Research Article

Ensemble Learning-Based Multi-Cues Fusion Object Tracking in Complex Surveillance Environment

Hui Du and Yanning Zhang

Beijing Polytechnic, Beijing 100176, China

Correspondence should be addressed to Hui Du; duhui@bpi.edu.cn

Received 13 June 2022; Revised 20 July 2022; Accepted 21 July 2022; Published 10 August 2022

Academic Editor: Zhao Kaifa

Copyright © 2022 Hui Du and Yanning Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The vast majority of currently available kernelized correlation filter (KCF)-based trackers simply make use of a single object feature to define the object of interest. It is impossible to avoid tracking instability while working with a wide variety of complex videos. In this piece of research, an ensemble learning-based multi-cues fusion object tracking method is offered as a potential solution to the issue at hand. Using ensemble learning to train multiple kernelized correlation filters with different features in order to obtain the optimal tracking parameters is the primary concept behind the improved KCF-based tracking algorithm. After that, the peak side lobe ratio and the response consistency of two adjacent frames are used to obtain the fusion weight. In addition, an adaptive weighted fusion technique is applied in order to combine the response findings in order to finish the location estimation; finally, the tracking confidence is applied in order to update the tracking model in order to prevent model deterioration. In order to increase the adaptability of the revised algorithm to size-change, a Bayesian estimate model based on scale pyramid has been presented. This model is able to determine the optimal scale of the object, which is the goal of this endeavor. The tracking results of a number of different benchmark movies demonstrate that the algorithm that we have suggested is able to effectively eliminate the effects of interference elements, and that its overall performance is superior to that of the comparison algorithms.

1. Introduction

Object tracking also plays a significant part in a large number of other industries, including human-computer interaction, aerospace, and virtual reality [1]. Object tracking technology has also advanced rapidly in recent years, and the associated application scenarios are becoming an increasing amount more complicated [2]. This is due to the ongoing development of a wide variety of distributed architectures as well as the rapid improvement of the performance of hardware processors. However, tracking in the actual world is complicated by a wide variety of interference elements, all of which have a significant negative impact on the accuracy of tracking. A pressing issue that needs to be addressed is how to make the tracking algorithm more reliable when applied to complicated settings.

The process of object tracking can be summed up as follows: given the state information of the object in the first frame, which may include location and scale, the object location is predicted in the subsequent frames by using the object characteristics, and the object motion state is analyzed through the motion trajectory [3]. The mobility condition of the object is difficult to forecast, so there may be significant discrepancies between various periods. Additionally, the posture of the object shifts regularly when it is moving; thus, the appearance of the object also frequently shifts as a result of these factors. The aforementioned interference problems not only make it much more difficult to follow an object, but they also severely restrict the amount of progress that can be made in this area [4]. Because of this, researchers in a variety of countries are dedicated to the improvement and development of tracking algorithms, and many tracking...
algorithms with excellent performance have emerged. In particular, the correlation filtering algorithm has been widely concerned by scholars both in the USA and in other countries because of its excellent real-time performance and excellent tracking effect, both of which enable researchers to achieve object location with high tracking efficiency.

The correlation filtering technique was initially implemented in the realm of signal processing, but it has since found widespread use in the domain of object tracking due to its lightning-fast processing speed and outstanding overall performance [5]. The minimum output sum of squared error (MOSSE) tracking algorithm was proposed in 2011 by Bolme et al. [6]. This algorithm creatively uses the minimization of the square sum error as a constraint to train the correlation filter, and it converts the calculation of the output response value into the frequency domain through the discrete Fourier transform, which significantly improves the tracking speed. Bolme et al. [6] used a basic gray feature from a single channel, and it lacks the capacity to represent object features. As a result, it has low robustness in situations where there is occlusion or scale shift. On the basis of this, Henriques et al. [7] proposed the circular structures with kernel (CSK) algorithm. In this algorithm, circulation sampling is used to replace sparse sampling, and the Fourier transform is also skillfully used to simplify the cyclic matrix operation. Both of these techniques were developed by Henriques et al. In addition, when the input data are nonlinear and separable, the kernel technique is used to project the correlation calculation to the high-dimensional space, which effectively enhances the performance of the filter [8]. CSK significantly enriches the sample database that is used to train the filter, which significantly enhances the discrimination ability of the classifier while ensuring that the operation efficiency is maintained. Nevertheless, the CSK method suffers from the same flaws as the MOSSE algorithm. The effect of this approach is not ideal in an environment with obvious background noise, and it is particularly vulnerable to the influence of background clutter. This is due to the fact that only gray features are employed to form the object model. Henriques et al. introduced the improved multi-channel directional gradient histogram feature into the framework of CSK algorithm, and they proposed a high-speed tracking with kernelized correlation filter (KCF) tracking algorithm [9]. This was done because of the poor ability of CSK algorithm to describe the object. The durability of the object model is improved thanks to this approach, and the algorithm’s accuracy is significantly increased even when operating in difficult contexts.

Because of its high level of accuracy, rapid tracking speed, and great comprehensive performance, the KCF algorithm has sparked a rush in research activity. Since then, many methods based on the KCF algorithm that are designed to improve and optimize performance have emerged [10]. The vast majority of algorithms have made the model better in a variety of ways, including feature selection and scale adaption. The selection of features is a very significant step in the process of developing the apparent model. The robustness of the apparent model can be improved by extracting additional discriminative features, which can subsequently lead to an improvement in the performance of the method [11, 12]. The KCF tracking algorithm was modified so that it includes the discriminant color descriptor that was developed by Danelljan et al. [13]. The discriminant color descriptor is an effective feature that can boost the discrimination between the object and the background, which ultimately leads to an improvement in the tracking performance [14]. Wang et al. introduced a scale adaptive multiple feature (SAMF) tracking algorithm, which is an improvement on the KCF algorithm in the sense that it improves the ability of the object model to be described. By cleverly making use of the strong complementarity among the three features, the SAMF model is able to significantly enhance the robustness of the discriminative model [15]. This is accomplished by combining the HoG feature, the color-name feature, and the gray feature to represent the appearance of the object. Ristic et al. developed a multi-color channel directional gradient histogram feature, which makes use of the gradient feature to connect the multi-channel color features. As a result, the accuracy of the technique is significantly improved [16]. Because it includes higher semantic information and can improve an item’s capacity to be represented, the deep learning feature has garnered a lot of attention in the field of object tracking. This is because of both of these reasons. The ability of the object to be represented has been substantially improved because of the work of Babenko and colleagues, who merged color characteristics and deep features [17]. Within the KCF correlation filtering algorithm, the convolution feature was initially developed and implemented by Freund and Schapire [18]. They were able to achieve rough to accurate object tracking by adaptively learning the correlation filter on each convolution layer. This allowed them to use the corresponding output response map of each layer to achieve this level of tracking, which showed good results in complex tracking environments.

