Robust RGB-T Tracking via Graph Attention-Based Bilinear Pooling

Bin Kang, Dong Liang, Junxi Mei, Xiaoyang Tan, Quan Zhou, Member, IEEE, and Dengyin Zhang, Member, IEEE

Abstract— RGB-T tracker possesses strong capability of fusing two different yet complementary target observations, thus providing a promising solution to fulfill all-weather tracking in intelligent transportation systems. Existing convolutional neural network (CNN)-based RGB-T tracking methods often consider the multisource-oriented deep feature fusion from global viewpoint, but fail to yield satisfactory performance when the target pair only contains partially useful information. To solve this problem, we propose a four-stream oriented Siamese network (FS-Siamese) for RGB-T tracking. The key innovation of our network structure lies in that we formulate multidomain multilayer feature map fusion as a multiple graph learning problem, based on which we develop a graph attention-based bilinear pooling module to explore the partial feature interaction between the RGB and the thermal targets. This can effectively avoid uninformed image blocks disturbing feature embedding fusion. To enhance the efficiency of the proposed Siamese network structure, we propose to adopt meta-learning to incorporate category information in the updating of bilinear pooling results, which can online enforce the exemplar and current target appearance obtaining similar semantic representation. Extensive experiments on grayscale-thermal object tracking (GTOT) and RGBT234 datasets demonstrate that the proposed method outperforms the state-of-the-art methods for the task of RGB-T tracking.

Index Terms—Bilinear pooling, meta-learning, RGB-T tracking, Siamese network.

I. INTRODUCTION

WITH the flourishment of multimedia, thermal infrared camera has become economically affordable. Such camera can capture the thermal infrared radiation emitted by the targets with temperature above absolute zero, and hence is suitable for night surveillance. For this reason, two advantages have been identified in jointly using RGB and thermal infrared cameras.

1) Thermal infrared camera is skilled in resisting illumination change, which can offer strong support to RGB camera under poor light condition. 2) RGB camera would help solve the crossover challenge suffered in thermal infrared camera-based surveillance. Therefore, RGB-T tracking with both RGB and thermal features can effectively tackle the bad weather challenge [1].

In RGB-T tracking, the RGB and thermal video sequences are obtained in pairs (see Fig. 1). The key idea is to exploit the complementarity of the RGB and thermal information for efficient multimodel fusion. To this end, many state-of-the-art methods have been developed over the past decade, which can be briefly categorized into three classes. The first class is the particle fusion-based RGB-T tracker, which requires effective representation of the appearance variation of the RGB and thermal targets for the estimation of particle fusion weights [2], [3]. The second one is to build multiple graph fusing model to effectively exploit the spatial relation between the RGB and the thermal target blocks [4], [5]. The third-class benefits from sparse representation, where the sparse codes and the correlation between two sparse representation models can be simultaneously estimated through solving the unified optimization problem [6]–[8]. All of aforementioned methods use handcraft feature for multimodel fusion. Compared with handcraft feature, deep convolutional neural networks (CNN) can extract the translation and light invariant deep semantic information for robust representation of the target. Thus, deep learning technology has appeared to have great potential in RGB-T tracking recently. For example, Zhu et al. [9]...
proposed a dense CNN for RGB-T tracking, which can recursively aggregate informative features of two kinds of convolutional paths (RGB image-oriented convolutional path and thermal image-oriented convolutional path). In [10], a multiadapter convolutional network is proposed to simultaneously explore the complementarity property and achieve instance-aware feature learning in an end-to-end manner.

Existing CNN-based RGB-T trackers often consider the multilayer convolutional feature maps as the hierarchically holistic feature, ignoring the partial feature interaction between the RGB and thermal targets. This may obviously reduce tracking accuracy in challenging video pairs such as in Fig. 1, where only a little useful information can be extracted from RGB or thermal video sequences for representing pairwise targets. What’s worse, the little useful information on the RGB or thermal targets may be partially matched or even unmatched in spatial domain. In this case, simply treating RGB and thermal targets. This can result in inevitable negative effect. Part-feature-based RGB-T trackers such as [4], [5] can achieve partial information fusion according to the importance of different image blocks. However, those methods pay attention to the handcraft feature only, which could not be easily extended to multiple convolutional network fusion.

In this article, we propose a simple yet efficient four-stream oriented Siamese network (FS-Siamese) for RGB-T tracking as shown in Fig. 2, where the feature embedding of four streams can be divided into exemplar embedding pair and candidate embedding pair. Two embedding pairs can be fused, respectively, through the graph attention-based bilinear pooling module for generating the reinforced exemplar and reinforced candidate, which are used to produce the subsequent similarity map. Bilinear pooling has shown superior performance over traditional linear fusion strategy on the fusion of heterogeneous partial information in fine-grained recognition [11] and visual question answering [12]. Although bilinear pooling has won a certain performance gain, it could not discriminate the importance of the elements in the deep feature maps. This may give rise to unavoidable negative effect when facing challenging scenarios such as in Fig. 1. In view of these observations, we introduce coattention mechanism in the bilinear pooling to formulate multimodel pooling as a multiple graph learning problem. Based on the new problem formulation, we develop a graph attention-based bilinear pooling module to integrate two tasks, namely the exploration of partial feature interaction and the fusion of multisource oriented feature embeddings, into a unified end-to-end network structure.

