Duality-Gated Mutual Condition Network for RGBT Tracking

Andong Lu, Cun Qian, Chenglong Li, Jin Tang, and Liang Wang

***Abstract*—Low-quality modalities contain not only a lot of noisy information but also some discriminative features in RGBT tracking. However, the potentials of low-quality modalities are not well explored in existing RGBT tracking algorithms. In this work, we propose a novel duality-gated mutual condition network to fully exploit the discriminative information of all modalities while suppressing the effects of data noise. In specific, we design a mutual condition module, which takes the discriminative information of a modality as the condition to guide feature learning of target appearance in another modality. Such module can effectively enhance target representations of all modalities even in the presence of low-quality modalities. To improve the quality of conditions and further reduce data noise, we propose a duality-gated mechanism and integrate it into the mutual condition module. To deal with the tracking failure caused by sudden camera motion, which often occurs in RGBT tracking, we design a resampling strategy based on optical flow algorithms. It does not increase much computational cost since we perform optical flow calculation only when the model prediction is unreliable and then execute resampling when the sudden camera motion is detected. Extensive experiments on four RGBT tracking benchmark datasets show that our method performs favorably against the state-of-the-art tracking algorithms.**

***Index Terms*—RGBT tracking, Gated scheme, Conditional learning, Bidirectional feature modulation.**

I. Introduction

RGBT tracking, a popular research stream of visual tracking, aims at estimating the states of target object in a RGBT sequence given the initial ground truth bounding box in the first frame pair. Benefiting from the strong complementary advantages of RGB and thermal infrared data, RGBT trackers could work well in all-day and all-weather conditions. Therefore, RGBT tracking receives more and more attentions and has achieved astonishing progress in recent years [1], [2], [3], [4], [5], [6], [7], [8], [9], [10].

Recent RGBT tracking methods mainly study how to effectively fuse RGB and thermal modalities. One aspect is to introduce modality weights for adaptive fusion of different modalities. For example, Zhang *et al.* [7] propose modality-aware attention network to generate modality weights for adaptive fusion of different modalities with competitive learning. Zhu *et al.* [11] propose a quality-aware feature aggregation network, which models each modality separately and then integrate different modality features by learning modality weights that represent qualities of different modalities. Another one aspect is to learn powerful features of each modality and then fuse them by ad-hoc ways. For example, Zhang *et al.* [6] propose a two-stream network structure and use a lager-scale generated RGBT dataset to train the network to learn the characteristics of each modality. Li *et al.* [12] introduce three types of adapters to capture modality-specific, modality-shared and instance-aware target representations.

However, all these methods do not explore the potentials of low-quality modalities well, which play a critical role in boosting RGBT tracking performance. An example is shown in Fig. 1. In the top row, one can see that each RGBT image pair contains a lot of noisy information and the quality of one modality in each pair is extremely low. The feature maps generated by a typical method MANet [12] are shown in the second row, and we can find that these features suffer from the effects of a lot of noises especially in the low-quality modalities. These noise effects would degrade tracking accuracy and robustness. In addition, we observe that low-quality modalities usually contains some discriminative features which are useful for target localization, as shown in the second row of Fig. 1. Therefore, simple suppression or removal of low-quality modalities can not fully explore the potentials of multi-source data.

To handle these issues, we propose a novel Duality-gated Mutual Condition Network (DMCNet) to fully exploit the discriminative information of all modalities while suppressing the effects of data noise. In real-world scenarios, some modalities are sometimes unreliable due to the existence of adverse environments like total darkness, bad weathers and thermal crossover. To make the full use of potentials of these modalities while suppressing the effects of data noise, we design a mutual condition module, which takes the discriminative information of a modality as the condition to guide feature learning of target appearance in another modality. Moreover, we enhance feature representations of target appearance through a multi-scale convolutional layer and integrate it into the mutual condition module. Because we use features of one modality as the condition of the other modality, and some noises are thus inevitably included in conditions. Meanwhile, the features guided by conditions also contain noisy information. To improve the quality of conditions and further reduce feature noise, we propose a duality-gated mechanism. Fig. 1 presents some examples to verify the effectiveness of the proposed duality-gated mutual condition network. As shown in the third row, noises in modalities (especially for low-quality modalities) are significantly suppressed and discriminative abilities of target features are greatly boosted.

In addition, we find that the tracking performance is easily affected by the challenge of sudden camera motion, which frequently occurs in RGBT tracking task. The major reason

The authors are with Anhui University.

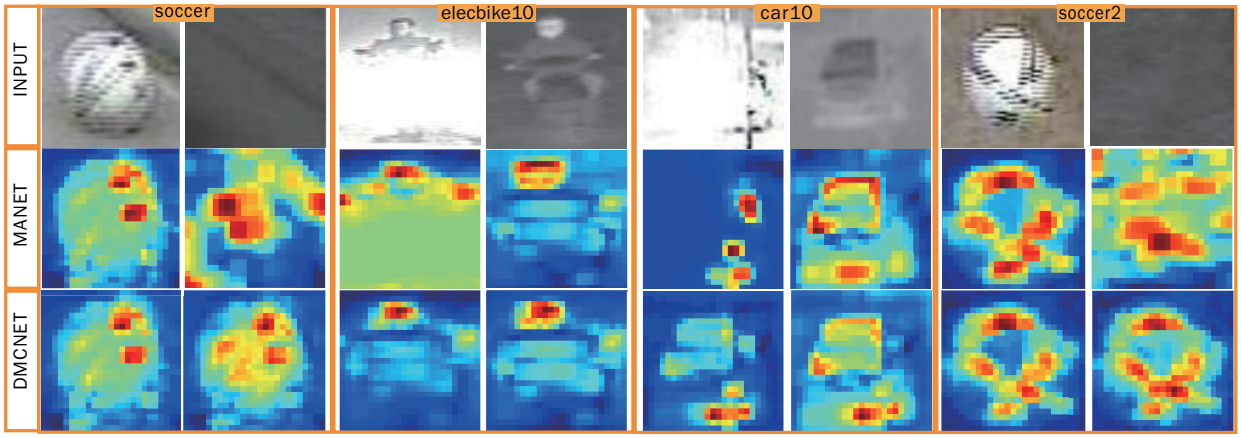


Fig. 1: Illustration of the effectiveness of our duality-gated mutual condition network (DMCNet) against a typical tracker MANet [12] on four examples in **soccer**, **elecbike10** , **car10** and **soccer2** respectively. Herein, the top row represents the input RGB and thermal frames, and the second row denotes the feature maps of the baseline method MANet, which does not include the duality-gated mutually conditional modules. The third row indicates the feature maps of our DMCNet.

is that under such challenge search windows are hardly cover target objects, which would lead to tracking failure. Common attempts are to expand search region [13] and perform global search [14], but these methods bring more background information and thus increase the risk of model drift. Meanwhile, the computational cost is usually greatly increased. To deal with this problem, we develop a simple yet effective resampling method based on a fast optical flow algorithm, DisFlow [15]. By comparing with a predefined threshold, we can judge whether the sudden camera motion occurs or not. If occurs, we resample candidate target regions along the direction and magnitude of camera motion. Note that our resampling method does not increase computational cost much since we execute it only when the tracking failure caused by sudden camera motion is detected and the optical flow computation is only performed on the local regions around target objects.