The object scale often changes in the process of movement. When the object scale becomes larger, the bounding-box cannot completely surround the object, resulting in the loss of edge information [19]; when the object scale becomes smaller, the larger bounding-box will introduce too much background information, which is easy to cause the accumulation of tracking errors and eventually lead to tracking drift. Recently, there are two ways to improve the tracking drift problem for object scale change: scale pooling and sub-patch. The method based on scale pooling is difficult to achieve the balance between accurate scale estimation and real-time tracking performance. Increasing the scale-search range can effectively improve the accuracy of scale estimation, but it will also lead to the increase of computational complexity. Reducing the scale-search range is difficult to deal with the problem of large-object scale change [18]. The object is divided into sub-blocks, and the tracking filter is used to track each block. The object scale change is effectively estimated by calculating the location change between sub-blocks. Kim et al. proposed a tracking algorithm based on reliable patch tracker (RPT) [19], proposed a measurement standard to evaluate the tracking reliability of sub-blocks, and cleverly used the location relationship between reliable
sub-blocks to estimate the object scale change. Lu et al. divided the object into four sub-blocks [20], tracked each sub-block with a tracker based on the color-name feature, and estimated the object scale by calculating the location change of the sub-block between the adjacent frames. Zhang et al. use the local filter based on sub-block to estimate the object scale, and the global filter based on the whole object uses it as a reference to estimate the object scale [21]. With the help of the complementary relationship between local filter and global filter, the scale change can be effectively estimated. Correlation filter algorithms have excelled in the field of object tracking because of their high tracking accuracy and rapid running speed, both of which are advantages. In addition, correlation filter algorithms have made significant achievements in the field. However, the present algorithms have not been able to obtain adequate results in complicated situations because of the difficulty of the real-world tracking environment and the unpredictability of the object’s appearance and motion state. Researching correlation filter tracking techniques with improved efficiencies and robustness is still of utmost importance. In this piece of research, an ensemble learning-based multico- cues fusion object tracking method is offered as a potential solution to the issue at hand. Using ensemble learning to train multiple kernelized correlation filters with different features in order to obtain the optimal tracking parameters is the primary concept behind our improved KCF-based tracking algorithm. After that, the peak side lobe ratio and the response consistency of two adjacent frames are used to obtain the fusion weight. In addition, an adaptive weighted fusion technique is applied in order to combine the response findings in order to finish the location estimation; finally, the tracking confidence is applied in order to update the tracking model in order to prevent model deterioration. In order to increase the adaptability of the revised algorithm to size-change, a Bayesian estimate model based on scale pyramid has been presented. This model is able to determine the optimal scale of the object, which is the goal of this endeavor.

2. Related Works

2.1. Object Tracking Based on Kernelized Correlation Filter. Object tracking based on kernelized correlation filter [9] trains a classifier function \( f(x, w) = \langle w, x \rangle \) through the training samples to minimize its loss under certain decision conditions, where \( x \) is the training sample, \( w \) is the parameter to be solved for the classifier function, and \( \langle \cdot, \cdot \rangle \) is the inner product operation. Taking the sum of the squares of error of the training sample \( x \) and its corresponding label \( y \) as the loss function, the solution form can be obtained as

\[
\min_{w} \sum_{i=1}^{n} [f(x_i) - y_i]^2 + \lambda \|w\|^2,
\]

where \( x_i \) and \( y_i \) are the \( i \)-th \((i = 1, 2, \ldots, n)\) training sample and its corresponding labels, respectively; \( n \) is the number of training samples; \( \lambda \) is the regularization coefficient to prevent the classifier from training over-fitting; and \( \|\cdot\| \) is denoted as norm, such as \( \|_0 \). Taking the partial derivative of equation (1) and setting it equal to 0, its general solution can be obtained as follows:

\[
w = (X^TX + \lambda I)^{-1}X^Ty,
\]

where \( X \) is a matrix composed of training samples, and each row represents a sample \( x \); \( I \) is the identity matrix with the same dimension as \( X^TX \); \( y \) is the set of labels \( y \) corresponding to each training sample \( x \). For the kernel correlation filter-based object tracking, its sample matrix \( X \) is obtained from the initial object sample through cyclic shift, so \( X \) has a cyclic structure. Using the discrete Fourier transform (DFT) property of the cyclic matrix [22], the expression of equation (2) in the frequency domain can be written as follows:

\[
\tilde{w}_j = \frac{\tilde{x}_j \tilde{y}_j}{\tilde{x}_j^* \tilde{x}_j + \lambda},
\]

where \( \tilde{x}_j \), \( \tilde{y}_j \), and \( \tilde{w}_j \) are elements in the discrete Fourier transform of the initial object sample \( x \), the label set \( y \), and the calculated parameters \( w \) of the classifier, respectively; \( \tilde{x}^* \) is the complex conjugate of \( \tilde{x} \). Further, the kernel correlation filter maps the input sample \( x \) to the high-dimensional feature space through the kernel function, so the parameters \( w \) to be solved of the classifier can be expressed as \( w = \sum a_i \phi(x_i) \) its dual space, where \( a_i \) is the coefficient in the dual space; \( \phi(x_i) \) is the representation after the training samples \( x_i \) are mapped to the high-dimensional feature space. The problem of solving \( w \) is transformed into solving \( a \) in its dual space, and the form in the frequency domain can be expressed as

\[
\tilde{a}_j = \frac{\tilde{y}_j}{\tilde{k}(x, x) + \lambda}.
\]

As for a new frame \( z \) in video sequences, the response output of its corresponding classifier in the frequency domain can be written as follows:

\[
\tilde{f}(z) = \tilde{k}(x, z)\Theta\tilde{a},
\]

where \( \tilde{k}(x, z) \) is the discrete Fourier transform of kernel function \( K = \langle \phi(x), \phi(z) \rangle \). In equation (5), the coordinates corresponding to the max-value of its inverse Fourier transform are the location of the object in the new frame in video.

2.2. Ensemble Learning for Object Tracking. The theory behind the KCF tracking algorithm is that the degree to which the algorithm is able to classify data is a significant factor in determining the tracking impact. Because of this, the classifier with the strongest classification capacity can be generated by merging numerous classifiers using ensemble learning in order to get more precise tracking [23].

The objective of the KCF tracking model, from a statistical point of view, is to locate an optimal space, one in

\[
\min_{w} \sum_{i=1}^{n} [f(x_i) - y_i]^2 + \lambda \|w\|^2,
\]

where \( x_i \) and \( y_i \) are the \( i \)-th \((i = 1, 2, \ldots, n)\) training sample and its corresponding labels, respectively; \( n \) is the number of training samples; \( \lambda \) is the regularization coefficient to prevent the classifier from training over-fitting; and \( \|\cdot\| \) is denoted as norm, such as \( \|_0 \). Taking the partial derivative of equation (1) and setting it equal to 0, its general solution can be obtained as follows:

\[
w = (X^TX + \lambda I)^{-1}X^Ty,
\]
which the training data and the predicted output can achieve the best possible state of fitting together. On the other hand, statistical mistakes could arise if the amount of training data that is provided is insufficient in comparison with the scope of the object space. In this situation, we may lessen the likelihood of selecting “weak” classifiers by utilizing the framework of ensemble learning [24–27] to cast votes for a number of different hypothetical outcomes. This will allow us to lower the danger of selecting “weak” classifiers. Figure 1 depicts an overview of the ensemble learning process in the form of a diagram. Although support vector machines (SVMs) are widely regarded as strong classifiers in classification problems and have excellent performance in machine vision, the classification ability of using a single SVM classifier is restricted, as are the data types that can be correctly classified. This is despite the fact that SVMs have excellent performance.

