Attribute-based Progressive Fusion Network for RGBT Tracking

Yun Xiao[1,2,4], **Mengmeng Yang**[3], **Chenglong Li**[1,2,4*], **Lei Liu**[3], **Jin Tang**[1,2,3]
[1]Information Materials and Intelligent Sensing Laboratory of Anhui Province, Hefei, China
[2]Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, Hefei, China
[3]School of Computer Science and Technology, Anhui University, Hefei, China
[4]School of Artificial Intelligence, Anhui University, Hefei, China
xiaoyun@ahu.edu.cn, xiuxiaoran@163.com, lcl1314@foxmail.com, liulei970507@163.com, tangjin@ahu.edu.cn

Abstract

RGBT tracking usually suffers from various challenging factors of fast motion, scale variation, illumination variation, thermal crossover and occlusion, to name a few. Existing works often study fusion models to solve all challenges simultaneously, which requires fusion models complex enough and training data large enough, and are usually difficult to be constructed in real-world scenarios. In this work, we disentangle the fusion process via the challenge attributes, and thus propose a novel Attribute-based Progressive Fusion Network (APFNet) to increase the fusion capacity with a small number of parameters while reducing the dependence on large-scale training data. In particular, we design five attribute-specific fusion branches to integrate RGB and thermal features under the challenges of thermal crossover, illumination variation, scale variation, occlusion and fast motion respectively. By disentangling the fusion process, we can use a small number of parameters for each branch to achieve robust fusion of different modalities and train each branch using the small training subset with the corresponding attribute annotation. Then, to adaptive fuse features of all branches, we design an aggregation fusion module based on SKNet. Finally, we also design an enhancement fusion transformer to strengthen the aggregated feature and modality-specific features. Experimental results on benchmark datasets demonstrate the effectiveness of our APFNet against other state-of-the-art methods. Code will be available at https://github.com/mmic-lcl/source-code.

Introduction

RGBT tracking is to use the complementary benefits of RGB and thermal infrared data to achieve robust visual tracking, which has various applications such as autonomous driving, surveillance security and robotics. RGBT tracking is a challenging task since it usually suffers from various factors, such as thermal crossover, illumination variation, scale variation, occlusion and fast motion.

Existing works often try studying various fusion models to solve all challenges simultaneously in RGBT tracking. Zhu et al. (2019) fuse features of RGB and thermal modalities in each layer and recursively aggregate these features in all layers. Li et al. (2019b) mine modality-shared and -specific information by a multi-adapter network which includes modality-shared and modality-specific feature learning modules in all layers. Zhang et al. (2019) design different fusion strategies in the DiMP tracking framework (Bhat et al. 2019), which requires a large-scale training dataset, including 9,335 videos with 1,403,359 frames in total. However, these methods need to either design complex fusion models or construct large-scale training data for all challenge factors, but the performance improvement is still limited since challenge factors are numerous in real-world scenarios. To relieve the burden of model design and data construction, Li et al. (2020) model the target appearance un-

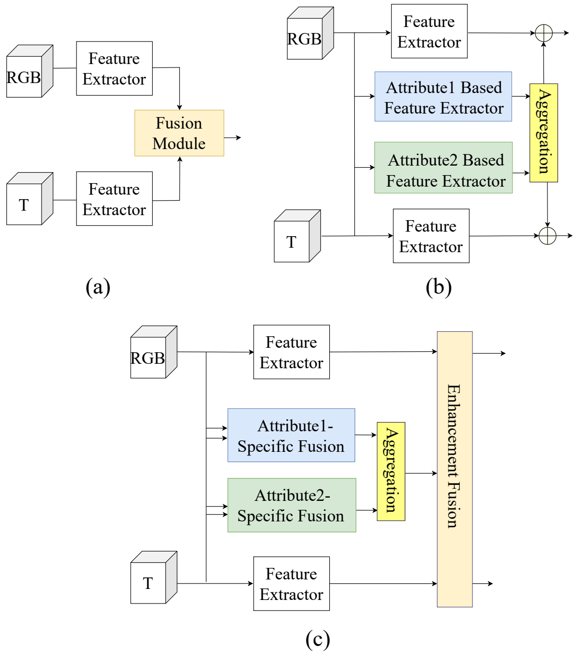

Figure 1: Advances of our fusion model over existing ones. (a) Common fusion models. (b) Attribute-aware fusion model (Li et al. 2020; Zhang et al. 2021a) which extracts appearance features under certain attributes and then perform feature fusion. (c) Our attribute-based progressive fusion model, which performs fusion under certain attributes and then aggregate and enhance them in a progressive manner.

[*]Chenglong Li is the corresponding author.

der different challenge attributes separately and then fuse them using an aggregation module. In this way, the target representations under certain attributes can be learnt by a few parameters even in the situation of insufficient training data. However, the fusion is performed in a simple manner, the capacity of the fusion model might thus be limited and the tracking performance would be degraded. Zhang et al. (2021a) design an complex attribute ensemble network for aggregating multiple attribute features better. However, neither of them considers the fusion method under specific attributes.

In this work, we disentangle the fusion process via the challenge attributes, and thus propose a novel Attribute-based Progressive Fusion Network (APFNet) to increase the fusion capacity with a small number of parameters while reducing the dependence on large-scale training data. Advances of our fusion model over existing ones as Fig. 1. In particular, we design the fusion branch for each attribute to learn the attribute-specific fusion parameters, then design an aggregation fusion module to integrate all fused features from each attribute-based fusion branch, and finally design an enhancement fusion transformer to enhance the aggregated feature and modality-specific features. We describe in detail as follows.

