IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Adaptive Kernel Correlation Filter Tracking Algorithm in Complex Scenes

Jinping Sun[1,2], Enjie Ding[1], Bo Sun[3], Zhongyu Liu[1], and Kailiang Zhang[2]

[1]School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221008, China
[2]School of Information Engineering(School of Big Data), Xuzhou University of Technology, Xuzhou 221008, China
[3]College of Information Science and Engineering, Shandong Agricultural University, Taian 271019, China

Corresponding authors: Enjie Ding (enjied@cumt.edu.cn) and Bo Sun (sunb@sdau.edu.cn).

This work was supported by the Ministry of Housing and Urban-rural Development Science and Technology Planning Project (2016-R2-060), the National Key Research and Development Plan of China (No.2017YFC0804400, No.2017YFC0804401), Jiangsu technology project of Housing and Urban-Rural Development (No.2018ZD265), Major Project of Natural Science Research of the Jiangsu Higher Education Institutions of China (18KJA520012), Xuzhou Science and Technology Plan Project (KC19197) and School-level Scientific Research Project of Xuzhou Institute of Technology (XKY20191070).

ABSTRACT The traditional kernel correlation filter (KCF) algorithm has poor tracking results in complex scenes with severe occlusion, deformation, and low resolution and cannot achieve long-term tracking. To improve the accuracy of the tracking algorithm in complex scenes, an adaptive kernel correlation filter algorithm is proposed. First, a multifeature complementary scheme is proposed that linearly weights the responses of the histogram of oriented gradient (HOG) features and color features and learns a target position estimation model to realize target position estimation. Then, an adaptive scale model for estimating the scale transformation of the target is learned by extracting the HOG features of the object. Finally, according to occlusion judgment criteria, the Kalman filter is introduced to correct the position of the tracking target. The accuracy and success rate of the proposed algorithm are verified by simulation analysis on TC-128/OBT2015 benchmarks. Extensive experimental results illustrate that the proposed tracker achieves competitive performance compared with state-of-the-art trackers. The distance precision rate and overlap success rate of the proposed algorithm on OTB2015 are 0.899 and 0.635, respectively. The proposed algorithm effectively solves the long-term object tracking problem in complex scenes. This study provides references for computer vision processing, such as image retrieval, behavior analysis, and intelligent driving.

INDEX TERMS Object tracking, histogram of oriented gradient, Kalman filter, kernel correlation filter

I. INTRODUCTION

Object tracking technology [1-3] is an important research direction in the current computer vision field. Moving object tracking based on vision has been widely used in fields of surveillance systems [4], drone vision systems, behavior understanding, human-computer interaction, and unmanned vehicle navigation [5]. Given the size and position in the initial frame, the essence of object tracking is to predict the size and position of the tracking object in subsequent frames. At present, object tracking algorithms are mainly based on the continuously adaptive mean shift (CAMShift), support vector machines, correlation filters, and deep learning. Bradski [6] proposed the CAMShift algorithm, which is a tracking algorithm based on the color probability distribution. The CAMShift algorithm has a good tracking effect for pure-color objects on a black and white background, but if the background color is close to that of the target, or there are objects close to the target with a similar hue, the tracking will fail. The CAMShift algorithm has been improved in terms of feature extraction and position prediction. Based on the color probability distribution, texture features [7] with distinguishing ability were introduced to compensate for the shortcomings of the traditional CAMShift algorithm. The target histogram template [8] was generated based on the combination of hue component H and saturation component S, and the adaptability to complex environments was enhanced using complete target color information features. The Kalman filter was employed to predict the position of a specific target in the next frame, but the accuracy of the target position was affected by the interference of similar colors [9-11]. Due to the excellent effect of the correlation tracking algorithm (CF), it was introduced into target tracking [12-15]. Bolme [16] first adopted the correlation filter framework, which used the minimum output sum of square error (MOSSE) algorithm, and the tracking speed was greatly improved. Henriques proposed the circulant structure of tracking-by-detection with kernels (CSK) algorithm [17-18], which used the diagonalization of the circulant matrix in the calculation process to simplify the calculation of nuclear regression, so the target tracking speed was greatly improved, and its tracking accuracy was also higher. Danelljan proposed a discriminative scale-space tracker algorithm that used the

histogram of oriented gradients (HOG) features to build a scale pyramid for target scale estimation based on MOSSE [19]. However, when the target scale continues to increase, the convolution calculation for extracting target features and training filters would increase, which would lead to a decrease in target tracking speed. The kernel correlation filter algorithm [20] is a further improvement of the CSK algorithm that uses HOG to track the target and improves the accuracy of tracking. The kernel correlation filter (KCF) algorithm [20] is a further improvement of the CSK algorithm. The HOG features were extracted to detect the object, improving the accuracy of tracking. Karunasekera [21] discussed the latest trends and the progress of tracking algorithms, compared the performance of trackers based on correlated filters and noncorrelated filters, and provided an important reference for the research of target tracking algorithms. The spatially regularized discriminative correlation filters (DCF) [22-24] tracker adopt large space support to learn the correlation filters, which effectively reduces the boundary effect but at a high computational cost.

To account for the appearance changes over time, considerable efforts have been made to design invariant manual features to represent target objects, such as color histograms [25], HOG [26], speed up robust feature (SURF) [27], scale-invariant feature transform (SIFT) [28], texture feature, and superpixels [29]. Combining HOG features and color features [30], Bertinetto [31] proposed a real-time tracking algorithm (CLRT) based on the ridge regression framework with a speed of 50 FPS. The Kalman filter was used to predict the state of the target, determine whether the target was occluded, and mark it to predict the location of the occluded target [32-33]. Zhang [34] established the descriptors of rotation and scale normalization, fused the color features and texture features to perform optimal similarity matching on the descriptors in the candidate frames, and obtained the optimal matching solution for object tracking.

Voigtlaender [35] proposed a Siam redetection architecture, combined with a trajectory-based dynamic planning algorithm, using the first frame of annotation and the previous frame to predict the target for double detection. The complete history of tracked objects and potential interfering objects is modeled, and the tracked objects can be redetected after a long period of occlusion. Given that each frame of the image was double-checked, the algorithm complexity was high. Liu [36] designed an algorithm for long-term target tracking. When the tracking failed, EdgeBox was employed to generate suggested regions. Xiong [37] proposed a target scale and rotation parameter estimation method based on kernel correlation filtering for the problem of target scale and rotation changes caused by long-term target tracking. When the target tracking was lost, a target search method that combined the color histogram and variance was started to determine the possible position of the target in the current frame, but the distance between the suggested area and the real position was large. Gade [38] proposed a multiobject tracking algorithm suitable for team sports to solve the problem of target tracking in complex scenes with similar target colors and rapid actions.

Deep features are learned through a large number of training samples, which are more discriminative than manually designed features. Therefore, tracking methods using depth features can usually obtain a good effect with ease. The HCFT [44] method used the convolutional features of each layer of the convolutional neural network and further improved the tracking effect based on the correlation filter. To effectively contend with the change in object shape, an object tracking algorithm based on hierarchical convolution features [39], and a scale-adaptive kernel correlation filter [40-42] combined HOG and color features [43] were used to achieve good results in open datasets. Ma [44] improved tracking accuracy and robustness by pretraining deep CNNs, extracting the last three layers of convolutional features, and learning adaptive correlation filters. In recent years, as the number of layers in the backbone network of deep trackers has gradually deepened, online model updates have gradually increased the effect of tracking efficiency. Therefore, most deep trackers have not introduced online update strategies, but model updates are still an important way to maintain robustness of long-term tracking.

Different tracking algorithms still have unsatisfactory or low efficiency in solving some different complex scenes or difficult problems. Therefore, further research on target tracking technology is necessary to improve the tracking efficiency and effect. In this paper, we mainly focus on the problem of long-term tracking in a complex environment, especially when the target object is under full occlusion, exhibits deformation, and contains objects of similar colors. Two models are established by kernel correlation filtering: a target position estimation model and an adaptive scale model. A target position estimation model CF_1 is learned by extracting HOG and color features. The multifeature response complementary scheme is proposed, and the response of the two features is linearly weighted to realize the target position estimation. By extracting the HOG features of the target, an adaptive scale model CF_2 is learned and used to estimate the target scale transformation. The average peak-to-correlation energy (APCE) [45] is introduced to determine whether the target is blocked. When the target is occluded, the Kalman filter is used to estimate the occluded target position based on the target's historical path data. The key contributions of the proposed algorithm are summarized as follows:

- By extracting HOG and color features, a multifeature response complementary scheme is proposed based on the correlation filter framework, and the response of the two features is linearly weighted to realize the target position estimation. The complementary scheme can adaptively and perfectly combine these advantages of different features and solve the problem of long-term tracking in a complex environment, such as occlusion and background clutter.
- By extracting the HOG features of the target, an adaptive scale model CF_2 is learned and used to estimate the target scale transformation, which can prevent the the model drift.
- Two judgment criteria consisting of the response peak $f_{\max}$ and APCE are used to determine whether to update the target model, which can solve the tracking

drift problem caused by an incorrect template update mode.

The remainder of this study is organized as follows. Section II describes the principle of the nuclear correlation filter. Section III describes different feature extraction methods. Section IV describes a robust target tracking model and discusses the execution steps of the algorithm in detail. Section V verifies the effectiveness and robustness of the proposed algorithm through experiments in two aspects, namely, quantitative and qualitative analysis. Section VI summarizes the conclusions.

