Learning Diverse Models for End-to-End Ensemble Tracking

Ning Wang, *Graduate Student Member, IEEE*, Wengang Zhou, *Senior Member, IEEE*, and Houqiang Li, *Fellow, IEEE*

***Abstract*—In visual tracking, how to effectively model the target appearance using limited prior information remains an open problem. In this paper, we leverage an ensemble of diverse models to learn manifold representations for robust object tracking. The proposed ensemble framework includes a shared backbone network for efficient feature extraction and multiple head networks for independent predictions. Trained by the shared data within an identical structure, the mutually correlated head models heavily hinder the potential of ensemble learning. To shrink the representational overlaps among multiple models while encouraging the diversity of individual predictions, we propose the model diversity and response diversity regularization terms during training. By fusing these distinctive prediction results via a fusion module, the tracking variance caused by the distractor objects can be largely restrained. Our whole framework is end-to-end trained in a data-driven manner, avoiding the heuristic designs of multiple base models and fusion strategies. The proposed method achieves state-of-the-art results on seven challenging benchmarks while operating in real-time.**

***Index Terms*—Visual tracking, ensemble learning, end-to-end tracker, diverse models.**

I. Introduction

VISUAL object tracking is a fundamental task in computer vision with various applications such as human-computer interaction, autonomous driving, and video surveillance [1]. Given the initial ground-truth annotation, tracking algorithms are required to consistently localize the target and cope with various challenging factors such as target occlusion, viewpoint change, and deformation. Despite the rapid progress in the past decades, how to effectively model the target appearance for robust visual tracking still remains a challenging problem.

A single tracker generally struggles to comprehensively model the target appearance and suffers from the occlusal drift. In the tracking process, unfortunately, accidental prediction biases will be gradually accumulated and lead to the unrecoverable failure. To alleviate the limitation of single models, a natural way is to assemble multiple trackers with diversified capabilities for cooperation. It has been well recognized that by reasonably fusing the results from different models, the tracking variance can be largely reduced. In a satisfying ensemble tracking framework, the following aspects are of vital importance: model diversity, fusion strategy, and tracking efficiency. Nevertheless, previous ensemble trackers mostly ignore the balance of these factors to some extent. First, model diversity guarantees the effectiveness of ensemble learning. Even though the previous ensemble approaches typically put some emphasis on the model designing, how to effectively enlarge the model diversity still leaves exploration room, and most of them build diversified models in a heuristic manner such as adopting different features [2]–[6], update scheme [7], or training data [8]. Second, fusion strategy, as another core component, has been widely investigated. Nevertheless, a majority of existing fusion strategies are manually designed with carefully tuned hyper-parameters [3], [5], [6], [9], potentially restricting the algorithm generalization. Finally, the online efficiency is also important since the tracking task is tightly related to the practical vision scenarios. However, by running multiple heavyweight trackers in parallel, some ensemble frameworks fail to achieve a real-time speed [5], [10], [11].

In this paper, we focus on the ensemble tracking framework to model diversified target representations on manifolds. Considering the aforementioned requirements, we design an end-to-end ensemble tracker consisting of a single backbone and multiple head networks. The shared backbone network extracts deep features once in each frame, greatly ensuring online tracking efficiency. The multiple head networks model the target appearance from manifold feature spaces, which complement each other to reduce the tracking variance and together contribute to a more robust tracking system. As shown in Figure 1, different head models predict distinctive results (e.g., confidence in discriminating distractors) despite their same network structures and input features. To further shrink the representational overlaps and encourage the prediction diversity, we propose the model diversity and response diversity regularizations in the training stage. The model diversity encourages multiple head networks to learn diversified discriminative models in the feature level, while the response diversity regularization enlarges the differences

Manuscript received July 12, 2020; revised November 28, 2020; accepted December 28, 2020. Date of publication January 20, 2021; date of current version January 27, 2021. This work was supported in part by NSFC under Grant 61836011, Grant 61836006, and Grant 61822208, in part by the Youth Innovation Promotion Association CAS under Grant 2018497, and in part by the GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Liang Wang. *(Corresponding authors: Wengang Zhou; Houqiang Li.)*

The authors are with the CAS Key Laboratory of Technology in Geo-spatial Information Processing and Application System, Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230052, China (e-mail: wn6149@mail.ustc.edu.cn; zhwg@ustc.edu.cn; lihq@ustc.edu.cn).

Digital Object Identifier 10.1109/TIP.2021.3051471

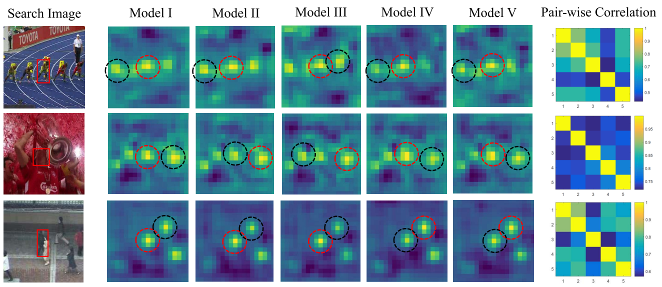


Fig. 1. Prediction results from different models in our ensemble framework. The highest and second-highest peaks are highlighted by the red and black circles, respectively. In the last column, we present the pair-wise similarity (cosine distance) of different response maps. Different models yield distinctive results, e.g., fine-grained target localizations and discrimination confidences on the distractors. Best view in color and zoom in.

among individual predictions. Finally, by introducing a simple yet effective fusion module, we dynamically integrate the parallelly generated tracking results. Different from previous ensemble trackers with heuristic designs in both model selection and fusion strategies, we free these requirements by jointly optimizing the diverse models and fusion scheme in a data-driven manner. It is also worth mentioning that compared with our baseline approach [12], our ensemble framework neither modifies the base network structure, nor leverages additional training data, and nor introduces additional hyper-parameters, while achieving superior performance with almost the same real-time efficiency.

In summary, the main contributions of this work are three-fold:

- We design an efficient ensemble tracking framework with a single backbone and multiple head networks. We further propose the model diversity and response diversity regularizations to encourage the diversity in both feature level and prediction level.
- As a small contribution, we introduce a response fusion module to adaptively integrate the results from individual models, facilitating the end-to-end training of the whole framework.
- We conduct extensive experiments on seven challenging benchmarks to validate the effectiveness of our approach. On the large-scale tracking datasets such as LaSOT [13], GOT-10k [14], and TrackingNet [15], our method achieves outstanding results in comparison with previous state-of-the-art trackers.

In the following of this paper, we describe the related work in Section II, our baseline approach in Section III, the proposed approach in Section IV, and experiments in Section V. Finally, we conclude the paper in Section VI.

II. Related Work

A. Visual Tracking

Given the initial target state, visual tracking aims to predict the target localization in successive frames. In the past decades, a variety of frameworks have been proposed in the tracking community. The correlation filter (CF) based methods [16], [17] solve the tracking task by ridge regression, which show attractive efficiency thanks to the closed-form solution in the Fourier domain. With the recent advancements [18]–[25], CF trackers achieve state-of-the-art performance. The classification based trackers [26]–[28] regard visual tracking as a binary classification problem and learn a binary classifier to distinguish the target from background candidates. Recently, Siamese network based tracking has attracted much popularity. The Siamese trackers [29]–[31] handle visual tracking via template matching and utilize a shared-weight network for similarity estimation. The recent improvements upon Siamese trackers include attention mechanism [32], model update [33]–[35], triplet loss [36], and target-aware fine-tuning [37]. Based on the SiamFC [29], SiamRPN [38] proposes another region proposal network (RPN) to estimate the target scale. By improving the training strategy [39] and adopting deeper backbone networks [40], [41], SiamRPN achieves remarkable performance. In recent works [12], [42], [43], regression based methods online learn a CNN kernel for target search, which is expected to be discriminative between the foreground and background contexts. In this work, we take the current state-of-the-art DiMP tracker [12] as our baseline approach.

