International Journal of Innovative Computing, Information and Control
Volume **18**, Number **5**, October **2022**

ICIC International ©2022 ISSN 1349-4198
pp. **1395–1411**

TRACKING METHOD OF ONLINE TARGET-AWARE VIA SHRINKAGE LOSS

Jianwei Zhang[1], He Wang[2], Huanlong Zhang[3,*], Jingchao Wang[1]
Meng-En Miao[1] and Jiandong Wang[4]

[1]College of Software Engineering
[2]College of Computer and Communication Engineering
[3]College of Electrical and Information Engineering
Zhengzhou University of Light Industry
No. 136, Kexue Avenue, Zhengzhou 450002, P. R. China
ing@zzuli.edu.cn; {332007040550; 332013020638; 332013020640}@email.zzuli.edu.cn
*Corresponding author: hlzhang@zzuli.edu.cn

[4]Zhengzhou Huajun Technology Co., LTD.
No. 3508, Block A, Longzihu Yonghe Longzihu Central Square, Zhengdong New Area
Zhengzhou 450046, P. R. China
huajuntec@126.com

Received February 2022; revised May 2022

Abstract. *In the process of target tracking, since the target of interest is arbitrary, it is very important to construct an effective model to represent the target. Usually, the tracking model uses pre-trained convolutional neural network to extract features and represent the target. However, it is found that pre-trained depth features have poor representation ability for target in modeling targets. In order to fully explore the target representation ability of feature channels in feature layer of convolutional neural network, an online target-aware tracking method via shrinkage loss is proposed in this paper. Specifically, we introduce a shrinkage loss function which can reduce the influence of simple negative samples in the background information on tracking results, and combine the ranking loss function to select feature channel with strong target representation ability. At the same time, we design a template updating method based on the linear combination of initial template and nearest optimal template, which is fully adapted to the subsequent changes of tracking target state. The experimental results of OTB-50 and OTB-100 datasets show that the proposed solution can represent the target features more effectively than the traditional pre-trained feature extraction network, efficaciously improve the accuracy of target tracking and ensure the real-time performance of object tracking.*
Keywords: Deep learning, Target-aware, Template update, Object tracking

1. **Introduction.** Object tracking has become an important research direction in the field of computer vision, the basic task [1] of the visual single target tracking is to predict the size and position of the target in the subsequent frames when the target size and position of the initial frame of a given video sequence are known, through model construction and model training. Single target tracking has been widely used in various fields, such as monitoring system, unmanned vehicle, UAV tracking and other realistic scenes, which requires target tracking algorithm to have better real-time performance. The existing single target tracking problem [2] mainly includes fast moving of target, the target deformation, occlusion issue, etc., and these problems increase the difficulty of accurate tracking of single target. In order to solve these problems, the researchers have developed deformation

DOI: 10.24507/ijicic.18.05.1395

sensing, target re-detection, template updating and other measures to reduce the impact of the problems encountered in the process of target tracking.

It is found that the effective representation of the target is very important in tracking tasks. Traditional tracking frameworks often use hand-designed invariant features to represent targets, such as color histogram [3], HOG [4], Haar-like feature [5], SURF [6], ORB [7], subspace representation [8] and super pixel [9]. This kind of methods mostly appears in correlation filter tracking algorithms, and the representative ones are Kernel Correlation Filter (KCF) [10] and Correlation Filter with Discriminative Scale Space Tracking (DSST) [11]. However, the experiment [12] finds that this method only defines the shallow features of the target, and cannot represent the features of the target effectively. It is only suitable for some specific scenes. As deep learning [13] has achieved better and better results [14, 15, 16, 17, 18, 19] in target tracking, convolutional network models [20] and deep convolutional features are also used to improve the performance of computer vision tasks. In recent years, it has become a trend to use trained feature extraction networks to extract target depth features, such as some frameworks of Siam series. Although SGD method is also used in these methods to fine-tune the multi-layer network, it is difficult to meet the real-time requirements.

To solve the real-time problem, Siamese network [21] is introduced into target tracking. The target tracking algorithm based on the full convolutional Siamese network considers the deep convolutional network as a more general similarity learning problem in the initial off-line stage, uses the full convolutional neural network for feature extraction, and realizes the cross-correlation operation between the search area and the template. Distractor-aware Siamese Networks (DaSiamRPN) [22] added the difficult sample of background interference in the network off-line training stage to improve the discriminant ability of the tracker. The above studies only focus on the global feature representation and ignore the representation of the target's own feature. Hierarchical Convolutional Features (HCF) [23] found through experiments that different convolutional layers (Conv3, Conv4 and Conv5) presented different performance to the target, and proposed to replace the original HOG feature with hierarchical convolutional feature, which effectively utilized the feature layer feature and further accurately represented the target feature. Target-Aware Deep Tracking (TADT) [24] found that the differences between classes were mainly related to a few feature channels, and the others were redundant information; therefore, the authors

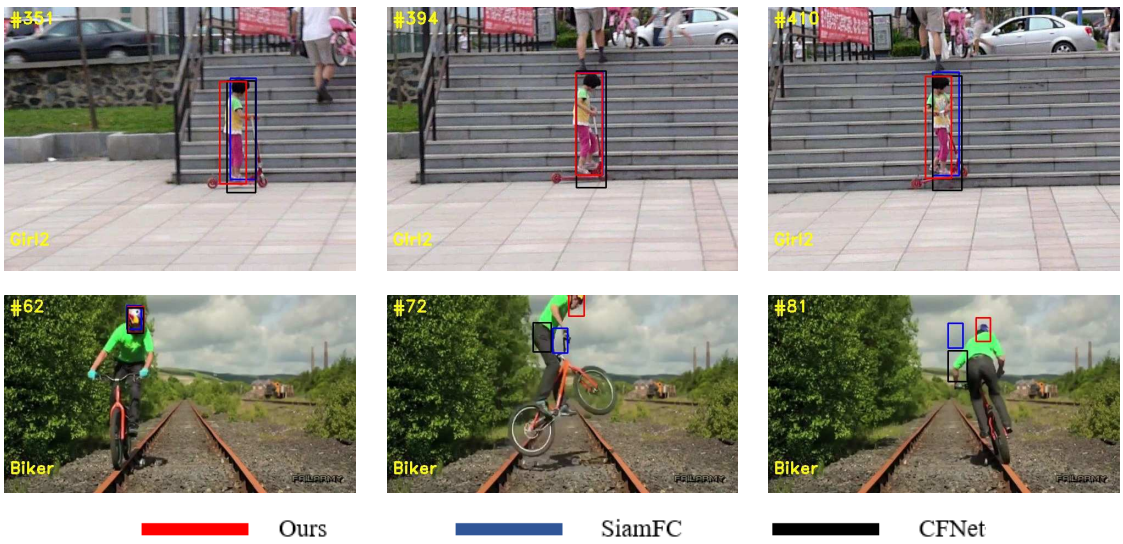


FIGURE 1. (color online) Tracking results in comparison with other trackers

proposed a target-aware method to enable the network to better identify the targets with significant appearance changes. The latest studies also embed the target-aware module into the target tracking network, but in different ways of thinking. For example, Guo et al. [25] used the prior knowledge of multi-branch interactive network and reference image to realize target-aware and candidate feature representation. Wang et al. [26] proposed a target-aware attention mechanism, which realized local and global target tracking by combining with multiple trackers. Similarly, in the small target detection task [27], the perceptual network also provides better prediction for the classification task.

