IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 December 2022, accepted 6 February 2023, date of publication 9 February 2023, date of current version 15 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3243628

RESEARCH ARTICLE

Accurate Object Tracking by Utilizing Diverse Prior Information

ZHAOHUA HU[1,2] **AND XIAO LIN**[1]

[1]School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China
[2]Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, Nanjing University of Information Science and Technology, Nanjing 210044, China

Corresponding author: Zhaohua Hu (zhaohua_hu@163.com)

This work was supported by the National Natural Science Foundation of China under Grant 61601230.

ABSTRACT Siamese trackers draw continuous attention in the object tracking community due to their proper balance between performance and inference speed. Despite that, it remains unclear how to effectively exploit the target appearance cues and motion cues involved in videos to improve trackers' performance. To address this problem, we develop a Siamese network with diverse prior information integrated, namely DPINet, by extending two novel blocks to a powerful anchor-free Siamese network. First, we design a channel- and space-aware feature enhancement (CSE) block to highlight target-specific feature weights in two aspects (channel and spatial dimensions). It is devoted to making full use of the target cues in the initial frame by considering them as guidances, in which way target-related representation in feature maps can be improved. It also facilitates the interplay between two input branches. Second, we advance a cross-correlation block with multi-dimensional information fusion (MDI-XCorr). In this block, target motion cues within adjacent frames can be mined and treated as supervisions to refine the response map in the current frame during inference. Hence, both tracking quality and stabilization can be enhanced. Evaluations on four popular benchmarks are conducted, showing that DPINet achieves 0.702 (AUC), 0.474 (EAO), 0.336 (EAO), 0.613 (AO), and 0.527 (AUC) on OTB100, VOT2018, VOT2019, GOT-10k, and LaSOT, respectively.

INDEX TERMS Visual object tracking, Siamese network, deep learning, information fusion, motion cue.

I. INTRODUCTION

Single object tracking is one of the fundamental tasks in the computer vision field. Given a target in an initial video frame, a tracker serves to search for and locate the target in the follow-up frames. Trackers can be deployed to embedded devices for a series of real-life applications, such as traffic flow monitoring [1], autonomous driving [2], and human-computer interaction [3]. Despite efforts made previously, tracking remains challenging due to several adverse factors in practice, e.g., low resolution, fast motion, and illumination variation.

Deep trackers can be divided into two categories. One is the discriminative correlation filter (DCF) group [4], [5], [6], [7], the other is the Siamese family [8], [9], [10], [11], [12]. A DCF tracker is considered as a discriminative model. It treats object tracking as a classification task to distinguish between foreground and background by learning an adaptive discriminative filter and conducting cross-correlation with the filter as a kernel on the video frames. One of the distinctive characteristics of DCF trackers is the adoption of online update strategies [5], [13]. More specifically, the filter can be constantly fine-tuned during inference, which facilitates to learn target-specific representation. Different from that, Siamese trackers are part of the generative model. They construct a Y-shape network with a template branch and a search branch to learn a feature embedding, in which a similarity comparison (implemented by a cross-correlation operation [8]) between a template image and candidate patches in a large search region is performed. Object tracking is thereby converted into a local matching problem. Siamese trackers learn feature representation by training convolutional neural networks (CNN) with numerous offline training samples. In this way, no extra computational consumption for kernel

The associate editor coordinating the review of this manuscript and approving it for publication was Jerry Chun-Wei Lin.

updating is required, and thus, Siamese trackers received a great deal of attention due to the proper trade-off between tracking accuracy and inference speed.

Siamese networks have been improved substantially in recent years, including the employment of deep backbones [10], [14], the upgrading of the cross-correlation module [10], [15], [16], the exploration of new sampling strategies for training [17], and the utilization of advanced prediction heads [11], [18], [19]. Despite all of those significant improvements, there is still an intrinsic limitation in Siamese networks—Siamese trackers treat visual object tracking as a local one-shot detection task, ignoring the distinctions between the two tasks. First, the detection task requires a detector to run in a class-aware manner, whereas the tracking task requires a tracker to locate a certain target regardless of its category. Particularly, target cues provided in the initial frame are interpreted as prior knowledge in tracking, which is absent in detection. Nevertheless, when designing Siamese trackers, the annotation (usually a bounding box) specifying a target-related area in the initial frame receives little attention—Siamese trackers just extract general features from the annotated frame as a template for subsequent similarity comparison, without further exploitation of the target cues. Second, temporal information can be involved usually in the tracking tasks, but barely in the detection tasks [20], [21]. Therefore, considering the temporal information (especially object motion cues) as prior information embedded in videos, an excellent tracker should be capable of leveraging it to handle the tracking tasks. However, standard Siamese trackers perform similarity calculations on each video frame without mining the time-dimensional information in consecutive frames. Although several works attempt to attack this problem by replacing the template during tracking [22] or updating target features [23], these methods remain indirect since the change in target appearance is not exactly time-dependent.

With careful analyses above, we propose two novel blocks to make full use of diverse prior information. First, we develop a channel- and space-aware feature enhancement (CSE) block. It endeavors to capture target-related cues in the channel and spatial domains from the given template image and the annotation in the initial frame. In addition, we identify that the consistency of target representations in the template branch and search branch is much significant for similarity calculation. Therefore, the CSE block is developed based on the recently popular attention mechanism [24] to improve the communication between the two input branches. Second, we design a cross-correlation block with multi-dimensional information fusion (MDI-XCorr) to exploit temporal cues in the video data. We argue that the target motion cues play a key role in discriminating the target from distractors. Hence, we propose to capture long-range dependency between response maps in adjacent frames to model target movement via the well-designed MDI-XCorr block. In this way, potential temporal cues are treated as supervisions, serving to modify the representation of raw feature weights induced by the cross-correlation operation.

Equipping a prevalent anchor-free Siamese tracker with the designed blocks, we propose a new tracker named DPINet. In contrast to the DCF trackers, our method requires no online updating, thereby working more efficiently. Besides, most general training strategies for Siamese trackers can be applied to DPINet without excessive adjustments, which indicates that it can benefit from numerous offline training data. We conduct extensive experiments on four popular benchmarks. Comparative results with other state-of-the-art (SOTA) CNN-based trackers show that our method achieves favorable performance on OTB100 [25], VOT2018 [26], VOT2019 [27], GOT-10k [28] test set and LaSOT [29] test set while running at a real-time speed.

Our contribution can be summarized as follows.

- We develop a channel- and space-aware feature enhancement (CSE) block. By interpreting target cues together with the annotation in the initial frame as inherent prior information in object tracking, the block merges them into the data flow of the network as guidances to advance target-related deep representation.
- We develop a cross-correlation block with multi-dimensional information fusion (MDI-XCorr) to address the problem of poor utilization of the temporal cues in Siamese networks. Unlike previous works that aim at adjusting or replacing template features, this block is assigned to capture target movement cues from response maps in adjacent frames.
- By extending the two blocks to an anchor-free Siamese network, we propose a new tracker with Diverse Prior Information exploited (DPINet). Its network can benefit from general training strategies developed for standard Siamese networks with only a few modifications and be trained end-to-end.

II. RELATED WORK

A. SIAMESE TRACKERS BASED ON DEEP REPRESENTATION

Recently, deep convolutional neural networks have made a great breakthrough [30], [31], [32], [33], [34], [35], which significantly promotes a series of fundamental tasks in computer vision, including image recognition [30], [36], object detection [37], [38], semantic segmentation [39], [40], etc. In object tracking community, two sorts of trackers substantially benefit from the improvement of convolutional neural networks. One of them is the Siamese group. Its pioneering work SiamFC [8] constructed a Y-shape fully-convolutional network by deploying a shared backbone in two input branches and a simple cross-correlation layer for feature combination (similarity calculation). It considered tracking as detection in image patches, which remarkably improved the inference speed of deep trackers. SiamRPN [9] advanced scale estimation in SiamFC by introducing Region Proposal Network (RPN) into the Siamese framework. It combined Siamese networks with an X-shape architecture and predicted target bounding boxes more accurately. DaSiam [17], belonging to the SiamRPN family, proposed a sampling strategy for Siamese trackers so that their discrimination and robustness can be substantially improved with extra network training on object detection datasets. SiamDW [14] and SiamRPN++ [10] enabled Siamese networks to be compatible with a ResNet [32] backbone by adjusting the convolutional blocks and training strategies, respectively. Inspired by the achievement of anchor-free task heads in the object detection community [41], [42], [43], several works migrated the design paradigm to Siamese networks, significantly reducing the number of parameters and computational consumption of the trackers [11], [18], [19].

B. FEATURE ENHANCEMENT

Feature enhancement involves the modification of visual features to achieve a specific representation. *Attention* mechanism is a sound candidate, which has achieved great success in many vision tasks [44], [45], [46]. To name a few, SENet [47] introduced S-and-E blocks, endeavoring to extract channel-wise representation and to adjust channel activation distribution of visual features. CBAM [48] proposed spatial attention and channel attention to emphasize feature representation in multiple domains. Different from those modules based on feature extraction, Non-Local [24] attention aimed at capturing and modeling long-range dependency of feature vectors in the spatial dimension, providing global receptive fields that facilitate the mining of potential relationships between candidate windows on the original image. Based on that, background distractors can be suppressed and target-related representation is allowed to be advanced.

Many Siamese trackers equip themselves with attention modules to promote target-specific representation for high-quality object tracking. RAR [49] emphasized specific visual patterns and leveraged both inter- and intra-frame attention by incorporating a hierarchical attentional module into a Siamese tracker. SATIN [50] introduced a novel cross-attentional module, in which way both channel-wise and spatial intermediate attentional information can be leveraged to improve contextual representations. RASNet [51] proposed two attention blocks, general attention and residual attention, to learn the general characteristics and distinctions of different targets in videos, respectively. SiamAttn [52] introduced self-attention and cross-attention implemented by Non-Local modules, which improved representation quality in a self-attentive manner. Nocal-Siam [53] learned to associate multiple response maps via attention modules, in which location cues are used to prevent response maps from diverse sharp peaks. STMTrack [54] leveraged Non-Local attention to retrieve diverse target cues in a video sequence and fuse them into the template feature map during inference. As a result, the representation of template features is advanced and the tracking performance gets more robust. HSSNet [55] proposed an attention-based spatial-aware network, aiming to make Siamese trackers more robust to spatial rotation, scaling, and translation in thermal infrared object tracking. Following that, MLSSNet [56] combined a multi-level similarity network with a Siamese framework, which demonstrated the ability of attention for modeling semantic and structural similarities. MMNet [57] constructed a fine-grained aware module, devoted to learning intra-class representation of objects via a Non-Local network.

Inspired by the notable achievements mentioned above, we develop a channel- and space-aware feature enhancement block (CSE) to make full use of the target-specific information via *attention*. It is combined with both *self-attention* and *cross-attention*. A recent work whose implementation seems like that of our method is SiamAttn. In fact, there are clear differences between the two networks. We emphasize that prior cues of the target in the initial frame are of great significance in object tracking. Contrastingly, SiamAttn pays little attention to the given cues when enhancing feature representation in the network. Besides, our proposed CSE block interprets the target cues and the initial annotation as guidances and incorporates them into the data flow of feature processing. Nevertheless, SiamAttn runs in a completely self-attentive fashion—without any guidances in feature processing.