The major contributions of this paper are summarized as follows.

- We propose an effective approach to handle low-quality modalities in RGBT tracking. The approach is able to enhance the discriminative ability and suppress the effects of data noise of low-quality modalities and thus achieves large improvements in tracking accuracy and robustness.
- We design a duality-gated mutual condition module to take the discriminative information of a modality as the condition to adaptively guide feature learning of target appearance in another modality.
- We develop a simple yet effective resampling mechanism to deal with the tracking failure caused by the challenge of sudden camera motion, with a modest impact on tracking speed.
- Extensive experiments on four RGBT tracking benchmark datasets, including GTOT [16], RGBT210 [1], VOT2019-RGBT [17] and RGBT234 [18] are conducted. The results show that our tracking approach achieves the outstanding performance comparing with the state-of-the-art methods.

II. Related Work

A. RGBT Tracking Methods

In recent years, more and more RGBT tracking algorithms has proposed [16], [19], [1], [2], [20], [4], [6], [10], [8], [9], making this field remarkable development. Recent works [19], [16], [2] propose to learn modality weights to guide adaptive fusion of RGB and thermal modalities via reconstruction residues or classification scores. However, these methods are susceptible to interference from low-quality modal information, and unreliable reconstruction residues or classification scores would lead to inaccurate weight computation. Some recent studies [1], [20], [4] are focus on learning robust feature representations of RGB and thermal modalities. A graph learning approach is proposed by Li *et al.* [1] that constructs a patch-based weighted RGBT feature descriptor, and performs online tracking using the structured SVM. Several improvements [1], [20], [4] are made in this research stream. However, these methods are based on handcrafted features and their performance is easily affected by challenging factors. Zhu *et al.* [11] propose a quality-aware feature aggregation network to integrate different modal features by learning modality weights that represent qualities of different modalities. Zhang et al. [6] use different levels of fusion strategies to integrate the information of RGB and thermal modalities adaptively in an end-to-end deep learning framework. Li et al. [12] propose a multi-adapter neural network (MANet) for RGBT tracking, which achieves outstanding performance. Wang et al. [8] propose a cross-modal pattern-propagation tracking method to model intra-modal paired pattern-affinities, which reveal the latent cues between heterogeneous modalities. Li et al. [10] propose a challenge-aware network to model the representation of modality-shared and modality-specific challenges. Zhang et al. [9] employ a later fusion network that combines with motion tracker to jointly model appearance and motion cues for RGBT tracking. However, these methods do not explore the potentials well of low-quality modalities, which play a critical role in feature enhancement and noise reduction.

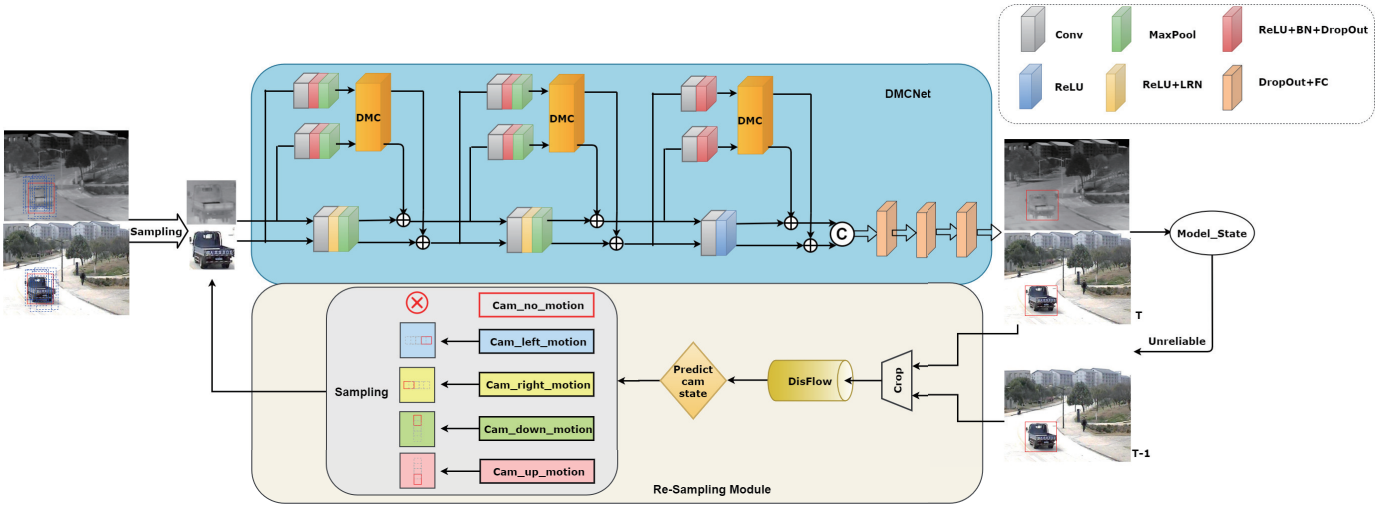


Fig. 2: Overall network architecture of our DMCNet. Herein, $\oplus$, © denote the operations of element-wise addition and concatenation respectively.

B. Deep Mutual Learning

The deep mutual learning aims to learn an ensemble of students collaboratively and teach each other using the losses based on the Kullback Leibler (KL) Divergence. Zhang et al. [21] propose a mutual learning framework which composes of two untrained student networks, and use two mimicry loss to guide learning of student networks. Yang et al. [22] propose the width-resolution mutual learning method to train a cohort of sub-networks with different widths using different input resolutions with KL loss. Wu et al. [23] propose a mutual learning module, which compose three student branches, and use a L2-based mimicry loss to optimize the network. In summary, these methods usually use the mimicry loss to supervise multi-branch networks collaboratively, where different initializations are used in different branches and the overall performance is thus improved by the mutual learning. Instead of mutual supervision by mimicry loss in existing works, our duality-gated mutual condition network is to exploit discriminative information of modalities as mutual conditions to enhance target representations of all modalities while mitigating noise effects.

III. Duality-Gated Mutual Condition Network

In this section, we first overview the our backbone architecture, and then introduce the details of the duality-gated mutual condition module and the resampling mechanism. Fig. 2 shows the overall framework of our tracking method, which consists of backbone architecture, duality-gated mutual condition module and resampling module. In the following, we present the details of each part.

A. Backbone architecture

As shown in Fig. 2, our inputs are the candidate patches cropped from aligned RGB and thermal image pair, and these patches are resize to size of 107×107. Our backbone is borrowed from the first three convolution layers of VGG-M [24], and their convolutional kernel sizes are $7\times7\times96$, $5\times5\times256$, $3\times3\times512$ respectively. The first and second convolutional layers is followed by a ReLU activation function, a local response normalization (LRN) and a max pooling layer. The third convolutional layer is just followed by a ReLU activation function. Followed by the last convolutional layer, the binary classification is performed, which consists of three fully connected layers with the output dimensions as 512, 512 and 2 respectively. And then we employ the multi-domain learning strategy to model appearance variations of instance objects [25]. As in MANet [12] to model robustly target representations using multi-modal information, we use the modality adapters to extract modality-specific features. In specific, the modality adapters are composed of convolutional layers, a ReLU activation function, a batch normalization, a dropout and a max pooling layer, and the each level modality adapter setting is same for different modalities. The size of the convolution kernel is $3\times3\times96$, $1\times1\times256$, $1\times1\times512$ in the three levels, as shown in Fig. 2.

