# Deep Adaptive Fusion Network for High Performance RGBT Tracking

Yuan Gao[1], Chenglong Li[1,3], Yabin Zhu[1], Jin Tang[1,2], Tao He[4], Futian Wang[1]

[1]Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education,
School of Computer Science and Technology, Anhui University, Hefei 230601, China

[2]Key Laboratory of Industrial Image Processing and Analysis of Anhui Province, Hefei 230601, China

[3]Institute of Physical Science and Information Technology, Anhui University, Hefei 230601, China

[4]Anhui COWAROBOT CO., Ltd, Wuhu 231000, China

{gaoyuankong, lcl1314,zhuyabin0726}@foxmail.com,tangjin@ahu.edu.cn,
Tommie.he@cowarobot.com, wft@ahu.edu.cn

## Abstract

*Due to the complementarity of RGB and thermal data, RGBT tracking has received more and more attention in recent years because it can effectively solve the degradation of tracking performance in dark environments and bad weather conditions. How to effectively fuse the information from RGB and thermal modality is the key to give full play to their complementarity for effective RGBT tracking. In this paper, we propose a high performance RGBT tracking framework based on a novel deep adaptive fusion network, named DAFNet. Our DAFNet consists of a recursive fusion chain that could adaptively integrate all layer features in an end-to-end manner. Due to simple yet effective operations in DAFNet, our tracker is able to reach the near-real-time speed. Comparing with the state-of-the-art trackers on two public datasets, our DAFNet tracker achieves the outstanding performance and yields a new state-of-the-art in RGBT tracking.*

## 1. Introduction

RGBT (visible light and thermal infrared spectrums) tracking is a basic research topic in the field of computer vision, and it aim at estimating the state of a target object in a RGBT video pair given the initial ground truth bounding box. It draws more and more research interests with the popularity and affordability of thermal infrared sensors, which could provide many complementary benefits to RGB cameras [17, 20]. Although much progress has been achieved in RGBT tracking, how to fully explore and employ the complementary information of these two modalities is still a very challenging problem.

Existing RGBT tracking algorithms [17, 22, 23, 15, 21, 24, 39] mainly focus on the following two aspects. First, how to design suitable representation learning frameworks for RGBT tracking. For example, Li *et al.* [24] propose a cross modal ranking algorithm to calculate importance weight of each patch, and then construct robust RGBT feature descriptions of target objects. However, this method relies handcraft features and is thus difficult to adapt to complex tracking scenarios. In [21], a two-stream convolutional neural network (CNN) is proposed, in which the features of RGB and thermal data are extracted using two branch networks respectively, and then fused in the last layer. Semantic information is rich in deep layers, but lacks of spatial details [28], which plays critical role in target localization. Second, how to achieve adaptive fusion of different modalities for RGBT tracking. Early works [17, 23] base on the collaborative sparse representation in Bayesian filtering framework, in which the fusion is performed online by optimizing a reliability weight for each modality. While Lan *et al.* [15] optimize the modality weights using the max-margin principle according to classification scores. However, these approaches would be fail when sparse representations or classification scores are insufficient to reflect modality reliabilities. It also should be noted that most existing RGBT tracking methods have long delay due to complex optimization procedures.

To solve the above issues, we propose a novel deep adaptive fusion network for high performance RGBT tracking, by taking both robust feature learning and adaptive modality fusion into account in an end-to-end deep learning framework. Our network architecture is shown in Figure 1 (a). In general, low level features contain abundant spatial information but lack of semantic information, while high-level features are the opposite [5, 16]. Therefore, we design a recursive chain of the adaptive fusion module (AFM) to integrate all modalities and layer features in a unified network. Herein, each AFM block takes features from differ-

ent modalities and previous layer a inputs and outputs the fused features. To suppress feature noises and redundancies introduced by the above aggregation, we calculate the channel weights for all feature maps using the global average pooling method. The details can be seen in Figure 1 (b). Given the feature maps outputted from the AFM chain, we extract features of each candidate directly on these feature maps and use the RoIAlign operation to keep accurate of feature extraction [12]. Then, the three fully-connected layers are adopted to adapt appearance and temporal variations of each instance [30]. The status of the target object is finally estimated through classification and regression layers.

In summary, this work has made the following three major contributions.

- We propose a novel end-to-end deep framework for near-real-time RGBT tracking. Our framework is able to efficiently, adaptively and recursively aggregating features from all layers of RGB and thermal modalities for robust RGBT representations.

- A lightweight adaptive fusion module is designed to integrate features from different modalities and previous layer while suppressing feature noises and redundancies.

- Extensive experiments on two public datasets, GTOT [17] and RGBT234 [20], are conducted to demonstrate the excellent performance in terms of both accuracy and speed of the proposed method against the state-of-the-art RGB and RGBT trackers.

## 2. Related Work

RGBT tracking has received more and more attention, and many algorithms are emerging [17, 22, 23, 15, 21, 24, 39].

In [22, 17], the authors use the reconstruction residues or coefficients to guide a learning a weight for each modality for adaptive fusion RGB and thermal information. While Lan *et al*. [15] employ the classification scores to guide the generation of fusion weights in a max-margin principle. However classification scores, reconstruction residues or coefficients sometimes are not reliable to represent modal information. There are also some work that focuses on how to construct robust RGBT feature representations [23, 24].In [23], a weighted sparse representation regularized graph is proposed to learn a robust RGBT target representation. The weights of image patches in the optimized graph are applied to construct robust weighted feature representations and the target is located by the structured SVM algorithm. In [24], a cross-modal manifold ranking algorithm is presented to compute patch weights, where the soft cross-modality consistency is used to explain the different properties between the two modes and a optimal query learning method is used to process seed noises.