B. Ensemble Tracking

To better cope with the complex scenarios and enhance the prediction robustness, ensemble trackers are widely explored. In [10], [11], off-the-shelf trackers are treated as the black boxes and their predicted bounding boxes are taken as the input of the fusion algorithms. The MEEM method [44] exploits the entropy-based relationship between the current tracker and its historical snapshots. In [5], three SVM based trackers with different features are constructed, which are adaptively selected according to the forward and backward trajectory consistency. To remove the redundancy among weak models, DEDT method [7] trains diverse models by generating an efficient set of artificial data. In CF trackers, since HCF [9] and C-COT [18] propose to fuse the response maps from different CNN levels, integrating multiple individual CFs has been a common technique in the following CF methods [23], [25], [45], [46]. The Staple tracker [2] consists of a correlation filter and a color-histogram based model to complement each other. The MCCT tracker [6] assigns the suitable weak experts based on their pair-wise and self-wise relationships. The SCT [3] and ACFN [47] approaches construct weak CF trackers with different features and design the attention mechanisms for model selection. In the classification based framework, BranchOut [8] extends MDNet [26] by introducing multiple fully-connected layers and selectively updates them to eliminate the overfitting issue. In [48], due to limited training samples, STCT aims to avoid the overfitting issue of the off-the-shelf CNNs (e.g., VGG) in visual tracking. STCT also uses binary masks to force the base learners (CNN channels) to learn different features. Under the Siamese tracking pipeline, SA-Siam approach [4] individually trains an appearance model and a semantic model for online combination. In POST tracker [49], an agent network trained via reinforcement learning is utilized to switch multiple Siamese trackers. Learning part-based models for joint tracking is also explored in the Siamese networks [50], [51].

Despite the recent success, the above approaches generally design the ensemble models and fusion strategies in a heuristic manner. The empirical fusion strategies such as the weighted manner [2], [4], [6], [9] require the manually tuned hyper-parameters. Some existing approaches rely on the expert intuition to choose features or design the base models [2]–[4], [6], [9], [47], and how to promote the diversity for more effective cooperation has been rarely investigated. In this work, we *automatically* learn diverse models by introducing additional penalty terms in the training stage and optimize the whole ensemble framework including the fusion module in an *end-to-end* manner.

Learning diverse visual representations are also explored in the CNN network training [52], [53]. For example, in [53], a DeCov loss is proposed to reduce the correlation of CNN channels to avoid overfitting. In [52], different orthogonality regularizations are explored to learn robust deep feature extractors. In this work, we aim to enlarge the prediction-level diversity for ensemble tracking. Although our method shares partial similarity with [52], [53], the motivation and technical details are quite different. Especially for our response diversity regularization and fusion module, they are specially designed for the visual tracking task.

III. Baseline Approach

Our baseline approach, DiMP tracker [12], adopts a Siamese-like pipeline consisting of a template branch for target model learning and a test branch for loss computation. The desirable target model f (i.e., a convolutional kernel) is first initialized using the foreground features. Then, different from Siamese trackers [29], [38], DiMP further optimizes the discriminative model f using both foreground and background contexts by minimizing the following loss:

$$L(f) = \frac{1}{|S_{\text{train}}|} \sum_{(x,c)\in S_{\text{train}}} \|r(x * f, c)\|^2 + \|\lambda f\|^2, \tag{1}$$

where x and c are the training samples and the corresponding ground-truth labels. The residual function $r(s, c)$ is a combined version of both regression loss and hinge loss, as follows:

$$r(s, c) = \upsilon_c \cdot (m_c s + (1 - m_c)\max(0, s) - y_c), \tag{2}$$

where υ_c, m_c, and y_c are the spatial weight, target mask, and regression target, respectively. These parameters are learned by the head network. Based on Eq. 1, the target model f can be optimized via the gradient descent:

$$f^{(i+1)} = f^{(i)} - \alpha \nabla L(f^{(i)}). \tag{3}$$

Instead of using a fixed learning rate α, DiMP computes an adaptive α for fast convergence:

$$\alpha = \frac{\nabla L(f^{(i)})^{\mathrm{T}} \nabla L(f^{(i)})}{\nabla L(f^{(i)})^{\mathrm{T}} Q^{(i)} \nabla L(f^{(i)})}, \tag{4}$$

where $Q^{(i)} = (J^{(i)})^{\mathrm{T}} J^{(i)}$ and $J^{(i)}$ is the Jacobian of the residuals at $f^{(i)}$.

After computing the target model $f^{(i)}$, the network is optimized by minimizing the classification loss of the test samples:

$$L_{\text{cls}} = \frac{1}{N_{\text{iter}}} \sum_{i=0}^{N_{\text{iter}}} \sum_{(x,c)\in S_{\text{test}}} \left\| \ell(x * f^{(i)}, z_c) \right\|^2, \tag{5}$$

where $\ell(s, z)$ is a hinge-like residual function and z_c is the ground-truth label of the test image. For more details, please refer to DiMP [12].

IV. Our Approach

A. Framework Overview

Despite the impressive performance of DiMP [12], it still struggles to comprehensively model the target appearance due to the limited model capacity and diversity. In this work, we go a step further by introducing multiple diverse head models to enrich the model representational capability of the original DiMP. The backbone network of DiMP is the widely used ResNet [54], which extracts powerful features for the subsequent target classification and bounding box regression. As shown in Figure 2, the head network of DiMP contains the following two parts with learnable parameters: (1) The feature refinement module, which leverages several convolutional layers to refine the backbone features to better suit the tracking task (i.e., the yellow boxes in Figure 2). (2) The model initializer and optimizer. These modules learn some free parameters (e.g., spatial weight and target mask in Section III) to guide the optimization of the final discriminative model.

The aforementioned modules are the core components in distinguishing the target from background objects. Nevertheless, relying on a single head network restricts the tracking performance. To this end, we construct multiple head models to complement each other for more convincing predictions. Furthermore, we force the independently optimized models and their output results to be diverse, as shown in Figure 2. Finally, relying on the response fusion module, we assess the quality of various tracking results and adaptively fuse them as the final prediction. By virtue of the joint training, our framework is free of special model designs and empirical fusion rules.

In the following, we first describe the diversity regularizations in subsection IV-B. Then, we present the response fusion module in subsection IV-C. We elaborate the training details in subsection IV-D. Finally, we briefly describe the online tracking process in subsection IV-E.

B. Enlarging Diversity in Ensemble Framework

To obtain diverse models with distinctive tracking capabilities, we first introduce the model diversity constraint. Then, we introduce another response diversity regularization to enlarge the differences among multiple tracking results.

1) Model Diversity: We construct N head networks on the top of a shared ResNet backbone. Constructing multiple head networks with different architectures may potentially acquire superior model diversity, while this needs special designs and time-consuming experimental validations. In our approach, we aim to automatically obtain diversified models