For a given training sample set \( (x_i, y_i)_{i=1}^D \), \( x_i \) represents the characteristics of the \( i \)-th sample, \( y_i = \{ \pm 1 \} \) represents the label of the \( i \)-th sample, \( \{ K_m \}_{m=1}^M \) is a set for multi-kernel model, and the classifier composed of multi-kernel can be expressed as

\[
K(x, x_i) = \sum_{m=1}^{M} \beta_m K_m(x, x_i),
\]  

(6)

where \( \beta_m \) is the weight of the \( m \)-th kernel and satisfies \( \sum_{m=1}^{M} \beta_m = 1 \), and \( K_m \) is the \( i \)-th kernel function. Through ensemble learning of different kernel models, the resulting strong classifier can be written as follows:

\[
F(x) = \sum_{i=1}^{D} a_i y_i \sum_{m=1}^{M} \beta_m K_m(x, x_i) + b,
\]  

(7)

where \( \{ a_i \}_{i=1}^D \) and \( b \) are the Lagrange multiplier and standard offset, respectively.

Owing to the fact that the capability of categorization that ensemble learning possesses is obviously superior to that of any single classifier. Important ensemble learning technologies include bagging, boosting, and others [25]. The bagging method trains classifier individuals on different training sets through resampling technology to obtain diversity, and the randomness and independence of training samples provide ensemble diversity. The boosting method employs a deterministic method to ensure that the training set contains more difficult samples to form a classification. A number of investigations have demonstrated that the effect of boosting is, on average, superior to that of bagging [27]. A significant number of tests demonstrate not only that boosting can improve the learning accuracy, but also that it does not easily lead to over-fitting of the data. It is more efficient, has the ability to control bias and variance, and does so without compromising the quality of the results. When compared to other algorithms, the bagging algorithm is the only one that can reduce the model’s extremely high variance. Consequently, boosting is the method that should be selected whenever the created model must satisfy requirements for both variance and deviation.

3. Ensemble Learning-Based Multi-Cues Fusion Object Tracking

When it comes to object tracking, the selection of features and the discrimination between those features can have a significant impact on the results of object tracking. By computing the direction gradient of the local area, the HOG feature is able to effectively express the contour and shape information of the object, and it can also retain the invariance of geometry and lighting to some extent [28]. The KCF algorithm is able to better adapt to shifting illumination and rotation because it makes use of the multi-channel HOG feature.

The color-name (CN) feature gives a description of the item from a global point of view. It is not easily influenced by changes in object scale or shape [29], and it is rotation invariant. CN feature is a type of color feature that uses probability mapping in order to move an image from its native three-dimensional RGB space to its destination, which is an 11-dimensional space containing color features. In comparison with the capabilities of other color features, the CN feature possesses superior object description and representation capabilities.

The HOG feature and the CN feature, in comparison with the gray-scale feature, have the potential to increase the tracking performance of KCF correlation filtering to some degree. However, because of the limitations of a single feature to describe an object, and because of the fact that parts of an object’s characteristics can change in complex scenes, a single feature is unable to effectively describe the object, which will have an impact on the quality of object tracking [30, 31]. Because the HOG feature and the CN feature can each extract object features from a different perspective, because the HOG feature has good geometric and illumination invariance, and because the CN feature is insensitive to object scale and shape change, so the two features have strong complementarity with one another. Traditional fusion approaches all include the weighted combination of many features; however, due to the fact that the dimensions of individual feature vectors might vary widely, the direct combination of weights does not produce the best results. In this section, we will examine the several multi-cues fusion methods that are currently in use and then propose a multi-cues fusion object tracking method that is based on ensemble learning.

3.1. Multi-Cues Fusion. As is common knowledge, the vast majority of object tracking algorithms are based on the KCF algorithm. This algorithm incorporates a wide range of low-level features and adds a re-detection mechanism to enable corrections to be made in a timely manner in the event that tracking drift occurs. This enables the tracking accuracy to be optimized. The tracking procedure of the KCF tracker makes it clear that determining the maximum value of CFR is the most important step in the process of discovering the ultimate position.

It can be seen from equations (4) and (5) that the key to solve \( \bar{y} \) is two inner products \( \langle \phi(x), \phi(x) \rangle \) and
\[ \langle \phi(x), \phi(z) \rangle \]. Because \( \phi \) is the projection function of kernel space, its inner product can be calculated by kernel correlation function. If the kernel function is defined as \( K^{x,z} = \langle \phi(x), \phi(z) \rangle \), the inner product \( \langle \phi(x), \phi(x) \rangle \) and \( \langle \phi(x), \phi(z) \rangle \) can be expressed as \( K^{x,x} \) and \( K^{x,z} \), respectively.

Some improved algorithms introduce Gaussian kernel correlation function to calculate the high-dimensional inner product of cyclic matrix, namely,

\[
K^{x,z} = \exp \left( -\frac{1}{\sigma^2} \left( \|x\|^2 + \|z\|^2 \right) - F^{-1} \left( \langle \tilde{x}', \Theta \tilde{x}' \rangle \right) \right), \tag{8}
\]

where \( \tilde{x}' \) is the discrete Fourier transformation of \( x \) and \( x'' \) is the complex conjugate of \( x' \). Thus, the object function for KCF correlation function only needs to compute the dot product. It can be seen that the solution of kernel correlation function only needs to calculate the dot product and modulus of vector. In this way, multi-features can be easily introduced into the KCF tracker. Assuming that the object feature \( x = [x_1, x_2, \ldots, x_D] \) is obtained by cascading \( D \) low-level features, equation (8) can be rewritten as

\[
K^{x,z} = \exp \left( -\frac{1}{\sigma^2} \left( \|x\|^2 + \|z\|^2 \right) - F^{-1} \left( \sum_{i=1}^{D} \tilde{x}_i' \Theta \tilde{x}_i' \right) \right). \tag{9}
\]

Thus, multiple features can be fused into the KCF tracking framework to improve the robustness of the framework. In this paper, three typical features are used, which are gray feature, HOG feature, and color naming feature. Gray-level feature is a low-level simple feature; the HOG feature emphasizes the gradient of the image and calculates the discrete direction to form a gradient histogram, which is one of the most popular features; color-name feature, also known as color attributes, pays more attention to the color information contained in the tracked object, which is a label to describe the color. The distance in the color label space is more similar to human feelings, so it is a perspective space superior to RGB space. Color-name features have performed well in many visual fields, such as visual classification, object detection, and behavior detection. This paper uses the mapping method described in literature [12] to convert RGB space into color-name space, which is an 11-dimensional color representation. Color-name feature usually contains important information about the object. The fusion of the selected three features will greatly improve the efficiency accuracy of the tracker.