Recently, the better algorithms are to apply deep learning techniques to RGBT tracking mainly include [21, 39]. In [21], a two-stream convolutional neural network is proposed to effectively fuse the two modalities information, and the target position is predicted effectively by multi-channel correlation filter. The most advanced algorithms such as DAPNet [39] are used for RGBT tracking through multi-layer feature fusion and collaborative feature pruning. This method achieves very high RGBT tracking performance. However, DAPNet does not consider the contributions of different layer and different modal features, but directly aggregates the features of all modal layers, which would introduce more redundant information and noise. In addition, the large number of parameters is harmful to the efficiency of tracking. By means of modal adaptive fusion module, We can effectively suppress noise and reduce the influence of redundant information, so as to realize RGBT tracking more accurately and efficiently.

## 3. Proposed Tracking Methodology

In this section, we describe the proposed tracker in detail, including network architecture, training and tracking details. Figure 1 gives an overview of the tracking framework.

### 3.1. Overall Network Architecture

As shown in Figure 1 (a), we adopt VGG-M [33] as the backbone network to extract the features of RGB and thermal images respectively. Considering the efficiency and effectiveness, we adopt the addition operation for the fusion of different layers and modalities, and the spatial sizes and channel numbers should thus be same. First, we add a max pooling operation in each layer and modality to downsample feature maps, and fuse feature maps outputted from each layer of RGB and thermal modalities using an adaptive fusion module (AFM). We will present the details of AFM in next section. Second, the features outputted from AFM are passed through a convolution layer with the size of $1\times1$ to increase the channel dimension.

Note that the AFM structures of the first layer and other layers are different due to different inputs. The inputs of AFM in the first layer are the features extracted from RGB and thermal modalities, while the inputs of other layers additionally include the features outputted from the previous layer, as shown in Figure 1 (a). These AFM blocks compose a recursive aggregation chain. The fusion chain can be expressed simply by the following formula:

$$X_o^i = \begin{cases} m(X_{rgb}^i, X_t^i), & i = 1 \\ m(X_{rgb}^i, X_t^i, X_o^{i-1}), & i > 1 \end{cases} \tag{1}$$

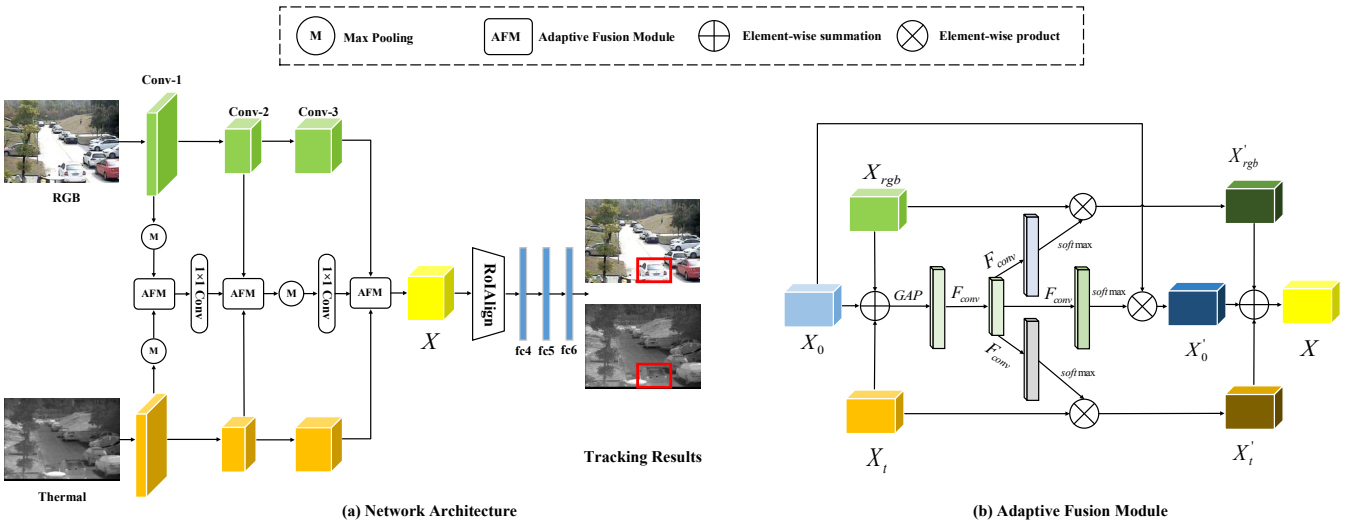



Figure 1. Illustration of the proposed network. (a) Details of the overall network. The orange and light green blocks represent the features extracted from thermal and RGB modalities respectively. (b) Details of the adaptive fusion module (AFM). $X_{rgb}$ and $X_t$ represent the features of RGB and thermal modalities respectively. $X_o$ represents the features outputted from previous AFM. $GAP$ denotes the global average pooling, and $F_{conv}$ represent the convolution operations.

where $X^i_{rgb}$ and $X^i_t$ represent the features extracted from RGB and thermal images in the $i$-th layer. $X^i_o$ represents the outputted features of the $i$-th layer. $m(\cdot)$ denotes the function of AFM.

