Complex & Intelligent Systems (2023) 9:5793–5806
https://doi.org/10.1007/s40747-023-01043-1

ORIGINAL ARTICLE

Transformer tracking with multi-scale dual-attention

Jun Wang[1] · Changwang Lai[1] · Wenshuang Zhang[1] · Yuanyun Wang[1] · Chenchen Meng[2]

Received: 22 November 2022 / Accepted: 9 March 2023 / Published online: 7 April 2023

Abstract
Transformer-based trackers greatly improve tracking success rate and precision rate. Attention mechanism in Transformer can fully explore the context information across successive frames. Nevertheless, it ignores the equally important local information and structured spatial information. And irrelevant regions may also affect the template features and search region features. In this work, a multi-scale feature fusion network is designed with box attention and instance attention in Encoder–Decoder architecture based on Transformer. After extracting features, the local information and structured spatial information is learnt by multi-scale box attention, and the global context information is explored by instance attention. Box attention samples grid features from the region of interest. Therefore, it effectively focuses on the region of interest (ROI) and avoids the influence of irrelevant regions in feature extraction. At the same time, instance attention can also pay attention to the context information across successive frames, and avoid falling into local optimum. The long-range feature dependencies are learned in this stage. Extensive experiments are conducted on six challenging tracking datasets to demonstrate the superiority of the proposed tracker MDTT, including UAV123, GOT-10k, LaSOT, VOT2018, TrackingNet, and NfS. In particular, the proposed tracker achieves AUC score of 64.7% on LaSOT, 78.1% on TrackingNet and precision score of 89.2% on UAV123, which outperforms the baseline and most recent advanced trackers.

Keywords Transformer · Attention mechanism · Context information · Local information · Feature fusion network

Introduction

Visual tracking has an important research significance in computer vision [1]. It has a large number of practical applications, such as video surveillance, automatic driving, and visual localization. The goal of visual tracking is to use the target information given in the previous frame to predict the position of target in the next frame. Visual tracking still confronts many challenges due to a lot of complicating factors in real-world scenes, such as partial occlusion, out-of-view, background clutter, viewpoint change, scale variation, *etc*.

Recently, Vaswani et al. [2] first propose an attention mechanism based on Transformer for nature language processing. The Transformer explores long-range dependencies in sequences by computing the attention weights with triples (i.e., query, key, and value). Based on the excellent ability of attention mechanism in feature fusion, Transformer structures have successfully been introduced to visual tracking and achieved encouraging results. Wang et al. [3] propose an encoder–decoder-based tracking framework to explore the rich context information across successive frames. It is a meaningful attempt and achieves great success.

Existing Transformer-based trackers use CNN (Convolutional Neural Network) as a backbone network for feature extracting. CNN focuses more on local information, and ignores the global information and the connection between them. These disadvantages may have some impacts on tracking performance, especially in the complicated tracking scenes, such as severe occlusion, out-of-view, and drastic illumination. Even these challenges can lead to tracking drift or failure. In Transformer-based trackers, the encoder–decoder structure compensates for this deficiency of CNN, and the global context information is fully explored. However, the structured spatial information is not adequately exploited. How to effectively explore the context information across successive frames without losing a lot of useful

✉ Yuanyun Wang
wangyy_abc@163.com

Jun Wang
wangjun012778@126.com

[1] School of Information Engineering, Nanchang Institute of Technology, Nanchang 330029, China

[2] NSFOCUS Technologies Group Co., Ltd, Beijing 100089, China

spatial information becomes a crucial factor to improve the tracking performance.

In this paper, a novel multi-scale dual-attention-based tracking method is proposed to further explore structured spatial information. The proposed method is inspired by the encouraging work of TrDiMP [3], which first introduces Transformer to the tracking field and builds a bridge to explore context information across successive frames. Different from TrDiMP, the proposed method uses a novel feature fusion network, which can not only explore context information, but also fully explore local information and structured spatial information across successive frames. The proposed method predicts the ROI by applying a geometric transformation to the reference window. So that it can focus more on the predicted regions. By this way, the structured spatial information can be fully explored. In addition, the instance attention is introduced to the decoder structure, which can focus more on the global context information across successive frames. The proposed tracker can make the attention module more flexible, and can quickly focus on the region of interest. The proposed tracker performs well on six tracking benchmarks, including UAV123 [4], VOT2018 [5], GOT-10k [6], NfS [7], LaSOT [8], and TrackingNet [9].

In summary, the main contributions of this work can be summarized as follows:

- A Transformer-based multi-scale feature fusion network with dual attentions is designed, namely, box attention and instance attention. With the feature fusion network, we can quickly obtain multiple bounding boxes with high confidence scores in encoder, and then refine and obtain the predicted bounding boxes in decoder.
- In the Transformer-based feature fusion network, box attention effectively extracts the structured spatial information, and instance attention explores the temporal context information by the Encoder–Decoder architecture. By this way, the tracker MDTT can explore enough global context information across successive frames while focusing on more local responses.
- A novel Transformer tracking framework with multi-scale dual-attention is proposed, which can effectively deal with complicated challenges, such as background clutter, fully occlusion, and viewpoint change. We have verified the effectiveness of the fusion network and tested the proposed tracker MDTT on six challenging tracking benchmark datasets. The experimental results on these test datasets show that MDTT achieves robust tracking performance while running on real-time tracking speed.

Related work

Siamese-based visual tracking

Recently, trackers based on Siamese networks achieve a well balance between tracking speed and accuracy. As the pioneering work, SiamFC [10] uses two branches (i.e., template branch and search branch) to extract the template image features and search region image features, respectively. It trains an end-to-end tracking network and computes the score maps by cross-correlation. SiamFC achieves superior tracking performance on some current tracking benchmarks. Based on SiamFC, Dong et al. [11] add a triplet loss to Siamese network as the training strategy. To save the time of multi-scale testing, SiamRPN [12] uses the Region Proposal Network (RPN) structure in Siamese tracking.

SiamFC and most Siamese-based trackers usually use the shallow AlexNet as the feature extractor. Li et al. [13] propose a layer-based feature aggregation structure to calculate similarity, which is helpful to obtain more accurate similarity maps from multiple layers. Instead of using AlexNet as the backbone, Abdelpakey et al. [14] design a new network structure with Dense blocks that reinforces template features by adding self-attention mechanism.

Due to the use of deep convolution operation in feature extraction, trackers usually only focus on local regions of images. Additionally, in Siamese-based trackers, the cross-correlation operation is usually used as the similarity matching method. It focuses more on the local information than global information, and easily traps in local optimum.

Attention mechanisms in computer vision

In recent years, attention mechanisms are increasingly being used in various fields of computer vision. It focuses on important information and ignores the irrelevant information. Attention mechanisms can be divided into channel attention, spatial attention, mixed attention, frequency domain attention, self-attention, and so on.

SENet [15] proposes to learn from the channel dimension to get the importance of each channel. Woo et al. [16] combine the channel attention and spatial attention, which can effectively help information transfer across the network by learning to reinforce or suppress relevant feature information. Self-attention uses a particular modeling method to explore the global context information. However, since the self-attention needs to capture the global context information, it will focus less on the local region. Xiao et al. [17] design a federated learning system, which uses attention mechanism and long short-term memory to explore the global relationships hidden in the data. Xing et al. [18] propose a robust semisupervised model, which simplifies

semisupervised learning techniques and achieves excellent performance.

Transformer is proposed by Vaswani et al. [2] and first used in NLP (Natural Language Processing). Due to its unique and superiority of parallel computing, it is gradually used in computer vision. Swin Transformer [19] builds a hierarchical Transformer by introducing a hierarchical construction. Based on Swin Transformer, Xia et al. [20] use the strategy of flow field migration to focus more on relevant areas for key and value, so as to obtain more context information. Although attention mechanisms are now well equipped to deal with associations between different features, it is still a question how to combine their advantages to obtain features with stronger representational ability.

