Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 15, no. 6, June 2019, pp. 1620-1630
DOI: 10.23940/ijpe.19.06.p13.16201630

Anti-Occlusion Moving Target Tracking Method

Hongan Li[a,*], Zhuoming Du[b], Zhanli Li[a], Shuai Hao[c], and Jiaying Chen[a]

[a]*College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an, 710054, China*
[b]*School of Computer Engineering, Jiangsu University of Technology, Changzhou, 213001, China*
[c]*College of Electric and Control Engineering, Xi'an University of Science and Technology, Xi'an, 710054, China*

Abstract

In the artificial intelligence field, using computer vision to track an object is an important research topic. Especially when the target reappears after being occluded for a while, it is hard to precisely track the moving target again. Therefore, this paper proposes an anti-occlusion target tracking strategy that can overcome the occluded problem. Firstly, to make the target clearer, we design a moving target detection method using the Gaussian mixture background subtraction method based on the wavelet transform, which removes the high-frequency noise of video images. Then, in the tracking process, altered strategies are taken to cope with different occlusion situations, which include three cases: no occlusion, partial occlusion, and severe occlusion. For the first two cases, we use the distance-based Kalman filter method to track the moving target. For the third case, we designed a method that combines the Camshift method with the distance-based Kalman filter method to track moving targets, which is more efficient than only using the distance-based Kalman filter method. According to one of the cases, our program automatically selects the corresponding method. Experimental results show that our strategy can track moving targets accurately whether targets are in occlusion situation or not.

Keywords: moving target tracking; occlusion; wavelet transform; Camshift; distance-based Kalman filter

(Submitted on March 20, 2019; Revised on April 10, 2019; Accepted on June 2, 2019)

1. Introduction

Artificial intelligence (AI) techniques have become an essential part of the technology industry, helping to solve many challenging problems in computer science. The use of object tracking (person, face, hand, car) has become increasingly important in automobile driver assistance, vehicle navigation, robotics, human-computer interaction, video surveillance, biometrics, video games, industrial automation, and security [1-2]. Therefore, moving target tracking technology based on video sequences has become a significant research topic in the AI field. Moving target tracking is a process that automatically identifies a target, determines the target location, and automatically tracks the target. When the target appears again after being occluded, it should be re-identified as the original target. Occlusion has become a difficult problem in the target tracking process [3].

Currently, various methods are used to solve the occlusion problem in the target tracking process. In target tracking methods based on feature matching, the feature of the target may be its speed, centroid, contour, corner, color, and so on [4-5]. Jia et al. [6] combined the Kalman filter [7] with the area characteristic of the target to track the moving target, improving the anti-occlusion performance. Tao et al. associated the Meanshift [8] with Harris corner points to track the target [9]. These methods can accurately a track rigid target. However, when the target is occluded for a long time or transforms, it is hard to ensure the accuracy of tracking results. There exists another kind of target tracking method that is based on blocking. At first, the target is divided into several sections, and each section is tracked. Then, the tracking results of each section are analyzed comprehensively, and the optimal section is selected to locate the target [10-13]. These methods reduce the computing time and improve the real-time performance. However, the disadvantage is that the accuracy of tracking cannot be guaranteed when the target is completely occluded by other objects in the video. Some target tracking methods are based on prediction. In Zhou et al.'s work [14], a curve fitting method was used to predict the target position in some complex scenes. Liu et al. [15] combined moving target prediction with the multi-template matching method. The

* Corresponding author.
E-mail address: an6860@126.com

Kalman filter model was used to track the target in [16-18]. These methods can track the object under partial or no occlusion and short time occlusion, but they do not effectively cope with the problems of complete or long time occlusion. Overall, the occlusion problem is difficult to solve in the target tracking field.

This paper first proposes a detection method of the Gaussian mixture background model based on the wavelet transform method, which removes the high-frequency noise in the video to extract the moving target. According to the occlusion degree, which is determined by the change rate of the target area between adjacent frames, the occlusion can be divided into three levels: no occlusion, partial occlusion, and severe occlusion. According to different occlusion levels, we adopt different methods automatically. In the case of no or partial occlusion, this paper adopts the distance-based Kalman filter method to track the target. The distance characteristic is used to determine and update the observation data, and then the location of the target can be confirmed. In the case of severe occlusion, we designed a method that combines the Camshift algorithm [19] with the distance-based Kalman filter to track the moving target. Using the above strategy to experiment, we achieve satisfactory results.