Given the feature maps outputted from the AFM chain, we extract features of each candidate directly on these feature maps and use the RoIAlign operation to keep accurate of feature extraction [12]. Then, the three fully-connected layers are adopted to adapt appearance and temporal variations of each instance [30]. The status of the target object is finally estimated through classification and regression layers.

### 3.2. Adaptive Fusion Module

In this section, we describe the details of the adaptive fusion module (AFM). As shown in Figure 1 (b), inspired by[25] we design a set of weighting operations to make the network focusing on more advantageous areas for effective fusion of different modalities. The AFM includes modality aggregation and adaptive weighting, which are described in detail below.

**Information aggregation**. At this stage, the network aggregates the information from multiple branches from the global view. Note that the additive operation is one of the most commonly used feature fusion method, which is simple and efficient. Moreover, the direct addition of feature maps with similar semantic information can not only fuse complementary information and but also save a lot of parameters to improve tracking efficiency. Therefore, we fuse features from two or three branches using a method of element-wise addition:

$$X^i_{fuse} = \begin{cases} X^i_{rgb} + X^i_t, i = 1 \\ X^i_{rgb} + X^i_t + X^{i-1}_o, i > 1 \end{cases} \tag{2}$$

where $X^i_{rgb}$ and $X^i_t$ represent the features extracted from RGB and thermal images in the $i$-th layer. $X^i_o$ represents the outputted features of the $i$-th layer. The global average pooling method is used to generate channel weights $g \in \mathbb{R}^C$, and the $c$-th element of $g$ is obtained by the following formula:

$$g^c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X^c_o(i, j) \tag{3}$$

To soften the information of $g$ representing global information, we compress it through a $1\times1$ convolution operation, and obtain $p \in \mathbb{R}^d$:

$$p = \phi(\psi(F_{conv}(g))) \tag{4}$$

where $\phi$ and $\psi$ represent the ReLU function [29] and Batch Normalization [10], respectively. $d$ is set to 32 in our experiments.

**Adaptive weighting**. To make full use of aggregate information while suppressing feature noises and redundancies, we generate adaptive weights for fusing all branches. First, we use the vector $p$ to represent the comprehensive information, and then use the softmax activation function to compute the final weights of different branches:

$$w^{rgb} = softmax(F^{rgb}_{conv}(p)), w^t = softmax(F^t_{conv}(p)), \tag{5}$$

where $softmax$ denotes the softmax activation function. When the adaptive weighting module is applied to other layers besides the first layer, the following formula is required, similar to the above formula:

$$w^o = softmax(F^o_{conv}(p)). \tag{6}$$

As shown in Figure 1 (b), the AFM block can be described by the following formula:

$$X^i_o = \begin{cases} w^i_{rgb} * X^i_{rgb} + w^i_t * X^i_t, i = 1 \\ w^i_{rgb} * X^i_{rgb} + w^i_t * X^i_t + w^{i-1}_o * X^{i-1}_o, i > 1 \end{cases} \tag{7}$$

where $*$ denotes the element-wise product, and other explanations are found in the above formula (2).

### 3.3. Training Details

The entire network is trained end-to-end. The parameters of the backbone VGG-M are initialized by the pre-trained model on the ImageNet dataset [33]. We adopt the Adam algorithm [14] to optimize the network, and set the learning rates to 0.0001 and 0.001 for convolution and fully-connected layers respectively. For each training iteration, we first randomly obtain 8 frames of image and corresponding tracking target location coordinates from a video sequence. Then 512 positive samples and 1568 negative samples are obtained from the above 8 frames by Gaussian sampling, where 64 positive samples and 196 negative samples are required for each image frame. When the overlap rate between the candidate box obtained by Gaussian sampling and the ground truth value is in the range of [0.7,1], it is regarded as a positive sample. When the range is [0,0.5], it is regarded as a negative sample. We train the network with 146 epoch iterations. It is worth noting that we train on the GTOT dataset to evaluate our tracker on the RGBT234 dataset, and train it on the RGBT234 to evaluate on the GTOT.

### 3.4. Online Tracking

In the tracking phase, first, as most trackers adopted, we use the first frame of a sequence and the location of the target to initialize the tracker. Gaussian sampling is used to obtain 500 positive samples of different scales around the target in the first frame, and the details are discussed above. At the same time, 1000 samples are taken to train the regressor. The regressor is used to modify the coordinates of the tracking results in the follow-up tracking, so as to obtain more accurate tracking results. It is worth noting that in the online tracking phase we only update the parameters of the fully-connected layers, the same strategy with [30]. When tracking the target in the $t$-th frame, we base the tracking result in the $t-1$-th frame combined with Gaussian sampling to obtain 256 samples in the current frame. We employ the trained model to calculate the scores of 256 samples, and take the 5 samples with the highest scores at present to calculate the mean value, and then refine the target position using the trained regressor. Similar to [30], the short-time updates used in tracking failures and long-time update Settings are adopted to ensure robustness of our algorithm.

## 4. Experiments

We evaluate the performance of our proposed approach on two public datasets, GTOT [17] and RGBT234 [20], Ablation study is used to verify the effectiveness of the major components in our method.