First, we disentangle the fusion process via five challenge attributes including thermal crossover (TC), illumination variation (IV), scale variation (SV), occlusion (OCC) and fast motion (FM) to achieve the attribute-specific fusion. Note that more attributes could be incorporated in our framework and we leave it in future work. For each attribute-specific fusion branch, we can employ a small number of parameters for model design since each branch just needs focusing on the feature fusion under a certain attribute. Moreover, these branches could be trained separately using small-scale training data with the corresponding attributes and the requirement of large-scale data to train fusion models is thus addressed.

Second, we design an attribute-based aggregation fusion module to adaptively aggregate all attribute-specific fusion features. Note that the attribute annotations are available in training stage but unavailable in testing stage, and thus we do not know which fusion branches should be activated in tracking process. To handle this problem, we design the aggregation model based on the SKNet (Li et al. 2019c) to adaptively select effective features from all fusion branches. Different from existing works (Qi et al. 2019; Li et al. 2020), the designed aggregation model could more effectively suppress noisy features from unappeared attributes by predicting the channel attention for each fusion features.

Finally, we design the enhancement fusion transformer to strengthen the aggregated feature and modality-specific features. Different from existing transformers (Vaswani et al. 2017; Wang et al. 2021; Chen et al. 2021), our enhancement fusion transformer equips three encoders and two decoders for self enhancement and interactive enhancement respectively. On one hand, we leverage three encoders to enhance aggregated and modality-specific features respectively. On the other hand, we use two decoders to interactively strengthen the above enhanced features.

We use a dual-stream hierarchical architecture to gradually integrate the modules of the attribute-based progressive fusion as shown in Fig. 2. In the training phase, we design a three-stage training algorithm to train our network effectively and efficiently. Experimental results on three benchmark datasets GTOT, RGBT234 and LasHeR demonstrate the effectiveness of the proposed APFNet against other state-of-the-art methods.

The contributions of this paper are summarized as follows.

- We propose a novel attribute-based progressive fusion network to increase the fusion capacity with a small number of parameters while reducing the dependence on large-scale training data in RGBT tracking.
- We design an attribute-specific fusion strategy to disentangle the fusion process via five challenge attributes. Each fusion branch only requires a small number of parameters which can be trained efficiently using small-scale training data, since it just needs focusing on the feature fusion under a certain attribute.
- We design an attribute-based aggregation fusion model to adaptively aggregate all attribute-specific fusion features. Although we do not know which fusion branches should be activated in tracking process, the model could effectively suppress noisy features from unappeared attributes via attention based weighting.
- We design an enhancement fusion transformer to further strengthen the aggregated feature and modality-specific features. The transformer consists of different numbers of encoders and decoders for self enhancement and interactive enhancement of aggregated and modality-specific features respectively.

Related Work

In this section, we give a briefly introduction about RGBT fusion tracking and transformer tracking.

RGBT Fusion Tracking

Recently, deep learning trackers have dominated this research field mainly benefit from powerful representation ability of CNN, and have significantly improved the tracking performance. Some of them focus on multi-modal information fusion for RGBT tracking. Zhu et al. (2020) fuse the features by calculating weights in both layer-wise and modality-wise to generate more discriminative features for RGBT tracking. To remove redundant and noisy features of target, Zhu et al. (2019) propose a recursive strategy to densely aggregate features in each modality and prune the aggregated features of all modalities in a cooperative manner. Zhang et al. (2021b) introduce both appearance and motion cues, and propose a switcher to switch the appearance and motion cues flexibly. To further analyze the effectiveness of multi-modal fusion, mfDiMP (Zhang et al. 2019) consider several fusion mechanisms including pixel-wise, feature-wise and response-wise fusions at different levels. Tu et al. (2021) extracts more robust feature representation through the strategy of dividing samples, and then designs

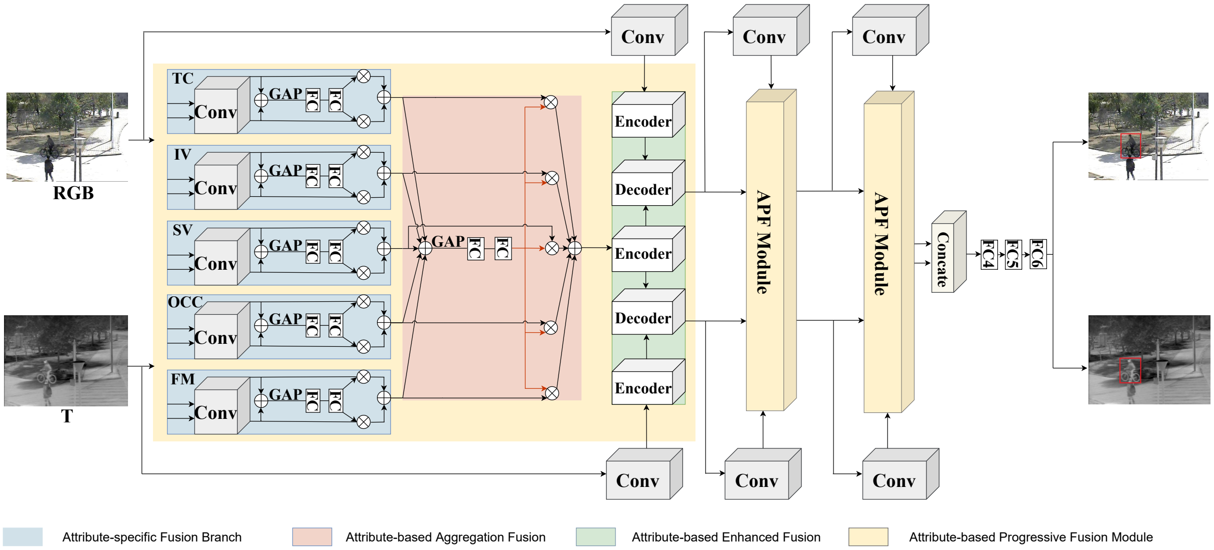
RGB
T
TC
IV
SV
OCC
FM
Conv
GAP
FC
Encoder
Decoder
APF Module
Concate
FC4
FC5
FC6
Attribute-specific Fusion Branch
Attribute-based Aggregation Fusion
Attribute-based Enhanced Fusion
Attribute-based Progressive Fusion Module