2. Target Extraction

Target extraction is an essential processing component for medical video applications. To extract a clear target from the background, we designed a target extraction method that is based on the wavelet transform and the Gaussian mixture background model. The target detection method based on the Gaussian mixture background model can resist some tiny changes in the video images, but it is the relatively sensitive to the changes of light, so it is not stable. In addition, the wavelet transform can remove noise in the image and decrease the data quantity of the image, so it could reduce the impact of tiny change and improve the computational efficiency. In view of the advantages of the two methods mentioned above, a Gaussian mixture background model based on the wavelet transform is designed to detect the moving target, and it builds and updates the background model after the wavelet transform is carried out on the image. Finally, we use the background subtraction method to extract the moving target [20].

2.1. The Gaussian Mixture Background Model

In the Gaussian mixture background models, the pixel in the time series is treated as independent to all other pixels and is modeled using a mixture of Gaussians [21]. The per-pixel models are updated as new observations are obtained, with older observations losing influence over time. At each time step, a subset of the Gaussians in each per-pixel model is selected as representative of the scenic background. The color values of corresponding pixels in the video sequences are expressed as a sum of finite numbers of different weighted Gaussian functions. In these models, the color of pixel p is x_p in the *T*-frame image, and the probability is described as shown in Equations (1) and (2).

$$P\left(x_p\right)=\sum_{i=1}^{K} w^i{}_{p,t}\times\eta\left(x_p,\mu^i{}_{p,t},\sum{}^i{}_{p,t}\right) \tag{1}$$

$$\eta\left(x_p,\mu^i_{p,t},\sum{}^i{}_{p,t}\right)=\frac{1}{(2\pi)^{\frac{d}{2}}\left|\sum{}^i{}_{p,t}\right|^{\frac{1}{2}}}\exp\left[-\frac{1}{2}\left(x_p-\mu^i_{p,t}\right)^T\left(\sum{}^i{}_{p,t}\right)^{-1}\left(x_p-\mu^i_{p,t}\right)\right] \tag{2}$$

Where d is the spatial dimension of $P_{(k+1)/k}$, η is a Gaussian density function, $w^i_{p,t}$ (described as Equation (3)) is the i^{th} Gaussian component weight of the p^{th} pixel at the moment t in the Gaussian mixture model, and $\mu^i_{p,t}$ is a mean value.

$$\sum\nolimits_{i=1}^{K} w^i_{p,t+1}=1 \tag{3}$$

For the RGB color space, the three components can be reconsidered mutually independent, so the covariance matrix is defined as Equation (4).

$$\sum{}^i{}_{p,t}=\left(\sigma^i_{p,t}\right)^2 I \tag{4}$$

Where $\sigma^i_{p,t}$ is a standard deviation and I is a unit matrix.

2.2. Algorithm Steps of the Gaussian Mixture Model based on the Wavelet Transform

Step 1 Capture the first frame and initialize the Gaussian mixture background model. In the image, the color of each pixel is set to a value in the range of 0 to 255, all K variances of the Gaussian function are set to be larger variances σ_{init}^2, and the initial weight coefficient of each Gaussian function is $w_{init} = 1/K$. The value of each pixel in the first-frame image is used to initialize the mean of K Gaussian distributions of the Gaussian mixture model.

Step 2 Read the next frame, and an n-layer wavelet transform will be operated on the image. Then, a threshold is set to filter the high-frequency data and leave the low-frequency data. The process of the wavelet transform is divided into three parts: divide, forecast, and update, which are implemented as Equations (5)-(7).

$$F(S_n) = (S_{n-1}, d_{n-1}) \tag{5}$$

$$d_{n-1} = d_{n-1} - P(S_{n-1}) \tag{6}$$

$$S_{n-1} = S_{n-1} + U(d_{n-1}) \tag{7}$$

Where S_n is the original signal, S_{n-1} and d_{n-1} are the signals after division, $P(S_{n-1})$ is the forecast function, and $U(d_{n-1})$ is the update function.

Step 3 Sort the K Gaussian functions with $w_{p,t}^i / \sigma_{p,t}^i$ of each pixel and match the pixel x_p of the current image with K Gaussian functions in the Gaussian mixture background model, described as Equation (8).

$$\left| x_p - \mu_{p,t}^i \right| < \delta \sigma_{p,t}^i \tag{8}$$

Step 4 If the pixel x_p matches with the i^{th} Gaussian distribution, the parameters of the i^{th} Gaussian distribution will be updated by the color of pixel x_p, and the rest of the Gaussian functions will be kept invariant. The update equations are described as Equations (9)-(12).

$$w_{p,t+1}^i = (1-\alpha) w_{p,t}^i + \alpha \tag{9}$$

$$\mu_{p,t+1}^i = (1-\beta) \mu_{p,t}^i + \beta x_p \tag{10}$$

$$\left(\sigma_{p,t+1}^i\right)^2 = (1-\beta)\left(\sigma_{p,t}^i\right) + \beta\left(x_p - \mu_{p,t+1}^i\right)^{\mathrm{T}} \left(x_p - \mu_{p,t+1}^i\right) \tag{11}$$

$$\rho = \frac{\alpha}{w_{j,t}^i} \tag{12}$$

Where α and β are learning parameters of the model.

