2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Masked Mutual Guidance Transformer Tracking

Baojie Fan*, Zhiquan Wang, Jiajun Ai and Caiyu Zhang

***Abstract*— Visual mask learning has received increasing attention in the field of visual object tracking. However, most existing studies merely utilize visual mask learning works as pre-training models without fully exploiting their potential for visual representation. In this paper, we present a novel approach for learning tracking target features, leveraging an encoder-decoder architecture with a masked mutual guidance tracking(MMG). Initially, we perform joint visual feature extraction on both the template and search areas. Subsequently, these features undergo separate self-decoding processes, followed by mutual guidance decoding to reconstruct the original search and template images. This process fosters mutual understanding between the images, facilitating improved learning of object states and shapes across different frames. During the inference phase, we offload the decoder and implement a simple and effective tracker. Experimental results indicate that our proposed method is effective that the mutual guidance strategy can achieve state-of-the-art performance on five tracking datasets.**

I. INTRODUCTION

Visual object tracking poses a significant challenge in the realm of computer vision, relying solely on the initial frame of a video to determine the subsequent state and position of an object. Thus, the efficacy of visual representation stands as a cornerstone for successful trackers. With the advancement of deep learning, the evolution of tracking algorithms has been catalyzed. Initially, trackers based on Siamese networks have demonstrated the ability to discern distinct visual features of an object across frames and align them using similarity calculations. In efforts to overcome the local constraints of CNN in feature alignment, some studies have integrated transformer into the process, harnessing their global features to enhance alignment efficiency. Nevertheless, the visual feature extraction phase remains constrained by the local recognition capabilities of CNN. Consequently, recent research endeavors have embraced a unified approach, seamlessly integrating visual feature extraction and alignment, thereby proposing a novel paradigm for visual object tracking tasks. To enhance the model's visual representation capability, prior studies have employed pre-training on ImageNet [1]–[4] followed by fine-tuning. Experiments have shown that using the pre-trained model has more visual representation performance. However, the ImageNet dataset is designed for classification tasks with categories, and the visual tracking task is generally category-less. Consequently, ImageNet might not be entirely suitable for visual tracking tasks. With the growing acclaim of MAE [5], attention in the visual target tracking domain has shifted towards exploring its potential. Nonetheless, some earlier studies merely incorporated MAE as a pre-trained model for visual feature extraction without delving deeply into its visual representation capabilities. In a bid to further probe the potential of MAE in visual target tracking, recent research, such as MAT, introduces a mask appearance transfer method. This method jointly encodes the visual appearance and search area of the template, followed by separate decoding processes. Its tracking paradigm is shown in Figure 2(a). Similarly, CTTrack adopts an asymmetric encoder-decoder architecture, albeit with a variation. Unlike MAT, CTTrack's approach involves random masking after the encoder output, followed by the fusion of template tokens and search tokens for holistic decoding. However, both MAT [6] and CTTrack [7] fall short of fully leveraging mask representation capabilities in visual target tracking. Its tracking paradigm is shown in Figure 2(b).They focus solely on separate decoding processes and neglect the fusion cognition of different target states across frames.

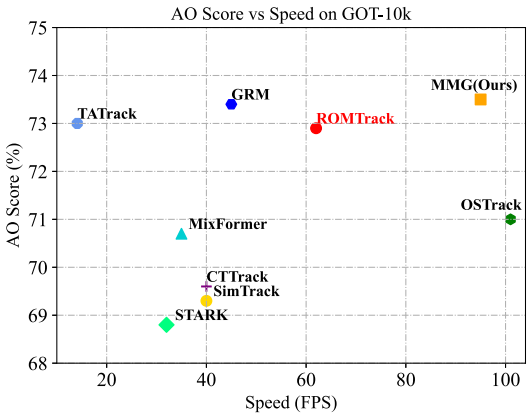

Fig. 1. Comparison with state-of-the-art trackers on GOT-10k. While our MMG outperforms state-of-the-art trackers in AO metrics, it also outperforms them in tracking speed.

