IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 26, 2021, accepted December 17, 2021, date of publication December 23, 2021,
date of current version January 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138080

Research on Anomaly Suppression Correlation Filtering Algorithm

HONGGE REN[1], YIN CUI[2], AND TAO SHI[3]

[1]School of Control and Mechanical Engineering, Tianjin Chengjian University, Tianjin 300384, China
[2]College of Electrical Engineering, North China University of Science and Technology, Tangshan 063210, China
[3]School of Electrical Engineering and Automation, Tianjin University of Technology, Tianjin 300384, China

Corresponding author: Tao Shi (st99@email.tjut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61203343, and in part by the Natural Science Foundation of Hebei Province under Grant F2018209289.

ABSTRACT Aiming at the problems of target occlusion, illumination change, and fast motion in target tracking, an anomaly suppression correlation filtering algorithm is proposed. The histogram of oriented gradient (HOG) feature and color name (CN) feature are extracted to construct the target appearance model, and an abnormal suppression correlation filter is constructed to suppress sudden changes in the filter response graph. The alternating direction method of multipliers (ADMM) method is used to speed up the calculation of the filter. A scale filter is added in the article to solve the problem of inaccurate tracking caused by a single scale. In the model update process, the three-layer rotating circle memory model is used to update, which improves the tracking ability of the algorithm. It is better to update the target model in the scene where the target is occluded. The proposed algorithm and BACF, KCC, KCF, MKCFup, ARCF are tested in the OTB50, OTB100, UAV123, TC128 experimental data sets. The results show that the visual tracking algorithm proposed in the article has a high success rate and accuracy, and has certain research value.

INDEX TERMS Computer vision, target tracking, correlation filtering, abnormal suppression, ADMM method, memory model.

I. INTRODUCTION

Visual tracking occurs in consecutive frames in a sequence of video frames to track manually or automatically calibrated targets. In the past and recent years, researchers had conducted a lot of research, which has been widely used in behavior recognition [1]–[3], human-computer interaction [4], [5], and traffic detection [6]–[9]. Because of many challenges in real life, such as lighting changes, complex backgrounds, etc., and lack of training data, target tracking is still challenging.

Traditional correlation filter algorithms have always had the problem of boundary effects since they appeared [10]–[12], some scholars use expanding the target search area to solve this problem [13]–[15]. However, they introduced more contextual information while expanding the target search area and the correlation filter would learn from contextual information more easily. And the challenges faced by the current method are eleven common challenges in the field of target tracking. The challenges can cause changes in the appearance of the target, which can significantly reduce the success rate of tracking. They are illumination variation (IV), scale variation (SV), background clutters (BC), out-of-plane rotation (OPR), motion blur (MB), occlusion (OCC), in-plane rotation (IPR), deformation (DEF), out-of-view (OV), fast motion (FM), low resolution (LR) respectively. Therefore, we design the anomaly suppression correlation filter algorithm to solve the above problems.

Correlation filtering algorithms are favored by most researchers because of their ability to be calculated in the frequency domain and speed advantages, and they are the most widely used in the field of target tracking. The correlation filter can convert the problem into a linear system function equivalently, and the filter weights can be solved by a convergent optimization algorithm or using an analytical solution. The correlation filtering algorithm generates a large number of training samples due to the cyclic matrix constructed, so there is no need to construct test and training samples. However, due to over-fitting problems and boundary effects, the performance of correlation filtering algorithms is limited. The unevenness of the number between the training sample and the constructed model causes the over-fitting problem. The boundary effect is that the target model responds at the

The associate editor coordinating the review of this manuscript and approving it for publication was Inês Domingues.

boundary of the response graph, which affects the judgment of the target position. SRDCF [16] added background penalty to the training correlation filter, BACF [14] expanded the search area with a lower computational cost, and used a clipping matrix to improve the boundary effect problem. The over-fitting problem still exists. Learning a suitable filter Weight helps to improve the problem.

The correlation filtering algorithm can be traced back to the MOSSE tracker [10], which used gray-scale features for tracking and achieved a very fast speed, but it was not very good in terms of accuracy. Danelljan *et al.* [17] added robust scale estimation to the correlation filter framework. They used a scale pyramid representation and improved the accuracy. Zhu *et al.* [18] proposed a new description operator multiple color-histogram of oriented gradient (MC-HOG), and it combined with correlation filtering framework to propose the MOCA tracker. They proposed saliency proposals to reduce background interference and a ranking strategy was used to improve system performance. The MOCA tracker could effectively solve the model drift problem. The FDSST tracker [19] used a new scale adaptive tracking method. It directly learned the target's appearance change and reduced computational cost. This gives the tracking algorithm a great advantage in speed. The LMCF algorithm [20] proposed high-confidence update and multi-peak detection, which improved the accuracy of algorithm tracking.

The correlation filtering method has certain advantages in speed and accuracy, and it has been studied by a large number of scholars one after another. It regards the process of performing correlation filtering operations on the image region of interest to obtain the result as the process of object tracking. The subsequent KCF [13] algorithm is a classic algorithm of correlation filtering algorithm, which has a fast speed and achieved good results in terms of accuracy. A large number of scholars make changes to algorithm improvement and comparison. They all achieved good results in recent years.

Recently, a large number of scholars propose many correlation filters algorithms [21]–[23]. Shu *et al.* [24] combined four features to build the target model, and they added the scale filter to prevent a single scale from causing inaccurate tracking. At the same time, the method handles the occlusion problem well. Liang *et al.* [25] proposed the multiple models to reduce the boundary effect. And they added the multi-expert to the filter, which enhanced the filter discrimination ability. The speed of this method can reach 58 frames per second (FPS), which met the real-time performance of the algorithm. Islam *et al.* [26] built the multiple features to extract the interest object and target location. And they used the high confidence score to prevent model drift. Moreover, Xuan *et al.* [27] combined motion trajectory averaging and the Kalman filter to reduce the inaccurate tracking problem. And the method could track the target well when the object was occluded. Yu *et al.* [28] used the log-Gabor filter to obtain the target feature and mitigate the boundary effect. They added the occlusion estimating strategy to update the object, which ensure the accuracy of tracking.

In recent years, many excellent articles have also appeared in major international conferences and international Journal. Zhang *et al.* [29] added the end-to-end deep architecture. The architecture was lightweight and it permitted real-time speed. It could handle samples and aspect ratio variations well. Yuan *et al.* [30] proposed a multiple feature fused model. It could combine all features. And the scale detection further improves accuracy. Xu *et al.* [31] improved algorithm based on DCF, and they introduced temporal consistent constraints to enhance the success rate of filter. Joakim *et al.* proposed DCCO [32], which regularized the sub-filter locations with an affine deformation model to avoid over-fitting. And the method improved the performance in challenging situations.

The hybrid between convolutional neural network (CNN) and correlation filter (CF) trackers had great performance in target tracking. Ma *et al.* [33] used deep feature to build the target model. The deep feature was added to the correlation filter to track target. The method achieved great performance. Qi *et al.* [34] improved the accuracy of tracking by using the Hedge method to adaptively learn the importance of each feature. Danelljan et al proposed the ECO tracker [35], which reduced the training parameters and simplified computational complexity. They used factorized convolution operator and learned the projection matrix and filter weights.

The new CNN-based trackers also track target well. Bertinetto *et al.* [36] introduced a fully-convolutional Siamese network. And the method could meet the real-time operation. Li *et al.* proposed the Siamese-RPN [37]. And the network was end-to-end trained. The speed of method could run at 160 FPS. SiamRPN++ [38] added a simple spatial aware sampling strategy. And they proposed a new model architecture. Cheng *et al.* [39] introduced a novel Relation Detector to filter the distractors from background and introduced Refinement Module. The method achieves state-of-the-art results.