II. KERNEL CORRELATION FILTER

The filter is trained by cyclically shifting the target sample to obtain the negative sample, and the kernel function is designed to predict the target position to solve the problem of insufficient training samples. The cycle of the vector can be obtained by the permutation matrix P. Assuming vector $A=[a_1,a_2,\cdots,a_n]^T$, the permutation matrix P is expressed as follows:

$$P=\begin{bmatrix}0&0&\cdots&0&1\\1&0&\cdots&0&0\\0&1&\cdots&0&0\\&&\vdots&&\\0&0&\cdots&1&0\end{bmatrix}, PA=[a_n,a_1,a_2,\cdots,a_{n-1}]^T. \quad (1)$$

The two-dimensional image can be cyclically moved through the x-axis and y-axis, which can be realized by matrix Q, as described in (2), to acquire the movement of different positions. The matrix Q is shown as follows:

$$Q=\begin{bmatrix}0&1&\cdots&0&0\\0&0&\cdots&0&0\\&&\vdots&&\\0&0&\cdots&0&1\\1&0&\cdots&0&0\end{bmatrix}. \quad (2)$$

First, the target position of the initial frame is given, and rectangular areas are drawn around the center of the target. The rectangular area is 1.5 to 2 times the target area, and it can contain enough samples and some background information so that the trained filter template is more robust. The rectangular area can be expressed as $(W,H)=sizeof(target)*(1+padding)$, where W and H are the width and height of the area, respectively. The center position coordinate is $(W/2, H/2)$. A training set is formed by a cyclic shift of feature x, and each shift sample $x_{ij},(i,j)\in\{0,1,\cdots,W\}\times\{0,1,\cdots,H\}$ has an expected output. The expected value is produced by a Gaussian function, and its peak is the target center position. Then, the expected output of the training image is

$$y_{ij}=e^{-\frac{(i-W/2)^2+(j-H/2)^2}{2\sigma^2}}, \quad (3)$$

where σ is the core bandwidth. The center position has the highest expectation, which is $y_{(W/2,H/2)}=1$. As the shift operation progresses, the farther the deviation from the target is, the lower the expected output, and y_{ij} will change from 1 to 0. The goal of the KCF algorithm is to find a classifier ω with the same size as x to minimize the error between the output of the filter and the expected output. ω is defined as follows:

$$\omega^*=\arg\min_{\omega}\sum_{ij}\left\|f(x_{ij})-y_{ij}\right\|^2+\lambda\|\omega\|^2, \quad (4)$$

where λ is the regularization parameter to prevent overfitting, and $f(x)=\omega^T x$ is the output function. The solution describing the above problem is as follows:

$$\omega=(X^H X+\lambda I)^{-1}X^T Y. \quad (5)$$

Each row of $X=[x_1,x_2,\cdots,x_n]^T$ represents a vector. X^H represents the complex conjugate transpose matrix, that is, $X^H=(X^*)^T$. I is the identity matrix. Y is a column vector in which each element represents the expected output $y_{ij},(i,j)\in\{0,1,\cdots,W\}\times\{0,1,\cdots,H\}$. The circulant matrix X can be diagonalized in the Fourier domain, and the result in the Fourier domain can be obtained by using this feature:

$$\hat{\omega}=\frac{\hat{x}\odot\hat{y}}{\hat{x}^*\odot\hat{x}+\lambda}. \quad (6)$$

The addition and division in (6) are carried out by element, where $\odot$ means multiply by element. It is easy to find the time domain ω through the inverse Fourier transform, which is $\omega=\mathcal{F}^{-1}(\hat{\omega})$. The $\hat{x}$ in (6) is described as $\hat{x}=\sqrt{n}Fx$, and F is the discrete Fourier constant matrix. F is expressed by the following formula:

$$F=\frac{1}{\sqrt{n}}\begin{bmatrix}1&1&\cdots&1\\1&\omega&\cdots&\omega^{n-1}\\\cdots&\cdots&\vdots&\cdots\\1&\omega^{n-1}&\omega^{(n-1)(n-2)}&\omega^{(n-1)^2}\end{bmatrix}. \quad (7)$$

To improve the accuracy of tracking, the input data x_i are mapped to a high-dimensional space while solving ω by way of introducing a nonlinear mapping function $\varphi(x_i)$, and the nonlinear regression problem can be converted into a linear solution. ω can be expressed as $\omega=\sum_i\alpha_i\varphi(x_i)$. The kernel function $k(x_i,x_i')=<\varphi(x_i),\varphi(x_i')>$ is introduced; then, (6) is modified to the following form:

$$\hat{\alpha}=\frac{\hat{y}}{\hat{k}^{xx}+\lambda}. \quad (8)$$

The symbol ^ represents the result of the Fourier transform. k^{xx} represents the first-row vector of the circulant matrix, which is expressed as $k^{xx}=[k(x,x),k(x,Px),\cdots k(x,P^{n-1}x)]$. The Gaussian kernel is expressed as

$k(x,x')=\exp\left(-\frac{1}{\sigma^2}\|x-x'\|^2\right)$, then k^{xx} is rewritten as:

$$k^{xx'}=\exp\left(-\frac{1}{\sigma^2}(\|x\|^2+\|x'\|^2-2F^{-1}(\hat{x}^*\odot\hat{x}')\right). \quad (9)$$

The feature vector of the image is represented by z, and the response is calculated with the following formula:

$$f(z)=\omega^T z=\sum_{i=0}^{n-1}\alpha_i k(z,x_i). \quad (10)$$

To speed up the calculation, formula (10) is processed by diagonalization, and the result is expressed as:

$$\hat{f}(z)=\hat{k}^{xz}\odot\hat{\alpha}. \quad (11)$$

$\hat{f}(z)$ implements the inverse Fourier transform to find the position with the largest response as the final target area. $f(z)$ is calculated as:

$$f(z) = F^{-1}\left(\hat{f}(z)\right) = F^{-1}\left(\hat{k}^{xz} \odot \hat{\alpha}\right). \tag{12}$$

III. MULTICHANNEL FEATURE EXTRACTION

An effective target feature model can improve the accuracy and efficiency of object tracking. Long-term target tracking will encounter complex and changeable environments. Determining how to effectively express the characteristics of the tracking object is the key to tracking. Single manual features, such as color histograms, HOG, scale-invariant features, edge features and texture features, are often difficult to meet the needs of long-term tracking. However, the extraction model of depth features is generally more complicated, which will affect the efficiency of the algorithm. To balance the accuracy and execution efficiency, the proposed algorithm combines HOG and color features to represent the target object to solve the object tracking problems, such as severe occlusions, deformations, light changes, short-term out-of-sight occurrences, and similar colors, during long-term tracking.

The feature sample x of the multichannel image is composed of m channels in series, namely, $x = [x_1, \cdots x_m]$. The dot product of each channel is summed to obtain a multichannel Gaussian kernel, which is described as follows:

$$k^{xx'} = \exp\left(-\frac{1}{\sigma^2}(\|x\|^2 + \|x'\|^2 - 2F^{-1}(\sum_m \hat{x}^* \odot \hat{x}'))\right). \tag{13}$$

A. HOG FEATURE

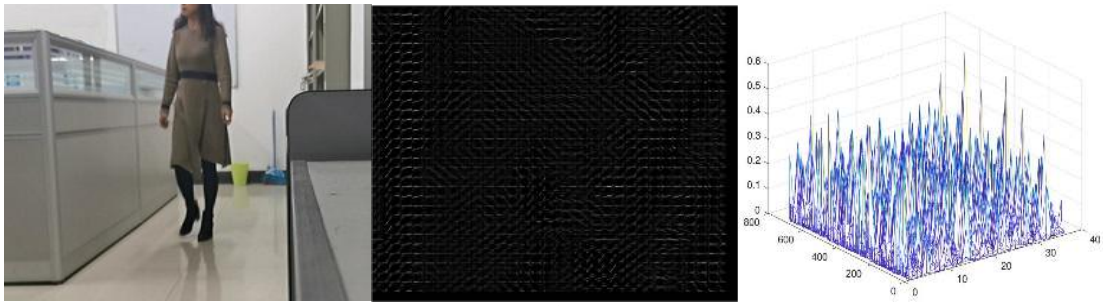
HOG describes the appearance and shape of local regions by calculating the distribution of directed gradients. These gradient descriptors are robust to changes in illumination. The image is divided into several connected regions, the directional gradient histogram within the region is calculated, and finally, the histogram of each region is combined to form the feature description of the entire image. The pixel value of a certain pixel (x, y) in the image is $H(x, y)$, and the horizontal gradients G_S and vertical gradients G_C are expressed as:

$$\begin{cases} G_S = H(x+1, y) - H(x-1, y) \\ G_C = H(x, y+1) - H(x, y-1) \end{cases}. \tag{14}$$

The gradient value and gradient direction of the pixel (x, y), respectively, are as follows:

$$\begin{cases} G(x, y) = \sqrt{G_S{}^2 + G_C{}^2} \\ \theta(x, y) = \arctan\dfrac{G_C}{G_S} \end{cases}. \tag{15}$$

Since the gradient intensity is greatly affected by changes in local illumination and contrast, it is also necessary to normalize the gradient. The abovementioned area is divided into multiple intervals, and then the gradient of the interval is calculated. Fig. 1 shows the extraction result of the HOG feature used to describe the target. Fig. 1(a) is the original image, Fig. 1(b) is the visual display of HOG features with clear gradient information, and Fig. 1(c) is a balanced histogram of HOG features.



(a) (b) (c)

FIGURE 1. HOG features. (a) Image. (b) Visual display of HOG features. (c) Histogram of HOG features. The HOG feature is robust to target tracking under illumination changes but is sensitive to object deformation.

B. COLOR FEATURE MODEL

The HOG feature contends with the serious deformation and occlusion of the target with difficulty, and it is also more sensitive to noise due to the nature of the gradient. Therefore, it is not sufficient to use HOG features to represent images in long-term target tracking. A color histogram is proposed to reduce drifting since the color-based tracker is easily drifted toward objects with a similar appearance. Therefore, the color histogram can handle scenes with similar objects.

The RGB color space is easily affected by changes in external lighting. Thus, converting the RGB color space into an HSV color space (as shown in Fig. 2(a)) that is not sensitive to changes in lighting is necessary. To improve the robustness of tracking and reduce the influence of changes in illumination brightness, only the H hue component is extracted to build a color feature histogram (as shown in Fig. 2(e)). The H hue component (as shown in Fig. 2(b)) is the basic attribute of color, and it represents the position of the spectrum color.

Assuming that the center of the target area is x_0 and that $\{x_1, x_2, \cdots x_n\}$ are the other pixels in the target area, n represents the total. The color feature value is set at $u = 1, 2, \cdots, m$ of each pixel, where m is the grading of the color feature. The normalized histogram can be expressed as follows:

$$h_u(x_0) = C\sum_{i=1}^{n} \delta(b(x_i) - u), \tag{16}$$

where C is the normalization coefficient, and $b(x_i)$ is the function that maps pixel x_i to the corresponding color feature. δ is the impulse function, and its value range is in [0,1]. The result of the color feature extraction is shown in Fig. 2.