Step 5 If the pixel x_p fails to match with all K Gaussian distributions, several smaller-weight Gaussian functions in the model will be replaced by a new Gaussian function whose mean value is set at x_j. The standard deviation and weight are set at the original values σ_{init} and w_{init}. The standard deviations and the mean values of other Gaussian functions remain constant, and the weight is updated as Equation (13).

$$w_{p,t+1}^i = (1-\alpha) w_{p,t}^i \tag{13}$$

Step 6 Modify the weight of each Gaussian function after updating the current Gaussian mixture model. Then, identify the background pixels, described as Equation (14).

$$B_p = \min\left(\sum_{i=1}^{K} w_{p,t+1}^{i} > T\right) \tag{14}$$

Where T is a threshold that judges whether the pixel is a background pixel or not. In the case of keeping the camera unmoved, generally, the background of the video sequence does not change or changes only slightly.

Step 7 Extract the moving target using the background subtraction method, which is described as Equations (15) and (16).

$$D_t(x,y) = C_t(x,y) - B_t(x,y) \tag{15}$$

$$R_t(x,y) = \begin{cases} 255, & \text{if } D_t(x,y) > T \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

Where $C_t(x,y)$ represents the original image of the time t, $B_t(x,y)$ is the background image of the time t, and T is the threshold that identifies whether the pixel is a moving target or not.

3. The Distance-based Kalman Filter

The Kalman filter, also known as linear quadratic estimation (LQE), is an algorithm that uses a series of measurements observed over time. It contains noise (random variations) and other inaccuracies and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone [22-23]. In the process of target tracking, the Kalman filter typically works in a two-step process: prediction and updating. Firstly, the state vector of the last frame is used to predict the state vector of the current frame, and then the observed values are utilized to update the predictive values to get the current state vector of the object. The distance-based Kalman filter is a method that combines the distance between the same object of the last frame and the current frame with the Kalman filter to implement the moving object tracking. In this method, firstly, the moving objects are extracted by the object detection method and recorded as $O_1, O_2, O_3, \cdots, O_k$. Then, those objects are sorted by area size, and the first n objects are taken into a list of object tracking that is expressed as $S = \{O_1, O_2, O_3, O_4, \cdots, O_n\}$, where O is the moving object. Finally, the Kalman filter is established for each object in the tracking list and can be described as follows:

Prediction equations: Equations (17) and (18).

$$X'_{(k+1)/k} = A_{k+1,k} X_{k/k} \tag{17}$$

$$P_{(k+1)/k} = A_{k+1,k} P_{k/k} A_{k+1,k}^{\mathrm{T}} + Q_k \tag{18}$$

Updating equations: Equations (19)-(21).

$$K_{k+1} = P_{(k+1)/k} H_{k+1}^{\mathrm{T}} \left[H_{k+1} P_{(k+1)/k} H_{k+1}^{\mathrm{T}} + R_{k+1} \right]^{-1} \tag{19}$$

$$X_{(k+1)/(k+1)} = X'_{(k+1)/k} + K_{k+1} \left[Z_{k+1} - H_{k+1} X'_{(k+1)/k} \right] \tag{20}$$

$$P_{(k+1)/(k+1)} = \left[I - K_{k+1} H_{k+1} \right] P_{(k+1)/k} \tag{21}$$

Where $X'_{(k+1)/k}$ is an a priori estimate of the state vector at the time $k+1$ and $X_{k/k}$ is a posterior estimate of the state vector at the time k. At the same time, $P_{(k+1)/k}$ is an a priori estimate of error covariance at the time $k+1$, $P_{k/k}$ is a posterior estimate of error covariance at the time k, and A is a state transition matrix.

Next, to initialize the Kalman filter of each object, X_k and Z_k are chosen as a motion state vector and an observation vector respectively, described as Equations (22) and (23).

$$X_k = \left(x_k, y_k, S_k, \Delta x_k, \Delta y_k, \Delta S_k\right)^{\mathrm{T}} \tag{22}$$

$$Z_k = \left(x_k, y_k, S_k\right)^{\mathrm{T}} \tag{23}$$

Where $\left(x_k, y_k\right)$ is the centroid of the object at the moment K, S_k represents the area of the object at the time *k*, and $\left(\Delta x_k, \Delta y_k\right)$ represents the displacement of objects between the k th frame and the $\left(k+1\right)$ th frame in the x and *y* directions, respectively. Meanwhile, ΔS_k represents the area difference of objects between the k th frame and the $\left(k+1\right)$ th frame. When reading the video frames, it is obvious that the time interval between adjacent frames is relatively tiny, so the motion model of the object can be regarded as uniform motion.

