Learning Deep Multi-Level Similarity for Thermal Infrared Object Tracking

Qiao Liu, Xin Li, Zhenyu He, *Senior Member, IEEE*, Nana Fan, Di Yuan, and Hongpeng Wang

***Abstract*—Existing deep Thermal InfraRed (TIR) trackers only use semantic features to represent the TIR object, which lack the sufficient discriminative capacity for handling distractors. This becomes worse when the feature extraction network is only trained on RGB images. To address this issue, we propose a multi-level similarity model under a Siamese framework for robust TIR object tracking. Specifically, we compute different pattern similarities using the proposed multi-level similarity network. One of them focuses on the global semantic similarity and the other computes the local structural similarity of the TIR object. These two similarities complement each other and hence enhance the discriminative capacity of the network for handling distractors. In addition, we design a simple while effective relative entropy based ensemble subnetwork to integrate the semantic and structural similarities. This subnetwork can adaptive learn the weights of the semantic and structural similarities at the training stage. To further enhance the discriminative capacity of the tracker, we propose a large-scale TIR video sequence dataset for training the proposed model. To the best of our knowledge, this is the first and the largest TIR object tracking training dataset to date. The proposed TIR dataset not only benefits the training for TIR object tracking but also can be applied to numerous TIR visual tasks. Extensive experimental results on three benchmarks demonstrate that the proposed algorithm performs favorably against the state-of-the-art methods.**

***Index Terms*—TIR object tracking, Multi-level similarity, Siamese network, Thermal infrared dataset.**

I. Introduction

THERMAL InfraRed (TIR) object tracking is a fundamental task in computer vision, which receives more and more attention recently. Compared with visual tracking, TIR object tracking has several superiorities, such as illumination insensitivity and privacy protection. Since the TIR object tracking method can track the object in total darkness, it can be used in a wide range of applications, such as video surveillance, maritime rescue, and driver assistance at night [1]. However, there are several problems in TIR object tracking that are still challenging, such as thermal crossover, intensity variation, and distractor [2].

To handle various challenges, numerous TIR trackers are proposed in the past decade. For instance, TBOOST [3] ensembles several MOSSE filters [4] using a continuously switching mechanism to choose a set of right base tracker. TBOOST can adapt the appearance variation of the object since it maintains a dynamics ensemble. Sparse-tir [5] explores the sparse representation with a compressive Harr-like feature for real-time TIR object tracking, which can alleviate the occlusion issue due to the feature of the spare representation. Similar to Sparse-tir, MF-tir [6] also uses the sparse representation method for TIR object tracking but explores multiple complemental features for getting more discriminative features. DSLT [7] uses an online structural support vector machine [8] with a combination of the motion feature and a modified Histogram of Oriented Gradient (HOG) [9] feature for TIR object tracking. DSLT obtains favorable performance mainly because of the dense online learning and the more robust feature representation. There are also a variety of TIR trackers are proposed based on kernel density estimation [10], multiple instances learning [11], low-rank sparse learning [12], discriminative correlation filter [13], [14], etc. Despite much progress, the performance of these trackers is limited by the hand-crafted feature representation.

Recently, inspired by the success of Convolution Neural Network (CNN) in visual tasks [15]–[23], several methods attempt to explore CNN to improve the performance of TIR object tracking. DSST-tir [24] shows that deep features are more effective than hand-crafted features in the Correlation Filter (CF) framework for TIR object tracking. MCFTS [25] uses a pre-trained VGGNet [26] to extract multiple convolutional deep features and then combine them with Kernel Correlation Filter (KCF) [27] to achieve an ensemble TIR tracker. LMSCO [28] integrates deep appearance features [26] and deep motion features [29] into a structural support vector machine [8] for TIR object tracking. HSSNet [30] trains a verification based Siamese CNN on RGB images for TIR object tracking. However, most of these methods only use a deep semantic feature, which is less effective to distinguish intra-class TIR objects. Unlike RGB images, TIR images do not have color information and lack rich texture features. Intra-class TIR objects usually have similar visual and semantic patterns. This indicates that only using a global semantic feature is insufficient for handling distractors in TIR object tracking. Furthermore, most of these deep TIR trackers are trained on

Manuscript received February 21, 2020; revised June 7, 2020; accepted July 4, 2020. Date of publication July 10, 2020; date of current version June 25, 2021. This research was supported in part by the National Natural Science Foundation of China under Grant 61672183 and in part by the Shenzhen Research Council under Grants JCYJ20170413104556946 and JCYJ20170815113552036. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Junwei Han. *(Qiao Liu and Xin Li contributed equally to this work.) (Corresponding authors: Zhenyu He; Hongpeng Wang.)*

The authors are with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518000, China (e-mail: liuqiao.hit@gmail.com; xinlihitsz@gmail.com; zhenyuhe@hit.edu.cn; nanafanhit@gmail.com; 1107449172@qq.com; wanghp@hit.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2020.3008028

RGB images due to lacking a large-scale TIR training dataset, which further degrades the discriminative capacity.

To address the above-mentioned problems, we propose a multi-level similarity model, called MLSSNet, under a Siamese framework for robust TIR object tracking. We note that the multi-level similarity is effective in enhancing the discriminative capacity of the Siamese network for handling distractors. To this end, we design a structural Correlation Similarity Network (CSN) and a semantic CSN to compute different pattern similarities between TIR objects. The structural CSN captures the local structural information of TIR objects and then computes the structural similarity of them. We identify that the structure information can help the network distinguish intra-class TIR objects on the fine-grained level. The semantic CSN enhances the global semantic representation capacity and then computes the similarity on the semantic level. These two similarities complement each other and hence boost the discriminative capacity for handling distractors. To obtain an optimal comprehensive similarity containing the structural and semantic similarities simultaneously, we design a Relative Entropy based adaptive ensemble Network (REN) to integrate them. In addition, to further enhance the discriminative capacity, we construct a large-scale TIR training dataset.[1] with manual annotations for training the proposed model. The dataset has 500 TIR sequences with 20 object classes, more than 228 K frames, and over 289 K bounding boxes. To the best of our knowledge, this dataset is the first and the largest TIR object tracking training dataset to date. We note that the tracker has a more powerful discriminative capacity for handling distractors when it is trained on the proposed TIR dataset. We analyze the multi-level similarity model with an ablation study and compare it with the state-of-the-art methods on the VOT-TIR2015 [31], VOT-TIR2017 [32], and PTB-TIR [2] benchmarks in Section IV-B and Section IV-C respectively. The favorable performance against the state-of-the-art methods demonstrates the effectiveness of the proposed method.

The contributions of the paper are three-fold:

- We propose a deep multi-level similarity model under the Siamese framework for robust TIR object tracking. We design two complementary correlation similarity models, which recognize TIR objects from the local structural level and the global semantic level, respectively. We also design a relative entropy ensemble network that can adaptive to learn an optimal comprehensive similarity.
- We construct the first large-scale TIR training dataset with manual annotations. The proposed dataset not only benefits to TIR object tracking but can also be used to train deep models for other TIR visual tasks.
- We carry out extensive experiments on three benchmarks and demonstrate that the proposed TIR object tracking algorithm performs favorably against the state-of-the-art methods.

The rest of the paper is organized as follows. We first introduce related tracking methods and TIR training datasets briefly in Section II. Then, we describe the architecture and training details of the proposed multi-level similarity network in Section III. Subsequently, the extensive experiments are reported in Section IV to show the proposed method achieves favorable performance. Finally, we draw a short conclusion in Section V.

II. Related Work

In this section, we first introduce the Siamese framework based trackers, which are most related to ours. Then, we discuss several ensemble learning strategies in CNN based tracking method. Finally, we describe several TIR training datasets used for tracking.

Siamese based trackers: Siamese based trackers treat object tracking as a similarity verification task, most of which is to off-line train a similarity metric network and then uses it to online compute the similarity between candidates and the target. For example, Siamese-FC [33] trains the first fully convolutional Siamese network for tracking and achieves promising results. In order to adapt the appearance variation of the target, DSiam [34] learns a dynamic Siamese network by two line regression models. One of these models can learn the target's appearance change and the other can learn to suppress the background. StructSiam [35] learns a structured Siamese network, which focuses on the local pattern of the target and their structural relationship. To obtain more powerful features, SiamFC-tri [36] uses a triplet loss to train the Siamese network, which learns the triplet relationship instead of the pairwise relationship. Quadruplet [37] extends the Siamese network to four branches which learn the potential connection of training samples using a triplet loss and a pair loss simultaneously. SA-Siam [38] exploits a twofold Siamese network which is composed by a semantic branch and an appearance branch. These two branches are trained from different tasks to complement each other. Our method is similar to SA-Siam, which uses two branches, but there are several significant differences. First, SA-Siam uses two separate branches trained with different tasks to compute different similarity, while our model uses two branches trained with one task (the same loss function) to compute different similarity. Second, the two branches of SA-Siam are trained separately, while ours is trained end-to-end. Third, the two branches are fused in the tracking stage via a simple weighted operation in SA-Siam, while ours are fused in the training stage using a relative entropy-based adaptive ensemble network.