In this way [24], based on the shrinkage loss function, the tracking effect can be improved, the training can be accelerated and the convergence can be accelerated. We introduce the shrinkage loss function into the Siamese network. Firstly, it regresses the pre-extracted features to the soft label generated by the Gaussian function. Then the network is trained by the corresponding soft label, and the filter with active objects is selected by the way of reverse gradient propagation. Combined with the sorting loss function, 23 pairs of scale training samples are used to train the network, and the filter active to the target scale is also selected by the reverse gradient propagation (the filter mentioned above refers to the appropriate filter kernel, and the processed features are obtained through convolution). Through the reverse gradient propagation of the two loss functions, the convolution filter of the corresponding channel is activated to guide the selection of characteristic channels with good target representation ability.

At the same time, in order to adapt to the change of subsequent tracking status, we designed a template update method, by setting a high quality frame selection mechanism to choose high quality video frame, using the linear combination of the way and the initial template frame for feature fusion, put the fused features into the target perception network for training similarly, and update the convolution filter of the corresponding channel. Thus, stronger target feature representation can be obtained to adapt to the morphological changes of subsequent targets. It avoids the tracking failure caused by the target template not being updated in time in target tracking.

The rest of this paper is organized as follows. We first introduce the CNN features of pre training in Section 2. Our method will be described in Section 3, which is mainly divided into perception part and template update part. The experimental results and ablation studies are shown in Section 4. Finally, summarize this paper in the fifth part.

2. **Features of Pre-Trained CNN.** The pre-trained deep VGG16 network is used as the basic feature extraction network, and the target-aware features are extracted from the trained target-aware network. As for the feature extraction network, there is often a big gap between the features extracted from the trained network and the features that can effectively represent the target in visual tracking. This framework extracts the preprocessed Conv4-1 and Conv4-3 output maps of VGG16 as the base depth features of 512 channels. Then, target activity features (300 channels) and target scale features (80 channels) are extracted by using target-aware module as the description of target depth features. In the tracking task, only a few convolution filters are active when describing the target. A large part of convolution filter contains redundant and irrelevant information, which requires a lot of calculation and is prone to over-fitting. Therefore, the importance of each convolution filter to the target feature is calculated through the target-aware framework, and the convolution filter that is interested in the target is selected.

The importance of convolution filter in capturing the information of category-level objects can be calculated by the corresponding gradient [28]. Therefore, based on the gradient descent method, a target-aware model using loss function to filter feature channels is constructed through gradient guidance. Assuming that the feature extracted by the

pre-trained VGG16 network is F, the sub-channel spatial feature f can be generated according to the importance of channel Δ. The calculation relation is as follows:

$$F = \varphi(f, \Delta) \tag{1}$$

where φ refers to the selection of important feature channel, Δ refers to the gradient value of each feature channel, reflecting the gradient value of the corresponding channel when the gradient drops. Δ is computed from Equation (2).

$$\Delta_i = \mathrm{GAP}\left(\frac{\partial L}{\partial z_i}\right) \tag{2}$$

where $\Delta_i = \mathrm{GAP}(\bullet)$ is often used to represent the gradient value of the corresponding channel in gradient descent, the magnitude of the gradient value represents the importance of the channel, L refers to the loss function, and z_i refers to the i-th feature channel output by the feature extraction network. The loss function represented by L includes the shrinkage loss function and the rank loss function, whose detailed content and attribute comparison of similar functions will be introduced in the next section.

3. **Proposed Work.** Our overall framework is shown in Figure 2, and it is divided into two parts: target-aware feature extraction and template update. The target-aware model consists of sorting loss and shrinkage loss. The weight of target-aware channel is determined by back gradient propagation algorithm. Cross-correlation operation obtains the final score response map, and the maximum value of the score represents the target location. The template updating module selects video frames through the high-quality video frame selection mechanism, adaptively updates the tracking template and updates the channel weight of the target-aware model.

3.1. **Target-aware feature extraction.** In this section, we will describe how target-aware features are extracted and how loss functions are used to select convolution filters.

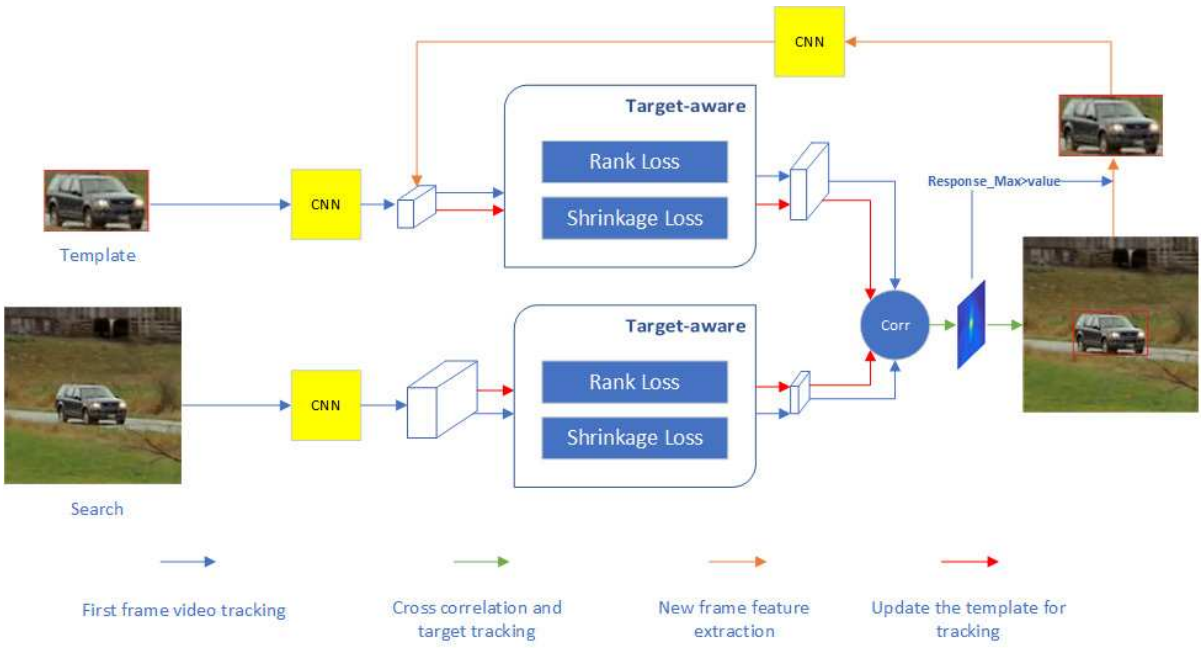


FIGURE 2. (color online) Overall tracking framework. The framework consists of pre-trained feature extraction network, target-aware module and template update module.