C. EXPLOITATION OF TEMPORAL INFORMATION

Most existing Siamese trackers that integrate temporal information into its feature processing adopt strategies of template replacement or residual feature updating. For instance, DROL [22] incorporated a plug-and-play component into Siamese trackers, which collects potential template images according to historical tracking results and replaces the template feature map once meeting the conditions. STMTrack [54] dynamically integrated diverse template feature maps into a single one during tracking, so that varied target appearance information could be imposed into the template map, which facilitates robust object tracking. UpdateNet [23] was proposed aiming at online template feature tuning, which is comprised of a base tracker and an updater. The updater was devoted to learning target appearance variation from adjacent frames and integrating it into template features frame by frame. Nevertheless, it failed to take the object detection datasets as its training sets, and its training strategy requires tedious adaptions of video data in advance compared to general Siamese strategies. Different from that, Siam R-CNN [58] introduced a re-detection scheme and a tracklet-based algorithm into Siamese trackers, interpreting object tracking as local feature matching of the Region of Interest (RoI) over video frames. Despite its significant success in terms of high-quality performance, it introduces heavy computation loads, which is uneconomical in practice.

Different from all the trackers mentioned above, our method equipped with the proposed MDI-XCorr block allows better use of temporal cues involved in response maps generated by adjacent frames. MDI-XCorr enables Siamese networks to capture and model target movement over the time dimension. In addition, all of the above-mentioned deep trackers require deliberate modification of their training

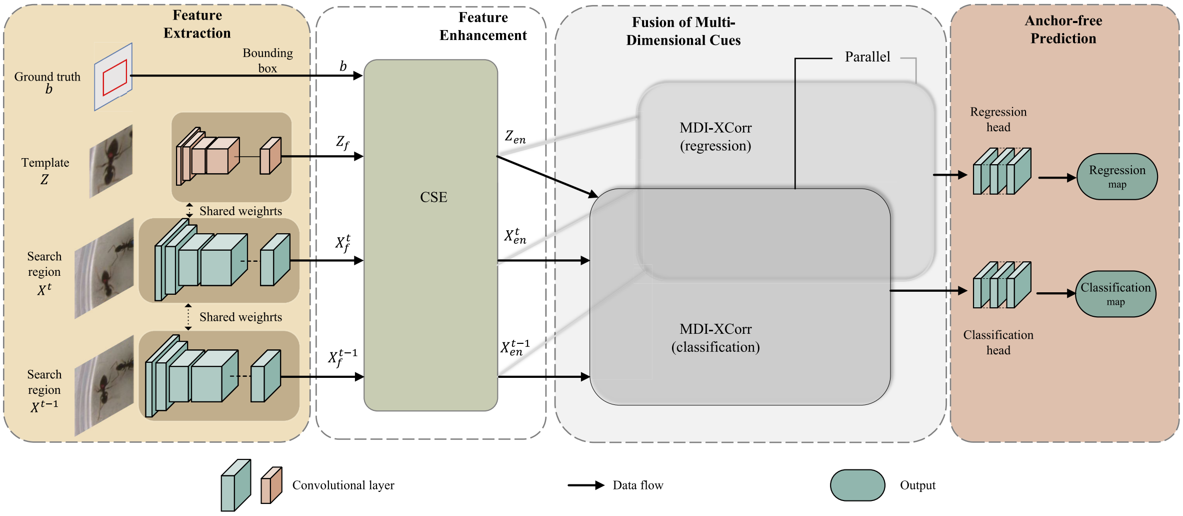


FIGURE 1. The overall structure of DPINet. There are two parallel branches in Fusion of Multi-Dimensional Cues for different subtasks (classification and regression) without sharing their parameters.

settings to accommodate the introduction of temporal information. In contrast, Siamese trackers combined with the MDI-XCorr block can fully utilize the general training settings tailored for themselves [10], [11], [17], requiring only a few adjustments to the training strategies.

III. PROPOSED METHOD

A. FRAMEWORK OF DPINet

As presented in Fig. 1, the proposed DPINet contains four core components: **Feature Extraction**, **Feature Enhancement**, **Fusion of Multi-dimensional Cues**, and **Anchor-free Prediction**. **Feature Extraction** takes triplet images as inputs, i.e., the template image Z and two search region images X^t and X^{t-1} in adjacent frames. It serves to map the images to an embedding space, from which visual features in the template and search branches can be derived. Subsequently, the obtained feature maps (Z_f, X_f^t, and X_f^{t-1}) are enhanced in **Feature Enhancement**. The CSE block in this stage tends to highlight target-specific cues in the channel and spatial dimensions with the guidance of the ground truth b. The resulting feature weights (Z_{en}, X_{en}^t, and X_{en}^{t-1}) are then delivered to the MDI-XCorr block for the **Fusion of Multi-dimensional Cues** as well as temporal information mining. It should be noted that there are two parallel MDI-XCorr blocks in this component, one for the classification branch and the other for the regression branch. The fused response maps, involving multi-dimensional cues, are finally fed to **Anchor-free Prediction** heads [11]. To be specific, a classification map (CM) and a regression map (RM) are induced via prediction heads that consist of stacked convolutional layers, and both of them have the same spatial dimensions. For each location (l_x, l_y) on the response maps, RM provides a proposal bounding box and CM estimates the confidence that the area within the bounding box belongs to the foreground. The proposal bounding box with the highest confidence score is considered as the final output of the tracker to locate the target in the current video frame. More details of the anchor-free heads are referred to [11].

1) MOTIVATION

As discussed previously, existing Siamese trackers treat the tracking task as a detection problem based on matching between a template and candidate windows in frames. However, there are limitations to the tracking-by-detection style. First, in terms of utilization of the initial target image, the original Siamese tracker delivers feature weights of the template image directly to the similarity estimation (cross-correlation) module without target-oriented modification. Although recent works [52], [53] address this problem by enhancing feature representation, they mostly neglect the prior information of targets, i.e., the given annotation in the initial frame. It remains unclear how to effectively make use of this kind of prior cues in the tracking task. Second, the cross-correlation module in a Siamese network involves no temporal information—the target motion cues embedded in video data cannot be captured and utilized. Despite a series of improvements to the cross-correlation module [10], [15], [16], to our best, there is no work managing to incorporate temporal information into the cross-correlation module to improve tracking quality. The proposed blocks, CSE and MDI-XCorr, are just tailored to solve the problems.

B. REVIEWING SIAMESE NETWORK AND NON-LOCAL ATTENTION

1) SIAMESE NETWORK

A Siamese network is comprised of two input branches, namely the template branch and the search branch. Their corresponding input data are the template image Z (a small area containing the tracked target) and the search region image X

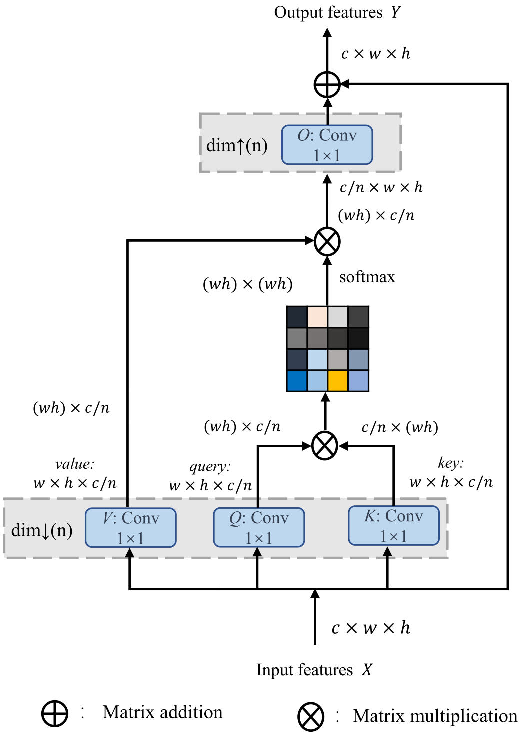


FIGURE 2. The original Non-Local attention module.

(a much larger one centered at the last estimated position of the target), respectively. Feature weights are extracted from the original image patches by a shared backbone, which is crucial for generic similarity learning. Then similarity calculation between the template and the search image is performed. The Siamese paradigm can be formulated as follows:

$$s_\theta(Z, X) = f_\theta(Z) \star f_\theta(X) + b \cdot \mathbb{1}, \tag{1}$$

where f_θ denotes the feature embedding, $\star$ the naive cross-correlation operation [8], $s_\theta(\cdot, \cdot)$ the response map, and b the bias. The larger the value on the map is, the more probable the target is located in the corresponding candidate region.

2) THE ORIGINAL NON-LOCAL ATTENTION

The original Non-Local attention is illustrated in Fig. 2, the core mechanism of which is similar to that of a look-up-table. Three convolutional layers with a kernel size of 1×1 (denoted by f_Q, f_K, and f_V) are employed for dimensionality reduction in the channel domain (from c to c/n) and linear projection of the input feature map $X \in \mathbf{R}^{c \times w \times h}$ (n is set to 2 in this work). We let *query*, *key*, and *value* denote the outputs of each convolutional layer. Subsequently, the similarity between *query* and *key* is computed by matrix multiplication. Each row of the resulting matrix represents the similarity estimation between a specific feature vector on *query* and all feature vectors on *key*. Afterwards, the matrix is scaled and multiplied by *value*. The obtained visual features then pass through an up-dimensional convolutional layer (denoted by f_O) to restore the channel dimensionality from c/n to c. The final output $Y \in \mathbf{R}^{c \times w \times h}$ of the Non-Local module is derived by residual learning.

The Non-Local attention can be formulated as:

$$Y = \frac{1}{C(X)} g(X, X) h(X) + X, \tag{2}$$

where the function $h(X) = f_V(X)$ is used to calculate *value* in each position of X, g is a similarity calculator, and $C(\cdot)$ is the scale factor. In this work, we adopt the Embedded Gaussian function [24] to model the similarity between vectors:

$$g(X, X) = e^{f_Q(X)'^T f_K(X)'}, \tag{3}$$

where $f_j(X)' \in \mathbf{R}^{c \times (wh)}, j \in \{Q, K\}$ denotes the reshaped *query* or *key* $f_j(X)$. In this way, the similarity calculation together with the scaling transformation is equivalent to a *softmax* operation on $f_Q(X)'^T f_K(X)'$.

We define:

$$\mathrm{Atten}(X_1, X_2) = \mathrm{softmax}(f_Q(X_1)'^T f_K(X_2)') f_V(X_2)'. \tag{4}$$

In this way, Non-Local attention can be written as follows:

$$\begin{aligned} Y &= \mathrm{NonLocal}_{\mathrm{self}}(X) \\ &= f_O(\mathrm{Atten}(X, X)) + X, \end{aligned} \tag{5}$$

where $\mathrm{NonLocal}_{\mathrm{self}}$ represents the so-called *self-attention*. Moreover, a Non-Local module can also take different feature maps as inputs:

$$\begin{aligned} Y &= \mathrm{NonLocal}_{\mathrm{cross}}(X_1, X_2) \\ &= f_O(\mathrm{Atten}(X_1, X_2)) + X_1, \end{aligned} \tag{6}$$

where $X_1 \in \mathbf{R}^{c \times w_1 \times h_1}$, $X_2 \in \mathbf{R}^{c \times w_2 \times h_2}$, and $Y \in \mathbf{R}^{c \times w_1 \times h_1}$. $\mathrm{NonLocal}_{\mathrm{cross}}$ is interpreted as a *cross-attention* operation.