Since the target appearance may dramatically change, it is necessary to introduce an effective strategy in updating the graph attention-based bilinear pooling module. The state-of-the-art updating strategies such as [13]–[15] only focus on exploring the temporal correlation between the current and previous target appearance, while ignoring a fact that online exploring the spatial relation between the target and its surrounding background is very important for locating the most similar candidate pairs. Considering this issue, we design a meta-learning-based updating strategy to effectively update the fully connected layer of the graph attention-based bilinear pooling module. This paves a way on utilizing category information to online update semantic representation of exemplar. The main contributions of this article are listed as follows.

1) We formulate the attention-based bilinear pooling as a multiple graph learning problem, based on which we integrate the graph attention network and outer product into a unified structure to highlight the discriminative local information in RGB and thermal targets. This can effectively eliminate the disturbance in target pair fusion.

2) Traditional multiple stream-oriented tracking networks only fuse the target regression results of different streams, without exploring the pairwise relation during the fusion of target embeddings. To overcome this limitation, we propose a four-stream oriented network structure using graph attention-based bilinear pooling for the effective fusion of multisource embedding pairs.
3) We adopt meta-learning to update the graph attention-based bilinear pooling, and thereby utilize the category information to online restrict the exemplar to learn a similar semantic representation as current tracking result, which is helpful for discriminating the reinforced exemplar and reinforced candidates.

4) Extensive experiments on grayscale-thermal object tracking (GTOT), RGBT234, CUB-200-2011, fine-grained visual classification (FGVC)-aircraft, and Cars datasets show that our graph attention-based bilinear pooling module not only can effectively fuse multimain multilayer feature maps in RGB-T tracking, but also can be extended to other multimodel fusion tasks.

II. RELATED WORKS

A. Siamese Network in RGB Tracking

Siamese network has popular in RGB camera-based visual tracking due to its simple network structure and fast tracking speed. In Siamese network-based RGB tracking, Bertinetto et al. [16] is the pioneer who designs the Siamese network structure, where the tracking result is obtained by orderly calculating the similarity between the exemplar embedding and each candidate embeddings within the search region. The cross correlation is often used as the similarity measure. Following Bertinetto’s work, the following studies have emerged which can be briefly divided into three scenarios.

1) The attention-based Siamese networks (e.g., [13], [17]) that effectively use the gradient of backward propagation and the channel attention mechanism to make the target appearance embedding concentrate on the informative subregion.

2) The local pattern-based Siamese Networks (e.g., [18]–[20]) that can explore the spatial relation between different target blocks.

3) The region proposal network (RPN)-based Siamese networks (e.g., [21]–[23]) that introduce RPN in Siamese network to avoid the time-consuming multiscale estimation step.

Unfortunately, all of aforementioned works cannot be easily extended to RGB-T tracking because of the following two main challenges.

1) The state-of-the-art RGB trackers have explored relation among different target blocks and introduced attention mechanism in the Siamese network, however those works are carried out in RGB domain. It is seen in Fig. 1 that there is a big gap between RGB and thermal images. Thus it not only requires the RGB-T tracker to explore the spatial correlation between target blocks in a single image domain, but also requires to overcome image gap challenge to effectively locate informative target blocks for exploring the partial feature interaction between two image domains.

2) We should not sacrifice the simple Siamese structure for exploring the partial feature interaction between RGB and thermal targets. Hence it requires to find a tradeoff between the complexity of the multimodel fusion and the efficiency of the network structure.

Similar to our network structure, the work in [24] introduced Siamese network in RGB-T tracking. However it still adopts linear fusion strategy to make RGB and thermal target feature complement with each other, ignoring theoretically study how the effective extract the common and the specific information between two image domains for unlinear fusion. Although the RGB tracker in [25] also has four embedding paths, it only fuses the tracking results of different streams, while ignoring the pairwise relation during feature embedding.

B. Bilinear Pooling

After the work in [26] that used multimodal compact bilinear pooling to explore the pairwise relation between two heterogeneous models, bilinear pooling has become an effective tool in visual question answering (VQA). Since the dimension of the output bilinear vector in [26] is high, Kim et al. [27] proposed a low-rank bilinear pooling to use two online estimated projection to project bilinear vector into a low-rank subspace, in which the redundant information in bilinear vector can be obviously reduced. Besides VQA, bilinear pooling has also been widely used in face recognition and fine-grained recognition, e.g., Chang et al. [28] proposed a Compound Rank-k Projections (CRP) algorithm for bilinear analysis, where the 2-D handcraft feature-based discriminant projections can be simultaneously learned in a collaborative way. Lin et al. [11] proposed a bilinear CNN model to use outer product to effectively fuse the pairwise fine-grained target information between two kinds of CNN networks. Wei et al. [29] used bilinear pooling to explore the partial feature interaction between two fine-grained models.

Unlike aforementioned works, we introduce graph attention CNN in bilinear pooling to simultaneously locate the informative target blocks in RGB-thermal image pairs. Although Gao et al. [30] has also used graph CNN in visual tracking, this work use predefined affinity matrix to build the graph. Our work is more challenging than the work in [30] because it requires to adaptively estimate multiple affinity matrices without any prior knowledge.