B. Duality-Gated Mutual Condition Module

Although above backbone can provide robust target representations, the interactions between modalities are ignored, which plays a critical role in strengthening discriminative ability of multi-modal representations while suppressing feature noise. To handle this problem, we propose a novel duality-gated mutually condition module to achieve bi-directional conditional feature modulation between two modalities.

RGB-to-T Feature Modulation. Feature modulation is an effective method to influence or change the output feature of a model. We want to leverage discriminative features of RGB data to guide feature learning of target appearance in thermal data, and thus design a scheme to modulate thermal features with RGB information as conditions. Our idea is inspired by FiLM [26], which use prior information to construct two conditions that scale and shift features respectively. However, some issues need to be addressed when we apply FiLM to our task. First, the diversity of scaling and shifting conditions

is low and the potential of conditional feature learning could not be fully explored. Second, RGB information might contain noises since we do not know whether it is high-quality or not, and RGB-based conditions might thus be harmful for feature learning of thermal data.

To handle the first issue, we propose a new scheme to generate high-quality and diverse scaling and shifting conditions. We first modulate the outputs of thermal modality adapter by applying a multi-scale scaling transformation, based on the outputs of RGB modality adapter. In specific, we design a $MSConv$ layer (denoted as W^{ms}), which is implemented differently in different layers, to capture multi-scale feature information for generate the scaling conditions. In the first layer, the receptive fields of feature maps are small and the inter-modal variations are large, and we thus use four different convolutions to capture multi-scale information from different sizes of receptive fields. We use the 1×1 and 3×3 convolutions to capture local details, and use the 3×3 dilated convolution with the dilated rate of 2 and the 5×5 convolution to model global information. In the middle and high layers, the $MSConv$ layer is implemented with a combination of 1×1, 3×3, and 1×1 convolutions, respectively. The multi-scale scaling conditions of RGB modality can be expressed as follows:

$$f_R^{ms} = (W^f * [f_R * W^{ms}]) \tag{1}$$

where f_R is the feature maps of the modality adapter in RGB modality, and $*$ represents the convolutional operation, and f_R^{ms} denotes multi-scale scaling conditions.

Then, we further modulate the above scaled features of thermal modality adapter by applying a multi-modal shifting transformation, based on both outputs of RGB and thermal modality adapters. On one hand, we generate the multi-scale features from thermal modality since these features will be beneficial to enhancing target representations in thermal modality. On the other hand, we fuse multi-scale thermal features and RGB features to form the high-quality rich shifting conditions. The details of condition generation are shown in Fig. 4. Thus we can express the RGB-to-T feature modulation as follows:

$$f_T^{out} = f_T \odot f_R^{ms} + f_{T2R}^{scaled} \tag{2}$$

where $\odot$ represents elemental-wise multiplication, and f_T^{out} and f_{T2R}^{scaled} denote the modulated thermal features and the fused features of multi-scale thermal features and RGB features, respectively. Note that f_{T2R}^{scaled} is the fused features in RGB-to-T modulation but the scaled features in T-to-RGB modulation.

To handle the second issue, we design a duality-gated strategy to avoid noisy information of RGB modality in generating conditions. Fig. 4 shows the details of the duality-gated structure, where the two gates have the same internal structure, which is formulated as follows.

$$G = \sigma(Conv(f)) \tag{3}$$

where $Conv(\cdot)$ and σ denote the operations of 1×1 convolution and sigmoid function respectively. f indicates the input features. Therefore, we embed the duality-gated formulation in RGB-to-T modulation as follow:

$$f_T^{out} = f_T \odot G_1(f_R^{ms}) + G_2(f_{T2R}^{scaled}) \tag{4}$$

where G_1 and G_2 represent two gates to mitigate the noises of multi-scale scaling conditions and the fused features respectively. The effectiveness of our duality-gated mechanism is shown in Fig. 3, and we can see that the duality-gated scheme is able to enhance the effectiveness of feature representations while suppressing the feature noises.

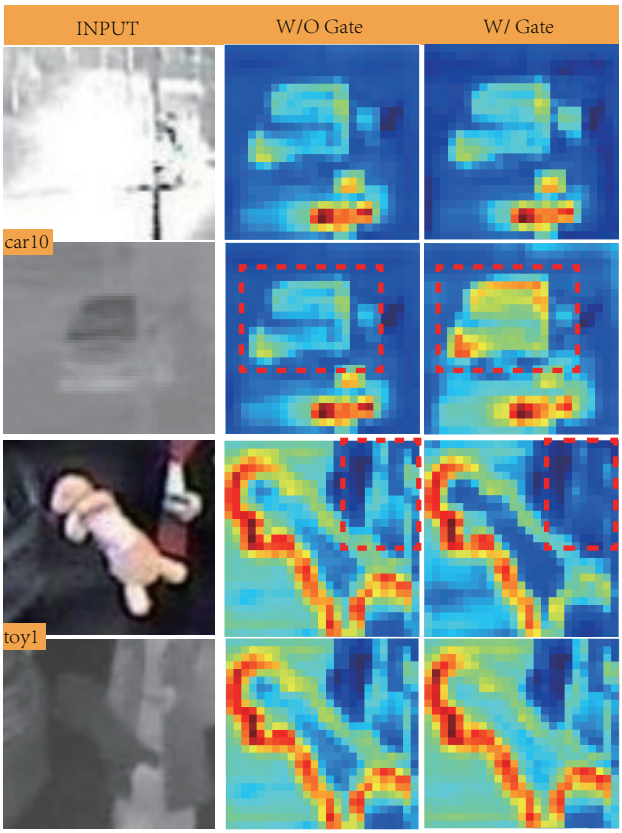


Fig. 3: Illustration of the effectiveness of our duality-gated mutual condition module against on two example frames. Herein, the first column represents the input RGB and thermal images, and the second column denotes the feature maps of DMCNet without all gates. The third column indicates the feature maps of DMCNet.

T-to-RGB Feature Modulation. In this work, we want to leverage all discriminative information of different modalities regardless of low-quality and high-quality modalities. Therefore, we adopt a bi-directional conditional feature learning structure to fully mine the discriminative features of all modalities. The structure of T-to-RGB feature modulation is symmetric to RGB-to-T one, and we thus obtain the final output of T-to-RGB feature modulation as follow:

$$f_R^{out} = f_R \odot G_3(f_T^{ms}) + G_4(f_{R2T}^{scaled}) \tag{5}$$

where the f_R^{out} represents the modulated RGB features.

.