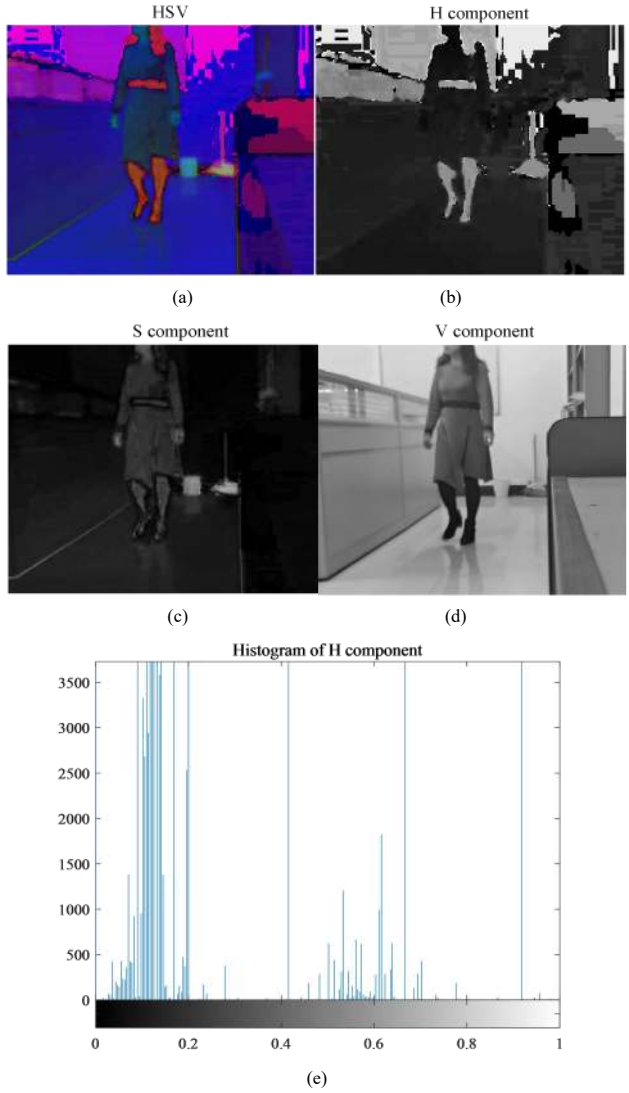


FIGURE 2. Color features. (a) HSV color space. (b) Hue component. (c) Saturation component. (d) Value component. (e) The histogram of the H hue component is used to build a color feature histogram. HSV color space can well improve the robustness of tracking and reduce the influence of changes in object deformation.

IV. ROBUST OBJECT TRACKING MODEL

A. OBJECT POSITION ESTIMATION MODEL BASED ON THE MULTIFEATURE RESPONSE

The kernel correlation filter based on HOG features is robust to motion blur and illumination changes but is very sensitive to target deformation. However, color features have strong robustness to target deformation. Therefore, a multifeature response complementary scheme is proposed to solve the problems of motion blur, illumination change, and target deformation during long-term tracking. According to HOG features and color features, two kernel correlation filters are separately learned. The responses of HOG features and color features are calculated, respectively, using (12), namely, $f_{hog}(z)$ and $f_{color}(z)$:

$$f_{hog}(z) = F^{-1}\left(\hat{k}_{hog}^{xz} \odot \hat{\alpha}_{hog}\right), \tag{17}$$

$$f_{color}(z) = F^{-1}\left(\hat{k}_{color}^{xz} \odot \hat{\alpha}_{color}\right). \tag{18}$$

The final filtered output is calculated as follows:

$$f(z) = \mu_{hog} f_{hog}(z) + \mu_{color} f_{color}(z). \tag{19}$$

Using (19) to achieve target position estimation, the target position estimation model CF_1 is also obtained. The contributions of the two characteristic responses are μ_{hog} and μ_{color} satisfying $\mu_{hog}+\mu_{color}=1$ which can be described as:

$$\begin{cases} \mu_{hog} = \dfrac{f_{hog}(z)}{f_{hog}(z)+f_{color}(z)} \\ \mu_{color} = \dfrac{f_{color}(z)}{f_{hog}(z)+f_{color}(z)} \end{cases}. \tag{20}$$

B. ADAPTIVE SCALE MODEL

The scale estimation model CF_2 can prevent the tracking drift problem caused by the deformation and blurring of the target in the target tracking process. When training the scale filter, only HOG features of the target position are extracted. Suppose the image size is (W,H); then, $x_{ij},(i,j)\in\{0,1,\cdots,W\}\times\{0,1,\cdots,H\}$ is obtained by cyclic shifting the sample. According to formula (4), the loss function of the scale estimation model can be obtained. Consequently, z is obtained by cyclic shifting the characteristic samples and then is substituted into (12) to obtain the maximum response of the kernel correlation filter model.

By assuming that the scale of the t-th frame target is $P*R$, the target scale pyramid with layers S is constructed around the target position for scale estimation. The scale p_m of any image patch in the target scale pyramid is expressed as:

$$p_m = a^m P \times a^m R. \tag{21}$$

Among them, $m\in\left\{\left[-\frac{S-1}{2}\right],\cdots,\left[\frac{S-1}{2}\right]\right\}$; a represents the scale factor of different scale layers. The image blocks obtained by (21) are readjusted to the same size as the target scale to construct a scale pyramid. The HOG features of each image patch in the scale pyramid are extracted and used to calculate the filter response. Then, the optimal size n of the target can be expressed as:

$$n = \arg\max_{m}\left\{\max f(z_1), \max f(z_2), \cdots, \max f(z_m)\right\}. \tag{22}$$

When the confidence corresponding to the optimal size n satisfies $\max f(z_n) > T_1$, the model CF_2 is updated; conversely, when the confidence satisfies $\max f(z_n) < T_1$, the model CF_2 is not updated.

The size of the target is expressed as:

$$(w_t, h_t) = \alpha(w_t^*, h_t^*) + (1-\alpha)(w_{t-1}, h_{t-1}), \tag{23}$$

where (w_t^*, h_t^*) represents the width and height of the candidate area with the maximum confidence, respectively, and (w_{t-1}, h_{t-1}) is the width and height of the previous tracking target. Additionally, α is the damping factor, which can control the smooth change of the center size and make the tracking more stable.

C. OCCLUSION JUDGMENT AND POSITION COMPENSATION

When the target is occluded, the traditional KCF algorithm cannot predict the target position. In this case, the Kalman filter is introduced to compensate for the target position to predict the target position. Note that the target model is not updated when the target is occluded.

1) OCCLUSION JUDGMENT

Due to the influence of background interference, when the target is occluded, the response value of the target is not necessarily low. In this case, it will mistakenly believe that the detection is correct and update the target template, failing in follow-up tracking. APCE [36] is introduced as a basis for judging whether there is occlusion, which is calculated as follows:

$$APCE = \frac{\left|f_{\max} - f_{\min}\right|^2}{mean\left(\sum_{w,h}(f_{w,h} - f_{\min})^2\right)}, \tag{24}$$

where $f_{\max}$, $f_{\min}$, and $f_{w,h}$ denote the maximum, minimum, and (w,h) response of $f(z)$, respectively. For sharper peaks and less noise, i.e., the appearance of the target in the detection scope, APCE will become larger and the response map will become smooth except for only one sharp peak, as shown in Fig. 3(b). When the target is occluded or tracking is lost, the response graph will oscillate significantly, and the value of APCE will also drop significantly (as shown in Fig. 3(d)). When APCE is greater than the threshold T_2, it means that the target has no occlusion or a small area; when APCE is less than the threshold T_2, it means that the target has a large area of occlusion or has full occlusion.

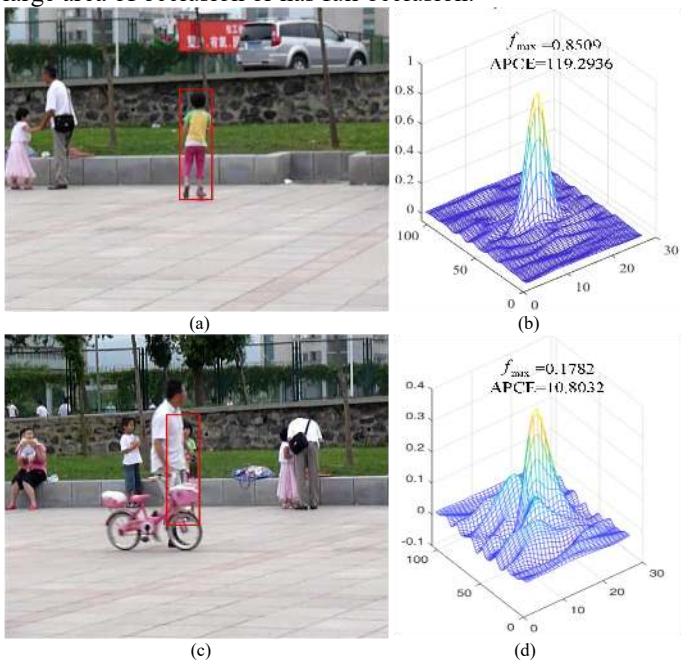


FIGURE 3. Response graph of occlusion and no occlusion. (a) No occlusion. (b) Response of no occlusion. (c) Occlusion. (d) Response of occlusion. When the target is occluded, $f_{\max}$ and APCE will also drop significantly.

2) POSITION COMPENSATION BASED ON KALMAN FILTER

The Kalman filter is applied to target trajectory tracking, especially in the case of target occlusion. Using the linear system state equation through the input and output observation data, the state of the system is optimally estimated. The state vector of the target is $X = [x, y, \tilde{x}, \tilde{y}]$, and the observed value is $Z = [x, y]^T$, where (x, y) is the target position coordinate, and $(\tilde{x}, \tilde{y})$ is the target speed.

Using the linear system state equation through the input and output observation data, the state of the system is optimally estimated. The system dynamic state equations at time k are X_k and Z_k:

$$X_k = AX_{k-1} + W_k, \tag{25}$$

$$Z_k = H_k X_k + V_k. \tag{26}$$

Among them, A is the transition matrix of the system state, and H_k is the observation matrix of the system. W_k and V_k are white noise sequences with a mean value of 0. The motion state of the target in two adjacent frames can be regarded as uniform linear motion, so A and H_k are defined as follows:

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{27}$$

Δt represents the interval between two adjacent frames. The proposed algorithm uses the optimal result $X_{k-1|k-1}$ of the previous state of the system to predict the current state of the system, and the prediction result is $X_{k|k-1}$, which is defined as follows:

$$X_{k|k-1} = AX_{k-1|k-1}. \tag{28}$$

The error estimate covariance matrix of the optimal result of the last state of the system is $P_{k-1|k-1}$. The covariance of $X_{k|k-1}$ can be updated to:

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q, \tag{29}$$

where Q is the covariance of the system. According to (28) and (29), the optimal prediction result $X_{k|k}$ of the current state is obtained, and $K_g(k)$ is the Kalman gain. $X_{k|k}$ and $K_g(k)$ are, respectively, defined as follows:

$$X_{k|k} = X_{k|k-1} + K_g(k)(Z_k - HX_{k|k-1}), \tag{30}$$

$$K_g(k) = P_{k|k-1}H^T / HP_{k|k-1}H^T + R. \tag{31}$$

Finally, to achieve the purpose of continuous tracking, the covariance $P_{k|k}$ of $X_{k|k}$ in the state k is updated:

$$P_{k|k} = (I - K_g(k)H)P_{k|k-1}. \tag{32}$$

The target position predicted by the target model is $(\hat{x}, \hat{y})$, and the target position predicted by the Kalman filter is $(\tilde{x}, \tilde{y})$. The target position compensation strategy is expressed as follows:

$$(x, y) = \beta(\hat{x}, \hat{y}) + (1 - \beta)(\tilde{x}, \tilde{y}), \tag{33}$$

where β is the position compensation weight. When the target is not occluded, the value of β is 1; when the target is occluded, the value of β is 0. That is, when the target is occluded, the Kalman filter is used to predict the target position.