Bai *et al.* [40] introduced deep learning in the feature optimization stage and achieved high performance. And they design a discrete wavelet multiscale attention mechanism to improve detector performance. The mechanism can make the detector more concentrate on the object area. Zhang *et al.* [41] proposed deep reinforcement learning in edgeAI-aided IoT. The method used a collaborative deep reinforcement learning in Edge-IoT to track the target. And it shows better performance compared with existing work. Moreover, aiming at driver drowsiness detection, Bai *et al.* [42] designed a novel and robust two-stream spatial-temporal graph convolutional network. The method used deep features to extract drivers' faces and it was applied in real-time videos. This is the first effort to adopt the method in the field of driver drowsiness detection. A novel regression loss method based on Gaussian Wasserstein distance (GWD) [43] was proposed to solve the problem of rotating detection regression loss. Then They used GWD to solve

the square-like and boundary problem and achieved well performance.

Many scholars have improved the correlation filtering algorithm, which has improved the overall performance of the algorithm, so that the algorithm can track the target more accurately. However, scholars have also encountered many problems while improving the algorithm. There is too much background information when learning the correlation filtering filter, which is easy for the filter to learn the background information, causing inaccurate tracking. And the eleven challenges in the field of target tracking always exists in the target tracking algorithm. They can make the appearance of the target change and make the target appearance information missing. In response to the above problems, we proposed a better tracking algorithm to deal with.

Our main contributions are as follows:

1) The CN feature and HOG feature are combined to extract features of the target.
2) The use of anomaly suppression correlation filters can suppress sudden changes in the filter response graph and suppress algorithm drift.
3) Add a scale filter and use bilinear interpolation to adjust the target size.
4) Use a memory mechanism to update the target model.

II. RELATE WORKS

Correlation filter algorithms have good tracking performance and fast speed. We used CF as our basic framework to track target. Our method can achieve state-of-the-art results. In the following, we will briefly review some workers correlation to our algorithm.

A. CORRELATION FILTER TRACKER

The quality of the correlation filter tracker directly affects the performance of the tracking algorithm. Correlation filter trackers have got a lot of attention recently. The correlation filtering algorithm started with the MOSSE tracker. It achieved good performance and the speed of MOSSE tracker was very fast. Henriques *et al.* [11] proposed CSK tracker. They train a general classifier framework under dense sampling. Danelljan *et al.* [12] added the color attributes and proved the feasibility of color attributes. Then, KCF tracker used HOG feature to build the target model. And correlation filter was added with a kernel space. They introduced cyclic shifts operate to train the target sample and produced sufficient samples. The fast detection and fast kernel correlation accelerated the calculation speed of the tracking system. Tang *et al.* [44] introduced a multi-kernel correlation filter named MKCF and added a novel scale estimation method. Gundogdu *et al.* [45] added a spatial windowing and provided gradient effective calculations. Li *et al.* [46] proposed a novel framework of similarity transformation estimation and a new approach for scale and rotation estimation. The method achieved good performance.

B. OBJECT MODEL UPDATE

Object model update largely affects the performance of the tracking algorithms. Choosing a suitable update method can make the algorithms track the interest target more accurately. There are a lot of ways to update the target model. But some tracking challenges also occur. For example, Scale Variation, Deformation, Background Clutters and so on. In response to the above challenges, different scholars have proposed many tracking algorithms. To a certain extent, the occurrence of challenges is reduced. The MOSSE tracker used peak sidelobe ratio (PSR) to update the target. And the speed of tracker reached 669 FPS. The LCT tracker [47] only updated in confident frames. The LMCF tracker used high-confidence update and they proposed a novel large margin object tracking method. While taking into account the accuracy, the tracker run at speed in excess of 80 frames per second.

Recently, Hu *et al.* [48] used an online model update strategy and the method was tested well on two public data sets. Mueller *et al.* [49] used the standard update rule. And they added the global context to the correlation filter framework. Xu *et al.* [50] considered all previous frames to update the discriminative classifier coefficients. The results of experiment proved the availability of this method.

III. ALGORITHM DESIGN

In this section, we will introduce our tracker in detail. The overall framework is shown in the Figure 1.

A. ALGORITHM DESCRIPTION

According to the above theoretical derivation and formula analysis, the proposed tracking algorithm flow is shown in the Figure 1:

First, initialize the constructed anomaly suppression correlation filters and targets. Start inputting the video frame, and then extract the CN feature and HOG feature from the target frame in the input current frame. Next, we construct the target feature model and optimize it by using the ADMM method to speed up the calculation of the corresponding responses of the two models to obtain the final response score and predict the target position of the next frame as its maximum value. After obtaining the predicted position, set the scale of the target. Calculate the scores of the target at each scale by adding the scale pool method, and the highest score is the target prediction scale. Finally, during the target model update stage, the update strategy of the three-layer rotating circle memory model is adopted. When the update is completed, the next frame operation is performed until all the video frame sequences are tracked to the target.

B. FILTER CONSTRUCTION

Inspired by the article [15], a correlation filter with abnormal suppression was constructed. When the response graph generated by constructing the target observation model input to the filter changes suddenly, the sudden change of the response graph can be suppressed. The two response graphs are defined

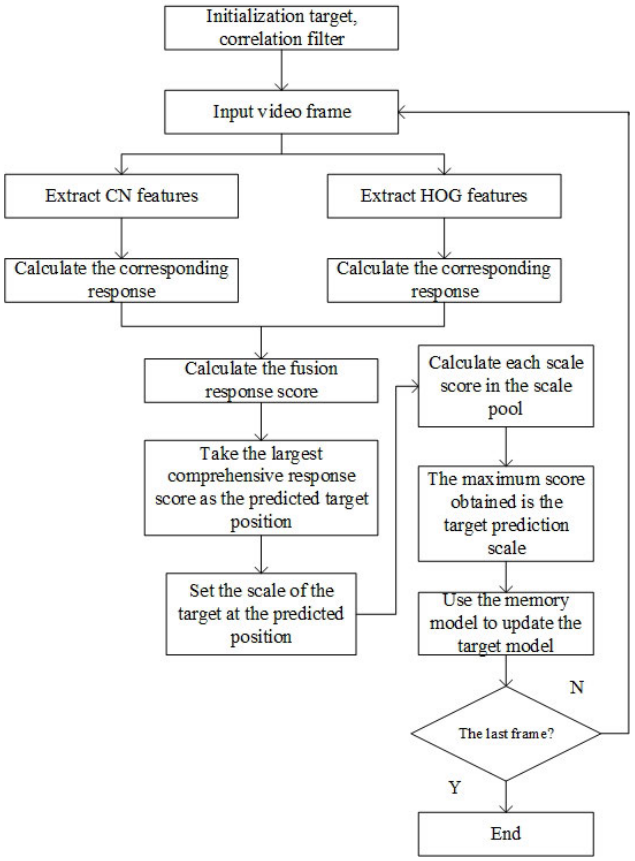


FIGURE 1. The tracking algorithm flow.