In our endeavor to delve deeper into the potential of visual mask learning within the realm of visual target tracking,

*This work is supported by the National Natural Science Foundation of China (No. 62473205, U2013210, 62103388), and the young and middle-aged leading scholar in Qinglan Project by Jiangsu Province

*corresponding authors. Baojie Fan is with the College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications jobfbj@gmail.com

Zhiquan Wang is with College of Electronic and Optical Engineering, College of Flexible Electronics (Future Technology), Nanjing University of Posts and Telecommunications czember@163.com

Jiajun Ai is with College of Automation and College of ArtificialIntelligence, Nanjing University of Posts and Telecommunications nakr000@163.com

Caiyu Zhang is with the College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications

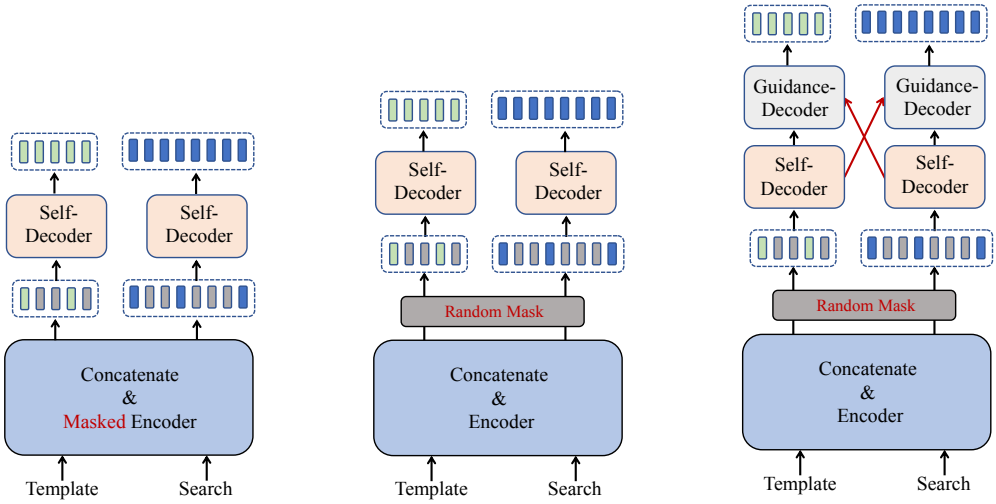

Fig. 2. Tracking pipelines for three different mask learning representations, where red font indicates masking at different locations.

we tailor the characteristics of target tracking to MAE and introduce a convenient and effective method for learning target alignment capabilities. Our approach aims to seamlessly integrate visual representation learning with alignment. Initially, we conduct visual representation learning utilizing a transformer-based autoencoder, followed by the integration of a tracking head for target tracking purposes. Concurrently, random masking is applied to both the search area and the online template area. We consider that the initial template remains unchanged throughout, recognizing its limited utility for cross-frame learning,and abandon masking it. We begin by restoring separate self-decoders for self-cognition, after which the decoded features are intercrossed to guide re-decoding, as shown in Figure 2(c). This iterative process facilitates the recognition of target states across different frames, guiding mutual learning between the search frame and the dynamic template frame to refine decoding features and bolster the model's ability to pinpoint targets. The acquired alignment proficiency is then seamlessly integrated into the autoencoder via network propagation, enhancing the model's discriminative capabilities.

During the inference phase, we discard the decoder and solely rely on the autoencoder along with the tracking head for evaluation. We compare the AO metrics and speeds of representative trackers on GOT-10k, as shown in Figure 1. It can be seen that the proposed MMG both outperform existing trackers on AO and performs at 95fps on an NVIDIA 2080ti GPU. Our model is rigorously benchmarked against other state-of-the-art trackers across renowned datasets such as LaSOT [8], Trankingnet [9], and GOT10K [10], as well as on three challenging datasets including VOT2020 [11], within the field of object tracking. Experiments prove that our proposed tracker enhances the model's representational cognitive capabilities almost without any burden.

To summarize, our main contributions include:

- We propose a learning paradigm for visual object tracking based on an encoder-decoder architecture, masked mutual guidance learning. This method involves conducting joint visual representation learning on templates and search images within the encoder, while simultaneously guiding the specific visual states through the decoder.
- We introduce a featherweight tracker in the evaluation stage. This tracker seamlessly integrates visual representation learning and alignment capabilities without the need for additional hyperparameters and with impressive speed effects.
- Extensive experiments substantiate the effectiveness and generalization ability of the proposed method. Comprehensive comparisons reveal that our tracker can attain state-of-the-art performance by enhancing representation capabilities through mutually guided learning.

II. Related Work

A. Visual Object Tracking

There are two popular methods for aligning template features and search features.