In feature extraction network, each filter helps us extract different features. In pre-trained neural network, convolution filter constructs depth feature space with different objective prior information to classify or recognize objects. Therefore, for the tracking task, in order to achieve more efficient target tracking, we can obtain the subset space of the convolution filter by training the target-aware network. Subset space is the subset of the convolution filter in which the pointer is active to the target, and the convolution feature is obtained from the subset of the convolution filter, which represents the depth feature with more difference between the tracked target and the background.

Ridge regression loss functions are often used to solve dichotomies, so they are suitable for tasks such as distinguishing target from backgrounds. Ridge regression loss function: ridge regression on the basis of the standard linear regression function of variable added a small square deviation factor (actually, that is the regularization), the square deviation factor to the model, the introduction of a small deviation, but greatly reduces the variance, compared with the commonly used linear regression model, and improved the operation efficiency and also to prevent the phenomenon of over fitting.

Before the regression loss function is used, Gaussian label mapping $Y(i,j)$ is generally performed on all sample $X_{i,j}$ with the target as the center. The calculation formula of Gaussian label mapping is shown in Equation (3).

$$Y(i,j) = e^{-\frac{i^2+j^2}{2\sigma^2}} \tag{3}$$

where (i,j) is the offset to the target, and σ is the kernel width. The derivation formula is as follows:

$$L = \arg\min \|Y(i,j) - W * X_{i,j}\|^2 + \lambda \|W\|^2 \tag{4}$$

where $*$ represents the convolution operation, W represents the weight of convolution, and $Y(i,j)$ is the soft label of the sample generated using the Gaussian function. The importance of each convolution filter can be calculated in terms of its contribution to the fitting of the corresponding label graph, that is, the derivation of the input feature X_{in} by the loss function L.

$$\frac{\partial L_{reg}}{\partial X_{in}} = \sum_{i,j} \frac{\partial L_{reg}}{\partial X_o(i,j)} \times \frac{\partial X_o(i,j)}{\partial X_{in}(i,j)} = \sum_{i,j} 2\left(Y(i,j) - X_o(i,j)\right) \times W \tag{5}$$

where $X_o(i,j)$ represents the output of the result obtained by the regression loss function. The gradient value of each convolution filter is calculated by back gradient propagation. Values of gradient represent the performance strength of the convolution filter to the target. Using the gradient, a fixed number of filters with the highest importance score can be selected from the pre-trained neural network to filter the feature channels.

In order to only add penalty for simple samples, so that their loss becomes smaller, and difficult samples are not affected, we introduce shrinkage loss [29] function as regression loss function to calculate the importance of each convolution filter:

$$L_s(W) = \frac{\exp(Y) \cdot \|W * X - Y\|^2}{1 + \exp(a \cdot (c - (W * X - Y)))} + \lambda \left\|W^2\right\| \tag{6}$$

The "dissimilarity" is defined as $l = |p - y|$, written in the form of mean square error, and the loss function can be defined as: $l^2 = |p - y|^2$. By adding penalties to the simple sample, the loss function becomes

$$L_s = f(l) \cdot L_2 \tag{7}$$

In the dissimilarity equation, the loss of simple samples is reduced, but the loss of difficult samples is also reduced, so the loss of such samples is a lot. Figure 3 shows the

advantages of our selected loss compared with other losses. Thus, a function is proposed to replace the loss function defined by dissimilarity:

$$f(l) = \frac{1}{1 + \exp(a \cdot (c - l))} \tag{8}$$

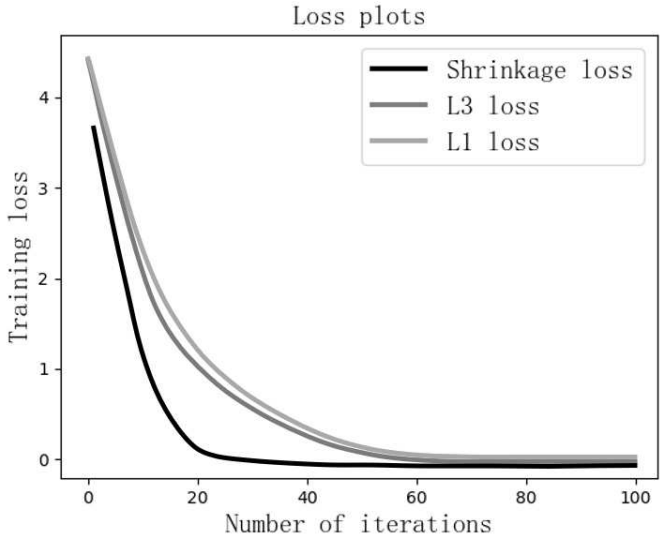


FIGURE 3. Training loss plot, which shows the comparison between the shrinkage loss with L1 loss and L3 loss. It can be seen from the figure that the shrinkage loss function has the fastest convergence speed, and it only needs a few to make the loss function converge.

The final loss function (9) is obtained from Equations (7) and (8):

$$L_{reg} = \frac{\exp(Y) \cdot \|W * X - Y\|^2}{1 + \exp(a \cdot (c - (W * X - Y)))} + \lambda \left\|W^2\right\| \tag{9}$$

In addition, for the change of target scale, smooth approximated ranking loss [30] is introduced, which aims to select the target-aware features with shrinkage loss. Its definition is as follows:

$$L_{\text{rank}} = \log\left(1 + \sum_{(x_i, x_j) \in \Omega} \exp\left(f\left(x_i\right) - f\left(x_j\right)\right)\right) \tag{10}$$

where (x_i, x_j) refers to paired training samples, which are sample labels of different scales (target scale ratio is set as 0.5-2) generated by extracted feature layers. The scale size of x_j is closer to that of the target than that of x_i. $f(\bullet)$ represents the scale prediction model, and represents the sample set containing different scales. A total of 23 groups of scale samples were prepared for target scale training. L_{rank}'s derivation of $f(x)$ is as follows:

$$\frac{\partial L_{\text{rank}}}{\partial f(x)} = -\frac{1}{L_{\text{rank}}} \sum_{\Omega} \Delta \mathbf{z}_{i,j} \exp\left(-f(x) \Delta \mathbf{z}_{i,j}\right) \tag{11}$$