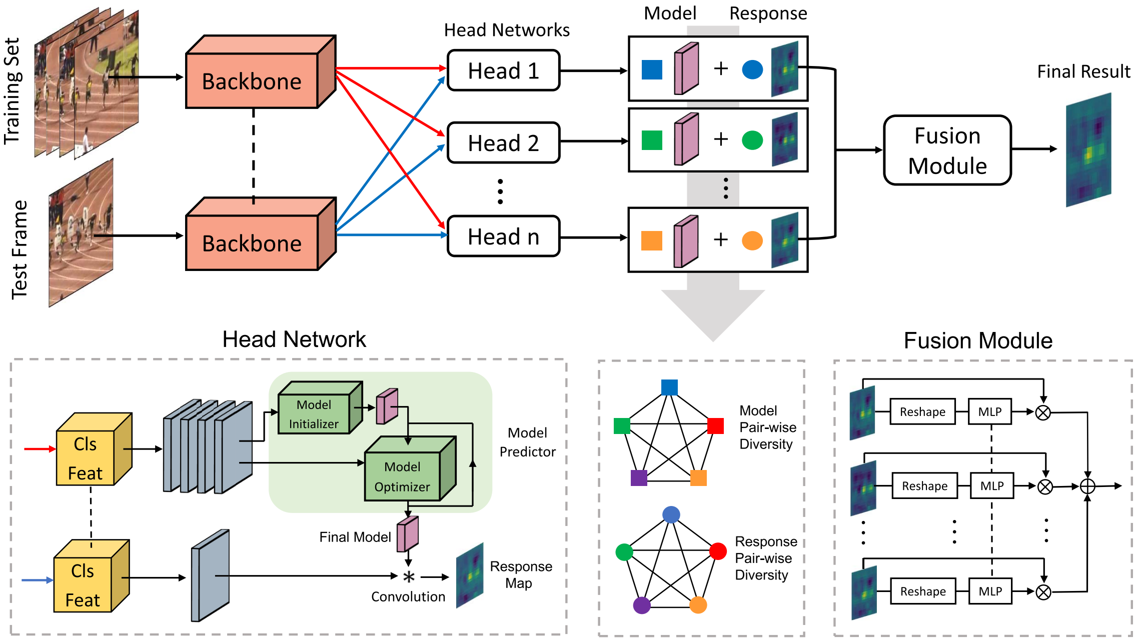


Fig. 2. Pipeline of the proposed ensemble tracking framework. Our method consists of a shared backbone network for feature extraction, multiple head networks to predict diverse results, and a fusion module to dynamically integrate the response maps.

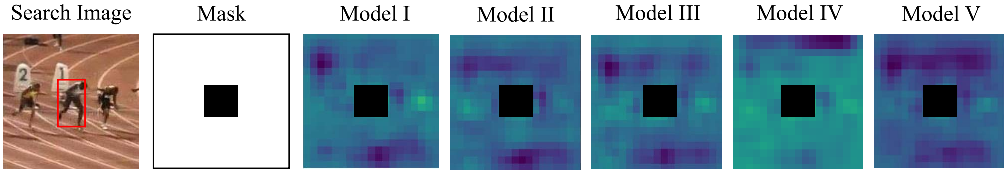

Search Image · Mask · Model I · Model II · Model III · Model IV · Model V

Fig. 3. In the training stage, we mask the target areas in the response maps for pair-wise divergence estimation. We aim to enlarge the prediction differences in the background areas in multiple tracking results to complement each other.

during the offline training stage in a data-driven manner. Thus, for simplicity, all the head models in our framework adopt the same architecture as in DiMP [12]. Ideally, the random initialization already enables these head networks to converge differently and lead to fairly distinctive models. To further encourage the diversity, we introduce an additional model regularization in the training stage.

Let f_j denote the discriminative model generated by the j-th head network, which can be computed via Eq. 1. We consider reducing the model pair-wise correlation to enlarge the diversity. To be specific, we leverage the widely used cosine distance to measure the correlation. The cosine distance between two vectors a and b can be calculated by $\cos(a, b) = \frac{a \cdot b}{\|a\| \cdot \|b\|}$. Given N models, the model diversity regularization term is defined as follows:

$$L_{\text{m-div}} = \frac{1}{N^2 - N} \sum_{j=1}^{N} \sum_{k=1, k \neq j}^{N} \left|\cos\left(\text{vec}(f_j), \text{vec}(f_k)\right)\right|, \quad (6)$$

where $\text{vec}(\cdot)$ denotes the vectoring operation that reshapes the convolutional kernel f to a one-dimensional vector. The range of cosine distance is $[-1, +1]$, where -1 and $+1$ denote the negative and positive correlations, respectively. We add the absolute function $|\cdot|$ to force the model vectors to be mutually orthometric without (positive or negative) correlation.

2) *Response Diversity:* In the experiments, we observe that multiple diverse models sometimes still yield similar tracking results. Ideally, all the head models are supposed to predict correctly and obtain the highest peak in the true target localization. At the same time, we also expect that the head models generate diversified results in the rest predictions except their highest peaks. Take Figure 3 as an example, Model II tends to be more easily misled by the left distractor object while model I and III show higher responses on the right distractor. In other words, the second-best, third-best, and other peak predictions from multiple models are expected to be different. After adaptively fusing these diverse response maps, the highest response peak (the target peak) will be strengthened and the rest sub-highest peaks (the distracting peaks) will be largely averaged and restrained. Thus, by fusing multiple reliable but diverse results, the distractor misguidance and occasional drift in visual tracking can be largely avoided. To this end, we introduce the response diversity regularization.

Given the ground-truth label in each frame, we first generate a binary mask to exclude the ground-truth target area, as shown in Figure 3. In the regression based trackers including DiMP, each search image has a Gaussian-shaped ground-truth map

z_c. Based on z_c, we generate the following binary mask:

$$M = \mathrm{H}(T - z_c), \tag{7}$$

where $\mathrm{H}(\cdot)$ is the Heaviside function (or step function) and T is a small threshold to separate the peak area from the rest background area in z_c.

Given the predicted response map R_j of the j-th head network, to focus on the distracting peaks in the background area, we (1) multiply R_j by the mask M to exclude the ground-truth target area and (2) ignore the negative values in the background. This can be achieved by the following equation:

$$\bar{R_j} = \mathrm{ReLU}(R_j \odot M), \tag{8}$$

where $\odot$ denotes the element-wise product and $\mathrm{ReLU}(\cdot)$ function maintains the subpeaks (i.e., positive values) in the background. Our goal is to enlarge the differences among $\{\bar{R_j}\}_{j=1}^{N}$. Similarly, we also adopt the cosine distance to compute the correlation of different response maps, as shown in Eq. 9.

$$L_{\text{r-div}} = \frac{1}{N^2 - N} \sum_{j=1}^{N} \sum_{k=1,k\neq j}^{N} \cos\left(\mathrm{vec}\left(\bar{R_j}\right), \mathrm{vec}\left(\bar{R_k}\right)\right). \tag{9}$$

Compared with the model diversity term, we omit the absolute function since our masked response maps $\{\bar{R_j}\}_{j=1}^{N}$ are nonnegative. By minimizing the above loss function, we force the subpeaks in the response maps $\{\bar{R_j}\}_{j=1}^{N}$ to localize in a variety of places. In this way, different predictions can potentially counteract each other at the nontarget areas and effectively alleviate the tracking drift caused by the distracting predictions.

3) Diversity Regularization: Finally, we equally combine the aforementioned Eq. 6 and Eq. 9 as the final regularization term to encourage the diversity during offline training: $L_{\text{div}} = L_{\text{m-div}} + L_{\text{r-div}}$.

C. Response Fusion Module

After computing the response maps from individual head networks, we integrate these results in an adaptive weighted manner. How to measure the response reliability has been widely explored and the empirical strategies such as peak-to-sidelobe ratio (PSR) [55] and average peak-to-correlation energy (APCE) [56] have been studied. However, these manually designed rules may not suit the ensemble task. As validated in the experiments, using the maximum value or PSR value of the response map cannot satisfactorily fuse the response maps. In this work, we leverage the CNN network to assess the quality of various responses. To be specific, we reshape the response map R_j to the one-dimensional vector and then use a two-layer multi-layer perceptron (MLP) to encode it to a value, representing the fusion weight. Then, the weight of each response map is normalized by the softmax function as follows:

$$w_j = \frac{\exp(\mathrm{MLP}(\mathrm{vec}(R_j)))}{\sum_{j=1}^{N} \exp(\mathrm{MLP}(\mathrm{vec}(R_j))))}, \tag{10}$$

where $\mathrm{MLP}(\cdot)$ denotes the multi-layer perceptron.

After obtaining the response weight, we fuse multiple independent results as the final response R_{final} in a weighted manner: $R_{\text{final}} = \sum_{j=1}^{N} w_j \cdot R_j$.