In order to enhance the discriminative capacity of the Siamese network, CFNet [39] introduces CF as a differentiable layer into the Siamese network. This layer can update the target branch using video-specific cues that could be helpful for discrimination. CFNet-Hy [40] uses a deep Q-learning method to automatically optimize the hyper-parameter of the tracker. Recently, attention mechanism is widely used in visual task [41], [42] for enhancing representation. For example, RASNet [43] introduces a residual attention to CFNet to further boost the discriminative capacity. HASiam [44] proposes a hierarchical attention module with multi-layers fusion strategy in the Siamese framework for object tracking. LSSiam [45] presents a local semantic Siamese network to extract more robust features for object tracking by using an auxiliary classification branch and a focal

[1] The dataset can be downloaded at [Online]. Available: https://mega.nz/file/80J23A5T#pFYFv_y5NFNVnsJ4zU3a6OH3kPyRwLZebKZV1FjoD-w

logistic loss. Considering the motion information is helpful for tracking, FlowTrack [46] trains an optical flow network and a CFNet model simultaneously. To achieve high performance and high speed simultaneously, SiamRPN [47] employs a Siamese region proposal network which consists of a feature extraction subnetwork and a region proposal subnetwork. It is formulated as a local one-shot detection task in the tracking stage. Subsequently, DaSiamRPN [48] extends SiamRPN by controlling the distribution of the training data and achieves top performance in the VOT2018 [49] challenge. However, most of these methods compute similarity from one single level e.g., the semantic level. Different from these methods, in this paper, we exploit the multi-level similarity to enhance the discriminative capacity of the Siamese network for handling distractors in TIR object tracking.

CNN based ensemble trackers: The ensemble learning is used at different stages in object tracking. For example, HDT [50] combines the multiple weak CNN based CF trackers into a stronger one by a Hedge algorithm. This algorithm can adaptively update the weights of each weak tracker. STCT [51] trains an ensemble based CNN classifier for tracking via a sequential sampling method. Similar to STCT, Branchout [52] trains an ensemble based CNN classifier by using a stochastic regularization technology. TCNN [53] manages the multiple CNNs in a tree structure to estimate target states and to update the model. EDCF [54] integrates a low-level fine-grained feature and a high-level semantic feature in a mutually reinforced way. Though both the proposed REN and the HDT methods use an adaptive ensemble strategy, the proposed REN model is trained end-to-end, which fuses multiple similarities at the learning stage.

TIR training dataset: The lack of a large-scale TIR training dataset hinders the development of CNN in TIR object tracking. Several methods attempt to train a CNN feature model on the TIR dataset for TIR object tracking. For instance, DSST-tir [24] investigates the deep CNN feature in the correlation filter framework for TIR object tracking. This CNN model is trained on a small TIR image dataset (18 K) with the classification task. Its experimental results show that the deep feature based CF tracker can obtain better performance than hand-crafted feature based CF tracker. Zhang *et al.* [55] train a Generative Adversarial Network (GAN) [56] to generate synthetic TIR images from RGB images. These synthetic images, the number of which is over 80 K, are used to train a Siamese network [39] for feature extraction. Then, they combine this deep feature model with the ECO [57] tracker for TIR object tracking. The experimental results show that the synthetic TIR training data significantly improves the performance of TIR object tracking. In this paper, we propose a large-scale real TIR dataset with manual annotations for training the proposed model. To the best of our knowledge, this is the first large-scale real TIR training dataset for object tracking task.

III. Multi-Level Similarity Network

In this section, we first describe the framework of the proposed multi-level similarity network in Section III-A, which mainly consists of three specific designed subnetworks: structural CSN, semantic CSN, and REN. Then, we introduce the proposed TIR training dataset in Section III-B and training details of the network in Section III-C. Finally, we present how to use the proposed network for TIR object tracking in Section III-D.

A. Network Architecture

To achieve more effective TIR object tracking, we construct a multi-level similarity model under the Siamese framework, as shown in Fig. 1. Unlike existing Siamese network which often computes the similarity based on one feature space (e.g., semantic level), we compute the similarity from multiple levels, including the local structure level and the global semantic level. We note that the multi-level similarity improves the discriminative capacity of the Siamese network, and hence improves the robustness of the TIR tracker. To this end, we design two complementary modules, including structural CSN and semantic CSN, which compute the structural similarity and the semantic similarity, respectively. Furthermore, we design a simple while effective adaptive ensemble module, REN, to integrate the structural similarity and semantic similarity. In the following, we highlight these specific networks in detail.

Structural CSN: To compute the structural similarity, we design a structure-aware subnetwork to capture the local structure feature of the object on the low-level convolution layer since the low-level feature contains more local pattern information. We note that the structural similarity is helpful for the accurate location of a tracker. Since TIR objects do not have color information and lack rich texture feature, intra-class TIR objects often have similar visual patterns. Therefore, we argue that the local structure feature is crucial for recognizing them and it is important for a tracker to distinguish distractors. Specifically, we first use two big convolutional kernels to capture the local structure information of the object on the shallow convolution layer. Then, we locate these structure parts by using two corresponding deconvolution layers. Next, we use a Sigmoid layer to generate a two dimension weight map which indicates the importance of every local structure. Finally, we use a scale layer to weight the original feature via the weight map. Given an input low-level convolutional feature map $\mathbf{X}_l \in \mathbb{R}^{H\times W\times C}$, the weighted feature $\omega(\mathbf{X}_l)$ can be formulation as:

$$\boldsymbol{\omega}(\mathbf{X}_l) = \mathbf{X}_l \odot \frac{\exp(\mathbf{W}_l\mathbf{X}_l)}{\exp(\mathbf{W}_l\mathbf{X}_l)+1}, \tag{1}$$

where $\mathbf{W}_l$ denotes the transform matrix which consists of two convolution and two deconvolution layers. The weighted feature is aware of the local structure of the object, as shown in structure-aware feature of Fig. 2. After the scale layer, we add a CF [39] layer to update the target template. Given an input target image $\mathbf{Z}$ and a search image $\mathbf{X}$, the structural similarity can be formulated as:

$$f_{\mathrm{struct}}(\mathbf{Z},\mathbf{X}) = Corr(\boldsymbol{\varphi}(\boldsymbol{\omega}(\boldsymbol{\phi}_{\mathrm{low}}(\mathbf{Z}))), \boldsymbol{\phi}_{\mathrm{low}}(\mathbf{X})), \tag{2}$$

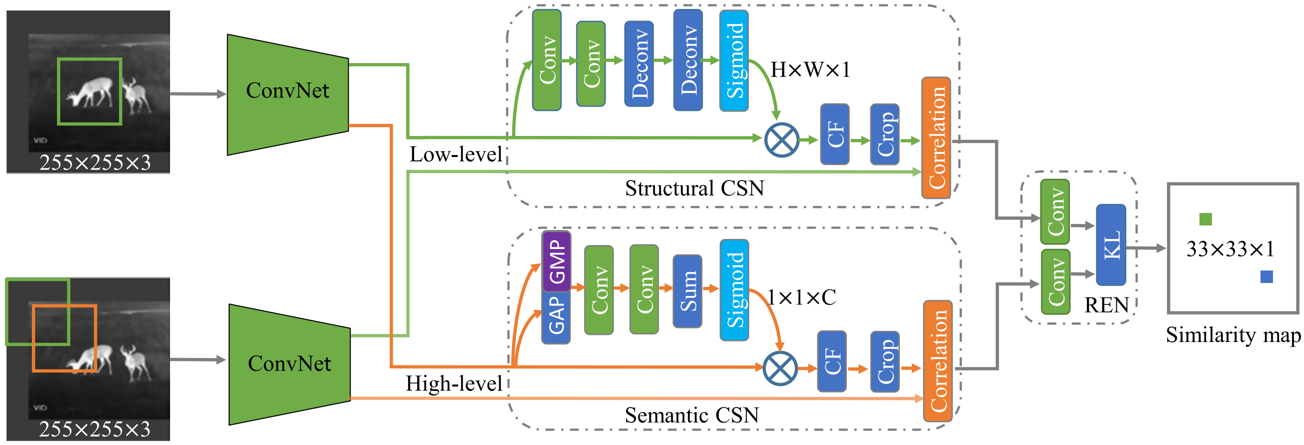


Fig. 1. Architecture of the proposed Multi-Level Similarity based Siamese Network (MLSSNet). MLSSNet is constituted by a shared feature extractor, a structural Correlation Similarity Network (CSN), a semantic CSN, and an adaptive fusion model (REN). Every block denotes a specific network layer and each convolution layer joins a hidden ReLU layer. GAP, GMP, CF, KL, and $\bigotimes$ denote the global average pooling, global max pooling, correlation filter, Kullback-Leibler divergence, and scale layer respectively.