C. One Shot Learning

One shot learning aims to study the ability of using a single class example to recognize novel categories. The representative works include [31]–[33], and so on. Since meta-learning methods own the capability of learning to learn, it has become a useful tool for achieving one shot learning. For example, Gidaris and Komodakis [34] used meta-learning to learn the mapping function between classification weight and semantic feature vectors for one shot recognition. Li et al. [21] introduced meta-learning-based one-shot detection module in Siamese network for adaptive scale estimation. Similar to our work, Dong et al. [35] also introduced one-shot learning-based classification in visual tracking. However, this work could not realize real-time updating because it adopted triplet loss that may not guarantee a fast convergence. In contrast, we design a meta-learning strategy to rapidly learn optimal classification parameters without much experimental setting involved.

III. PROPOSED APPROACH

A. Overview

The network structure of our four-stream oriented Siamese network is shown in Fig. 2, where the network contains
four embedding streams. Two streams are used for embedding the target exemplar (target template) pair \(Z_1\) and \(Z_2\). And the other two streams are used for embedding the candidate pair \((x^1_i, x^2_i)\) within the search regions. After feature embedding, the exemplar embedding pair and the \(i\)th candidate embedding pair are, respectively, fused in a reinforcement way through graph attention-based bilinear pooling. This can yield a reinforced target appearance representation for the inner product calculation. It is noted that in traditional Siamese networks, the accuracy of the target location relies on the cross correlation between the exemplar and target candidates. In contrast, our network structure can give a more accurate similarity calculation result. The reason for that is we fully exploit the inherent partial feature interaction existing in the multisource embedding pair through adopting graph attention-based bilinear pooling module. Section III-B introduces the graph attention-based bilinear pooling module in detail. Before introduction, we summarize the notation of main mathematical symbols as follows.

1) \(X_t\) is the video frame at time \(t\). In the proposed network, \(X^1\) and \(X^2\) means the video frame in RGB and thermal image domains, respectively. For simplicity, the subscript \(t\) can be omitted. The \(i\)th candidate pair extracted from \(X^1\) and \(X^2\) are denoted as \(x^1_i\) and \(x^2_i\).
2) \(F\) denotes the feature map tensor obtained from CNNs. The reshaped feature map tensor is denoted as \(\tilde{F}\) and the projection result of feature map tensor is \(\hat{F}\).
3) \(V\) is the bilinear pooling results. Notation \(V\) with a subscript in the follow up section indicates the inputs of Siamese networks. Specifically, the bilinear pooling of exemplar pair is denoted as \(V_x\), while the bilinear pooling of \(i\)th candidate pair is \(V_{x_i}\). The reduced dimensional vector of \(V_x\) and \(V_{x_i}\) are denoted as \(\tilde{V}_x\) and \(\tilde{V}_{x_i}\), respectively.
4) \(Q\) denotes similarity matrix in Siamese network structure.

### B. Graph Attention-Based Bilinear Pooling

The deep CNN has acquired remarkable achievement in visible spectrum camera-based classification. However, for RGB-T tracking, the state-of-the-art network structures often use linear pooling, e.g., concatenation or element-wise addition, to fuse multilayer multichannel feature maps, which may not make the target fusion result sufficiently expressive to capture the complementary advantages among isolate targets. Above limitation arises from the fact that the deep feature maps are considered as holistic features, and the intrinsic elementwise interaction between different feature maps cannot be fully explored. Bilinear pooling is a promising module that can overcome the limitation of linear pooling because it uses outer product to explore pairwise correlation between feature channels. Suppose we have obtained two domain feature map tensors \(F^1 \in \mathbb{R}^{N \times K \times C}\) and \(F^2 \in \mathbb{R}^{N \times K \times C}\) (\(N\) and \(K\) are the length and width of a single feature map, and \(C\) indicates the number of the feature map channels). After using outer product to multiply the locations of the two tensors and pooling all products together, we can finally obtain the bilinear vector \(V \in \mathbb{R}^{C^2 \times 1}\). Since a single element in feature map corresponds to a certain block in original images, if considering the target block as local pattern, the outer product in bilinear pooling can actually explore the structural relationship among local patterns in two image domains. In this way, we can use conditional partial information to represent the target appearance. Reformulating tensors \(F^1\) and \(F^2\) in matrix form \(\tilde{F}_1 \in \mathbb{R}^{NK \times C}\) and \(\tilde{F}_2 \in \mathbb{R}^{NK \times C}\), the bilinear pooling vector can be formulated as

\[
V = \text{bilinear}(\tilde{F}_1^1, \tilde{F}_2^2) = \text{vec}(\tilde{F}_1^1)^T \tilde{F}_2^2
\]

where \(\tilde{F}_1^1 = [\tilde{f}_1^1, \ldots, \tilde{f}_1^1, \ldots, \tilde{f}_1^1]\) and \(\tilde{F}_2^2 = [\tilde{f}_2^2, \ldots, \tilde{f}_2^2, \ldots, \tilde{f}_2^2]\), and the \((i-1)C+j\)th element in vector \(V\) is denoted as \(V_{(i-1)C+j}\) (\(i\)) represents the bilinear operator. Each element in vector \(\tilde{F}_1^1\) (or \(\tilde{F}_2^2\)) indicates the conditioned local pattern representation for an image block. Equation (1) implies each local pattern representation has equal importance, while ignoring a fact that the contribution of the columns in \(\tilde{F}_1^1\) and \(\tilde{F}_2^2\) for multimodal fusion are actually varied. Taking Fig. 1 as an example, there exist only a few image blocks that contain useful yet matched information in RGB-thermal pair. The uninformative image blocks would severely degrade the pooling performance. Thus it is of crucial importance to discriminate the contribution of image blocks. From this observation, we design a graph attention-based bilinear pooling module to exploit coattention mechanism [36]. Specifically, the element of \(V\) is reformulated as

\[
V_{(i-1)C+j} = (\tilde{f}_1^i)^T W_{jj} \tilde{f}_j^2
\]

where the coattention weight matrix \(W_{jj}\) is aimed to indicate the correlation between elements in vectors \(\tilde{f}_1^i\) and \(\tilde{f}_j^2\). Based on this design, we can highlight those elements in \(\tilde{f}_1^i\) and \(\tilde{f}_j^2\) that yield informative yet matched information.