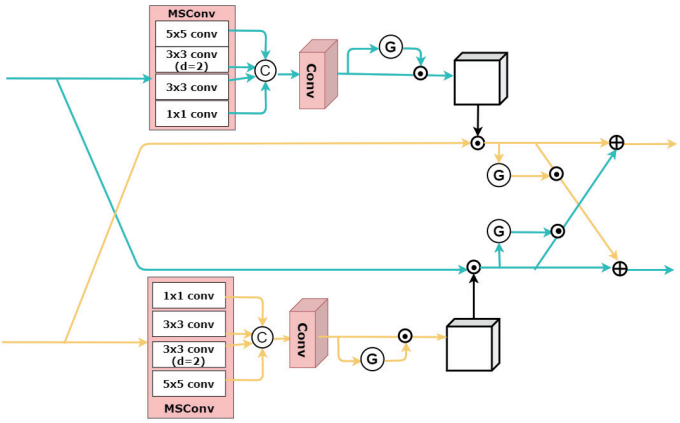


Fig. 4: Illustration of our duality-gated mutual conditional module. Herein, $\oplus$ and $\odot$ denote the operations of element-wise addition and element-wise multiplication respectively. G denotes the gated operation.

C. Re-Sampling Module

In RGBT tracking task, abrupt camera motion is a common challenge, which affects the performance much. The major reason is that under such challenge the search window usually can not cover target objects, which would lead to tracking failure. Common attempts are that one can expand search region [13] and perform global search [14], but these methods bring more background information and thus increase the risk of model drift. Meanwhile, the computational cost is usually greatly increased.

To handle these issues, we develop a re-sampling scheme based on a fast optical flow algorithm [15]. Specifically, when model state is unreliable we employ the optical flow to detect the motion state of camera, and then determine whether the re-sampling is executed or not. First, we start optical flow estimation when tracking failure is detected, i.e., the predicted target score is below 0. Second, we use Disflow [15] to compute displacements of all pixels in a local region around target object, and then calculate the mean displacement vector $[dx, dy]$. In this work, this local region is centered at the target position in previous frame and its size is three times of the size of target bounding box. Third, we judge whether abrupt camera motion occurs or not by comparing the amplitude of $[dx, dy]$ with a predefined threshold u. If $|dx|$ or $|dy|$ is below than u, we think the failure is not caused by abrupt camera motion and do not execute re-sampling. Otherwise, we judge that abrupt camera motion occurs and employ $[dx, dy]$ to guide re-sampling. Fourth, we perform re-sampling according to $[dx, dy]$. In specific, we empirically re-sample 16 candidate regions along the judged direction which is opposite to camera motion and the step of re-sampling is set as follows. In the horizontal direction, we take the quarter of the width of target bounding box as the step, while in the vertical direction we take the quarter of the height of target bounding box. Finally, we feed these samples into our network to compute their scores and combine them with the results of Gaussian sampling to compute the final predicted result. We present the more details in Algorithm 1.

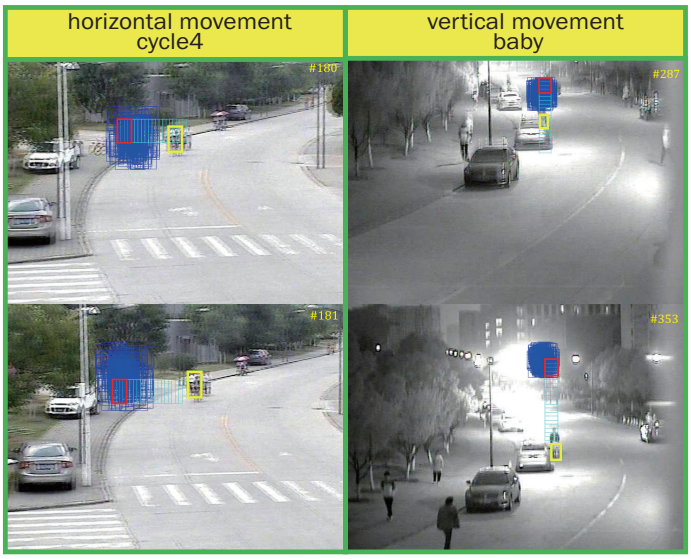


Fig. 5: Example of the re-sampling scheme. Herein, the blue boxes represent the original sampling regions, the cyan boxes represent the re-sampling regions, the red boxes indicate the final tracking results, and the green boxes are the ground truths. Note that the images in top row involve horizontal movement of camera and the images in second row involve vertical movement of camera.

D. Network Training

We employ the Stochastic Gradient Descent (SGD) algorithm to train our network effectively. First, we load the pre-trained model, which is trained on the ImageNet [27] dataset, and then use a multi-domain learning algorithm [25] to learn the parameters of the backbone sub-network and the first two fully connected layers, while the parameters of other sub-networks are initialized randomly. Then, we use RGBT dataset to train the whole network with 200 epoch iterations by softmax cross-entropy loss, and set different learning rates to different sub-networks. In specific, we set the learning rates of backbone sub-network and first two fully connected layers and binary gate layer to 0.01, and modality-specific sub-network and mutual-conditional sub-network to 0.02. In each iteration, we select 8 frames which are randomly chosen in each video sequence from training dataset to construct a mini-batch. Next, we draw 32 positive samples from each frame and 96 negative samples from each frame to form the input data in a mini-batch. Here, we use the RGBT234 [18] dataset as training dataset when conducting evaluations on the GTOT [16] dataset. In contrast, when performing evaluations on the RGBT234 [18] and RGBT210 [1] datasets, we use the GTOT [16] dataset as the training set as RGBT234 and RGBT210 have some overlaps in videos.

IV. Online RGBT Tracking

In the tracking process, we re-construct a new last fully connect layer as the video-specific layer for each instance

object in each sequence. Then, we freeze all the parameters of convolution layers (W_{conv}) and fine-tune the three fully connect layers ($W_{fc4,fc5,fc6}$) use the initial target state. Specifically, given the first frame pair of the sequence and the ground truth bounding box, we draw 500 positive and 5000 negative samples as the training samples, where we define the samples whose IoU with the ground truth is larger than 0.7 as positive samples and smaller than 0.5 as negative samples. In the initial train process, we employ these samples to train the three fully connected layers with 50 iterations, and we set the learning rate are 0.005 and 0.0005 for the last layer and other two fully connected layers, respectively. We apply bounding box regression technique [25] to improve target localization accuracy and solve the target scale change during the tracking process. To prevent the potential unreliability int the subsequent frames, we only train a bounding box regressor in the initial frame, and we only use it to adjust target states in tracked-successful frames.