D. TARGET MODEL UPDATE BASED ON HIGH CONFIDENCE

Some existing trackers update object models [13, 16] at each frame without considering whether the detection is accurate or not. However, when the target is severely occluded, the target model is still updated, which is equivalent to updating the background as the target, resulting in template drift. Therefore, determining how to update the target model is the

key to target tracking. The peak value and the fluctuation of the response map, as shown in Fig. (3), can reveal the confidence degree about the tracking results to some extent. The ideal response graph should have a peak at the target position and smoothly drop at other positions. When there is more than one similar object in the scene, the tracker treats the similar target as the background. However, affected by background interference, the location with the highest response may not necessarily be the target. Consequently, it is inaccurate to update the template only based on the maximum response value. Unimodal detection will regard the highest response as the target leading to false detection. The proposed target detection will redetect the areas centered at other peaks to find the maximum peak among these response maps as the correct subfigure and locate the correct position of the target. Therefore, two discrimination methods are used as the basis for the judgment of the target model, which ensures the accuracy of the tracker. The first one is the maximum response score $f_{\max}$, which is defined as follows:

$$f_{\max} = \arg\max \left\{ f(z) = F^{-1}\left(\hat{f}(z)\right) \right\}. \tag{34}$$

The other one is the APCE measure defined as (24). The appearance, background, and environment of the target may be constantly changing in the process of target tracking, and the designed model should be able to adaptively adjust the parameters to continuously adapt to the complex and changeable environment. The training sample of frame t-1 is recorded as x_{t-1}, and the model parameter of ridge regression is α_{t-1}. The new samples x_t and model parameters α_t are obtained at the t-th frame. If the response peak value $f_{\max}$ and APCE of the current frame are greater than the threshold T_1 and T_2, the target model is updated. By assuming the learning rate is γ, the update strategy is as follows:

$$x_t = (1-\gamma)x_{t-1} + \gamma x_t, \tag{35}$$

$$\alpha_t = (1-\gamma)\alpha_{t-1} + \gamma\alpha_t. \tag{36}$$

E. ADAPTIVE OBJECT TRACKING ALGORITHM

The use of a single feature makes it difficult to meet the needs of long-term target tracking in a complex environment. Combining HOG features and color features to represent the object can solve long-term tracking problems, such as occlusion, deformation, light changes, short-term out-of-sight occurrences, and similar colors. The feature extraction module is introduced in Section III. To improve the robustness of long-term tracking, the target position estimation model CF_1 and scale model CF_2 are trained. The correlation filter CF_1 (see Section IV-A for details) is used to estimate the displacement of the target during the movement and predict the position of the target. The correlation filter CF_2 (see Section IV-B for details) is used to estimate the optimal scale of the target. APCE is used to determine whether the target is occluded, and the Kalman filter is applied for position compensation, which solves the tracking drift problem caused by occlusion (see Section IV-C for details). Two template update methods are used to update the template (see Section IV-D for details): maximum response $f_{\max}$ and APCE, which solves the problem of subsequent tracking failures caused by incorrect updating of the template. The proposed algorithm model is shown in Fig. 4. The main working steps of the algorithm are shown as Algorithm 1:

Algorithm 1. Proposed Tracking Algorithm

Input: Image I_t, previous object position: $(x_{t-1}, y_{t-1}, w_{t-1}, h_{t-1})$

Output: Estimated object position: (x_t, y_t, w_t, h_t)

1: Crop the search window in I_t centered at (x_{t-1}, y_{t-1}) and extract HOG features and color features.

2: Compute the response map $f_{hog}(z)$ using (17).

3: Compute the response map $f_{color}(z)$ using (18).

4: Compute the final response map $f(z)$ using (19) and APCE using Eq. 22.

5: if $(f_{\max} > T_1)$ && $(APCE > T_2)$, then

6: Estimate the current position $(\hat{x}_t, \hat{y}_t)$ and update the position estimation filter CF_1.

7: else

8: Compensate position using (31) and obtain the new current position (x_t, y_t).

9: end if

10: Construct a target pyramid for scale estimation around $(\hat{x}_t, \hat{y}_t)$ and obtain the best scale n

11: if $\max f(z_n) > T_1$, then

12: Update the scale model CF_2 and obtain estimated position (x_t, y_t, w_t, h_t).

13: end if

V. RESULT ANALYSIS AND DISCUSSION

We evaluated the proposed tracker by comparing it with some state-of-the-art trackers, including CSK [17], KCF [20], SRDCF [22], SRDCF* [23], CLRT [31], and HCFT [44] on two widely used datasets TB-128 [46] and OTB2015[47]. The TC-128 benchmark contains 128 color video sequences with 11 annotated attributes, and it aims at analyzing the impact of color information on tracking. The OTB2015 contains more than 100 manually annotated datasets with 11 attributes that cover various challenging factors, including scale variation (SV), illumination variation (IV), occlusion (OCC), motion blur (MB), deformation (DEF), fast motion (FM), out-of-plane rotation (OPR), out-of-view (OV), in-plane rotation (IPR), background clutter (BC) and low resolution (LR), which are summarized in Table I.

A. IMPLEMENTATION DETAILS

The proposed tracker is implemented in MATLAB R2018b on an Intel Core i7-8550U 2.0 GHZ CPU with 8 GB RAM. The optimization takes 10 iterations in the first frame and 3 iterations for each online update. The number of scales is 33, and a scale factor of 1.02 is used in the scale model. To compare each algorithm fairly, the same parameters are used for experiments, as shown in Table II.

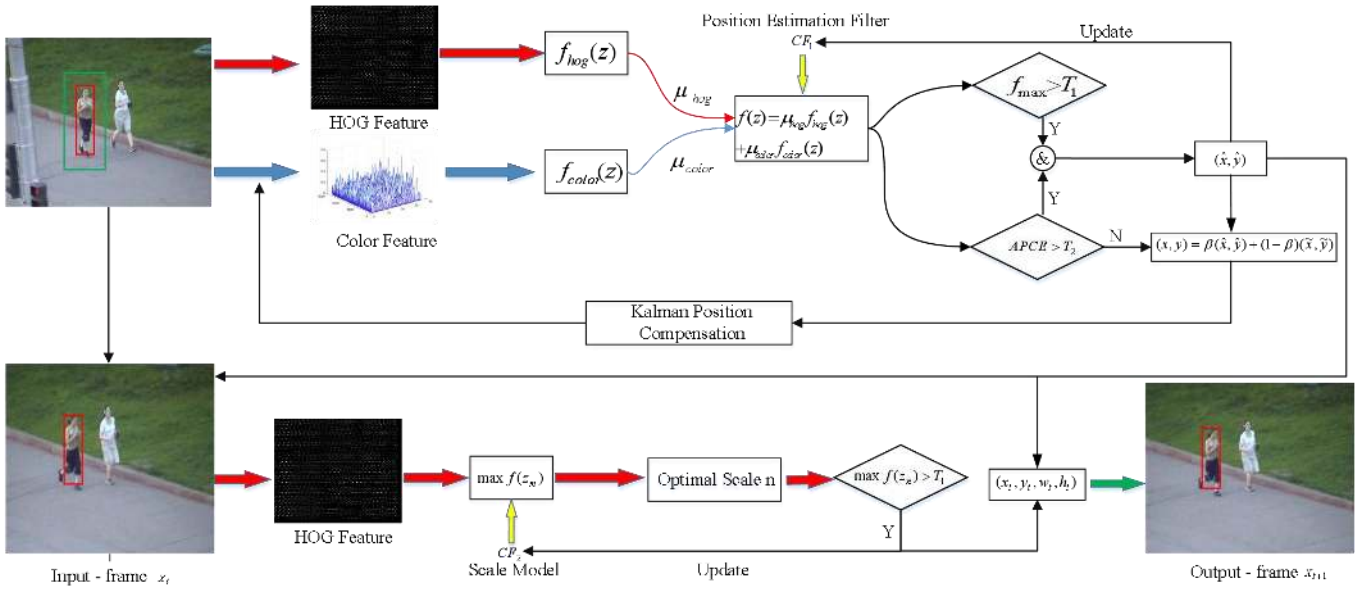


FIGURE 4. Algorithm model. HOG and color features for the prediction are extracted around the location in the previous image and used to calculate the response maps using correlation filter. The features have different discriminatory powers in different scenarios, which are combined with the fusion model. Combining the scale model and high-confidence update strategy, the object target can be located accurately.

TABLE I
PART OF TEST VIDEOS CATEGORIZED WITH 11 ATTRIBUTES

Sequence	SV	IV	OCC	MBB	DEF	FM	OPR	OV	IPR	BC	LR
Basketball		√	√		√		√				
Boy	√			√		√	√		√		
Car4	√	√									
Deer				√		√			√	√	√
Girl	√		√				√		√		
Jogging			√		√		√				
Lemming	√	√	√			√	√	√			
MotorRolling	√	√		√		√			√	√	√
Singer1	√	√	√		√						
SUV			√					√	√		
Walking	√		√								√
Woman	√	√	√	√	√	√	√				

TABLE II
PARAMETER SETTINGS

Parameter	Value
Regularization term λ	10^{-3}
Padding	0.5
Learning rate γ	0.015
$f_{\max}$ threshold T_1	0.3
APCE threshold T_2	20
Gaussian kernel bandwidth σ	0.5

B. EVALUATION INDEX

1) DISTANCE PRECISION RATE (DPR)

The tracking algorithm estimates the Euclidean distance between the center point of the target position and the center point of the manually marked target. Otherwise, the smaller the Euclidean distance is, the higher the tracking accuracy. DPR represents the percentage of frames whose center location errors are smaller than a given threshold. With different thresholds, the ratios are different such that a curve can be obtained, and the threshold is set to 20 pixels.