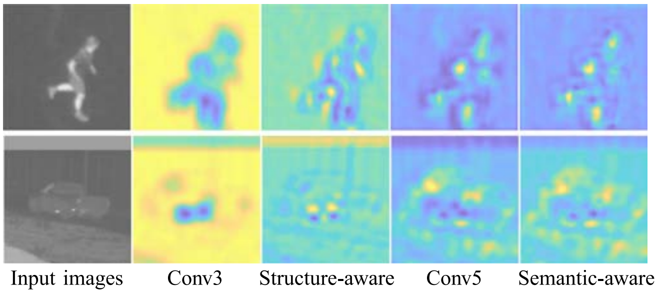


Fig. 2. Visualization of the original and the learned structure-aware and semantic-aware features. The visualized feature maps are generated by summing all channels. From left to right on each column are the input images, the original low-level feature (e.g., Conv3), the learned structure-aware feature form the low-level feature, the original high-level feature (e.g., Conv5), and the learned semantic-aware feature from the high-level feature respectively. We can see that the structure-aware features tend to focus on the local structure parts, e.g., head and leg, while the semantic-aware features emphasize the more discriminative global semantic regions.

where $\phi_{\text{low}}(\cdot)$ denotes the low-level convolutional features of the shared feature extraction network, $\omega(\cdot)$ represents the structure-aware subnetwork formulated by Eq. 1, $\varphi(\cdot)$ is the CF block [39], and $Corr(\cdot,\cdot)$ denotes the cross-correlation operator.

Semantic CSN: To compute the semantic similarity, we design a semantic-aware subnetwork to enhance the semantic representation ability on the high-level convolutional layer since the high-level feature mainly represents a global semantic abstract. Since the discriminative capacity of the network to the inter-class objects mainly comes from the semantic feature, it is important to obtain a more powerful semantic feature. To this end, our semantic-aware subnetwork formulates the relationship of feature channels to generate the more powerful feature, which is similar to SENet [58]. Specifically, we first squeeze the feature map into two one dimension vectors by a global average pooling and a global max pooling respectively. Then, we use two shared full-connected layers to formulate the relationship between these channels and then we fuse the two kinds of relationship vectors via a Sum layer. Different from previous methods, we use two kinds of global pooling because we note that they provide different clues for the global semantic information. Next, we use a Sigmoid layer to generate a one dimension weight vector which indicates the importance of each feature channel. Finally, we employ a scale layer to weight the origin feature via the weight vector. Given an input high-level convolutional feature map $\mathbf{X}_h \in \mathbb{R}^{H\times W\times C}$, the weighted feature $\nu(\mathbf{X}_h)$ can be formulation as:

$$\nu(\mathbf{X}_h) = \mathbf{X}_h \odot \frac{\exp(\mathbf{W}_h \cdot Gap(\mathbf{X}_h) + \mathbf{W}_h \cdot Gmp(\mathbf{X}_h))}{\exp(\mathbf{W}_h \cdot Gap(\mathbf{X}_h) + \mathbf{W}_h \cdot Gmp(\mathbf{X}_h)) + 1}, \tag{3}$$

where $\mathbf{W}_h$ denotes the transform matrix comprised by two shared convolution layers, $Gap(\cdot)$ and $Gmp(\cdot)$ represent the global average pooling and global max pooling layers, respectively. The weighed feature emphasizes the discriminative region and hence obtains more powerful semantic feature representation, as shown in the semantic-aware feature maps of Fig. 2. Similar to the structural similarity, the semantic similarity can be formulated as:

$$f_{\text{semantic}}(\mathbf{Z}, \mathbf{X}) = Corr(\boldsymbol{\varphi}(\boldsymbol{\nu}(\boldsymbol{\phi}_{\text{high}}(\mathbf{Z}))), \boldsymbol{\phi}_{\text{high}}(\mathbf{X})), \tag{4}$$

where $\phi_{\text{high}}(\cdot)$ denotes the high-level convolutional features of the shared feature extraction network, $\nu(\cdot)$ represents the semantic-aware subnetwork formulated by Eq. 3.

REN: To obtain an optimal comprehensive similarity containing the structural similarity and semantic similarity simultaneously, we propose an adaptive ensemble subnetwork which is constituted by two 1×1 convolution layers and a specific designed Kullback-Leibler (KL) divergence layer. The aim of this subnetwork is to obtain a comprehensive similarity map which has a minimum distance from the structural and semantic similarities. Given n similarity maps $S = \{S^1, S^2, \ldots, S^n\}$, we hope to get an optimal integrated similarity map $Q \in \mathbb{R}^{M\times N}$. Each similarity map can be regarded as a probability distribution

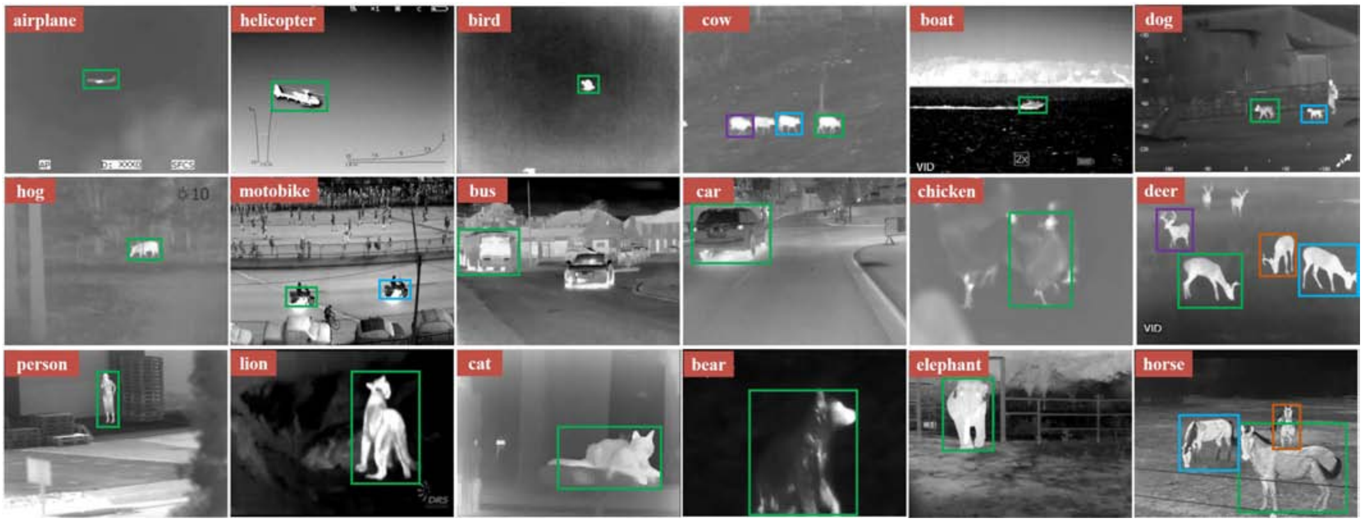


Fig. 3. Examples of the proposed TIR training dataset. We annotate the class name and bounding box of objects in each frame of all sequences. Some sequences have multiple objects, such as the deer, horse, and dog sequences.

of the tracked object. Each element of the similarity map denotes the probability of whether it is the tracked target. We can use KL divergence to measure the distance between the similarity map $S^k (k = 1, 2, \ldots, n)$ and the integrated similarity map Q. Then, we minimize the distance to optimize the similarity map Q by:

$$\arg\min_{Q} \sum_{k=1}^{n} KL(S^k \| Q) \quad \text{s.t.} \sum q_{ij} = 1, \tag{5}$$

where

$$KL\left(S^k \| Q\right) = \sum_{ij} s_{ij}^k \log \frac{s_{ij}^k}{q_{ij}}, \tag{6}$$

s_{ij} and q_{ij} denote the $(i, j)th$ element of the similarity map S and Q respectively. We use the Lagrange multiplier method to solve Eq. 5 and the solution has a simple formulation as:

$$Q = \frac{1}{n} \sum_{k=1}^{n} S^k. \tag{7}$$

Therefore, the KL layer can be regarded as a weighted sum operator. According to Eq. 7, the final comprehensive similarity can be formulated by:

$$f(\mathbf{Z}, \mathbf{X}) = \frac{1}{2}\left(\alpha f_{\text{struct}}(\mathbf{Z}, \mathbf{X}) + \beta f_{\text{semantic}}(\mathbf{Z}, \mathbf{X})\right) + b, \tag{8}$$

where α and β denote the parameter of the two convolution filters respectively. b is the sum of bias of the two convolution layers. These parameters are learned adaptively.