In the subsequent frame, we collect 50 positive samples whose IoU is larger than 0.7 and 200 negative samples whose IoU is smaller than 0.3 as training samples for short-term and long-term update [25], and here the learning rate of the last fully connected layer and the other two fully connected layers are set to 0.01 and 0.001 respectively. Given the t-th frame, we first draw a candidate set X_t^i from a Gaussian distribution of previous frame tracking result X_{t-1}^*, where the mean of Gaussian function is set to $X_{t-1}^* = (cx_{t-1}, cy_{t-1}, s_{t-1})$ and the covariance is set as a diagonal matrix $diag(0.09r^2, 0.09r^2, 0.25)$. where r is the mean of the width and height of target, and the (cx, cy) and s indicate the location and scale respectively in the previous frame. In tracking process, we feed all candidate samples from X_t^i into our network, and compute the positive scores and negative scores of candidate samples using the trained networks as $f^+(X_t^i)$ and $f^-(X_t^i)$, respectively. We sort the candidate samples by their scores and select the candidate samples with the top five highest scores, and then compute its the mean value as the tracking result X_t^* of the current frame t, where the X_m^* is denoted as the samples set with top five highest scores and the $mean()$ represents the averaging operation. The formula expression is as follows:

$$\begin{aligned} X_m^* &= \underset{i=1,\ldots,256}{\arg\max}\ f^+(X_t^i) \\ X_t^* &= mean(X_m^*). \end{aligned} \tag{6}$$

If the mean of the top five scores $F(X_t)$ less than 0, we calculate the average moving vector $[dx, dy]$ and judge abrupt camera motion to be occurred when the amplitude of $[dx, dy]$ exceeds the predefined threshold u. Empirically, we set u to 5 in this work, and its setting is validated in Fig 7. We can see that When u is equal to 0, the performance is the lowest at this time. This is because all tracking failures are caused by camera motion by default, so more interference candidate samples will inevitably be brought, causing performance degradation We also can see that when u is equal to 5, the performance is the highest, and then as the value of u increases, the performance has a downward trend. This is because the larger the value of u, the fewer failure cases that are recognized as camera movement, so the resampling mechanism cannot be activated. Thus we select the value of u based above analyses. When the abrupt camera motion occurs, we execute the re-sampling strategy to obtain a new candidate set and then feed it into our network to compute their scores, obtaining the top score $RF(X_t)$. To improve the robustness, we combine the Gaussian sampling and re-sampling methods to determine the final tracking results. In specific, we use the candidate sample with higher score from $F(X_t)$ and $RF(X_t)$ as the predicted tracking result. More details can be referred to Algorithm 1.

Algorithm 1 Online RGBT Tracking Process

Input: Pretrainded CNN filters W_{conv},$W_{fc4,fc5}$; Initial target state X_1; Threshold u.
Output: Estimated target state X_t^*.

```
Randomly initialize the last layer W_fc6;
Train a bounding box regression model BB(·) [25] ;
Draw positive samples S_1^+ and negative samples S_1^-;
Fine-tune W_{fc4,fc5,fc6} using S_1^+ and S_1^-;
Initialize short-term [25] and long-term [25] sample set φ_s, φ_l.
repeat
    Draw target candidate set X_t^i;
    Compute target state X_m^* and score F(X_t) by Eq. 5.
    if F(X_t) > 0 then
        Estimated target state: X_t^* = mean(X_m^*);
        X_t^* = BB(X_t^*);
        Update φ_s, φ_l.
    else
        Compute [dx, dy] by Disflow [15];
        if |dx| > u or |dy| > u then
            Perform re-sampling;
            Compute target state RX_t^* and score RF(X_t).
            if RF(X_t) > (F(X_t) then
                Estimated target state: X_t^* = RX_t^*.
            else
                Estimated target state: X_t^*.
        else
            Execute short-term update W_{fc4,fc5,fc6} using φ_s.
    if t mod 10 = 0 then
        Execute long-term update W_{fc4,fc5,fc6} using φ_l.
until end of sequence
```

V. Performance Evaluation

In this section, we evaluate our duality-gated mutual-conditional network (named DMCNet in this paper) with existing RGBT and RGB trackers on four popular RGBT tracking benchmark datasets including GTOT [16], RGBT210 [1], RGBT234 [18] and VOT-RGBT2019 [17]. The experimental environment is configured as follows: Pytorch 1.0+, 8 NVIDIA GeForce GTX 2080Ti GPU server.

A. Evaluation Setting

Datasets. We use four large challenging tracking datasets, GTOT [16] and RGBT210 [1], RGBT234 [18] and VOT-RGBT2019 [17], to comprehensively evaluate our DMCNet.

TABLE I: Attribute-based PR/SR scores (%) of our DMCNet on RGBT234 dataset against with eight RGBT trackers. The best, second and third results are in red,blue and green.

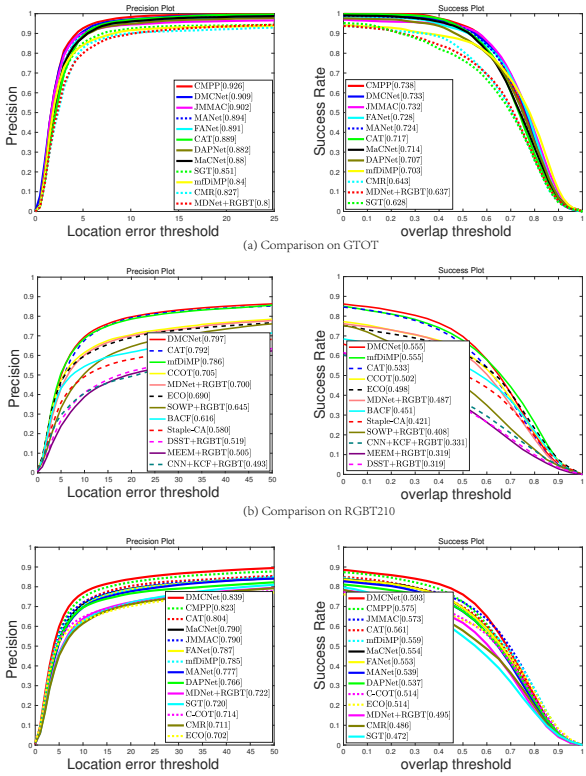
Trackers	MDNet+RGBT	SGT	CMR	DAPNet	MANet	MaCNet	FANet	CMPP	JAMMC	CAT	DMCNet
Pub. Info.	CVPR2016	ACM MM2017	ECCV2018	ACM MM2019	ICCVW2019	Sensors2020	TIV2020	CVPR2020	TIP2020	ECCV2020	
NO	86.2/61.1	87.7/55.5	89.5/61.6	90.0/64.4	88.7/64.6	92.7/66.5	88.2/65.7	95.6/67.8	93.2/69.4	93.2/66.8	92.3/ 67.1
PO	76.1/51.8	77.9/51.3	77.7/53.6	82.1/57.4	81.6/56.6	81.1/57.2	86.6/60.2	85.5/60.1	84.1/61.1	85.1/59.3	89.5/63.1
HO	61.9/42.1	59.2/39.4	56.3/37.7	66.0/45.7	68.9/46.5	70.9/48.8	66.5/45.8	73.2/50.3	67.7/48.3	70.0/48.0	74.5/52.1
LI	67.0/45.5	70.5/46.2	74.2/49.8	77.5/53.0	76.9/51.3	77.7/52.7	80.3/54.8	86.2/58.4	84.0/58.8	81.0/54.7	85.3/58.7
LR	67.0/45.5	72.5/46.2	72.0/47.6	75.9/51.5	70.8/48.7	75.1/47.6	75.0/51.0	86.5/57.1	84.0/58.8	82.0/53.9	85.4/57.9
TC	75.6/51.7	76.0/47.0	67.5/44.3	76.8/54.3	75.4/54.3	77.0/56.3	76.6/54.9	83.5/58.3	74.9/52.6	82.0/53.9	87.2/ 61.2
DEF	66.9/47.3	68.5/47.4	66.7/47.3	71.7/57.8	72.0/52.4	73.1/51.4	72.2/52.6	75.0/54.1	70.6/52.9	76.2/54.1	77.9/56.5
FM	58.6/36.3	67.7/40.2	61.3/38.4	67.0/44.3	69.4/44.9	69.4/44.9	68.1/43.6	78.6/50.8	61.0/41.7	73.1/47.0	80.0/52.4
SV	73.5/50.5	69.0/43.4	71.0/49.3	78.0/54.2	77.7/54.2	78.7/56.1	78.5/56.3	81.5/57.2	83.7/ 61.6	79.7/56.6	84.6/ 59.8
MB	65.4/46.3	64.7/43.6	60.0/42.7	65.3/46.7	72.6/51.6	71.6/52.5	70.0/50.3	75.4/54.1	75.1/54.9	68.3/49.0	77.3/ 55.9
CM	64.0/45.4	66.7/45.2	62.9/44.7	66.8/47.4	71.9/50.8	71.7/51.7	72.4/52.3	75.6/54.1	76.2/55.6	75.2/52.7	80.1/ 57.6
BC	64.4/43.2	65.8/41.8	63.1/39.8	71.7/48.4	73.9/48.6	77.8/50.1	75.7/50.2	83.2/53.8	68.7/48.5	81.1/51.9	83.8/ 55.9
ALL	72.2/49.5	72.0/47.2	71.1/48.6	76.6/53.7	77.7/53.9	79.0/55.4	78.7/55.3	82.3/57.5	79.0/57.3	80.4/56.1	83.9/ 59.3