C. CHANNEL- AND SPACE-AWARE FEATURE ENHANCEMENT

An essential requirement of object tracking is that a tracker should have the ability to locate an agnostic object, including that in categories not covered in its training sets. In practice, when objects of the same category appear at the same time, it is still challenging for the tracker to continuously identify and locate one of the objects. Clearly, target-specific features are substantially crucial for the tracker to discriminate the target from distractors. Nevertheless, how to obtain high-quality representation for a specific target remains to be explored.

To attack this issue, we design a channel- and space-aware feature enhancement (CSE) block. This block is devoted to exploiting prior information in the initial frame and merging the information into the data flow of Siamese networks. The block is comprised of three submodules, i.e., Spatial Attention, Foreground-Background Attention, and Channel Attention, which are illustrated in Fig. 3. It should be noted that, since the procedure of feature enhancement for X^t and X^{t-1} are exactly identical (the two search branches share their parameters in CSE), the one for X^{t-1} is not displayed in Fig. 3 for simplicity and clarity.

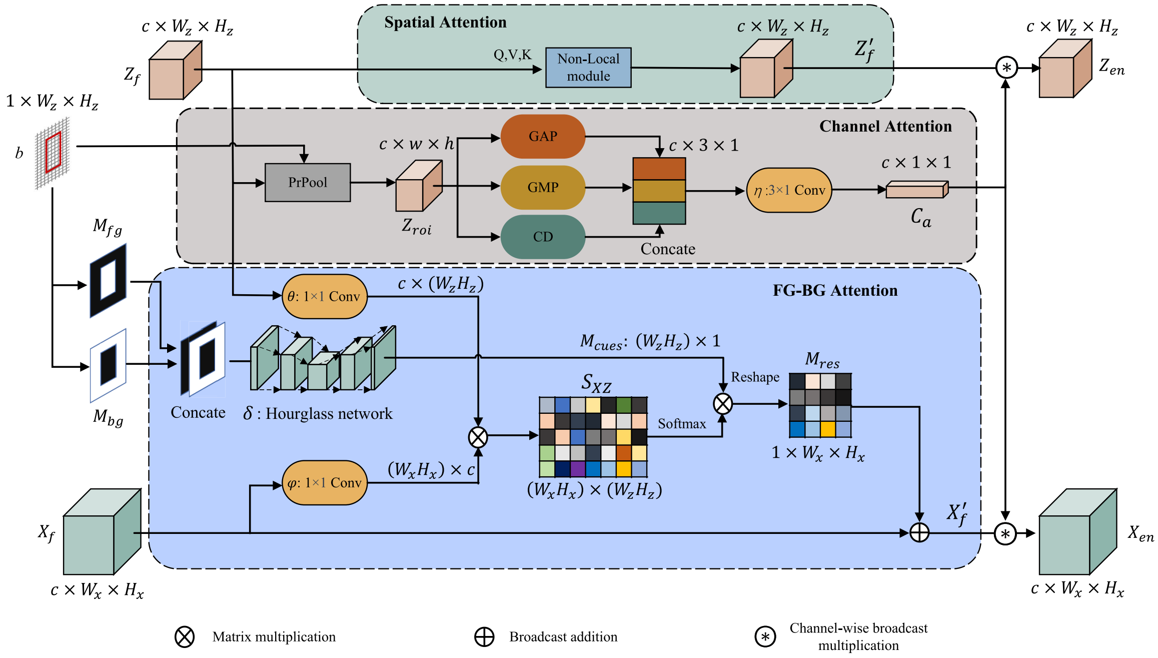


FIGURE 3. The architecture of CSE.

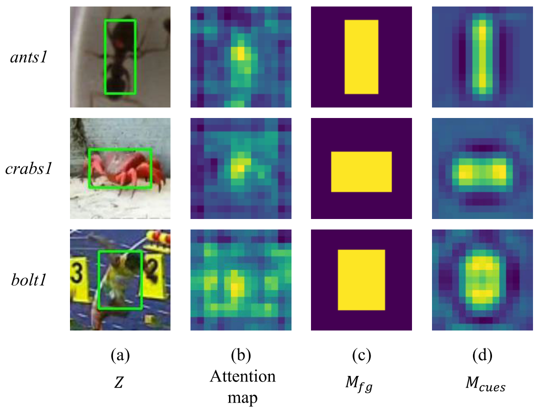


FIGURE 4. Visualization of (a) template image Z, (b) attention map derived from the Non-Local module in the template branch, (c) binary foreground mask M_{fg}, and (d) map of foreground-background cues M_{cues}. Three video sequences, *ants1*, *crabs1*, and *bolt1*, are derived from VOT2019 [27] dataset.

1) SPATIAL ATTENTION

A template feature map Z_f, which is extracted from the first frame in a video sequence, is usually fixed and stored in a cache pool during inference. Thus, the quality of feature representation of the tracked target is crucial for similarity calculation in subsequent frames. However, as exhibited in Fig. 4(a), background areas or distractors may be involved in the template image patches, which lead to inaccurate feature representation and weaken the reliability of the response maps obtained via cross-correlation calculation. Thanks to the notable ability of Non-Local attention in modeling long-range dependency, a self-attention module is employed to emphasize the target-related visual features and to suppress unnecessary representation. The enhanced feature map can be calculated by:

$$Z'_f = \mathrm{NonLocal}_{\mathrm{self}}(A_f). \tag{7}$$

In our empirical studies, a Siamese network can learn target-aware feature weights to some extent when employing a Non-Local attention module in the template branch. We visualize the attention heatmap[1] in the Non-Local module, and the results are shown in Fig. 4(b). We observe that the tracker equipped with Spatial Attention in the network tries to see the center area of the target rather than the background areas, which demonstrates the efficacy of Spatial Attention.

2) FOREGROUND-BACKGROUND ATTENTION

Previous Siamese networks pay little attention to the exploitation of prior information, e.g., the given annotation of the target in the initial image. We argue that target-specific cues provided during tracking should be fully utilized to meet the requirement of category-independent tracking. Thus, we develop a Foreground-Background (FG-BG) Attention to focus on the utilization of the prior information and the enhancement of target-aware representations in the search branch. Since the Siamese architecture developed for category-independent tracking is based on local matching, the consistency of target representations in templates and search regions is much significant for similarity calculation. With this in mind, FG-BG attention focuses on better utilization of target appearance cues in templates and improvement

[1] For the method of visualization, please refer to https://colab.research.google.com/github/facebookresearch/detr/blob/colab/notebooks/detr_attention.ipynb

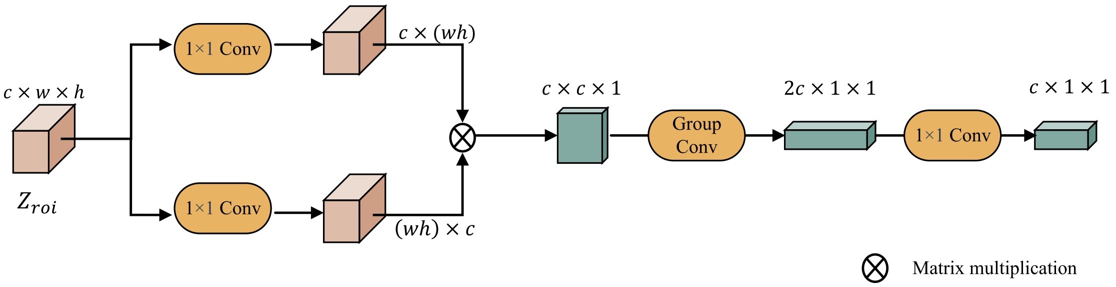


FIGURE 5. The architecture of CD.

of consistency between the search branch and the template branch for specific representations. The basic idea behind it is that the annotation bounding box can be interpreted as a spatial guidance for search region features—it marks out an area containing as many target cues and as little background information as possible.

Fig. 3 details the proposed FG-BG Attention. The binary foreground mask M_{fg} and background mask M_{bg} have the same spatial size as the template feature map Z_f. Both masks are derived from the annotation bounding box in the initial frame. They are firstly concatenated along channel dimension (Concate) and fed to an hourglass network δ, which is comprised of two stacked convolutional layers and two up-sample layers. The hourglass network can learn a spatial distribution of target-related regions on the template feature map, which is represented by a single-channel map M_{cues}:

$$M_{cues} = \delta(\text{Concate}(M_{fg}, M_{bg})), \tag{8}$$

where δ is the hourglass network as shown in Fig. 3. The map M_{cues} is a soft mask that contains foreground-background cues and is merged into the data flow of the search branch as a guidance signal. The rest of FG-BG Attention is considered as a *cross-attention* operation, which can be formulated by:

$$X_f' = \text{softmax}\left(\varphi(X_f)'^{T}\theta(Z_f)'\right) M_{cues} \oplus X_f, \tag{9}$$

where $\oplus$ denotes broadcast addition, and φ and θ are the feature embeddings. Herein, $\varphi(X_f)$, $\theta(Z_f)$, and M_{cues} play the same roles as *query*, *key*, and *value* in a standard cross-attention module, respectively. According to the properties of Non-Local attention, elements of *value* greater than and less than zero can be used to enhance the response on the similar elements of *query* and *key* and suppress the others via matrix multiplications as well as additions, respectively. Based on that, in FG-BG attention, target feature representations in the search branch can be enhanced by specifying positive and negative activation values on *value* (i.e., M_{cues}) to distinguish between targets and distractors. In this regard, the hourglass network plays a key role. It learns from M_{fg} and M_{bg} and automatically marks target-related and non-target areas on M_{cues} using positive and negative response values respectively. More discussions of M_{cues} are presented in Section IV-C2.

3) CHANNEL ATTENTION

Since the input branches share their backbone, feature maps derived from the branches tend to show the same activation pattern in the channel domain when representing the same object. We expect Siamese networks to learn the pattern respecting the channel distribution automatically.

For this purpose, we impose Channel Attention into the CSE block. The diagram of the submodule is shown in Fig. 3. This submodule is tailored to exploit target-specific representation over feature channels of Z_f and to reinforce the consistency between Z_f and X_f. It leverages a Precise RoI Pooling [59] (PrPool) layer to extract RoI feature weights Z_{roi} from the template feature map Z_f, which is supervised by the given annotation bounding box b, and then captures channel cues over Z_{roi} via three different units, i.e., GAP, GMP, and CD.

The units GAP and GMP perform global average pooling and global max pooling over the spatial domain on feature map Z_{roi}, respectively. CD is another core unit of the Channel Attention submodule, details of which are presented in Fig. 5. It serves to find a convolutional embedding, in which channel dependency within the input feature map can be captured by matrix multiplication. The resulting matrix is interpreted as a new feature map, adjusted via a group convolutional layer (Group Conv), and mapped linearly to the original feature space through a convolutional layer with a kernel size of 1×1 (1×1 Conv). In general, GAP provides generic representation of the target in feature channels, GMP gathers channel-wise target-distinctive cues [60], and CD identifies the inter-channel dependency. Thus, the three units in Channel Attention are complementary to each other.