### 3.2 Proposed Fusion Strategy

In KCF correlation filtering, the peak side lobe ratio \( R_{PS} \) represents the peak sharpness of a correlation filter response (CFR), which is usually used to measure the confidence of object tracking. For the correlation filter response, the value \( R_{PS} \) at the peak location can be expressed as

\[
R_{PS}(x) = \frac{\max[x] - \mu(x)}{\sigma(x)}, \tag{10}
\]

where \( \max[x] \) is the maximum value of \( x \) in the filter response, and \( \mu(x) \) and \( \sigma(x) \) are the mean and standard deviation of \( x \), respectively. The greater the value of \( R_{PS} \), the higher the confidence of object tracking; otherwise, it means that the confidence of object tracking is lower. However, it is not enough to only use the peak side lobe ratio to represent the confidence of object tracking; especially when the object is occluded or similar object interference occurs, it is easy to lead to tracking drift or even failure. In the literature [20], the average peak-to-correlation energy (APCE) is proposed as a confidence evaluation index for object tracking; in the literature [21], the response smoothness constraint (RSC) is proposed as a confidence evaluation index to measure the tracking performance of each sub-block.

Inspired by literature [21], this paper defines the consistency \( C_{CFR} \) of the correlation filter response for two adjacent frames, and the expression is

\[
C_{CFR} = \| \bar{f}_i(x, y) - \bar{f}_{i-1}(x + \Delta x, y + \Delta y) \|_2^2, \tag{11}
\]

where \( f_i(x, y) \) and \( f_{i-1}(x + \Delta x, y + \Delta y) \) are the CFR map of the object at \( (t - 1) \)-th frame and the \( t \)-th frame, respectively; \( \Delta x \) and \( \Delta y \) are relative changes of object location between two adjacent frames; and \( \| \cdot \|_2 \) is a \( L_2 \) norm operation. In object tracking, because the time interval

![Figure 1: Schematic diagram of the ensemble learning.](image-url)
between two adjacent frames is usually only 20 ms or less, the changes of object and background between two adjacent frames are continuous, and their CFR maps have high similarity. Therefore, the value $C_{CFR}$ between two adjacent frames can be used as the confidence evaluation index of object tracking: when the value $C_{CFR}$ is small, the correlation filter response of the two adjacent frames has high similarity and the stability of object tracking is high; otherwise, the stability of object tracking is low.

Through the above analysis, this paper takes $R_{PS}$ and $C_{CFR}$ as the confidence evaluation index of object tracking and constructs a binary function $f(R_{PS}, C_{CFR})$ as the confidence evaluation function of object tracking, and its definition formula is shown as follows:

$$f(R_{PS}, C_{CFR}) = (1 - \rho)R_{PS} + \rho \frac{1}{C_{CFR} + \varepsilon}$$  \hspace{1cm} (12)$$

where $\rho \in [0, 1]$ is weight adjustment coefficient between $R_{PS}$ and $C_{CFR}$ and $\varepsilon$ is set to 0.01, which can avoid the denominator 0.

Firstly, two kernel correlation filters are trained by using the HoG feature and the CN feature, then the HoG feature and the CN feature of the candidate region are extracted, respectively, and the kernel correlation filter responses of the two regions are calculated. The two responses are Gaussian filtered to eliminate the abnormal response values. Finally, their confidence $f_{HOG}$ and $f_{CN}$ after filtering is calculated, whose calculation formulas are denoted as follows:

$$f_{HOG} = (1 - \rho)R_{PS,HOG} + \rho \frac{1}{C_{CFR,HOG} + \varepsilon}$$ \hspace{1cm} (13)$$

3.3. Multiscale Estimation. To solve the problem of scale change in improved KCF-based object tracking, three different solutions are proposed in the literature [21], but the scale change between two adjacent frames is not considered. In other words, the improved KCF tracking algorithm does not adapt to object with scale change.

In object tracking, the scale change of the object between two adjacent frames is small and continuous. This change can be approximated as a Gaussian distribution, where the scale change $s_i$ of the object in the current frame obeys the Gaussian distribution with its scale $s_{i-1}$ in the previous frame as the mean and $\sigma$ as the variance.

Once the prior distribution of the scale change between two adjacent frames is obtained, then a likelihood function $p(f_i|s_i)$ can be found, and the Bayesian estimation of the object scale change (maximum a posteriori probability) can be completed according to the following formula:

$$\max \{ p(s_i|f_i) \} = p(f_i|s_i) p(s_i),$$  \hspace{1cm} (14)$$

where $p(s_i|f_i)$ is likelihood probability and $p(s_i)$ is a priori probability.

When the scale of an object is given, the maximum similarity between the tracking candidate region and the object under this scale can be obtained by using the kernel correlation filter, and the maximum similarity can represent the probability of the object under this scale.

$$f_{CN} = (1 - \rho)R_{PS,CN} + \rho \frac{1}{C_{CFR,CN} + \varepsilon},$$ \hspace{1cm} (15)$$

where $R_{PS,HOG}$ and $C_{CFR,CN}$ are the value of the correlation filter response corresponding to the HoG feature and the CN feature, respectively, and are the values of the HoG feature and the CN feature between two adjacent frames, respectively. Finally, taking these two confidence degrees as the weight factors of feature fusion, the correlation filter response $R_{PS, CN}$ after fusion can be obtained as

$$\tilde{f}_{final} = \frac{f_{HOG}}{f_{HOG} + f_{CN}} \tilde{f}_{HOG} + \frac{f_{CN}}{f_{HOG} + f_{CN}} f_{CN},$$ \hspace{1cm} (16)$$

where $\tilde{f}_{HOG}$ and $\tilde{f}_{CN}$ are the kernel CFR maps corresponding to HoG feature and CN feature, respectively, and $f_{HOG}/f_{HOG} + f_{CN}$ and $f_{CN}/f_{HOG} + f_{CN}$ are the weight factors.

By constructing a scale pyramid to complete the maximum likelihood estimation of the scale change, namely, taking the estimated location of the object in the current frame as the object center, taking the object scale $s_{i-1}$ of the previous frame as the benchmark scale, and multi-scale sampling, so we can obtain

$$s_i = \{ s_m = s_{i-1} + md, m \in \left[ \frac{-M}{2} \frac{M}{2} \right] \},$$ \hspace{1cm} (17)$$

where $m$ is the number of sub-layers of multi-scale sampling; $s_m$ is the scale of each sub-layer sampled for multi-scale; $M$ is the number of layers of the scale pyramid; and $d$ is the change between two adjacent frames. By extracting the HoG features of multi-scale samples, a scale filter is constructed to complete the maximum likelihood estimation of the object scale. Then, the maximum a posteriori probability of each layer scale is obtained through equation (17), and the scale $s_m$ with the maximum a posteriori probability is taken as the optimal estimation of the object scale $s_i$ of the current frame.