The GTOT consists of 50 aligned RGB and thermal infrared video pairs, containing approximately 15K frames of images and seven visual tracking challenge attributes. The RGBT210 consists of 210 aligned RGB and thermal infrared video pairs, containing 210K frames in total and a maximum of 8K frames per video pair and a total of 12 visual tracking challenge attributes. The RGBT234 dataset, an extension of the RGBT210 [1] tracking dataset, consists of 234 aligned RGB and thermal infrared video pairs, containing approximately 200K frames of images and 12 visual tracking challenge attributes, such as camera moving, large scale variations and environmental challenges. The VOT-RGBT2019 dataset, a subset of the RGBT234 dataset [18], which consists of 60 high aligned RGBT video sequences selected from RGBT234 dataset [18], with a total of over 20K frames.

Evaluation metrics. Precision rate (PR) and success rate (SR) are used to evaluate RGBT tracking performance on three RGBT tracking datasets. PR is the percentage of frames whose distance of the output position with the ground truth is below a predefined threshold, and the thresholds in the GTOT and RGBT234 tracking datasets are set to 5 and 20 pixels respectively to obtain the representative PR score (because the target objects in the GTOT dataset are usually small). SR is the percentage of frames where the overlap rate between the output bounding boxes and the ground truth bounding boxes are greater than a threshold. By changing the threshold, SR curve can be obtained, and the area under the curve of SR curve is used to define the representative SR. To more comprehensive evaluate different RGBT tracking algorithms, we also follow the VOT official evaluation protocol. Specifically, here three evaluation metrics, Expected Average Overlap (EAO), robustness (R) and accuracy (A), are used. A is the average overlap between the predicted and ground truth bounding boxes during successful tracking periods. R measures how many times the tracker loses the target (fails) during tracking. EAO is a combined measure of A and R.

B. Evaluation on GTOT Dataset

In GTOT dataset, we compare our DMCNet with 11 state-of-the-art RGBT trackers, among which SGT [1], DAPNet [5], MANet [12], MaCNet [7] and CMR [4], CMPP [8], CAT [10], FANet [11], mfDiMP [6], JMMAC [9] are RGBT trackers,



(a) Comparison on GTOT

(b) Comparison on RGBT210

(c) Comparison on RGBT234

Fig. 6: Evaluation curves on GTOT, RGBT210 and RGBT234 datasets compare with RGBT trackers. The representative scores of PR/SR are presented in the legend.

while MDNet [25] +RGBT are the extended RGBT tracker from existing RGB tracking algorithm by concatenating thermal and RGB features. The evaluation results are shown in Fig. 6 (a), and our DMCNet has a comparable performance with the state-of-the-arts on GTOT dataset. Compare with CMPP [8] method, which employs a lot of history samples as temporal information to enhance the current frame representation, while our DMCNet just use the current frame information.

C. Evaluation on RGBT210 Dataset

In RGBT210 [1] dataset, we compare our method with 11 trackers using two evaluation metrics. From Fig. 6 (b), we can see that the performance of our method exceeds mfDiMP [6] 1.1% in PR and the CAT [10] 2.2% in SR , and significantly outperforms other trackers, including CCOT [28], MDNet [29] +RGBT, ECO [30], SGT [1], SOWP [31] +RGBT, DSST [32] +RGBT, BACF [33], Staple-CA [34], MEEM [35] +RGBT and CNN +KCF [36] +RGBT

It is worth noting that mfDiMP [6] is the winner of the VOT2019-RGBT tracking competition. Furthermore, it uses a larger-scale generated RGBT dataset (9335 sequences) to train network, but our network only uses GTOT dataset (50 sequences) as training set.

D. Evaluation on RGBT234 Dataset

To further evaluate our method, we conduct the experiments on RGBT234 tracking dataset, including overall comparison and challenge-based performance.

Overall comparison. For more comprehensive evaluation, we compare our DMCNet with 13 state-of-the-art trackers, among which CMPP [8], CAT [10], JMMAC [9], FANet [11], mfDiMP [6], SGT [1], CMR [4], CCOT [28], ECO [30], MDNet [25] +RGBT, DAPNet [5], MANet [12] and MaCNet [7] are RGBT trackers. The evaluation results are shown in Fig. 6 (c). We can see that the performance of our DMCNet has a clearly superior comparing with the state-of-the-art RGBT methods in all metrics. It fully demonstrates the effectiveness of our method. In particular, our DMCNet (83.0%/58.5% in PR/SR) achieves 1.1%/1.5%, 3.0%/2.9% and 4.4%/1.7% gains in PR/SR over CMPP [8], CAT [10] and JMMAC [9] respectively.

In addition, to further validate the effectiveness of DMCNet, we present the visualization results with some advanced trackers in Fig. 8, including CMPP [8], CAT [10], JMMAC [9], MANet [12], DAPNet [5] and MaCNet [7]. From the Fig. 8, we can see that our approach performs obviously better than other trackers in several challenges, such as camera motion, thermal crossover, background clutter, heavy occlusion, low resolution and low illumination. For instance, Fig. 8 (a) and (c) show the tracking results of our method with other trackers on the video sequences with camera motion, low resolution and low illumination. Obviously, DMCNet can more robustly localize the target while other advanced algorithms lose the tracked object when the sudden camera motion happens. In Fig. 8 (b), in which the data has background clutter, heavy occlusion and low illumination challenge attributes, most advanced trackers are failed. While our method and CMPP [8] can continuously track the target. The video sequence shown in Fig. 8 (d) has a serious thermal crossover phenomenon. Other trackers just locate the target in part of frames, but our tracker can be deal with this challenge well. To sum up, DMCNet is very robust in adverse conditions due to the benefits from the duality-gated mutual condition module and the re-sampling strategy.