B. TIR Training Dataset

To further enhance the performance of the proposed method, we construct a TIR video training dataset, as shown in Fig. 3, for training the proposed network. The dataset contains 500 TIR image sequences with 20 object classes and more than 228 K frames. We manually annotate the bounding box of objects in each frame of all sequences according to the VID2015 [65] format and generate over 289 K bounding boxes. The source of the dataset comes from existing TIR datasets and Youtube websites, such as RGB-T [64], BU-TIV [61], and OTCBVS [66]. These datasets aim for different tasks, including object tracking, detection, and counting, etc. Since the dataset is collected from different sources and its shot scene and shot time of the videos are also various. Therefore, the dataset has real data distribution and high diversity. Table I compares the proposed TIR training dataset with existing TIR object tracking datasets. From Table I, we can see that the proposed TIR dataset is captured from more than 20 device sources with 20 object classes in various scenarios, which ensures the diversity of the proposed dataset. All of the images of the dataset are shown as the white-hot palette style and stored with an 8 bits depth. Most of the videos are shotted at night, thus, the most object targets are warmer than its background. In addition, we can see that the proposed dataset contains the largest number of videos, frames, and bounding boxes among these compared datasets. To the best of our knowledge, this is the largest and most diverse TIR tracking training dataset to date. We believe this dataset will contribute not only to TIR object tracking but also to other TIR visual tasks, such as image classification, and object detection.

C. Network Training

Training samples: As shown in Fig. 1, the network needs a pair of cropped samples as inputs. In the experiment, we find that using mixed TIR and grayscale training samples can boost the tracking performance of the proposed method. Therefore, we first mix the VID2015 dataset with our TIR dataset using the different proportions. We test several proportions in Ablation study (see Section IV-B). Then, we convert RGB images of VID2015 to the grayscale since the TIR object does not have color information. Finally, we crop the image and choose the positive and

TABLE I
COMPARISON OF THE PROPOSED TIR TRAINING DATASET WITH OTHER TIR OBJECT TRACKING DATASETS

Datasets	Num. of sequences	Max frames	Min frames	Mean frames	Total frames	Frame rates	Object classes	Bit depth	Device sources	Max resolution
OSU [59]	6	2,031	601	1,424	8K	30 fps	1	8	1	320× 240
PDT-ATV [60]	8	775	77	486	4K	20 fps	3	8	1	324× 256
BU-TIV [61]	16	26,760	150	3,750	60K	30 fps	5	16	1	1,024× 512
LTIR [62]	20	1,451	71	563	11K	30 fps	6	8	8	1,920× 480
VOT-TIR2017 [63]	25	1,451	71	555	14K	30 fps	8	8	8	1,920× 480
PTB-TIR [2]	60	1,451	50	502	30K	30 fps	1	8	8+	1,024× 720
RGB-T [64]	234	4,000	45	500	117K	30 fps	6	8	1	640× 480
Ours	**500**	3,056	47	457	**228K**	30 fps	**20**	8	**20+**	**1, 920× 1, 080**

negative training pairs from the whole mixed training dataset like in CFNet [39].

Loss function: We use the logistic loss to train the proposed network. Since the similarity map measures the similarity between a target and multiple candidates, the loss function should be a mean loss:

$$L(y,o) = \frac{1}{|\mathscr{D}|}\sum_{u\in\mathscr{D}} \log(1+\exp(-y[u]o[u])), \tag{9}$$

where $\mathscr{D} \in \mathbb{R}^2$ denotes the similarity map, $o[u]$ represents the real score of a single target-candidate pair and $y[u]$ is the ground-truth of this pair.

D. Tracking Interface

After training of the proposed model, we just use it as a match function at the tracking stage without any online updating. Given a target image $\mathbf{Z}_{t-1}$ at the $(t-1)$-th frame and a search image region $\mathbf{X}_t$ at the t-th frame, the tracked target at the t-th frame can be formulated by:

$$\hat{\mathbf{x}}_{t,i} = \underset{\mathbf{x}_{t,i}}{\arg\max}\, f(\mathbf{Z}_{t-1}, \mathbf{X}_t), \tag{10}$$

where $\mathbf{x}_{t,i} \in \mathbf{X}_t$ is the i-th candidate in the search region $\mathbf{X}_t$. The function $f(.,.)$ denotes the comprehensive similarity defined as E.q 8. To handle the scale variation of the object, we use a simple scale-pyramid mechanism like that in SiamFC [33].

IV. EXPERIMENTS

In this section, we first present the implementation details in Section IV-A. Then, we analyse the effectiveness of each component of the proposed method in Section IV-B. Finally, we compare the proposed algorithm with the state-of-the-art methods in Section IV-C.

A. Experimental Details

We use a modified AlexNet [67] as the shared feature extractor, which all the paddings are removed. Before using the structure CSN, we reduce the channel number of the low-level convolution layer to 64 for accelerating computation. We set the two convolution kernels of the structure-aware module to 5×5 and 7×7 and the corresponding deconvolution kernels to 7×7 and 5×5 respectively. We train the proposed network via Stochastic Gradient Descent (SGD) with the momentum of 0.9 and weight decay of 0.0005 using MatConvNet [68]. The learning rate exponentially decays from 10^{-2} to 10^{-5}. The network is trained for 50 epochs and we set the mini-batch size to 8. In the tracking stage, we set three fixed scales to $\{0.9745, 1, 1.0375\}$ for handling scale variation of the object. The current scale is updated by a linear interpolation with a factor of 0.59 on the predicted scale. The proposed method is carried out on a PC with a GTX 1080 GPU card and achieves an average speed of 18 frames per second (FPS).

B. Ablation Studies

To demonstrate that each component of the proposed network architecture is effective, we first compare the proposed method with its variants on two benchmarks, including VOT-TIR2015 [31], VOT-TIR2017 [32]. Then, we show that which low-level convolution feature is more suitable for TIR object tracking in the proposed framework on the VOT-TIR2017 [32] and PTB-TIR [2] benchmarks. Finally, we validate the effectiveness of the proposed TIR training dataset using several different mixed proportions of the TIR and grayscale training data on the PTB-TIR [2] benchmark.

Datasets: VOT-TIR2015 [31] is a first standard TIR object tracking benchmark which provides the dataset and toolkit to fair evaluate TIR trackers. The dataset contains 20 TIR image sequences and five kinds of challenges, such as Dynamics Change (DC), Occlusion (Occ), Camera Motion (CM), Motion Change (MC), and Size Change (SC). VOT-TIR2017 [32] has 25 TIR image sequences, which is more challenging than VOT-TIR2015. It also has five kinds of challenges which can be used to evaluate the corresponding performance of a tracker. PTB-TIR [2] is a recently proposed TIR object tracking benchmark which focuses on the TIR pedestrian tracking and contains 60 TIR pedestrian sequences. It has nine challenge attributes, such as thermal crossover, distractor, and background clutter, which can be used for attribute-based evaluation.

Evaluation criteria: Accuracy (Acc.) and Robustness (Rob.) are often used to evaluate the performance of a tracker on VOT-TIR2015 and VOT-TIR2017 due to their high interpretability. While accuracy is computed from the overlap rate between

TABLE II
ABLATION STUDIES OF THE NETWORK ARCHITECTURE OF THE PROPOSED METHOD ON TWO BENCHMARKS, INCLUDING VOT-TIR2015, VOT-TIR2017. LOW-LEVEL, STR, HIGH-LEVEL, AND SEM DENOTE THE BASELINE TRACKER ON THE LOW-LEVEL FEATURE (E.G., CONV3), STRUCTURE SIMILARITY MODULE, THE BASELINE TRACKER ON THE HIGH-LEVEL FEATURE (E.G., CONV5), SEMANTIC SIMILARITY MODULE, RESPECTIVELY. THE UP ARROW AND DOWN ARROW DENOTE THE BIGGER OR SMALLER VALUE IS, THE BETTER CORRESPONDING PERFORMANCE HAS

Tracker				VOT-TIR2015 [31]			VOT-TIR2017 [32]		
Low-level	Str	High-level	Sem	EAO ↑	Acc. ↑	Rob. ↓	EAO ↑	Acc. ↑	Rob. ↓
✓				0.288	0.53	2.70	0.265	0.57	3.40
✓	✓			0.310	0.59	2.39	0.270	0.59	3.28
		✓		0.282	0.55	2.82	0.254	0.52	3.45
		✓	✓	0.312	0.60	2.51	0.268	0.55	3.21
✓	✓	✓	✓	**0.326**	0.58	2.53	**0.276**	0.56	3.27

the prediction and ground truth, robustness is measured in term of the frequency of tracking failure. Furthermore, there is a comprehensive evaluation criterion called Expected Average Overlap (EAO) [69] which is adopted to measure the overall performance of a tracker. Different from VOT-TIR2017, PTB-TIR uses Center Location Error (CLE) and Overlap Ratio (OR) as the metrics [70]. Base on these two metrics, Success (Suc.) and Precision (Pre.) are computed to measure the performance of a tracker. Success is defined as that the percentage of the successful frame whose OR is larger than a given threshold. A dynamic threshold [0 1] is often used and the corresponding Area Under Curve (AUC) is used to rank the trackers. Precision denotes that the percentage of the successful frame whose CLE is within a given threshold (*e.g*, 20 pixels).

Network architecture: We use two CFNet [39] using the low-level (e.g., Conv3) and high-level (e.g., Conv5) convolution features respectively as the baseline methods, which trained on VID2015 [65]. First, to demonstrate that the structure-aware is effective, we compare the baseline (low-level) with its variation (low-level+Str) adding the structure-aware module. The first and second rows of Table II show that the structure-aware improves the accuracy and EAO of the baseline by 6% and 2.2% on VOT-TIR2015 respectively. This shows that the structure-aware is helpful for precisely object location.