Figure 2: The structure of Attribute-based Progressive Fusion Network (APFNet). TC, IV, SV, OCC and FM represent five challenge attribute-specific branches including thermal crossover, illumination variation, scale variation, occlusion and fast movement. Herein, + and × denote the operation of element-wise addition and multiplication. GAP is the global average pooling.

an attention-based fusion module to fuse the features of the two modalities. Lu et al. (2021) design an instance adapter to predict modality weights for achieving quality-aware fusion of different modalities. But all of the above methods are just designing more complex models to solve all challenging situations. To make better use of training data, Li et al. (2020) mine modality-shared and modality-specific information with different challenges, and thus propose a challenge-aware network to enhance the discriminative ability of some weak modality, and have achieved excellent tracking performance. Furthermore, Zhang et al. (2021a) design a more complex aggregation module for aggregating different attributes better.

Transformer Tracking

The core of the transformer is the attention mechanism, which is first proposed to apply in the field of machine translation (Vaswani et al. 2017) and has achieved great success. Transformer has also been applied to visual tracking and achieved great success. Wang et al. (2021) use transformer to aggregate information from multiple template frames to distinguish the target. Chen et al. (2021) design a self-attention-based self-enhancement module and a cross-attention-based feature fusion module in the Siamese framework to focus on global information. Different from the above transformer models, we design an enhancement fusion transformer which consists of three encoders and two decoders to enhance aggregated feature and modality-specific features.

Attribute-based Progressive Fusion Tracking

In this part, we first present the architecture of the Attribute-based Progressive Fusion Network (APFNet), then introduce the three-stage training algorithm, and finally give the process of online tracking in details.

Attribute-based Progressive Fusion Network

Overview The overall architecture of APFNet is shown in Fig. 2, and the major component of APFNet is the attribute-based progressive fusion (APF) module which consists of five attribute-specific fusion branches, attribute-based aggregation fusion module and enhancement fusion transformer. In specific, we use the first three layers of the VGG-M (Simonyan and Zisserman 2015) as the backbone and expand it to a dual-stream architecture. In each layer, we embed APF module in order to gradually fuse the information of the two modalities. First, we input the visible and thermal images. The backbone network extracts modality-specific features separately, and the five branches perform attribute-specific fusion at the same time. Then we send all attribute-specific fused features to the attribute-based aggregation module to obtain robust aggregated feature. Next, the two modality-specific features and aggregated feature are respectively sent to the enhancement fusion transformer to form a more robust feature representation, which is used as the input in the next convolutional layer and APF module. After the last APF module, three fully connected layers are used to extract global features for classification and regression (Nam and Han 2016). We present the details of three components of APF module and our training method below.

Attribute-specific Fusion Branches We first disentangle the fusion process via five challenge attributes including thermal crossover (TC), illumination variation (IV), scale variation (SV), occlusion (OCC) and fast motion (FM) to achieve the attribute-specific fusion. In most of existing datasets such as GTOT (Li et al. 2016) and RGBT234 (Li et al. 2019a), each attribute is manually annotated for each video frame. It supports us to train each attribute-specific fusion branch individually. For each attribute-specific fusion branch, we can employ a small number of parameters for model design since each branch just needs focusing on the feature fusion under a certain attribute. Moreover, these branches could be trained separately using small-scale training data with the corresponding attributes and the requirement of large-scale data to train fusion models is thus addressed.

For simplicity, we set the fusion branches under different attributes to the same structure. In specific, for each branch, we first extract features from two modalities using a convolutional layer with the kernel size of 5×5, a rectified linear unit (ReLU) and a convolutional layer with the kernel size of 4×4. Then, we use the SKNet (Li et al. 2019c) to adaptively select the channel-wise features from both modalities. The details are also shown in Fig. 2.

Aggregation Fusion Module The attribute annotations are available in training stage but unavailable in testing stage, and thus we do not know which fusion branches should be activated in tracking process. To handle this problem, we design an attribute-based aggregation fusion module to adaptively aggregate all attribute-specific fusion features effectively. In particular, we employ the SKNet to adaptively select effective features from all attribute-specific fusion branches. In specific, we first send the features from each attribute-specific fusion branch to the aggregation layer to generate channel-wise weights. Then, we perform weighting operation on these five branch features to obtain more robust aggregated feature. The details are also shown in Fig. 2.