(i) CNN-based feature alignment. SiamFC [12] first proposed using a fully convolutional network to train a visual target tracking network and using CNN to obtain visual features and feature alignment. SiamRPN++ [13] improves cross correlation into depth-wise cross correlation, which can more accurately align template images and search images to return target position information. SiamBAN [1] changes the position regression method to anchor-free type, which can return the target shape more freely. (ii) Transformer-based feature alignment. Early transformer-based feature alignment methods still relied on CNN in the visual feature extraction stage to obtain images. For example, TransT [4] introduces a feature fusion network and utilizes an attention mechanism to blend the features of the template and search region. Building on the concepts from DETR [14], STARK [3] extends these ideas to propose an end-to-end transformer-based tracker. The inner attention module proposed in AiATrack [15] en-

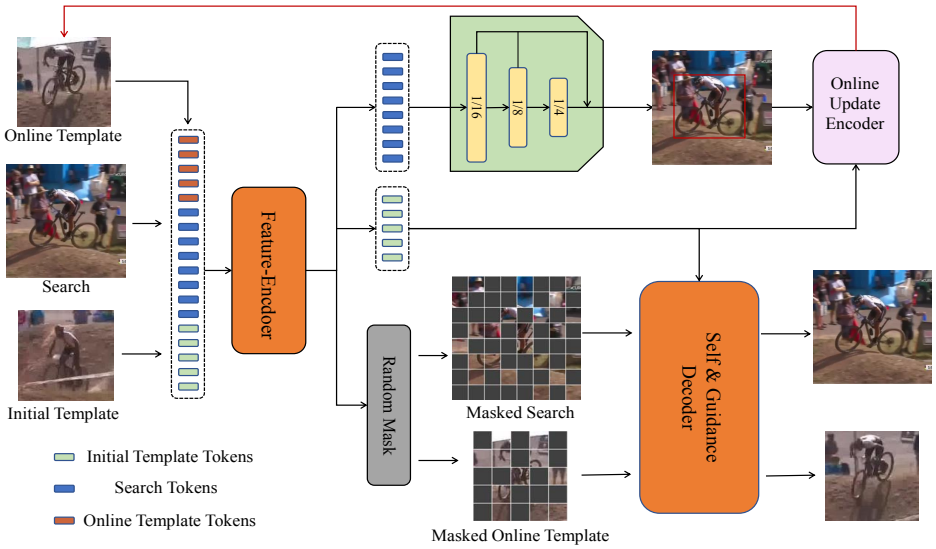

Fig. 3. The proposed overall framework of masked mutual guidance(MMG). The original template, online template and search area are divided into blocks, connected and sent to the visual encoder to extract features, and then decoded and tracked separately to generate target position information.

hances attention by seeking consensus among all correlation vectors. These trackers are efficient but have poor recognition capabilities because they separate feature extraction and interaction, and the template and search area have no previous interaction and cannot adapt to continuous changes in objects. In order to solve the above problems, MixFormer [16] creates a concise tracking framework and introduces a mixed attention module that consolidates the processes of feature extraction and information matching. Similarly, OSTrack [17] concatenates the flattened template and search region and inputs them into self-attention layers for feature extraction. Inspired by the Masked image modeling task, MAT [6] proposes masked appearance transfer, a novel representation learning method for visual object tracking that jointly encodes the template and search region images, and learns tracking-specified representation with a simple encoder-decoder pipeline. Unlike MAT, CTTrack [7] masks after the encoder stage,and then splices template tokens and search tokens together for decoding directly.

B. Masked image modeling

MIM is a self-supervised representation learning algorithm designed to enhance the model's representation ability. The main concept revolves around performing block and random masking operations on the input image, followed by predicting the masked areas. Recently, iGPT [18] first proposes a transformer model that predicts position pixels from low-resolution pixel sequences. Further exploration of the impact of using the Transformer + self-supervised training mode on images expands the understanding of the model's capabilities in image analysis tasks. Masked Autoencoder (MAE) [5] exhibited competitive performance and efficiency, it takes only visible patches and mask tokens representing the masked patches as input, which limits the representation quality. The ConvMAE [19] framework utilizes masked convolution to prevent information leakage within the convolution blocks. It shows that a multi-scale hybrid convolution-transformer can acquire more distinctive representations through the mask auto-encoding scheme. LocalMIM [20] proposes a local multi-scale reconstruction task, where the lower and higher layers reconstruct fine-scale and coarse-scale supervision respectively, offering enhanced reconstruction capabilities across multiple scales.