Transformer in visual tracking

In recent years, Transformer-based tracking algorithms are proposed and applied in vision fields. Existing trackers based on Transformer use encoder or decoder structure to incorporate or enhance features extracted by CNN. Wang et al. [3] first apply Transformer in visual tracking and propose a remarkable tracking method TrDiMP. TrDiMP uses the Transformer encoder and Transformer decoder structures to build the relationship across successive frames, which explores the rich context information across them. The tracking algorithm [21] uses a full convolutional network to predict response maps of the upper left and lower right corner, and obtains an optimal bounding box for each frame. It does not use any pre-defined anchors for bounding box regressions.

Lin et al. [22] propose an attention-based tracking method SwinTrack. It uses Transformer for feature extraction and feature fusion. Zhao et al. [23] use multi-head self-attention and multi-head cross-attention to adequately explore the global rich context information instead of using the cross-correlation operation. Inspired by Transformer, Chen et al. [24] propose a novel attention-based feature fusion network. It directly extracts search region features without using any relevant operations. Mayer et al. [25] propose a tracking structure based on Transformer model. It captures global relationships with less inductive bias, and enables it to learn stronger target model predictions. In these Transformer-based trackers, the structured spatial information is not fully exploited.

In this work, a Siamese network architecture is designed. The difference is that the Encoder–Decoder structure is used instead of the cross-correlation layer. Two different efficient attention mechanisms are introduced to the feature fusion network, which can more accurately focus on the region of interest.

Overall architecture

In this section, a novel Transformer-based feature fusion network with dual attentions is designed in a Siamese tracking framework, as shown in Fig. 1. After extracting the deep features from template images and search region images, they are fed into Transformer Encoder and Decoder, respectively. In Transformer Encoder, the multi-head box attention effectively extracts structured spatial information and learns robust feature representations. The attention weights are computed in box attention module. The encoded template features are inputed into Transformer Decoder. In Transformer Decoder, the multi-head instance attention can refine the reference windows of object proposals from Encoder. The rich global context information is fully explored. Based on Transformer structure, the long-range feature dependencies are adequately learnt.

Based the designed feature fusion network, a Multi-scale Dual-attention Transformer Tracker is proposed. Next, the multi-head box attention is analyzed in Transformer Encoder, the multi-head instance attention in Transformer Decoder, and the proposed tracking framework.

Box attention in transformer encoder

In this section, first, the computation of multi-head self-attention is briefly reviewed. Then, the multi-head box attention is introduced, which can focus more on the region of interest in the feature map. Some object proposals with high confidence scores are obtained by geometrically transforming the reference windows on the input feature map. By introducing box attention to Encoder, the proposed tracker MDTT can perform well to appearance variations, such as occlusion, out-of-view, fast motion, and scale variation.

Multi-head self-attention was first proposed in [2]. Self-attention is computed using the scaled dot product in attention map as

$$\mathrm{SA}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V, \tag{1}$$

where the inputs of attention function are Q, K, and V. They are obtained by the linear transformation of query, key, and value. d_k is the dimension of key. The multi-head self-attention (MHSA) of n attention heads is calculated as follows:

$$\mathrm{MHSA}\,(Q_i, K_i, V_i) = \mathrm{Concat}\,(h_1, \ldots, h_l)\,W^{O}, \tag{2}$$

$$h_i = \mathrm{SA}\left(QW_i^{Q}, KW_i^{K}, VW_i^{V}\right), \tag{3}$$

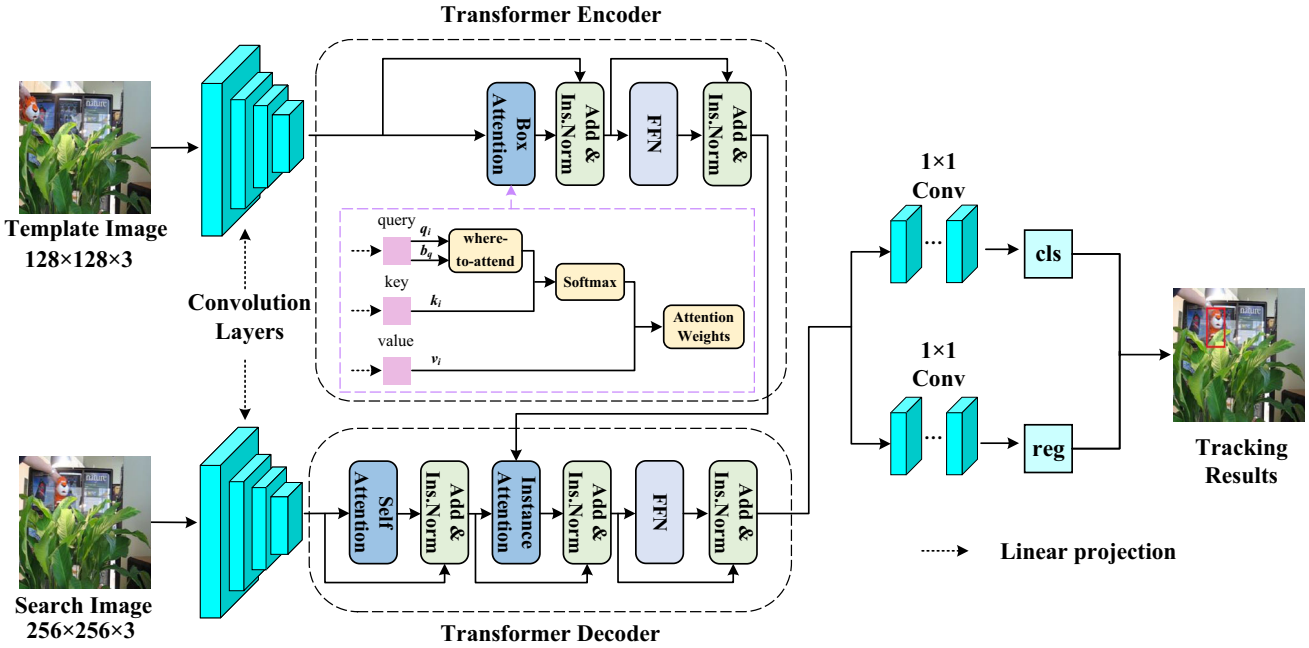


Fig. 1 An overview of the proposed architecture. Given the target template image and the search image in subsequent frames, multi-scale feature maps are extracted from the backbone network. Convolution layers share common network weight. Then, multi-scale feature maps are fed to the Encoder–Decoder structure. Unlike TrDiMP, the box attention and instance attention are added to Encoder and Decoder, respectively. The optimized model can focus on the necessary region and pay attention to the appearance changes of object at any time, so it achieves robust visual tracking

where W^O is a learnable projection matrix, $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_q}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$, and h_l is the number of attention heads.

As shown in Fig. 1, similar to the calculation of MHSA, when calculating the attention of the i^{th} head, given the bounding boxes $b_i \in \mathbb{R}^d$ in the i^{th} head, a $m \times m$ grid feature map $v_i \in \mathbb{R}^{m \times m \times d_h}$ centered on b_i is extracted by the bilinear interpolation. After that, the attention on the extracted grid feature map is computed.

Here, an important module named Where-to-Attend is used after generating the $m \times m$ grid feature map v_i. The module is an important part of box attention, which can transform v_i into an attended region through a geometric transformation. Therefore, the region of attention can adapt to the appearance changes of target. Finally, the attention weights are generated by computing matrix multiplication between query q and key v. Using bilinear interpolation to extract grid features can effectively reduce quantization errors in bounding box regression. This operation is actually identical to RoIAlign [26], which also extracts a finite number of bounding box proposals within regions of interest. This method can capture more accurate target information and obtain more accurate pixel-level information.

After that, the attention weights will be calculated by softmax function to get $QK_i^\top$. Finally, the final box attention $h_i \in \mathbb{R}^{d_h}$ is obtained by calculating the weighted average of linear transformation matrix V_i of $QK_i^\top$ and $m \times m$ grid feature map v_i

$$h_i = \text{BoxAttention}(Q, K_i, V_i) = \sum_{m \times m} \text{softmax}\left(QK_i^\top\right) * V_i. \tag{4}$$

For calculating the attention weight, the attention should be focused around the center of target, and the critical Where-to-Attend module is used. The role of Where-to-Attend module is to make the box attention focusing more on the necessary regions and predicting the bounding boxes more accurately. And the module can transform reference window of query vector q into a more accurate region through geometric transformations. It can predict bounding box proposals in grid feature map using structured spatial information.