### 4.1. Dataset and Evaluation Metric

**Datasets**. The popular datasets, GTOT [17] and RGBT234 [20], in the RGBT tracking field are used in this paper. The GTOT dataset consists of 50 pairs of RGBT video pairs captured in different environments. The challenges are divided into seven categories based on the weather and time of the shoot and the status of the target. The RGBT234 dataset is extended from the RGBT210 dataset [23]. It contains 234 video pairs that are strictly aligned in two modalities. There are 234,000 images in total, where the longest video reaching 4,000 frames. The dataset contains more target occlusion, motion, and camera movement, as well as the tracking challenges posed by bad weather and insufficient light.

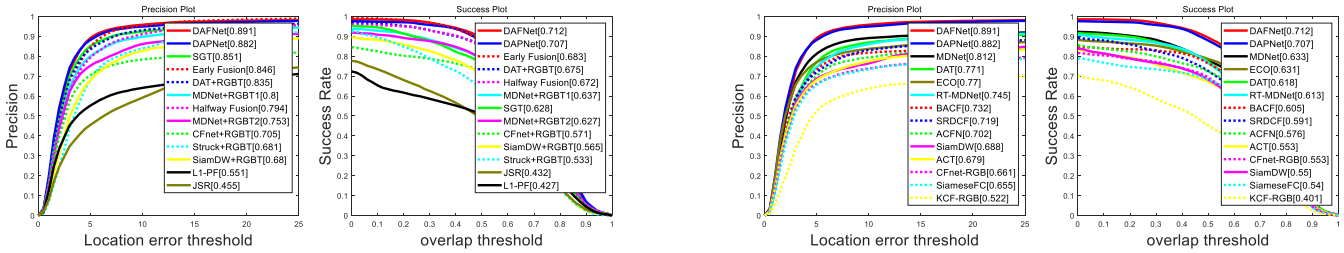
**Evaluation metrics**. We use two widely used metrics, Precision rate (PR) and success rate (SR) to evaluate the performance of the tracker. PR is the percentage of frames whose output position is within a given threshold distance. We employ the PR score with the threshold as 5 (GTOT) and 20 (RGBT234) pixels to define the representative PR. SR is the ratio of the number of successful frames whose overlaps are larger the predefined threshold. And we employ the area under the curves of success rate as the representative SR for quantitative performance evaluation.

### 4.2. Effectiveness of Deep Fusion Scheme

The feature maps in different layers shows different emphases, where the visual details at lower level are finer, and the semantics at deeper level are richer. To fully understand the effects of different fusion schemes on tracking performance, we design three fusion structures based on RT-MDNet [12], namely Early Fusion, Halfway Fusion, and Late Fusion. In a specific, Early Fusion directly concatenates two modal images form a six channel image and then input it into the network. Halfway Fusion refers to concatenating the convolution features of two modalities after the first layer. And Late Fusion concatenates the two modal feature maps from the last convolution layer. The experimental results are shown in Table 1, which suggests that our

Table 1. PR(%) and SR(%) scores of our algorithm on GTOT and RGBT234 comparing with different fusion stages.

| | | RT-MDNet | Early Fusion | Halfway Fusion | Late Fusion | DAFNet |
|---|---|---|---|---|---|---|
| GTOT | PR | 74.5 | 84.6 | 79.4 | 82.9 | **89.1** |
| | SR | 61.3 | 68.3 | 67.2 | 66.7 | **71.2** |
| RGBT234 | PR | 71.4 | 73.7 | 72.2 | 73.4 | **79.6** |
| | SR | 50.0 | 50.0 | 49.3 | 48.3 | **54.4** |




(a) Comparison with RGBT trackers

(b) Comparison with RGB trackers

Figure 2. Evaluation curves on GTOT dataset. PR(%) and SR(%) curves are used to evaluate the performance of trackers. (a) and (b) represent the comparison with RGBT trackers and RGB trackers, respectively.

Table 2. Evaluation results of our method with its variants on GTOT and RGBT234 datasets.

| | | DAFNet-noAFM | DAFNet-noAW | DAFNet |
|---|---|---|---|---|
| GTOT | PR | 84.6 | 87.3 | **89.1** |
| | SR | 68.3 | 69.6 | **71.2** |
| RGBT234 | PR | 73.7 | 76.2 | **79.6** |
| | SR | 50.0 | 50.8 | **54.4** |

Table 3. PR(%) and SR(%) scores and Speed of our DAFNet comparing with DAPNet [39] on RGBT234 and GTOT datasets.

| | | SGT | DAPNet | DAFNet |
|---|---|---|---|---|
| GTOT | Speed | 5fps | 2fps | **23fps** |
| | PR/SR | 85.1/62.8 | 88.2/70.7 | **89.1/71.2** |
| RGBT234 | Speed | 5fps | 2fps | **20fps** |
| | PR/SR | 72.0/47.2 | 76.6/53.7 | **79.6/54.4** |

DAFNet significantly outperforms other baseline methods, demonstrating the effectiveness of our deep fusion scheme.