The motivation of this article is to integrate the target embedding, coattention weight matrix estimation, and feature embedding fusion into a unified end-to-end network structure. To achieve this purpose, the proposed graph attention-based bilinear pooling module combines graph attention convolutional network (GACN) and outer product together, which can effectively utilize message passing to locate the informative image block in both RGB and thermal images with low-computational complexity. The problem formulation for the proposed graph attention-based bilinear pooling module is described as follows.

Based on matrix decomposition, \(W_{ij}\) can be decomposed into

\[
W_{ij} = P^T D_{ij} P
\]

where \(D_{ij}\) is the diagonal matrix, which can be further decomposed into two diagonal matrices \(D_{ij} = (S_j)^T S_j\). Defining \(D_i = S_i P\) and \(D_j = S_j P\). Based on this definition, taking (3) into (2), we can obtain

\[
V_{(i-1)C+j} = (\tilde{f}_1^i)^T (D_j)^T P^T P D_i \tilde{f}_j^2 = (P_i \tilde{f}_1^i)^T (P_j \tilde{f}_j^2).
\]

From (4) we can see that \(P_i = PD_i\). Defining \(\tilde{f}_1^i = P_i^T \tilde{f}_1^i\), we can obtain

\[
P_i \tilde{f}_1^i = (PD_i P^T)^T \tilde{f}_1^i.
\]
\( \mathbf{D}_i \) is the square matrix, it can be further decomposed. Based on this observation and suppose \( \mathbf{P} \) is the eigenvector of Laplacian matrix, \( (\mathbf{PD}_i \mathbf{P}^T) \hat{f}_i \) can also be updated using graph convolution. Similarly, \( \mathbf{D}_j \) can be also updated using graph convolution. Based on above analysis, let \( \hat{G}( \hat{\mathbf{F}}^1, \hat{\mathbf{A}}^1 ) \) and \( \hat{G}( \hat{\mathbf{F}}^2, \hat{\mathbf{A}}^2 ) \) be the attributed graphs for the RGB and thermal feature map tensors, respectively, where the rows in \( \hat{\mathbf{F}}^i \) (\( i = 1, 2 \)) are denoted as the nodes in the \( i \)-th graph and \( \hat{\mathbf{A}}^i \) is the adjacent matrix which encodes the pairwise similarity between nodes pairs. The bilinear pooling-based multiple graphs learning problem is formulated as

\[
\mathbf{V} = \text{bilinear}( \hat{G}( \hat{\mathbf{F}}^1, \hat{\mathbf{A}}^1 ), \hat{G}( \hat{\mathbf{F}}^2, \hat{\mathbf{A}}^2 ); \Theta) \tag{6}
\]

where graphs \( \hat{G}( \hat{\mathbf{F}}^1, \hat{\mathbf{A}}^1 ) \) and \( \hat{G}( \hat{\mathbf{F}}^2, \hat{\mathbf{A}}^2 ) \) can be learned by graph CNNs, \( \Theta = (\Theta^1, \Theta^2) \) is defined as the parameter set of graph CNNs, bilinear(-) means the bilinear operator that uses outer product to dynamically aggregate two graph CNNs. Traditional graph CNN often use predefined adjacent matrices for the single graph learning. In (6), due to challenging factors such as occlusion and thermal crossover (TC) and so on, the graph nodes correlation in different image domains dynamically changes, thus it is extremely difficult to simultaneously define the suitable adjacent matrices \( \hat{\mathbf{A}}^i \). Here, we build GACN for achieving graph learning without any prior knowledge. Specifically, we simplify (5) as

\[
\mathbf{P}_i \hat{f}_i = \mathbf{\sigma}(\mathbf{\sum}_{k \in \mathcal{N}(i)} \eta(i, k) \hat{\mathbf{f}}_k^1) \tag{7}
\]

where \( \eta(i, k) \) denotes the weight of the edge between nodes \( i \) and \( k \), \( \mathbf{\sigma}(-) \) is the activation function, \( \mathcal{N}(i) \) denotes neighbor set of node \( i \). Based on (6), we adaptively learn \( \eta(i, j) \) to estimate \( \mathbf{P}_i \hat{f}_i \). Similarly, \( \mathbf{P}_j \hat{f}_j \) can be estimated in the same way. Similar to [37], the weight \( \eta(i, k) \) for \( \hat{G}( \hat{\mathbf{F}}^1, \hat{\mathbf{A}}^1 ) \) is calculated by

\[
\eta(i, k) = \frac{\exp (\text{LeakyReLU}(\mathbf{\beta}^T [\mathbf{U}^1_i \| \mathbf{U}^1_k]))}{\mathbf{\sum}_{k \in \mathcal{N}(i)} \exp (\text{LeakyReLU}(\mathbf{\beta}^T [\mathbf{U}^1_i \| \mathbf{U}^1_k]))} \tag{8}
\]

where \( \mathbf{\beta} \) denotes the parameter vector of the single-layer feedforward neural network and \( \mathbf{U} \) is the parameter matrix that indicates the relation between \( \hat{\mathbf{A}} \) and \( \hat{\mathbf{A}}^i \). Different from traditional graph attention methods, \( \mathbf{U} \) is to learn the normalized row-wise representation of coattention matrix, which can use pairwise information from the other image domain to give a restriction to the estimation of attention weights, avoiding static attention [38] drawback. \( \| \cdot \| \) is the concatenation operator, and LeakyReLU(-) is the activation function.