D. Offline Training

Given the training samples and corresponding ground-truth labels $(x, c) \in S_{\text{test}}$, similar to Eq. 5, we introduce an additional classification loss L_{fusion}. The loss L_{fusion}, as shown in Eq. 11, is based on the fused response map R_{final}, which aims to train the aforementioned fusion module.

$$L_{\text{fusion}} = \sum_{(x,c)\in S_{\text{test}}} \|\ell(R_{\text{final}}, z_c)\|^2. \tag{11}$$

To jointly train the backbone network, multiple head networks, and the fusion module, we integrate all the loss functions to form the final training objective, as follows:

$$L_{\text{final}} = L_{\text{reg}} + \lambda \cdot (L_{\text{cls}} + L_{\text{fusion}}) + L_{\text{div}}, \tag{12}$$

where λ is the weighting parameter as in DiMP, L_{reg} denotes the loss of the IoU predictor [57] and readers can refer to [57] for more details.

E. Online Tracking

The aforementioned backbone network, multiple head models, and fusion module are jointly learned in the offline training stage. After offline training, all the CNN parameters are frozen without further fine-tuning during online tracking. In the tracking stage, our method is similar to our baseline approach DiMP. Given the initial target location, multiple head networks learn diverse discriminative models (i.e., convolutional kernel f in Eq. 1) for visual tracking. After parallel prediction, the fusion module adaptively integrates diverse response maps. The highest peak in the fused response map is determined as the target localization. The target scale is further refined by the IoU predictor [57]. For each head model, we maintain an independent template ensemble. The head models are incrementally updated using their corresponding template ensembles.

V. Experiments

In this section, we first introduce the implementation details of our approach. Then, we verify the effectiveness of the proposed techniques by extensive ablative studies. Finally, we evaluate our approach on seven challenging benchmarks including TrackingNet [15], GOT-10k [14], LaSOT [13], VOT-2019 [58], NFS [59], OTB-2015 [60], and UAV123 [61].

A. Implementation Details

In the experiments, following DiMP [12], we utilize the training splits of TrackingNet [15], LaSOT [13], GOT-10k [14], and COCO [62] for offline training. The ADAM optimizer [63] is employed with an initial learning rate of 0.01, and use a decay factor 0.2 for every 15 epochs. We do not modify the network structure and simply initialize 5 identical head networks in our ensemble framework. The threshold T in Eq. 7 is set to 0.05. The weighting parameter λ in Eq. 11 is

TABLE I

COMPARISON WITH OUR INDIVIDUAL HEAD NETWORKS (I.E., MODEL I, II, III, IV, AND V) ON THE UAV123 [61], LASOT TEST SET [13], AND GOT-10K VALIDATION SET [14]. THE EVALUATION METRIC IS THE AREA-UNDER-CURVE (AUC) SCORE OF THE SUCCESS PLOT. OUR METHOD NOTABLY OUTPERFORMS THE INDIVIDUAL MODELS

	I	II	III	IV	V	**DET-18**
UAV123	62.9	63.3	63.1	63.4	63.0	**64.6**
LaSOT test	53.2	53.1	52.8	53.4	53.6	**55.2**
GOT-10k val	73.1	73.7	73.5	73.5	73.1	**75.3**

set to 100 as in DiMP. The multi-layer perceptron (MLP) has a hidden layer with 128 neurons, and the non-linear function of the hidden layer is ReLU. For other details, we just follow DiMP without modifications.

We denote our Diverse Ensemble Tracker as DET. Our tracker is implemented in Python using PyTorch and operates at about 40 frames per second (FPS) with a ResNet-18 backbone (denoted as DET-18) and 35 FPS with a ResNet-50 backbone (denoted as DET-50) on a single Nvidia GTX 1080Ti GPU. On each dataset, we run our method three times and take the average performance to alleviate the algorithm fluctuation.

B. Ablation Study

In this subsection, we utilize DET-18 to verify the effectiveness of the proposed components. The UAV123 [61], LaSOT test set [13], and GOT-10k validation set [14] are selected for ablative experiments, which contain 123, 270, and 160 videos, respectively. Note that GOT-10k validation set with diverse object classes is also suitable for assessing the generalization of our framework.

1) Performance of Individual Models: In Table I, we evaluate the performance of each single head network. Different models have the same network structure but are forced to model distinctive target representations. By fusing the results of these models, the final performance is obviously improved.

It is worth mentioning that DiMP is an extremely strong baseline, which has achieved a remarkable performance plateau on various benchmarks. Nevertheless, our ensemble framework makes a further advance and steadily improves the tracking accuracy by exploiting diverse representations. As for the tracking efficiency, individual models operate at about 48 FPS and our ensemble framework with 5 head models almost maintains the efficiency with a real-time speed of 40 FPS. Adopting more heads can further promote the performance, while we observe that 5 heads already makes a good balance of efficiency and accuracy.

2) Performance of More Head Models: In our experiments, we adopt 5 head networks for cooperation. In each frame, the main computational cost lies in the feature extraction by the deep backbone network (e.g., ResNet [54]). Multiple head networks bring an ignorable computational burden in the feed-forward pass. The main cost of multiple heads is their model updates. Nevertheless, DiMP [12] is robust and updates the discriminative model f in a sparse frequency (per 20 frames), which also ensures the high efficiency of our approach.

TABLE II

TRACKING PERFORMANCE OF DET-18 WITH DIFFERENT NUMBER OF HEAD MODELS ON THE ON THE UAV123 [61], LASOT TEST SET [13], AND GOT-10K VALIDATION SET [14]. THE EVALUATION METRIC IS THE AUC SCORE

Head Num	1	2	3	5	7	10
UAV123	63.4	64.1	64.0	64.6	64.8	64.8
LaSOT test	53.2	54.6	54.8	55.2	55.3	55.5
GOT-10k val	73.1	74.9	75.1	75.3	75.3	75.4
Speed	48	46	44	40	35	28

TABLE III

ANALYSIS OF DIFFERENT DIVERSITY REGULARIZATION TERMS ON THE UAV123 [61], LASOT TEST SET [13], AND GOT-10K VALIDATION SET [14]. BOTH MODEL DIVERSITY $L_{\text{M-DIV}}$ AND RESPONSE DIVERSITY $L_{\text{R-DIV}}$ REGULARIZATIONS IMPROVE THE OVERALL PERFORMANCE

	w/o L_{div}	w/o $L_{\text{m-div}}$	w/o $L_{\text{r-div}}$	**DET-18**
UAV123	63.5	64.1	63.9	**64.6**
LaSOT test	53.3	54.8	54.5	**55.2**
GOT-10k val	74.0	74.9	74.3	**75.3**

We test different numbers of the head network for ensemble tracking, as shown in Table II. Only one head network degenerates our framework to the standard DiMP. With more head models, the tracking performance steadily improves. But the performance gain becomes less obvious after adopting more than 3 heads, which means the performance has gradually reached saturation. We observe that the 5-head version achieves a good balance of efficiency and accuracy. Adopting 7 models yields identical or slightly better results compared with the 5-head version. By constructing 10 diverse head models, our method achieves further improvement (about 0.3% on the large-scale LaSOT [13]). However, more head networks also reduce online tracking efficiency. For example, the 10-head version only maintains a near real-time speed.

3) Effectiveness of the Diversity Regularizations: In Table III, we assess the effectiveness of the proposed model diversity and response diversity regularizations. Without any regularization (i.e., "w/o L_{div}" in Table III), the ensemble framework does not show obvious performance advantage compared with the individual models in Table I. For example, the "w/o L_{div}" version exhibits an AUC score of 63.5% on the UAV123, only slightly outperforming the base models in Table I. This indicates that simply training 5 models for fusion cannot acquire satisfactory performance gain due to the existence of model redundancy. From Table III, we can observe that both model diversity and response diversity enhance the tracking accuracy. The response regularization seems to be more effective since it directly enlarges the differences among multiple predictions. By combining these two regularization terms, superior performance can be obtained.