Second, to show that the semantic-aware is effective, we compare the baseline tracker (high-level) with its variation (high-level+Sem) adding the semantic-aware module. The third and fourth rows of Table II show that the semantic-aware module improves the accuracy of the baseline by 3% and 5% on VOT-TIR2017 and VOT-TIR2015, respectively. We can see that the semantic-aware module also enhances the robustness of the baseline on two benchmarks remarkably. These results demonstrate that the semantic-aware module can enhance the discriminative capacity of the original feature representation. Third, to show that the multi-level similarity can further improve the discriminative capacity, we compare the proposed network with other variations. The last row of Table II shows that compared with the baseline tracker (high-level) using only high-level convolution feature, the multi-level similarity improves EAO by 4.4% and 2.2% on VOT-TIR2015 and VOT-TIR2017 respectively. Compared with the baseline (low-level+Str) using the low-level convolution feature and the structure-aware module, it also enhances EAO by 1.6% and 0.6% on VOT-TIR2015 and VOT-TIR2017 respectively. These results demonstrate that the multi-level similarity can enhance the discriminative capacity of the Siamese network due to the complementarity between the structure similarity and semantic similarity.

TABLE III
COMPARISON OF THE PROPOSED METHOD USING DIFFERENT LOW-LEVEL CONVOLUTION LAYERS ON THE VOT-TIR2017 AND PTB-TIR BENCHMARKS

Low-level	VOT-TIR2017 [32]			PTB-TIR [2]	
	EAO ↑	Acc. ↑	Rob. ↓	Pre.↑	Suc.↑
Conv1	0.269	0.55	3.28	0.699	0.512
Conv2	0.274	0.58	3.31	0.700	0.516
Conv3	0.276	0.56	3.27	0.722	0.516
Conv4	0.283	0.56	3.12	0.697	0.510

Feature selection: Since the proposed structure CSN and semantic CSN are computed from low-level and high-level convolution layers respectively, their results should be different when using different convolution layers. We use the last convolution layer, i.e., conv5, to compute the semantic similarity due to it contains the highest semantic abstract. Meanwhile, we test several different low-level convolution features to compute the structure similarity, the results are shown in Table III. We can see that the results are slightly different on two benchmarks when using different low-level convolution layers. The proposed method achieves the best EAO score (0.283) on VOT-TIR2017 when using the fourth convolution layer, which enhances only 0.7% than the second-best (0.276). While it obtains the best success rate (0.516) on PTB-TIR when using the third convolution layer, which improves only 0.6% than the worst (0.510). These results indicate that the proposed method is insensitive to these low-level features. We suggest that this is mainly because these convolution layers have similar receptive fields and hence provide similar features.

Training data: We find that using the mixed TIR and grayscale training data can boost the performance of the proposed method. Here, we test several different proportions between the TIR training data and the grayscale training data (VID2015) on the PTB-TIR [2] benchmark, as shown in Fig. 4. The results show that using the proportion of 1:1 between the TIR

TABLE IV
COMPARISON OF THE PROPOSED METHOD WITH THE STATE-OF-THE-ART METHODS ON VOT-TIR2017 AND VOT-TIR2015. THE BOLD, ITALIC, AND UNDERLINE DENOTE THE BEST, THE SECOND-BEST, AND THE THIRD-BEST SCORE RESPECTIVELY. THE NOTATION "*" DENOTES THE SPEED IS REPORTED BY THE AUTHORS

Category	Tracker	VOT-TIR2017 [32]			VOT-TIR2015 [31]			Speed
		EAO ↑	Acc. ↑	Rob. ↓	EAO ↑	Acc. ↑	Rob. ↓	FPS
Hand-crafted feature based CF tracker	SRDCF [71]	0.197	0.59	3.84	0.225	0.62	3.06	12.3
	Staple-TIR [63]	0.264	*0.65*	3.31	-	-	-	80.0*
Deep feature based CF tracker	MCFTS [25]	0.193	0.55	4.72	0.218	0.59	4.12	4.7
	HDT [50]	0.196	0.51	4.93	0.188	0.53	5.22	10.6
	deepMKCF [72]	0.213	0.61	3.90	-	-	-	5.0*
	CREST [73]	0.252	0.59	3.26	0.258	0.62	3.11	0.6
	DeepSTRCF [74]	0.262	0.62	3.32	0.257	0.63	2.93	5.5
	ECO-deep [57]	0.267	0.61	2.73	0.286	0.64	2.36	16.3
Other deep tracker	MDNet-N [63]	0.243	0.57	3.33	-	-	-	1.0*
	VITAL [75]	0.272	0.64	2.68	0.289	0.63	**2.18**	4.7
	ATOM [76]	*0.290*	0.61	*2.43*	**0.334**	0.65	2.24	30*
	DiMP [77]	**0.328**	**0.66**	**2.38**	*0.330*	**0.69**	*2.23*	40*
Siamese based deep tracker	Siamese-FC [33]	0.225	0.57	4.29	0.219	0.60	4.10	66.9
	SiamRPN [47]	0.242	0.60	3.19	0.267	0.63	2.53	160.0*
	CFNet [39]	0.254	0.52	3.45	0.282	0.55	2.82	37.0
	DaSiamRPN [48]	0.258	0.62	2.90	0.311	0.67	2.33	160
	HSSNet [30]	0.262	0.58	3.33	0.311	*0.67*	2.53	10.0*
	TADT [78]	0.262	0.60	3.18	0.234	0.61	3.33	42.7
	MLSSNet-N-TIR (Ours)	0.276	0.56	3.27	0.326	0.58	2.53	18.0
	MLSSNet (Ours)	0.286	0.56	3.11	0.329	0.57	2.42	18.0

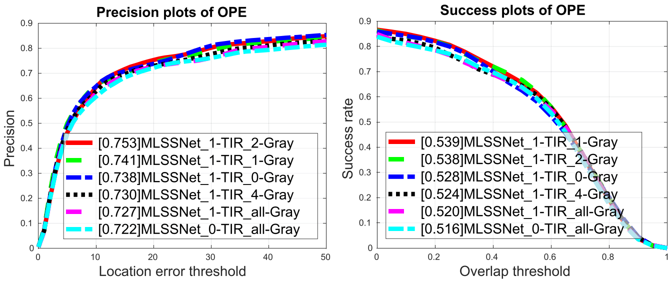


Fig. 4. Comparison of the proposed method using different proportions between the TIR and grayscale training data on the PTB-TIR benchmark. The legend 'MLSSNet_1-TIR_1-Gray' denotes the proposed method using the proportion of 1:1 between the TIR and grayscale training data.

and grayscale training data, the proposed method (MLSSNet_1-TIR_1-Gray) achieves the best success score (0.539), which is higher than the proposed method using only the grayscale training data (MLSSNet_0-TIR_all-Gray) by 2.3% and is higher than the proposed method using only the TIR training data (MLSSNet_1-TIR_0-Gray) by 1.1%. This shows that the TIR image and grayscale image have certain complementary characteristics, which is useful for TIR tracking. Although the proposed TIR training dataset is about 8 times smaller than the grayscale training dataset (VID2015), the proposed method using only the TIR training dataset achieves the higher success score (0.528, ↑ 1.2%) than the proposed method using only the grayscale training dataset. This shows that the proposed TIR training dataset can help the network learn more discriminative features for TIR tracking. In addition, we find that the performance of the proposed method gradually decreases as the proportion of the grayscale training data increases. This shows that the TIR training dataset is crucial for learning discriminative features for TIR tracking.

C. Comparison With the State-of-the-Arts

To evaluate the proposed algorithm comprehensively, we compare our method with the state-of-the-art methods on the VOT-TIR2017 [32], VOT-TIR2015 [31], and PTB-TIR [2] benchmarks.

Compared trackers: We compare the proposed method MLSSNet and its variant (MLSSNet-N-TIR is trained without the proposed TIR dataset) with the state-of-the-art trackers. These methods can be divided into four categories. Seven trackers are based on the deep correlation filter, such as deepMKCF [72], HDT [50], MCFTS [25], CREST [73], ECO-deep [57], UDT [79], and DeepSTRCF [74]. Seven trackers are based on the Siamese framework such as Siamese-FC [33], SiamFC-tri [36], CFNet [39], SiamRPN [47], DaSiamRPN [48], TADT [78], and HSSNet [30]. Two hand-crafted feature based CF trackers: SRDCF [71], Staple-TIR [63]. Four other deep trackers, such as the classification based trackers, MDNet-N [63] and VITAL [75], and the overlap prediction based trackers, ATOM [76] and DiMP [77].