### 4.3. Ablation Study

To verify the effectiveness of the main components of our proposed approach, we conduct an ablation study on GTOT and RGBT234 datasets. We implement two variants namely DAFNet-noAFM and DAFNet-noAW in this experiment. 1) DAFNet-noAFM, in this model we remove the adaptive fusion modules and it is the same with Early Fusion in Table 1. 2) DAFNet-noAW, here we remove the adaptive weighting operations for all layers. As can be seen from the experimental results in Table 2, RGBT tracking performance can be significantly improved through multi-layer adaptive feature fusion. This is mainly because the fusion of multi-layer features can not only ensure the aggregation of information at each scale of the two modalities, but also integrate the complementary advantages of low-level and high-level features. In addition, after adding the adaptive weighting operations, the performance is further improved, with the increase of PR and SR by 1.5% and 2.2% respectively. It is perhaps because that when aggregating multi-layer and multi-modal information, a large amount of redundant information and noises would inevitably be introduced, and thus we introduce channel weights to achieve the further improvement of tracker performance.

### 4.4. Efficiency Analysis

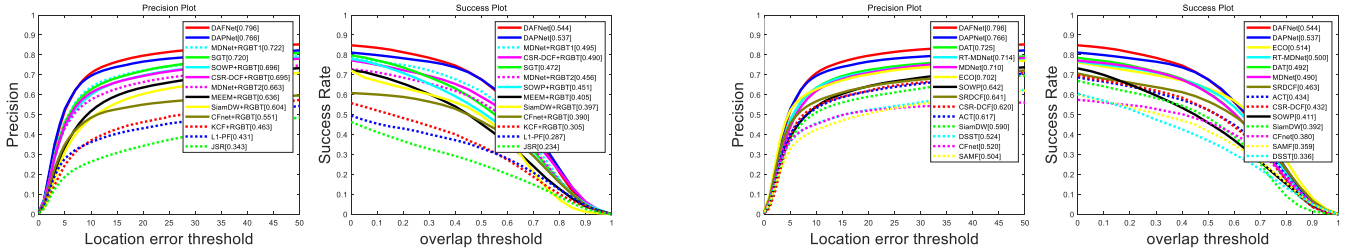
We implement the proposed method based on the platforms of Pytorch 0.4.0, an Intel(R) Xeon(R) CPU E5-2620 with a single CPU core (2.10GHz), 64GB RAM and a NVIDIA GeForce RTX 2080Ti GPU with 11GB of memory. Due to the operational efficiency of our network, the tracking speed can reach 23 FPS on average. Compared with the current best RGBT tracking algorithm DAPNet [39], we can achieve the current best performance in terms of both accuracy and speed. See table 3 for detailed comparison.

### 4.5. Comparison with State-of-the-art Methods

**Evaluation on GTOT**. We compare the RGBT tracker proposed in this paper with some of the state-of-the-art trackers available recently, including DAPNet [39], MDNet [30]+RGBT1, MDNet+RGBT2, DAT+RGBT [31], SiamDW+RGBT [38], CSR [17], L1-PF [36], SGT [23] and JSR [34]. Several of these are RGBT trackers [19, 18, 23, 39].The rest is to extend RGB trackers to RGBT ones, mainly by concatenating directly along the channel through

Table 4. PR(%) and SR(%) scores of the challenge-based performance comparison on the RGBT234 dataset. The best and second best results are marked in red and green, respectively.

| | SOWP+RGBT | CFNet+RGBT | KCF+RGBT | L1-PF | CSR-DCF+RGBT | MEEM+RGBT | SGT | Early Fusion | Halfway Fusion | DAPNet | DAFNet |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NO | 86.8/53.7 | 76.4/56.3 | 57.1/37.1 | 56.5/37.9 | 82.6/60.0 | 74.1/47.4 | 87.7/55.5 | 86.7/61.1 | 86.1/61.1 | 90.0/64.4 | 90.0/63.6 |
| PO | 74.7/48.4 | 59.7/41.7 | 52.6/34.4 | 47.5/31.4 | 73.7/52.2 | 68.3/42.9 | 77.9/51.3 | 81.3/55.2 | 75.8/51.5 | 82.1/57.4 | 85.9/58.8 |
| HO | 57.0/37.9 | 41.7/29.0 | 35.6/23.9 | 33.2/22.2 | 59.3/40.9 | 54.0/34.9 | 59.2/39.4 | 60.3/39.7 | 62.4/42.0 | 66.0/45.7 | 68.6/45.9 |
| LI | 72.3/46.8 | 52.3/36.9 | 51.8/34.0 | 40.1/26.0 | 69.1/47.4 | 67.1/42.1 | 70.5/46.2 | 71.3/47.4 | 72.6/49.3 | 77.5/53.0 | 81.2/54.2 |
| LR | 72.5/46.2 | 55.1/36.5 | 49.2/31.3 | 46.9/27.4 | 72.0/47.6 | 60.8/37.3 | 75.1/47.6 | 74.6/48.8 | 73.3/47.9 | 75.0/51.0 | 81.8/53.8 |
| TC | 70.1/44.2 | 45.7/32.7 | 38.7/25.0 | 37.5/23.8 | 66.8/46.2 | 61.2/40.8 | 76.0/47.0 | 72.1/50.2 | 73.8/51.6 | 76.8/54.3 | 81.1/58.3 |
| DEF | 65.0/46.0 | 52.3/36.7 | 41.0/29.6 | 36.4/24.4 | 63.0/46.2 | 61.7/41.3 | 68.5/47.4 | 64.8/46.2 | 66.9/47.1 | 71.7/51.8 | 74.1/51.5 |
| FM | 63.7/38.7 | 37.6/25.0 | 37.9/22.3 | 32.0/19.6 | 52.9/35.8 | 59.7/36.5 | 67.7/40.2 | 64.6/39.6 | 61.8/38.8 | 67.0/44.3 | 74.0/46.5 |
| SV | 66.4/40.4 | 59.8/43.3 | 44.1/28.7 | 45.5/30.6 | 70.7/49.9 | 61.6/37.6 | 69.2/43.4 | 73.9/50.5 | 73.9/50.5 | 78.0/54.2 | 79.1/54.4 |
| MB | 63.9/42.1 | 35.7/27.1 | 32.3/22.1 | 28.6/20.6 | 58.0/42.5 | 55.1/36.7 | 64.7/43.6 | 63.5/45.1 | 63.6/44.6 | 65.3/46.7 | 70.8/50.0 |
| CM | 65.2/43.0 | 41.7/31.8 | 40.1/27.8 | 31.6/22.5 | 61.1/44.5 | 58.5/38.3 | 66.7/45.2 | 64.6/44.7 | 63.7/44.8 | 66.8/47.4 | 72.3/50.6 |
| BC | 64.7/41.9 | 46.3/30.8 | 42.9/27.5 | 34.2/22.0 | 61.8/41.0 | 62.9/38.3 | 65.8/41.8 | 64.2/40.5 | 64.2/41.0 | 71.7/48.4 | 79.1/49.3 |
| ALL | 69.6/45.1 | 55.1/39.0 | 46.3/30.5 | 43.11/28.7 | 69.5/49.0 | 63.6/40.5 | 72.0/47.2 | 73.7/50.0 | 72.2/49.3 | 76.6/53.7 | 79.6/54.4 |