Enhancement Fusion Transformer Some methods (Chen et al. 2021; Wang et al. 2021) employ the transformer in the Siamese structure to fuse the features of the template frame and the search frame for visual tracking. However, these works use single encoder and decoder to model the relation between template and search frames and thus do not achieve multiple self enhancements and interactive enhancements. To handle this problem, we propose an enhancement fusion transformer which consists of three encoders and two decoders to perform self enhancements and interactive enhancements of aggregated and modality-specific features respectively.

To achieve this goal, we separate the encoder and the decoder in the original transformer (Vaswani et al. 2017). Then, we use three separated encoders to self enhance the aggregated feature outputted from the aggregation fusion module and two modality-specific features outputted from the convolutional layer while using two separated decoders to interactively further enhance these encoding features. To reduce the complexity of the model, we use the single-head attention mechanism and the k and v matrices share weights in both encoders and decoders. Fig. 3 shows the details of the separated encoder and decoder.

Let $\boldsymbol{X}^{i}_{agg}$ be the aggregated feature computed from aggregation fusion module of the i-th layer, the visible feature $\boldsymbol{X}^{i}_{vis}$ and thermal infrared feature $\boldsymbol{X}^{i}_{inf}$ be the modality-specific features extracting from two modalities by the i-th convolutional layer in backbone. Each input feature is transformed into three vectors including query, key and value through linear layer mapping. The attention weight matrix is generated by query $\boldsymbol{q}$ and key $\boldsymbol{k}$, and multiplied by the value $\boldsymbol{v}$ for the i-th layer. Finally, $\boldsymbol{v}$ is added to the original feature vector with the residual. $\boldsymbol{X}^{e,i}_{vis}$ and $\boldsymbol{X}^{e,i}_{inf}$ are the visible feature and thermal infrared feature after the the i-th encoder self enhancement, separately. We show the details in the left in Fig. 3. In the i-th encoding phase, $\boldsymbol{X}^{i}_{vis}$, $\boldsymbol{X}^{i}_{inf}$ and $\boldsymbol{X}^{i}_{agg}$ are inputted to three encoders for self enhancement, and then compute the outputs denoted as $\boldsymbol{X}^{e,i}_{vis}$, $\boldsymbol{X}^{e,i}_{inf}$ and $\boldsymbol{X}^{e,i}_{agg}$ respectively. The self enhancement encoders with aggregated feature and modality-specific features can be described as follows:

$$\boldsymbol{X}^{e,i}_{vis} = \boldsymbol{Encoder}(\boldsymbol{X}^{i}_{vis}) \in \mathbb{R}^{C\times H\times W}$$

$$\boldsymbol{X}^{e,i}_{inf} = \boldsymbol{Encoder}(\boldsymbol{X}^{i}_{inf}) \in \mathbb{R}^{C\times H\times W}$$

$$\boldsymbol{X}^{e,i}_{agg} = \boldsymbol{Encoder}(\boldsymbol{X}^{i}_{agg}) \in \mathbb{R}^{C\times H\times W}$$

where $\boldsymbol{X}^{i}_{vis}, \boldsymbol{X}^{i}_{inf}, \boldsymbol{X}^{i}_{agg} \in \mathbb{R}^{C\times H\times W}$. C, H and W indicate the channel number, height and width of the feature matrix.

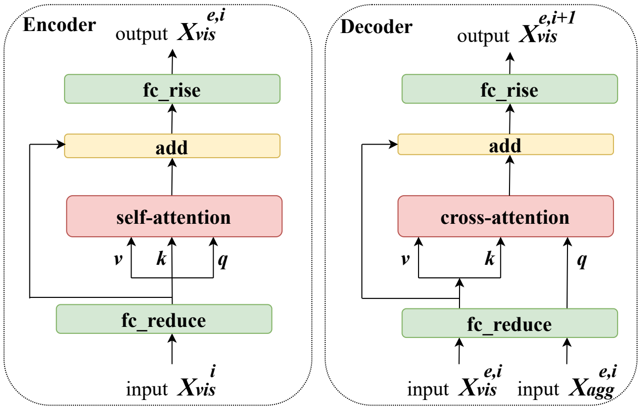

Figure 3: Detailed structure of the separated encoder and decoder in the enhancement fusion transformer. The left is the structure of the encoder and the right is the structure of the decoder.

The decoder is used for interactive enhancement of the i-th encoded aggregated feature ($\boldsymbol{X}^{e,i}_{agg}$) and the i-th modality-specific features ($\boldsymbol{X}^{e,i}_{vis}$ and $\boldsymbol{X}^{e,i}_{inf}$). We show the details in the right in Fig. 3. In the decoding phase, the left input is $\boldsymbol{X}^{e,i}_{vis}$ or $\boldsymbol{X}^{e,i}_{inf}$, and the right one is $\boldsymbol{X}^{e,i}_{agg}$ using as the auxiliary information, and then obtain the output as the input of the next layer as $\boldsymbol{X}^{e,i+1}_{vis}$ or $\boldsymbol{X}^{e,i+1}_{inf}$. The decoding process can be described as follows:

$$\boldsymbol{X}^{e,i+1}_{vis} = \boldsymbol{Decoder}(\boldsymbol{X}^{e,i}_{vis}, \boldsymbol{X}^{e,i}_{agg}) \in \mathbb{R}^{C\times H\times W}$$

$$\boldsymbol{X}^{e,i+1}_{inf} = \boldsymbol{Decoder}(\boldsymbol{X}^{e,i}_{inf}, \boldsymbol{X}^{e,i}_{agg}) \in \mathbb{R}^{C\times H\times W}$$