Challenge-based performance. We show the results of our DMCNet against other state-of-the-art RGBT trackers, including MDNet [25] +RGBT, SGT [1], DAPNet [5], MANet [12], MaCNet [7], CMPP [8], CAT [10], FANet [11] and JMMAC [9] on different subsets with different challenge attributes. The challenge attributes 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). The evaluation results are shown in Table I. The results show that the our method performs the best under the most challenging conditions. It demonstrates the robustness of our DMCNet in handling most adverse conditions.

E. Evaluation on VOT-RGBT2019 Dataset

We further evaluate our method against several state-of-the-art trackers on VOT-RGBT2019 [17] dataset, including mfDiMP [6], MaCNet [7], MANet [12] and so on. We follow the VOT protocol and adopt EAO, R and A as the metrics. We compare DMCNet with seven RGBT tracking algorithms and directly use the results reported in papers to ensure the best performance. From the results of Table II, we can see that our DMCNet has comparable performance against mfDiMP, and outperforms other state-of-the-art methods including MANet and MaCNet. Some state-of-the-art RGBT tracking methods, such as CMPP, CAT, do not report the VOT2019-RGBT evaluation results, thus we do not consider the comparison with these algorithms in this dataset. Note that, from the results we can find that our DMCNet lower than mfDiMP in R metric. It is mainly due to two reasons. First, mfDiMP employs the conception of IoU network to improve tracking results. Second, mfDiMP uses the training set of GOT-10k dataset to generate a large-scale synthetic RGBT dataset as their training data (9,335 videos with 1,403,359 frames in total), while we only use GTOT dataset (50 videos with 15,000 frames in total) to train our network. We will improve the performance of DMCNet from these considerations in the future.

F. Analysis of Our DMCNet

Impact of parameter u. In the module of re-sampling, u is a critical hyper parameter. We manually set u ={ 0, 5, 10, 15, 20, 25, 30} to analyze the impact on the tracking performance. The tracking results on RGBT234 dataset are shown in the Fig. 7. From the results, we can see the tracker achieves the best performance with u=5. By observing Fig. 7, we can find that the tracking performance first rises and then falls with the increasing of u value. The main reason is that the re-sampling module will more easily be executed when u is smaller, which will cause more false activations.

Ablation study. To validate the effectiveness of major components in our method, we implement two variants and evaluate them on RGBT234 and GTOT datasets. They are: 1) DMCNet-v1, that removes all duality gated mutually conditional modules and the resampling strategy in our DMCNet; 2) DMCNet-v2, that removes the resampling strategy in our DMCNet;

TABLE II: Comparison results on VOT-RGBT2019 dataset.

	GESBTT	CISRDCF	MPAT	FSRPN	mfDiMP	MaCNet	MANet	DMCNet
A(↑)	0.6163	0.5215	0.5723	0.6362	0.6019	0.5451	0.5823	0.6009
R(↑)	0.6350	0.6904	0.7242	0.7069	0.8036	0.5914	0.7010	0.7088
EAO	0.2896	0.2923	0.3180	0.3553	0.3879	0.3052	0.3463	0.3796

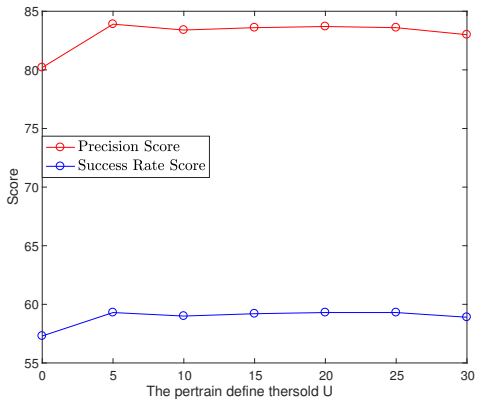


Fig. 7: Results of our method with different u values on RGBT234 [18].

TABLE III: PR/SR scores of different variants induced from our method on RGBT234 dataset and GTOT dataset. ✓means adding the corresponding component.

			RGBT234		GTOT	
	DMC	RS	PR	SR	PR	SR
DMCNet-v1			0.802	0.565	0.866	0.693
DMCNet-v2	✓		0.820	0.584	**0.909**	**0.733**
DMCNet	✓	✓	**0.839**	**0.593**	**0.909**	**0.733**

Table III presents the comparison results on RGBT234 and GTOT datasets, and the results fully demonstrate the effectiveness of the proposed components. Note that the RS module looks like useless in the evaluation of GTOT, and the major reason is that GTOT dataset does not include the camera motion attribute. Therefore, our RS module will not be activated in the evaluation of GTOT dataset, which leads to performance unchanged.

To validate the effectiveness of major components in our duality-gated mutual condition module, we implement three variants and evaluate them on RGBT234 and GTOT datasets. They are: 1) DMC-no-msconv, that removes multi-scale convolutions in DMC; 2) DMC-no-gate, that removes all gates in DMC; 3) DMC-one-gate, that removes all second gates in DMC. Table IV presents the comparison results on RGBT234 and GTOT datasets, and the results fully demonstrate the effectiveness of the components in DMC. Form the above results, we can conclude that duality-gated strategy is truly bring more performance gains, which suggest that each gate play a certain role in the DMC module.

To validate the effectiveness of the shifting operations used in DMC module, which is different from FiLM [26]. We implement two variants and evaluate them on RGBT234 dataset. There are: 1) w/o-shifting, that removes the shifting operation in DMC; 2) w/-FiLM [26] shifting, that replaces the existing shifting operation with the shifting of FiLM. Table V presents the comparison results on RGBT234 dataset, and the results fully prove that our shift operation is better than the FiLM.

TABLE IV: PR/SR scores of different DMC variants induced from our method on RGBT234 dataset and GTOT dataset.

	RGBT234		GTOT	
	PR	SR	PR	SR
DMC	**0.820**	**0.584**	**0.909**	**0.733**
DMC-no-msconv	0.811	0.575	0.901	0.725
DMC-no-gate	0.807	0.572	0.897	0.722
DMC-one-gate	0.813	0.577	0.902	0.724

TABLE V: PR/SR scores of different shifting methods induced from our method on RGBT234 dataset.

	PR	SR
DMC	**0.820**	**0.584**
w/o-shifting	0.790	0.563
w/-FiLM [26] shifting	0.816	0.575

VI. Conclusion

In this paper, we propose a duality-gated mutual condition network to make full use of the discriminative information of all modalities especially for low-quality modalities. Our method employs mutual condition module to transform the effective information of RGB and thermal modalities as the mutual conditions, and then use them to fully enhance the discriminative ability of two modalities. A duality-gated mechanism is also introduced to improve the quality of generated conditions. Extensive experiments on four RGBT benchmark datasets show that our method achieves outstanding performance comparing with the state-of-the-art methods. In future work, we will explore effective external knowledge to enlarge the power of duality-gated mutual conditions for more robust RGBT tracking.

References

[1] C. Li, N. Zhao, Y. Lu, C. Zhu, and J. Tang, "Weighted sparse representation regularized graph learning for rgb-t object tracking," in *Proceedings of the ACM International Conference on Multimedia*, 2017.

[2] X. Lan, M. Ye, S. Zhang, and P. C. Yuen, "Robust collaborative discriminative learning for rgb-infrared tracking," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[3] Y. Wang, C. Li, and J. Tang, "Learning soft-consistent correlation filters for rgb-t object tracking," in *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, 2018.