Channel cues derived from the units are represented by three feature vectors with the same size of $c \times 1 \times 1$, which serve as supervisions for target-oriented feature enhancement. To be integrated into the raw visual features, they are concatenated along the spatial dimension and fed to a kernel-asymmetric convolutional layer η (with a kernel size of 3×1) for information integration. In this way, the final feature vector with the size of $c \times 1 \times 1$ can be simply merged into the data flow of both the template branch and search

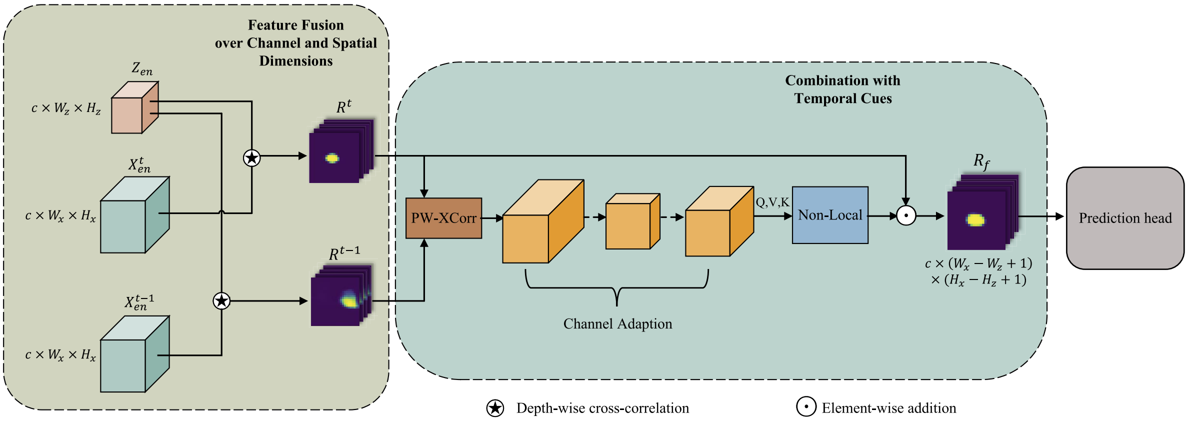


FIGURE 6. The overall structure of MDI-XCorr. Branches for classification and regression are with the same architecture, and only the classification one is displayed for simplicity.

branch by channel-wise broadcast multiplication. Channel Attention C_{att} can be formulated as:

$$C_{att} = \eta(\text{Concate}(\text{GAP}, \text{GMP}, \text{CD})), \tag{10}$$

where $Z_{roi} = \text{PrPool}(Z_f, b)$. In this way, the enhanced feature maps can be written as:

$$\begin{aligned} Z_{en} &= Z_f' \circledast C_{att}, \\ X_{en} &= X_f' \circledast C_{att}, \end{aligned} \tag{11}$$

where $\circledast$ denotes channel-wise broadcast multiplication.

D. FUSION OF MULTI-DIMENSIONAL CUES

The seminal Siamese network is not concerned with the temporal cues involved in the video data. Most recent works on the utilization of temporal cues focused on improving the template branch, e.g., replacement [22] or adjustment [23] of the template features. Nevertheless, they failed to make use of the relative position change of the tracked object w.r.t. the background or distractors for accurate target positioning. Different objects tend to have different movement patterns that can be utilized as a criterion to distinguish between the target and others. We expect Siamese trackers to learn to analyze and capture the movement of objects by considering such the property as prior knowledge in videos.

With the motivation of that, we improve the cross-correlation module in Siamese networks and propose a cross-correlation block with multi-dimensional information fusion (MDI-XCorr) to utilize temporal information in videos. It is arranged to aggregate feature weights from the template and search region over the spatial and channel domains and to mine motion cues of objects in response maps of adjacent frames. Compared with previous methods, our method aims to explore target motion on response maps rather than template features. This is because the response maps embrace diverse visual features derived from both the template and search region images, the consistency of which is boosted in CSE in advance. Meanwhile, not only the tracked target but also distractors are involved in a response map. This makes it feasible for Siamese networks to learn to capture and model the relative motion of the target w.r.t. distractors when the CSE block and the MDI-XCorr block are both employed in networks.

The MDI-XCorr block is comprised of two components, Feature Fusion over Channel and Spatial Dimensions and Combination with Temporal Cues. Fig. 6 details the architecture of the MDI-XCorr block, where Z_{en}, X^t_{en}, and X^{t-1}_{en} are the feature maps that have been enhanced in the CSE block. Triplet feature maps are taken as inputs of MDI-XCorr, in which two depth-wise cross-correlations are firstly performed, and the response maps $R^t, R^{t-1} \in \mathbf{R}^{c\times(W_X-W_Z+1)\times(H_X-H_Z+1)}$ on adjacent frames X^t and X^{t-1}, respectively, are induced. R^t and R^{t-1} embrace information about channel-wise and position-wise correlation. Since the response maps at different time points (t and $t-1$) are in the same embedding space, it is feasible for them to be explicitly fused. Thus, we employ a pixel-wise cross-correlation [61] (PW-XCorr) layer to explore the motion cues of objects on R^t and R^{t-1}.

PW-XCorr is a variant of naive cross-correlation. Given a template feature map $T \in \mathbf{R}^{c\times w_0 \times h_0}$ and a search region feature map $S \in \mathbf{R}^{c\times w_1 \times h_1}$, the PW-XCorr operation generates a feature map $F \in \mathbf{R}^{(w_0 h_0)\times w_1 \times h_1}$. It can be formulated as:

$$F = \{F_n \mid F_n = T_n \star S\}_{n\in\{0,1,\ldots,w_0\times h_0-1\}}, \tag{12}$$

where $\star$ represents the naive cross-correlation. To be specific, when performing PW-XCorr, we consider the feature vector T_n at each spatial position on the template feature map T as a kernel for naive cross-correlation calculation. The basic idea behind the adoption of PW-XCorr is that PW-XCorr intrinsically models the pair-wise relationship of vectors (corresponding to small candidate windows in the raw images) on two feature maps, which allows searching for the same object on both maps and capturing information about the relative position to the others. Besides, since both the naive cross-correlation and the depth-wise one [10] consider the whole template features as a correlation kernel, the integrity of spatial information of the response map can be virtually broken. Differently, PW-XCorr maintains spatial information in detail, meanwhile keeping the size (width and height) of the output feature map consistent with that of the inputs.

In fact, PW-XCorr transfers spatially relevant information about the objects into the channels of the response map. In order to recover the information back to the spatial domain, the resulting response map is delivered to successive stacks of convolutional layers (Channel Adaption) as shown in Fig. 6. It should also be noted that PW-XCorr can only detect the movement of each object but cannot find the target in the response maps according to the characteristics of the PW-XCorr operation. Hence, a vanilla Non-Local module is employed in MDI-XCorr to recognize the most salient target signal that has been highlighted in CSE. The output of Non-Local is merged into the response map R^t of the current frame via a residual connection, and the final feature map R_f embracing multi-dimensional (channel, spatial, and temporal) cues is fed to a classification head and a regression head for further target state estimation.

E. TRAINING AND INFERENCE

1) TRAINING STRATEGIES

Previous Siamese trackers that aim at taking advantage of temporal information in videos have certain requirements for training strategies, including removing image datasets [58] and sampling training data continuously [23] (from the initial frame to the last one in a video sequence). We expect our proposed network to benefit from general training settings and sampling strategies designed for Siamese trackers, especially the data augmentation methods that improve the discriminative ability of the tracker by learning from semantic negative pairs in object detection datasets [17].

To this end, we make a few modifications to conventional training strategies tailored for Siamese trackers. In adaption to the triplet input of DPINet, from video datasets, we take triplet images in the same sequence as a positive sample, while the template and search region images from different sequences are considered as a negative sample. It is worth noting that X^t and X^{t-1} are always sampled from adjacent frames. To take advantage of multi-category annotation in object detection datasets, we treat both template and search region images containing the same instance as a positive sample and those involving different instances as a negative sample. X^t and X^{t-1} are always the same image patch in case of being generated from image datasets.

Moreover, previous Siamese networks take double images as inputs. In this case, the back-propagation gradients in the template branch and the search branch are symmetric. Nevertheless, the search branch in DPINet receives double images in training, resulting in the propagation of a double gradient when parameter optimization of the search branch is performed. Such an asymmetric learning pace may lead to poor convergence of the network. To address this problem, we detach the gradient flow respecting X^{t-1} in training to maintain the consistency of parameter optimization in the input branches.

By simply adjusting training strategies with the approaches mentioned above, our proposed Siamese network can make full use of numerous offline training samples and be trained end-to-end.

2) LOSS FUNCTION

Following [11], we employ the center-based anchor-free heads, including a classification head and a regression head, for target state estimation. We adopt cross-entropy loss L_{CE} and IoU (Intersection over Union) loss L_{IoU} as their loss function, respectively. The final training objective is defined as follows:

$$loss = \lambda_1 L_{\mathrm{CE}} + \lambda_2 L_{\mathrm{IoU}}, \tag{13}$$

where λ_1 and λ_2 are set to 1 in network training. More details about center-based anchor-free settings are referred to [11].

3) INFERENCE

The proposed tracker is initialized using the initial frame, and the template features are cached and not released until the end of inference on a video. For subsequent detection on video frames, the tracker receives a single image continuously, predicts the target state using the cached response features corresponding to the previous frame as specified in Section III-A (the MDI-XCorr block does not utilize motion cues when $t = 1$), and caches the necessary data corresponding to the current frame for tracking on the next frame. In this way, despite only two adjacent frames taken for feature interaction fusion, the features of the reference frame used for motion detection are constantly updated to maintain the utilization of motion cues.

IV. EXPERIMENTS

A. IMPLEMENTATION DETAILS

We apply general training settings for DPINet (except a few adaptions discussed in section III-E1). To be specific, we set the mini-batch size to 48 and the training epoch to 20. A Stochastic Gradient Descent (SGD) optimizer is deployed for network training, momentum and weight decay of which are set to 0.9 and 0.0001, respectively. A warm-up learning rate from 0.002 to 0.01 is adopted in the first 5 epochs. During the rest of the training phase, the learning rate decays exponentially from 0.01 to 0.0001. ResNet-50 [32] pretrained on ImageNet dataset [62] is employed as the backbone of DPINet, and we remove its layers after the fourth stage for computational efficiency. All backbone parameters are frozen in the first 10 epochs, and parameters of the third and fourth stages in the last 10 epochs are unfrozen and fine-tuned with a small learning rate of 0.1 times. Six datasets are used for offline training, including GOT-10k [28], MS COCO [21], LaSOT [29], ImageNet VID [62], ImageNet DET [62], and Youtube-BB [63]. The size of template images is set to 127×127 and that of search images is set to 255×255. All of

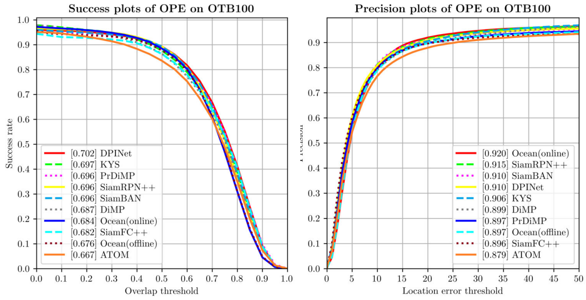


FIGURE 7. Success and Precision plots on OTB100 benchmark. AUC scores and distance precision rates at the threshold of 20 are displayed and ranked in the legends.

the images are sampled centering on the targets, with random center shift, random blur, and random color enhancement as data augmentation. This work is implemented in Python using Pytorch on a PC with Nvidia RTX 2080ti. The code is available at https://github.com/qq1018408006/DPINet.

