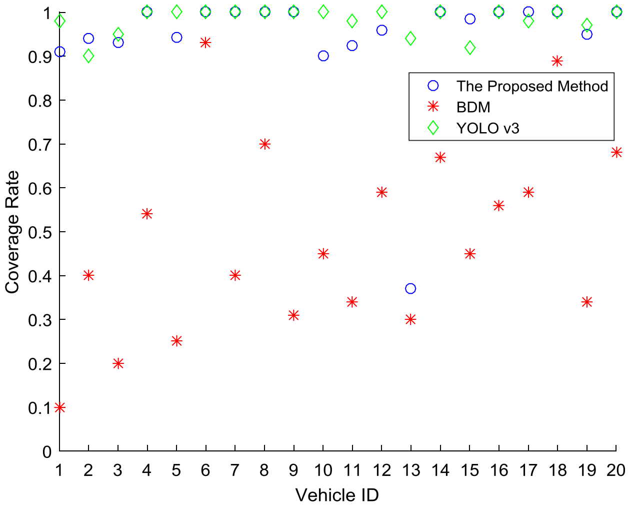


Fig. 23 The coverage rate of vehicle trajectories

vehicles and occlusion between a vehicle and a roadside obstruction.

The proposed algorithm can be further improved in two aspects. The algorithm needs to input relatively accurate background images during initialization. If the initial background image is not accurate enough, it will seriously affect background update and the accuracy of tracking. In addition, if the algorithm lose the target continuously for too long, it will result in the loss of the target even if the target is subsequently detected again.

Funding This research is supported by the National Natural Science Foundation of China (No. 61573030), and the National Key R&D Program of China (No. 2017YFC0803906) under the project (No. 2017YFC0803900).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Chen X, Zhu M, Liu C (2009) Review of tracking algorithms under occlusions. Chin J Opt Appl Opt 2(5):388–394
2. Wang C, Liu H, Zhang H, Hu M, Li X (2017) Anti-occluding tracking algorithm based on grey prediction and mean-shift. Control Eng China 24(7):1323–1328
3. Yan J, Wu M, Chen S, Zhang Q (2010) Anti-occlusion tracking algorithm based on mean Shift and fragments. Opt Precis Eng 18(6):1413–1419
4. Bao X, Zhan X, Wang Q, Hu L, Gui J (2018) Anti occlusion target tracking algorithm based on KCF and SIFT feature. Comput Meas Control 26(5):148–152
5. Liang G, Niu W, Li B, Li Q, Zhao P (2018) Structural sparse tracking method based on importance weighting. Chin J Sens Actuators 31(1):61–67
6. Ghasemi A, Safabakhsh R (2012) A real-time multiple vehicle classification and tracking system with occlusion handling. In: 2012 IEEE 8th international conference on intelligent computer communication and processing, Cluj-Napoca, pp 109–115
7. Wu B, Kao C, Jen C, Li Y, Chen Y, Juang J (2014) A relative-discriminative-histogram-of-oriented-gradients-based particle filter approach to vehicle occlusion handling and tracking. IEEE Trans Ind Electron 61(8):4228–4237
8. Krieger E, Sidike P, Aspiras T, Asari VK (2015) Vehicle tracking under occlusion conditions using directional ringlet intensity feature transform. IN: 2015 national aerospace and electronics conference (NAECON), Dayton, OH, pp 70–74

9. Qing M, Hoang V, Jo K (2012) Localization and tracking of same color vehicle under occlusion problem. In: 2012 9th France-Japan & 7th Europe-Asia Congress on Mechatronics (MECATRONICS)/13th international workshop on research and education in mechatronics (REM), Paris, pp 245–250
10. Qin X, Dai S, Li F (2017) Target tracking algorithm based on improved kernel correlation filter. Meas Control Technol 36(10):3–6
11. Aeschliman C, Park J, Kak AC (2014) Tracking vehicles through shadows and occlusions in wide-area aerial video. IEEE Trans Aerosp Electron Syst 50(1):429–444
12. Heidari V, Ahmadzadeh MR (2013) A method for vehicle classification and resolving vehicle occlusion in traffic images. In: 2013 first Iranian conference on pattern recognition and image analysis (PRIA), Birjand, 2013, pp 1–6
13. Phan HN, Pham LH, Tran DN, Ha SV (2017) Occlusion vehicle detection algorithm in crowded scene for Traffic Surveillance System. In: 2017 international conference on system science and engineering (ICSSE), Ho Chi Minh City, pp 215–220
14. Valiere P, Khoudour L, Crouzil A, Cong DNT (2015) Robust vehicle counting with severe shadows and occlusions. In: 2015 international conference on image processing theory, tools and applications (IPTA), Orleans, 2015, pp 99–104
15. Chang J, Wang L, Meng G, Xiang S, Pan C (2018) Vision-based occlusion handling and vehicle classification for traffic surveillance systems. IEEE Intell Transp Syst Mag 10(2):80–92

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.