### C. Updating Strategy

In this article, we would like to reformulate the updating of graph attention-based bilinear pooling results as a one-shot learning problem. This intuition is derived from an observation: the tracking result of the current frame is actually the positive sample. Those candidates that have less similarity with exemplar can be considered as the negative samples. No matter what dramatic changes the current candidates have suffered, the exemplar and current tracking result should still have the same category. Based on this observation, we can incorporate the category information in the online updating of \( \mathbf{\tilde{V}}_z \) (\( \mathbf{\tilde{V}}_z \) is the fully connected layer after yielding the bilinear vector of exemplar pair). Specifically, we define the \( k \)-th classification score of \( \mathbf{\tilde{V}}_z \) at the first frame as \( s_k \), where \( s_k = (\mathbf{\tilde{V}}_z \circ \mathbf{\phi})^T \mathbf{M}_k \), with \( \mathbf{M}_k \) denoting the weight vector for the \( k \)-th classification. Inspired by Gidaris and Komodakis [34], we introduce parameter vector \( \mathbf{\phi} \) in the classification. In this case, the \( k \)-th classification score after the first frame is changed as \( s_k = (\mathbf{\tilde{V}}_z \circ \mathbf{\phi})^T \mathbf{M}_k \), where \( \circ \) denotes Hadamard product. Based on this definition, we can adopt meta-learning to online learn \( \mathbf{\phi} \) for the fine-tuning of \( \mathbf{\tilde{V}}_z \). The detailed fine-tuning process is achieved using the category information to enforce \( \mathbf{\tilde{V}}_z \circ \mathbf{\phi} \) to be similar to the bilinear vector of positive candidate pair. Aforementioned strategy is helpful for enhancing the capability of discriminating the exemplar and background.

To achieve meta-learning, we define the \( i \)-th candidate pooling result \( \mathbf{\tilde{V}}_e \) that is most similar to \( \mathbf{\tilde{V}}_z \) as positive sample \( c_1 \), while the \( j \)-th candidate pooling result \( \mathbf{\tilde{V}}_e \) that has lowest similarity with \( \mathbf{\tilde{V}}_z \) is defined as the negative example \( c_2 \). The loss for the online training of \( \mathbf{\phi} \) is defined as

\[
J(\mathbf{\phi}) = -\log \mathcal{P}(y = 1 | \mathbf{\tilde{V}}_z)
\]

where \( \mathcal{P}(y = 1 | \mathbf{\tilde{V}}_z) \) is defined as

\[
\mathcal{P}(y = 1 | \mathbf{\tilde{V}}_z) = \frac{\exp (\| \mathbf{\tilde{f}}(\mathbf{\tilde{V}}_z) - c_1 \|^2)}{\sum_{k=1}^2 \exp (\| \mathbf{\tilde{f}}(\mathbf{\tilde{V}}_z) - c_k \|^2)}. \tag{10}
\]

It should be noted that the motivation of the meta-learning strategy in [34] and our meta-learning are quite different, i.e., [34] adopts meta-learning to train parameter vector \( \mathbf{\phi} \) for fine-tuning classification weight matrix \( \mathbf{M}_k \). Its aim is to recognize new categories. In contrast, we adopt \( \mathbf{\phi} \) to fine-tune semantic representation for template updating.

### D. Inner Product-Based Logistical Loss

As it is shown in Fig. 2, the outputs from two graph attention bilinear pooling modules are defined as bilinear vectors \( \mathbf{V}_z \) and \( \mathbf{V}_e \). The dimension of \( \mathbf{V}_z \) and \( \mathbf{V}_e \) are all 65536, thus we apply two fully connected layers after obtaining \( \mathbf{V}_z \) and \( \mathbf{V}_e \). This can reduce the dimension of \( \mathbf{V}_z \) and \( \mathbf{V}_e \) to 256, making them yield dense feature representation. The final outputs of the two graph attention bilinear pooling modules are \( \mathbf{\tilde{V}}_z \) and \( \mathbf{\tilde{V}}_e \). Since the exemplar and candidate pooling results are not the matrices as that in traditional Siamese network, we use inner product to measure the similarity between \( \mathbf{\tilde{V}}_z \) and \( \mathbf{\tilde{V}}_e \). Defining \( Q(\mathbf{\tilde{V}}_z, \mathbf{\tilde{V}}_e) \) as a similarity score in similarity map, the final similarity map is represented as

\[
Q = \begin{bmatrix}
Q_{11} & Q_{12} & \cdots & Q_{1r} \\
Q_{21} & Q_{22} & \cdots & Q_{2r} \\
\vdots & \vdots & \ddots & \vdots \\
Q_{l1} & Q_{l2} & \cdots & Q_{lr}
\end{bmatrix}
\]