as A_1 and A_2. The difference between the response graphs calculated by Euclidean norm is:

$$\left\|A_1\left[\psi_{b,c}\right]-A_2\right\|_2^2 \tag{1}$$

In the formula, $\psi\ [b,\ c]$ represents the cyclic shift operation for the coincidence of two peaks in adjacent frames, and b and c respectively represent the positions of the two peaks in the two-dimensional space. By judging the subtracted value of the formula (1), it is judged whether there is an abnormality according to the result. When the value of formula (1) is very high, there is an abnormality in the tracking, and the similarity of the response graph will drop quickly. In order to reduce the occurrence of abnormalities in the system, the training target is optimized, and the minimized loss function is constructed as:

$$\begin{aligned}L\left(h_k\right)=&\frac{1}{2}\left\|y-\sum_{e=1}^{E}Dx_k^e*h_k^e\right\|_2^2+\frac{\alpha}{2}\sum_{e=1}^{E}\left\|h_k^e\right\|_2^2\\&+\frac{\beta}{2}\left\|\sum_{e=1}^{E}\left(Dx_{k-1}^e*h_{k-1}^e\right)\left[\psi_{b,c}\right]-\sum_{e=1}^{E}Dx_k^e*h_k^e\right\|_2^2\end{aligned} \tag{2}$$

In the above formula, h_k is the correlation filter, $h^e\in R^M$. k and k-1 are the kth and k-1th frames of the video frame, respectively. And y is the ideal response of the target, x is the vector sample, $x^e\in R^N(e=1,2,\ldots,E)$, $y\in R^N$. D is a cropping matrix, and it is used to select the center M pixels of each channel x^e of the input sample, $D\in R^{M\times N}$, * indicated as correlation operations. The third term in the formula is a regularization term added to suppress abnormalities. In order to facilitate the calculation, the loss function is converted to the frequency domain for calculation, as:

$$\begin{aligned}\hat{L}\left(h_k,\hat{p}_k\right)=&\frac{1}{2}\left\|\hat{y}-\hat{X}_k\hat{p}_k\right\|_2^2+\frac{\alpha}{2}\left\|h_k\right\|_2^2\\&+\frac{\beta}{2}\left\|\hat{A}_{k-1}^s-\hat{X}_k\hat{p}_k\right\|_2^2\end{aligned} \tag{3}$$

Among them, Λ is the discrete Fourier transform, $\hat{p}_k$ is the new parameter introduced for the optimization function, and X_k is the matrix form of the input target sample x_k. A_{k-1} is the response graph of the previous frame, and $\hat{A}_{k-1}^s$ can be regarded as a constant signal to simplify the calculation. $\hat{p}_k$ can be expressed as:

$$\hat{p}_k=\sqrt{N}\left(I_E\otimes FD^\top\right)h_k \tag{4}$$

In the formula, I_E is the recognition matrix of size $E\times E$. $\otimes$ is the Kronecker product, which acts on the matrix. T is the transpose of the matrix.

C. ADMM METHOD OPTIMIZATION

In order to speed up the calculation, the ADMM method [51] is used for optimization. Formula (3) is first extended to Lagrangian form, as follows:

$$\begin{aligned}\hat{L}\left(h_k,\hat{p}_k,\hat{q}\right)=&\frac{1}{2}\left\|\hat{y}-\hat{X}_k\hat{p}_k\right\|_2^2+\frac{\alpha}{2}\left\|h_k\right\|_2^2\\&+\frac{\beta}{2}\left\|\hat{A}_{k-1}^s-\hat{X}_k\hat{p}_k\right\|_2^2\\&+\hat{q}^\top\left(\hat{p}_k-\sqrt{N}\left(I_E\otimes FD^\top\right)h_k\right)\\&+\frac{\gamma}{2}\left\|\hat{p}_k-\sqrt{N}\left(I_E\otimes FD^\top\right)h_k\right\|_2^2\end{aligned} \tag{5}$$

In (5), A is the Fourier domain Lagrangian vector, $\hat{q}=[\hat{q}^{1T},\ldots,\hat{q}^{ET}]^T$, $\hat{q}$ is a variable with a size of DE $\times$ 1. γ is the penalty factor. By using the ADMM method, two sub-problems can be solved in the kth frame to calculate the filter of the k+1th frame. Formula (5) can be divided into two sub-problems to solve, and the sub-problems have closed-form solutions, the h_{k+1}^* equation is as follows:

$$\begin{aligned}h_{k+1}^*=arg\min_{h_k}\Big\{&\frac{\alpha}{2}\left\|h_k\right\|_2^2+\hat{q}^\top\left(\hat{p}_k-\sqrt{N}\left(I_E\otimes FD^\top\right)h_k\right)\\&+\frac{\gamma}{2}\left\|p_k-\sqrt{N}\left(I_E\otimes FD^\top\right)h_k\right\|_2^2\Big\}\end{aligned} \tag{6}$$

Formula (6) can be expressed as (7):

$$\begin{aligned}h_{k+1}^*=&(\alpha+\gamma N)^{-1}\left(\sqrt{N}\left(I_E\otimes DF^\top\right)\hat{q}\right.\\&\left.+\gamma\sqrt{N}\left(I_E\otimes DF^\top\right)\hat{p}_k\right)\\=&\left(\frac{\alpha}{N}+\gamma\right)^{-1}(q+\gamma p_k)\end{aligned} \tag{7}$$

In the above formula, q and p_k are obtained by inverse fast Fourier transform respectively, namely:

$$\begin{cases} q = \frac{1}{\sqrt{N}} \left(I_E \otimes DF^\top \right) \hat{q} \\ p_k = \frac{1}{\sqrt{N}} \left(I_E \otimes DF^\top \right) \hat{p}_k. \end{cases} \tag{8}$$

The p^*_{k+1} equation can be expressed as:

$$\begin{aligned} p^*_{k+1} = arg\min_{p_k} \Bigg\{ & \frac{1}{2} \left\| \hat{y} - \hat{X}_k \hat{p}_k \right\|_2^2 + \frac{\beta}{2} \left\| \hat{A}^s_{k-1} - \hat{X}_k \hat{p}_k \right\|_2^2 \\ & + \hat{q}^\top \left(\hat{p}_k - \sqrt{N} \left(I_E \otimes FD^\top \right) h_k \right) \\ & + \frac{\gamma}{2} \left\| p_k - \sqrt{N} \left(I_E \otimes FD^\top \right) h_k \right\|_2^2 \Bigg\} \end{aligned} \tag{9}$$

Due to the sparsity of $\hat{X}_k$, the problem of solving $\hat{p}^*_k$ is relatively complicated, and calculations are required in each ADMM iteration. $\hat{p}^*_k$ is decomposed into N small problems to solve, and it is expressed as:

$$\begin{aligned} \hat{p}_{k+1}(n)^* = arg\min_{p_k(n)} \Bigg\{ & \frac{1}{2} \left\| \hat{y}(n) - \hat{x}_k^\top(n) \hat{p}_k(n) \right\|_2^2 \\ & + \frac{\beta}{2} \left\| \hat{A}^s_{k-1} - \hat{x}_k^\top(n) \hat{p}_k(n) \right\|_2^2 \\ & + \hat{q}^\top \left(\hat{p}_k(n) - \hat{h}_k(n) \right) \\ & + \frac{\gamma}{2} \left\| \hat{p}_k(n) - \hat{h}_k(n) \right\|_2^2 \Bigg\} \end{aligned} \tag{10}$$

In (10), $\hat{x}_k(n) = [\hat{x}^1_k(n), \hat{x}^2_k(n), \ldots, \hat{x}^E_k(n)]^T$, and other elements are the same as its rules. $\hat{h}^e_k = \sqrt{E} FD^T h^e_k$ is the discrete Fourier transform operation. The solution of N small problems is written as:

$$\begin{aligned} \hat{p}_{k+1}(n)^* = & \frac{1}{1+\beta} \left(\hat{x}_k(n) \hat{x}_k^\top(n) + \frac{\gamma}{1+\beta} I_E \right)^{-1} \\ & \times \left(\hat{x}_k(n) \hat{y}(n) + \beta \hat{x}_k(n) \hat{A}^s_{k-1} \right. \\ & \left. - \hat{q}(n) + \gamma \hat{h}_k(n) \right) \end{aligned} \tag{11}$$

Combining the solutions of the above two sub-problems, the loss function and correlation filters can be obtained.