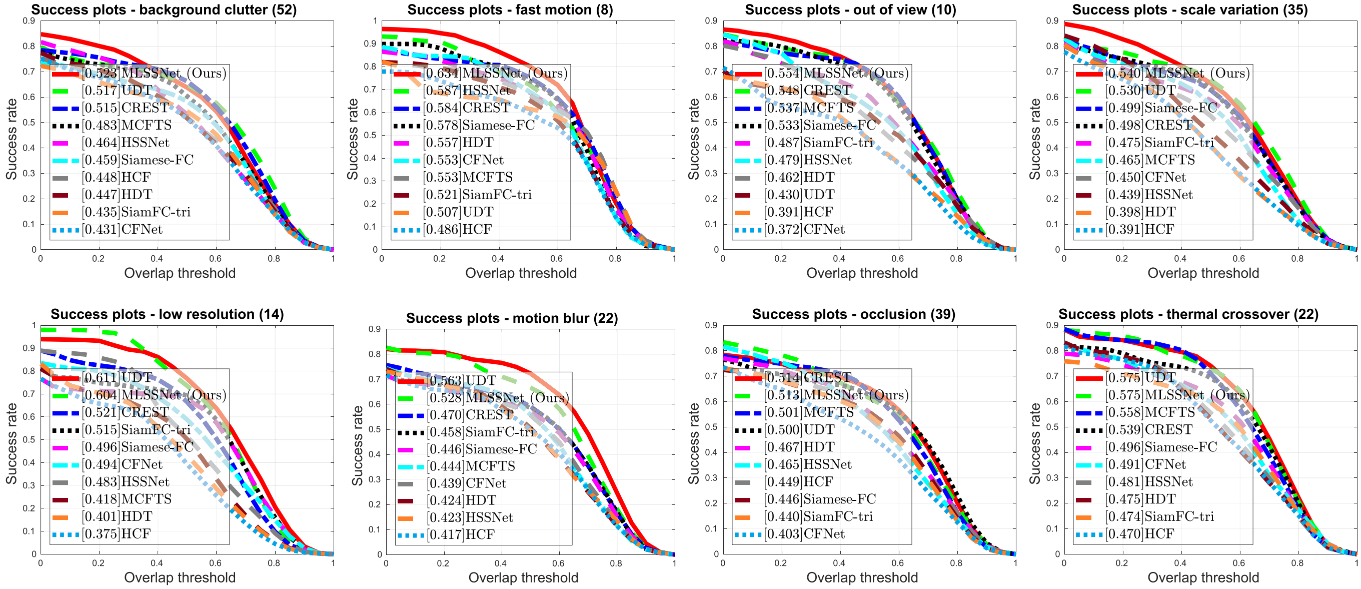


Fig. 5. Comparison of the proposed method with ten deep trackers on eight attribute subsets of the PTB-TIR benchmark.

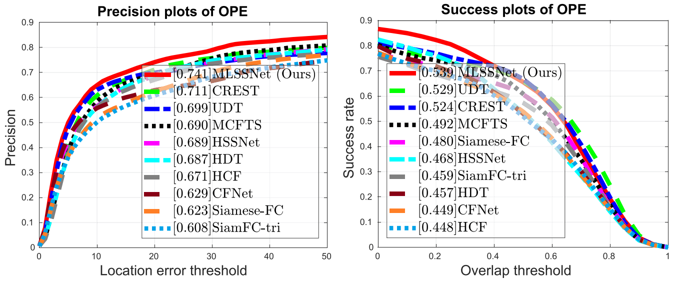


Fig. 6. Comparison of the proposed method with ten deep trackers on the PTB-TIR benchmark.

Overall performance: We compare the comprehensive performance of the proposed algorithm with the state-of-the-art methods on VOT-TIR2015, VOT-TIR2017, and PTB-TIR, as shown in Table IV and Fig. 6. The results show that our algorithm achieves the third-best EAO of 0.286 and 0.329 on the VOT-TIR2017 and VOT-TIR2015 benchmarks, respectively. We can see that although the proposed method does not use the proposed TIR training dataset, it also achieves the competitive EAO 0.276 and 0.326 on both two benchmarks, respectively. These results demonstrate that the proposed method performs favorably against the state-of-the-art methods. Compared with the Siamese based tracker, Siamese-FC [33], the proposed method obtains a 11.0% EAO gain and a 6.1% EAO gain on the VOT-TIR2015 and VOT-TIR2017 benchmarks respectively. Compared with the baseline tracker, CFNet [39], our algorithm achieves a 3.2% EAO score gain and a 4.0% accuracy gain on the VOT-TIR2017 benchmark. It also obtains a 9.0% success score gain on the PTB-TIR benchmark. We attribute the good performance to the proposed multi-level similarity network, which can enhance the discriminative capacity of the network by two complementary similarity modules. Compared with CF based deep tracker CREST [73], our method has better overall performance on three benchmarks despite CREST online updates the target template. We argue that the superior performance of the proposed method comes from the training on the proposed TIR dataset. Though MDNet-N [63] online trains a deep classification network for TIR tracking, our method gets better robustness on VOT-TIR2017, which benefits from both the multi-level similarity and the proposed TIR training dataset.

Attribute-based results: In order to show the effectiveness of the proposed method for handling different challenges, we compare the proposed method with the state-of-the-art methods on the five challenging attributes of the VOT-TIR2017 and VOT-TIR2015 benchmarks, as shown in Table V and Table VI respectively. The results show that our method achieves the best EAO on the dynamics change (0.285) and the camera motion (0.263) challenges of VOT-TIR2017. It also obtains the best EAO, 0.482 and 0.588 on motion change and camera motion challenges of VOT-TIR2015 respectively. Compared with CFNet [39], our method enhances EAO by 6.3% and 13.8% on the dynamic change of VOT-TIR2017 and VOT-TIR2015 respectively. This shows that the proposed multi-level similarity model is more robust to the dynamic change challenge. We can see that our method obtains the competitive performance on the size change challenge of two benchmarks. Compared with Siamese-FC [33], our tracker achieves 6.2% and 12.8% EAO gains on size change of VOT-TIR2017 and VOT-TIR2015 respectively. This demonstrates that the multi-level similarity model improves the robustness of the Siamese network remarkably, since these two trackers use a same scale estimation strategy. We also compare the proposed method with the ten state-of-the-art deep trackers on the eight attribute subsets of the PTB-TIR benchmark, as shown in Fig. 5. The results show that the proposed method performs the best on most attributes, such as background clutter, fast motion, out-of-view, and scale variation. This is consistent with the results on VOT-TIR2017

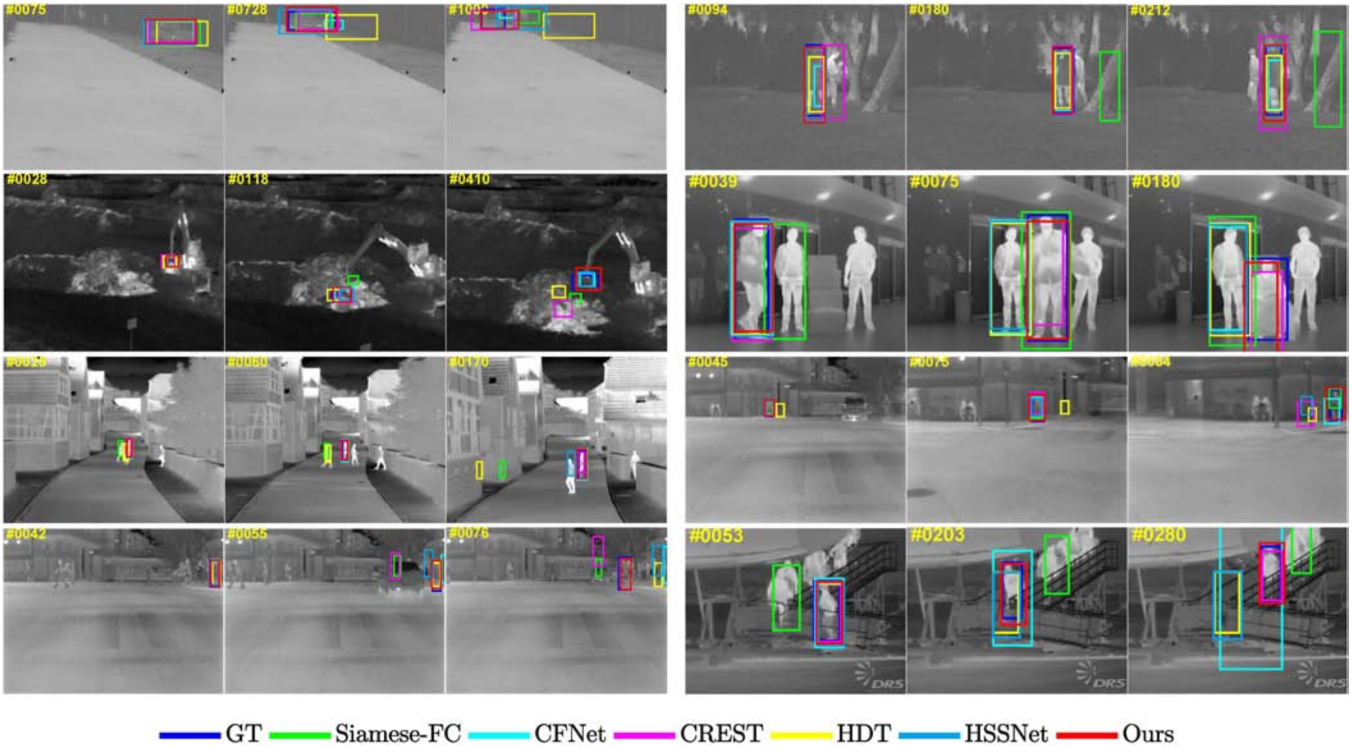


Fig. 7. Tracking results visualized comparison of the proposed method and the state-of-the-art methods on several challenging sequences. From left to right and top to bottom, the sequences are ‘car1,’‘birds,’‘excavator,’‘mixed-distractors,’‘street,’‘sidewalk1,’‘sidewalk2,’‘airplane,’ respectively. The first five sequences are from VOT-TIR2017 and the last three are from PTB-TIR.