Dual-stream Hierarchical Architecture We use a dual-stream network (Li et al. 2018) as the backbone to extract the features of RGB and thermal infrared images separately with accounting for the difference of multiple modalities. The backbone is a lightweight network which is borrowed from the first three convolution layers of VGG-M, and the kernel sizes of the three convolutional layers are 7×7, 5×5 and 3×3 respectively. The initial parameters in the convolution kernel are gained from the model pre-trained on imageNet-vid (Nam and Han 2016). To achieve sufficient fusion of different modalities, we add the attribute-based progressive fusion(APF) module into all layers in the backbone and thus get a hierarchical architecture. Finally, we connect three fully connected layers after the last convolutional layer, in which the last fully connected layer FC6 is adapted to different domains that is similar to MDNet (Nam and Han 2016).

Three-stage Training Algorithm

There are three issues need to be considered during the training process. First, if we train the network with all training data directly, the loss of training data with any attributes will be propagated backward to all attribute-specific fusion branches. Second, attribute annotations are not available in the testing phase and we do not know what attributes will be appeared in a frame during the tracking process. Finally, we want to enhance the features under the appeared attributes while suppress the features under the unappeared attributes. To handle these issues, we design a three-stage training method as follows to train the network effectively and efficiently.

Train all Attribute-specific Fusion Branches At the first stage, each attribute-specific fusion branch is trained individually. All aggregation fusion modules are moved and the training data with specific attributes are applied to train corresponding attribute-specific fusion branches one by one. Specifically, the dual-stream CNN is firstly initialized through the pre-trained model parameters on imageNet-vid (Nam and Han 2016), which includes three convolutional layers and two fully connected layers FC4 and FC5. Then, we initialize the parameters of all attribute-specific fusion branches and add the new classification branches FC6. The learning rates are set to 0.001 and 0.0005 for attribute-specific fusion branches and FC6, respectively. Since the data under illumination variation is very small, the learning rate under this attribute-specific fusion branch is 0.002. The stochastic gradient descent (SGD) method is adopted as the optimization strategy with momentum of 0.9, and the weight attenuation is set to 0.0005. The number of training periods is 200. At this stage, we only save the parameters of the attribute-specific fusion branches to eliminate the influence of the FC layer.

Train Aggregation Fusion Module In the second stage, we fix the previously trained attribute-specific branches and only train the aggregation fusion modules using all training data. We initialize the parameters of the aggregation fusion modules and the fully connected layers FC6 randomly, and set the learning rates to 0.001 and 0.0005 respectively. The number of training periods is 500. The other settings are the same as the first stage. At this stage, we save the parameters of aggregation fusion modules, FC4 and FC5.

Train Attribute-based Enhancement Module Finally, we train the enhancement fusion transformer modules using all training data and fine-tune other modules. We initialize the parameters of the enhancement fusion transformer modules and the fully connected layers FC6 randomly, and we set the learning rates to 0.001 for transformer, 0.0005 for FC6 and 0.0001 for other modules in the network. The number of training periods is 1000. The other settings are the same as the first stage.At this stage, we save the parameters of the whole model.

Online Tracking with APFNet

For each new video sequence, we initialize a new FC layer (FC6) branch during the tracking process randomly, then fix all the model parameters that were trained before, and fine-tune FC4, FC5 and FC6 to adapt to various appearance variations during the tracking process. In the first frame, 500 positive samples and 5000 negative samples are collected firstly for the given initial target to fine-tune the fully connected layers. Here we define that the samples whose IoU with the ground truth is larger than 0.7 as positive, and smaller than 0.5 as negative. To make the tracking results more accurate, 1000 samples are collected in the first frame to train the regressor to adapt to the new target domain. The tracking result of previous frame are used to sample 256 candidate samples and sent to APFNet for tracking in the current frame. We select the five candidate samples with the highest scores and take the average of their bounding boxes as the tracking results of the current frame. When the score of the tracking result is greater than 0, the tracking is considered to be successful and the regressor is used to adjust the result for more accurate localization. 20 positive samples and 100 negative samples are collected to update the network later dynamically. Note that APFNet is updated every 10 frames, and once our tracker fails (i.e., the tracking score is lower than 0), we update the model immediately. More details can be referred to MDNet (Nam and Han 2016).

Experiments

In this section, we evaluate our algorithm by comparing the tracking performance with some state-of-the-art trackers on three RGBT tracking benchmarks including GTOT (Li et al. 2016), RGBT234 (Li et al. 2019a) and LasHeR (Li et al. 2021) to validate the effectiveness of proposed method and analyze effectiveness of each major component in the algorithm. In this part, we will introduce the details of the datasets, the evaluation metrics, and the implementation details of training and testing. Our tracker is implemented in pytorch 1.0.1, python 3.7, CUDA 10.2 and runs on a computer with 8 NVIDIA GeForce RTX 1080Ti GPU cards.