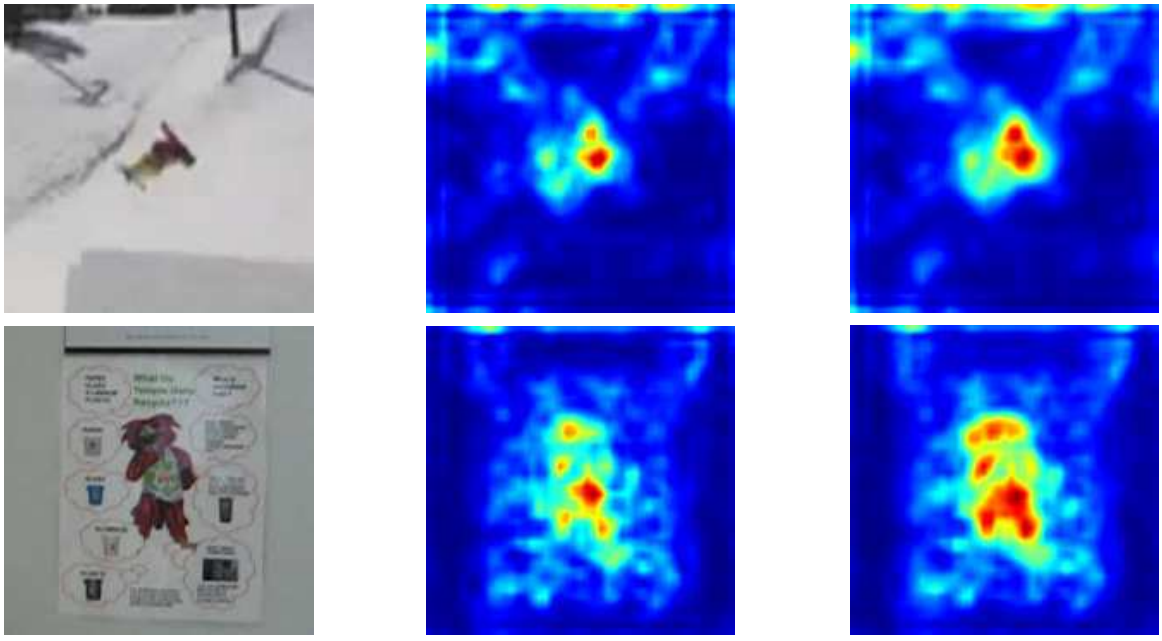
where $\Delta \mathbf{z}_{i,j} = z_i - z_j$, z_i and z_j are a single heat vector, z_i means the i-th element is 1 and everything else is 0, z_j means the j-th element is 1 and everything else is 0. According to the chain rule and Equation (11), the gradient of L_{rank} relative to depth features is

calculated as follows:

$$\frac{\partial L_{\text{rank}}}{\partial x_{\text{in}}} = \frac{\partial L_{\text{rank}}}{\partial x_o} \times \frac{\partial x_o}{\partial x_{\text{in}}} = \frac{\partial L_{\text{rank}}}{\partial f\left(x_{\text{in}}\right)} \times W \tag{12}$$

where W represents the weight of the convolution filter. The convolution filter sensitive to the target scale is found by the gradient obtained from the target scale rank loss.

We combine shrinkage loss function and rank loss function to select feature channels with strong target characterization ability through gradient descent. Specifically, the effect is displayed in Figure 4, and each line from left to right is the search area of the input image, the visual feature image of the target-aware module and the visual feature image formed after adding the template and updating the module. Note that it can be seen that the target can be easily separated from the background by obtaining the target perceptual features, and the updated template is more representative of the target.



(a) Search region (b) Target-aware features (c) Updated features

FIGURE 4. Visualization of the learned target-aware features. Target-aware module and template update module feature visualization image in the search area.

Compared with the features extracted from the pre-trained depth network, our method finds more effective depth features for the target. This not only alleviates the over fitting problem in the target tracking model, but also reduces the interference of other feature layers to the target itself. Experiments show that the shrinkage loss function can improve the tracking effect, speed up the training speed and accelerate the convergence speed. The target perception function can effectively represent any target or invisible object in the training dataset.

3.2. **Template update mechanism.** We propose an online updating mechanism based on the problem of target deformation and mutation during tracking. In this section, we will show you how to implement target template updates during online tracking.

Standard Template Update Method.

In order to adapt to the problem of state changes of subsequent targets in video sequences, researchers have proposed many methods to avoid this situation. The recent

tracking approach [10, 31, 32, 33, 34, 35] uses a simple averaging strategy to update the target's appearance model. This strategy method is derived from the previous tracking method, and has become a standard for online tracking, although there are certain limitations, but it does play a very good effect.

The standard template update is called the running average, and its weight decreases exponentially over time. The selection of exponential weight mainly depends on the recursive formula of the standard template update:

$$\tilde{T}_i = (1-\gamma)\tilde{T}_{i-1} + \gamma T_i \tag{13}$$

where i represents the serial frame label, T_i refers to using the current frame as the latest template sample, $\tilde{T}_{i-1}$ refers to the accumulated template sample, and the update rate γ is usually set to a fixed value (e.g., = 0.0102, 0.25). T in here refers to the feature extracted by the feature extraction network. In the discriminant correlation filtering tracking method, T refers to the correlation filter.

Although the standard template update algorithm provides a way to integrate historical frame information, it also contains many drawbacks.

1) It requires that the template update rate of each video be the same, and the template should be updated without considering whether the updated template is reasonable. However, in actual tracking tasks, the status of the target in the video cannot be guaranteed under the condition of constant template update.

2) If the updated template added in the tracking process is partially or completely occluded, template drift will occur, which will make it difficult for the tracker to capture the features of the target in the update process, resulting in tracking failure.

3) Once template drift occurs, the tracker cannot restore the template again, because it loses the most accurate template T_0, and the original template is undoubtedly the most accurate frame describing target information.

Our Template Update Method.

In the traditional method, the network usually uses the comprehensive feature of linear combination of the target feature obtained in the current frame and the template feature accumulated in the previous frame to guide the target tracking. This method not only leads to the exponential decay of target information with time, but also leads to the consumption of a lot of computing power. In addition, the traditional method lacks the information of the template frame. Once it follows the wrong target, it cannot be corrected to tracking the target.

We solved the above problems by designing an adaptive template update strategy, as shown in Equation (14):

$$\tilde{T}_i = (1-\gamma){T_0}^{GT} + \gamma T_i \tag{14}$$

where T_i refers to the video frame that can be used as template sample and the high-quality video frame selected by template selection mechanism; ${T_0}^{GT}$ refers to the template frame with initial truth value; γ refers to the template update rate, which can adjust the learning rate according to different video frames.