where for the sake of simple expression, let \( Q(\mathbf{\tilde{V}}_z, \mathbf{\tilde{V}}_e) = Q_i \), and \( Q_i \) is the \( i \)-th element in matrix \( Q \). The point with highest similarity score indicates the location of the target. After locating the highest similarity score, we can use interpolation to find the bounding box of the target in the search area.
Authorized licensed use limited to: NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS. Downloaded on March 28, 2023 at 12:42:58 UTC from IEEE Xplore. Restrictions apply.
operators for visual tracking (C-COT) [51], efficient convolution operator (ECO) [52], multi-cue correlation filters for robust visual tracking (MCCT) [53], hierarchical convolutional feature (HCF) [54], MDNet [55], CFnet [56], joint sparse representation (JSR) [57], channel and spatial reliability (CSR) [6], cross-modal ranking (CMR) [58], sparse representation regularized graph tracking (SGT) [4], graph convolutional tracking (GLT) [39], cross-modal pattern-propagation (CMPP) [59], and self-SDCT+RGB [60]. In those competitors, JSR, CSR, CMR, SGT, GLT, and CMPP are the state-of-the-art RGB-T trackers. Beside of RGB-T trackers, other competitors are RGB camera-based trackers. It should be noted that all of RGB camera-based competitors are extended to RGB-thermal version for fair comparison. Specifically, we stack the RGB and thermal features into a single vector for traditional handcraft-based RGB trackers (e.g., MTT, MEEM, and INLCF). Meanwhile, we consider the thermal video sequence as an extra channel in the correlation filter and deep learning-based trackers (e.g., C-COT, ECO, MCCT, MDNet, HCF, and CFnet). The original RGB trackers that have been extended to RGB-T version are given annotation “+RGBT.” Since ECO is a representative RGB tracker, we also compare our method with this method to test the tracking performance between two-model fusion and a single model-based trackers. The original RGB tracker is given annotation “+RGB.”

C. Quantitative Tracking Experiments

1) GTOT Dataset:

a) Overall performance: The overall tracking performance on GTOT dataset is shown in Fig. 4. We can clearly see that our method gives the best precision performance. Specially, the distance precision score of our method is higher than ECO-RGBT by over 5%. Since ECO-RGBT involves thermal information, its distance precision score is slight higher than ECO-RGB. The tracking performance in Fig. 4(a) can verify the effectiveness of the proposed fusion module. It is seen from Fig. 4(b) that our method also gives the highest the area under curve (AUC) score. Especially, the AUC score of our method is higher than top RGB-T tracker CMR by over 1%. This can illustrate that our method can use an appropriate bounding box scale to locate the target.

b) Attribute-based performance: The position error only measures the distance between key pixels, which could not reflect the scale of the target. Comparing with position error, the overlap score often gives more comprehensive evaluation on tracking methods because it can evaluate the scale of target bounding box. Thus, we use averaged overlap score to evaluate the tracking performance of different method over seven challenging factors (As seen from Fig. 5), the averaged overlap score of our method in OCC, LSV, LI, and DEF scenarios are higher than other 13 methods. This result can validate our advantage that the graph attention-based bilinear pooling module can explore the partial feature interaction between the RGB and thermal targets. Beside OCC, LSV, LI, and DEF, our method still gives top-2 overlap score in other three attributes, which can illustrate that our method can locate the target with an appropriate bounding box in various challenging scenarios.

2) RGBT234 Dataset:

a) Overall performance: The overall tracking performance on RGBT234 dataset is shown in Fig. 6. RGBT234 contains much more video pairs and involves more challenging factors than GTOT dataset. Thus, it can give a comprehensive and convincing testing on the tracking performance. From Fig. 6(a), we could clearly see that the distance precision score of our method is obviously higher than other 13 comparing methods. Similarly, our method also wins the first place in the successful plot [see Fig. 6(b)]. Especially, the AUC score of our method is higher than well-known deep learning and correlation filter-based trackers such as multi-domain network (MDNet)+RGBT and ECO+RGBT by over 1.5%. This can give a strong support to validate the effectiveness of the proposed network structure. Besides this test, we also give the overall tracking performance on RGBT210 dataset in Fig. 7. RGBT234 dataset is the extension of RGBT210 dataset. From Fig. 7, we could clearly see that our method still win the top place when comparing with other methods.

b) Attribute-based performance: The precision plots over 12 challenging factors are shown in Table I. From this test we could clearly see that our method wins the first place in most challenging factors. Specifically, HO is very challenging because there only a few useful information can be extracted from the RGB and thermal targets. Due to this reason, state-of-the-art tracking methods such as ECO, CMR, and GLT give poor tracking performance in this scenario. Different from traditional methods, the success rate of our method is higher than top method CMPP over 10%. Beside of HO, BC, CM, Fast Motion (FM), LI, and PO are often be considered as the challenging scenarios that can be used as the representative tests to verify the tracking accuracy. Obviously, our method can also enhance the success rate over 6% when comparing with CMPP. Since the thermal target appearance would be seriously disturbed in TC, pooling module may suffer more negative effect when exploring the block relation. Due to this reason, the holistic deep feature-oriented tracker (ECO+RGBT) gives the best success rate. From the testing results in Table I, we can give a confident conclusion that our method can effectively use graph attention-based bilinear pooling module to enhance the tracking performance in challenging scenarios.
The best two results are denoted as red and blue.