B. COMPARISON WITH SOTA CNN-BASED TRACKERS

In this section, we compare our proposed tracker DPINet with other SOTA CNN-based methods on four prevalent benchmarks, i.e., OTB100 [25], VOT2018 [26], VOT2019 [27], GOT-10k [28] test set, and LaSOT [29] test set.

1) OTB100

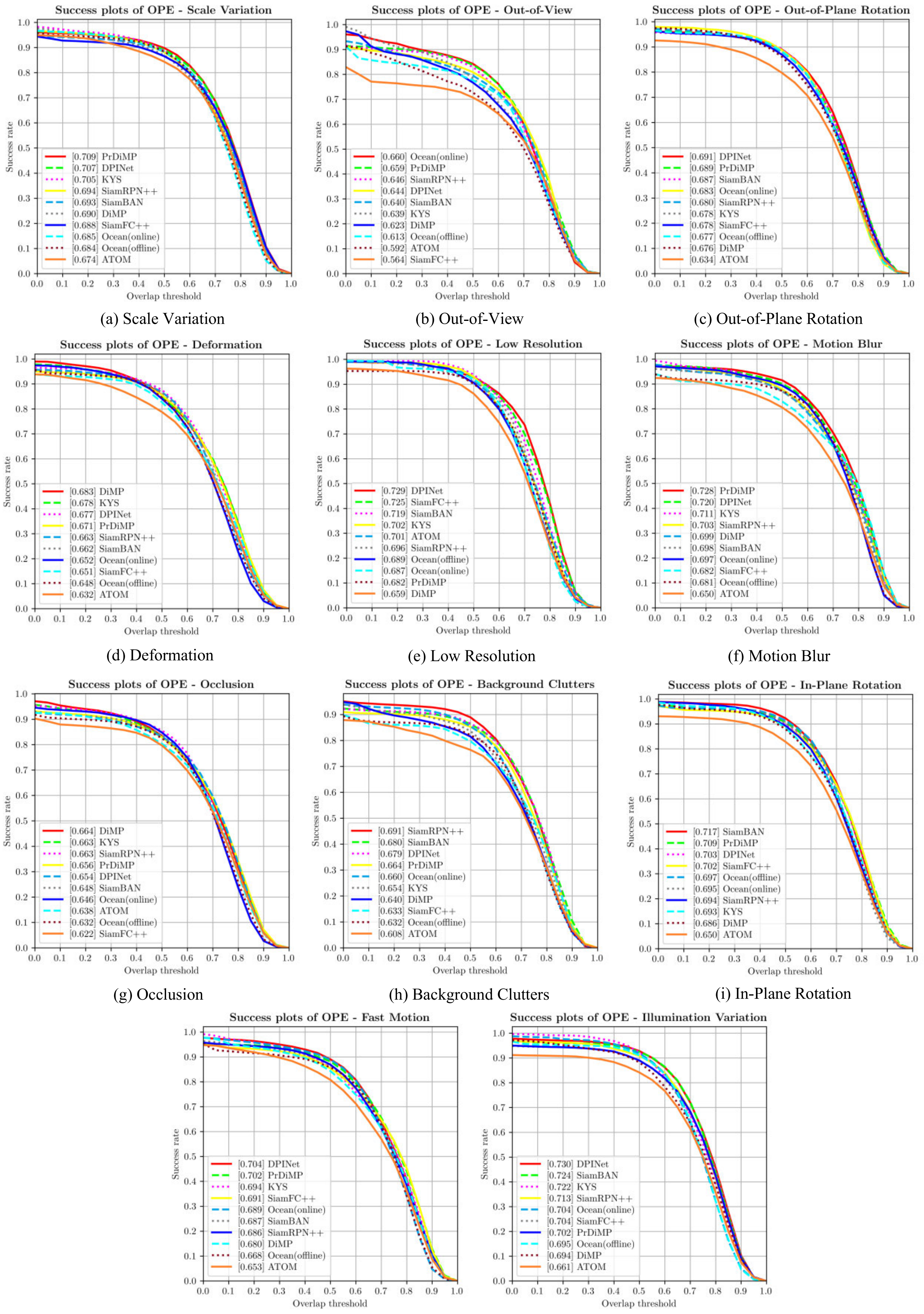
OTB100 [25] is a classical benchmark in object tracking. It contains 100 public video sequences with average frames of 590. The videos are collected in many challenging scenarios, including background clutters, low resolution, deformation, etc. We conform to the criteria in [25] and conduct One-Pass-Evaluation (OPE) for our tracker with metrics of Precision and Area Under Curve (AUC) of the Success plot. The proposed DPINet is compared with SiamRPN++ [10], ATOM [4], SiamFC++ [18], Ocean (offline) [64], Ocean (online) [64], DiMP [5], SiamBAN [11], PrDiMP [6], and KYS [7], evaluation curves of which are presented in Fig. 7. Our proposed tracker achieves a leading performance on AUC of 0.702, outperforming all of the other comparison trackers, particularly the recent powerful DCF methods KYS, PrDiMP, and DiMP. Besides, DPINet ranks fourth on Precision evaluation, showing a favorable achievement as well.

We also analyze the performance of trackers under several video attributes in Fig. 8. We can see that the proposed method is effective in dealing with Out-of-Plane Rotation (c), Low Resolution (e), and Illumination Variation (k), obtaining the highest score on AUC of 0.691, 0.729, and 0.730, respectively. This is mainly ascribed to the well-designed CSE block, which facilitates to capture high-quality and target-specific information from the initial frame despite the aforementioned disadvantages. Besides, DPINet can handle Fast Motion (j) well and achieves a top-ranked performance (an AUC score of 0.704) in the challenging scenario. It indicates that the tracker DPINet equipped with MDI-XCorr can cope with fast-moving targets better than its counterparts, demonstrating the advancement of our proposed MDI-XCorr block. The proposed tracker does not perform well in Out-of-View (b) and Occlusion (g), ranking fourth and fifth, respectively. This is attributed to the inherent flaw of the Siamese trackers. That is, when some portion of the target leaves the view of search region or is blocked by distractors, the response maps obtained from appearance-based similarity calculation are hardly reliable. In these cases, a false positive is usually predicted by the tracker, causing the target bounding box to drift. Moreover, despite the lack of online adaptation of our tracker, DPINet still shows satisfactory results in the remaining challenging scenarios compared to the powerful DCF trackers or updating-based approaches including DiMP, PrDiMP, KYS, and Ocean (online).

We additionally report the tracking results on several sequences of OTB100. As shown in Fig. 9, our approach is more responsive to challenges than other trackers. From Fig. 9(a) and (b), more specifically, we can find that DPINet is good at tracking fast-moving targets. In contrast, ATOM, Ocean (online), and PrDiMP are prone to drift, leading to inferior tracking quality. Fig. 9(c) and (d) illustrate trackers' ability to deal with distractors. The proposed tracker can handle the challenge factor well, endeavoring to infer a tightly wrapped bounding box for the tracked object. ATOM and SiamFC++, sensitive to distractors, can hardly identify targets in these two videos. DPINet also performs well in open outdoor scenes and indoor scenes, tracking results of which are displayed in Fig. 9(e), (f) and Fig. 9(g), (h), respectively. Particularly, performances of most DCF trackers, i.e., ATOM, DiMP, and PrDiMP, are inferior to that of DPINet when the illumination changes dramatically as Fig. 9(g) and (h) reveal. It suggests that the online-updating strategy adopted by the DCF family is barely effective in this case, and contrastingly, the utilization of diverse prior information does enable a favorable tracking performance for the Siamese trackers.

2) VOT2018 AND VOT2019

The VOT datasets embrace videos with more severe deformation of objects than those in OTB100. Meanwhile, the annotations for targets are not axis-aligned bounding boxes but rotated ones, which is more challenging for high-quality tracking. Both VOT2018 [26] and VOT2019 [27] contain 60 video sequences. Three measures in this type of benchmark are Expected Average Overlap (EAO), Accuracy, and Robustness. The higher the scores of Accuracy and EAO are, the better the tracker performs, while the opposite is true for Robustness. We compare DPINet with SiamRPN++ [10], ATOM [4], SiamFC++ [18], DiMP [5], SiamBAN [11], Ocean (offline) [64], Ocean (online) [64], PrDiMP [6], STMTrack [54], and SiamRN [65] on VOT2018, and with SiamRPN++ [10], SiamMask [66], SPM [67], ATOM [4], Ocean (offline) [64], Ocean (online) [64], and SiamRCR [68] on VOT2019. Comparative results on the two benchmarks are detailed in Table 1 and Table 2, where the top three results of each metric are boldfaced, underlined, and



(a) Scale Variation

(b) Out-of-View

(c) Out-of-Plane Rotation

(d) Deformation

(e) Low Resolution

(f) Motion Blur

(g) Occlusion

(h) Background Clutters

(i) In-Plane Rotation

(j) Fast Motion

(k) Illumination Variation

FIGURE 8. Success evaluation on OTB100 in terms of challenge attributes, including (a) Scale Variation, (b) Out-of-View, (c) Out-of-Plane Rotation, (d) Deformation, (e) Low Resolution, (f) Motion Blur, (g) Occlusion, (h) Background Clutters, (i) In-Plane Rotation, (j) Fast Motion, and (k) Illumination Variation. Trackers are ranked based on their AUC scores.