D. TARGET LOCATION AND SCALE ESTIMATION

Input the target sample into the associated filter, and finally obtaining a response graph after a series of calculations. The position of the maximum response value in the graph is the estimated position of the target. The position of the tracking target can be expressed in the Fourier domain as:

$$\hat{A} = \hat{x}_k \odot \hat{h}_k \tag{12}$$

Many filters are unable to deal with target scale changes, resulting in low tracking accuracy and even tracking failure. Therefore, the article was inspired by [52] to train the scale filter S_k to deal with the scale problem of the target.

The size of target template is set to $R_t = (R_u, R_v)$. Then the scale pool is set to $W = \{t_1, t_2, \ldots, t_k\}$. It is supposed that the size of the target in the initial image space is r_t, and k size $\{t_i R_t | t_i \in W\}$ are sampled in the current video frame to estimate the target of interest. In this paper, bilinear interpolation is used to adjust the target sample size and adjust it to R_t. The response formula is as follows:

$$J = arg\max F^{-1} \hat{f} \left(z^{t_i} \right) \tag{13}$$

In (13), z^{t_i} is the sample patch of $t_i R_t$, and f is the sample function. The maximum value of the scale response is defined as the scale of the target.

The HOG feature and CN feature are extracted to construct the target appearance model. The HOG feature can describe the shape information of the target part and the quantification of the position and direction space can suppress the influence of translation and rotation to a certain extent. And it uses the normalized histogram in a local area, which can partially offset the impact of light changes. Moreover, due to this processing method of block and unit, the relationship between the local pixels of the image can be well characterized. The CN feature is a global feature that describes the surface properties of the scene corresponding to the image or image area. It is the most widely used visual feature in image retrieval. And it is not affected by image rotation and translation changes. Moreover, color name feature has high robustness.

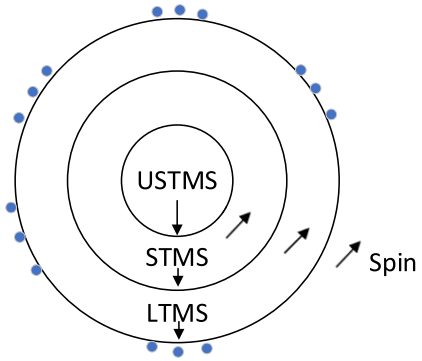


FIGURE 2. Three-layer rotating circle memory model.

E. LAGRANGIAN PARAMETERS UPDATE STRATEGY

When using the ADMM method to optimize the calculation speed, the Lagrangian parameters are updated, and the equation is as follows:

$$\hat{q}^{(i+1)}_{k+1} = \hat{q}^i_{k+1} + \gamma \left(\hat{p}^{*(i+1)}_{k+1} - \hat{h}^{*(i+1)}_{k+1} \right) \tag{14}$$

In the above formula, i is the i-th iteration, and i + 1 is the same. The solution to h^*_{k+1} is $\hat{h}^{*(i+1)}_{k+1}$. And the solution to p^*_{k+1} is $\hat{p}^{*(i+1)}_{k+1}$. At the same time, the equation of $\hat{h}^{*(i+1)}_{k+1}$ is as follow:

$$\hat{h}^{*(i+1)}_{k+1} = (I_E \otimes FD^T) h^{*(i+1)}_{k+1} \tag{15}$$

F. APPEARANCE MODEL UPDATE

The article uses a three-layer rotating circle memory model [53] to update the target appearance model. Use the

three memory spaces of the human brain to represent the three layers of circles, namely instantaneous memory space, short-term memory space, and long-term memory space. As shown in Figure 2:

When the target sample enters the memory space, the three memory spaces rotate at different speeds, and the target templates of the input and output windows of the rotating circle are continuously replaced, namely:

$$\delta[s] = \delta[(s+V)\,\%M] \tag{16}$$

Among them, s is the s-th template of the target. After rotation, the target template $\delta[s]$ is replaced by the latter part of formula (16), V is the rotation speed. The capacity of the circular memory space is M, and % is the remainder calculation.

IV. EXPERIMENTS

A. IMPLEMENTATION DETAILS

This chapter proposes anomaly suppression correlation filtering methods. After constructing the filter, ADMM is used to optimize the calculation speed. The memory mechanism updates the model, which can deal with problems such as occlusion of targets, complex backgrounds, and re-targets after they disappear. In order to verify the performance of the algorithm proposed in this chapter, it was compared with five tracking algorithms: BACF, KCC [54], KCF, MKCFup [55], and ARCF [15]. The experiment environment is Intel(R) i5-8265U CPU @ 1.6GHz 1.8GHz, and the experiment platform is simulated on Matlab R2014a. The performance of the six algorithms is tested on the experimental data set. The data set uses OTB50 [56], OTB100 [57], TC128 [58], UAV123 [59], which contains various scenes of target tracking problems. The visual tracking algorithm based on peak sidelobe ratio proposed in this paper uses an object-interference model and a kernel-correlation filter to deal with target deformation and occlusion problem. In order to verify the proposed algorithm's effectiveness, it is compared with tracking.

This chapter uses commonly used parameters for evaluation, and the success rate curve and algorithm accuracy diagram are shown as follow. The algorithm proposed in this chapter is represented by PRO3. The parameter α used in the algorithm in this chapter is 0.2, β is 0.7, and the learning rate is set to 0.021. The ADMM iteration step is set to 5, the scale pool W = {0.99,0.995,1.0,1.005,1.01}, the sizes of the three memory space input windows are all set to 1, and their storage capacities are 11, 13, and 15. In this chapter, the algorithm extracts the target HOG feature and CN feature to build the target model to improve the accuracy of the model.

B. RESULTS ON OTB50

OTB50 contains 49 challenging video sequences, it is manually tagged with 11 attributes that infect different problems, which are shown in Table 1. The OTB50 data set is the earliest data set that has been publicly used by a large number of scholars. And it is the most widely used data set in the field of target tracking. We compared PRO3 with other five algorithms on OTB50. The algorithms are BACF, KCC, KCF, MKCFup, and ARCF.

Table 1 selects 12 sequences from the OTB data set, and each sequence is labeled with a different challenge. And in each video sequence, the tracking target can be a person or an object, and they have various shapes. For example, in the BlurBody sequence, the person in the video is tracked, and in the Coke video, the Coke bottle is tracked. During the tracking process, the target's motion form will change under the influence of various situations, which will increase the difficulty of target tracking.

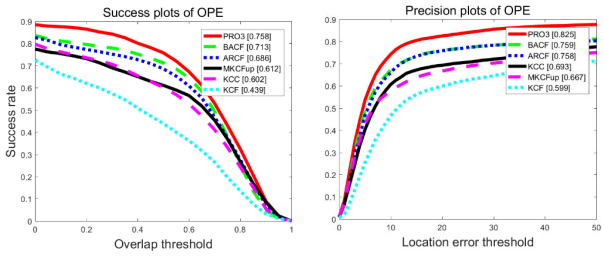


FIGURE 3. PRO3 and five trackers on OTB50 success rate graph and accuracy graph.