2) OVERLAP SUCCESS RATE (OSR)

OSR represents the percentage of frames whose overlap scores with the ground truth are larger than another given threshold. The overlap rate of the bounding box S_1 of the target position and the target position's true bounding box S_2 is used to estimate the tracking algorithm, which is calculated by the following formula:

$$S = \frac{\left|Area(S_1 \cap S_2)\right|}{\left|Area(S_1 \cup S_2)\right|}. \tag{37}$$

3) CENTER POSITION ERROR (CPE)

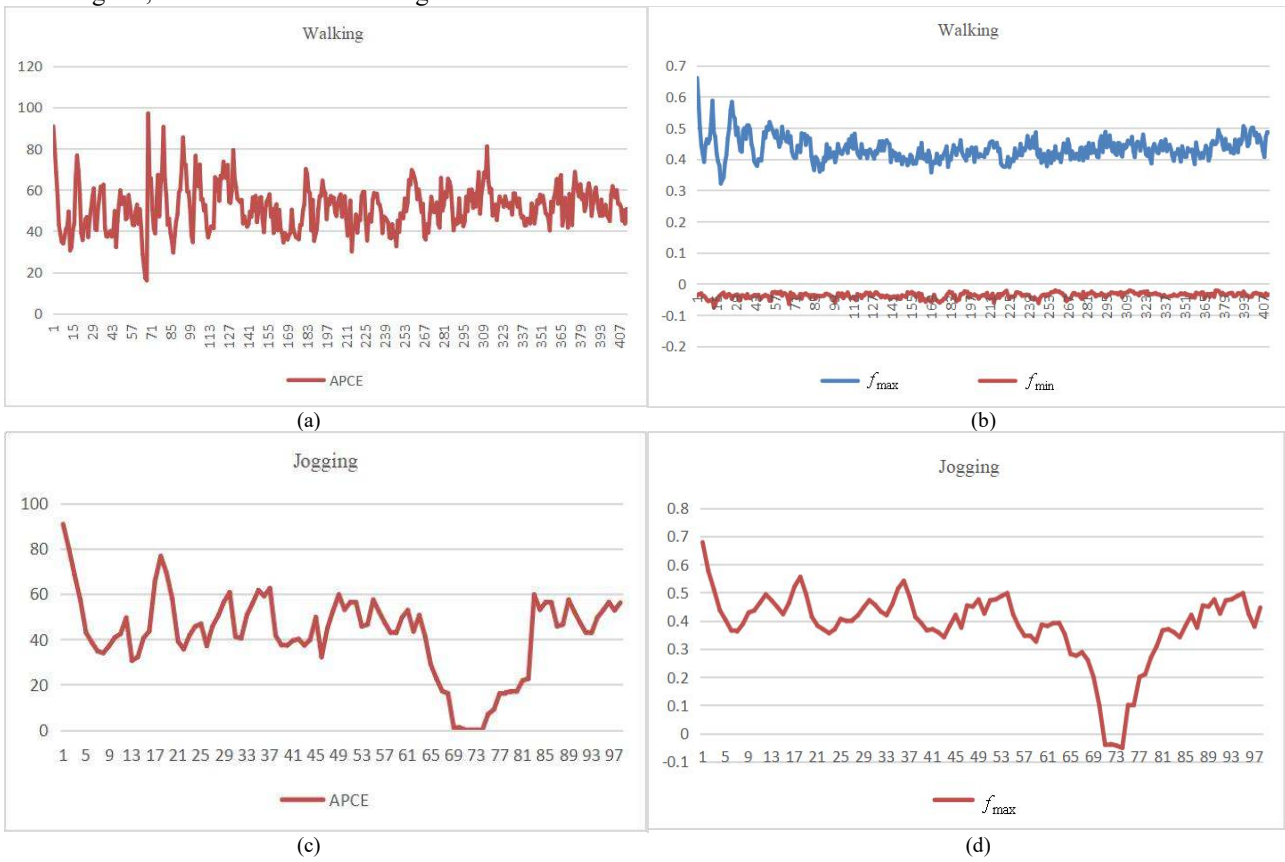
To analyze the accuracy and stability of the proposed algorithm, the concept of center position error is introduced in the experimental analysis. The center position error refers to the Euclidean distance between the estimated position (x', y') obtained by iteration and the true position (x, y), which can be calculated with (38):

$$D = \sqrt{(x - x')^2 + (y - y')^2}. \tag{38}$$

C. ANALYSIS OF MAXIMUM, MINIMUM, AND APCE OF THE RESPONSE

In this section, we analyze the maximum, minimum and APCE of response maps to verify the effectiveness of the proposed algorithm. The maximum response $f_{\max}$ and APCE are compared with preset thresholds T_1 and T_2, respectively, to determine whether the tracking is successful or has failed. Fig. 5(a) and Fig. 5 (b) are the APCE and the maximum and minimum of the Walking image sequence, respectively. In the whole tracking process, $f_{\max}$ is greater than 0.3,

satisfying T_2 , the condition of being greater than the threshold T_1 . APCE is greater than 16, satisfying the condition of being greater than the threshold. The minimum value of APCE at 16.0741 appears in frame 68. Approximately 87 frames are affected by the occlusion of the telephone poles, and APCE also reached 29.5708. The above situation shows that the tracking effect of the Walking image sequence is good, and there is no tracking drift and failure. Fig. 5(c) and Fig. 5(d) are the APCE and the maximum of the image sequence Jogging, respectively. Figs. 5(c-d) show that $f_{\max}$ and APCE from the 69th-76th frame do not meet the threshold T_1 and T_2 . In this case, the template should not be updated, but the Kalman filter is used for position compensation to predict the target position. After the target reappears from frame 77 onward, the normal tracking state is restored.



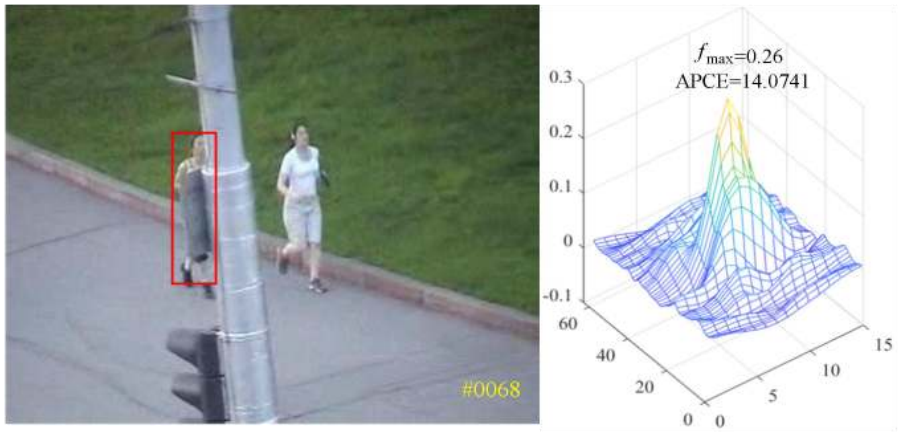
(a) (b)

(c) (d)

FIGURE 5. Response of frame. (a) APCE of Walking. (b) Maximum and minimum of Walking. (c) APCE of Jogging. (d) Maximum and minimum of Jogging. The change curve clearly shows the response change of each frame in the video sequence (e.g., Walking and Jogging), which also verifies the rationality of the proposed scheme.

Fig. 6 is the response diagram of the target in different frames. Fig. 6(b) shows that when the target is not blocked, the response value is the largest at the target position, and the entire response curve has a relatively small oscillation. In Fig. 6(d), the target is severely occluded, the response diagram exhibits a certain shock, and some small peaks appear near the target. When $f_{\max}$ and APCE do not satisfy the threshold, the Kalman filter is called for position compensation, and the position and scale filters are not updated.

(a) (b)

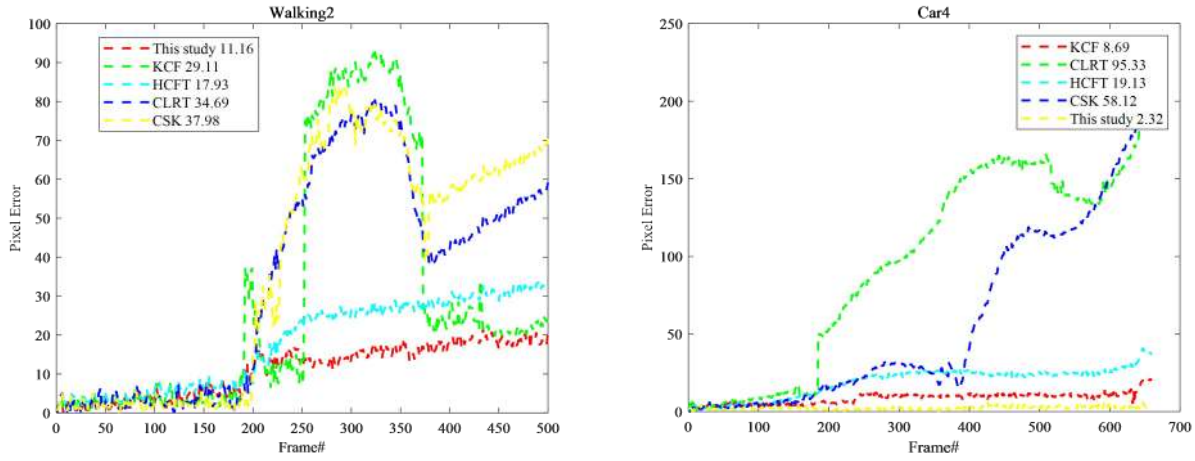


(c) (d)

FIGURE 6. **Response of frame in the Jogging sequence of OTB2015. (a) Frame #0037. (b) Response of object in frame 37. (c) Frame #0068. (d) Response of object in frame 68. The red bounding boxes indicate the tracking results of the proposed approach with the high-confidence update strategy. When the target is occluded, $f_{\max}$ and APCE decrease obviously, and the template update is not carried out.**

D. CENTER POSITION ERROR ANALYSIS

In this section, we analyze the proposed tracker by comparing the center position error with CSK [17], KCF [20], CLRT [31], and HCFT[44] algorithms on the OTB2015 benchmark. The smaller the D is, the higher the accuracy and stability of the algorithm. Fig. 7(a) is the result of the center position error in the Walking2 dataset. The CPE of the proposed algorithm maintains a low value, and the maximum is only 21. Starting from frame 370, the target is blocked, and the CPEs of KCF, CLRT, and CSK algorithms exceed 70, resulting in tracking drifts. After the target reappeared, the CPE of the KCF algorithm decreases to 20, and the tracking is resumed. However, the CPEs of CSK and CLRT algorithms are still greater than 50, and the tracking cannot be correctly completed in the subsequent frame. Fig. 7(b) is the result of the center position error in the Car4 dataset. The average CPE of the proposed algorithm is only 2.32. The average CPE of the KCF algorithm is also low, which is 8.69. However, the average CPEs of CLRE and CSK algorithms exceed 58, which are affected by light changes, resulting in tracking failure.