3.4. Updating Strategy. Since the background of the object and the object will inevitably change in process of object tracking, it cannot accurately describe the changed state if the tracking model obtained in the first frame is fixed. Therefore, the updating strategy of tracking model will directly affect the performance of object tracking. According to the analysis of feature fusion, the $R_{PS}$ and $C_{CFR}$ of the fused correlation filter response can be used as the confidence evaluation index of object tracking. Similarly, the confidence after feature fusion is defined as

$$f_{final} = (1 - \rho)R_{PS, final} + \rho \frac{1}{C_{CFR, final} + \varepsilon},$$ \hspace{1cm} (18)$$

where $C_{CFR, final}$ is the value of the correlation filter response after the feature fusion of two adjacent frames. The calculation method of $R_{PS, final}$ and $C_{CFR, final}$ is consistent with that of $R_{PS}$ and $C_{CFR}$, and can be obtained by the correlation filter response after feature fusion. When the value $f_{final}$ is small,
it means that the confidence of object tracking is low, where the object appearance has changed greatly or tracking drift has occurred, so the tracking model cannot be updated; on the contrary, when the value \( f_{\text{final}} \) is large, it is considered as the confidence of object tracking at this time. In this paper, a confidence threshold \( f_{\text{th}} \) is set to judge whether to update the tracking model; in addition, to make the updating strategy more reliable, the historical frame information is also adopted when updating the model. In other words, if the confidence \( f_{\text{final}} \) of continuous multi-frame is greater than the threshold \( f_{\text{th}} \), the tracking model is updated by using the current frame information; otherwise, it is not updated due to interference. In KCF-based object tracking, there are two parameters that need to be updated in the frequency domain: one is the dual matrix parameter \( \tilde{a} \), and the other is the appearance parameter \( x \) of the object. In this paper, since we use ensemble learning to derive the multi-feature fusion formula, two tracking models need to be updated, where the specific parameters are updated as

\[
\begin{align*}
\tilde{a}_{\text{HOG}} &= (1 - \eta)\tilde{a}_{\text{HOG}} + \eta \frac{\tilde{y}_{\text{HOG}}}{(x_{\text{HOG}}, y_{\text{HOG}}) + \lambda}, \\
\tilde{x}_{\text{HOG}} &= (1 - \eta)\tilde{x}_{\text{HOG}} + \eta \tilde{x}_{\text{HOG}}, \\
\tilde{a}_{\text{CN}} &= (1 - \eta)\tilde{a}_{\text{CN}} + \eta \frac{\tilde{y}_{\text{CN}}}{(x_{\text{CN}}, y_{\text{CN}}) + \lambda}, \\
\tilde{x}_{\text{CN}} &= (1 - \eta)\tilde{x}_{\text{CN}} + \eta \tilde{x}_{\text{CN}},
\end{align*}
\]

where \( \tilde{a}_{\text{HOG}} \) and \( \tilde{a}_{\text{CN}} \) are the dual matrix parameters of the two ensemble learning-based KCF filters before model updating at frame \( t \); \( \tilde{x}_{\text{HOG}} \) and \( \tilde{x}_{\text{CN}} \) are the dual matrix parameters of two improved KCF filters before model updating at frame \( t \); \( \tilde{y}_{\text{HOG}} \) and \( \tilde{y}_{\text{CN}} \) are the object appearance parameters of two improved KCF filters before model updating at frame \( t \); \( \tilde{y}_{\text{HOG}}/(k(x', x') + \lambda) \) and \( \tilde{y}_{\text{CN}}/(k(x', x') + \lambda) \) are the dual matrix parameters obtained at frame \( t \); \( \tilde{x}_{\text{HOG}} \) and \( \tilde{x}_{\text{CN}} \) are the object appearance parameters obtained at the \( t \) frame, respectively; and \( \eta \) is the update rate. Finally, the tracking model can be updated by using the linear interpolation method, which can not only retain the relevant information in the previous frame, but also update the information of the current frame.

### 4. Experimental Results and Performance Analysis

#### 4.1. Parameter Setting

The proposed algorithm and several comparison algorithms are tested on the open challenge sequences for tracking performance, which can be downloaded in [http://www.votchallenge.net/challenges.html](http://www.votchallenge.net/challenges.html). According to the common challenge factors in object tracking, the video sequence attributes in the testing set are divided into 11 categories, specifically including illumination change (IV), object deformation (DEF), scale change (SC), occlusion (OCC), motion blur (MB), motion change (MOC), in-planar rotation (IPR), out-of-plane rotation (OPR), fast background change (FBC), scene complexity (SCO), and object color change (OCO). Each video in the testing set contains at least one of the above attributes. It should be noted that all quantitative evaluation results of the proposed method use the average of six independent tests.

The hardware configuration of the experimental simulation platform is as follows: the CPU model is Intel (R) core (TM) i5-7500 with the main frequency 3.30 GHz, and the memory is 8 GB; the software development platform is MATLAB R2016b.

The specific parameters of our proposed algorithm are set as follows: the template size used for Gaussian filtering of CFR map is \( 3 \times 3 \); the pyramid layers of scale estimation \( m = 17 \); and the change step \( d = 0.025 \); the confidence threshold is set to \( f_{\text{th}} = 6 \) in object tracking; and the number of historical frames is 3, which means that the tracking parameters of the first three frames are combined in the tracking process; the updating rate \( \eta \) is equal to 0.01; the parameter \( \rho \) used to adjust the weight of \( R_{\text{PS}} \) and \( C_{\text{CFR}} \) is set to 0.5; other parameter settings are consistent with KCF algorithm. It is worth noting that the adaptive parameter \( \rho \) can improve the tracking accuracy.

#### 4.2. Evaluation Criteria

In order to evaluate the effectiveness of our improved KCF-based tracking algorithm, the center location error \( E_{\text{CL}} \), distance accuracy \( P_{\text{d}} \), and overlap accuracy \( P_{\text{o}} \) are used as the evaluation indexes for object tracking results. \( E_{\text{CL}} \) is the Euclidean distance between the tracked object center location \( (x_\text{t}, y_\text{t}) \) and the benchmark center location \( (x'_\text{t}, y'_\text{t}) \), which can be denoted as follows:

\[
E_{\text{CL}} = \sqrt{(x_\text{t} - x'_\text{t})^2 + (y_\text{t} - y'_\text{t})^2}.
\]

The distance accuracy \( P_{\text{d}} \) refers to the percentage of frames whose \( E_{\text{CL}} \) is less than a certain threshold in the total frames of the video sequence. The overlap accuracy \( P_{\text{o}} \) refers to the overlap rate between the tracked object area \( R \) and the benchmark object area \( R_\text{g} \). In the percentage of frames, \( P_{\text{o}} \) is larger than a certain threshold in the total frames of the image sequences, where \( S_\text{c} \) can be expressed as

\[
S_\text{c} = \frac{|R \cap R_\text{g}|}{|R \cup R_\text{g}|},
\]

where \( | \cdot | \) is the number of pixels in the region.