4) Effectiveness of the Fusion Module: In this work, we propose a simple fusion module. In the experiments, we also test other widely adopted techniques. In Table IV, "Average Fusion" means the simple average of all the response maps, "Max Fusion" denotes the maximum value of each response is determined as the fusion weight, and "PSR Fusion" represents the PSR value [55] of each response is set as the fusion weight. As shown in Table IV, our fusion module, benefitting from the end-to-end training, achieves the superior performance.

TABLE IV

COMPARISON WITH THE EMPIRICAL FUSION STRATEGIES (I.E., AVERAGE FUSION, MAX FUSION, AND PSR FUSION) ON THE UAV123 [61], LASOT TEST SET [13], AND GOT-10K VALIDATION SET [14] IN TERMS OF AUC SCORE

	Individual	Avg	Max	PSR	**DET-18**
UAV123	62.9∼63.4	63.6	64.1	63.7	**64.6**
LaSOT test	52.8∼53.6	53.5	54.1	54.4	**55.2**
GOT-10k val	73.1∼73.7	74.1	74.8	74.2	**75.3**

TABLE V

ABLATION STUDY BASED ON THE SIAMFC-LIKE PIPELINE ON THE UAV123 [61], LASOT TEST SET [13], AND GOT-10K VALIDATION SET [14] IN AUC SCORE. BOTH MODEL DIVERSITY $L_{\text{M-DIV}}$ AND RESPONSE DIVERSITY $L_{\text{R-DIV}}$ REGULARIZATIONS IMPROVE THE OVERALL PERFORMANCE

	w/o L_{div}	w/o $L_{\text{m-div}}$	w/o $L_{\text{r-div}}$	**Ours**
UAV123	60.1	62.9	63.1	**63.5**
LaSOT test	48.4	52.2	51.9	**53.1**
GOT-10k val	68.1	73.2	72.7	**73.8**

5) Improvements Upon a SiamFC-like Baseline: To verify the generalization capability of our approach, we further adopt our ensemble framework to a clean SiamFC-like baseline. To be specific, similar to DiMP-18, we still use the ResNet-18 for feature extraction and IoUNet for target scale estimation. Differently, we simply utilize the target feature as the template kernel to convolve with the search feature for response generation, which is identical to the cross-correlation in SiamFC [29]. Without sophisticated model optimization techniques in DiMP [12], we can better assess the effectiveness of our method. Our head models consist of two convolutional layers (3 × 3 Conv + BN) to map the shared backbone features to diverse feature subspaces. Similar to our DET-18, we also construct five diverse head models.

As shown in Table V, we can observe that without our regularization terms, simply combining 5 models (i.e., "w/o L_{div}") cannot achieve satisfactory results. Our model and response regularizations consistently improve the baseline performance. Finally, with both $L_{\text{m-div}}$ and $L_{\text{r-div}}$, our complete version significantly outperforms the baseline by 5.7% AUC on the GOT-10k validation set and 4.7% AUC on the LaSOT test set. Note that SiamFC-like tracking baseline is simple without bells and whistles, where the performance gains can be attributed to our designed diversity constraints.

C. State-of-the-Art Comparisons

In this subsection, we present the comparison results with recent methods on the following 7 challenging datasets. Our baseline tracker DiMP has already achieved remarkable results on these benchmarks. Despite the limited improvement room, our method still steadily improves DiMP on *all* the datasets.

1) TrackingNet [15]: TrackingNet is a recently released large-scale dataset. We evaluate our method on the test set of TrackingNet, which consists of 511 videos. In this benchmark, we compare our approaches with the state-of-the-art C-RPN [64], SPM [65], ATOM [57], SiamRPN++ [40], and DiMP-50. As shown in Table VI, the proposed DET-50 achieves a normalized precision score of 81.0% and a success score of 75.5%, which is superior to all previous top-performing trackers such as DiMP-50 and SiamRPN++. Specially, our DET-50 improves the baseline DiMP-50 by relative gains of 2.0% in AUC score.

TABLE VI

STATE-OF-THE-ART COMPARISON ON THE TRACKINGNET TEST SET [15] IN TERMS OF PRECISION (PREC), NORMALIZED PRECISION (N. PREC), AND SUCCESS (AUC)

Trackers	Prec	N. Prec	AUC
ECO [66]	49.2	61.8	55.4
SiamFC [29]	53.3	66.6	57.1
CFNet [67]	53.3	65.4	57.8
MDNet [26]	56.5	70.5	60.6
UPDT [46]	55.7	70.2	61.1
DaSiamRPN [39]	59.1	73.3	63.8
C-RPN [64]	61.9	74.6	66.9
SPM [65]	66.1	77.8	71.2
ATOM [57]	64.8	77.1	70.3
SiamRPN++ [40]	**69.4**	80.0	73.3
DiMP-18 [12]	66.6	78.5	72.3
DiMP-50 [12]	68.7	**80.1**	74.0
DET-18	68.6	79.7	**74.1**
DET-50	**70.3**	**81.0**	**75.5**

TABLE VII

STATE-OF-THE-ART COMPARISON ON THE GOT-10K TEST SET [14] IN TERMS OF AVERAGE OVERLAP (AO), AND SUCCESS RATES (SR) AT OVERLAP THRESHOLDS 0.5 AND 0.75

Trackers	$\text{SR}_{0.50}$	$\text{SR}_{0.75}$	AO
MDNet [26]	30.3	9.9	29.9
CF2 [9]	29.7	8.8	31.5
ECO [66]	30.9	11.1	31.6
C-COT [18]	32.8	10.7	32.5
GOTURN [31]	37.5	12.4	34.7
SiamFC [29]	35.3	9.8	34.8
SiamFCv2 [67]	40.4	14.4	37.4
SiamRPN [38]	54.9	25.3	46.3
SPM [65]	59.3	35.9	51.3
ATOM [57]	63.4	40.2	55.6
DiMP-18 [12]	67.2	44.6	57.9
DiMP-50 [12]	**71.7**	**49.2**	**61.1**
DET-18	69.0	43.4	58.2
DET-50	**74.7**	**50.4**	**63.0**

2) GOT-10k [14]: GOT-10k is a large-scale dataset including more than 10,000 videos. We test our methods on the test set of GOT-10k with 180 sequences. The main characteristic of GOT-10k is that the test set does not have overlap in object classes with the train set, which is designed to assess the generalization of the visual tracker. Similar to TrackingNet, GOT-10k also hides the ground-truth labels of the test set. The comparison results in Table VII are obtained via the official server. On this benchmark, as shown in Table VII, our DET-50 exhibits the best performance with a $\text{SR}_{0.50}$ score of 74.7% and an AO score of 63.0%, outperforming our baseline DiMP-50 by relative gains of 4.2% in $\text{SR}_{0.50}$ and 3.1% in AO, verifying the strong generalization of our tracker.

3) LaSOT [13]: LaSOT is a large-scale benchmark consisting of 1200 videos. The average video length in LaSOT is about 2500 frames, which is more challenging than classic short-term tracking datasets. Therefore, how to handle the drastic target appearance changes is vital in this dataset. We evaluate our approach on the LaSOT test set with 280 videos.