As in Fig. 2, $b_q = [x, y, w, h]$ are used to denote the reference windows of q, where x and y denote the central position coordinates of the reference window, and w and h denote the width and height of the reference window, respectively.

Here, translation function $\mathcal{F}_t$ is used to convert the reference window b_q. $\mathcal{F}_t$ takes query q and b_q as its inputs and adjusts the center of the reference window. The output of $\mathcal{F}_t$ is calculated as follows:

$$\mathcal{F}_t\left(b_q, q\right) = [x + \Delta x, y + \Delta y, w, h], \tag{5}$$

where Δx and Δy are offsets relating to the central position of reference window b_q. In addition, we resize the reference

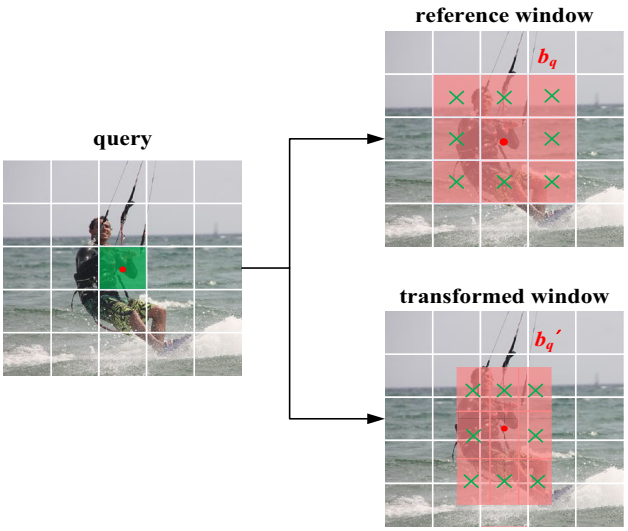


Fig. 2 Where-to-attend modules. It allows box attention to spotlight on the dynamic region of target and effectively use limited attention calculations

window b_q by another translation function $\mathcal{F}_s$. $\mathcal{F}_s$ has the same input as $\mathcal{F}_t$, and its output is computed as follows:

$$\mathcal{F}_s\left(b_q, q\right) = [x, y, w + \Delta w, h + \Delta h], \tag{6}$$

where Δw and Δh are offsets of the size of the reference window b_q. The offset parameters Δx, Δy, Δw, and Δh are implemented by a linear projection of query q as follows:

$$\begin{aligned}
\Delta x &= \left(q W_x^{\top} + b_x\right) * \frac{w}{\tau}, \\
\Delta y &= \left(q W_y^{\top} + b_y\right) * \frac{h}{\tau}, \\
\Delta w &= \max\left(q W_w^{\top} + b_w, 0\right) * \frac{w}{\tau}, \\
\Delta h &= \max\left(q W_h^{\top} + b_h, 0\right) * \frac{h}{\tau},
\end{aligned} \tag{7}$$

where $W^{\top}$ is the weight of the linear projection. τ is the temperature hyperparameter and set to 2. b_x, b_y, b_w, and b_h are bias vectors. The reference window is resized by multiplication, which preserves the scale invariance. Finally, the translation result of reference window b_q is computed with $\mathcal{F}_t$ and $\mathcal{F}_s$ as follows:

$$b_q' = \mathcal{F}\left(\mathcal{F}_t, \mathcal{F}_s\right) = [x + \Delta x, y + \Delta y, w + \Delta w, h + \Delta h]. \tag{8}$$

Then, the box attention calculation of a single head is completed. Furthermore, the box attention calculation is easily extended to multi-head box attention with multiple heads. Given multiple attention heads, the boxes of interest $b_i \in \mathbb{R}^d$ in the query $q \in \mathbb{R}^d$ are expanded to a set of boxes $\left\{b_i^1, \ldots, b_i^t\right\}$. Next, a grid of feature $v_i \in \mathbb{R}^{(t\times m\times m)\times d_h}$ from each box is sampled and the multi-head box attention is computed.

Instance attention in transformer decoder

The purpose of using box attention in Encoder is to generate high-quality object proposals. Similarly, the instance attention is used in Decoder to generate accurate bounding boxes. Different from the box attention, in the i^{th} attention head, instance attention takes the grid features of object proposals in Encoder as input, and then generates two outputs, h_i and h_i^{mask}. Here, only $h_i \in \mathbb{R}^d$ is used for classification to distinguish foreground from surrounding background.

Similarly, the instance attention is extended to multi-head instance attention calculation with multiple heads. First, $v_i \in \mathbb{R}^{(t\times m\times m)\times d_h}$ is obtained by the same way in the box attention. Before creating h_i, the softmax function is used to normalize $t \times m \times m$ attention scores and then applied to v_i.

Tracking with box attention and instance attention

As shown in Fig. 1, a Transformer-based feature fusion network is designed with both box attention and instance attention. To enable the model to make full use of sequential information across successive frames, the positional encoding is added at the bottom of Encoder and Decoder as follows:

$$\begin{aligned}
\mathrm{PE}_{\mathrm{pos,2i}} &= \sin\left(\frac{\mathrm{pos}}{10000^{2\mathrm{i}/\mathrm{d}_{\mathrm{model}}}}\right) \\
\mathrm{PE}_{\mathrm{pos,2i+1}} &= \cos\left(\frac{\mathrm{pos}}{10000^{2\mathrm{i}/\mathrm{d}_{\mathrm{model}}}}\right).
\end{aligned} \tag{9}$$

where d_{model} is set to 256, i is the dimension, and pos is the location information.

The Encoder encodes feature maps $\left\{x^j\right\}_{j=1}^{t-1}$ $(t = 4)$ extracted from the backbone network and obtains the multi-scale contextual representations $\left\{e^j\right\}_{j=1}^{t}$. Here, the ResNet50 [27] is used as the feature extraction network. In the Transformer structure, each of the Encoder layers includes the box attention, and each feed-forward layer is followed by a normalized layer with a residual structure. Encoder takes the target template features as its input and outputs multiple object proposals with high confidence scores. Experiments show that the Encoder with box attention makes the proposed tracker MDTT more effective in dealing with some tracking challenges, such as occlusion, scale variation, and fast motion.

The Decoder predicts bounding boxes and distinguishes the foreground from the background. In decoder layer, the instance attention is used instead of the cross-attention. The object proposals in Encoder were put as the input of Decoder, which will be detailed to the object proposals so as to get a

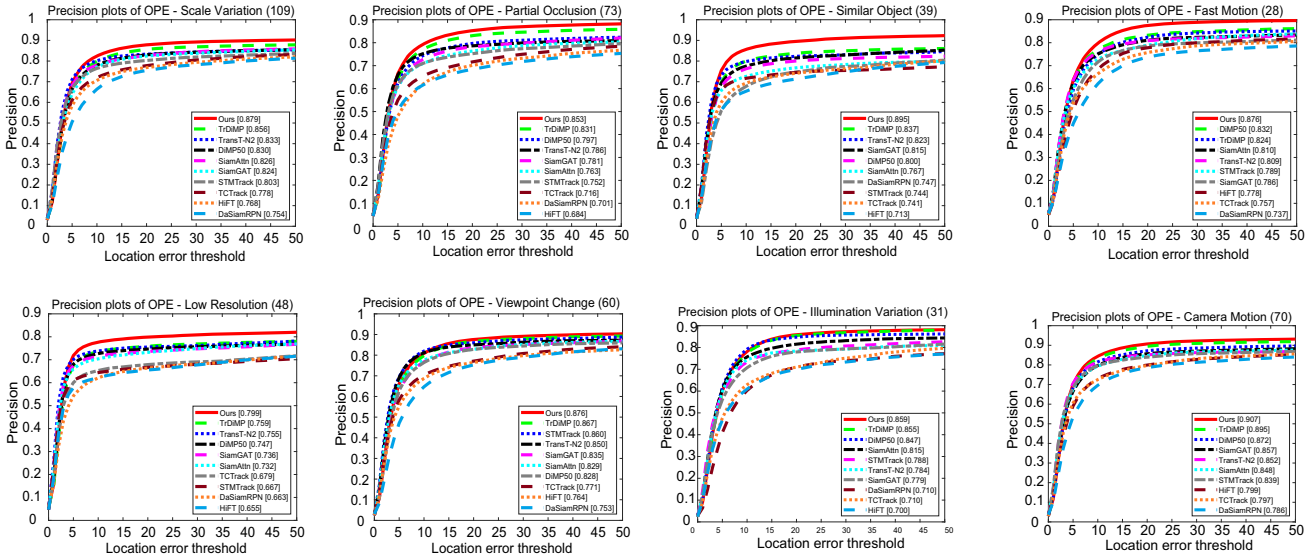


Fig. 3 Precision plots on UAV123 for eight challenging aspects: scale variation, partial occlusion, similar object, fast motion, low resolution, viewpoint change, illumination variation, and camera motion. The proposed MDTT performs well on all these aspects, especially on similar objects, low resolution, and fast motion