It is not hard to see in Equation (14), we design the template update algorithm including the initial frame target information and recently optimal template information, make full use of the target information effectively, and from Figure 5, we also add the initial template frame feature in the template updating algorithm, also prevent caused by target hide partially or completely block template drift phenomenon. We add high-quality frame filtering mechanism and learning update rate fine-tuning mechanism to the template update mechanism for adaptive template update.

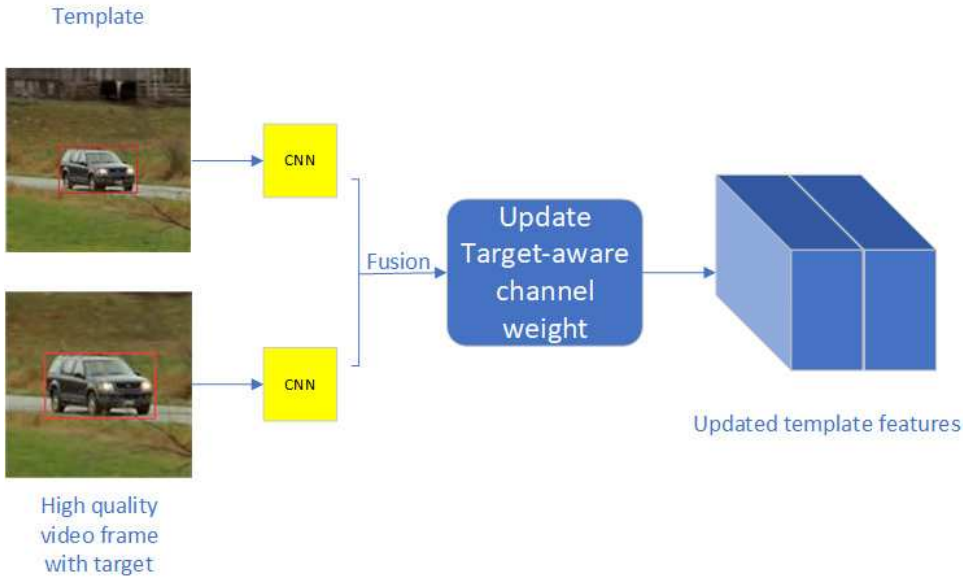


FIGURE 5. Template update mechanism. The template update mechanism only works when the high quality frame selection mechanism is triggered. In other words, when the evaluation score of the video frame is greater than the set threshold, the template update mechanism will start and update the latest template features.

The experimental results prove that the adaptive template updating method proposed by us is effective in improving the ability of target representation, and its renderings can be seen in the third column of Figure 4.

4. **The Experiment.**

4.1. **Tracking process.** Our tracking framework includes pre-trained deep feature extraction network, target-aware module and template update module. The feature extraction module is used to extract simple depth features by training in the classification task, and the awareness module firstly determines the target-aware weight by training in the first frame. Then the template update module selects high-quality video frames by using high-quality frame selection mechanism. High-quality video frames are mainly selected by a certain threshold. Then, the filtered video frames and template frames are linearly combined to update the latest template. Meanwhile, the target-aware module is retrained to update the more advantageous target-aware weights for continuous target tracking.

4.2. **Implementation details.** All experiments were performed on a server with 8 TITAN RTX GPUs, with an average tracking speed of 42.5. We used the VGG16 model as the backbone network. In order to maintain a more efficient use of spatial and semantic information, we chose to use the activation features of CONV4-1 and CONV4-3 layers as the base depth features. When training the target-aware model for the first time, we use the initial frame as the template. Since different tracking targets have different sizes, the search area is expanded to 3 times the size of the target area in order to adapt to different tracking objects. We selected the top 80 important convolutional filters from 512 channels in Conv4-1 layer as the target scale sensitive feature model by rank, and selected the top 250 important convolutional filters from 512 channels in Conv4-3 layer as the feature model to extract the active target. We set the convergence loss threshold of

the regression loss function as 0.02 and the maximum number of iterations as 100. Updating mechanism, in addition, we also set up in the template update mechanism, we set a certain threshold to filter the high quality video frame, this paper sets the threshold value as 2.96, by adjusting the weight of the initial template and the latest template updates the template online, by theoretical analysis, experiment and mathematical weight value is set to 0.25. In the estimation of target scale, the target scale is evaluated by generating a scale pyramid, which contains three scales, respectively 45/47, 1 and 45/43 times the previous target size. We set the corresponding scale change penalties for pyramids to 0.990, 1, and 1.005, respectively.

4.3. **Overall performance.** We evaluate the proposed algorithm on OTB-50 and OTB-100 benchmark datasets, compared with existing trackers CFNet [36], fDSST [11], KCF [10], LCT [37], LMCF [38], SAMF [39], SiamFC [40], SRDCF [41], Staple [42], etc. In addition, we also conducted tests on TC-128 dataset with KCF [10], Frag [43], VTD [44], MIL [45], OAB [46], etc. We present the results on each dataset and analyze them below.

OTB-50 dataset

OTB-50 dataset refers to a dataset containing 50 video sequences, which, together with OTB-100 dataset, is the most commonly used tracking benchmark in the field of tracking. This class of dataset is characterized by a manually annotated ground truth boxes, and also contains 25% gray scale datasets. OTB dataset involves 11 attributes of target tracking, including illumination change, scale change, occlusion, deformation, motion blur, fast motion, in-plane rotation, out-of-plane rotation, out of field of view, background interference, low pixel and so on. Each image sequence corresponds to two or more attributes, and each sequence corresponds to a text file, which records the target center position and the size of the target manually marked. Generally, the basic parameters to measure the target tracking accuracy include precision plot and success plot. Precision plot mainly refers to the Euclidean distance between the center point of the predicted position and the center point marked in the benchmark, which is calculated in pixels. Success plot mainly refers to the degree of coincidence of the benchmark where the predicted target is located.

Table 1 shows the performance of each tracker on OTB-50 dataset. Among many trackers, this algorithm has the best score in the OTB dataset accuracy statistics and ranks second in the success rate score. This is because the proposed target-aware depth feature and template update algorithm effectively make use of the unique appearance and semantic features of the target. Other trackers also achieve good tracking results, but the

Table 1. Performance of each tracker on OTB-50 dataset

Tracker	Success	Precision
SRDCF [41]	0.539	0.731
CFNet [36]	0.535	0.724
LMCF [38]	0.533	0.729
SiamFC [40]	0.519	0.693
Staple [42]	0.506	0.683
LCT [37]	0.488	0.689
SAMF [39]	0.464	0.649
fDSST [11]	0.460	0.616
KCF [10]	0.403	0.610
Ours	**0.543**	**0.757**