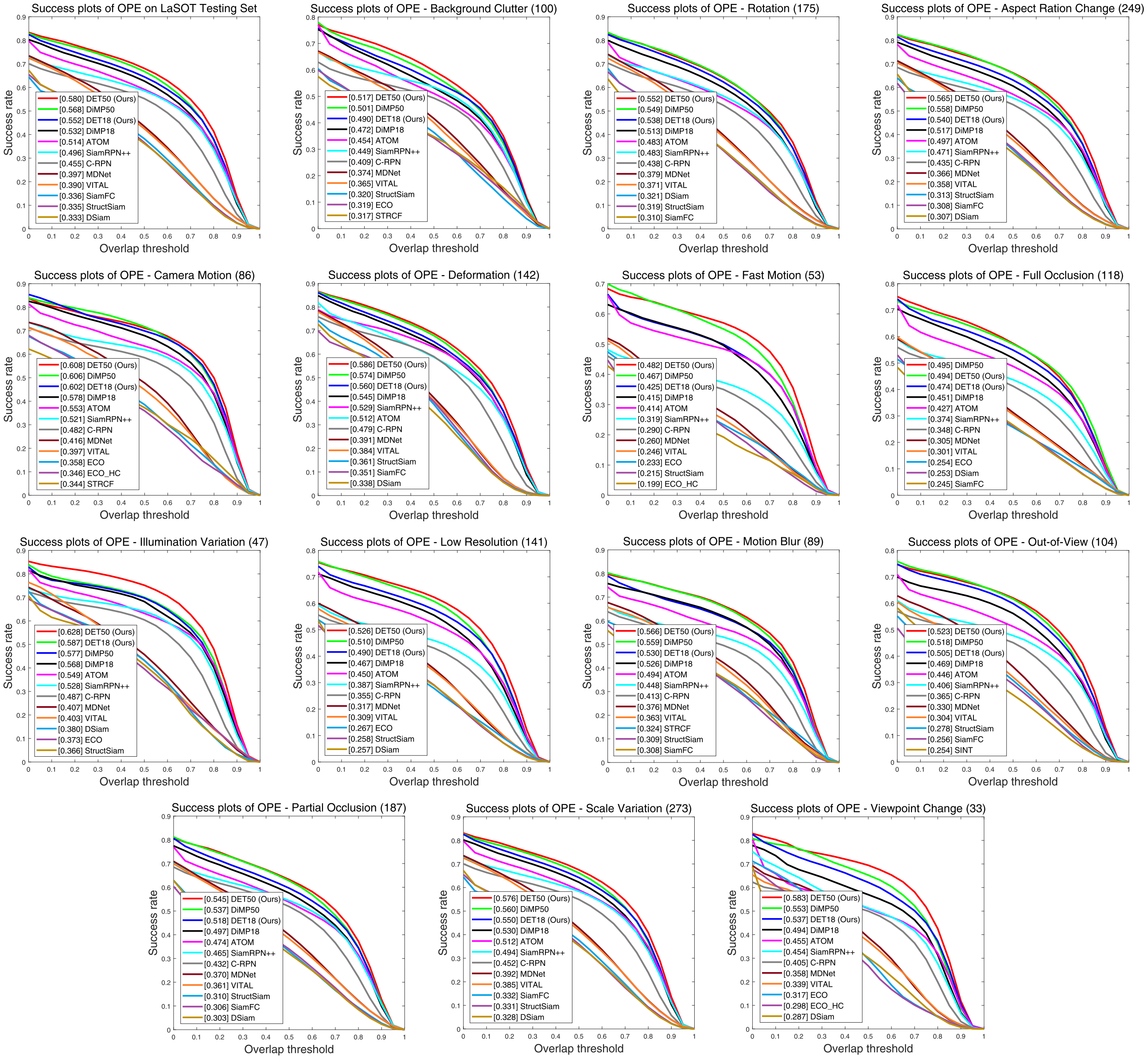


Fig. 4. Success plots of the state-of-the-art trackers on the LaSOT test set [13]. The legend shows the AUC score. In the upper-left figure, we present the overall performance of the methods. In the rest figures, we show success plots on the challenging attributes.

TABLE VIII
THE ACCURACY, ROBUSTNESS (FAILURE RATE) AND EXPECTED AVERAGE OVERLAP (EAO) OF STATE-OF-THE-ART METHODS ON THE VOT-2019 [58]

	SPM [65]	ROAM++ [58]	SiamRPN++ [40]	SiamMask [68]	ATOM [57]	SiamDW [41]	DCFST [58]	DiMP-50 [12]	**DET-50**
Accuracy (↑)	0.577	0.561	0.599	0.594	**0.603**	**0.600**	0.578	0.594	0.595
Robustness (↓)	0.507	0.438	0.482	0.461	0.411	0.467	0.321	**0.278**	**0.237**
EAO (↑)	0.275	0.281	0.285	0.287	0.292	0.299	0.361	**0.379**	**0.394**

The success plots of the state-of-the-art methods are shown in Figure 4. As reported in LaSOT [13], MDNet [26] is the previous best tracker on this benchmark. Our DET-50 achieves an AUC score of 58.0%, surpassing MDNet by a considerable margin of 18.3% in AUC. For completeness, we also include the recently proposed C-RPN, SiamRPN++, ATOM, DiMP-18, and DiMP-50 for comparison. Our DET-50 outperforms all previous methods. Compared with the recent C-RPN, SiamRPN++, ATOM, and DiMP-18, our CARE surpasses them by 9.2%, 5.1%, 3.3%, and 1.2% in AUC, respectively. Compared with our strong baselines DiMP-50 and DiMP-18, the proposed DET-50 and DET-18 outperform them by relative gains of 2.1% and 3.8%, respectively.

To further investigate how our method works, we exhibit some tracking attributes that benefit from our ensemble framework. In Figure 4, we can observe that our main performance

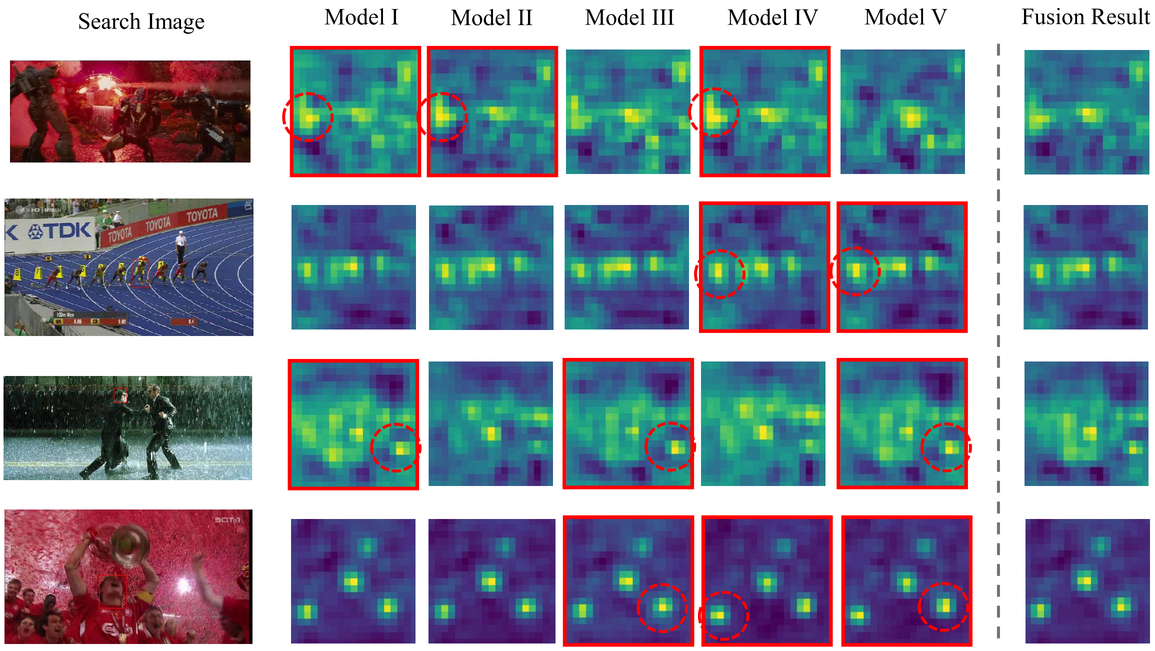


Fig. 5. Visualization of the response maps from the individual head networks and fusion module. Different models yield distinctive tracking responses. The prediction failures are highlighted by the red circles. By adaptive fusion, the final prediction effectively restrains the prediction biases and promotes the tracking robustness. The videos from top to down are *Ironman*, *Walking*, *Bolt*, *Matrix*, and *Soccer* from OTB-2015 [60].

gains are from the attributes such as viewpoint change (VC), low resolution (LR), illumination variation (IV), fast motion (FM), and background clutter (BC). A common characteristic in these attributes is that the target generally undergoes drastic appearance changes. Thanks to the diverse models for manifold representations, our ensemble framework shows superior accuracy in the above scenarios. On the rest attributes such as deformation, our trackers also exhibit better or similar results in comparison with their baselines.