The state transition matrix A is described as Equation (24).

$$A = \begin{bmatrix} 1 & 0 & 0 & t & 0 & 0 \\ 0 & 1 & 0 & 0 & t & 0 \\ 0 & 0 & 1 & 0 & 0 & t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

The observation matrix H is described as Equation (25).

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{25}$$

The initial error covariance matrix is defined as Equation (26).

$$P_0 = \left\{b_1, b_2, b_3, b_4, \cdots, b_m\right\} \tag{26}$$

When the next frame image is captured, all the foreground pixel blocks are extracted and put into a list of B, $B = \left\{b_1, b_2, b_3, b_4, \cdots, b_m\right\}$, where b is a pixel block. Generally, a pixel block is associated with a moving target, so a moving target only corresponds to a foreground pixel block. However, when the moving target is occluded by other objects, a foreground pixel block may correspond to multiple moving objects, so the number n of moving objects is always much more than or equal to the number m of foreground pixel blocks. That is to say, $m \le n$. Next, the distances between the target O in the k th frame and the pixel blocks b_i in the $\left(k+1\right)$ th frame are calculated. In accordance with the rules of the nearest neighbor, the nearest pixel block b_i is singled out as an object O^*, described as Equation (27).

$$O^* = \min\left\{D\left(O, b_i\right)\right\} \tag{27}$$

Where $O^* \in O_n$, $O_{n+1} = O^*$. Then, use the parameter of the object O_{n+1} to update the current parameter of the object O_n in the Kalman filter. This method takes full advantages of the space distribution of objects, avoiding biasing tracking caused by the noise. Therefore, the distance of the same object between adjacent frames is viewed as a characteristic to track objects accurately in the situation of partial occlusion.

4. Combination of the Distance-based Kalman Filter with Camshift

In this section, we design an anti-occlusion object tracking method that combines the Camshift algorithm with the distance-based Kalman filter. The Camshift algorithm is a modification of the Meanshift algorithm, which is a robust statistical method of finding the mode (top) of a probability distribution. It is a very fast and simple tracking method, because Camshift tracks the centre and size of the probability distribution of an object and is only as good as the probability distribution that is produced for the object. Typically, the probability distribution is derived from color via a histogram, although it could be produced from correlation and recognition scores or bolstered by frame differencing, motion detection schemes, joint probabilities of different colors/motions, etc. Moreover, the search strategy is reviewed as follows. Firstly, the detection method is applied to extract the moving object in the video sequence. Each moving object is measured to obtain its motion position parameters. Then, those motion parameters are used to initialize the Kalman filter algorithm to predict the next position of the moving object in the image. Next, according to the results of occlusion detection, the anti-occlusion target tracking algorithm can cope with two conditions using a different strategy: no or partial occlusion and severe occlusion. In the case of no or partial occlusion, the distance-based Kalman filter is carried out to track the moving object, and the position of the target is used to update the state of the Kalman filter. In the case of severe occlusion, we combine the iterative Camshift algorithm with the prediction mechanism of the distance-based Kalman filter to search for the moving object, and it is not necessary to update the state of the Kalman filter. Finally, the actual position of the object is located in the image. The overview of target tracking is shown in Figure 1.