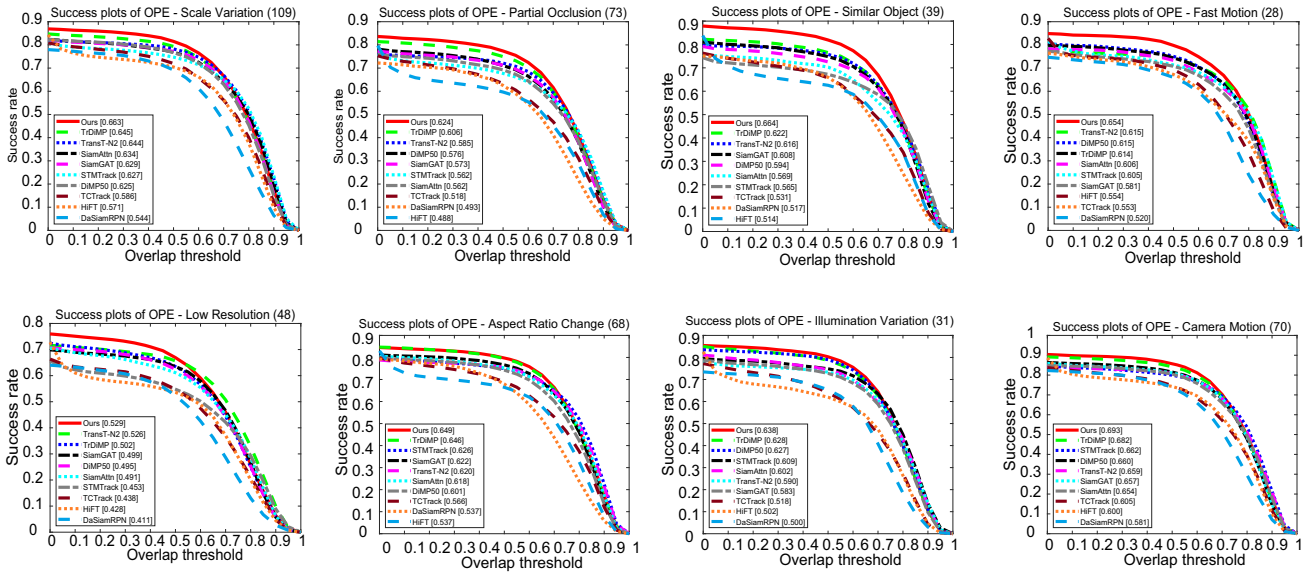


Fig. 4 Area Under the Curve (AUC) plots on UAV123 for eight challenging aspects: scale variation, partial occlusion, similar object, fast motion, low resolution, aspect ratio change, illumination variation, and camera motion. The proposed MDTT performs well on all these aspects, especially on similar object and fast motion

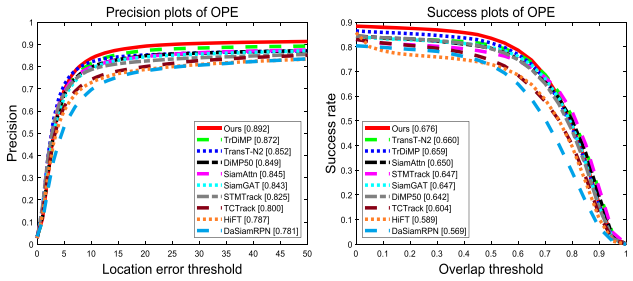
more precise proposal. Since the Decoder use the encoder features with highest classification scores as the input features, this will provide more effective context information to Decoder. This is crucial for the tracking process, since there is a lot of context information across the successive frames.

Experiments

In this section, first, the implement details are given. Then, the proposed tracker MDTT is compared with many recent state-of-the-art trackers on six tracking benchmarks. Finally, the ablation study is conducted and the effects of the key components of the feature fusion network are analyzed.

Implementation details

The proposed method is implemented in Python 3.7 and PyTorch 1.4.0. Four datasets, including COCO [28], GOT-10k [6], LaSOT [8], and TrackingNet [9], are used to train



(a) Precision Plots (b) Success Plots

Fig. 5 Precision and success plots on UAV123 benchmark

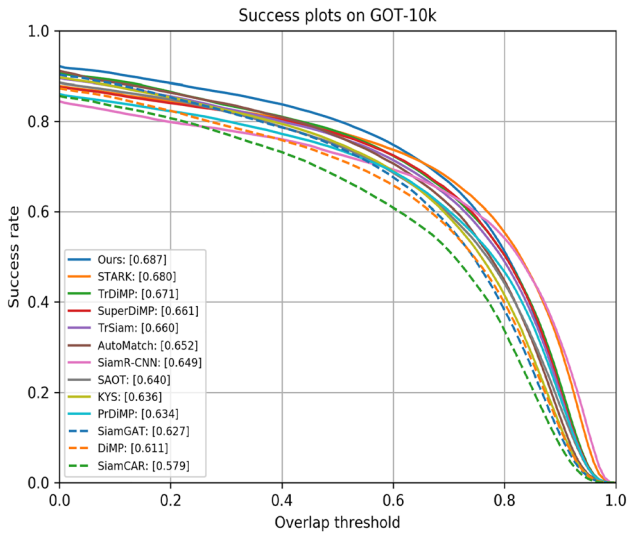


Fig. 6 Comparison with state-of-the-art trackers on GOT-10k in term of success rates

Table 1 Comparison results of the competing trackers on GOT-10k in terms of average overlap (AO) and success rate (SR). The best two results are highlighted in bold and italic, respectively

Tracker	Year	AO (%)	$SR_{0.50}$ (%)	$SR_{0.75}$ (%)
Ours		**68.7**	**80.2**	60.0
STARK [21]	2021	*68.0*	77.7	**62.3**
UTT [39]	2022	67.2	76.3	60.5
TrDiMP [3]	2021	67.1	77.7	58.3
TransT-N2 [24]	2021	67.1	76.8	*60.9*
TREG [40]	2021	66.8	*77.8*	57.2
SBT [41]	2022	66.8	77.3	58.7
SuperDiMP [42]	2019	66.1	77.2	59.2
TrSiam [3]	2021	66.0	76.6	57.1
AutoMatch [43]	2021	65.2	76.6	54.3
SiamR-CNN [44]	2020	64.9	72.8	59.7
SiamPW-RBO [37]	2022	64.4	76.7	50.9
STMTrack [34]	2021	64.2	73.7	57.5
SAOT [45]	2021	64.0	74.7	53.0
KYS [46]	2020	63.6	75.1	51.5
FCOT [47]	2020	63.4	76.6	52.1
PrDiMP [48]	2020	63.4	73.8	54.3
SiamGAT [35]	2021	62.7	74.3	48.8
SiamLA [49]	2022	61.9	72.4	51.0
OCEAN [50]	2020	61.6	72.1	47.3
DiMP [30]	2019	61.1	71.7	49.2
D3S [51]	2020	59.7	67.6	46.2
SiamCAR [52]	2020	57.9	67.7	43.7
ATOM [53]	2019	55.6	63.4	40.2

TrDiMP as the baseline on one Nvidia 2060 GPU. The model is trained for 50 epochs with 3571 iterations per epoch and 14 image pairs per batch. The Resnet-50 [27] is used as the feature extraction backbone network. The size of template images are 128 × 128 pixels and search images are set to 256 × 256 pixels. The proposed tracking method is optimized by ADAM [29] and the initial learning rate is set to 0.01. During the process of tracking, the proposed tracker MDTT runs about 20 FPS on one GPU.