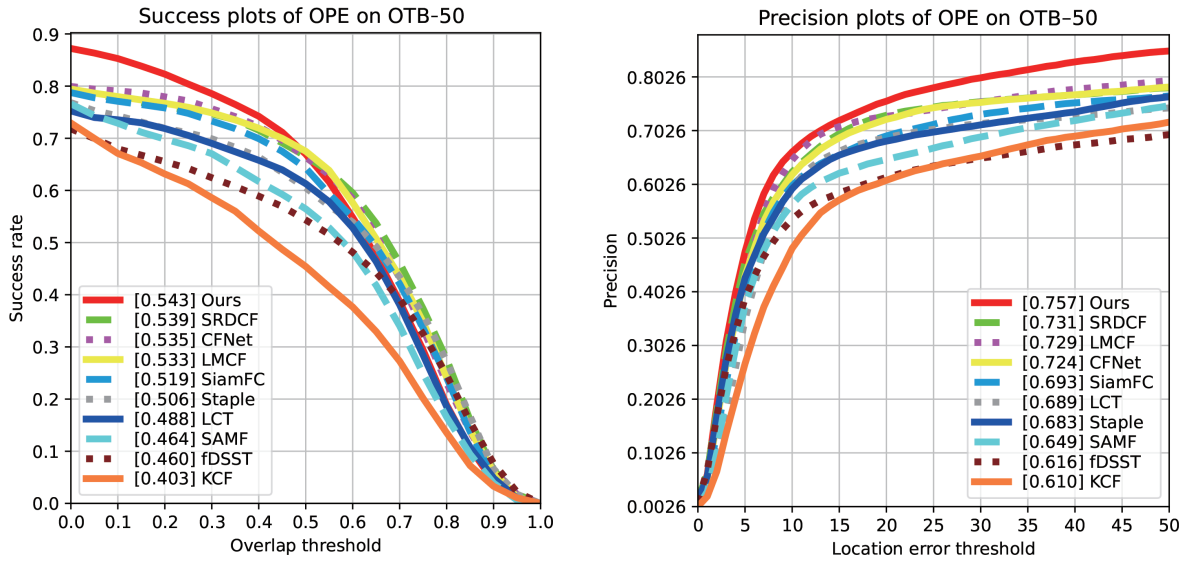


FIGURE 6. Success and precision plots on the OTB-50 dataset

tracking results are not good because of time-consuming online training and the limitation of over-fitting model. The algorithm adopts Siamese network which is more suitable for target tracking and feature layer which is more suitable for target expression to achieve 42.526 FPS real-time tracking speed. It is proved that the target-aware feature and template update method are effective. Figure 6 shows the good performance of the proposed tracker in OTB-50 compared to advanced real-time trackers.

OTB-100 dataset

The OTB-100 dataset is a dataset containing an additional 50 video sequences in addition to the OTB-50 dataset. The characteristics of OTB-50 dataset are the same, but the difference is that its dataset contains more video sequences, which greatly tests the tracking stability of the tracker. OTB dataset starts from random frames, or rectangular frames are initialized with random interference to run, which is more consistent with the target frame given by the detection algorithm. The OTB-100 benchmark adopts center positioning error and overlap ratio as evaluation criteria. On the basis of the two evaluation criteria, accuracy graph and success graph are often used to evaluate the overall performance of the tracker.

Table 2 shows the performance of each tracker on OTB-100. Among many trackers, the algorithm has the best score in precision and success rate of OTB-100 dataset, 0.597 and 0.816 respectively. Meanwhile, the high success rate reflects that the coincidence rate between the obtained target position and the original target position is very high, and it has strong adaptability in target positioning and scale prediction. The high accuracy indicates that the center offset between the target position estimated by the tracking algorithm and the manually marked target position is very small. To be specific, our estimation of the target location is very accurate. In general, the proposed tracking framework achieves good performance in accuracy, robustness and running speed. This proves the validity of the target scale sensitive feature and target semantic depth feature, which contributes to better target tracking. Figure 7 shows the good performance of the proposed tracker in OTB-100 compared to advanced real-time trackers.

Temple Color 128 dataset

Temple Color 128 is also referred to as TC-128. The TC-128 dataset contains 128 original image sequences (color sequences), which contain a large amount of color information and provide rich discrimination clues for visual reasoning. In addition, the TC-128 dataset

TABLE 2. Performance of each tracker on OTB-100 dataset

Tracker	Success	Precision
SRDCF [41]	0.598	0.789
CFNet [36]	0.587	0.778
SiamFC [40]	0.587	0.772
LMCF [38]	0.578	0.784
Staple [42]	0.578	0.783
LCT [37]	0.558	0.761
SAMF [39]	0.548	0.750
fDSST [11]	0.517	0.686
KCF [10]	0.477	0.695
Ours	**0.597**	**0.816**

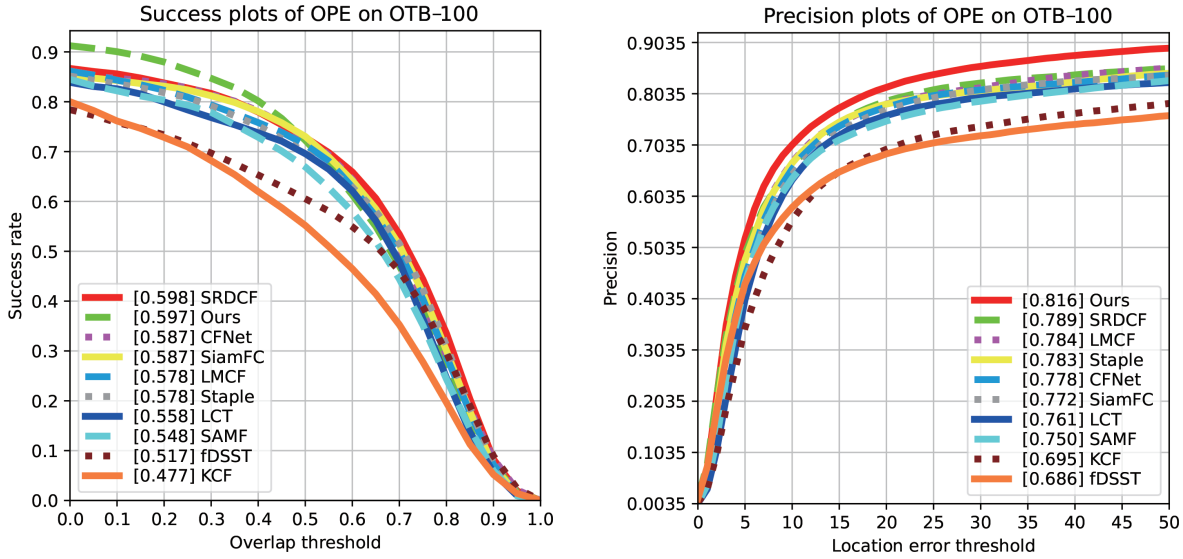


FIGURE 7. Success and precision plots on the OTB-100 dataset

TABLE 3. Experimental results on the Temple Color 128 dataset

Strategy	Success (AUC)	Precision
KCF [10]	0.418	0.588
Frag [43]	0.408	0.538
VTD [44]	0.407	0.527
MIL [45]	0.393	0.539
OAB [46]	0.389	0.526
Ours	**0.437**	**0.606**

contains other files, such as a file for evaluating the initial and last frames of the image sequence, a file containing the real positions of all targets, and a file containing the challenge sequence. AUC score is the evaluation index of TC-128. As can be seen from Table 3, the AUC score of the algorithm designed by us is only 0.437. The reason is that our algorithm is a model specially created for targets and lacks solutions for colors, but the score is not too bad. This shows that the algorithm also has some generalization ability.