The results on OTB50 are shown in Figure 3. From the algorithm accuracy chart, it can be concluded that the accuracy of the algorithm proposed in this paper is 0.825, which is higher than the baseline tracking algorithm KCF, which illustrates the superiority of the algorithm in this paper. The result chart is shown in the figure 3 below.

In order to clearly show the success rate and accuracy of various algorithms, we draw a table a s shown in Table 2. And we add ECO to compare with various algorithms. Table 2 shows the results of the six algorithms in the OTB50 data set. The leftmost column of the table is the names of different target tracking algorithms. The middle column represents the success rate data of various algorithms after the data set is run. The greater the success rate, the better. And the last column shows the accuracy data of various algorithms after the data set is run. The higher the precision, the better the algorithm performance.

Compared with the recent ARCF algorithm, our method improves the success rate by 7.2% and the accuracy by 6.7%, which proves the superiority of the method proposed in this chapter.

The results of the PRO3 algorithm running on this data set are very good, with a success rate of 0.758 and an accuracy of 0.825. This algorithm surpasses the results of other algorithms on the data set, and has improved data compared with other algorithms.

C. RESULTS ON OTB100

The OTB100 data set contains 11 challenges, and a large number of scholars have tested the quality of the target tracking algorithm on this data set. The data set contains 25% of the gray data set, and the groundtruth of the video sequence is

TABLE 1. Part of challenging video sequences with 11 challenges.

Sequence	OCC	SV	MB	IV	DEF	BC	FM	IPR	OPR	LR	OV
BlurBody		√	√		√		√	√			
Box	√	√	√	√		√		√	√	√	√
Coke	√			√		√	√	√	√		
Crossing		√			√	√	√		√		
Singer1	√	√		√					√		
Trans	√	√		√	√						
Surfer		√					√	√	√	√	
Skating1	√	√		√	√	√			√		
Walking	√	√			√						
Toy		√					√	√	√		
Subway	√				√	√					
Gym		√			√			√	√		

TABLE 2. Performance comparison between PRO3 and five tracking algorithms on OTB50. means bigger is better.

Tracker	OTB50	
	Success↑	Precision↑
PRO3	0.758	0.825
BACF	0.713	0.759
KCC	0.602	0.693
KCF	0.439	0.599
MKCFup	0.612	0.667
ARCF	0.686	0.758
ECO	0.668	0.716

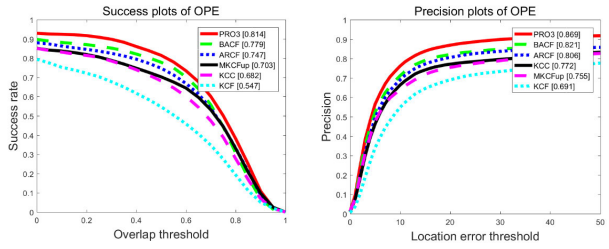


FIGURE 4. PRO3 and five trackers on OTB100 success rate graph and accuracy graph.

TABLE 3. Performance comparison between PRO3 and five tracking algorithms on OTB100. means bigger is better.

Tracker	OTB100	
	Success↑	Precision↑
PRO3	0.814	0.869
BACF	0.779	0.821
KCC	0.682	0.772
KCF	0.547	0.691
MKCFup	0.703	0.755
ARCF	0.747	0.806
ECO	0.698	0.794
DCCO	0.690	0.595

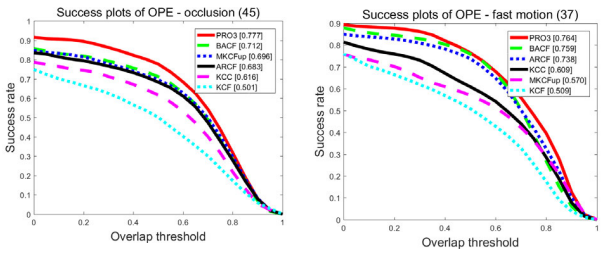


FIGURE 5. Precision plots of OPE of six trackers in fast motion scene and occlusion scene.

manually annotated. The OTB100 data set has a rectangular frame plus random interference initialization to run, and part of the sequence starts from a random frame, so many scholars choose this data set for testing. The PRO3 algorithm and other five algorithms are tested on the OTB100 data set, and the running results are shown in Figure 4:

In order to clearly show the success rate and accuracy of various algorithms, we draw a table as shown in Table 3. And we add ECO and DCCO to compare with various algorithms. Table 3 shows the results of seven algorithms on the OTB100 data set. The ECO tracking algorithm and the DCCO tracking algorithm are recent algorithms with excellent performance. Adding them to compare with the PRO3 algorithm can better illustrate the effectiveness of the work done. In the algorithm comparison, the PRO3 algorithm has the highest success rate and accuracy, which is better than other tracking algorithms.

The above data shows that the PRO3 algorithm leads other trackers in terms of success rate and accuracy. Compared with the second-ranked BACF algorithm, the success rate has increased by 3.5%, and the accuracy has increased by 4.8%.

The PRO3 algorithm performed well on this data set, with a success rate of 0.814 and an accuracy of 0.869. Compared with the recent ECO and DCCO algorithms, the success rate has increased by 0.116 and 0.124 respectively, demonstrating the superiority of the PRO3 algorithm.

Figure 5 is a comparison of the fast motion scene and occlusion scene in the tracking challenge between the PRO3 and the state-of-the-art algorithms. We use a memory mechanism to update the target model. The accuracy of the two challenges reaches 0.764 and 0.777 respectively.

D. RESULTS ON UAV123

The UAV123 data set is a special data set. All of its videos are captured by drones, and there are many long video frame sequences, which can detect the performance of the algorithm very well. The shooting angle of the data set changes greatly, and the background is relatively clean. In addition, the tracking target in the video sequence is small, and the video is too long to easily degrade the target model, which requires the tracking algorithm to have high performance. The six algorithms in this chapter track the target on this data set, and the results obtained are shown in Figure 6:

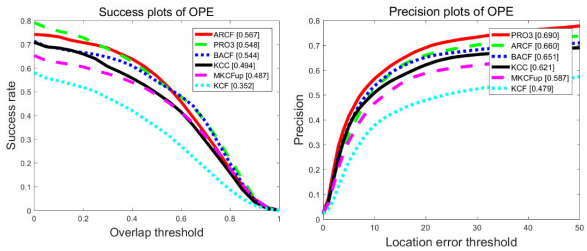


FIGURE 6. PRO3 and five trackers on UAV123 success rate graph and accuracy graph.

TABLE 4. Performance comparison between PRO3 and five tracking algorithms on UAV123. means bigger is better.

Tracker	UAV123	
	Success↑	Precision↑
PRO3	0.548	0.690
BACF	0.544	0.651
KCC	0.494	0.621
KCF	0.352	0.479
MKCFup	0.487	0.587
ARCF	0.567	0.660
ECO	0.530	0.617

In order to clearly show the success rate and accuracy of various algorithms, we draw a table as shown in Table 4. And we add ECO to compare with various algorithms.

In the UAV123 data set, because the video is too long and the angle of change is large, the success rate and accuracy of various algorithms are not very high. The success rate of the proposed algorithm is 0.548 and the accuracy is 0.690.

Compared with the recent ARCF algorithm in terms of success rate, the performance of the proposed algorithm is slightly inferior, but it ranks first in terms of accuracy and has a strong overall tracking ability.