TABLE V
COMPARISON OF THE PROPOSED TRACKER AND THE STATE-OF-THE-ART METHODS ON THE FIVE CHALLENGES OF THE VOT-TIR2017 BENCHMARK UNDER EAO EVALUATION CRITERION. THE BOLD AND UNDERLINE DENOTE THE BEST AND THE SECOND-BEST SCORES RESPECTIVELY

	DC	MC	CM	SC	Occ
MCFTS [25]	0.072	0.278	0.156	0.212	0.181
HDT [50]	0.090	0.258	0.167	0.243	0.181
deepMKCF [72]	0.074	0.319	0.179	0.255	0.189
Siamese-FC [33]	0.188	0.319	0.196	0.277	0.222
SiamRPN [47]	0.167	0.339	0.215	0.303	0.226
MDNet-N [63]	0.209	0.381	0.216	0.290	0.280
CREST [73]	0.130	0.348	0.256	0.300	0.278
CFNet [39]	0.222	0.410	0.219	0.285	0.306
DaSiamRPN [48]	0.114	0.336	0.208	0.309	0.214
HSSNet [30]	0.204	0.430	0.204	0.309	**0.317**
DeepSTRCF [74]	0.217	0.359	0.233	**0.370**	0.268
Staple-TIR [63]	0.164	0.414	0.186	0.342	0.258
ECO-deep [57]	0.192	0.387	0.233	0.344	0.280
VITAL [75]	0.157	**0.440**	0.254	0.299	0.253
MLSSNet-N-TIR	0.256	0.436	0.233	0.315	0.281
MLSSNet (Ours)	**0.285**	0.424	**0.263**	0.339	0.285

TABLE VI
COMPARISON OF THE PROPOSED TRACKER AND THE STATE-OF-THE-ART METHODS ON THE FIVE CHALLENGES OF THE VOT-TIR2015 BENCHMARK UNDER EAO EVALUATION CRITERION

	DC	MC	CM	SC	Occ
MCFTS [25]	0.707	0.257	0.362	0.214	0.483
HDT [50]	0.689	0.224	0.215	0.187	0.463
Siamese-FC [33]	0.671	0.319	0.226	0.273	0.406
SiamRPN [47]	0.725	0.316	0.372	0.352	0.404
CREST [73]	0.708	0.331	0.475	0.278	**0.652**
CFNet [39]	0.590	0.387	0.459	0.326	0.495
DaSiamRPN [48]	0.718	0.369	0.370	**0.443**	0.457
HSSNet [30]	0.661	0.426	0.407	0.383	0.496
DeepSTRCF [74]	0.693	0.337	0.277	0.354	0.492
ECO-deep [57]	**0.745**	0.371	0.335	0.417	0.614
VITAL [75]	0.435	0.392	0.561	0.358	0.526
MLSSNet-N-TIR	0.732	0.447	0.502	0.436	0.554
MLSSNet (Ours)	0.728	**0.482**	**0.588**	0.401	0.630

and VOT-TIR2015. We can also see that the proposed method performs well on the other attributes, such as thermal crossover, occlusion. It is easy to integrate an independent re-detection strategy [80], [81] using the confidence of tracked object [82] for further improving the robustness of the tracker to handle these challenges. All of these attributes based results demonstrate that our algorithm achieves a powerful discriminative capacity and favorable performance.

Visualized results: To show the tracking performance more intuitionally, we compare the visualized tracking results of the proposed algorithm with several state-of-the-art trackers on eight challenging sequences, as shown in Fig. 7. The results show that the proposed method tracks the objects more accurate and robust in the most challenges. Especially, when two similar objects cross each other, such as ‘mixed-distractors,’ ‘street,’ and ‘airplane,’ most trackers drift to distractors, while our algorithm locates the object accurately. We attribute the good performance to the proposed multi-level similarity network, which can recognize the intra-class objects from their subtle differences. In addition, we can see that the proposed method also performs better than most trackers when the background is clutter, such as ‘excavator,’‘sidewalk1,’ and ‘sidewalk2’. This shows that the proposed method achieves favorably discriminative capacity.

V. Conclusion

This paper proposes a multi-level similarity model under the Siamese framework for robust Thermal InfraRed (TIR) object tracking. The network consists of a multi-level similarity network and a relative entropy based adaptive ensemble network. The structural correlation similarity network captures the local structure information of the TIR object for the precise location. While the semantic correlation similarity network enhances the global semantic representation of the feature for robust identification. The multi-level similarity improves the discriminative capacity of the Siamese network. In addition, to further enhance the discriminative capacity, we construct a large-scale TIR image dataset to train the proposed model. The dataset not only benefits the training for TIR object tracking but also can be applied to numerous TIR visual tasks such as classification and detection. Extensive experimental results on three benchmarks show that the proposed method performs favorably against the state-of-the-art methods.

References

[1] R. Gade and T. B. Moeslund, “Thermal cameras and applications: A survey,” *Mach. Vision Appl.*, vol. 25, no. 1, pp. 245–262, 2014.

[2] Q. Liu, Z. He, X. Li, and Y. Zhen, “PTB-TIR: A thermal infrared pedestrian tracking benchmark,” *IEEE Trans. Multimedia*, vol. 22, no. 3, pp. 666–675, Mar. 2020.

[3] E. Gundogdu *et al.*, “Comparison of infrared and visible imagery for object tracking: Toward trackers with superior ir performance,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshop*, 2015, pp. 1–9.

[4] D. S. 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. Vision Pattern Recognit.*, 2010, pp. 2544–2550.

[5] Y. Li, P. Li, and Q. Shen, “Real-time infrared target tracking based on ℓ_1 minimization and compressive features,” *Appl. Opt.*, vol. 53, no. 28, pp. 6518–6526, 2014.

[6] S. J. Gao and S. T. Jhang, “Infrared target tracking using multi-feature joint sparse representation,” in *Proc. Int. Conf. Res. Adaptive Convergent Syst.*, 2016, pp. 40–45.

[7] X. Yu, Q. Yu, Y. Shang, and H. Zhang, “Dense structural learning for infrared object tracking at 200+ frames per second,” *Pattern Recognit. Lett.*, vol. 100, pp. 152–159, 2017.

[8] S. Hare *et al.*, “Struck: Structured output tracking with kernels,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2096–2109, Oct. 2016.

[9] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, 2005, vol. 1, pp. 886–893.

[10] R. Liu and Y. Lu, “Infrared target tracking in multiple feature pseudo-color image with kernel density estimation,” *Infrared Phys. Technol.*, vol. 55, no. 6, pp. 505–512, 2012.

[11] X. Shi, W. Hu, Y. Cheng, G. Chen, and J. J. H. Ling, “Infrared target tracking using multiple instance learning with adaptive motion prediction and spatially template weighting,” in *Proc. Sensors Syst. Space Appl. VI*, 2013, vol. 8739, Art. no. 873912.

[12] Y. He, M. Li, J. Zhang, and J. Yao, “Infrared target tracking based on robust low-rank sparse learning,” *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 232–236, Feb. 2016.

[13] Y.-J. He, M. Li, J. Zhang, and J.-P. Yao, “Infrared target tracking via weighted correlation filter,” *Infrared Phys. Technol.*, vol. 73, pp. 103–114, 2015.

[14] C. Asha and A. Narasimhadhan, “Robust infrared target tracking using discriminative and generative approaches,” *Infrared Phys. Technol.*, vol. 85, pp. 114–127, 2017.

[15] L. Wang, W. Ouyang, X. Wang, and H. Lu, “Visual tracking with fully convolutional networks,” in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 3119–3127.

[16] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, “Hierarchical convolutional features for visual tracking,” in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 3074–3082.

[17] C. Tian, Y. Xu, and W. Zuo, “Image denoising using deep cnn with batch renormalization,” *Neural Netw.*, vol. 121, pp. 461–473, 2020.

[18] C. Tian *et al.*, “Attention-guided cnn for image denoising,” *Neural Netw.*, vol. 124, pp. 117–129, 2020.

[19] K. Chen and W. Tao, “Learning linear regression via single-convolutional layer for visual object tracking,” *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 86–97, Jan. 2019.