Evaluation Data and Metrics

GTOT dataset includes 50 RGBT video pairs registered in time and space under different scenes and conditions, with a total of about 15K frames. The entire dataset is divided into 7 subsets according to the different attributes for analyzing the sensitivity of the RGBT tracker in different challenges. We adopt the precision rate (PR) and success rate (SR) in the one-pass evaluation (OPE) as evaluation metrics for quantitative performance evaluation. Herein, PR measures the percentage of all frames whose distance between the center point of the tracking result and ground-truth is less than threshold, and we compute the representative PR score by setting the threshold to be 5 pixels in GTOT datasets and 20 pixels in other datasets. SR measures the percentage of successfully tracked frames whose overlaps are larger than thresholds, and we calculate the representative SR score by the area under the curve. **RGBT234** is a larger RGBT tracking dataset than GTOT, which is extended from the RGBT210 (Li et al. 2017) dataset and provides a more accurate annotations that takes into full consideration of various environmental challenges. Contains 234 RGBT highly aligned video pairs with about 234K frames in total, and 12 attributes are annotated to facilitate analyzing the effectiveness of different tracking algorithms for different challenges. **LasHeR** is the largest RGBT tracking dataset at present, which contains 1224 aligned video sequences including more diverse attribute annotations, in which 245 sequence are divided separately as testing datasets, and the remaining are designed for training datasets.

Implementation Details

During the experiments, for the testing on GTOT dataset, we train our attribute-specific fusion branches with corresponding challenge-based training data extracted from RGBT234 dataset by challenge labels. Then, we use the entire dataset of RGBT234 to train the attribute-based aggregation SKNet and the enhancement fusion transformer module. While for the testing on RGBT234 and LasHeR datasets, we exchange training and test sets, in other words, our training dataset is GTOT, and training process is similar to the mentioned above.

Quantitative Comparison

We test our Attribute-based Progressive Fusion Network (APFNet) on three popular RGBT tracking benchmarks and compare performance with some state-of-the-art trackers, such as ADRNet (Zhang et al. 2021a), MANet++ (Lu et al. 2021), HDINet (Mei et al. 2021), M^{5}L (Tu et al. 2021), CMPP (Wang et al. 2020), JMMAC (Zhang et al. 2021b), CAT (Li et al. 2020), MANet (Li et al. 2019b), DAFNet (Gao et al. 2019), mfDiMP (Zhang et al. 2019), DAPNet (Zhu et al. 2019), SGT (Li et al. 2017), FANet (Zhu et al. 2020), MaCNet (Zhang et al. 2020) and MDNet (Nam and Han 2016)+RGBT, to validate the effectiveness of proposed method.

Evaluation on GTOT Dataset Comparison results on GTOT dataset are shown in Fig. 4. We can find that our APFNet outperforms the top traditional algorithms SGT (Li et al. 2017) with performance gains 5.4%/10.9% in PR/SR respectively. Moreover, we advance our baseline tracker MDNet+RGBT (Nam and Han 2016) with a large margin, $i.e.$ 10.5%/10.0% in PR/SR respectively. Although comparing with the state-of-the-art method CMPP (Wang et al. 2020), our PR/SR is lower 2.1%/0.1% respectively. But compared with CAT (Li et al. 2020), a tracker that also mines challenge information, we advance it with 1.6%/2.0% in PR/SR. Moreover, compared with ADRNet (Zhang et al. 2021a), we have also achieved comparable results. These results show the effectiveness of APFNet.

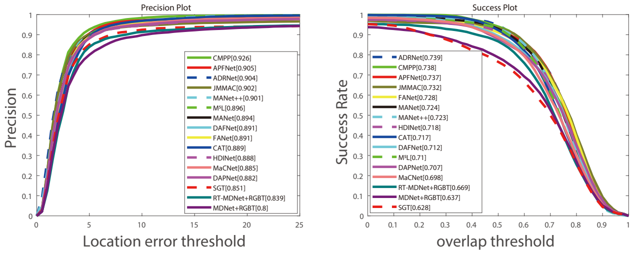

Figure 4: The evaluation curve on GTOT dataset. The representative scores of PR/SR is presented in the legend.

Evaluation on RGBT234 Dataset As can be seen from Fig. 5, our algorithm achieves the best tracking performance on the RGBT234 dataset in all state-of-the-art trackers. Compared with the baseline tracker MDNet+RGBT, our method achieves significant improvement in PR/SR, $i.e.$ 10.5%/8.4%. Besides, compared with CMPP that is the top advanced trackers in this dataset, our tracker outperforms with 0.4%/0.4% in PR/SR respectively. While compared with CAT, we have achieved a huge performance improvement of 2.3%/1.8% in PR/SR. In addition, compared with ADRNet, we exceed 2.0%/0.9% in PR/SR. These results fully demonstrate the effectiveness of our method.

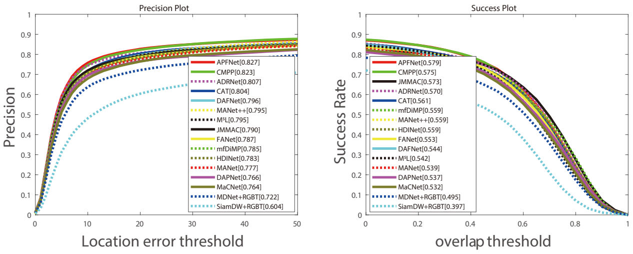

Figure 5: The evaluation curve on RGBT234 dataset. The representative scores of PR/SR is presented in the legend.