(a) (b)

FIGURE 7. ***Comparison results of center position error with state-of-the-art trackers (KCF, HCFT, CLRT, and CSK) in different test videos. (a) Comparison result of the CPE in Walking2. (b) Comparison result of center position error in Car4. Our approach provides consistent results with the lowest pixel error in challenging scenarios, such as illumination variation, background clutter and target rotations.***

E. QUANTITATIVE ANALYSIS

In this section, to validate the effectiveness of the proposed tracker, we make comparisons with some state-of-the-art trackers, including CSK [17], KCF [20], SRDCF [22], SRDCF* [23], CLRT [31], and HCFT[44], on the OTB2015 [47] dataset.

The images of OTB2015 are different in contrast, background interference, and image noise. According to the response of the HOG feature and color feature, the feature contribution is adaptively adjusted to achieve a better tracking effect. The Car1 video sequence has 1020 frames, which have the characteristics of scale variation, fast motion, illumination variation, and low resolution. Fig. 8 shows the HOG feature (red curve) and color feature (blue curve) contribution distribution of 500 frames in the Car1 video sequence. The changing curve shows that the contribution of HOG features is high. When the object deforms in some frames, the contribution of color features is high, which compensates for the shortcomings of HOG features that are sensitive to deformation.

The tracking speed is described in terms of FPS (frame per second) and is shown in Table III. Although our tracker is not the fastest one, the accuracy of our tracker is better than some of the most advanced trackers and can meet the basic real-time tracking requirements

TABLE III
COMPARISON OF THE TRACKING SPEED (FPS) ON THE OTB2015 DATASET

Trackers	Ours	CSK	KCF	SRDCF	SRDCF*	CLRT	HCFT
Tracking speed	55	53	109	6	4	50	15

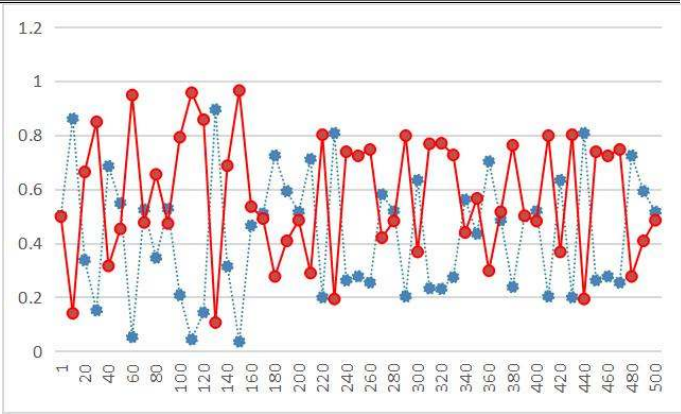


FIGURE 8. Feature contribution. The red curve and blue curve represents the contribution of the HOG feature and color feature, respectively. These two features play a complementary role in different scenarios.

The evaluation index mentioned in Section B belongs to the one-pass evaluation (OPE) standard. The tracking algorithm may be more sensitive to the initial position given in the first frame, and starting at a different position will have a greater impact on the tracking. After disrupting the initial state in time (starting from different frames) and space (different target positions), the temporal robustness evaluation (TRE), and spatial robustness evaluation (SRE) are obtained, respectively. Fig. 9 shows the comprehensive statistical results of the proposed algorithm and the comparison algorithm on OTB2015.

Under the three evaluation criteria of OPE, TRE, and SRE, the distance precision rate and overlap success rate of the proposed algorithm are the highest. Particularly when using the OPE evaluation criteria, the distance precision rate of the proposed algorithm is 0.899, which is 9.63% higher than that of the second-ranked HCFT algorithm (0.821). The overlap success rate is 0.635, which is 9.29% higher than that of the second-ranked HCFT (0.581).

Fig. 10 shows the tracking accuracy and tracking speed of the proposed tracker and other compared trackers on the OTB2015 dataset. From this figure, we can see that our tracker has achieved favorable tracking accuracy. However, the tracking speed still needs to be improved.

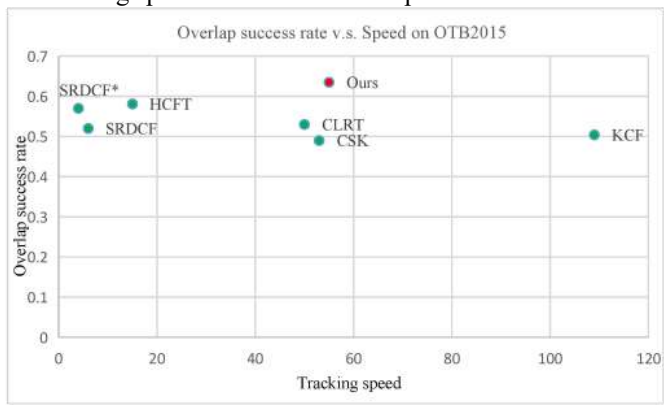
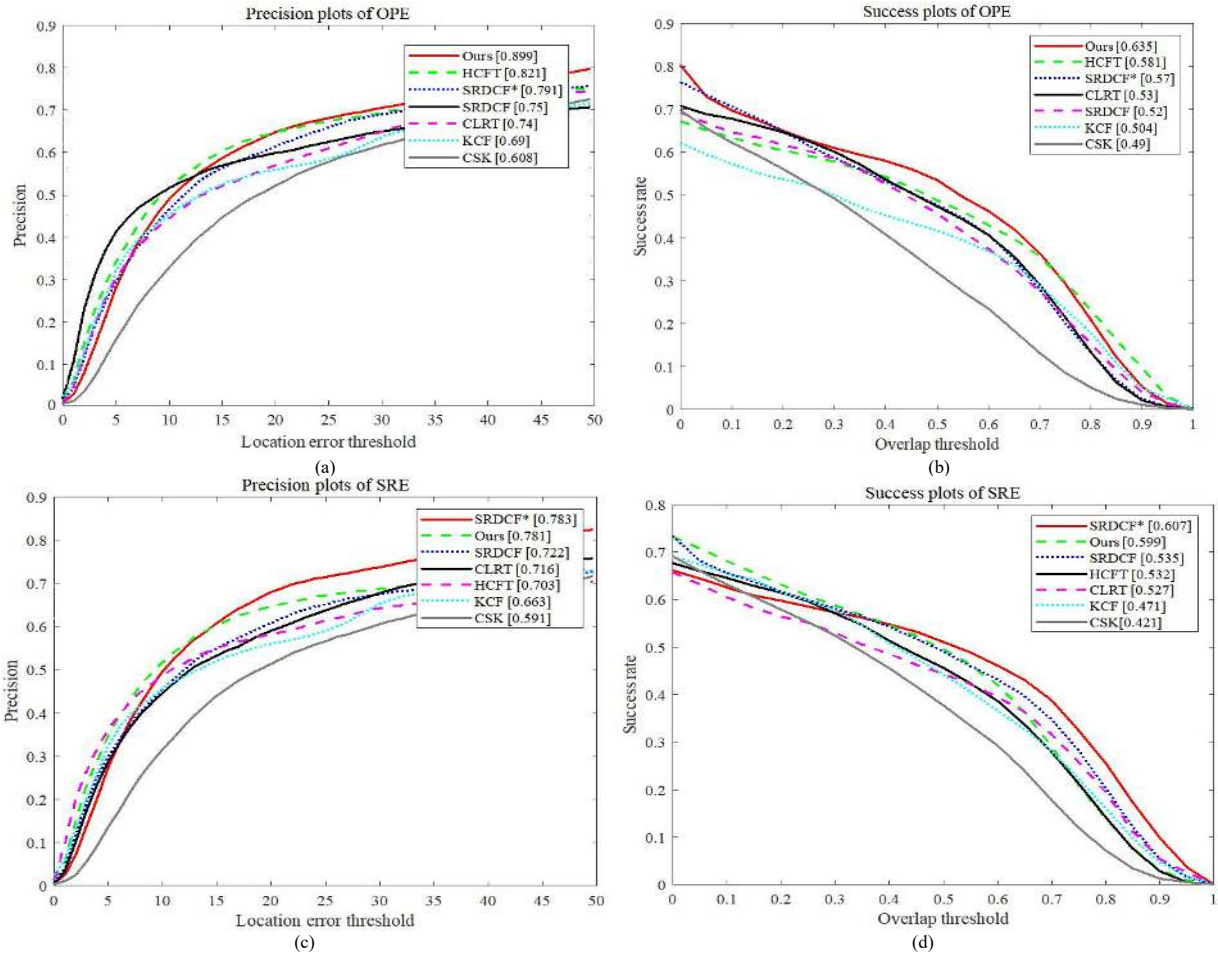


FIGURE 10. The comparison of OSR and tracking speed on the OTB2015 dataset. The horizontal and vertical coordinates represent the tracking speed and overlap success rate, respectively.



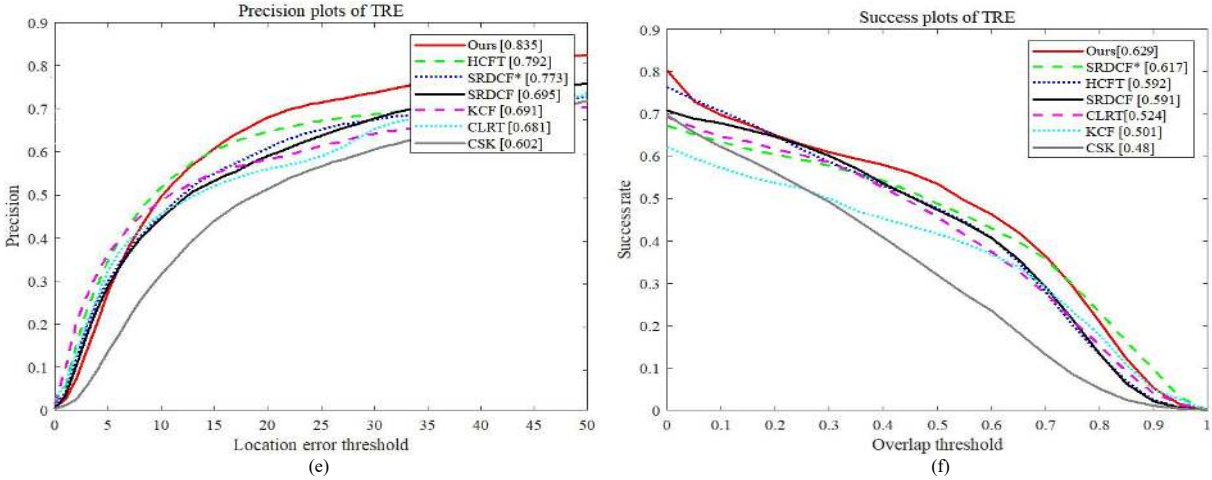


FIGURE 9. OPE, SRE and TRE precision and success plots on OTB2015. (a) Precision plots of OPE. (b) Success plots of OPE. (c) Precision plots of SRE. (d) Success plots of SRE. (e) Precision plots of TRE. (f) Success plots of TRE. The numbers in the legend indicate the representative precision at 20 pixels for precision plots, and the average area-undercurve scores for success rate plots.