TABLE I
MEAN VALUE OF SUCCESS RATE OVER DIFFERENT VIDEO SUBSETS IN RGB-234 DATASET.

<table>
<thead>
<tr>
<th>Method</th>
<th>DEF</th>
<th>FM</th>
<th>LI</th>
<th>OCC</th>
<th>SO</th>
<th>TC</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC</td>
<td>54.1</td>
<td>53.0</td>
<td>49.9</td>
<td>50.7</td>
<td>38.4</td>
<td>34.1</td>
<td>37.5</td>
</tr>
<tr>
<td>CM</td>
<td>53.3</td>
<td>50.8</td>
<td>47.0</td>
<td>43.1</td>
<td>31.5</td>
<td>31.8</td>
<td>30.8</td>
</tr>
<tr>
<td>DBF</td>
<td>53.2</td>
<td>52.3</td>
<td>46.9</td>
<td>41.2</td>
<td>30.1</td>
<td>31.4</td>
<td>30.6</td>
</tr>
<tr>
<td>FM</td>
<td>54.5</td>
<td>53.5</td>
<td>49.8</td>
<td>43.8</td>
<td>31.5</td>
<td>30.7</td>
<td>30.7</td>
</tr>
<tr>
<td>HO</td>
<td>54.1</td>
<td>50.7</td>
<td>47.2</td>
<td>43.8</td>
<td>30.5</td>
<td>30.2</td>
<td>30.7</td>
</tr>
<tr>
<td>LE</td>
<td>56.2</td>
<td>52.1</td>
<td>50.3</td>
<td>48.2</td>
<td>33.9</td>
<td>34.4</td>
<td>37.5</td>
</tr>
<tr>
<td>LR</td>
<td>53.2</td>
<td>58.4</td>
<td>56.3</td>
<td>53.6</td>
<td>59.4</td>
<td>48.2</td>
<td>53.3</td>
</tr>
<tr>
<td>MB</td>
<td>54.5</td>
<td>55.2</td>
<td>49.5</td>
<td>43.3</td>
<td>33.8</td>
<td>34.1</td>
<td>29.5</td>
</tr>
<tr>
<td>NO</td>
<td>77.1</td>
<td>68.2</td>
<td>59.7</td>
<td>53.8</td>
<td>37.1</td>
<td>34.1</td>
<td>34.1</td>
</tr>
<tr>
<td>PO</td>
<td>62.5</td>
<td>62.2</td>
<td>52.2</td>
<td>43.8</td>
<td>34.6</td>
<td>33.7</td>
<td>30.8</td>
</tr>
<tr>
<td>SC</td>
<td>59.7</td>
<td>61.2</td>
<td>55.6</td>
<td>48.7</td>
<td>37.0</td>
<td>41.4</td>
<td>24.7</td>
</tr>
<tr>
<td>TC</td>
<td>57.2</td>
<td>70.0</td>
<td>62.4</td>
<td>51.3</td>
<td>41.1</td>
<td>35.6</td>
<td>37.0</td>
</tr>
<tr>
<td>Average</td>
<td>59.7</td>
<td>58.6</td>
<td>54.0</td>
<td>48.3</td>
<td>33.6</td>
<td>41.7</td>
<td>37.0</td>
</tr>
</tbody>
</table>

D. Qualitative Tracking Experiments

Here we show the qualitative tracking performance in Fig. 8, where three video sequences are randomly selected from each scenario. The moving target is often occluded by the tree trunk in diamond sequence. State-of-the-art methods often lose the target after serious occlusion. From Fig. 8(a), we see that our method can still follow the target no matter the partial or HO. The target and adjacent pedestrians move together, causing serious BC in Fig. 8(b). In this scenario, our method can do the same as ECO-RGBT that gives a good tracking performance. In kite sequence, other methods would begin to drift in some extent after the 300th frame, while our method can still track the kite in whole video frames [Fig. 8(c)]. It contains severe haze in Fig. 8(e). Besides this challenging factor, it also involves occlusion and BC in Fig. 8(e). From this test, we observe that our method can still use an appropriate
Fig. 8. Qualitative results on six video pairs: (a) diamond video pair; (b) Elecbike3 video pair; (c) Kite4 video pair; (d) Manafterrain video pair; (e) Fog video pair; and (f) Nightthreepeople video pair.

Fig. 9. Qualitative example of RGB-T tracking result in kite sequence pair. Tracking result of (a) RGB image and (b) thermal image.

bounding box to locate the target appearance, while the scale of CMR has dramatically changed when facing occlusion. Kite sequence is a very challenging sequence because the target is really small. Fig. 8(d) and (f) suffer LI in raining and night scenarios. From the two examples, we can see that our method can effectively use the thermal information to complement the RGB sequences.