Evaluation on LasHeR Dataset The evaluation on LasHeR testing set are shown in Fig. 6. We can find that our tracker has achieved excellent performance compared with some start-of-the-art trackers. Compared with mfDiMP (Zhang et al. 2019), which is the champion of VOT2019-RGBT, our PR/SR is 5.3%/1.9% higher than it. While compared with MaCNet and CAT, we also advance them with 1.8%/1.2% and 5.0%/4.8% respectively in PR/SR, which prove the huge performance advantage of our method. It is noted that we only use GTOT dataset training to achieve the best results on RGBT 234 and LasHeR, which shows that our model does not depend on large-scale training data.

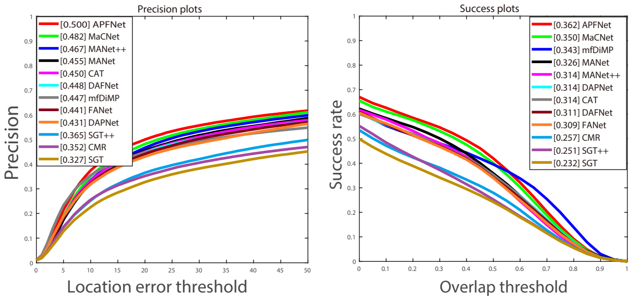

Figure 6: The evaluation curve on LasHeR testing set. The representative scores of PR/SR is presented in the legend.

Ablation Study

To validate the effectiveness of major components in our method, we carry out the ablation study on the GTOT and RGBT234.

Analysis of the APFNet In order to verify the effectiveness of the designed network, we make the following variants of the network: 1. APFNet-SKNet, that all challenge branches are aggregated by element addition, removing the structure of SKNet and transformer, to verify the effectiveness of the proposed aggregation method. 2. APFNet-transformer, that all challenge branches are aggregated by SKNet, removing the proposed transformer structure, to verify the effectiveness of the transformer enhancement module. The experimental results on GTOT and RGBT234 dataset are shown in Table 1.

Table 1: PR/SR scores of different variants induced from our method on GTOT and RGBT234 dataset for verify the effectiveness of our tracker.

		APFNet -SKNet	APFNet -transformer	APFNet
GTOT	PR	88.7	89.9	**90.5**
	SR	71.9	72.9	**73.7**
RGBT234	PR	80.4	81.9	**82.7**
	SR	56.0	56.8	**57.9**

From the result, we can conclude that: 1) APFNet-SKNet is some worse than APFNet-transformer and APFNet, proving that the necessity of introducing aggregation module for challenge branches. 2) The performance of APFNet is significantly better than APFNet-transformer, which verifies the effectiveness of the proposed transformer module for aggregating multi-challenge features.

Evaluation of Attribute Branches We quantitatively analyze the effectiveness of the challenge branches to verify that each challenge branch is specific and solves the corresponding challenge. In this part, we select five typical sequences from GTOT and RGBT234 dataset, and each sequence has a corresponding main challenge (Fan et al. 2021; Qi et al. 2019). On the basis of MDNet+RGBT, we only add a specific challenge fusion branch to test each sequence. The Correlative results on GTOT and RGBT234 dataset are shown in Table 2 and Table 3 respectively. The experimental results show that each challenge fusion branch has achieved best performance on sequence with the same challenge as the mainstay compared with other challenge fusion branches, which verifies the effectiveness of the challenge fusion branches.

Table 2: Tracking results in terms of success rate on five GTOT sequences (A:CarNig, B:Minibus, C:OCCCar-2, D:Otcbvs, E:FastMotoNig). We have marked the dominant attribute of the video after the name of each video sequence. We only add each attribute-specific fusion branch to the backbone to get the result.

	IV- Branch	TC- Branch	OCC- Branch	SC- Branch	FM- Branch
A:IV	**41.2**	33.6	37.2	40.5	33.8
B:TC	65.9	**67.5**	64.8	65.9	63.3
C:OCC	68.4	67.0	**68.7**	65.7	65.7
D:SC	75.8	71.3	73.1	**76.9**	71.2
E:FM	71.9	71.8	73.9	73.2	**58.3**

Table 3: Tracking results in terms of success rate on five RGBT234 sequences (A:elecbike, B:blackwoman, C:man24, D:bike, E:night2). We use the same strategy to get the result on the RGBT234 dataset.

	IV- Branch	TC- Branch	OCC- Branch	SC- Branch	FM- Branch
A:IV	**63.8**	22.9	23.0	62.4	43.7
B:TC	69.5	**72.3**	69.5	69.2	69.5
C:OCC	26.1	19.7	**30.9**	18.6	19.1
D:SC	73.6	77.6	71.3	**79.2**	75.7
E:FM	40.9	32.5	35.6	19.5	**58.3**

Conclusion

In this paper, we propose an Attribute-based Progressive Fusion Network (APFNet) in order to make full use of the information between multi-model challenge attributes. We have designed attribute-specific fusion branches for each attribute to learn the different fusion parameters, and use attribute-based aggregation fusion module to aggregate a variety of attribute feature. Finally, enhancement fusion transformer is introduced to enhance aggregated feature and modality-specific features. Extensive experiments on three benchmark datasets demonstrate the effectiveness of our method against the state-of-the-art trackers. In the future, we will explore more fusion structures under more challenges to fully explore the information between multi-model attributes.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 62006002), the Natural Science Foundation of Anhui Province (No. 2008085QF329, No. 1908085QF264), the University Synergy Innovation Program of Anhui Province (No. GXXT-2020–051), and the Anhui Energy Internet Joint Fund Project (No. 2008085UD07).