Then, the settings of some parameters used by the tracking model are given. In Encoder, the size of grid feature maps v_i extracted from b_i is set to 2×2 ($m = 2$). The number of attention heads l is set to 8. $d_{feed-forward}$ is set to 1024. The number of encoder layers and decoder layers is 6 ($s = 6$).

State-of-the-art comparison

The proposed MDTT is compared with recent state-of-the-art trackers on six challenging benchmarks, including UAV123 [4], GOT-10k [6], LaSOT [8], VOT2018 [5], TrackingNet [9], and NfS [7].

UAV123 [4]: UAV123 is a challenging dataset captured from low-altitude UAVs. It contains 123 video sequences with many challenging factors, such as fast motion, small object, similar object, motion blur, scale variation, and so on. The tracked targets in UAV123 have low resolutions. In spite of this, the proposed method performs well in dealing with various challenges, as shown in Figs. 3 and 4. The evaluation metrics for UAV123 include precision (P) and area under the curve (AUC). Figure 5 shows the tracking results against state-of-the-art trackers on UAV123 dataset. Compared to the recent trackers TrDiMP [3], TransT [24], DiMP [30], DaSiamRPN [31], HiFT [32], TCTrack [33], STMTrack [34], SiamGAT [35], and SiamAttn [36], the proposed method obtains the highest success score of 67.6% and precision score of 89.2%, and outperforms the baseline by 1.7% on success and 2.0% on precision. The proposed method also performs well in comparison with some recent trackers ToMP [25], SiamBAN-RBO [37], and CNNInMo [38], as shown in Table 5.

VOT2018 [5]: VOT2018 dataset consists of 60 video sequences and the ground truth in VOT is a bounding box

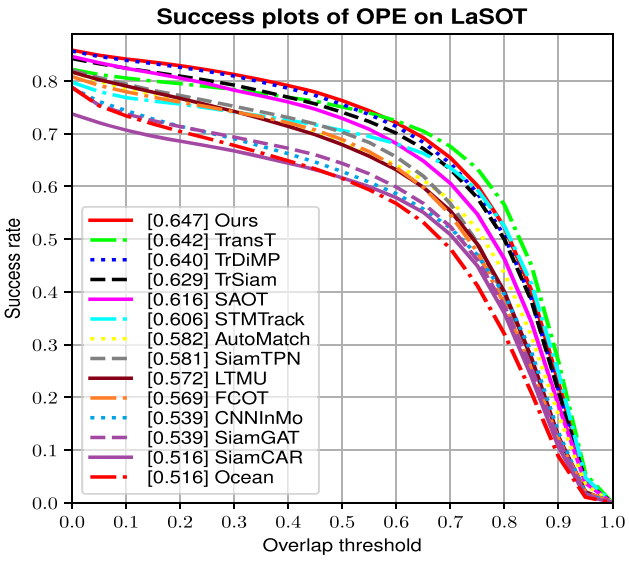
Table 2 Results on LaSOT. Trackers are evaluated by the area under the curve (AUC), precision (P), and normalized precision (P_{Norm}). The best two results are highlighted in bold and italic, respectively

Tracker	Year	AUC (%)	P (%)	P_{Norm} (%)
Ours		**64.7**	*67.5*	**73.9**
UTT [39]	2022	*64.6*	67.2	–
TransT-N2 [24]	2021	64.2	**68.2**	*73.5*
TrDiMP [3]	2021	64.0	66.6	73.2
DualTFR [54]	2021	63.5	66.5	72.0
SuperDiMP [42]	2019	63.1	65.3	72.2
TrSiam [3]	2021	62.9	65.0	71.8
SAOT [45]	2021	61.6	62.9	70.8
SBT [41]	2022	61.1	63.8	–
STMTrack [34]	2021	60.6	63.3	69.3
PrDiMP [48]	2020	59.8	60.8	68.8
AutoMatch [43]	2021	58.2	59.9	67.4
SiamTPN [55]	2022	58.1	57.8	68.3
CAJMU [56]	2022	57.3	57.2	66.3
LTMU [57]	2020	57.2	57.8	66.5
FCOT [47]	2020	56.9	58.9	67.8
SRRTransT [58]	2022	56.9	57.1	64.0
SiamLA [49]	2022	56.1	56.0	65.2
CNNInMo [38]	2022	53.9	53.9	61.6
SiamGAT [35]	2021	53.9	53.0	63.3
CGACD [59]	2020	51.8	62.6	–
OCEAN [50]	2020	51.6	52.6	60.7
SiamCAR [52]	2020	51.6	52.4	61.0
ULAST [60]	2022	47.1	45.1	–

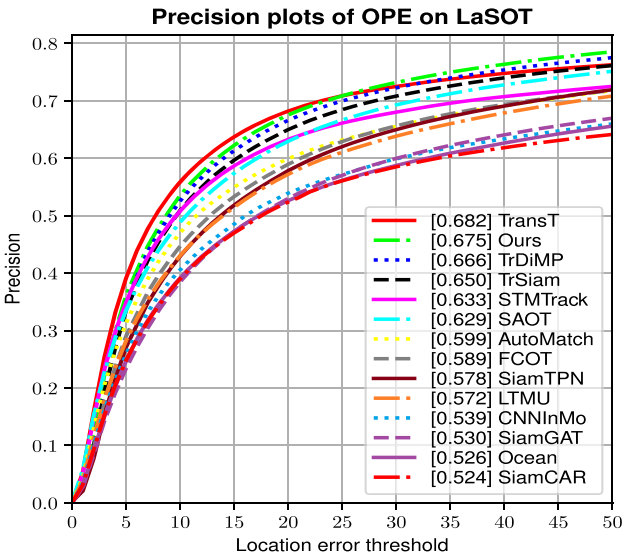
with rotation and scale transformation. Robustness refers to the percentage of subsequence frames that are successfully tracked. EAO is the value that combines accuracy and robustness for comprehensive evaluation. Figure 8 shows the EAO scores for the proposed MDTT and nine recent trackers, including CGACD [59], STMTrack [34], PrDiMP [48], TrDiMP [3], DCFST [66], SiamR-CNN [44], LADCF [64], MFT [5], and SiamRPN [12].

GOT-10k [6]: Got-10k is a large-scale tracking benchmark. In particular, the training set and test set in GOT-10k have different tracking targets, respectively. It contains 560 classes of outdoor moving objects in real scenes, and the test set contains 180 video sequences. As well as the TrackingNet benchmark, the ground truth of GOT-10k is not publicly available, so we evaluate the proposed tracker by an online evaluation. The average overlap (AO) and success rate (SR) are used to compare the performance of trackers.