4) VOT-2019 [58]: VOT-2019 benchmark consists of 60 videos for short-term tracking evaluation. VOT-2019 updates VOT-2018 by replacing 10 least challenging videos. The tracking performance in VOT is evaluated using Expected Average Overlap (EAO), which takes both accuracy (average overlap over successful frames) and robustness (failure rate) into account. We evaluate our DET-18 and DET-50 with state-of-the-art participants in VOT-2019 for comparison. Table VIII shows the accuracy, robustness, and EAO scores of different trackers. Compared with DiMP-50, our DET-50 shows similar tracking accuracy but exhibits a 14.7% lower failure rate (i.e., robustness score). By reducing the tracking variance, our ensemble framework improves DiMP-50 by a relative gain of 4.0% in EAO. Among all the comparison methods, our DET-50 also achieves the best robustness. Compared with other recent deep trackers with the ResNet-50 backbone, our DET-50 significantly surpasses SiamRPN, DWSiam [41] and SiamMask [68] by relative gains of 38.2%, 31.8%, and 37.3% in EAO, respectively. The VOT-2019 challenge winner (i.e., DRNet) shows an EAO score of 0.395 [58]. Overall, the proposed DET-50 is comparable with the current top-performing trackers with a very competitive EAO of 0.394.

5) Need for Speed [59]: NFS dataset contains 100 videos with fast-moving objects. We evaluate our approaches on the 30 FPS version of this benchmark. As shown in Table IX, our DET-50 achieves an AUC score of 63.4%, surpassing all the state-of-the-art methods such as UPDT [46], ATOM, and DiMP-50. As shown in Table IX, our DET-50 and DET-18 exhibit competitive tracking speeds of 35 FPS and 40 FPS, respectively, which are slightly lower than their baseline approaches. As for the tracking accuracy, our methods steadily outperform their baselines. Overall, our ensemble framework achieves a good balance of performance and efficiency.

6) UAV123 [61]: UAV123 dataset includes 123 challenging videos collected by the UAV platforms. In Table IX, we present the tracking results of the state-of-the-art trackers. Compared with these approaches, our DET-50 achieves the best result with an AUC score of 66.4%. Note that DET-50, DiMP-50, and SiamRPN++ all adopt the same backbone network (i.e., ResNet-50), our superior performance verifies the effectiveness of the proposed ensemble framework.

7) OTB-2015 [60]: OTB-2015 is a popular benchmark with 100 videos. On this dataset, as shown in Table IX, our DET-18 and DET-50 achieve the AUC scores of 67.8% and 69.2%, respectively. Compared with the top-performing trackers on this dataset such as ECO [66] and UPDT [46], our DET-50 overall exhibits competitive performance with a real-time speed.

D. Result Visualization

In Figure 5, we present some visualization examples of our ensemble framework (DET-50). From the results, we can observe that multiple head models generate diverse tracking

TABLE IX

STATE-OF-THE-ART COMPARISON ON THE NFS [59], OTB-2015 [60], AND UAV123 [61] DATASETS IN TERMS OF AUC SCORE

	SiamFC [29]	MDNet [26]	C-COT [18]	ECO [66]	ATOM [57]	UPDT [46]	SiamRPN++ [40]	DiMP-18 [12]	DiMP-50 [12]	**DET-18**	**DET-50**
NFS	-	42.9	48.8	46.6	58.4	53.7	50.2	61.0	**62.0**	61.8	**63.4**
OTB-2015	58.2	67.8	68.2	69.1	66.9	**70.2**	**69.6**	66.0	68.4	67.8	69.2
UAV123	49.8	52.8	51.3	52.2	63.5	54.5	61.3	63.4	64.5	**64.6**	**66.4**
FPS	86	1	0.3	8	35	<1	30	48	42	40	35

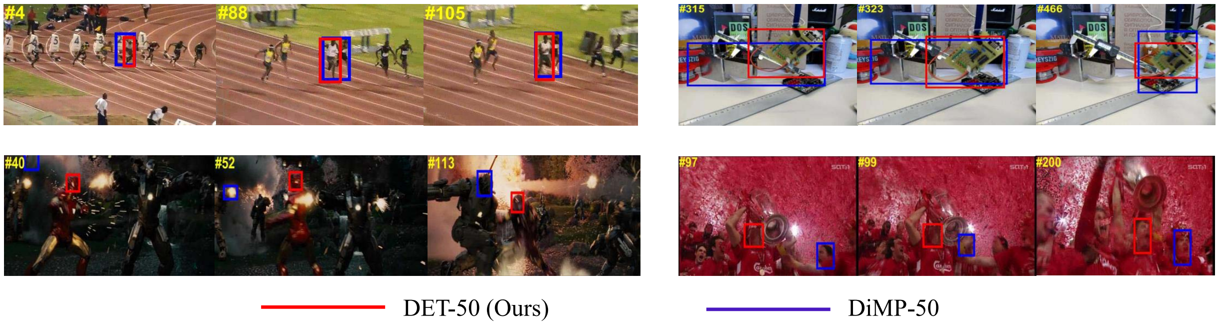


Fig. 6. Tracking results of our DET-50 and its baseline approach DiMP. Our ensemble framework, by constructing diverse models, exhibits superior tracking robustness.

response maps thanks to the proposed diversity regularizations. Even though some head networks generate unsatisfactory results, our fusion module still shows robust fusion by adaptively weighing multiple predictions. By fusing these diversified predictions, accidental tracking drift can be largely alleviated.

It is interesting that our fusion module potentially judges the reliability of different results to dynamically fuse them. For example, in the first row in Figure 5 (video *Ironman*), most individual response maps fail to localize the target with multiple subpeaks. However, our fusion module weighs the reliable response more (i.e., the response from Modle III and V, which have high peaks in the center). In the last two videos *Matrix* and *Soccer*, most models still drift to the distracting objects. By virtue of our fusion module, the final fused results exhibit better robustness with less ambiguity. The tracking examples in Figure 6 also indicate the high robustness of our ensemble framework compared with the baseline model DiMP-50.

VI. CONCLUSION

In this paper, we propose a simple, clean yet effective ensemble framework for robust visual tracking. Based on the shared backbone features, we construct multiple head models to cooperatively discriminate the target. To obtain accurate and diversified predictions for complementary fusion, we propose the model diversity and response diversity regularizations during training. The model diversity regularization enlarges the representational divergences among multiple head networks, while the response diversity encourages the prediction differences in the background areas. To adaptively integrate the response maps from parallel head models, we further introduce a fusion module to achieve the joint optimization of the whole pipeline. Extensive ablation studies verify the effectiveness of the proposed techniques. On the recent large-scale tracking datasets such as LaSOT, TrackingNet, and GOT-10k, our approach outperforms previous state-of-the-art trackers while running in real-time.

Our future works include searching diversified head networks with mutually different architectures (e.g., using neural architecture search (NAS) techniques) and training multiple models using different data to better explore the potential of ensemble tracking.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, p. 13, 2006.

[2] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1401–1409.

[3] J. Choi, H. J. Chang, J. Jeong, Y. Demiris, and J. Y. Choi, "Visual tracking using attention-modulated disintegration and integration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4321–4330.

[4] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold siamese network for real-time object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4834–4843.

[5] D.-Y. Lee, J.-Y. Sim, and C.-S. Kim, "Multihypothesis trajectory analysis for robust visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5088–5096.