References

Bhat, G.; Danelljan, M.; Gool, L. V.; and Timofte, R. 2019. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, 6182–6191.

Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; and Lu, H. 2021. Transformer tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8126–8135.

Fan, H.; Yang, F.; Chu, P.; Lin, Y.; Yuan, L.; and Ling, H. 2021. TracKlinic: Diagnosis of challenge factors in visual tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 970–979.

Gao, Y.; Li, C.; Zhu, Y.; Tang, J.; He, T.; and Wang, F. 2019. Deep adaptive fusion network for high performance RGBT tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 91–99.

Li, C.; Cheng, H.; Hu, S.; Liu, X.; Tang, J.; and Lin, L. 2016. Learning collaborative sparse representation for grayscale-thermal tracking. *IEEE Transactions on Image Processing*, 25(12): 5743–5756.

Li, C.; Liang, X.; Lu, Y.; Zhao, N.; and Tang, J. 2019a. RGB-T object tracking: Benchmark and baseline. *Pattern Recognition*, 96: 106977.

Li, C.; Liu, L.; Lu, A.; Ji, Q.; and Tang, J. 2020. Challenge-aware RGBT tracking. In *Proceedings of the European Conference on Computer Vision*, 222–237. Springer.

Li, C.; Lu, A.; Zheng, A.; Tu, Z.; and Tang, J. 2019b. Multi-adapter RGBT tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2262–2270.

Li, C.; Wu, X.; Zhao, N.; Cao, X.; and Tang, J. 2018. Fusing two-stream convolutional neural networks for RGB-T object tracking. *Neurocomputing*, 281: 78–85.

Li, C.; Xue, W.; Jia, Y.; Qu, Z.; Luo, B.; Tang, J.; and Sun, D. 2021. LasHeR: A large-scale high-diversity benchmark for RGBT tracking. *IEEE Transactions on Image Processing*, 31: 392–404.

Li, C.; Zhao, N.; Lu, Y.; Zhu, C.; and Tang, J. 2017. Weighted sparse representation regularized graph learning for RGB-T object tracking. In *Proceedings of the ACM International Conference on Multimedia*, 1856–1864.

Li, X.; Wang, W.; Hu, X.; and Yang, J. 2019c. Selective kernel networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 510–519.

Lu, A.; Li, C.; Yan, Y.; Tang, J.; and Luo, B. 2021. RGBT Tracking via Multi-Adapter Network with Hierarchical Divergence Loss. *IEEE Transactions on Image Processing*.

Mei, J.; Zhou, D.; Cao, J.; Nie, R.; and Guo, Y. 2021. HDINet: Hierarchical dual-sensor interaction network for RGBT tracking. *IEEE Sensors Journal*.

Nam, H.; and Han, B. 2016. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4293–4302.

Qi, Y.; Zhang, S.; Zhang, W.; Su, L.; Huang, Q.; and Yang, M.-H. 2019. Learning attribute-specific representations for visual tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 8835–8842.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 1–14.

Tu, Z.; Lin, C.; Zhao, W.; Li, C.; and Tang, J. 2021. M5L: Multi-modal multi-margin metric learning for RGBT tracking. *IEEE Transactions on Image Processing*, 31: 85–98.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Proceedings of the Advances in Neural Information Processing Systems*, 5998–6008.

Wang, C.; Xu, C.; Cui, Z.; Zhou, L.; Zhang, T.; Zhang, X.; and Yang, J. 2020. Cross-modal pattern-propagation for RGB-T tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7062–7071.

Wang, N.; Zhou, W.; Wang, J.; and Li, H. 2021. Transformer meets tracker: exploiting temporal context for robust visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1571–1580.

Zhang, H.; Zhang, L.; Zhuo, L.; and Zhang, J. 2020. Object tracking in RGB-T videos using modal-aware attention network and competitive learning. *Sensors*, 20(2): 393.

Zhang, L.; Danelljan, M.; Gonzalez-Garcia, A.; van de Weijer, J.; and Shahbaz Khan, F. 2019. Multi-modal fusion for end-to-end RGB-T tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2252–2261.

Zhang, P.; Wang, D.; Lu, H.; and Yang, X. 2021a. Learning adaptive attribute-driven representation for real-time RGB-T tracking. *International Journal of Computer Vision*, 129(9): 2714–2729.

Zhang, P.; Zhao, J.; Bo, C.; Wang, D.; Lu, H.; and Yang, X. 2021b. Jointly modeling motion and appearance cues for robust RGB-T tracking. *IEEE Transactions on Image Processing*, 30: 3335–3347.

Zhu, Y.; Li, C.; Luo, B.; Tang, J.; and Wang, X. 2019. Dense feature aggregation and pruning for RGBT tracking. In *Proceedings of the ACM International Conference on Multimedia*, 465–472.

Zhu, Y.; Li, C.; Tang, J.; and Luo, B. 2020. Quality-aware feature aggregation network for robust rgbt tracking. *IEEE Transactions on Intelligent Vehicles*, 6(1): 121–130.