Our approach aims to thoroughly explore the potential of MIM in the field of visual tracking. Adhering to the characteristics of the MIM task, we have developed a self-decoder to enhance the self-recognition ability of target markers. Moreover, we guide the decoder to cross-identify target markers across different frames, facilitating alignment between template and search regions for improved tracking performance.

III. Method

In this section, we introduce the transformer tracker with masked mutual guidance in detail. Figure 3 illustrates the developed tracking structure. It consists of an autoencoder, a self-decoder, and a guided decoder. The encoder is tasked with amalgamating the visual features from both the template region and the search region, while the decoder is responsible for reproducing the masked template and search tokens.

A. Visual Representation Learning

The encoder of this model maps the input X_v to the potential representation Z_v, which is formed by vanilla ViT [21]. The input are three images as comprising a template image $z \in \mathbb{R}^{3\times H_z\times W_z}$, an online template images $o \in \mathbb{R}^{3\times H_o\times W_o}$, and search images $x \in \mathbb{R}^{3\times H_x\times W_x}$. They are split and flattened into sequences of patches $z_p \in \mathbb{R}^{N_z\times(3\times p^2)}$, $o_p \in \mathbb{R}^{N_o\times(3\times p^2)}$, and $x_p \in \mathbb{R}^{N_x\times(3\times p^2)}$, where $P \times P$ is the resolution of each patch, and $N_z = H_zW_z/p^2$, $N_o = H_oW_o/p^2$, $N_x = H_xW_x/p^2$ are the number of patches

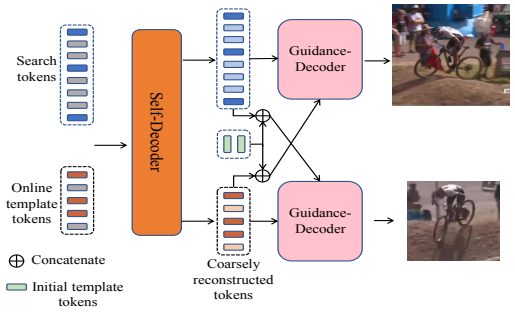

Fig. 4. The decoder architecture comprises a self-decoder and a guide-decoder. The self-decoder initially conducts a coarse reconstruction process. Subsequently, it combines the original template tokens output by the encoder with the self-decoder's output, which serves as input to the guided decoder for further refinement.

of template and search region respectively. The 2D patches are mapped to 1D tokens with C dimensions through a linear projection. After adding the positional embedding, we get the input sequences of the backbone, including the template tokens $X_z = N_z + P_z,\ P_z \in \mathbb{R}^{N_z \times C}$, the online template tokens $X_o = N_o + P_z$, and the search tokens $X_x = N_x + P_x,\ P_x \in \mathbb{R}^{N_x \times C}$.

X_z and X_x are stitched together to form the input patch X_v. Then it sends into a sequence of transformer blocks that are based on self-attention [22], where the features of Q_v, K_v, and V_v can be expressed as follows.

$$\begin{aligned} Q_v &= Cat(X_z, X_o, X_x)W_q, \\ K_v &= Cat(X_z, X_o, X_x)W_k, \\ V_v &= Cat(X_z, X_o, X_x)W_v, \end{aligned} \tag{1}$$

where d is the number of channels. The Q_v, K_v, and V_v $\in \mathbb{R}^{C \times (X_v) \times d}$ are query, key and value matrices separately with mapping weights W_q, W_k, $W_v \in \mathbb{R}^{C \times d}$.

The global self-attentive output Att can be wirtten as:

$$Attn = Softmax(\frac{Q_v K_v^T}{\sqrt{d}})V_v \tag{2}$$

The forward propagation process of mapping the input X_v to the potential representation Z_v can be written as:

$$\begin{aligned} X_v^* &= X_v + Attn(LN(X_v)), \\ Z_v &= X_v^* + FFN(LN(X_v^*)), \end{aligned} \tag{3}$$

where FFN is a feed forward network, LN means Layernorm and Attn is self-attention module.

B. Mask Mutual Guidance Strategy

The encoder combines a set of template tokens and search tokens into a unified embedding. Subsequently, we partition the encoded tokens into template tokens and search tokens. Then, tokens are randomly sampled with masking rate of 75% for each category. Our decoder then forecasts pixel values for each masked token, and the resulting decoder output is reshaped to generate a reconstructed image. Our loss function is determined by calculating the mean squared error(MSE) between the reconstructed image and the original image, specifically focusing on the masked tokens.