F. QUALITATIVE ANALYSIS

To demonstrate the effectiveness and robustness of the proposed tracker indirectly, we compare it with four state-of-the-art trackers, i.e., CSK [17], KCF [20], CLRT [31], and HCFT [44] on some selected challenging sequences of OTB2015. The images in the MotorRolling video sequence have problems such as out-of-plane rotation, illumination variation, deformation, and rotation. The images in the Jogging video sequence have problems such as occlusion, deformation, fast motion, and low resolution. The images in the Walking video sequence have problems such as deformation, low resolution, and dim light. The images in the Car4 video sequence have problems such as uneven illumination, similar colors, scale changes, fast motion, and background doping.

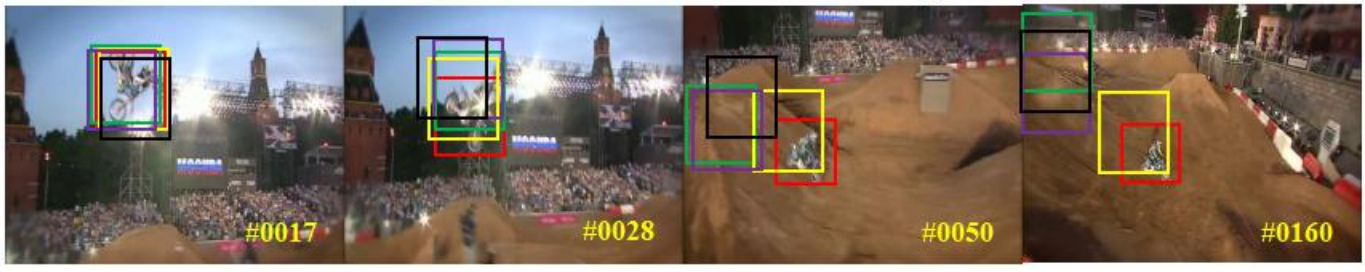
Fig. 11 shows the qualitative comparison of the five trackers on these challenging sequences. When the target is severely occluded and occluded for a long time, the proposed algorithm and the HCFT algorithm achieve better tracking results, as shown in Fig. 11(a). Other comparison algorithms show tracking drift and can recover redetection. The object in the Jogging video sequence shown in Fig. 11(b) is also severely occluded, and the shape of the target is constantly changing. The proposed algorithm and CSK have achieved better tracking results. After the object reappears, the proposed algorithm immediately resumes normal tracking, while the KCF algorithm has a short-term tracking loss, but the tracking ability is quickly restored. The CLRT algorithm shows tracking drift in the tracking results of the Walking video sequence, as shown in Fig. 11(c), and other comparison algorithms achieve better tracking results. The object in the Car4 video sequence shown in Fig. 11(d) has a fast speed and large illumination changes, which leads the proposed algorithm to some errors in tracking when the target changes lanes; however, no failure occurs. Other comparison algorithms show varying degrees of drift and even lead to tracking failures. The qualitative analysis results show that the advantages of the proposed algorithm are obvious, which further verifies the effectiveness of the redetection module in the tracking process.

G. EXPERIMENT ON TC-128

In this section, we use the TC-128 [46] dataset to validate the performance of the proposed tracker. The comparison with some state-of-the-art trackers, including CSK [17], KCF [20], SRDCF [22], SRDCF* [23], CLRT [31], and HCFT[44], is shown in Table IV. Among them, the proposed tracker obtains the best distance precision rate (73%), and SRDCF* obtains the best overlap success rate (53.4%). Compared with HCFT, the proposed tracker achieved significant improvements, which clearly shows the benefits of using the high-confidence update strategy.

TABLE IV
DPR AND OSR COMPARISON OF THE ABOVEMENTIONED TRACKS ON TC-128

Algorithms	Ours	CSK	KCF	SRDCF	SRDCF*	CLRT	HCFT
DPR	0.73	0.54	0.549	0.696	0.729	0.591	0.68
OSR	0.525	0.407	0.387	0.509	0.534	0.483	0.492



(a)

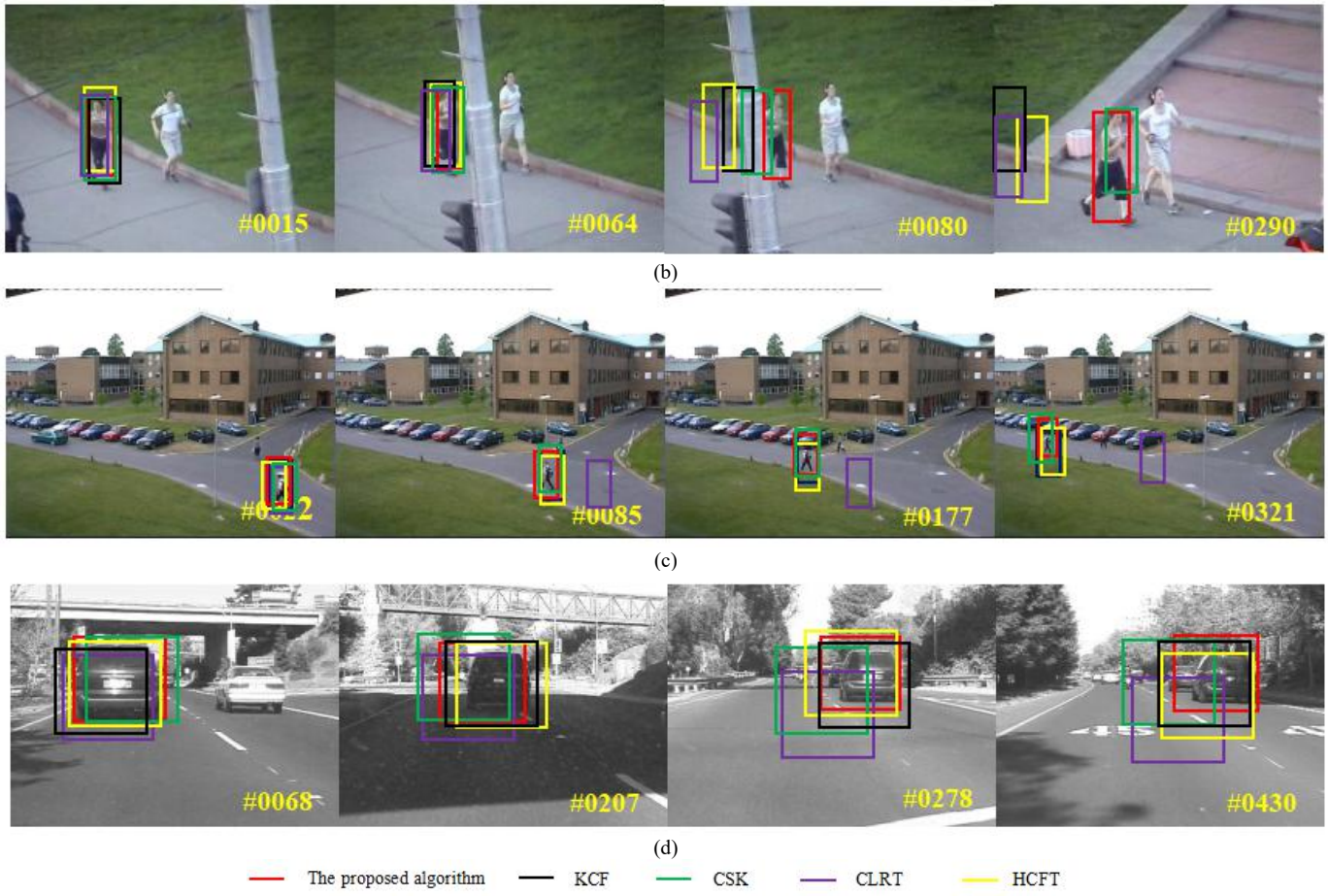


FIGURE 11. Qualitative comparison of our tracker and other representative trackers (CSK [17], KCF [20], CLRT [31], and HCFT [44]) on some visual object tracking sequences. (a) Test video of MotorRolling. (b) Test video of Jogging. (c) Test video of Walking. (d) Test video of Car4. The proposed model provides consistent results in challenging scenarios, such as occlusions, illumination variation, fast motion, background clutter, and target rotations.

H. DISCUSSION

In this section, we mainly discuss the impact of several factors that are essential to our tracking performance and speed using the OTB2015 dataset, including the regularization term λ, $f_{\max}$ threshold T_1, and APCE threshold T_2.

1) ANALYSIS OF THE REGULARIZATION TERM

The regularization term is used to prevent model overfitting, and its value directly affects the tracking performance. If λ is too small, the regularization term is inactive. In contrast, if λ is too large, the regularization term dominates the overall error. The DPR and OSR achieved by our tracker with different λ values are listed in Table V. It shows that the best performance is achieved at approximately $\lambda = 10^{-3}$.

TABLE V
DPR AND OSR RESULTS ACHIEVED WITH DIFFERENT VALUES OF THE REGULARIZATION TERM λ

Regularization term λ	DPR	OSR
1	0.726	0.552
10^{-1}	0.801	0.603
10^{-2}	0.826	0.623
10^{-3}	0.899	0.635
10^{-4}	0.845	0.628
10^{-5}	0.792	0.582

2) ANALYSIS OF THRESHOLDS T_1 AND T_2

The maximum response threshold T_1 and APCE threshold T_2 are used to determine when to update the tracker model. If T_1 and T_2 are too small, the tracker can easily drift due to noisy updating. However, if T_1 and T_2 are too large, the tracker cannot adapt to the appearance changes of the tracking object. Table VI shows the DPR and OSR results achieved by the proposed tracker with different values of T_1 and T_2.

TABLE VI
DPR AND OSR RESULTS ACHIEVED WITH DIFFERENT VALUES OF T_1 AND T_2

$f_{\max}$ threshold T_1	APCE threshold T_2	DPR	OSR
0.6	50	0.626	0.532
0.5	40	0.821	0.594
0.4	30	0.887	0.613
0.3	20	0.899	0.635
0.25	15	0.885	0.578
0.2	15	0.632	0.482

VI. CONCLUSIONS

To solve the problem of long-term moving object tracking in complex scenes, a scale-adaptive correlation filtering algorithm combined with the Kalman filter is proposed to achieve good performance on TC-128/OTB2015 benchmarks. The proposed algorithm started with the extraction of robust features and analyzed the features with different discrimination capabilities. The following conclusions can be drawn:

- A target position estimation model CF_1 and an adaptive scale model CF_2 are learned to locate the target position and estimate the target scale transformation, respectively.
- Two judgment criteria are used to determine whether to update the target model, which can solve the tracking

drift problem caused by the incorrect template update mode.