For the above three evaluation indicators, the one-time pass evaluation (OPE) test method is used to evaluate the object tracking performance. Firstly, the location and scale of the object in the initial frame of the image sequences are given, and then the location and scale of the object can be determined by the tracking algorithm in each subsequent frame. This is an intuitive evaluation method, and it is also an evaluation method suitable for real-world practical application. In this paper, the threshold value for center location error is set to 20 pixel, and the threshold value for overlap accuracy is set to 0.5.
4.3. Comparative Experiment for Selection of Weight Coefficient. In equation (11), this paper defines the confidence evaluation function through \( R_{PS} \) and \( C_{CFR} \), and the weight adjustment coefficient ranges from 0 to 1, where \( R_{PS} \) represents the peak sharpness of the CFR of the current frame. The larger the value, the sharper the peak of the CFR, indicating that the higher the reliability of object tracking; \( C_{CFR} \) represents the similarity of the CFR of the two adjacent frames. The larger the value, the higher the similarity of the CFR of the two adjacent frames, indicating that the stability of object tracking is higher. \( R_{PS} \) and \( C_{CFR} \) reflect the confidence from different perspectives in object tracking, where \( R_{PS} \) represents the reliability of the current frame and \( C_{CFR} \) represents the stability of continuous frames. The weight adjustment coefficient \( \rho \) increases, indicating that the weight of \( C_{CFR} \) increases in the confidence evaluation of object tracking. In Figure 2, we give comparative experiment for selection of weight coefficient.

If \( \rho \) is larger than 0.5, it means that \( C_{CFR} \) is the dominant factor in the confidence evaluation of object tracking; otherwise, it means that \( R_{PS} \) is the dominant factor in the confidence evaluation of object tracking. In the OTB test set, the value \( \rho_d \) when the threshold value of \( E_{GI} \) is 20 pixel is selected as the evaluation index of tracking results to obtain the relationship between parameter \( \rho \) and tracking performance, as shown in Figure 2. It can be seen that a good balance has been achieved between \( R_{PS} \) and \( C_{CFR} \) at \( \rho = 0.5 \), and the tracking effect is also the best.

4.4. Tracking Performance Analysis. The comparison algorithms that were chosen for this paper in order to fairly verify the robustness and tracking accuracy of our improved KCF-based tracking algorithm are fast compression tracking (FCT) [30], spatiotemporal context tracking (STC) [31], kernel correlation tracking algorithm (KCF) [9], MIL tracking algorithm [16], and TLD tracking algorithm [15]. This was done in order to ensure that our improved KCF-based tracking algorithm is presented in the most accurate light possible.

The effectiveness of the tracking in the several typical video sequences was chosen. Four of these image sequences—a automobile, a boat, a surfer, and a woman—were subjected to analysis. Poor picture quality and interference are present in every video sequence; examples of these issues include fast background change (FBC), scene complexity (SCO), and object color change (OCO). The objects in boat suffer from severe partial occlusion while they are in the distance, and the scale change in the field of view causes the objects to have a great deal of difficulty when trying to see them; when the object in woman and car is moving quickly, there is motion blur, and especially when turning quickly, the contour of the object is almost invisible. The color of the object in surfer is similar to the background, and especially when the object passes through the gray spray, it is almost disappeared in the background; when the object in woman and car is moving rapidly, there is motion blur. The tracking performance of the entire sequence will invariably be impacted due to the interference caused by these circumstances.

The results of the tracking performed by various tracker algorithms are presented in Figure 3. The first row is from the car sequence, and the tracking process includes occlusion, interference from the background, rotation, and other similar objects; the second row is from the surfer sequence, and the tracking process includes attitude changes, occlusion, blur, and other interference; the third row is from the woman sequence. Interference, such as illumination and shadow, might have an effect on the tracking process. The backdrop is interfering with the item in the 37th frame from the boat series, which causes the TLD and MIL tracking algorithms to drift while they are in the midst of tracking the object. Because of insufficient generalization and representation ability of TLD and MIL tracking models to interference factors, as well as an inability to adapt to complex backgrounds and obvious appearance changes, the tracking bounding-box will gradually deviate from the object as errors continue to accumulate. This is primarily attributable to the fact that the tracking bounding-box will gradually deviate from the object over time. Both of the algorithms that we have proposed, FCT and STC, are able to follow the target; nevertheless, STC has a certain drift. The qualitative analysis demonstrates that our proposed method has superior tracking stability than other existing tracking technologies when it comes to coping with a wide variety of demanding movies (especially occlusion and shape deformation). It is possible to draw the conclusion that the tracking algorithm that was proposed in this research is much better than MIL, STC, and FCT, and that some outcomes are even better than KCF based on the findings presented in Tables 1–3. The results of the tracking performed by various algorithms are depicted frame by frame in Figure 4. The performance of the modified method presented in this paper demonstrates that it is capable of stable tracking.

It is not hard to notice that our suggested ensemble learning-based multi-cues fusion object tracking method has the best stability in the process of object tracking when
Figure 3: Tracking performance analysis.

Table 1: Comparison of overlap accuracy for different tracking algorithms.

| Videos       | TLD   | STC   | MIL   | FCT   | KCF   | Our   |
|--------------|-------|-------|-------|-------|-------|-------|
| Deer         | 0.611 | 0.618 | 0.614 | 0.551 | 0.724 | 0.739 |
| Car 4        | 0.905 | 0.911 | 0.538 | 0.881 | 0.871 | 0.873 |
| Car 11       | 0.821 | 0.812 | 0.592 | 0.772 | 0.841 | 0.855 |
| Surfer 3     | 0.456 | 0.538 | 0.504 | 0.516 | 0.698 | 0.715 |
| David indoor | 0.753 | 0.815 | 0.696 | 0.783 | 0.850 | 0.852 |
| Faceocc 2    | 0.821 | 0.832 | 0.718 | 0.802 | 0.826 | 0.835 |
| Girl         | 0.717 | 0.583 | 0.625 | 0.486 | 0.717 | 0.725 |
| Jumping      | 0.680 | 0.677 | 0.108 | 0.722 | 0.763 | 0.774 |
| Occlusion 1  | 0.872 | 0.941 | 0.798 | 0.814 | 0.812 | 0.865 |
| Singer 1     | 0.795 | 0.832 | 0.382 | 0.858 | 0.712 | 0.835 |
| Bird 1       | 0.501 | 0.653 | 0.357 | 0.606 | 0.628 | 0.629 |
| Woman        | 0.752 | 0.721 | 0.614 | 0.667 | 0.790 | 0.815 |
compared to other comparative algorithms. This is something that can be seen quite easily. Even when there is interference between the object and the backdrop, the enhanced algorithm is still able to follow the item. Because of this, the suggested algorithm is still capable of achieving accurate tracking ability, which demonstrates that the adaptive weighting method may successfully limit the interference caused by background information. Only our improved KCF-based method is able to complete the entire tracking process in the surfer sequences with background interference, which demonstrates that the strategy is able to successfully widen the feature difference between the object and the backdrop. This can be explained by the utilization of ensemble learning, which is able to not only keep the pertinent information from the previous frame, but also update the information of the frame that is now being viewed. However, it is important to keep in mind that the optimization of the tracking technique is the primary contributor to the tracking performance. Once occlusion has