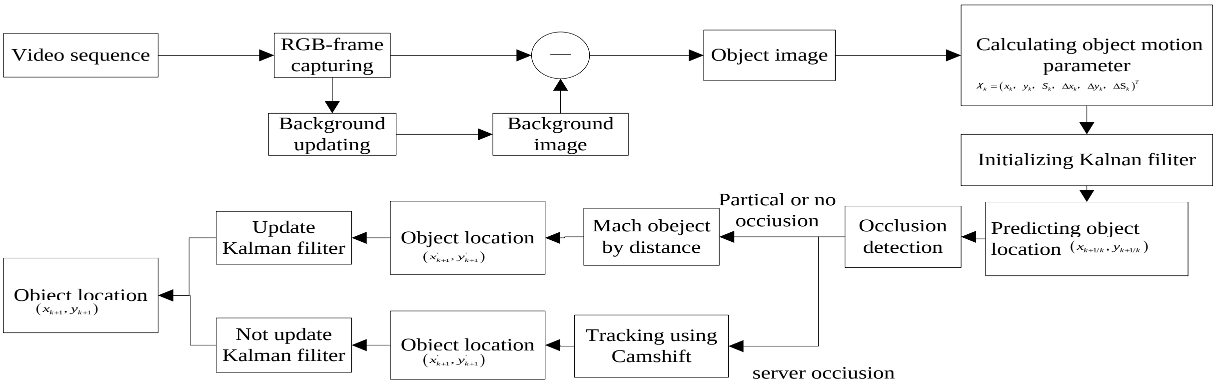


Figure 1. The overview of target tracking method

4.1. Occlusion Judgment

In the tracking process, the occlusion between the target and the background usually triggers some serious tracking errors. Therefore, it is vital to detect the occlusion and distinguish the level of occlusion. According to experiments, it was found that the area of an object has fine-grained changes in different frames. Moreover, the object area changes obviously when the occlusion occurs. Thus, we take advantage of area changes as a characteristic to estimate the level of occlusion. In this paper, we define the area of a moving object at times t and $t+1$ as S_t and S_{t+1}, respectively. The change rate F_s of an object area between the adjacent frames is defined as Equation (28).

$$F_s = \left| \frac{S_{t+1} - S_t}{S_t} \right| \tag{28}$$

Then, according to the change rate, we define three occlusion levels:

- $F_s < \alpha_1$ indicates no occlusion or the change rate of the object area is relatively small within an acceptable error range.
- $\alpha_1 < F_s < \alpha_2$ indicates partial occlusion.
- $F_s > \alpha_2$ indicates severe occlusion.

Where α_1 is a relatively low threshold value and α_2 is a relatively high threshold value. In our experiments $\alpha_1 = 5\%$ and $\alpha_2 = 60\%$.

4.2. Algorithm Steps

The points located in the region of the moving object are $\{x_i\}$, $i=1,2,3,\cdots,n$, where n is the total number of pixels of the moving object and x is the centroid of the object.

Step 1 Establish the Kalman filter, which is based on distance for each new object, and use the Kalman filter to predict the position x of the object.

Step 2 Calculate the change rate F_s of the object area and estimate the occlusion level of the object. If $F_s<\alpha_1$ or $\alpha_1<F_s<\alpha_2$, the distance characteristic will be implemented to obtain the position of the object. If the Camshift is initialized, proceed to Step 3. Otherwise, the probability distribution histogram $p_u(x)$ of the object is calculated to initialize the Camshift under no occlusion, described as Equation (29).

$$p_u(x)=C\sum_{i=1}^{n}K\left(\left\|\frac{x-x_i}{h}\right\|^2\right)\delta\left[b(x_i)-u\right],\ u=1,2,3,\cdots,m \tag{29}$$

Where C is the unit density, $K(x)$ is the kernel function, h is the step length during sampling pixels, m is the number of characteristics, $b(x_i)$ is the series of x_i, and $\delta(x)$ is the Kronecker function. If the eigenvalue of x_i is u, then $\delta(x)=1$, whereas $\delta(x)=0$.

Step 3 If $F_s>\alpha_2$, use the Kalman filter to predict the object position, and take the possible region in which the moving object may be located as the candidate object region. The centroid of the object is x_c. Then, the probability distribution histogram $q_u(x)$ of the candidate object region is calculated as Equation (30).

$$q_u(x_c)=C_h\sum_{i=1}^{n}K\left(\left\|\frac{x_c-x_i}{h}\right\|^2\right)\delta\left[b(x_i)-u\right],\ u=1,2,3,\cdots,m \tag{30}$$

Furthermore, the Bhattacharyya coefficient between the object and the candidate object region is defined as Equation (31).

$$\rho(x)=\sum_{u=1}^{m}\sqrt{p_u(x)q_u(x_c)} \tag{31}$$

Then, the centroid of the object is described as Equation (32).

$$x_1=\frac{\sum_{i=1}^{n}x_i w_i G\left(\left\|\frac{x_i-x}{h}\right\|^2\right)}{\sum_{i=1}^{n}w_i G\left(\left\|\frac{x_i-x}{h}\right\|^2\right)} \tag{32}$$

Where w_i is the weight of the i^{th} pixel.