Decoder. During the feature alignment stage, the decoder comprises a self-decoder and a guided-decoder, as shown in Figure 4. The original template tokens are retained unchanged, serving as a crucial component of guided learning. The original image is reconstructed by extracting the masked online template tokens and search tokens from self-decoder $f_s(o, s) \rightarrow O_g, X_g$.

Taking the search branch as an example, following the initial reconstruction of the original image by the self-decoder, the original template tokens and the initially constructed online template tokens are concatenated to form the guide tokens for the search image. Here, the K_g and V_g pairs are utilized as inputs to the decoder. Utilizing the initially constructed search tokens as query tokens, acting as the decoder Q_g, the search image is refined by guiding the decoder $f_g(X_g, (z, O_g)) \rightarrow X$. The computational formula is expressed as follows:

$$\begin{aligned} Q_g &= f_s(s)W_q, \\ K_g &= Cat(z, f_s(o))W_k, \\ V_g &= Cat(z, f_s(o))W_v, \\ X &= Softmax(\frac{Q_g K_g^T}{\sqrt{d}})V_g \end{aligned} \tag{4}$$

where z, o and s represent the original template tokens output by the encoder, the masked dynamic template tokens and the search template tokens. We utilize the MSE to calculate the loss L_m, which measures the discrepancy between the reconstructed and original images. This guided image reconstruction process reinforces the target based on the varying frame shapes, enhancing the tracker's capability to align with the target.

Head. Since our visual feature extraction block employs a non-hierarchical backbone network with a step size of 16, it inherently lacks multi-scale information. We employ a pyramid-shaped corner head to acquire three images at different resolutions (1/4, 1/8, and 1/16) through three convolutional and interpolation layers to remedy this limitation. Subsequently, convolutional fusion is conducted, enabling the tracking head to possess multi-scale capabilities. for MMG training, a combination of MSE loss, L_1 loss and CIoU loss is employed as follows:

$$L = \lambda_{L_m} L_m + \lambda_{L_1} L_1(P_i, G_i) + \lambda_{L_{ciou}}(P_i, G_i) \tag{5}$$

where $\lambda_{L_m} = 1$, $\lambda_{L_1} = 5$, and $\lambda_{L_{ciou}} = 5$ are the weights of the three losses, G_i is the ground-truth bounding box and P_i is the predicted bounding box of the targets. For MMG inference, we forego the decoder branch and instead rely solely on visual feature extraction and tracking heads to execute inference.

Online Update. To capture the status of the target across different frames, we have implemented an online template update module. However, low-quality online templates may

TABLE I

COMPARISON WITH STATE-OF-THE-ARTS ON THREE LARGE-SCALE BENCHMARKS: LASOT [8], TRACKINGNET [9] AND GOT-10K [10]. THE BEST TWO RESULTS ARE SHOWN IN RED AND BLUE FONTS.