Figure 7 selects three relatively complex sequences in the experimental data set UAV123, and compares the proposed algorithm with the five algorithms proposed by previous researchers. The BACF algorithm is improved on the correlation filter framework to solve the sample problem. It expands the sampling area of the circulant matrix, adds a large number of target samples, and cuts out useful samples.

In Figure 7(a), the cyclist is tracked. In the first half of the video, all algorithms can keep up with the target movement. The scale and deformation of the target in the video occurred. In the second half of the video sequence, the KCC algorithm drifted and finally failed to track the target. At frame 2987 of the video, the companion who is tracking the target is riding in front and the tracking target is behind. As the video continues to play, the target scale changes. The algorithm in this chapter can quickly adapt to the change of target scale and accurately track the target, which verifies the effectiveness of the algorithm in this chapter.

In Figure 7(b), the characters in the game are tracked. At the beginning, the target was running on a straight road, and the six tracking algorithms in this chapter quickly tracked the character. At frame 131 of the video, the shooting lens zooms in towards the target person, and the target person becomes larger. At 264 frames, the drone flies directly above the character, and the ARCF algorithm tracking frame becomes larger at this time, resulting in inaccurate tracking. In the 494 frames of the video, the character jumped from the high platform, and the character scale changed. In the complex scene, the ARCF and KCC tracking algorithms quickly shifted, and the tracking failed in the second half of the video.

In Figure 7(c), the tracked object is a person wearing a black suit. In the first half of the video, the characters in the video get off the car and bed in a black suit. All six algorithms are sufficient to keep up with the tracking object, and the tracking is relatively stable. But at frame 1204, the MKCFup algorithm and KCF algorithm showed unstable tracking when the drone went from shooting the whole person to only shooting the half body.

The above analysis shows that the PRO3 algorithm adds a scale filter and uses a three-layer rotating circle memory model to update the target appearance model, so that the algorithm has better tracking performance in scenes such as scale changes, complex backgrounds, and target deformations. And it can better cope with challenges in target tracking.

E. RESULTS ON TEMPLE COLOR 128

Temple Color 128 contains 128 challenging video sequences. The color model is helpful for tracking the targe. And the color sequence contains 11 different problem attributes. Color information can provide rich discriminative clues for visual reasoning. The TC128 data set contains rich color sequences

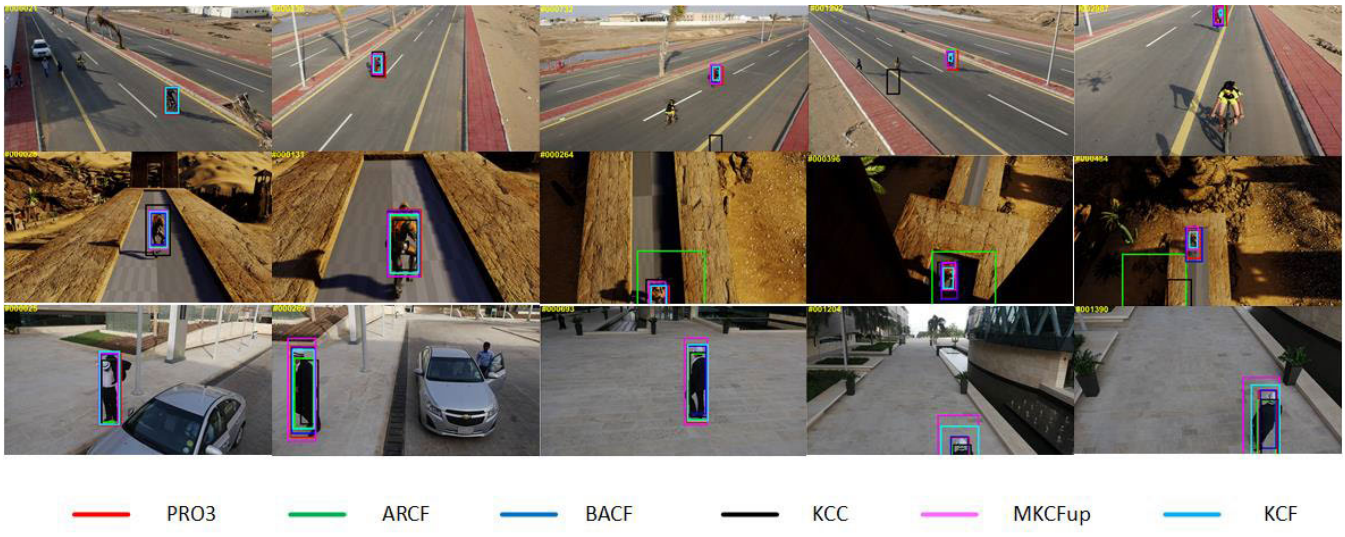


FIGURE 7. (a) Bike sequence (b) person3_s sequence (c) person18 sequence. Comparison PRO3 and ARCF, BACF, KCC, KCF, MKCFup on UAV123.

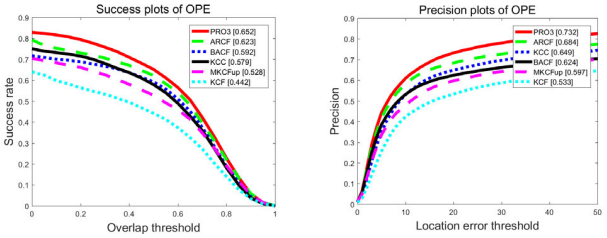


FIGURE 8. PRO3 and five trackers on TC128 success rate graph and accuracy graph.

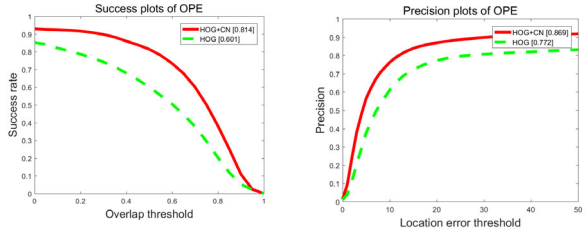


FIGURE 9. Algorithm with different characteristics on OTB100 success rate graph and accuracy graph.

TABLE 5. Performance comparison between PRO3 and five tracking algorithms on TC128. means bigger is better.

Tracker	TC128	
	Success↑	Precision↑
PRO3	0.652	0.732
BACF	0.592	0.624
KCC	0.579	0.649
KCF	0.442	0.533
MKCFup	0.528	0.597
ARCF	0.623	0.684
ECO	0.601	0.669
DCCO	0.594	0.660

and it can better evaluate the performance of the algorithm. There are 50 videos in the data set, and 78 videos have never been seen before. Compared with five trackers, the results obtained are shown in Figure 8:

In order to clearly show the success rate and accuracy of various algorithms, we draw a table as shown in Table 5. And we add ECO and DCCO to compare with various algorithms.

The success rate of our tracker is 65.2% and the precision is 73.2%. Compared with DCCO, PRO3 achieves some

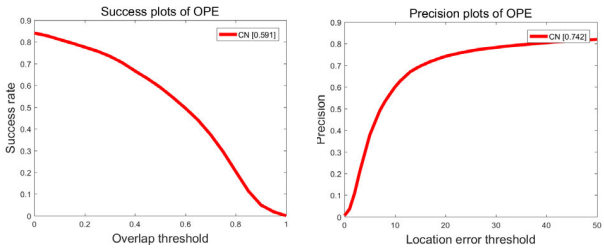


FIGURE 10. Algorithm with CN characteristic on OTB100 success rate graph and accuracy graph.

improvement by a gain of 5.8% and 7.2%, in terms of success and precision.