<table>
<thead>
<tr>
<th>Backbone</th>
<th>Outer product</th>
<th>GACN</th>
<th>Updating module</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG-16</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Ours</td>
</tr>
<tr>
<td>VGG-16</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Ours I</td>
</tr>
<tr>
<td>VGG-16</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Ours II</td>
</tr>
</tbody>
</table>

**Detailed Discussion:** Here we take kite sequence pair as example to show the advantage of our method in more detail. Kite sequence is very challenging because the target is small with a long tail. From the local enlarged image in Fig. 9, we can see that the tail of the kite is immersed in the background in both RGB and thermal images. Especially in thermal image, there exists only a few pieces of useful information for the representation of the kite appearance. From the comparison between top-3 tracking results in Fig. 9, we can get a conclusion that our method can effectively use the partial information in RGB and thermal target to guarantee the tracking accuracy.
TABLE III
FPS PERFORMANCE ON DIFFERENT RGB CAMERA-BASED TRACKERS

<table>
<thead>
<tr>
<th>Tracker</th>
<th>Our</th>
<th>DSST</th>
<th>CPnet</th>
<th>KCP</th>
<th>MEBM</th>
<th>MTT</th>
<th>INCLF</th>
<th>MCCT</th>
<th>SWOP</th>
<th>HCP</th>
<th>MDnet</th>
<th>MCPF</th>
<th>SimaFC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete FPS</td>
<td>9.3</td>
<td>10.6</td>
<td>29.5</td>
<td>40.2</td>
<td>33.2</td>
<td>4.8</td>
<td>2.5</td>
<td>2.9</td>
<td>4.9</td>
<td>7.8</td>
<td>0.8</td>
<td>0.2</td>
<td>31.5</td>
</tr>
<tr>
<td>Success rate</td>
<td>0.701</td>
<td>0.481</td>
<td>0.466</td>
<td>0.419</td>
<td>0.564</td>
<td>0.558</td>
<td>0.545</td>
<td>0.642</td>
<td>0.609</td>
<td>0.583</td>
<td>0.654</td>
<td>0.625</td>
<td>0.591</td>
</tr>
</tbody>
</table>

Fig. 10. Ablation test on GTOT dataset: (a) precision plot and (b) success plot.

Fig. 11. Ablation test on RGBT234 dataset: (a) precision plot and (b) success plot.

E. Ablation Study

1) Ablation on Graph Attention-Based Bilinear Pooling Module: Graph attention-based bilinear pooling module is the core in our FS-Siamese network, which mainly contains three components: GACN, outer product, and updating module. In this test, we carry out ablation study on GTOT and RGBT234 datasets to show the effectiveness of different components. The detailed experiment setting is shown in Table II. From Figs. 10 and 11 we can see that the precision and success plots on two datasets indicate the effectiveness of our graph attention-based bilinear pooling module.

2) Effectiveness of GACN: GACN is the key point in graph attention-based bilinear pooling, which can highlight the important image blocks through exploring partial feature interaction. In this section, we design a fine-grained classification test to show the effectiveness of GACN. Specifically, we add GACN at the end of Conv layers of multi-attention (MA)-CNN network [61]. In this way, the target embedding would pay more concentration on informative target block. The estimated subregion masks in Fig. 12(c) and (d) can indicate the effectiveness of GACN. For example, although the resolution of the local enlarged images in the first and third rows are low, MA-CNN+GACN can still locate the informative subregion [see Fig. 12(c)]. In contrast, the original method may involve uninformative background noise in the mask [see Fig. 12(d)].

3) Generality of Graph Attention-Based Bilinear Pooling Module: B-CNN [11] is a well-known method in fine-grained recognition that can use bilinear pooling to fuse feature maps of two network structures. Here we extend our graph attention-based bilinear pooling module to this method, namely, “B-CNN+GACN,” for verifying the generality of the key innovation of FS-Siamese. The testing is carried out on three fine-grained recognition datasets: CUB-200-2011, FGVC-aircraft, and Cars. From Fig. 13 we could clearly see that B-CNN+GACN can obviously enhance the recognition accuracy over 3% when comparing when original B-CNN methods.

F. Tracking Speed

In this test, we use the number of frames per second (FPS) as the objective measure for evaluating the online tracking speed. The FPS testing is carried out on computer workstation with single NVIDIA GTX1060Ti GPU. It should be noted that this test is carried out on RGBT234 dataset. Table III gives the tracking speed comparison between our method and RGB camera-based tracking methods. Since our method

Authorized licensed use limited to: NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS. Downloaded on March 28,2023 at 12:42:58 UTC from IEEE Xplore. Restrictions apply.
involves two more VGG-16-based feature embedding paths and 2) achieves updating at every frame when comparing with traditional SimaFC, this may reduce the tracking speed in some extent. We have also shown speed comparison between our method and the state-of-the-art RGB-T trackers in Table IV. Clearly, our tracking speed surpasses traditional handcraft based RGB-T trackers.

10 V. CONCLUSION

In this article, we have proposed a four-stream oriented Siamese network (FS-Siamese) to effectively fuse RGB and thermal information. Our network has benefited from the proposed graph attention-based bilinear pooling module that can adopt coattention mechanism to explore the partial feature interaction between the RGB and the thermal targets. Besides, we have adopted meta-learning to update the bilinear pooling result, which can perform online updating on the spatial relation between the target and its surrounding background. Extensive experiments on GTOT and RGBT234 datasets indicated that the proposed FS-Siamese network can give a superior performance as compared to the state-of-the-art RGB and RGB-T trackers.

11 REFERENCES


Table IV

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Compiler</td>
<td>python</td>
<td>matlab &amp; C++</td>
<td>matlab</td>
<td>matlab</td>
<td>matlab</td>
</tr>
<tr>
<td>FPS</td>
<td>9.3</td>
<td>7.0</td>
<td>0.9</td>
<td>2.3</td>
<td>1.3</td>
</tr>
<tr>
<td>Success rate</td>
<td>0.703</td>
<td>0.421</td>
<td>0.463</td>
<td>0.636</td>
<td>0.698</td>
</tr>
</tbody>
</table>

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.