(a) Comparison with RGBT trackers

(b) Comparison with RGB trackers

Figure 3. Evaluation curves on RGBT234 dataset. PR(%) and SR(%) curves are used to evaluate the performance of trackers. (a) and (b) represent the comparison with RGBT trackers and RGB trackers, respectively.

two modalities, or by using thermal information as an additional channel, including, MDNet+RGBT1(concatenate two modes of data to form 6 channels of input data), DAT+RGBT, SiamDW+RGBT, CFnet+RGBT and MDNet+RGBT2(concatenate the feature maps of the two modes at conv3). From the result curves in Figure 2 (a), we can see that our tracker has significant performance gains over others. Note that DAPNet is the second best tracker, but the tracking speed of DAPNet is too slow ,it's only a tenth of our speed.

We also compare our tracker with some popular RGB trackers to justify the importance of thermal information in visual tracking, includes DAT [31], RT-MDNet [12], SiamDW [38], ACT [2], MDNet [30], ECO [4], BACF [8], SRDCF [7], ACFN [3], SiameseFC [1], CFnet [35] and KCF [9]. It can be seen from Figure 2(b) that our algorithm achieves clear improvement over RGB trackers, justifying the importance of thermal information in visual tracking.

**Evaluation on RGBT234**. To further verify the effectiveness of our method, we used a larger dataset RGBT234, so as to comprehensively verify the generalization ability of our tracker. As shown in Figure 3, we compare the proposed algorithm with 12 RGB algorithms (DAT [31], RT-MDNet [12], SiamDW [38], ACT [2], MDNet [30], ECO [4], SOWP [13], SRDCF [7], CSR-DCF [27], DSST [6], CFnet [35] and SAMF [26]) and 12 RGBT algorithms(MDNet+RGBT1, MDNet+RGBT2, SGT [23], SOWP+RGBT, CSR-DCF+RGBT, MEEM [37]+RGBT, CFnet+RGBT, KCF [9]+RGBT, JSR [34] and L1-PF [36]). After reviewing the results, we can find that our method is superior to all existing RGB algorithms in all evaluation metrics. In the comparison of the results of RGBT tracking algorithm, the precision rate (PR) of our algorithm is 3.0% higher than that of the recently proposed DAPNet [39], and the success rate (SR) is 0.7% higher than DAPNet. Note that we also achieve faster performance than DAPNet.

The following challenges are presented in the RGBT234 dataset based on weather, occlusion, camera shake, and target scale variations to comprehensively evaluate the performance of the tracker. The challenges include, no occlusion (NO), partial occlusion (PO), heavy occlusion (HO), low illumination (LI), low resolution (LR), thermal crossover (TC), deformation (DEF), fast motion (FM), scale variation (SV), motion blur (MB), camera moving (CM) and background clutter (BC). As shown in Table 4, we evaluate some of the most recent advanced tracking algorithms