As shown in Fig. 6, MDTT is compared against several recent popular trackers, including STARK [21], TrDiMP [3], SuperDiMP [42], SiamLA [49], AutoMatch [43], and TREG [40]. MDTT achieves the best tracking performance on suc-



(a) Success Plots



(b) Precision Plots

Fig. 7 Success and precision plots against competitive trackers on LaSOT dataset

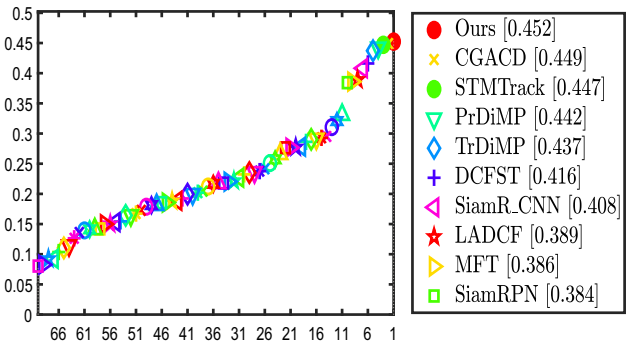


Fig. 8 Expected average overlap (EAO) graph with trackers ranked. Our tracker outperforms all the other trackers on VOT2018

Table 3 Comparison on VOT2018 in terms of accuracy (A), robustness (R), and expected average overlap (EAO). The best two results are highlighted in bold and italic, respectively

Tracker	Year	A(↑)	R(↓)	EAO(↑)
Ours		**61.9**	16.0	**45.2**
Retina-MAML [61]	2020	60.4	15.9	**45.2**
CGACD [59]	2020	61.5	17.2	*44.9*
PGNet [62]	2020	*61.8*	19.2	44.7
STMTrack [34]	2021	59.0	15.9	44.7
PrDiMP [48]	2020	*61.8*	16.5	44.2
DiMP [30]	2019	59.7	15.3	44.0
TrDiMP [3]	2021	60.0	16.2	43.7
SiamFC++ [63]	2020	58.7	18.3	42.6
SiamCAR [52]	2020	57.8	19.7	42.3
SiamRPN++ [13]	2019	60.0	23.4	41.4
SiamR-CNN [44]	2020	60.9	22.0	40.8
ATOM [53]	2019	59.0	20.4	40.1
LADCF [64]	2019	50.3	15.9	38.9
MFT [5]	2018	50.5	14.0	38.5
SiamRPN [12]	2018	58.6	27.6	38.3
UPDT [65]	2018	53.6	18.4	37.9

Table 4 Comparison on the TrackingNet in terms of the area under the curve (AUC), precision (P), and normalized precision (P_{Norm}). The best two results are highlighted in bold and italic, respectively

Tracker	Year	AUC (%)	P (%)	P_{Norm} (%)
Ours		**78.1**	**73.4**	**83.3**
SiamLA [49]	2022	*76.7*	71.8	82.1
AutoMatch [43]	2021	76.0	*72.6*	–
SRRTransT [58]	2022	76.0	71.9	81.3
PrDiMP [48]	2020	75.8	70.4	81.6
FCOS-MAML [61]	2020	75.7	72.5	*82.2*
SiamFC++ [63]	2020	75.4	70.5	80.0
SiamGAT [35]	2021	75.3	69.8	80.7
DCFST [66]	2020	75.2	70.0	80.9
CAJMU [56]	2022	74.2	68.9	80.1
KYS [46]	2020	74.0	68.8	80.0
DiMP [30]	2019	74.0	68.7	80.1
SiamCAR [52]	2020	74.0	68.4	80.4
SiamLTR [67]	2021	73.6	69.1	80.2
SiamRPN++ [13]	2019	73.3	69.4	80.0
D3S [51]	2020	72.8	66.4	76.8
OCEAN [50]	2020	70.3	68.8	–
ATOM [53]	2019	70.3	64.8	77.1
CRPN [68]	2019	66.9	61.9	74.6
ULAST [60]	2022	65.4	59.2	73.2
DaSiamRPN [31]	2018	63.8	59.1	73.3
UPDT [65]	2018	61.1	55.7	70.2

cess. Also, Table 1 presents the tracking performance of the proposed method and recent state-of-the-art trackers on GOT-10k, such as UTT [39], SBT [41], SiamPW-RBO [37], and SiamLA [49]. As shown in Table 1, the proposed tracker outperforms most of them. Compared with the popular tracker TrDiMP, MDTT is higher on AO, $SR_{0.5}$, and $SR_{0.75}$ by 1.6%, 2.5%, and 1.7%, respectively. The above results demonstrate that the proposed method is adapt to a large number of different scenarios and challenges.

LaSOT [8]: LaSOT is a large-scale and complex single object dataset. It contains 280 sequences with an average 2448 frames per sequence in the test set. We evaluate the proposed tracker on LaSOT dataset to validate its long-term capability. The proposed tracker is compared with some recent trackers, including UTT [39], TransT [24], TrDiMP [3], DualTFR [54], SAOT [45], SBT [41], SiamTPN [55], STMTrack [34], PrDiMP [48], AutoMatch [43], CAJMU [56], SRRTransT [58], SiamLA [49], CNNInMo [38], SiamGAT [35], and ULAST [60].

Figure 7 shows the success and precision plots of MDTT tracker and 13 state-of-the-art trackers. These trackers are ranked according to the AUC and precision scores. From Fig. 7, it can be seen that MDTT achieves the top-rank AUC score of 64.7% and achieves the performance on precision of 67.5%. Table 2 shows the tracking performance on success, precision and normalized precision metrics. Compared with UTT, TrDiMP, and DualTFR, the proposed method improves the AUC score by 0.1%, 0.7%, and 1.2%, respectively. The above results demonstrate that MDTT adapts to long-term tracking and performs well in terms of success, precision, and normalized precision.

Table 3 shows more details compared with these trackers. As shown in Table 3, although CGACD achieves an EAO score of 44.9%, MDTT achieves the better performance on EAO of 45.2% and accuracy rate of 61.9%, which outperforms all other SOTA trackers. In addition, MDTT outperforms the baseline by 1.5% on EAO and 1.9% on accuracy, respectively.

TrackingNet [9]: TrackingNet dataset includes 30k video sequences, and the test set consists of 511 video sequences with various object classes in real scenes. The evaluation process is performed on the online evaluation server. The proposed tracker achieves significant results on success, precision, and normalized precision. In this benchmark, the proposed MDTT is compared with SOTA trackers, such as SiamLA [49], AutoMatch [43], SRRTransT [58], CAJMU [56], SiamLTR [67], PrDiMP [48], and ULAST [60]. As shown in Table 4, the proposed tracker obtains the best performance with 78.1%, 73.4%, and 83.3% in terms of AUC, P, and P_{Norm}, respectively. The proposed tracker performs bet-

Table 5 Comparison with SOTA trackers on NfS and UAV123 datasets in terms of AUC. Bold and Italic fonts indicate the top-2 trackers

Tracker	Ours	Tr DiMP [3]	ToMP 101 [25]	Trans T [24]	Auto Match [43]	CAJ MU [56]	SiamBAN RBO [37]	STM Track [34]	CNN InMo [38]	DCF ST [66]	CRACT [69]	OCEAN [50]	SiamDW [70]	KYS [46]	Siam R-CNN [44]
Year		2021	2022	2021	2021	2022	2022	2021	2022	2020	2020	2020	2019	2020	2020
NfS	*66.2*	64.8	**66.7**	65.7	60.6	62.7	61.3	–	56.0	64.1	62.5	55.3	52.1	63.5	63.9
UAV123	**67.6**	65.9	*66.9*	66.0	64.4	–	64.1	64.7	62.9	–	66.4	62.1	53.6	–	64.9

Table 6 The ablation study on UAV123 in terms of Precision (P) and Area Under the Curve (AUC). Box-Att denotes box attention. Ins-Att denotes instance attention

Method	Training sets	Devices	Different variations		UAV123	
			Box-Att	Ins-Att	AUC (%)	P (%)
TrDiMP	aSOT GOT-10k TrackingNet COCO	4*GTX1080Ti	×	×	67.0	87.6
Baseline		1*RTX2060	×	×	65.9	87.2
Ours			√	×	66.7	87.9
			×	√	66.5	87.7
			√	√	67.6	89.2

Table 7 The ablation study on GOT-10k in terms of AO, $SR_{0.5}$, and $SR_{0.75}$, respectively. Box-Att denotes box attention. Ins-Att denotes instance attention

Method	Training sets	Devices	Different variations		GOT-10k		
			Box-Att	Ins-Att	AO (%)	$SR_{0.5}$ (%)	$SR_{0.75}$ (%)
TrDiMP	LaSOT GOT-10k TrackingNet COCO	4*GTX1080Ti	×	×	67.1	77.7	58.3
Baseline		1*RTX2060	×	×	66.0	76.6	57.1
Ours			√	×	68.0	79.1	59.6
			×	√	67.6	78.8	58.5
			√	√	68.7	80.2	60.0

ter than SiamLA with 76.7% on SUC score and SRRTransT with 76.0% on SUC score.