4.4. **Ablation studies.** In this section, we will introduce the performance of our proposed method on OTB datasets in different states. We analyzed the proposed method on OTB datasets, including OTB-50 and OTB-100 (OTB-2015) datasets, to study the contribution of different losses to different layer characteristics and the addition of templates.

Table 4 lists the tracking effects under different conditions. Conv4-1 and Conv4-3 represent the output characteristics of CONV4-1 and CONV4-3, respectively. We compare the effects of different conditions on tracking results based on shrinkage regression losses, rank losses, and template updates, which are represented by rank, shrinkage, and update, respectively. By comparison, it was found that the AUC score obtained by Conv4-3 or Conv4-1 was lower than that obtained by the combination of the two datasets, OTB-50 and OTB-100. This is attributed to the effective use of deep semantic features for the target and appearance features for the scale, and the selection of the most effective convolution filter to generate target-aware features. In addition, from Table 4, we find that the AUC score (0.548 and 0.597) of OTB-50 and OTB-100 datasets has gained significant gains after the template updating mechanism is added. This indicates that although some channels of the feature layer were effectively utilized before, the target information was not updated. The performance of both datasets was slightly improved by adding a template update algorithm. This indicates that the added template updating mechanism plays a certain role, and can adapt to the changes of subsequent targets, ensuring the effectiveness of tracking.

Table 4. Tracking effects under different conditions

Conv4-1	Conv4-3	Update template	OTB-50	OTB-100
Null	Shrinkage	Null	0.475	0.596
Rank	Null	Null	0.452	0.595
Rank	Shrinkage	Null	0.544	0.596
Rank	Shrinkage	Update	0.548	0.597

5. **Conclusions.** In this paper, we mainly filter the optimal feature channel from the extracted features through the two loss functions of shrinkage loss and rank loss to learn the target-aware features, which makes up for the defect that the features extracted from the pre-trained feature extraction network have poor ability to represent the target. In addition, we design a linear template updating method, which uses the template selection mechanism to select the latest video frame as the latest template, and fuse the features with the initial template frame through linear combination as the new template features. The fusion features are input into the target-aware network as input items to update the learned target-aware features. It avoids the tracking failure caused by failure to update the template in target tracking. We integrate the target-aware feature model and template update model with Siamese tracking framework to prove its effectiveness and efficiency in visual tracking. A large number of experimental results on public datasets show that the proposed algorithm effectively enhances the representation ability of the extracted features to the target and improves the tracking efficiency of the tracking algorithm. Nevertheless, we also have some deficiencies. For instance, in the part of template updating, the new research method has used simplified memory network to remember a batch of favorable video frames. Using only one historical video frame is not enough for template updating. We also explore new research methods in this respect.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (62072416), Zhongyuan Science and Technology Innovation Leadership Program

(214200510026), Fourth Batch of Innovative Leading Talents of Zhihui Zhengzhou 1125 Talent Gathering Plan (ZhengZheng [2019] No. 21).

REFERENCES

[1] S. M. Marvasti-Zadeh, L. Cheng, H. Ghanei-Yakhdan and S. Kasaei, *IEEE Trans. Intelligent Transportation Systems*, vol.23, no.5, pp.3943-3968, DOI: 10.1109/TITS.2020.3046478, 2022.

[2] X. Q. Zhang, R. H. Jiang, C. X. Fan, T. Y. Tong, T. Wang and P. C. Huang, Advances in deep learning methods for visual tracking: Literature review and fundamentals, *International Journal of Automation and Computing*, vol.18, no.3, pp.311-333, 2021.

[3] J. van de Weijer, C. Schmid and J. Verbeek, Learning color names from real-world images, *IEEE Conference on Computer Vision & Pattern Recognition*, 2007.

[4] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[5] R. Lienhart and J. Maydt, An extended set of Haar-like features for rapid object detection, *Proc. of International Conference on Image Processing*, 2002.

[6] Z. X. Geng and Y. Q. Qiao, An improved illumination invariant surf image feature descriptor, *2017 International Conference on Virtual Reality and Visualization (ICVRV)*, 2017.

[7] F. Meng and F. You, A tracking algorithm based on ORB, *Proc. of 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)*, pp.1187-1190, 2013.

[8] B. Y. Wang, C. Fei and D. Ping, Online object tracking based on sparse subspace representation, *Control & Decision Conference*, 2014.

[9] Z. Ying, Y. Long, J. Wang and Z. Qin, An improved SLIC superpixel algorithm based on nonlinear filtering, *2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2015.

[10] J. F. Henriques, R. Caseiro, P. Martins and J. Batista, High-speed tracking with kernelized correlation filters, *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol.37, no.3, pp.583-596, 2015.

[11] M. Danelljan, G Häger, F. S. Khan and M. Felsberg, Accurate scale estimation for robust visual tracking, *Proc. of British Machine Vision Conference*, 2014.

[12] K. Sun, B. Xiao, D. Liu and J. Wang, Deep high-resolution representation learning for human pose estimation, *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[13] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Communications of the ACM*, vol.60, no.6, pp.84-90, 2017.

[14] M. Kristan et al., The visual object tracking VOT2015 challenge results, *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp.564-586, 2015.

[15] A. Berg, M. Felsberg, G Häger and J. Ahlberg, An overview of the thermal infrared visual object tracking VOT-TIR2015 challenge, *Swedish Symposium on Image Analysis*, 2016.

[16] M. Kristan et al., The visual object tracking VOT2017 challenge results, *2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp.1949-1972, DOI: 10.1109/ICCVW.2017.230, 2017.

[17] M. Kristan et al., The sixth visual object tracking VOT2018 challenge results, in *Computer Vision – ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science*, L. Leal-Taixé and S. Roth (eds.), Cham, Springer, 2018.

[18] M. Kristan et al., The seventh visual object tracking VOT2019 challenge results, *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp.2206-2241, DOI: 10.1109/ICCVW.2019.00276, 2019.

[19] M. Kristan et al., The eighth visual object tracking VOT2020 challenge results, in *Computer Vision – ECCV 2020 Workshops. ECCV 2020. Lecture Notes in Computer Science*, A. Bartoli and A. Fusiello (eds.), Cham, Springer, 2020.