In the color data set, the PRO3 algorithm and ARCF algorithm perform well. The ARCF algorithm ranks second among eight algorithms, second only to the PRO3 algorithm. Moreover, our tracker can track the target more accurately in the color sequence.

F. ABLATION EXPERIMENTS

In this section, we conduct experiments on the OTB100 data set and the TC128 data set to verify the feasibility of the proposed algorithm. These two data sets have been used by a large number of scholars for testing because

TABLE 6. The fps value before and after using the ADMM method on the TC28 data set.

Tracker	TC128
	fps
ADMM-	16.247
ADMM+	24.584

of their rich video types and many types of tracking objects.

First, we use the HOG feature alone to add it to the filter framework we built, and test on two data sets. And the results are as follows:

In the above figure, HOG represents the algorithm that uses HOG features alone, and HOG+CN represents the algorithm generated by adding the fused features into the correlation filtering framework.

It can be seen from the figure that the success rate and accuracy of the HOG feature algorithm after running on the OTB100 data set are 0.601 and 0.772, respectively, and the algorithm after feature fusion is 0.814 and 0.869. Therefore, it can be obtained that, compared with the algorithm before the fusion, the success rate of the fusion algorithm is improved by 21.3% and the accuracy is improved by 9.7%, which proves that the fusion algorithm is better than using it alone.

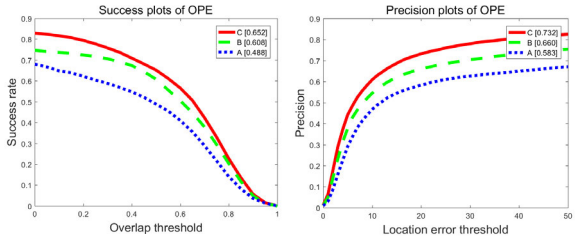


FIGURE 11. Algorithm with different filters on TC128 success rate graph and accuracy graph.

In addition, we continue to do a set of experiments on the OTB100 data set. CN represents the algorithm that uses CN features alone. It can be seen from the figure that the success rate and accuracy of the CN feature algorithm after running on the OTB100 data set are 0.591 and 0.742, respectively. Comparing the result with the above figure again, it proves the superiority of the fusion feature.

Second, we use the ADMM algorithm to speed up the calculation. Before adopting this method, the running speed was slower. The algorithm is tested on TC128 before and after acceleration. The fps value before acceleration is 16.247, and the fps value after acceleration is 24.584, which is a numerical increase of 8.337. In order to clearly show the fps value, we draw a table as shown in Table 6.

ADMM- means before the algorithm is accelerated, ADMM+ means after the algorithm is accelerated.

Third, we use the most basic correlation filter and test it on the TC128 data set without adding a scale filter. The algorithm for this case is named A. The algorithm that adds anomaly suppression related filters separately and does not add a scale filter is named B. The algorithm for adding scale filter and anomaly suppression correlation filter is also the algorithm we proposed, which is named C here. The experiments are all carried out under the same conditions, and the results obtained are as follows:

It can be seen from the figure that the success rate and accuracy of the A algorithm is the lowest. The C algorithm ranks first in terms of accuracy and success rate, which are 0.732 and 0.652, respectively. The C algorithm is improved compared to the A and B algorithms, showing the superiority of the proposed algorithm.

V. CONCLUSION

We use anomaly suppression correlation filtering method to extract HOG features and CN features to construct the target appearance model, avoiding the problem of inaccurate description of the target model due to a single feature. At the same time, scale filters and anomaly suppression correlation filters are added to improve the tracking ability of the system. In addition, the proposed algorithm uses the ADMM method to speed up the filter calculation speed, while updating the Lagrangian coefficients. And we use a three-layer rotating circle memory model to update the target appearance model. The proposed method timely updates the drift and damage models caused by inaccurate tracking to improve the accuracy of tracking. The proposed algorithm and other algorithms are tested on four data sets: OTB50, OTB100, UAV123, and TC128. The results show that the proposed algorithm has high parameters in all aspects. It proves the feasibility of the algorithm of the article, which is helpful for the development of target tracking.

REFERENCES

[1] S. Liu, P. Chen, Y. L. Yu, X. Cui, and D. Špelič, "Group behavior recognition in videos based on cam-shift tracking and histogram changing rate," *Int. J. Performability Eng.*, vol. 14, no. 7, pp. 1600–1608, 2018.
[2] A. Jalal and S. Kamal, "Improved behavior monitoring and classification using cues parameters extraction from camera array images," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 5, no. 5, pp. 71–78, Jun. 2019.
[3] Y. Jihong and Y. Lu, "Behavior recognition based on track space-time characteristics," *J. Intell. Fuzzy Syst.*, vol. 39, no. 6, pp. 8675–8684, Dec. 2020.
[4] A. Adigun, A. Osofisan, O. Longe, and M. Kolawole, "Improving method of evaluating semantic filtering for human computer interaction in an adaptive collaborative learning environment," *Brit. J. Math. Comput. Sci.*, vol. 7, no. 4, pp. 293–298, Jan. 2015.
[5] M. H. Sun and Y. Y. Chen, "Simulation of real-time recognition method of gesture track in human-computer interaction system," *Comput. Simul.*, vol. 37, no. 8, pp. 380–383 and 413, 2020.
[6] R. Khilar and S. Chitrakala, "A novel method for effective vehicle detection and tracking eliminating occlusion," *Res. Social Sci. Hum.*, vol. 1, no. cs1, pp. 714–727, 2016.
[7] N. Y. Ershadi, J. M. Menéndez, and D. Jiménez, "Robust vehicle detection in different weather conditions: Using MIPM," *PLoS ONE*, vol. 13, no. 3, pp. 1–30, 2018.
[8] X. X. Li, W. Sun, M. M. Liu, L. L. Zheng, and S. Y. Chen, "Research on vehicle detection and tracking algorithms in traffic monitoring scenes," *Comput. Eng. Appl.*, vol. 57, no. 8, pp. 103–111, 2021.

[9] S. Song, Y. Li, Q. Huang, and G. Li, "A new real-time detection and tracking method in videos for small target traffic signs," *Appl. Sci.*, vol. 11, no. 7, p. 3061, Mar. 2021.

[10] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.

[11] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 702–715.

[12] M. Danelljan, F. S. Khan, M. Felsberg, and J. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1090–1097.

[13] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[14] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1135–1143.

[15] Z. Huang, C. Fu, Y. Li, F. Lin, and P. Lu, "Learning aberrance repressed correlation filters for real-time UAV tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2891–2900.

[16] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4310–4318.

[17] M. Danelljan, G. Häger, F. Shahbaz Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, Nottingham, U.K., 2014, pp. 1–11.

[18] G. Zhu, J. Wang, and Y. Wu, "MC-HOG correlation tracking with saliency proposal," in *Proc. 30th AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 3690–3696.

[19] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Aug. 2017.

[20] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4021–4029.

[21] X. Meng and J. Duan, "Advances in correlation filter-based object tracking algorithms: A review," *J. Beijing Univ. Technol.*, vol. 46, no. 5, pp. 28–51, May 2020.

[22] Y. Chen, Y. Wu, and W. Zhang, "Survey of target tracking algorithm based on Siamese network structure," *Comput. Eng. Appl.*, vol. 56, no. 6, pp. 10–18, 2020.