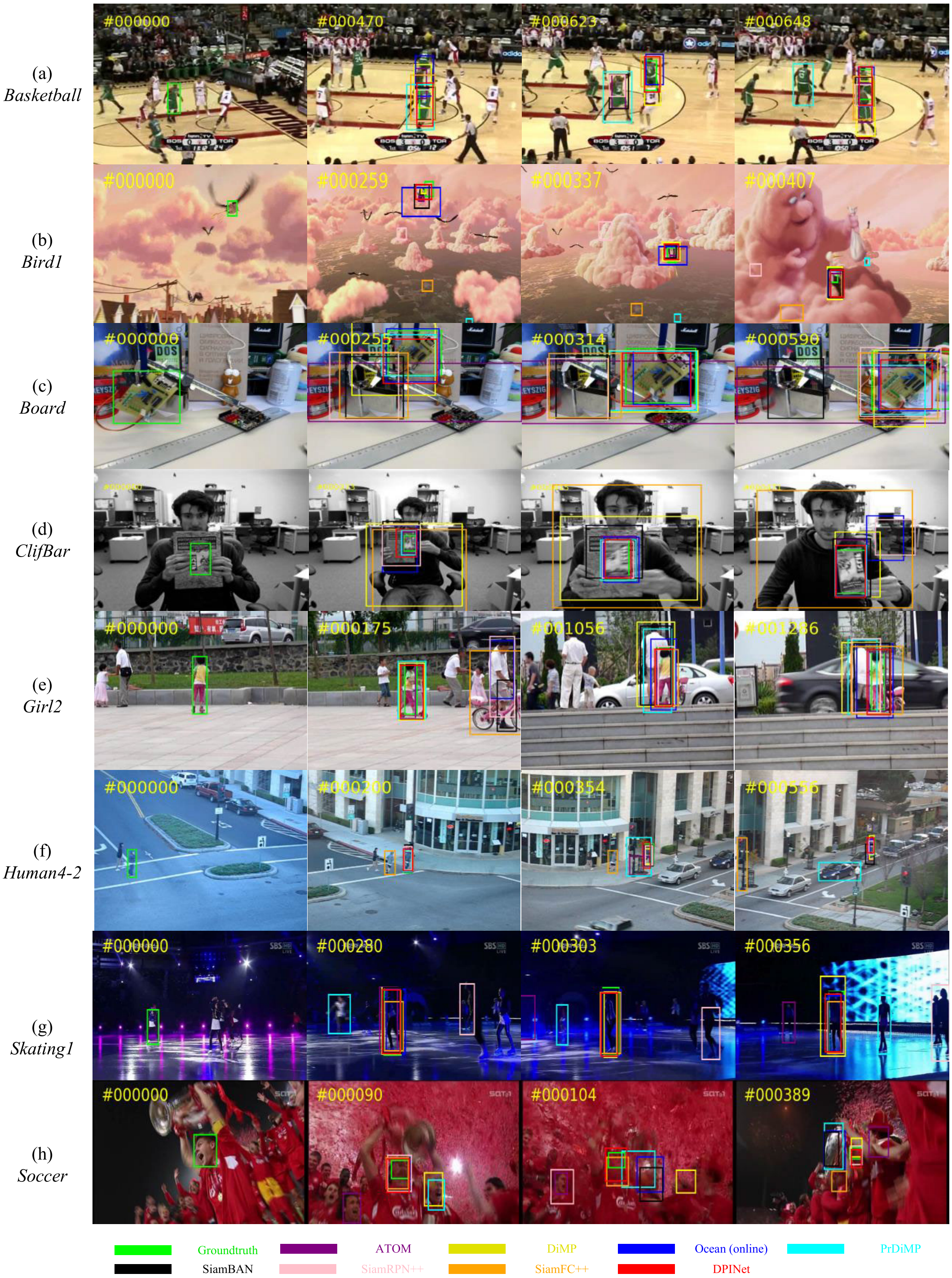


FIGURE 9. Tracking results of the comparison trackers on 8 challenging video sequences of OTB100, including (a) *Basketball*, (b) *Bird1*, (c) *Board*, (d) *ClifBar*, (e) *Girl2*, (f) *Human4-2*, (g) *Skating1*, and (h) *Soccer*.

italicized, respectively. As shown in Table 1, our proposed method achieves an EAO score of 0.474 and an Accuracy score of 0.609 on VOT2018, ranking second in terms of the two metrics. It should be noticed that both the top-ranked trackers in terms of EAO and Accuracy, Ocean (online) and PrDiMP, employ an online updating module that introduces heavy computation loads during inference (see the running speeds provided in Table 3). Differently, the proposed DPINet achieves a proper balance of EAO and Accuracy. It runs in an updating-free manner, which is more computationally efficient. In regard to the evaluation on VOT2019 (Table 2), DPINet achieves the second best performance on EAO of 0.336, trailing the updating-based tracker Ocean (online) by a margin of 0.014. Our tracker ranks first and second in terms of

TABLE 1. Performance comparisons on VOT2018 in terms of EAO, Accuracy, and Robustness.

Tracker	SiamRPN++	ATOM	DiMP	SiamFC++	SiamBAN	Ocean (offline)	Ocean (online)	PrDiMP	STMTrack	SiamRN	DPINet
EAO	0.414	0.401	0.440	0.426	0.452	0.467	**0.489**	0.442	0.447	*0.470*	0.474
Accuracy	*0.600*	0.590	0.597	0.587	0.597	0.598	0.592	**0.618**	0.590	0.595	0.609
Robustness	0.234	0.204	*0.153*	0.183	0.178	0.169	**0.117**	0.165	0.159	0.131	0.164

TABLE 2. Performance comparisons on VOT2019 in terms of EAO, Accuracy, and Robustness.

Tracker	SiamRPN++	SiamMask	SPM	ATOM	Ocean (offline)	Ocean (online)	SiamRCR	DPINet
EAO	0.285	0.287	0.275	0.292	*0.327*	**0.350**	0.336	0.336
Accuracy	0.599	0.594	0.577	0.603	0.590	0.594	*0.602*	**0.607**
Robustness	0.482	0.461	0.507	0.411	*0.376*	**0.316**	0.386	0.351

TABLE 3. Evaluations on GOT-10k test dataset in terms of Average Overlap (AO) and Success Rate (SR, the subscript 0.5 indicates the threshold of the metric).

Tracker	SiamRPN++	ATOM	DiMP	Ocean (offline)	Ocean (online)	SiamFC++	SiamCAR	PrDiMP	DPINet
AO	0.518	0.556	*0.611*	0.592	*0.611*	0.595	0.569	**0.634**	0.613
$SR_{0.5}$	0.618	0.634	0.717	0.695	0.721	0.695	0.670	**0.738**	*0.718*
Speed (FPS)	35	30	43	*58*	25	**90**	52	30	71

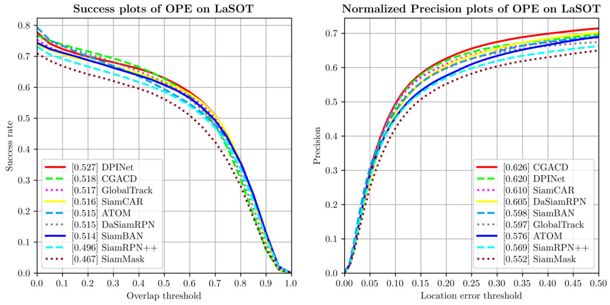


FIGURE 10. Success and Precision plots on LaSOT test set.

Accuracy and Robustness, respectively, outperforming most of the comparison trackers. These results confirm the superiority of the well-established Siamese tracker DPINet.

3) GOT-10k

GOT-10k [28] is a popular large-scale dataset derived from the wild. Following protocols in [28], we retrain DPINet on GOT-10k training set and evaluate our tracker on the test set for a fair comparison. Two metrics of the benchmark are Average Overlap (AO) and Success Rate (SR). Comparative results of SiamRPN++ [10], ATOM [4], Ocean (offline) [64], Ocean (online) [64], SiamFC++ [18], SiamCAR [19], DiMP [5], PrDiMP [6], and DPINet are presented in Table 3, where the top three results of each metric are boldfaced, underlined, and italicized, respectively. Our method achieves desirable performance, ranking second with an AO score of 0.613. The proposed tracker outperforms the online trackers, Ocean (online) and DiMP, by a margin of 0.2% on AO (0.613 vs. 0.611). In addition, although surpassed by Ocean (online) and PrDiMP in terms of $SR_{0.5}$, our approach runs more than twice as fast as them (71 FPS vs. 25 FPS and 30 FPS). DPINet ranks second in terms of operation speed, meeting the requirement of real-time operation and achieving a better balance between the tracking quality and the inference speed. Overall, DPINet shows comparable performance against most SOTA DCF trackers and excellent achievement among Siamese ones.

4) LaSOT

LaSOT is one of the recently released large-scale object datasets. Its test subset offers 280 video sequences with 70 categories and more than 680k frames along with high-quality annotations. We follow protocol II [29], retrain DPANet on LaSOT training set, and evaluate the tracker on the test set. The proposed tracker is compared with SiamMask [66], DaSiamRPN [17], SiamRPN++ [10], ATOM [4], GlobalTrack [69], SiamCAR [19], SiamBAN [11], and CGACD [70]. Since the precision metric is sensitive to the target size and image resolution [29], herein, we take AUC of the Success plot and Normalized Precision as metrics for evaluations. As shown in Fig. 10, the proposed approach achieves a leading AUC score of 0.527, which is 0.009 higher than the second place. Besides, DPINet ranks second in terms of Normalized Precision by a score of 0.620, lagging behind the recently proposed two-stage refinement tracker CGACD by only 0.006. Moreover, compared with SiamBAN, which employs the same anchor-free prediction heads, our DPINet improves the scores by 0.013 and 0.022 respectively in terms of the two metrics. All the comparative results can demonstrate the favorable performance of the well-established DPINet in large-scale object tracking.

C. ABLATION STUDY

In this section, we conduct a component-by-component analysis, involving the proposed CSE and MDI-XCorr block.

TABLE 4. Ablation study of different blocks and modules of DPINet on OTB100 and VOT2018.

Tracker	CSE			MDI-XCorr			AUC (OTB100)	EAO (VOT2018)	Speed
1	-			-			0.674	0.377	117 FPS
2	✓			-			0.700	0.442	112 FPS
3	-			✓			0.673	0.370	88 FPS
4	✓			✓			0.702	0.474	71 FPS
	SA	FGBG	CA	PW-XCorr	Channel Adaption	Non-Local			
5	-	✓	✓	-	-	-	0.694	0.398	113 FPS
6	✓	-	✓	-	-	-	0.689	0.366	114 FPS
7	✓	✓	-	-	-	-	0.690	0.380	114 FPS
8	-	-	-	-	✓	✓	0.690	0.410	93 FPS
9	-	-	-	✓	-	✓	0.056	0.012	—
10	-	-	-	✓	✓	-	0.669	0.338	92 FPS
11	✓	✓	✓	-	✓	✓	0.685	0.440	87 FPS

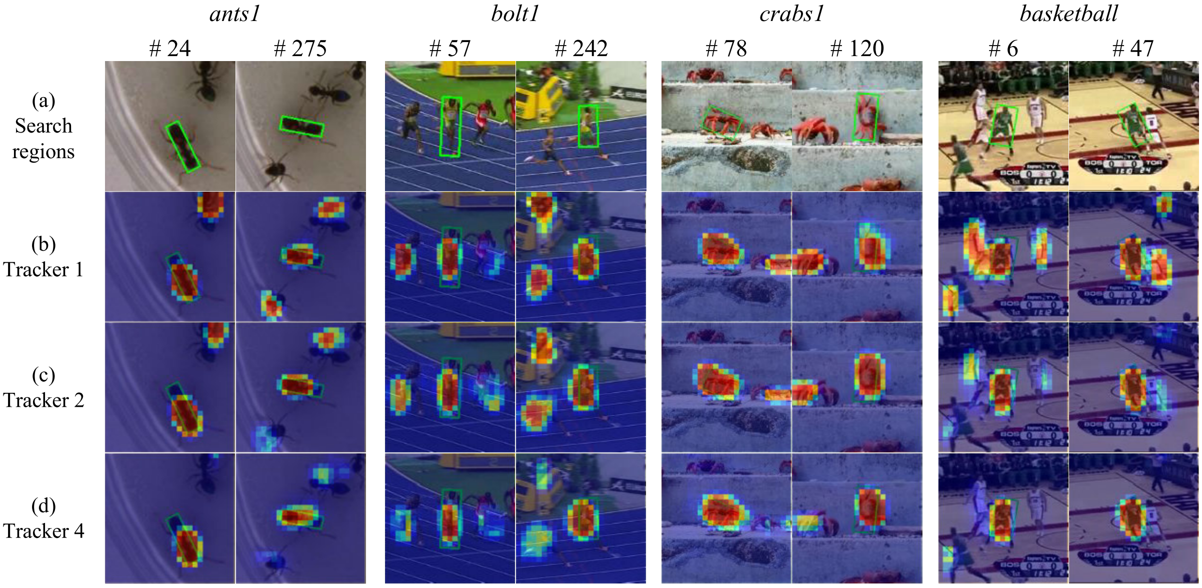


FIGURE 11. Visualization of (a) search region patches and corresponding classification maps output by the classification head of (b) tracker 1, (c) tracker 2, and (d) tracker 4 on four sequences of VOT2019.