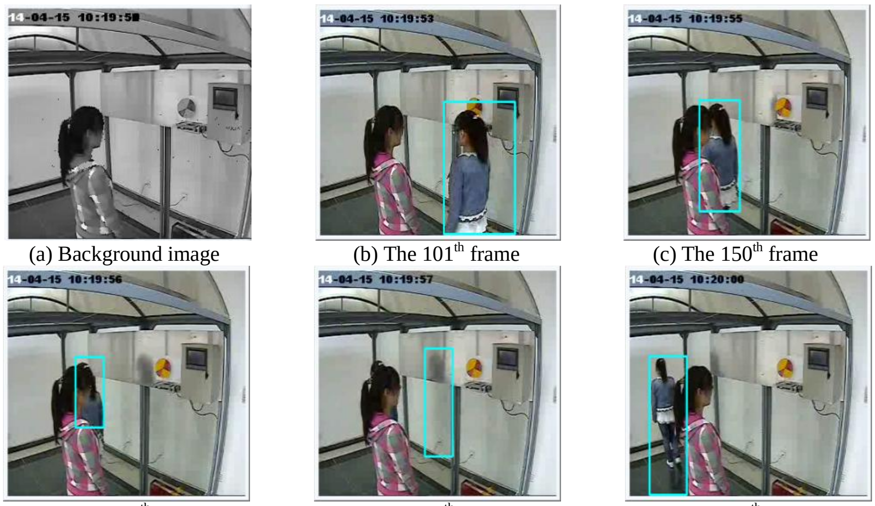
Step 4 If $x_1-x_0\le\alpha$, the iteration condition is met. Then, x_1 is determined as the target position, and the Kalman filter is updated. If the iteration condition is not matched, the Kalman filter does not need to be updated, and x_1 will be defined as the object position in the current frame. Return to Step 1 to process the next frame.

5. Experiment and Analysis

In order to verify the validity of our algorithm, we use the video frames recorded by a network camera in our laboratory. All the programs are operated by Windows7, Core i5 processors, 2.8GHz, 2G RAM, and VS2010. Three methods are respectively applied to track the target in the video frames: the distance-based Kalman filter, the Camshift method, and our anti-occlusion tracking method. The results of those three methods are shown in Figures 2, 3, and 4, respectively. Finally, by analyzing and studying those tracking results, it is obvious that our proposed method, which combines the distance-based Kalman filter with Camshift, is superior to the other two methods.

In these video frames, the pedestrian walks from one side to another side and is occluded by another pedestrian in the walking process. Figure 2 shows the tracking results of the distance-based Kalman filter in the case of occlusion. The blue box represents the tracking box that contains the target in the frame. The frame sequence consists of the 101th, 150th, 170th, 200th, and 260th frame.

Figure 2(a) shows the grey background image extracted by the method of the Gaussian mixture model based on wavelet transform. In Figures 2(b)-2(f), the Kalman filter method based on distance is used to track the moving target. In Figure 2(b), the target enters the scene of the experimental shed and can be tracked correctly in the case of no occlusion. In Figures 2(c) and 2(d), when the target is occluded partially, the tracking results are not affected since the main information of the target can still be detected in the image. However, when the target is severely occluded by other obstacles such as in Figure 2(e), most of the target information fails to be detected. At this moment, the Kalman filter begins to predict the next position of the target. As shown in Figure 2(e), the target is lost and the shadow in the wall is mistakenly regarded as the object to be tracked. In Figure 2(f), as the object leaves the occlusion region and reappears again in the image, the target will be tracked immediately. Experimental results show that the distance-based Kalman filter can deal with the partial occlusion problem. However, when the moving target is severely occluded, this method fails to track the moving target because most of the information of the target cannot be identified.



(a) Background image (b) The 101th frame (c) The 150th frame
(d) The 170th frame (e) The 200th frame (f) The 260th frame

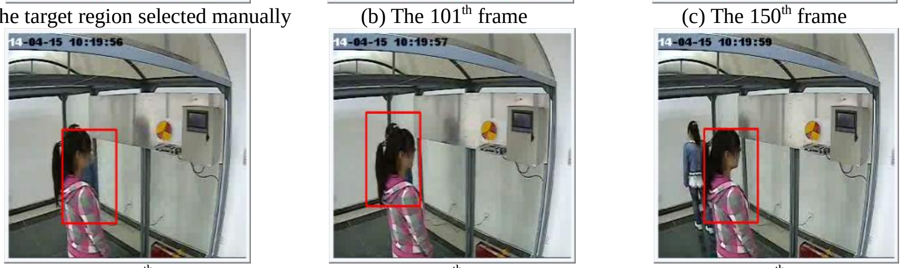
Figure 2. The tracking results of the distance-based Kalman filter

The tracking results of the Camshift method in the case of occlusion are shown in Figure 3, in which the red box is the tracking box. Firstly, as shown in Figure 3(a), we manually select the target region from the image sequence as a standard target template to match and identify the moving target. In Figures 3(b)-3(f), the Camshift method is applied to track the moving target.