- Aimed at the tracking failure caused by the severe occlusion of the target, the Kalman filter is introduced to correct the position of the tracking target to ensure the accuracy of the tracking result.

The proposed algorithm shows good tracking performances when contending with challenging video scenes, such as occlusion, scale conversion, and uneven illumination. However, its tracking ability is slightly weaker under video sequences with attributes of rotation and low resolution. In future studies, the algorithm will be improved by extracting image hierarchical convolution features.

REFERENCES

[1] L. Meng, and X. Yang, "A survey of object tracking algorithms," *Acta Autom. Sin.*, vol. 45, no. 7, pp. 1244−1260, Jan. 2019.

[2] Q. Liu, Y. Wang, Y. Zhang, and M. Yin, "Research progress of visual tracking methods based on correlation filter," *Acta Autom. Sin.*, vol. 45, no. 2, pp. 265−275, Feb. 2019.

[3] C. Liu, W. Zhao, P. Liu, and X. Tang, "Selection, tracking and updating of auxiliary targets in target tracking," *J. Autom.*, vol. 44, no. 7, pp. 1195−1211, Dec. 2018.

[4] L. Chen, D. Jiang, H. Song, P. Wang, R. Bao, K. Zhang, and Y. Li, "A lightweight end-side user experience data collection system for quality evaluation of multimedia communications," *IEEE Access,* vol. 6, no. 1, pp. 15408−15419, Apr. 2018.

[5] L. Chen, and L. Zhang, "Spectral efficiency analysis for massive MIMO system under QoS constraint: an effective capacity perspective," *Mobile Networks Appl.*, Jan. 2020. DOI: 10.1007/s11036-019-01414-4.

[6] G. Bradski, and S. Clara, "Computer vision face tracking for use in a perceptual user interface," *Intel Technol. J.*, vol. 2, pp. 1−15, Oct. 1998.

[7] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust mean-shift tracking with corrected background-weighted histogram," *IET Comput. Vis.*, vol. 6, no. 1, pp. 62–69, Jan. 2012.

[8] X. Li, G. Xu, X. Yang, and G. Zhao, "A multi feature tracking algorithm based on camshift," *Comput. Digital Eng.*, vol. 48, no. 1, pp. 73−77, Jan. 2020.

[9] G. Zhao, S. Zhuo, and X. Xu, "Multi-object tracking algorithm based on kalman filter," *Comput. Sci.*, vol. 45, no. 8, pp. 253−257, Aug. 2018.

[10] D. Yuan, X. Lu, D. Li, Y. Liang, and X. Zhang, "Particle filter re-detection for visual tracking via correlation filters," *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 14277−14301, Jun. 2019.

[11] D. Yuan, W. Kang, and Z. He, "Robust visual tracking with correlation filters and metric learning," *Knowl.-Based Syst.*, vol. 195, pp. 105697, May 2020.

[12] M. Majd, and R. Safabakhsh, "Correlational convolutional LSTM for human action recognition," *Neurocomputing*, vol. 396, pp. 224−229, Apr. 2020.

[13] W. Howard, S. Nguang, and J. Wen, "Robust video tracking algorithm: a multi-feature fusion approach," *IET Comput. Vis.,* vol. 12, no. 5, pp. 640−650, Feb. 2018.

[14] M. Razzaq, J. Quero, I. Cleland, C. Nugent, U. Akhtar, B. Ali, U. Rehman, S. Lee, "uMoDT: An unobtrusive multi-occupant detection and tracking using robust kalman filter for real-time activity recognition," *Multimedia Syst.*, vol. 26, no. 5, pp. 553–569, Jun. 2020.

[15] Z. Zhang, and Y. Wang, "SiamRPN target tracking method based on kalman filter," *Intell. Comput. Appl.*, vol. 10, no. 3, pp. 44−50, Jul. 2020.

[16] D. Bolme, J. Beveridge, B. Draper, and Y. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Conf. Comput. Vis. Pattern Recognit.*. San Francisco, California, USA, 2010, pp. 2544–2550.

[17] J. Henriques, R. Caseiror, P. Martins, and P. Jorge, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, 2012, pp. 702–715.

[18] J. Henriques, J. Carreira, C. Rui, and B. Jorge, "Beyond hard negative mining: efficient detector learning via block-circulant decomposition," in *IEEE Int. Conf. Comput. Vis.*, Sydney, Australia, 2013, pp. 2760–2767.

[19] M. Danelljan, F. Khan, M. Felsberg, and J. Weijer, "Adaptive color attributes for real-time visual tracking," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, America, 2014, pp. 1090−1097.

[20] J. 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.

[21] H. Karunasekera, H. Wang, and H. Zhang, "Multiple object tracking with attention to appearance, structure, motion and size," *IEEE Access*, vol. 7, pp.104423−104434, Jul. 2019.

[22] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, America, 2016, pp. 1430−1438.

[23] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, 2015, pp. 4310−4318.

[24] D. Yuan, X. Shu, and Z. He, "TRBACF: Learning temporal regularized correlation filters for high performance online visual object tracking," *J. Vis. Commun. & Image Representation*, vol. 72, pp. 102882, Oct. 2020.

[25] Q. Zhao, Z. Yang, and H. Tao, "Differential earth mover's distance with its applications to visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 2, pp. 274−287, Feb. 2010.

[26] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.,* San Diego, CA, USA, 2005, pp. 886-893.

[27] D. Ta, W. Chen, N. Gelfand, and K. Pulli, "SURFTrac: Efficient tracking and continuous object recognition using local feature descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami Beach, FL, USA, 2009, pp. 2937−2944.

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE Int. Conf. Comput. Vis.*, Barcelona, Spain, 2011, pp. 2564−2571.

[29] S. Wang, H. Lu, F. Yang, and M. Yang, "Superpixel tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Barcelona, Spain, 2011, pp. 1323−1330.

[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, pp. 27271−27290, Jun. 2019.

[31] B. Luca, V. Jack, G. Stuart, M. Ondrej, and T. Philip, "Staple: complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, America, 2016, pp. 1401−1409.

[32] G. Zhao, S. Zhuo, and X. Xu, "Multi-object tracking algorithm based on kalman filter," *Comput. Sci.*, vol. 45, no. 8, pp.253−257, Aug. 2018.

[33] S. Li, Z. Qin, and H. Song, "A temporal-spatial method for group detection, locating and tracking," *IEEE Access*, vol. 4, pp. 4484–4494, Sep. 2016.

[34] J. Zhang, H. Huang, J. Wang, and J. Bao, "An improved TLD real-time target tracking algorithm based on CN algorithm," *Comput. Eng. Sci.*, vol. 42, no. 7, pp.1215−1225, Jul. 2020.

[35] P. Voigtlaender, J. Luiten, P. Torr, and B. Leibe, "Siam R-CNN: visual tracking by re-detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp.1–17.

[36] H. Liu, Q. Hu, B. Li, and Y. Guo, "Robust long-term tracking via instance-specific proposals," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp.950－962, May 2020.

[37] D. Xiong, H. Lu, J. Xiao, and Z. Zheng, "Robust long-term object tracking with adaptive scale and rotation estimation," *Acta Autom. Sinica*, vol. 45, no. 2, pp.289－304, Apr. 2019.

[38] R. Gade, and T. Moeslund, "Constrained multi-target tracking for team sports activities," *Ipsj Trans. Comput. Vis. Appl.*, vol. 10, no. 1, pp.1－11, Dec. 2018.

[39] D. Yuan, N. Fan, and Z. He, "Learning target-focusing convolutional regression model for visual object tracking," *Knowl.-Based Syst.*, vol. 194, pp. 105526, Apr. 2020.

[40] W. Chen, J. Li, J. Xing, J. Xing, Q. Yang, and Q. Zhou, "Long-term object tracking based on kernelized correlation filter and hierarchical convolution features," *Comput. Sci.*, vol. 46, no. 9, pp. 271－276, Sep. 2019.

[41] B. Bai, B. Zhong, and Y. Ou, "Object tracking based on hierarchical convolution feature and scale adaptive kernel correlation filter," *J. Chin. Comput. Syst.*, vol. 38, no. 9, pp. 2062－2066, Sep. 2017.

[42] T. Zhu, and H. Liu, "Object tracking via hierarchical convolutional features with high robustness," *Opt. Optoelectron. Technol.*, vol. 17, no. 4, pp. 16－21, Aug. 2019.

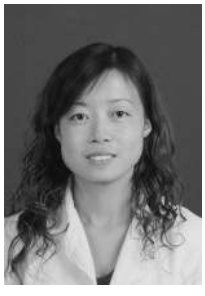
[43] H. Liu, Q. Hu, B. Li, and Y. Guo, "Robust long-term tracking via instance-specifific proposals," *IEEE Trans. Instrum. And Meas.*, vol. 69, no. 4, pp. 950－962, Mar. 2020.

[44] C. Ma, J. Huang, X. Yang, and M. Yang. "Robust visual tracking via hierarchical convolutional features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2709－2723, Nov. 2019.

[45] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit.*, Hawaii, America, 2017, pp. 4800－4808.

[46] 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.

[47] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 37, no. 9, pp. 1834－1848, Sep. 2015.



JIN-PING SUN received the M.S. and B.S. degree in Shandong University of Science and Technology. She is currently pursuing the Ph.D. degree in China University of Mining and Technology, Xuzhou, China. She is currently a associate professor in the School of Information Engineering(School of Big Data), Xuzhou University of Technology, Xuzhou, China. Her research field: signal processing, image processing, and deep learning application.

EN-JIE DING is a professor in School of Information and Control Engineering, China University of Mining and Technology since 1999. He is the executive deputy director of China University of Mining and Technology Dept. of IoT Perception Mine Research Center. He received the Ministry of Education Science and Technology Progress Award 2016, The Sixth Production Safety Science and Technology Achievement Award 2015 and many other awards. He received a Ph.D. degree in Information and Communication Engineering from China University of Mining and Technology in 1999. His current research interests include Internet of things, signal processing, fault diagnosis, Wireless sensor networks, Coal rock interface recognition and so on.

BO SUN received the Ph.D. degree in Shandong Agricultural University, the master's and bachelor's degree in Shandong University of Science and Technology. He is currently a lecturer in the School of Information Science and Engineering, handong Agricultural University, Tai'an Shandong, China. His research field: GNSS-R soil surface monitoring and deep learning application.

ZHONG-YU LIU received the M.S. degree from the China University of Mining and Technology, Xuzhou, China, in 2013, where he is currently pursuing the Ph.D. degree. His research interests include computer vision, machine learning, and signal processing.

KAI-LIANG ZHANG is currently a lab master in the School of Information Engineering(School of Big Data), Xuzhou University of Technology, Xuzhou, China. His research interests include network measurement, computer networks, and data mining.