[20] Q. Wang, C. Yuan, J. Wang, and W. Zeng, “Learning attentional recurrent neural network for visual tracking,” *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 930–942, Apr. 2019.

[21] J. Wen *et al.*, “Robust sparse linear discriminant analysis,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 390–403, Feb. 2019.

[22] F. Zheng, L. Shao, and J. Han, “Robust and long-term object tracking with an application to vehicles,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 10, pp. 3387–3399, Oct. 2018.

[23] H. Hu *et al.*, “Robust object tracking using manifold regularized convolutional neural networks,” *IEEE Trans. Multimedia*, vol. 21, no. 2, pp. 510–521, Feb. 2019.

[24] E. Gundogdu, A. Koc, B. Solmaz, R. I. Hammoud, and A. A. Alatan, “Evaluation of feature channels for correlation-filter-based visual object tracking in infrared spectrum,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2016, pp. 24–32.

[25] Q. Liu *et al.*, “Deep convolutional neural networks for thermal infrared object tracking,” *Knowl.-Based Syst.*, vol. 134, pp. 189–198, 2017.

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

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

[28] P. Gao *et al.*, “Large margin structured convolution operator for thermal infrared object tracking,” in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2018, pp. 2380–2385.

[29] G. Gkioxari and J. Malik, “Finding action tubes,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 759–768.

[30] X. Li *et al.*, “Hierarchical spatial-aware siamese network for thermal infrared object tracking,” *Knowl.-Based Syst.*, vol. 166, pp. 71–81, 2019.

[31] M. Felsberg *et al.*, “The thermal infrared visual object tracking vot-tir2015 challenge results,” in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, 2015, pp. 76–88.

[32] M. Kristan *et al.*, “The visual object tracking vot2017 challenge results,” in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, 2017, pp. 1949–1972.

[33] L. Bertinetto *et al.*, “Fully-convolutional siamese networks for object tracking,” in *Proc. Eur. Conf. Comput. Vision Workshops*, 2016, pp. 850–865.

[34] Q. Guo *et al.*, “Learning dynamic siamese network for visual object tracking,” in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 1763–1771.

[35] Y. Zhang *et al.*, “Structured siamese network for real-time visual tracking,” in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 351–366.

[36] X. Dong and J. Shen, “Triplet loss in siamese network for object tracking,” in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 459–474.

[37] X. Dong *et al.*, “Quadruplet network with one-shot learning for fast visual object tracking,” *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3516–3527, Jul. 2019.

[38] A. He, C. Luo, X. Tian, and W. Zeng, “A twofold siamese network for real-time object tracking,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 4834–4843.

[39] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 5000–5008.

[40] X. Dong *et al.*, "Dynamical hyperparameter optimization via deep reinforcement learning in tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2019.2956703.

[41] W. Wang, X. Lu, J. Shen, D. J. Crandall, and L. Shao, "Zero-shot video object segmentation via attentive graph neural networks," in *Proc. IEEE Int. Conf. Comput. Vision*, 2019, pp. 9236–9245.

[42] W. Wang, S. Zhao, J. Shen, S. C. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 1448–1457.

[43] Q. Wang *et al.*, "Learning attentions: Residual attentional siamese network for high performance online visual tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 4854–4863.

[44] J. Shen, X. Tang, X. Dong, and L. Shao, "Visual object tracking by hierarchical attention siamese network," *IEEE Trans. Cybern.*, vol. 50, no. 7, pp. 3068–3080, Jul. 2020.

[45] Z. Liang and J. Shen, "Local semantic siamese networks for fast tracking," *IEEE Trans. Image Process.*, vol. 29, pp. 3351–3364, 2020.

[46] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 548–557.

[47] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 8971–8980.

[48] Z. Zhu *et al.*, "Distractor-aware siamese networks for visual object tracking," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 103–119.

[49] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vision Workshops*, 2018, pp. 1–23.

[50] Y. Qi *et al.*, "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 4303–4311.

[51] L. Wang, W. Ouyang, X. Wang, and H. Lu, "STCT: Sequentially training convolutional networks for visual tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 1373–1381.

[52] B. Han, J. Sim, and H. Adam, "Branchout: Regularization for online ensemble tracking with convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 2217–2224.

[53] H. Nam, M. Baek, and B. Han, "Modeling and propagating CNNs in a tree structure for visual tracking," 2016, *arXiv:1608.07242*.

[54] Q. Wang *et al.*, "Do not lose the details: Reinforced representation learning for high performance visual tracking," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 985–991.

[55] L. Zhang, A. Gonzalez-Garcia, J. van de Weijer, M. Danelljan, and F. S. Khan, "Synthetic data generation for end-to-end thermal infrared tracking," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1837–1850, Apr. 2019.

[56] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 1125–1134.

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

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

[59] J. W. Davis and V. Sharma, "Background-subtraction using contour-based fusion of thermal and visible imagery," *Comput. Vision Image Understanding*, vol. 106, no. 2, pp. 162–182, 2007.

[60] J. Portmann, S. Lynen, M. Chli, and R. Siegwart, "People detection and tracking from aerial thermal views," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 1794–1800.

[61] Z. Wu, N. Fuller, D. Theriault, and M. Betke, "A thermal infrared video benchmark for visual analysis," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, 2014, pp. 201–208.

[62] A. Berg, J. Ahlberg, and M. Felsberg, "A thermal object tracking benchmark," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveillance*, 2015, pp. 1–6.

[63] M. Felsberg *et al.*, "The thermal infrared visual object tracking VOT-TIR2016 challenge results," in *Proc. Eur. Conf. Comput. Vision Workshops*, 2016, pp. 824–849.

[64] C. Li *et al.*, "RGB-T object tracking: Benchmark and baseline," *Pattern Recognit.*, vol. 96, 2019, Art. no. 106977.

[65] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Conmput. Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[66] R. Miezianko, "Ieee otcbvs ws series benchmark," [Online]. Available: http://vcipl-okstate.org/pbvs/bench/ Accessed: Mar. 4, 2018.

[67] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[68] A. Vedaldi and K. Lenc, "MatconvNet: Convolutional neural networks for matlab," in *Proc. ACM Int. Conf. Multimedia*, 2015, pp. 689–692.

[69] M. Kristan *et al.*, "The visual object tracking VOT2015 challenge results," in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, 2015, pp. 1–23.

[70] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 2411–2418.

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

[72] M. Tang and J. Feng, "Multi-kernel correlation filter for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, 2015, pp. 3038–3046.

[73] Y. Song *et al.*, "CREST: Convolutional residual learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, 2017, pp. 2574–2583.

[74] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 4904–4913.

[75] Y. Song *et al.*, "VITAL: Visual tracking via adversarial learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 8990–8999.

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

[77] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, 2019, pp. 6182–6191.

[78] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 1369–1378.

[79] N. Wang *et al.*, "Unsupervised deep tracking," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 1308–1317.

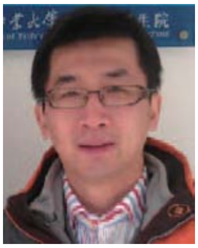
[80] X. Dong *et al.*, "Occlusion-aware real-time object tracking," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 763–771, Apr. 2017.

[81] J. Shen, D. Yu, L. Deng, and X. Dong, "Fast online tracking with detection refinement," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 162–173, Jan. 2018.

[82] B. Yan, H. Zhao, D. Wang, H. Lu, and X. Yang, "'skimming-perusal'tracking: A framework for real-time and robust long-term tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, 2019, pp. 2385–2393.

Qiao Liu received the master's degree in computer science from Guizhou Normal University, Guiyang, China, in 2016. He is currently working toward the Ph.D. degree with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His current research interests include thermal infrared object tracking, and machine learning.

Xin Li received the B.E. degree in 2014 from Harbin Institute of Technology, Shenzhen, China, where he is currently working toward the Ph.D. degree with the Department of Computer Science. His research interests lie primarily in visual tracking, image processing, and machine learning.



Zhenyu He (Senior Member, IEEE) received the Ph.D. degree from the Department of Computer Science, Hong Kong Baptist University, Hong Kong, in 2007. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His research interests include sparse representation and its applications, deep learning and its applications, pattern recognition, image processing, and computer vision.

Nana Fan received the master's degree, in 2016 from the Harbin Institute of Technology, Shenzhen, China, where she is currently working toward the Ph.D degree with the Department of Computer Science. Her research interests include visual tracking, deep learning, and machine learning.

Di Yuan received the master degree's from the Harbin Institute of Technology, Shenzhen, China, where he is currently working toward the Ph.D. degree with the Department of Computer Science and Technology. His current research interests include visual object tracking, and person re-identification.

Hongpeng Wang received the B.Eng., M. Eng., and Ph.D. degrees from the Harbin Institute of Technology, Shenzhen, China, where he is currently a Full Professor and Ph.D. Supervisor with the School of Computer Science and Technology. His research interests include intelligent robot, computer vision, artificial intelligence, and bioinformatics.