[20] O. Natan, D. U. K. Putri and A. Dharmawan, Deep learning-based weld spot segmentation using modified UNet with various convolutional blocks, *ICIC Express Letters, Part B: Applications*, vol.12, no.12, pp.1169-1176, 2021.

[21] R. Tao, E. Gavves and A. W. M. Smeulders, Siamese instance search for tracking, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[22] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan and W. Hu, *Distractor-Aware Siamese Networks for Visual Object Tracking*, Springer, Cham, 2018.

[23] M. Chao, J. B. Huang, X. Yang and M. H. Yang, Hierarchical convolutional features for visual tracking, *2016 IEEE International Conference on Computer Vision (ICCV)*, 2016.
[24] X. Li, C. Ma, B. Wu, Z. He and M. H. Yang, Target-aware deep tracking, *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
[25] M. Guo, Z. Zhang, H. Fan, L. Jing, Y. Lyu, B. Li and W. Hu, Learning target-aware representation for visual tracking via informative interactions, *arXiv Preprint*, arXiv: 2201.02526, 2022.
[26] X. Wang, J. Tang, B. Luo, Y. Wang, Y. Tian and F. Wu, Tracking by joint local and global search: A target-aware attention-based approach, *IEEE Trans. Neural Networks and Learning Systems*, 2021.
[27] K. Wang, S. Du, C. Liu and Z. Cao, Interior attention-aware network for infrared small target detection, *IEEE Trans. Geoscience and Remote Sensing*, vol.60, pp.1-13, 2022.
[28] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, *2017 IEEE International Conference on Computer Vision (ICCV)*, pp.618-626, 2017.
[29] X. Lu, C. Ma, B. Ni, X. Yang and M. H. Yang, Deep regression tracking with shrinkage loss, *Proc. of the 15th European Conference*, Munich, Germany, 2018.
[30] Y. Li, Y. Song and J. Luo, Improving pairwise ranking for multi-label image classification, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3617-3625, 2017.
[31] D. S. Bolme, J. R. Beveridge, B. A. Draper and Y. M. Lui, Visual object tracking using adaptive correlation filters, *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, USA, 2010.
[32] L. Bo, J. Yan, W. Wei, Z. Zheng and X. Hu, High performance visual tracking with Siamese region proposal network, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.8971-8980, 2018.
[33] Q. Wang, Z. Teng, J. Xing, J. Gao and S. Maybank, Learning attentions: Residual attentional Siamese network for high performance online visual tracking, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.4854-4863, 2018.
[34] J. Kwon, Particle swarm optimization-Markov Chain Monte Carlo for accurate visual tracking with adaptive template update, *Applied Soft Computing*, vol.97, DOI: 10.1016/j.asoc.2019.04.014, 2019.
[35] J. Leng, H. Cai, W. Wang and Z. Ma, Double stage Siamese network object tracking algorithm based on template update, *2021 International Conference on Electronic Information Engineering and Computer Science (EIECS)*, pp.140-143, 2021.
[36] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi and P. H. S. Torr, End-to-end representation learning for correlation filter based tracking, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.5000-5008, 2017.
[37] C. Ma, X. Yang, C. Zhang and M.-H. Yang, Long-term correlation tracking, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.5388-5396, 2015.
[38] M. Wang, Y. Liu and Z. Huang, Large margin object tracking with circulant feature maps, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.4800-4808, 2017.
[39] Y. Li and J. Zhu, A scale adaptive kernel correlation filter tracker with feature integration, in *Computer Vision – ECCV 2014 Workshops. ECCV 2014. Lecture Notes in Computer Science*, L. Agapito, M. Bronstein and C. Rother (eds.), Zurich, Switzerland, Springer, 2015.
[40] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi and P. H. S. Torr, Fully-convolutional Siamese networks for object tracking, in *Computer Vision – ECCV 2016 Workshops. ECCV 2016. Lecture Notes in Computer Science*, G. Hua and H. Jégou (eds.), pp.850-865, Amsterdam, Netherlands, Springer, 2016.
[41] M. Danelljan, G. Hager, F. S. Khan and M. Felsberg, Learning spatially regularized correlation filters for visual tracking, *International Conference on Computer Vision (ICCV2015)*, Santiago, Chile, pp.4310-4318, 2015.
[42] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik and P. H. S. Torr, Staple: Complementary learners for real-time tracking, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1401-1409, 2016.
[43] A. Adam, E. Rivlin and I. Shimshoni, Robust fragments-based tracking using the integral histogram, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.798-805, 2006.
[44] J. Kwon and K. M. Lee, Visual tracking decomposition, *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1269-1276, 2010.
[45] B. Babenko, M.-H. Yang and S. Belongie, Robust object tracking with online multiple instance learning, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.33, no.8, pp.1619-1632, 2011.

[46] H. Grabner, M. Grabner and H. Bischof, Real-time tracking via on-line boosting, *Proc. of the British Machine Vision Conference*, pp.6-11, 2006.

Author Biography

Jianwei Zhang received his Ph.D. degree in computer application technology from PLA Information Engineering University in 2010. He is a professor at Zhengzhou University of Light Industry, Zhengzhou, China. His research interest covers broadband information networks, network security and visual tracking.

He Wang graduated from the International Education College, Zhengzhou University of Light Industry, Zhengzhou, China, in 2020. He is pursuing a master's degree at the College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China. His current research interests include deep learning and visual tracking.

Huanlong Zhang received his Ph.D. degree in control science and engineering from Shanghai Jiao Tong University, China, in 2015. He is currently an associate professor with the College of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou, China. His research has been funded by the National Natural Science Foundation of China (NSFC), the Key Science and Technology Program of Henan Province, etc. He has published more than 40 technical articles in refereed journals and conference proceedings. His research interests include pattern recognition, machine learning, image processing, computer vision, and intelligent human machine systems.

Jingchao Wang graduated from the College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China, in 2020. He is currently pursuing a master's degree at the College of Software, Zhengzhou University of Light Industry, Zhengzhou, China. His research interests include computer vision, object detection, and object tracking.

Meng-En Miao graduated from International Education College, Zhengzhou University of Light Industry, Zhengzhou, China, in 2020. He is currently pursuing a master's degree at the College of Software, Zhengzhou University of Light Industry, Zhengzhou, China. He has published an SCI paper. His research interests include computer vision tracking, and object tracking.

Jiandong Wang received his master's degree in computer engineering from PLA National Defense University in 1992. He founded Zhengzhou Huajun Technology Co., LTD. in 2006. Currently, he is a senior engineer in Zhengzhou Huajun Technology Co., LTD. His main research field is computer information processing, including big data, artificial intelligence, machine learning, target tracking, etc.