In the tracking process, we can see that partial occlusion has almost no influence on the performance of Camshift. With the moving target being occluded seriously, only a few color characteristics can be detected in Figures 3(b)-3(d), but the target can still be tracked accurately. As the target is completely occluded in Figure 3(e), the target region could not be detected any more, while the tracking box converges on the occlusion object. Disappointingly, with the moving object appearing again in the image, the tracking box does not converge to the moving target object but remains in the occlusion object. The target is lost, as shown in Figure 3(f).

Figure 4 shows the tracking results of our anti-occlusion method that combines the distance-based Kalman filter with Camshift using the tracking strategy. Both the blue box and the red box are tracking boxes. In addition, the blue box represents the tracking box of the distance-based Kalman filter under no or partial occlusion, and the red box represents the tracking box of Camshift under severe occlusion.

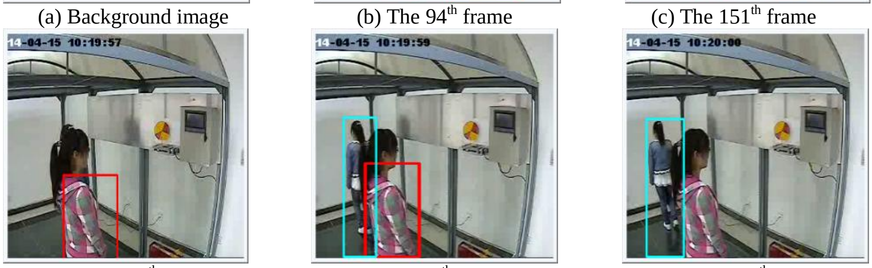
(a) The target region selected manually (b) The 101th frame (c) The 150th frame



(d) The 180th frame (e) The 200th frame (f) The 250th frame

Figure 3. The tracking results of the Camshift method

(a) Background image (b) The 94th frame (c) The 151th frame



(d) The 194th frame (e) The 250th frame (f) The 267th frame

Figure 4. The tracking results of our anti-occlusion tracking method

In Figure 4(d), when the target almost completely disappears behind the occlusion object, the Camshift method is employed to track the target. Because the color characteristics of the target cannot be detected, the moving object is considered behind the occlusion object. When a new object emerges in the scene, its color characteristics are used to confirm whether it is the occluded target or not, as shown in Figure 4(e). In the image, the red box indicates the position of the moving target. With the target completely reappearing in Figure 4(f), it is recognized as the original target before occlusion. At the same time, the tracking method is transformed from Camshift to the distance-based Kalman filter, and the blue box in the image shows the position of the moving target. At last, the experimental results indicate that our proposed method can correctly track the target in the three cases of no, partial, and severe occlusion. Moreover, when the target reappears again in the image, it can be re-identified accurately as the original target.

6. Conclusions

In this paper, to extract the clearer moving target from video frames, we proposed the Gaussian mixture background model based on the wavelet transform at the stage of target detection, which efficiently decreases the data quantity of the image and removes some high-frequency noise in the image. Then, to track the target, we proposed an anti-occlusion tracking method that combines the distance-based Kalman filter with Camshift. When the target is between two frames, the distance-based Kalman filter calculates the spatial distance that is characteristic to track the moving target. When the target is seriously or completely occluded, our designed method searches and tracks the moving occluded target. Finally, according to the experimental results, our designed tracking method is superior in terms of precision compared with the Kalman filter and the Camshift under the condition of severe and complete occlusion.

Acknowledgments

This work is partially supported by the Natural Science Basic Research Plan in Shaanxi Province of China (2019JM-162, 2019JM-020) and the Doctoral Research Startup Foundation of Xi'an University of Science and Technology (2019QDJ007). We declare that there is no conflict of interests regarding the publication of this article and would like to thank the anonymous reviewers for their valuable comments and suggestions.