NfS [7]: NfS is a high frame rate dataset with a total of 380K frames. It contains 100 challenging videos, and includes 30 FPS version and 240 FPS version. Here, NfS of the 30 FPS version is used to evaluate MDTT. Table 5 shows the AUC scores against recent trackers. MDTT has significant advantages over many previous trackers, such as TransT [24], CAJMU [56], CNNInMo [38], SiamBAN-RBO [37], and STMTrack [34]. The proposed tracker is on par with the latest tracker ToMP, and outperforms the baseline by 1.4% on success.

Ablation study and analysis

The ablation study is performed on UAV123 and GOT-10k to verify the effectiveness of box attention and instance attention in the designed feature fusion network in MDTT. Here, also four datasets, including COCO [28], GOT-10k [6], LaSOT [8], and TrackingNet [9], are used to train TrDiMP as the baseline on one Nvidia 2060 GPU.

Only using box attention. Here, the box attention is used in the Encoder structure. The Decoder structure still uses cross-attention. MDTT is evaluated in UAV123 and GOT-10k to verify the effect of the multi-head box attention. As shown in Tables 6 and 7, MDTT improves the success rate by 0.8% and precision rate by 0.7% on UAV123 compared to the baseline when only using box attention. On GOT-10k, the AO rate increased by 1.9% compared with the baseline. Experiment results show that box attention plays a key role in the designed structure.

Only using instance attention. The original self-attention is used in Encoder and the instance attention is used in Decoder. The proposed tracker is evaluated in UAV123 and GOT-10k to verify the influence of instance attention. As shown in Tables 6 and 7, compared to the baseline when only using the instance attention, MDTT improves the success rate by 0.6% and precision rate by 0.5% on UAV123. On GOT-10k,

Table 8 Comparison about the speed, FLOPs, and Params

Tracker	Year	Backbone	Image size		Speed (fps)	FLOPs (G)	Params (M)
			Template	Search region			
Ours		ResNet-50	128×128	256×256	20	19.3	23.5
SiamRPN++ [13]	2019	ResNet-50	127×127	255×255	35.0	48.9	54.0
STARK-S50 [21]	2021	ResNet-50	128×128	320×320	42.2	10.5	23.3
Mixformer [72]	2022	MAM	128×128	320×320	25	23.04	–
TransT [24]	2021	ResNet-50	128×128	256×256	50	19.1	23
DualTFR [54]	2021	LAB	112×112	224×224	40	18.9	44.1

the AO rate increases by 1.5% compared with the baseline. Compared with the method only with box attention, the performance of instance attention is almost the same as using the box attention. Nevertheless, it also shows the effectiveness of instance attention.

Using both. Finally, both of box attention and instance attention are introduced to the baseline and the experiments are conducted on UAV123 and GOT-10k. As shown in Tables 6 and 7, the proposed method improves the success rate by 1.7% and precision rate by 2.0% on UAV123 compared with the baseline. On GOT-10k, the AO rate increases by 2.6% compared with the baseline. The experimental results show that using both of box attention and instance attention greatly improved the performance of our tracker by 2.0%. In addition, the proposed method also shows better performance compared to TrDiMP. It reaches 1.6% improvement of AO in GOT-10k and 1.6% improvement of precision in UAV123 than TrDiMP.

Speed, FLOPs, and params

We use ResNet-50 as the backbone network of the proposed tracker MDTT. The results in Table 8 refer to the official website and the author's personal homepage. As shown in Table 8, MDTT can run in real-time tracking speed. And the number of parameters does not increase significantly. In addition, the FLOPs and Params of MDTT are less than SiamRPN++. Although we use more advanced attention mechanisms in the feature fusion network, the increasing in FLOPs and Params was not significant. This indicates that using of the box attention and instance attention allows the tracker MDTT to explore structured spatial information and global information at a lower cost.

Limitations and future works

Limitations. Although the proposed feature fusion network is effective, we did not optimize the feature extraction network. In Fig. 9, we show two tracking failure cases of the proposed tracker MDTT on Bird1 and Car1 video sequences.

(a) *Bird1* (b) *Car1*

Fig. 9 Two cases of failure

The tracking results of MDTT and the corresponding ground truths are shown in green and red boxes. As shown in Fig. 9, when the proposed tracker deals with challenging, such as out-of-view and motion blur, it fails to track the targets. This illustrates that the designed feature fusion network is not robust in capturing the appearance variations in some scenes. The proposed tracker MDTT is not very robust in dealing with large appearance variations, such as out-of-view or motion blur. Meanwhile, the proposed MDTT lacks a target template updating mechanism.

Future Works. The proposed tracker can capture the structured spatial information and global information well. However, when occurring the appearance variations of the target disappearing or target blurred, the structure information captured by the tracker is not accurate, which leads to tracking drift or failure. In the future, we will further optimize the feature extraction network to improve the feature representation ability. On the other hand, we will design a target template updating mechanism to capture the target appearance variations. In addition, we found that some statistical tests [71] can be used to verify the tracking performance. In future study, we will use some statistical tests for experimental comparisons to evaluate the tracking performance.

Conclusion

A Transformer-based tracking framework with the multi-scale feature fusion network is proposed. The box attention can capture more relevant information and the structured spatial information, and gives more attention to the region of interest in template images. Then, instance attention exploits the temporal information. By integrating the box attention and instance attention in Encoder–Decoder architecture, the feature fusion network not only focuses on the temporal information across successive frames, but also focuses more on the ROI. And the network effectively improves the accuracy of classification and regression. The ablation study on UAV123 and GOT-10k verifies the effectiveness of the multi-scale feature fusion network. Experimental results on six challenging tracking datasets show that MDTT outperforms many recent trackers.

Acknowledgements This work is supported by the National Natural Science Foundation of China (No: 61861032).

Data availability Data will be made available on request.

References

1. Chen K, Guo X, Xu L, Zhou T, Li R (2022) A robust target tracking algorithm based on spatial regularization and adaptive updating model. Complex Intell Syst 2:1–15
2. Vaswani A, Shazeer A, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. Adv Neural Inform Process Syst 30:2
3. Wang N, Zhou W, Wang J, Li H (2021) Transformer meets tracker: Exploiting temporal context for robust visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1571–1580
4. Mueller M, Smith N, Ghanem B (2016) A benchmark and simulator for uav tracking. In: European conference on computer vision, Springer, pp. 445–461
5. Kristan M, Leonardis A, Matas J, Felsberg M, Pflugfelder R, Cehovin Zajc L, Vojir T, Bhat G, Lukezic A, Eldesokey A et al (2018) The sixth visual object tracking vot2018 challenge results. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops
6. Huang L, Zhao X, Huang K (2019) Got-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE Trans Pattern Anal Mach Intell 43(5):1562–1577
7. Kiani Galoogahi H, Fagg A, Huang C, Ramanan D, Lucey S (2007) Need for speed: A benchmark for higher frame rate object tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1125–1134
8. Fan H, Lin L, Yang F, Chu P, Deng G, Yu S, Bai H, Xu Y, Liao C, Ling H (2019) Lasot: A high-quality benchmark for large-scale single object tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5374–5383
9. Muller M, Bibi A, Giancola S, Alsubaihi S, Ghanem B (2018) Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In: Proceedings of the European conference on computer vision (ECCV), pp 300–317
10. Bertinetto L, Valmadre J, Henriques JF, Vedaldi A, Torr PH (2016) Fully-convolutional siamese networks for object tracking. In: European conference on computer vision, Springer, pp 850–865
11. Dong X, Shen J (2018) Triplet loss in siamese network for object tracking. In: Proceedings of the European conference on computer vision (ECCV), pp 459–474
12. Li B, Yan J, Wu W, Zhu Z, Hu X (2018) High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8971–8980
13. Li B, Wu W, Wang Q, Zhang F, Xing J, Yan J (2019) Siamrpn++: Evolution of siamese visual tracking with very deep networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4282–4291
14. Abdelpakey MH, Shehata MS, Mohamed MM (2018) Denssiam: End-to-end densely-siamese network with self-attention model for object tracking. In: International Symposium on Visual Computing, Springer, pp. 463–473
15. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132–7141
16. Woo S, Park J, Lee JY, Kweon IS (2018) Cbam: Convolutional block attention module. In: Proceedings of the European conference on computer vision (ECCV), pp 3–19
17. Xiao Z, Xu X, Xing H, Song F, Wang X, Zhao B (2021) A federated learning system with enhanced feature extraction for human activity recognition. Knowl-Based Syst 229:107338
18. Xing H, Xiao Z, Zhan D, Luo S, Dai P, Li K (2022) Selfmatch: Robust semisupervised time-series classification with self-distillation. Int J Intell Syst 37(11):8583–8610
19. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 10012–10022
20. Xia Z, Pan X, Song S, Li LE, Huang G (2022) Vision transformer with deformable attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4794–4803
21. Yan B, Peng H, Fu J, Wang D, Lu H (2021) Learning spatio-temporal transformer for visual tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 10448–10457
22. Lin L, Fan H, Xu Y, Ling H (2021) Swintrack: A simple and strong baseline for transformer tracking, arXiv preprint arXiv:2112.00995
23. Zhao M, Okada K, Inaba M (2021) Trtr: Visual tracking with transformer, arXiv preprint arXiv:2105.03817
24. Chen X, Yan B, Zhu J, Wang D, Yang X, Lu H (2021) Transformer tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8126–8135
25. Mayer C, Danelljan M, Bhat G, Paul M, Paudel DP, Yu F, Van Gool L (2022) Transforming model prediction for tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8731–8740

26. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp. 2961–2969
27. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
28. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár D, Zitnick CL (2014) Microsoft coco: Common objects in context. In: European conference on computer vision, Springer, pp 740–755
29. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980
30. Bhat G, Danelljan M, Gool LV, Timofte R (2019) Learning discriminative model prediction for tracking. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 6182–6191
31. Zhu Z, Wang Q, Li B, Wu W, Yan J, Hu W (2018) Distractor-aware siamese networks for visual object tracking. In: Proceedings of the European conference on computer vision (ECCV), pp 101–117
32. Cao Z, Fu C, Ye J, Li B, Li Y (2021) Hift: Hierarchical feature transformer for aerial tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15457–15466
33. Cao Z, Huang Z, Pan L, Zhang S, Liu Z, Fu C (2022) Tctrack: Temporal contexts for aerial tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14798–14808
34. Fu Z, Liu Q, Fu Z, Wang Y (2021) Stmtrack: Template-free visual tracking with space-time memory networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 13774–13783
35. Guo D, Shao Y, Cui Y, Wang Z, Zhang L, Shen C (2021) Graph attention tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9543–9552
36. Yu Y, Xiong Y, Huang E, Scott MR (2020) Deformable siamese attention networks for visual object tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6728–6737
37. Tang F, Ling Q (2022) Ranking-based siamese visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8741–8750
38. Guo M, Zhang Z, Fan H, Jing L, Lyu Y, Li B, Hu W (2022) Learning target-aware representation for visual tracking via informative interactions, arXiv preprint arXiv:2201.02526
39. Ma F, Shou MZ, Zhu L, Fan H, Xu Y, Yang Y, Yan Z (2022) Unified transformer tracker for object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8781–8790
40. Cui Y, Jiang C, Wang L, Wu G (2021) Target transformed regression for accurate tracking, arXiv preprint arXiv:2104.00403
41. Xie F, Wang C, Wang G, Cao Y, Yang W, Zeng W (2022) Correlation-aware deep tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8751–8760
42. Danelljan M, Bhat G (2019) Pytracking: Visual tracking library based on pytorch
43. Zhang Z, Liu Y, Wang X, Li B, Hu W (2021) Learn to match: Automatic matching network design for visual tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 13339–13348
44. Voigtlaender P, Luiten J, Torr PH, Leibe B (2020) Siam r-cnn: Visual tracking by re-detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6578–6588
45. Zhou Z, Pei W, Li X, Wang H, Zheng F, He Z (2021) Saliency-associated object tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9866–9875
46. Bhat G, Danelljan M, Gool LV, Timofte R (2020) Know your surroundings: Exploiting scene information for object tracking. In: European Conference on Computer Vision, Springer, pp 205–221
47. Cui Y, Jiang C, Wang L, Wu G (2020) Fully convolutional online tracking, arXiv preprint arXiv:2004.07109
48. Danelljan M, Gool LV, Timofte R (2020) Probabilistic regression for visual tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7183–7192
49. Nie J, Wu H, He Z, Yang Y, Gao M, Dong Z (2022) Learning localization-aware target confidence for siamese visual tracking, arXiv preprint arXiv:2204.14093
50. Zhang Z, Peng H, Fu J, Li B, Hu W (2020) Ocean: Object-aware anchor-free tracking. In: European Conference on Computer Vision, Springer, pp 771–787
51. Lukezic A, Matas J, Kristan M (2020) D3s-a discriminative single shot segmentation tracker. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7133–7142
52. Guo D, Wang J, Cui Y, Wang Z, Chen S (2020) Siamcar: Siamese fully convolutional classification and regression for visual tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6269–6277
53. Danelljan M, Bhat G, Khan FS, Felsberg M (2019) Atom: Accurate tracking by overlap maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4660–4669
54. Xie F, Wang C, Wang G, Yang W, Zeng W (2021) Learning tracking representations via dual-branch fully transformer networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 2688–2697
55. Xing D, Evangeliou N, Tsoukalas A, Tzes A (2022) Siamese transformer pyramid networks for real-time uav tracking. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 2139–2148
56. Shen Q, Li X, Meng F, Liang Y (2022) Context-aware visual tracking with joint meta-updating, arXiv preprint arXiv:2204.01513
57. Dai K, Zhang Y, Wang D, Li J, Lu H, Yang X (2020) High-performance long-term tracking with meta- updater. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6298–6307
58. Zhu J, Chen X, Wang D, Zhao W, Lu H (2022) Srrt: Search region regulation tracking, arXiv preprint arXiv:2207.04438
59. Du F, Liu P, Zhao W, Tang X (2020) Correlation-guided attention for corner detection based visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6836–6845
60. Shen Q, Qiao L, Guo J, Li P, Li X, Li B, Feng W, Gan W, Wu W, Ouyang W (2022) Unsupervised learning of accurate siamese tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8101–8110
61. Wang G, Luo C, Sun X, Xiong Z, Zeng W (2020) Tracking by instance detection: A meta-learning approach. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 6288–6297
62. Liao B, Wang C, Wang Y, Wang Y, Yin J (2020) Pg-net: Pixel to global matching network for visual tracking. In: European Conference on Computer Vision, Springer, pp 429–444
63. Xu Y, Wang Z, Li Z, Yuan Y, Yu G (2020) Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In: Proceedings of the AAAI Conference on Artificial Intelligence 34:12549–12556
64. Xu T, Feng Z-H, Wu X-J, Kittler J (2019) Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. IEEE Trans Image Process 28(11):5596–5609

65. Bhat G, Johnander J, Danelljan M, Khan FS, Felsberg M (2018) Unveiling the power of deep tracking. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 483–498
66. Zheng L, Tang M, Chen Y, Wang J, Lu H (2020) Learning feature embeddings for discriminant model based tracking. In: European Conference on Computer Vision, Springer, pp 759–775
67. Tang F, Ling Q (2021) Learning to rank proposals for siamese visual tracking. IEEE Trans Image Process 30:8785–8796
68. Fan H, Ling H (2019) Siamese cascaded region proposal networks for real-time visual tracking, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 7952–7961
69. Fan H, Ling H (2020) Cract: Cascaded regression-align-classification for robust visual tracking, arXiv preprint arXiv:2011.12483
70. Zhang Z, Peng H (2019) Deeper and wider siamese networks for real-time visual tracking, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 4591–4600
71. Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18
72. Cui Y, Jiang C, Wang L, Wu G (2022) Mixformer: End-to-end tracking with iterative mixed attention, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 13608–13618