[6] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4844–4853.

[7] K. Meshgi, S. Oba, and S. Ishii, "Efficient diverse ensemble for discriminative co-tracking," in *Proc. CVPR*, 2018, pp. 4814–4823.

[8] B. Han, J. Sim, and H. Adam, "BranchOut: Regularization for online ensemble tracking with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3356–3365.

[9] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3074–3082.

[10] N. Wang and D. Y. Yeung, "Ensemble-based tracking: Aggregating crowdsourced structured time series data," in *Proc. ICML*, 2014, pp. 1107–1115.

[11] C. Bailer, A. Pagani, and D. Stricker, "A superior tracking approach: Building a strong tracker through fusion," in *Proc. ECCV*, 2014, pp. 170–185.

[12] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6182–6191.

[13] H. Fan *et al.*, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5374–5383.

[14] L. Huang, X. Zhao, and K. Huang, "GOT-10k: A large high-diversity benchmark for generic object tracking in the wild," 2018, *arXiv:1810.11981*. [Online]. Available: http://arxiv.org/abs/1810.11981

[15] M. Müller, A. Bibi, S. Giancola, S. Al-Subaihi, and B. Ghanem, "Trackingnet: A large-scale dataset and benchmark for object tracking in the wild," in *Proc. ECCV*, 2018, pp. 300–317.

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

[17] M. Danelljan, G. Häger, F. Shahbaz Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, p. 4.

[18] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. ECCV*, 2016, pp. 472–488.

[19] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1135–1143.

[20] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1396–1404.

[21] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Correlation tracking via joint discrimination and reliability learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 489–497.

[22] M. Zhang *et al.*, "Visual tracking via spatially aligned correlation filters network," in *Proc. ECCV*, 2018, pp. 469–485.

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

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

[25] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4670–4679.

[26] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016., pp. 4293–4302.

[27] Y. Song *et al.*, "VITAL: VIsual tracking via adversarial learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8990–8999.

[28] I. Jung, J. Son, M. Baek, and B. Han, "Real-time mdnet," in *Proc. ECCV*, 2018, pp. 83–98.

[29] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. ECCV Workshops*, 2016, pp. 850–865.

[30] R. Tao, E. Gavves, and A. W. M. Smeulders, "Siamese instance search for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1420–1429.

[31] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 fps with deep regression networks," in *Proc. ECCV*, 2016, pp. 749–765.

[32] Q. Wang, Z. Teng, J. Xing, J. Gao, W. Hu, and S. Maybank, "Learning attentions: Residual attentional siamese network for high performance online visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4854–4863.

[33] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1763–1771.

[34] T. Yang and A. B. Chan, "Learning dynamic memory networks for object tracking," in *Proc. ECCV*, 2018, pp. 152–167.

[35] L. Zhang, A. Gonzalez-Garcia, J. V. D. Weijer, M. Danelljan, and F. S. Khan, "Learning the model update for siamese trackers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4010–4019.

[36] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *Proc. ECCV*, 2018, pp. 459–474.

[37] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1369–1378.

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

[39] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware siamese networks for visual object tracking," in *Proc. ECCV*, 2018, pp. 101–117.

[40] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of siamese visual tracking with very deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4282–4291.

[41] Z. Zhang and H. Peng, "Deeper and wider siamese networks for real-time visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4591–4600.

[42] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau, and M.-H. Yang, "CREST: Convolutional residual learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2555–2564.

[43] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M.-H. Yang, "Deep regression tracking with shrinkage loss," in *Proc. ECCV*, 2018, pp. 353–369.

[44] J. Zhang, S. Ma, and S. Sclaroff, "Meem: Robust tracking via multiple experts using entropy minimization," in *Proc. ECCV*, 2014, pp. 188–203.

[45] T. Zhang, C. Xu, and M.-H. Yang, "Multi-task correlation particle filter for robust object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4335–4343.

[46] G. Bhat, J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg, "Unveiling the power of deep tracking," in *Proc. ECCV*, 2018, pp. 483–498.

[47] J. Choi, H. Jin Chang, S. Yun, T. Fischer, Y. Demiris, and J. Y. Choi, "Attentional correlation filter network for adaptive visual tracking," in *Proc. CVPR*, 2017, pp. 4807–4816.

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

[49] N. Wang, W. Zhou, G. Qi, and H. Li, "Post: Policy-based switch tracking," in *Proc. AAAI*, 2020, pp. 12184–12191.

[50] Y. Zhang, L. Wang, J. Qi, D. Wang, M. Feng, and H. Lu, "Structured siamese network for real-time visual tracking," in *Proc. ECCV*, 2018, pp. 351–366.

[51] J. Gao, T. Zhang, and C. Xu, "Graph convolutional tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4649–4659.

[52] N. Bansal, X. Chen, and Z. Wang, "Can we gain more from orthogonality regularizations in training deep CNNs?" in *Proc. NeurIPS*, 2018, pp. 4261–4271.

[53] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, and D. Batra, "Reducing overfitting in deep networks by decorrelating representations," in *Proc. ICLR*, 2016, pp. 1–5.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[55] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.

[56] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4021–4029.

[57] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4660–4669.

[58] M. Kristan *et al.*, "The seventh visual object tracking vot 2019 challenge results," in *Proc. ICCV Workshops*, 2019, pp. 1–36.

[59] H. Kiani Galoogahi, A. Fagg, C. Huang, D. Ramanan, and S. Lucey, "Need for speed: A benchmark for higher frame rate object tracking," in *Proc. ICCV*, 2017, pp. 1125–1134.

[60] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[61] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. ECCV*, 2016, pp. 445–461.

[62] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.

[63] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[64] H. Fan and H. Ling, "Siamese cascaded region proposal networks for real-time visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7952–7961.

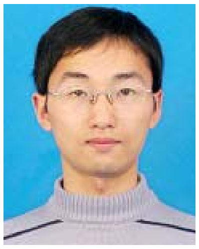
[65] G. Wang, C. Luo, Z. Xiong, and W. Zeng, "SPM-tracker: Series-parallel matching for real-time visual object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3643–3652.

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

[67] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2805–2813.

[68] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1328–1338.

Ning Wang (Graduate Student Member, IEEE) received the B.E. degree in communication engineering from Tianjin University (TJU), in 2016. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineer and Information Science, University of Science and Technology of China (USTC). His research interest is computer vision and his current research work is focused on video object tracking.



Wengang Zhou (Senior Member, IEEE) received the B.E. degree in electronic information engineering from Wuhan University, China, in 2006, and the Ph.D. degree in electronic engineering and information science from the University of Science and Technology of China (USTC), China, in 2011.

From September 2011 to 2013, he worked as a Post-Doc Researcher with the Computer Science Department, University of Texas at San Antonio. He is currently a Professor with the EEIS Department, USTC. His research interests include multimedia information retrieval and computer vision.

Houqiang Li (Fellow, IEEE) received the B.S., M.Eng., and Ph.D. degrees in electronic engineering from the University of Science and Technology of China, Hefei, China, in 1992, 1997, and 2000, respectively.

He is currently a Professor with the Department of Electronic Engineering and Information Science. His research interests include multimedia search, image/video analysis, video coding and communication. He has authored and coauthored over 200 papers in journals and conferences. He is the winner of National Science Funds (NSFC) for Distinguished Young Scientists, the Distinguished Professor of Changjiang Scholars Program of China, and the Leading Scientist of Ten Thousand Talent Program of China. He served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2010 to 2013. He served as the TPC Co-Chair of VCIP 2010, and he will serve as the General Co-Chair of ICME 2021. He was a recipient of the National Technological Invention Award of China (second class) in 2019 and a recipient of the National Natural Science Award of China (second class) in 2015. He was a recipient of the Best Paper Award for VCIP 2012, a recipient of the Best Paper Award for ICIMCS 2012, and a recipient of the Best Paper Award for ACM MUM in 2011.