References

1. H. A. Li, Z. M. Du, Z. L. Li, and B. S. Kang, "Research on Aerial Target Recognition and Tracking based on Grey and Textural Features," *Journal of Graphics*, Vol. 37, No. 2, pp. 224-229, 2016
2. H. Wei and H. Q. Guo, "A Target Tracking Algorithm for Medical Images via Sparse Decomposition and Support Vector Machine," *Research & Exploration in Laboratory*, 2013
3. M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, et al., "The Visual Object Tracking Vot2015 Challenge Results," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1-23, 2015
4. F. L. Chang, L. Ma, and Y. Z. Qiao, "Target Tracking under Occlusion based on Feature Correlation Matching," *Journal of Image and Graphics*, Vol. 11, No. 3, pp. 877-882, 2006
5. X. Yang, J. Liu, P. Y. Zhou, and D. Zhou, "Adaptive Particle Filter for Object Tracking based on Fusing Multiple Features," *Journal of Jilin University* (*Engineering and Technology Edition*), Vol. 45, No. 2, pp. 533-539, 2015
6. H. R. Jia, "Research of Target Tracking Algorithm under Occlusion based on Feature Correlation Matching," Northeast Normal University, 2009
7. R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, pp. 35-45, 1960
8. D. Comaniciu, V. Ramesh, and P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean Shift," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, IEEE, 2000
9. T. Li, "Research on Video Segmentation under Occlusion," Jiangnan University, 2012
10. J. Wang, "Visual Tracking based on Sub-Space Motion Model," Shangdong University, 2013
11. B. Xu, D. Kai, and T. Hao, "Object Tracking based on Improved Mean Shift and SURF," *Computer Engineering and Applications*, Vol. 49, No. 21, pp. 133-137, 2013
12. Q. C. Dai, X. Y. Feng, and J. Liu, "Tracking Method for Object of Partial Occlusion based on Combination of Blob Modelling and Mean-Shif," *Computer Engineering and Applications*, Vol. 18, No. 47, pp. 183-185, 2011
13. W. S. Yu, X. H. Tian, Z. Q. Hou, and Y. F. Zha, "Online Visual Tracking based on Local Patch Learning," *Acta Electronica Sinica*, Vol. 43, No. 1, pp. 74-78, 2015
14. Z. Zhou, "Research on Anti-Occluding Target Tracking Algorithm in Complex Background," *Aero Weaponry*, No. 6, pp. 36-39, 2007
15. Y. Liu, Y. F. Zhang, and Y. F. Dong, "Anti-Occlusion Algorithm of Tracking Moving Object in Clutter Background," *Chinese Journal of Liquid Crysls and Displays*, Vol. 25, No. 6, pp. 890-895, 2010
16. T. W. Xia and X. Hou, "Robot Moving Target Tracking Algorithm based on Adaptive Kalman Filter," *Computer Measurement & Control*, Vol. 23, No. 1, pp. 173-175, 2015
17. W. J. Liu and Y. J. Zhang, "Edge-Color-Histogram and Kalman Filter-based Real-Time Object Tracking," *Journal of Tsinghua University* (*Science and Technology*), Vol. 48, No. 7, pp. 1104-1107, 2008
18. X. F. Wang, "Vehicle Tracking Algorithm under Occlusion in Video," Shandong Normal University, 2013
19. G. R. Bradski, "Real Time Face and Object Tracking as a Component of a Perceptual User Interface," in *Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision*, pp. 214-219, 1998
20. W. Wang and Z. H. Meng, "An Object Tracking Algorithm based on Improved Camshift," *Information Technology*, Vol. 1, No. 23, pp. 85-88, 2015
21. J. X. Huang, W. L. Wu, C. J. Long, and M. J. Zhang, "Study of Moving Object Detection in Video and Its Application," *Computer Technology and Development*, Vol. 24, No. 3, pp. 15-18, 2014
22. Q. M. Shi, "Research of Object Tracking based on Camshift," Anhui University, 2013
23. P. Vasilis and A. Argyros, "Multiple Objects Tracking in the Presence of Long-Term Occlusions," *Computer Vision & Image Understanding*, Vol. 114, No. 7, pp. 835-846, 2010

Hongan Li received his Ph.D. in computer science from Northwest University, Xi'an, China in 2014. He is currently an associate professor at Xi'an University of Science and Technology, Xi'an, China. His research interests include image processing, computational photography, and intelligent information processing.

Zhuoming Du received his Ph.D. in computer science from Northwest University, Xi'an, China in 2012. He is currently a postdoctoral researcher in the School of Mathematical Sciences at Nanjing Normal University, Nanjing, China. His research interests include computational photography, deep learning, and optimization theory.

Zhanli Li received his Ph.D. from the School of Mechanical Engineering at Xi'an Jiaotong University, Xi'an, China in 1997.

He is currently a professor in the College of Computer Science and Technology at Xi'an University of Science and Technology, Xi'an, China. His research interests include intelligent video processing and intelligent information processing.

Shuai Hao received his Ph.D. from the School of Automation at Northwestern Polytechnical University, Xi'an, China in 2014. He is currently a postdoctoral researcher at Xi'an University of Science and Technology, Xi'an, China. His research interests include intelligent video processing and intelligent information processing.

Jiaying Chen is a master's student in the College of Computer Science and Technology at Xi'an University of Science and Technology, Xi'an, China. Her research interest is video target tracking.