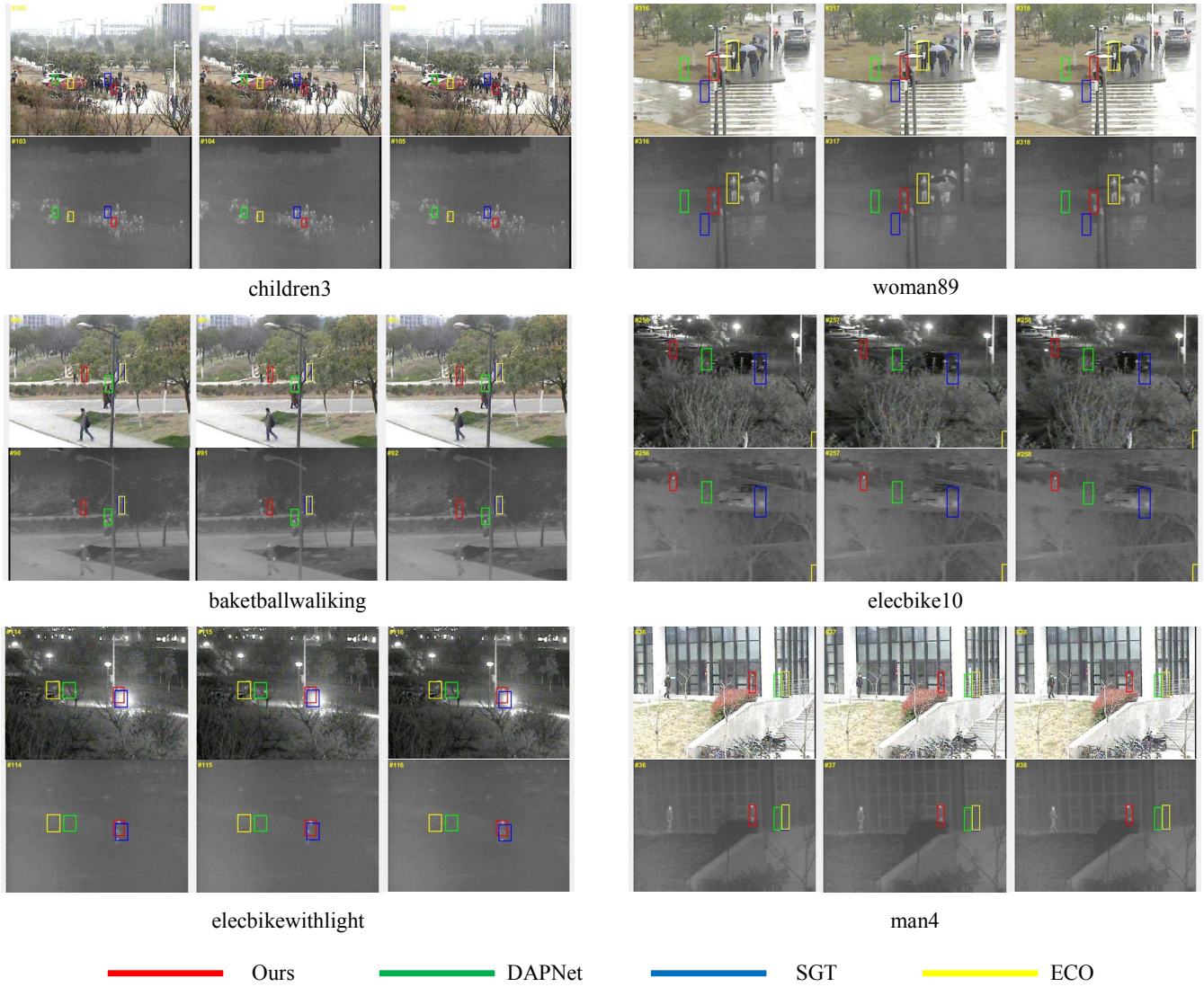



Figure 4. Visual results in complex scenarios comparing with the four advanced tracking algorithms, including DAPNet [39], SGT [23] and ECO [4].

based on these challenges comparing to our proposed algorithms. The comparison of tracking algorithms includes, L1-PF, KCF+RGBT, MEEM+RGBT, SOWP+RGBT, CSR-DCF+RGBT, CFNet+RGBT, SGT, MDNet+RGBT1 and DAPNet. As we can see from Table 4, our tracker achieves state-of-the-art performance in PR on all challenges. SR scores of our DAFNet are slightly lower than DAPNet in the deformation (DEF) and fast motion (FM) challenges. You can also see that in the LI challenge, our performance gains over DAPNet are significant. The main reason is that the information of RGB images will become fuzzy under low lighting conditions and the thermal information becomes very important. Our DAFNet employs a deep adaptive fusion scheme to incorporate more useful information and mitigate effects of noisy ones. In addition, on the TC challenge, our tracking results are 4.3% higher in the PR score and 4% higher in the SR score than the second best algorithm. It fully demonstrates that our adaptive weighting operation can adaptively integrate different feature maps.

Some qualitative results are shown in Figure 4, where some video sequences with different challenging factors are presented. By comparing it to the best recent tracker results, the approach we've come up with can better address the challenges of low illumination, thermal crossover, and bad weather, etc.

## 5. Conclusion

In this paper, we propose a robust RGBT tracking method based on a deep adaptive fusion network, in which we make full use of the complementary advantages of shallow and deep modal features and also introduce adaptive

weighting operations to effectively reduce feature noises and redundant information. Extensive experiments show that our method can effectively solve RGBT tracking challenges in difficult environments such as insufficient light, severe weather and occlusion. The state-of-the-art performance is achieved on public RGBT tracking datasets. In the future, we will develop an effective scale handling method in our framework to improve the robustness of tracking, similar to the Iou-Net algorithm [11] and RPN algorithm [32] in the field of object detection.

## Acknowledgement

This research is jointly supported by the National Natural Science Foundation of China (No. 61702002, 61976003, 61872005), Natural Science Foundation of Anhui Province (1808085QF187,1908085MF206), Open fund for Discipline Construction, Institute of Computer Science and Technology, Anhui University, and Natural Science Foundation of Anhui Higher Education Institutions of China (KJ2018A0023).