All evaluations for variants of DPINet are conducted on OTB100 and VOT2018. We prune each key block or module of the original tracker separately and record whether there is any degradation or improvement in tracking performance. Table 4 shows the architecture as well as the performance of each variant, where the equipping of each key module is indicated by ✓and the removal of the module is indicated by -. As shown in the table, the first group of trackers (from tracker 1 to tracker 4) is evaluated to demonstrate the efficacy of CSE and MDI-Xcorr as explained in Section IV-C1. Moreover, the second group of experiments on variants (from tracker 5 to tracker 10) is conducted to reveal the potential of each key module within CSE and MDI-XCorr as mentioned in Section IV-C2 and Section IV-C3, and the additional group of variants (tracker 11) is assessed to demonstrate complementarity between CSE and PW-XCorr as discussed in Section IV-C3.

1) DISCUSSION ON THE PROPOSED BLOCKS

We conduct experiments to evaluate the effects of the proposed CSE and MDI-XCorr blocks. Experimental results are shown in Table 4. Tracker 4 with both CSE and MDI-XCorr employed represents the proposed DPINet, and tracker 1 without any extra blocks is taken as the baseline. When activating the CSE block, the baseline yields a substantial gain of 0.026 (0.700 vs. 0.674) and 0.065 (0.442 vs. 0.377) in terms of AUC (OTB100) and EAO (VOT2018), respectively. The CSE block is actually a lightweight component, which introduces few computational burdens into tracker 2 during inference (a running speed reduction of 5 FPS compared to tracker 1). The tracking quality can be further improved (a gain of 0.002 on AUC and 0.032 on EAO) once another core component MDI-XCorr is prepared, as the evaluations of tracker 4 and tracker 2 in Table 4 reveal. The combination of temporal information brings a

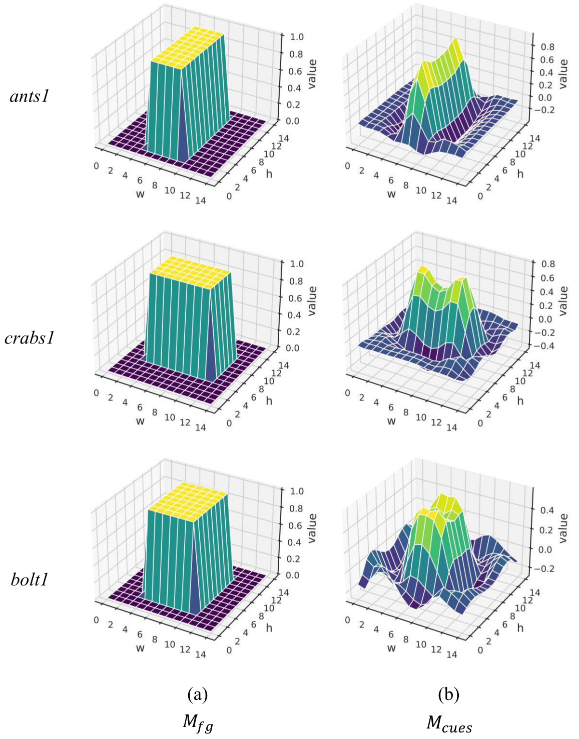


FIGURE 12. Three-D visualization of (a) M_{fg} and (b) M_{cues} on *ants1*, *crabs1*, and *bolt1* of VOT2019.

deal of computational consumption, but the tracker can still run at real time (71 FPS). Interestingly, the baseline's performance is degraded once equipped with only the MDI-XCorr block—a performance drop of 0.001 on AUC and 0.007 on EAO (tracker 3 vs. tracker 1). This is because, as mentioned in section III-D, the Non-Local module in MDI-XCorr endeavors to capture the most salient cues in feature maps, which relies on the enhancement of target-related features in CSE. That is, the MDI-XCorr block can hardly discriminate between the target and distractors by itself. The comparison between tracker 4 and tracker 3 in Table 4 confirms the relationship between the two blocks—CSE makes MDI-XCorr more effective, meanwhile delivering a performance gain of 0.029 on AUC and 0.104 on EAO.

We additionally report the classification maps output by tracker 1, 2, and 4 in Fig. 11. The baseline (tracker 1) is sensitive to all distractors around the tracked target and fails to distinguish between the distractors and the target since no target-specific prior information is used (row (b)). Once equipped with the CSE block (tracker 2), the tracker is able to take account of the target-specific cues and discriminate objects with spatially distinctive characteristics from the target, such as the ant that is climbing in *ants1* and the players dressed differently in *basketball* (row (c)). In this case, nevertheless, the tracker cannot deal with objects with a similar appearance to the tracked target. Upon the MDI block being prepared (tracker 4), objects with distinctly different movement patterns can be detected by the tracker, and such object motion cues can be incorporated into feature processing as a supervision to refine the response maps. In this way, objects with similar appearance but different motion modes, e.g., the runner in green near the tracked one in *bolt1* and the stationary crabs around the moving target in *crabs1*, can be identified (row (d)). Sharp peaks generated by them on the classification maps can be suppressed to some extent, which substantially improves tracking quality and stability of the baseline.

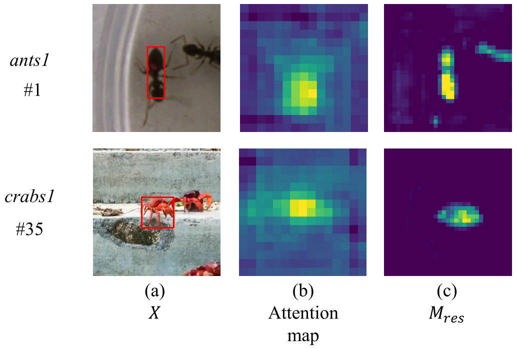


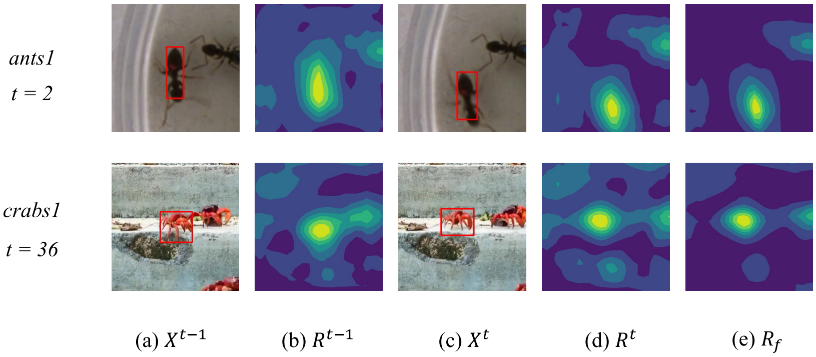
FIGURE 13. Visualization of (a) image patches X, (b) corresponding attention map within S_{XZ}, and (c) residual attention M_{res}.

2) DISCUSSION ON MODULES OF CSE

In this section, we discuss the impact of each submodule in the CSE block on tracking accuracy. We conduct a group of experiments with CSE by removing Spatial Attention (SA), Foreground-Background Attention (FG-BG), and Channel Attention (CA) separately, which corresponds to tracker 5, 6, and 7 in Table 4, respectively. Evaluations of the variants of tracker 2 are reported in Table 4. It can be observed that the performance of tracker 5, 6, and 7 is inferior to that of tracker 2. This suggests that all three submodules are virtually functional. Besides, tracker 6 and 7 show a more significant degradation than tracker 5 compared with tracker 2 (e.g., a drop of 7.6% and 6.2% against that of 4.4% on EAO). In other words, FG-BG and CA play a more crucial role than SA to improve tracking quality. This supports the benefit of using target-specific guidance (in both spatial and channel domains) to advance the representation power of visual features in Siamese networks.

For intuitive comprehension, foreground mask M_{fg} and 2-D visualization of M_{cues} in three videos are shown in Fig. 4(c) and (d), respectively. It can be observed that the hourglass network in FG-BG pays extra attention to the background region comparing M_{cues} with M_{fg}. Three-D visualization of M_{cues} and M_{fg} are additionally exhibited in Fig. 12. Clearly, the activation values in background areas of M_{cues} are smaller than 0. This is suitable to weaken the negative effects of background areas due to the nature of matrix multiplication in Non-Local attention.

Furthermore, the attention map in the similarity matrix S_{XZ} and residual map M_{res} in FG-BG Attention are displayed in Fig. 13(b) and (c), respectively. It indicates that the Non-Local module preliminarily searches for the most poten-



(a) X^{t-1} (b) R^{t-1} (c) X^{t} (d) R^{t} (e) R_f

FIGURE 14. **Visualization of image patches and corresponding feature maps. (a) Search region X^{t-1} at time point $t-1$. (b) Response map R^{t-1} corresponding to (a). (c) Search region X^{t} at time point t. (d) Response map R^{t} corresponding to (c). (e) Response map R_f refined and enhanced via MDI-XCorr.**

tial areas of the target as shown in Fig. 13(b). After aggregating guidances (Z_f and M_{cues}) into search region features, the block generates a residual mask M_{res} that visibly marks out the most target-related regions as illustrated in Fig. 13(c). The residual mask embraces barely distractors or background areas and is used to assign spatial areas on the search region feature map supposed to be directionally enhanced. These attention maps show that the block manages to see the target instead of the background areas or distractors, to which the favorable performance of DPINet is partly attributed, and it is also helpful for MDI-XCorr in capturing salient target cues.

3) DISCUSSION ON MODULES OF MDI-XCorr

Comprehensive experiments on MDI-XCorr are also conducted, involving the PW-XCorr operation (tracker 8), Channel Adaption (tracker 9), and the Non-Local module (tracker 10). First, we directly concatenate R^t and R^{t-1} along the channel domain instead of performing PW-XCorr operation in MDI-XCorr (tracker 8). As signified in Table 4, tracker 8 outperforms tracker 3 by a gain of 0.017 on AUC (OTB100) and 0.040 on EAO (VOT2018). That is, without the highlight of target-related cues in the upstream of network data flow (i.e., the enhancement of CSE), simple concatenation of response maps in adjacent search frames enables more accurate tracking for Siamese tracker than performing PW-XCorr calculation. However, the tracker combined with PW-XCorr can handle tracking tasks better in tracking once the CSE block in the network is prepared (tracker 4 against tracker 11): a performance gain of 0.017 and 0.034 on AUC (OTB) and EAO (VOT2018). In addition, the performances of tracker 9 on the two benchmarks are extremely poor, which suggests that the tracker fails to recognize any objects. This is not surprising at all since object motion cues in channels of the response map are not transferred back into the spatial dimension (as specified in section III-D), and the loss of spatial cues compromises the tracking performance. Additionally, the comparison between tracker 10 and tracker 3 shows that the Non-Local module in the MDI-XCorr block is indispensable as well—the removal of Non-Local attention in MDI-XCorr leads to a substantial performance drop of 0.004 on AUC and 0.032 on EAO.