| Videos | TLD | STC | MIL | FCT | KCF | Our |
|--------|-----|-----|-----|-----|-----|-----|
| Deer   | 8.2 | 10.1| 18.8| 8.2 | 7.6 | 6.5 |
| Car 4  | 3.7 | 4.2 | 4.0 | 4.9 | 3.0 | 4.4 |
| Car 11 | 4.7 | 4.9 | 3.8 | 2.3 | 2.2 | 1.6 |
| Surfer 3| 24.3| 11.3| 22.7| 27.5| 21.4| 4.3 |
| David indoor | 4.1 | 3.6 | 3.7 | 6.2 | 3.7 | 4.0 |
| Faceocc 2| 4.5 | 6.9 | 4.2 | 4.7 | 4.0 | 3.8 |
| Girl   | 12.7| 19.0| 41.8| 36.4| 12.4| 12.5|
| Jumping| 4.0 | 8.2 | 4.7 | 10.1| 4.0 | 4.6 |
| Occlusion 1| 7.0 | 9.1 | 3.4 | 4.9 | 4.7 | 3.1 |
| Singer 1| 4.3 | 4.9 | 3.3 | 6.6 | 4.7 | 7.1 |
| Bird 1 | 3.6 | 2.9 | 2.4 | 3.8 | 1.7 | 2.2 |
| Woman  | 7.3 | 10.2| 16.9| 14.1| 6.9 | 2.4 |

Table 3: Comparison of tracking variance for different tracking algorithms.

| Videos   | TLD | STC | MIL | FCT | KCF | Our |
|----------|-----|-----|-----|-----|-----|-----|
| Deer     | 8.8 | 6.2 | 16.7| 6.1 | 9.4 | 5.0 |
| Car 4    | 2.8 | 3.8 | 4.8 | 2.3 | 8.0 | 3.9 |
| Car 11   | 6.8 | 6.9 | 4.9 | 5.5 | 2.6 | 1.9 |
| Surfer 3| 18.5| 15.5| 13.9| 33.7| 31.0| 3.6 |
| David indoor | 5.8 | 2.3 | 4.8 | 15.8| 10.4| 4.5 |
| Faceocc 2| 3.5 | 2.2 | 3.1 | 5.9 | 4.0 | 2.7 |
| Girl     | 12.9| 15.5| 60.5| 42.2| 12.4| 10.1|
| Jumping  | 3.8 | 6.7 | 3.1 | 14.1| 3.3 | 5.2 |
| Occlusion 1| 6.0 | 2.5 | 4.8 | 3.1 | 3.6 | 2.1 |
| Singer 1 | 4.8 | 2.8 | 2.2 | 6.7 | 4.3 | 6.1 |
| Bird 1   | 3.3 | 3.0 | 3.1 | 4.5 | 7.7 | 4.2 |
| Woman    | 6.2 | 11.7| 12.7| 11.5| 3.7 | 1.4 |

Table 4: Performance comparison for multi-cues fusion.

| Indexes | Tracking models with different features | HoG_IKCF | CN_IKCF | IKCF |
|---------|----------------------------------------|----------|---------|------|
| Center location error $E_{CL}$       | 8.6       | 9.2     | 8.2    |
| Overlap accuracy $P_*$               | 0.77      | 0.76    | 0.81   |
been identified, the object parameters will be frozen, and the object confidence will be calculated in the search region. This process will continue until the object is once again captured. When occlusion happens using this method, the recapture mode will be put into effect because there will be no superimposed tracking frame. In spite of the fact that the accuracy of object tracking has been significantly enhanced, there is still a possibility of tracking loss (defined as the confidence being lower than the tracking threshold) in the process of object tracking.

4.5. Ablation Analysis. In order to verify the tracking performance of the improved KCF-based algorithm, ablation analysis is done in testing set. In this paper, we designed two experiments to analyze the correlation filter response and scale estimation. For the convenience of analysis, the complete multi-feature fusion algorithm is recorded as Full_IKCF, the improved algorithm using only HoG feature and CN feature is recorded as HoG_IKCF and CN_IKCF, which only uses a single feature for object tracking, and other parameter settings are consistent with the proposed algorithm. The comparative experimental results of the three algorithms are shown in Table 4, where the threshold value for center location error is 20 pixel, and the threshold value for overlap accuracy is 0.5. As can be seen from Table 3, compared with the HoG_IKCF and CN_IKCF using only a single HoG feature and CN feature, the proposed ensemble learning-based multi-cues fusion object tracking (Full_IKCF) is increased by 8.9% and 11.2%, respectively, and the value $P_c$ is increased by 9.4% and 13.7%, respectively. This shows that the adaptive feature fusion strategy proposed in this paper can effectively improve the overall performance of object tracking.

In order to verify the performance of the proposed algorithm for object scale estimation, four representative video sequences with scale changes are selected. The experiment first takes the object size of the first frame as the benchmark, then compares the estimated size with the object size of the first frame in the subsequent frames to obtain the estimated object scale change, and finally compares it with the benchmark scale change. In the four groups of videos, the object scale of the video sequence woman has a small range of 0.4 and 3.7; the object scale of video sequences car and boat varies widely, which is equivalent to 2.1 and 7.9 times of the initial scale from the first frame; the object scale of video sequence surfer has the largest variation range, which is equivalent to more than 32 times of the first frame.

The findings of the comparison between the estimated scale and the benchmark scale are presented in Figure 5. Figure 5 demonstrates that the multi-scale estimating method is capable of making correct predictions regarding the object scale change. The proposed method is nevertheless capable of making a more accurate prediction of the object’s scale, and this is true even when the object’s scale is subject to significant variations. It is plain to observe that the proposed method has been successful in achieving a high level of tracking performance in these difficult sequences.