## References

[1] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. In *Proceedings of IEEE European Conference on Computer Vision*, 2016.

[2] Boyu Chen, Dong Wang, Peixia Li, Shuang Wang, and Huchuan Lu. Real-time actor-critic tracking. In *Proceedings of IEEE Conference on European Conference on Computer Vision*, 2018.

[3] Jongwon Choi, Hyung Jin Chang, Sangdoo Yun, Tobias Fischer, Yiannis Demiris, and Jin Young Choi. Attentional correlation filter network for adaptive visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[4] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg. Eco: Efficient convolution operators for tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[6] Martin Danelljan, Gustav Häger, Fahad Khan, and Michael Felsberg. Accurate scale estimation for robust visual tracking. In *Proceedings of British Machine Vision Conference*, 2014.

[7] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[8] Hamed Kiani Galoogahi, Ashton Fagg, and Simon Lucey. Learning background-aware correlation filters for visual tracking. In *Proceedings of IEEE Conference on International Conference on Computer Vision*, 2017.

[9] J. F. Henriques, R Caseiro, P Martins, and J Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of International Conference on Machine Learning*, 2015.

[11] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European Conference on Computer Vision*, 2018.

[12] Ilchae Jung, Jeany Son, Mooyeol Baek, and Bohyung Han. Real-time mdnet. In *Proceedings of IEEE European Conference on Computer Vision*, 2018.

[13] H.-U. Kim, D.-Y. Lee, J.-Y. Sim, and C.-S. Kim. Sowp: Spatially ordered and weighted patch descriptor for visual tracking. In *Proceedings of IEEE International Conference on Computer Vision*, 2015.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Xiangyuan Lan, Mang Ye, Shengping Zhang, and Pong C. Yuen. Robust collaborative discriminative learning for rgb-infrared tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[16] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[17] C. Li, H. Cheng, S. Hu, X. Liu, J. Tang, and L. Lin. Learning collaborative sparse representation for grayscale-thermal tracking. *IEEE Transactions on Image Processing*, 2016.

[18] C. Li, H. Cheng, S. Hu, X. Liu, J. Tang, and L. Lin. Learning collaborative sparse representation for grayscale-thermal tracking. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, 2016.

[19] Chenglong Li, Shiyi Hu, Sihan Gao, and Jin Tang. Real-time grayscale-thermal tracking via laplacian sparse representation. In *International Conference on Multimedia Modeling*, 2016.

[20] Chenglong Li, Xinyan Liang, Yijuan Lu, Nan Zhao, and Jin Tang. Rgb-t object tracking: benchmark and baseline. *Pattern Recognition*, 2019.

[21] C. Li, X. Wu, N. Zhao, X. Cao, and J. Tang. Fusing two-stream convolutional neural networks for rgb-t object tracking. *IEEE Transactions on Information Theory*, 2018.

[22] Chenglong Li, Sun Xiang, Wang Xiao, Zhang Lei, and Tang Jin. Grayscale-thermal object tracking via multitask laplacian sparse representation. *IEEE Transactions on Systems Man and Cybernetics Systems*, 2017.

[23] Chenglong Li, Nan Zhao, Yijuan Lu, Chengli Zhu, and Jin Tang. Weighted sparse representation regularized graph learning for rgb-t object tracking. In *Proceedings of ACM International Conference on Multimedia*, 2017.

[24] C. Li, C. Zhu, Y. Huang, J. Tang, and L. Wang. Cross-modal ranking with soft consistency and noisy labels for robust rgbt tracking. In *Proceedings of European Conference on Computer Vision*, 2018.

[25] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[26] Yang Li and Jianke Zhu. A scale adaptive kernel correlation filter tracker with feature integration. In *Proceedings of IEEE European Conference on Computer Vision*, 2014.

[27] A. Lukezic, T. Vojir, L. C. Zajc, J. Matas, and M. Kristan. Discriminative correlation filter with channel and spatial reliability. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[28] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang. Hierarchical convolutional features for visual tracking. In *Proceedings of IEEE International Conference on Computer Vision*, 2015.

[29] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.

[30] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[31] Shi Pu, Yibing Song, Chao Ma, Honggang Zhang, and Ming-Hsuan Yang. Deep attentive tracking via reciprocative learning. In *Advances in Neural Information Processing Systems*, 2018.

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of International Conference on Learning Representations*, 2015.

[34] Fuchun Sun and Huaping Liu. Fusion tracking in color and infrared images using joint sparse representation. *Science China(Information Sciences)*, 2012.

[35] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr. End-to-end representation learning for correlation filter based tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[36] Yi Wu, Erik Blasch, Genshe Chen, Li Bai, and Haibing Ling. Multiple source data fusion via sparse representation for robust visual tracking. In *Proceedings of International Conference on Information Fusion*, 2011.

[37] J. Zhang, S. Ma, and S. Sclaroff. MEEM: robust tracking via multiple experts using entropy minimization. In *Proceedings of IEEE European Conference on Computer Vision*, 2014.

[38] Zhipeng Zhang and Houwen Peng. Deeper and wider siamese networks for real-time visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4591–4600, 2019.

[39] Yabin Zhu, Chenglong Li, Bin Luo, Jin Tang, and Xiao Wang. Dense feature aggregation and pruning for rgbt tracking. In *Proceedings of the ACM International Conference on Multimedia*, 2019.