We visualize search region patches and corresponding response maps in adjacent frames in Fig. 14 to demonstrate the efficacy of integrating temporal cues into visual features. As in video *crabs1*, for instance, two response maps R^t and R^{t-1} corresponding to search images X^t and X^{t-1} in adjacent frames, respectively, show high-value activations in background regions. Thanks to the combination with motion cues, the cluttered activations in background areas of the refined response map R_f tend to be suppressed to some extent, and distinctly sparse feature weights on the response map can be induced (comparing subfigure (e) with subfigure (d)). This reduces the sensitivity of the tracker to distractors, thereby improving the stability and accuracy in tracking.

V. CONCLUSION

In this work, we enable Siamese trackers to focus on the utilization of diverse prior information in single object tracking tasks for accurate object tracking. The information includes the target cues provided in the initial frame and motion cues involved in video sequences. In order to merge the prior knowledge into the data flow of Siamese networks, we propose two novel blocks: CSE and MDI-XCorr. The former considers target-specific representation as guidances and manages to enhance target-related feature weights on feature maps, and the latter endeavors to mine motion cues of objects in response maps of adjacent search region images. Extensive experiments are conducted to demonstrate the efficacy and efficiency of CSE and MDI-XCorr. Evaluations on four benchmarks suggest that our proposed tracker DPINet equipped with the two complementary blocks surpasses most CNN-based SOTA methods in terms of both accuracy and stability, achieving favorable performances.

REFERENCES

[1] J. Azimjonov and A. Özmen, "A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways," *Adv. Eng. Informat.*, vol. 50, Oct. 2021, Art. no. 101393.

[2] C. Luo, X. Yang, and A. Yuille, "Exploring simple 3D multi-object tracking for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10468–10477.

[3] V. A. Prisacariu and I. Reid, "3D hand tracking for human computer interaction," *Image Vis. Comput.*, vol. 30, no. 3, pp. 236–250, Mar. 2012.

[4] M. Danelljan, G. Bhat, F. S. Goutam, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Aug. 2019, pp. 4660–4669.

[5] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6181–6190.

[6] M. Danelljan, L. Van Gool, and R. Timofte, "Probabilistic regression for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7181–7190.

[7] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Know your surroundings: Exploiting scene information for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, vol. 12368, Aug. 2020, pp. 205–221.

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

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

[10] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4282–4291.

[11] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, "Siamese box adaptive network for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6667–6676.

[12] B. Huang, T. Xu, S. Jiang, Y. Bai, and Y. Chen, "SVTN: Siamese visual tracking networks with spatially constrained correlation filter and saliency prior context model," *IEEE Access*, vol. 7, pp. 144339–144353, 2019.

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

[14] Z. Zhang and H. Peng, "Deeper and wider Siamese networks for real-time visual tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4591–4600.

[15] B. Liao, C. Wang, Y. Wang, Y. Wang, and J. Yin, "PG-Net: Pixel to global matching network for visual tracking," in *Proc. Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 12367, Aug. 2020, pp. 429–444.

[16] D. Guo, Y. Shao, Y. Cui, Z. Wang, L. Zhang, and C. Shen, "Graph attention tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9543–9552.

[17] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware Siamese networks for visual object tracking," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 103–119.

[18] Y. Xu, Z. Wang, Z. Li, Y. Yuan, and G. Yu, "SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, pp. 12549–12556.

[19] D. Guo, J. Wang, Y. Cui, Z. Wang, and S. Chen, "SiamCAR: Siamese fully convolutional classification and regression for visual tracking," in *Proc. IEEE CVPR*, Jun. 2020, pp. 6268–6276.

[20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2009.

[21] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 740–755.

[22] J. Zhou, P. Wang, and H. Sun, "Discriminative and robust online learning for Siamese visual tracking," in *Proc. 34th AAAI Conf. Artif. Intell.*, Feb. 2020, pp. 13017–13024.

[23] L. Zhang, A. Gonzalez-Garcia, J. van de Weijer, M. Danelljan, and F. S. Khan, "Learning the model update for Siamese trackers," in *Proc. ICCV*, Oct. 2019, pp. 4009–4018.

[24] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7794–7803.

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

[26] M. Kristan et al., "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, Sep. 2018, pp. 3–53.

[27] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kamarainen, and L. C. Zajc, "The seventh visual object tracking VOT2019 challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2019, pp. 2206–2241.

[28] L. Huang, X. Zhao, and K. Huang, "GOT-10 k: A large high-diversity benchmark for generic object tracking in the wild," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1562–1577, May 2021.

[29] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5374–5383.

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1106–1114.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Feb. 2016, pp. 770–778.

[33] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[34] E. Wang, D. Wang, Y. Huang, G. Tong, S. Xu, and T. Pang, "Siamese attentional cascade keypoints network for visual object tracking," *IEEE Access*, vol. 9, pp. 7243–7254, 2021.

[35] J. Zhu, T. Chen, and J. Cao, "Siamese network using adaptive background superposition initialization for real-time object tracking," *IEEE Access*, vol. 7, pp. 119454–119464, 2019.

[36] C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su, "This looks like that: Deep learning for interpretable image recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8928–8939.

[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[38] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[39] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[40] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[41] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[42] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," *Int. J. Comput. Vis.*, vol. 128, pp. 642–656, Dec. 2020.

[43] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.

[44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[45] Z. Zhang, Y. Liu, B. Li, W. Hu, and H. Peng, "Toward accurate pixel-wise object tracking via attention retrieval," *IEEE Trans. Image Process.*, vol. 30, pp. 8553–8566, 2021.

[46] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6450–6458.

[47] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2018, pp. 7132–7141.

[48] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. IEEE Eur. Conf. Comput. Vis.*, in Lecture Notes in Computer Science, vol. 11211, Sep. 2018, pp. 3–19.

[49] P. Gao, Q. Zhang, F. Wang, L. Xiao, H. Fujita, and Y. Zhang, "Learning reinforced attentional representation for end-to-end visual tracking," *Inf. Sci.*, vol. 517, pp. 52–67, May 2020.

[50] P. Gao, R. Yuan, F. Wang, L. Xiao, H. Fujita, and Y. Zhang, "Siamese attentional keypoint network for high performance visual tracking," *Knowl.-Based Syst.*, vol. 193, Apr. 2020, Art. no. 105448.

[51] Q. Wang, Z. Teng, J. Xing, J. Gao, W. Hu, and S. Maybank, ''Learning attentions: Residual attentional Siamese network for high performance online visual tracking,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4854–4863.

[52] Y. Yu, Y. Xiong, W. Huang, and M. R. Scott, ''Deformable Siamese attention networks for visual object tracking,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6727–6736.

[53] H. Tan, X. Zhang, Z. Zhang, L. Lan, W. Zhang, and Z. Luo, ''Nocal-Siam: Refining visual features and response with advanced non-local blocks for real-time Siamese tracking,'' *IEEE Trans. Image Process.*, vol. 30, pp. 2656–2668, 2021.

[54] Z. Fu, Q. Liu, Z. Fu, and Y. Wang, ''STMTrack: Template-free visual tracking with space-time memory networks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 13774–13783.

[55] X. Li, Q. Liu, N. Fan, Z. He, and H. Wang, ''Hierarchical spatial-aware Siamese network for thermal infrared object tracking,'' *Knowl.-Based Syst.*, vol. 166, pp. 71–81, Feb. 2019.

[56] Q. Liu, X. Li, Z. He, N. Fan, D. Yuan, and H. Wang, ''Learning deep multi-level similarity for thermal infrared object tracking,'' *IEEE Trans. Multimedia*, vol. 23, pp. 2114–2126, 2020.

[57] Q. Liu, D. Yuan, N. Fan, P. Gao, X. Li, and Z. He, ''Learning dual-level deep representation for thermal infrared tracking,'' *IEEE Trans. Multimedia*, early access, Jan. 6, 2022, doi: 10.1109/TMM.2022.3140929.

[58] P. Voigtlaender, J. Luiten, P. H. S. Torr, and B. Leibe, ''Siam R-CNN: visual tracking by re-detection,'' in *Proc. CVPR*, Jun. 2020, pp. 6577–6587.

[59] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, ''Acquisition of localization confidence for accurate object detection,'' in *Proc. IEEE ECCV*, Sep. 2018, pp. 816–832.

[60] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, ''Learning deep features for discriminative localization,'' in *Proc. 29th IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2921–2929.

[61] Z. Wang, J. Xu, L. Liu, F. Zhu, and L. Shao, ''RANet: Ranking attention network for fast video object segmentation,'' in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3977–3986.

[62] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, and Z. Huang, ''ImageNet large scale visual recognition challenge,'' *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[63] E. Real, J. Shlens, S. Mazzocchi, X. Pan, and V. Vanhoucke, ''Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 7464–7473.

[64] Z. Zhang, H. Peng, J. Fu, B. Li, and W. Hu, ''Ocean: Object-aware anchor-free tracking,'' in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2020, pp. 771–787.

[65] S. Cheng, B. Zhong, G. Li, X. Liu, Z. Tang, X. Li, and J. Wang, ''Learning to filter: Siamese relation network for robust tracking,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4421–4431.

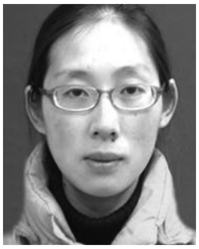
[66] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, ''Fast online object tracking and segmentation: A unifying approach,'' in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1328–1338.

[67] G. Wang, C. Luo, Z. Xiong, and W. Zeng, ''SPM-tracker: Series-parallel matching for real-time visual object tracking,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3643–3652.

[68] J. Peng, Z. Jiang, Y. Gu, Y. Wu, Y. Wang, Y. Tai, C. Wang, and W. Lin, ''SiamRCR: Reciprocal classification and regression for visual object tracking,'' in *Proc. 30th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 952–958.

[69] L. Huang, X. Zhao, and K. Huang, ''GlobalTrack: A simple and strong baseline for long-term tracking,'' in *Proc. AAAI*, Apr. 2020, pp. 11037–11044.

[70] F. Du, P. Liu, W. Zhao, and X. Tang, ''Correlation-guided attention for corner detection based visual tracking,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6835–6844.



ZHAOHUA HU was born in 1981. She received the M.S. and Ph.D. degrees from the Nanjing University of Science and Technology, China. She was an Associate Professor with the Nanjing University of Information Science and Technology, China. Her main research interests include computer vision, deep learning, and visual tracking.

XIAO LIN was born in 1996. He is currently pursuing the master's degree with the Nanjing University of Information Science and Technology, China. His research interests include visual tracking and machine learning.