[23] Y. Tang, Y. Liu, and H. Huang, "Survey of single-target visual tracking algorithms," *Meas. Control Technol.*, vol. 39, no. 8, pp. 21–34, 2020.

[24] Q. Shu, H. Lai, L. Wang, and Z. Jia, "Multi-feature fusion target relocation tracking based on correlation filters," *IEEE Access*, vol. 9, pp. 28954–28964, 2021.

[25] M. Liang, X. Wu, Y. Wang, Z. Zhu, B. Cao, and J. Xu, "Multi-model and multi-expert correlation filter for high-speed tracking," *IEEE Access*, vol. 9, pp. 52326–52335, 2021.

[26] M. Islam, G. Hu, W. Dan, C. Lyu, and Q. Liu, "Correlation filter based moving object tracking with scale adaptation and online re-detection," *IEEE Access*, vol. 6, pp. 75244–75258, 2018.

[27] S. Y. Xuan, S. Y. Li, M. F. Han, X. Wan, and G. S. Xia, "Object tracking in satellite videos by improved correlation filters with motion estimations," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1–13, Feb. 2020.

[28] M. Yu, Y. Zhang, Y. Li, Z.-L. Lin, J. Li, and C. Wang, "Saliency guided visual tracking via correlation filter with log-Gabor filter," *IEEE Access*, vol. 8, pp. 158184–158196, 2020.

[29] M. D. Zhang, Q. Wang, J. L. Xing, J. Gao, P. X. Peng, W. M. Hu, and S. Maybank, "Visual tracking via spatially aligned correlation filters network," in *Proc. Eur. Conf. Comp. Vis. (ECCV)*, Sep. 2018, pp. 484–500.

[30] D. Yuan, X. Zhang, J. Liu, and D. Li, "A multiple feature fused model for visual object tracking via correlation filters," *Multimedia Tools Appl.*, vol. 78, no. 19, pp. 27271–27290, Oct. 2019.

[31] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5596–5609, Nov. 2019.

[32] M. Felsberg, A. Heyden, and N. Krüger, "DCCO: Towards deformable continuous convolution operators for visual tracking," in *Proc. Int. Conf. Comput. Anal. Images Patterns*. Cham, Switzerland: Springer, 2017, pp. 55–67.

[33] C. Ma, J. B. Huang, X. K. Yang, and M. H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3074–3082.

[34] Y. Qi, S. Zhang, L. Qin, H. Hao, Q. Huang, J. Lim, and M. H. Yang, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4303–4311.

[35] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6638–6646.

[36] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 850–865.

[37] B. Li, J. J. Yan, W. Wu, Z. Zhu, and X. L. Hu, "High performance visual tracking with Siamese region proposal network," in *Proc. CVPR*, Jun. 2018, pp. 8971–8980.

[38] B. Li, W. Wu, Q. Wang, F. Y. Zhang, J. L. Xing, and J. J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," in *Proc. CVPR*, Jun. 2019, pp. 4282–4291.

[39] S. Y. Cheng, B. N. Zhong, G. R. Li, X. Liu, Z. J. Tang, X. X. Li, and J. Wang, "Learning to filter: Siamese relation network for robust tracking," in *Proc. CVPR*, Jun. 2021, pp. 4421–4431.

[40] J. Bai, J. Ren, Y. Yang, Z. Xiao, W. Yu, V. Havyarimana, and L. Jiao, "Object detection in large-scale remote-sensing images based on time-frequency analysis and feature optimization," *IEEE Trans. Geosci. Remote Sens.*, early access, Oct. 21, 2021, doi: 10.1109/TGRS.2021.3119344.

[41] J. Zhang, M. Z. A. Bhuiyan, X. Yang, A. K. Singh, D. F. Hsu, and E. Luo, "Trustworthy target tracking with collaborative deep reinforcement learning in EdgeAI-aided IoT," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1301–1309, Feb. 2022.

[42] J. Bai, W. Yu, Z. Xiao, V. Havyarimana, A. C. Regan, H. Jiang, and L. Jiao, "Two-stream spatial-temporal graph convolutional networks for driver drowsiness detection," *IEEE Trans. Cybern.*, early access, Oct. 4, 2021, doi: 10.1109/TCYB.2021.3110813.

[43] X. Yang, J. Yan, Q. Ming, W. Wang, X. Zhang, and Q. Tian, "Rethinking rotated object detection with Gaussian Wasserstein distance loss," 2021, *arXiv:2101.11952*.

[44] M. Tang and J. Y. Feng, "Multi-kernel correlation filter for visual tracking," in *Proc. ICCV*, 2015, pp. 3038–3046.

[45] E. Gundogdu and A. A. Alatan, "Spatial windowing for correlation filter based visual tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1684–1688.

[46] Y. Li, J. Zhu, S. C. H. Hoi, W. Song, Z. Wang, and H. Liu, "Robust estimation of similarity transformation for visual object tracking," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 8666–8673, Jul. 2019.

[47] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5388–5396.

[48] H. Hu, B. Ma, J. Shen, and L. Shao, "Manifold regularized correlation object tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1786–1795, May 2018.

[49] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1396–1404.

[50] F. Xu, H. Wang, Y. Song, and J. Liu, "A multi-scale kernel correlation filter tracker with feature integration and robust model updater," in *Proc. Chin. Control Decis. Conf. (CCDC)*, 2017, pp. 1934–1939.

[51] Y. Sun, C. Sun, D. Wang, Y. He, and H. Lu, "ROI pooled correlation filters for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5776–5784.

[52] L. Yang and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 254–265.

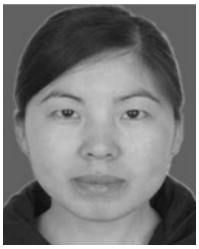
[53] Y. J. Wang, H. Li, and Y. J. Qi, "Robust moving object tracking method inspired by the mechanism of human brain three-stage memory model," *ACTA Electron. Sinica*, vol. 45, no. 9, pp. 2065–2070, Sep. 2017.

[54] C. Wang, L. Zhang, L. H. Xie, and J. Yuan, "Kernel cross-correlator," in *Proc. 32nd AAAI Conf. Artif. Intell.*, Feb. 2018, pp. 4179–4186.

[55] M. Tang, B. Yu, F. Zhang, and J. Wang, "High-speed tracking with multi-kernel correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4874–4883.

[56] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A bench-mark," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2411–2418.

[57] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015, doi: 10.1109/TPAMI.2014.2388226.
[58] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.
[59] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, Oct. 2016, pp. 445–461.



HONGGE REN received the B.E. degree in measurement and control technology and instrument from the North China University of Science and Technology, Qinhuangdao, Hebei, China, in 2003, and the M.E. and Ph.D. degrees from the Beijing University of Technology, Beijing, China, in 2007 and 2011, respectively.

She is currently an Associate Professor with the School of Control and Mechanical Engineering, Tianjin Chengjian University. Her research interests include cognitive robots, image processing, and object tracking.

YIN CUI received the B.E. degree in electrical engineering and automation from the Qinggong College, North China University of Science and Technology, Tangshan, Hebei, China, in 2018. He is currently pursuing the master's degree with the North China University of Science and Technology.

His research interests include computer vision, correlation filter, and object tracking.

TAO SHI received the Ph.D. degree in control science and engineering from the Beijing University of Science and Technology, Beijing, China, in 2015.

He is currently an Associate Professor with the School of Electrical Engineering and Automation, Tianjin University of Technology. His research interests include brain-like intelligent robots, robot vision, and biologically inspired intelligent computing.