[4] C. Li, C. Zhu, Y. Huang, J. Tang, and L. Wang, "Cross-modal ranking with soft consistency and noisy labels for robust rgb-t tracking," in *Proceedings of the European Conference on Computer Vision*, 2018.

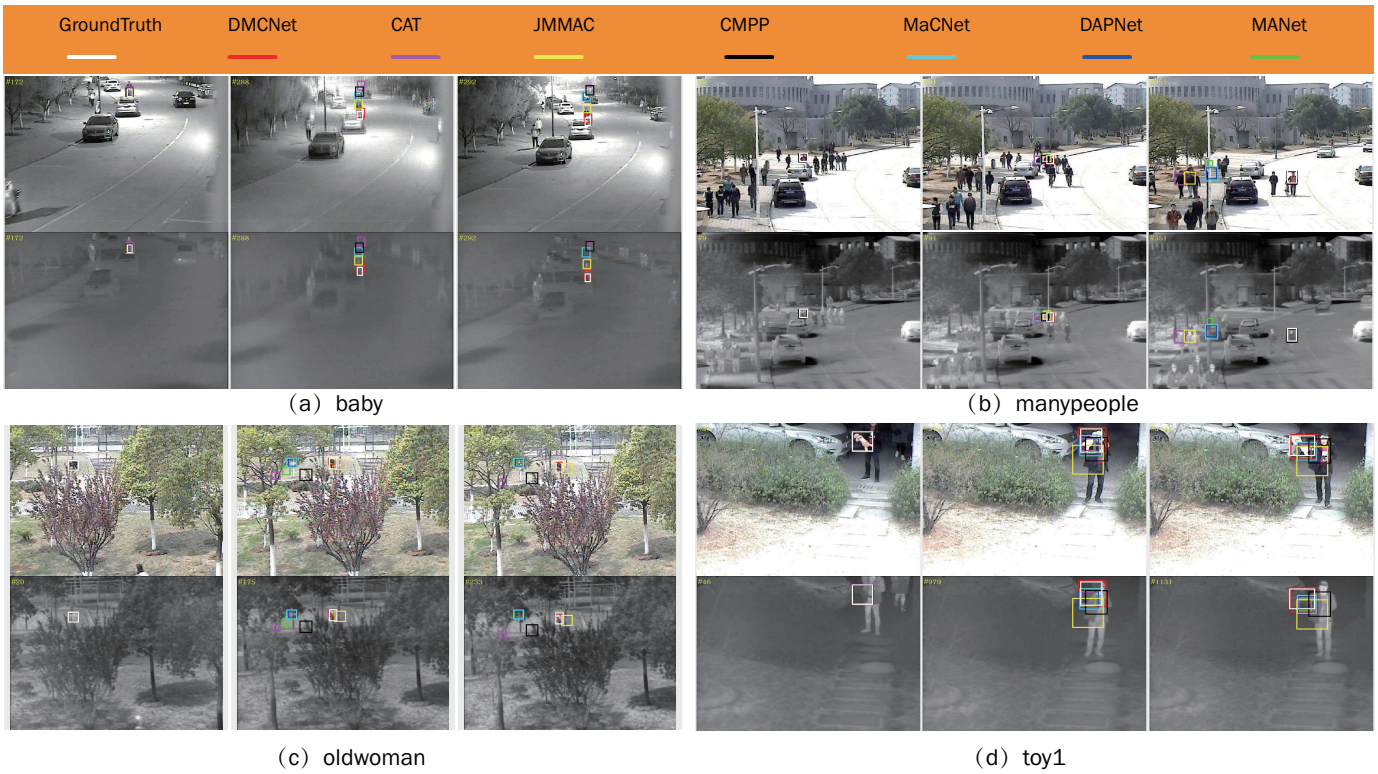


Fig. 8: Qualitative comparison between DMCNet and other state-of-the-art trackers on four video sequences.

[5] Y. Zhu, C. Li, B. Luo, J. Tang, and X. Wang, "Dense feature aggregation and pruning for rgbt tracking," in *Proceddings of ACM International Conference on Multimedia*, 2019.

[6] L. Zhang, M. Danelljan, A. Gonzalez-Garcia, J. van de Weijer, and F. Shahbaz Khan, "Multi-modal fusion for end-to-end rgb-t tracking," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[7] L. Z. L. Z. J. Zhang, Hui; Zhang, "Object tracking in rgb-t videos using modal-aware attention network and competitive learning," *Sensors*, 2020.

[8] C. Wang, C. Xu, Z. Cui, L. Zhou, and J. Yang, "Cross-modal pattern-propagation for rgb-t tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[9] P. Zhang, J. Zhao, D. Wang, H. Lu, and X. Yang, "Jointly modeling motion and appearance cues for robust rgb-t tracking," *IEEE Transactions on Image Processing*, 2020.

[10] C. Li, L. Liu, A. Lu, Q. Ji, and J. Tang, "Challenge-aware rgbt tracking," 2020.

[11] Y. Zhu, C. Li, J. Tang, and B. Luo, "Quality-aware feature aggregation network for robust rgbt tracking," *IEEE Transactions on Intelligent Vehicles*, 2020.

[12] C. Li, A. Lu, A. Zheng, Z. Tu, and J. Tang, "Multi-adapter rgbt tracking," in *Proceedings of IEEE International Conference on Computer Vision Workshops*, 2019.

[13] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[14] G. Zhu, F. Porikli, and H. Li, "Beyond local search: Tracking objects everywhere with instance-specific proposals," 2016.

[15] T. Kroeger, R. Timofte, D. Dai, and L. Van Gool, "Fast optical flow using dense inverse search," in *Proceedings of the IEEE European Conference on Computer Vision*, 2016.

[16] 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*, vol. 25, no. 12, pp. 5743–5756, 2016.

[17] www.votchallenge.net/vot2019.

[18] C. Li, X. Liang, Y. Lu, N. Zhao, and J. Tang, "Rgb-t object tracking: benchmark and baseline," *Pattern Recognition*, vol. 96, p. 106977, 2019.

[19] C. Li, X. Sun, X. Wang, L. Zhang, and J. Tang, "Grayscale-thermal object tracking via multi-task laplacian sparse representation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 673–681, 2017.

[20] C. Li, X. Wu, N. Zhao, X. Cao, and J. Tang, "Fusing two-stream convolutional neural networks for rgb-t object tracking," *Neurocomputing*, vol. 281, pp. 78–85, 2018.

[21] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[22] T. Yang, S. Zhu, C. Chen, S. Yan, M. Zhang, and A. Willis, "Mutualnet: Adaptive convnet via mutual learning from network width and resolution," 2020.

[23] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, "A mutual learning method for salient object detection with intertwined multi-supervision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

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

[25] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[26] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[27] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Feifei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[28] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proceedings of IEEE European Conference on Computer Vision*, 2016.

[29] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[30] 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.

[31] 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 the IEEE International Conference on Computer Vision*, 2015.

[32] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proceedings of the British Machine Vision Conference*, 2014.
[33] H. Kiani Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[34] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
[35] 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.
[36] 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*, vol. 37, no. 3, pp. 583–596, 2015.