5. Conclusion

An ensemble learning-based multi-cues fusion object tracking is proposed in this study as a solution to the problem of tracking drift. Using ensemble learning to train multiple kernelized correlation filters with different features in order to obtain the optimal tracking parameters is the primary concept behind the improved KCF-based tracking algorithm. After that, the peak side lobe ratio and the response consistency of two adjacent frames are used to obtain the fusion weight. In addition, an adaptive weighted fusion technique is applied in order to combine the response findings in order to finish the location estimation; finally, the tracking confidence is applied in order to update the tracking model in order to prevent model deterioration. In order to
improve the adaptability of the modified algorithm to scale change and ultimately achieve the ideal scale for the item, a Bayesian estimate model that is based on the scale pyramid has been offered as a solution. The tracking results of a number of different benchmark movies demonstrate that the algorithm that we have suggested is able to effectively eliminate the effects of interference elements, and that its overall performance is superior to that of the comparison method. In the future, in order to enhance the anti-interference capability of the tracking process, we are planning to take into consideration and summarize the deep feature for feature representation.

Data Availability

The dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Beijing Polytechnic.

References

[1] S. He, Q. Yang, W. H. Laur, J. Wang, and M. Yang, “Visual tracking via locality sensitive histograms,” in Proceedings of the Computer Vision and Pattern Recognition, pp. 2427–2434, Portland, OR, USA, 2013.

[2] N. Alt, S. Hinterstoisser, and N. Navab, “Rapid Selection of Reliable Templates for Visual Tracking,” in Proceedings of the Computer Vision & Pattern Recognition, pp. 1355–1362, San Francisco, CA, USA, June 2010.

[3] X. Mei and H. Ling, “Robust visual tracking using l1 minimization,” in Proceedings of the IEEE International Conference on Computer Vision, pp. 1436–1443, Kyoto, Japan, September 2009.

[4] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang, “Incremental learning for robust visual tracking,” International Journal of Computer Vision, vol. 77, no. 1-3, pp. 125–141, 2008.

[5] S. Avidan, “Support vector tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 8, pp. 1064–1072, 2004.

[6] D. S. Bolme, J. R. Beveridge, and B. A. Draper, “Visual Object Tracking Using Adaptive Correlation filters,” in Proceedings of the Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, IEEE, San Francisco, CA, USA, June 2010.

[7] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the Circulant Structure of Tracking-By-Detection with kernels,” European Conference on Computer Vision, Springer, Berlin, Germany, 2012.

[8] Y. Bai and TongM, “Robust tracking via weakly supervised ranking SVM,” in Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1854–1861, Providence, RI, USA, June 2012.

[9] S. Hare, A. Saffari, and P. H. S. Torr, “Struck: structured output tracking with kernels,” in Proceedings of the International Conference on Computer Vision, pp. 263–270, Barcelona, Spain, November 2011.

[10] J. Batista, C. Rui, P. Martins, and J. F. Henriques, “Exploiting the circulant structure of tracking-by-detection with kernels,” European Conference on Computer Vision, Springer-Verlag, Berlin, Germany, 2012.

[11] J. Batista, C. Rui, P. Martins, and J. F. Henriques, “High-speed tracking with kernelized correlation filters,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 3, pp. 583–596, 2015.

[12] Y. Li, J. Zhu, and C. C. H. Hoi, “Reliable Patch Trackers: robust visual tracking by exploiting reliable patches,” in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 353–361, Boston, MA, USA, 2015.

[13] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, “Accurate scale estimation for robust visual tracking,” Proceedings of the British Machine Vision Conference, pp. 65.1–65.11, BMVA Press, UK, 2014.

[14] W. Chen, K. Zhang, and Q. Liu, “Robust visual tracking via patch based kernel correlation filters with adaptive multiple feature ensemble,” Neurocomputing, vol. 214, pp. 607–617, 2016.

[15] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learning-detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 7, pp. 1409–1422, 2012.

[16] B. Babenko, M. H. Yang, and S. Belongie, “Visual Tracking with Online Multiple Instance learning,” in Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 983–990, IEEE, Miami, FL, USA, June 2009.

[17] B. Ristic, S. Arulampalam, and N. Gordon, “Beyond the kalman filter-particle filters for tracking applications,” IEEE Transactions on Aerospace and Electronic Systems, vol. 19, no. 7, pp. 37–38, 2003.

[18] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119–139, 1997.

[19] H. C. Kim, S. Pang, H. M. Je, K. Dajin, and B. Sung Yang, “Pattern classification using support vector machine ensemble,” in Proceedings of the International Conference on Pattern Recognition, pp. 160–163, Quebec City, QC, Canada, August 2002.

[20] H. Lu, S. Lu, D. Wang, S. Wang, and H Leung, “Pixel-wise spatial pyramid-based hybrid tracking,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 9, pp. 1365–1376, 2012.

[21] J. Zhang, S. Ma, and S. Stan, “MEEM: Robust Tracking via Multiple Experts Using Entropy Minimization,” European Conference on Computer Vision, Springer Cham, New York, NY, USA, 2014.

[22] N. Wang, J. Shi, D. Y. Yeung, and J. Jia, “Understanding and diagnosing visual tracking systems,” in Proceedings of the IEEE International Conference on Computer Vision, pp. 3101–3109, Santiago, Chile, December 2015.

[23] A. Blum, C. Dwork, F. McSherry, and et al, “Practical privacy: the SuLQ framework,” in Proceedings of the Twenty-Fourth ACM Symposium-Sigmod-Sigact-Sigart Symposium on Principles of Database Systems, pp. 128–138, Baltimore, Maryland, June 2005.

[24] Z. Hu, H. Chen, and G. Li, “Deep ensemble object tracking based on temporal and spatial networks,” IEEE Access, vol. 8, pp. 7490–7505, 2020.

[25] M. Harandi, J. Taheri, and B. C. Lovell, “Ensemble learning for object recognition and tracking,” Pattern Recognition, Machine Intelligence and Biometrics, Springer, Berlin, Germany, 2011.
[26] B. Uzkent and Y. W. Seo, “Enkcf: Ensemble of Kernelized Correlation Filters for High-Speed Object tracking,” in Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1133–1141, IEEE, Lake Tahoe, NV, USA, March 2018.

[27] X. Tian, S. Zhao, and L. Jiao, “Visual Tracking Based on Ensemble Learning with Logistic Regression,” International Conference on Bio-Inspired Computing: Theories and Applications, Springer, Singapore, 2016.

[28] A. Blum and T. Mitchell, “Combining Labeled and Unlabeled Data with Co-Training,” in Proceedings of the Eleventh Conference on Computational Learning Theory, pp. 92–100, Madison, Wisconsin, USA, June 1998.

[29] Y. Huang, Z. Zhao, B. Wu, Z. Mei, Z. Cui, and G. Gao, “Visual object tracking with discriminative correlation filtering and hybrid color feature,” Multimedia Tools and Applications, vol. 78, no. 24, Article ID 34725, 2019.

[30] K. Zhang, L. Zhang, and M. H. Yang, “Fast compressive tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 10, pp. 2002–2015, 2014.

[31] X. Tian, H. Li, and H. Deng, “Object tracking algorithm based on improved Context Model in combination with Detection Mechanism for Suspected Objects,” Multimedia Tools and Applications, vol. 78, no. 12, Article ID 16907, 2019.