Method	Source	LaSOT [8]			TrackingNet [9]			GOT-10k [10]		
		AUC	P_{Norm}	P	AUC	P_{Norm}	P	AO	$SR_{0.75}$	$SR_{0.5}$
SiamFC [12]	ECCVW16	33.6	42	33.9	57.1	66.3	53.3	34.8	9.8	35.3
SiamRPN++ [13]	CVPR2019	49.6	56.9	49.1	73.3	80	69.4	51.7	32.5	61.6
SiamFC++ [23]	AAAI2020	54.4	62.3	54.7	75.4	80	70.5	59.5	47.9	69.5
Ocean [24]	ECCV2020	56.0	64.8	-	-	-	-	61.1	47.3	72.1
PrDiMP [25]	CVPR2021	63.9	-	61.4	78.4	83.3	73.1	67.1	58.3	77.7
TransT [4]	CVPR2021	64.9	73.8	69.0	81.4	86.7	80.3	67.1	60.9	76.8
STARK [3]	ICCV2021	67.1	77.0	-	82.0	86.9	-	68.8	64.1	78.1
CSWinTT [2]	CVPR2022	66.2	75.2	70.9	81.99	86.7	79.5	69.4	63.6	80.4
SBT [26]	CVPR2022	66.7	-	71.1	-	-	-	70.4	64.7	80.8
MixFormer [16]	CVPR2022	69.2	78.7	74.7	83.1	88.1	81.6	70.7	67.8	80.0
SwinTrack-B [27]	NeurIPS2022	69.6	78.6	74.1	82.5	87.0	80.4	69.4	64.3	78.0
AiATrack [15]	ECCV2022	69.0	79.4	73.8	82.7	87.8	80.4	68.6	63.2	80.0
SimTrack [28]	ECCV2022	69.3	78.5	-	82.3	86.5	-	68.6	-	-
OSTrack [17]	ECCV2022	69.1	78.7	75.2	83.1	87.8	82.0	71.0	68.2	80.4
Unicorn [29]	ECCV2022	68.5	76.6	74.1	83.0	86.4	82.2	70.7	67.8	80.0
TATrack-B [30]	AAAI2023	69.4	78.2	74.1	83.5	88.3	81.8	73.0	68.5	83.3
CTTrack [7]	AAAI2023	67.8	77.8	74.0	82.5	87.1	80.3	71.3	70.3	80.7
MAT [6]	CVPR2023	67.8	77.3	-	81.9	86.8	-	67.7	-	78.4
ROMTrack [31]	ICCV2023	69.3	78.8	75.6	83.6	88.4	82.7	72.9	70.2	82.9
F-BDMTrack [32]	ICCV2023	69.9	79.4	75.8	83.7	88.3	82.6	72.7	69.9	82.0
DETRack [33]	Preprint2023	69.0	78.9	75.1	83.2	88.3	83.1	72.9	69.9	82.1
GRM [34]	CVPR2023	69.9	79.3	75.8	84.0	87.7	83.3	73.4	70.4	82.9
MMG	Ours	70.3	80.1	76.2	84.0	88.5	83.4	73.5	70.2	83.0

fail to depict the target's status accurately. Thus, we introduced an online update module structured similarly to the visual feature extraction module. In this module, template tokens, search tokens, and a cls token are concatenated to produce the final cls output. Subsequently, an MLP layer and sigmoid activation are applied to activate this output, resulting in the final score value. Templates with scores exceeding 0.5 are deemed of high quality and subsequently updated.

IV. EXPERIMENTS

A. Implementation Details

Model. The vanilla ViT-Base model [21], pre-trained with MAE [5], is utilized as the backbone for joint feature extraction and relation modelling. The head consists of 12 stacked Conv-BN-ReLU layers for the output. We employ 8 self-decoders and 8 guided decoders. Dynamic templates are updated when the default update interval of 100 is reached (note that this interval may vary slightly with different test datasets). The template with the highest predicted score within the interval is selected to replace the previous one.

Train. We implemented our trackers using Python 3.7 and PyTorch 1.10.0, training MMG across 4 RTX3080Ti GPUs. For training, we used specific splits of COCO [35], LaSOT [8], GOT-10k [10] (with 1k forbidden sequences from the GOT-10k training set removed, following convention [3]), and TrackingNet [9]. Common data augmentations, including horizontal flip and brightness jittering, were applied during training. Each GPU handled 20 image pairs, resulting in a total batch size of 80. We employed the AdamW optimizer [36] with a weight decay of 10^{-4}. The initial learning rate was set to 1×10^{-5} for the backbone and 1×10^{-4} for other parameters. Training spanned 500 epochs, each containing 60k image pairs, with the learning rate decreased by a factor of 10 after 400 epochs. The sizes of search images and templates were 256×256 pixels and 128×128 pixels, respectively.

B. Comparison with State-of-the-arts

We verify the performance of our proposed MMG on eight benchmarks, including LaSOT [8], LaSOT$_{ext}$, TrackingNet [9], GOT10k [10], UAV123 [38] and VOT2020 [11].

LaSOT. LaSOT [8] is a large-scale, long-term tracking benchmark containing 280 videos with an average length of 2448 frames in the test set. We compare the result of the MMG withe previous SOTA tracker in Tab. I. The result show that proposed tracker obtained 70.3 AUC (Accuracy), 80.1 P_{Norm} (Normalize Precision) and 76.2 P (Precision) scores for MMG with same input resolutionwhich outperforms the previous best result by OSTrack [17] and CTTrack [7]. We demonstrate the performance of our proposed MMG in the LaSOT dataset when facing challenges such as motion blur, fast movement, perspective switching, and similar object interference in Figure 5.

GOT-10K. GOT-10k [10] comprises over 10 thousand videos and is notably challenging due to the absence of

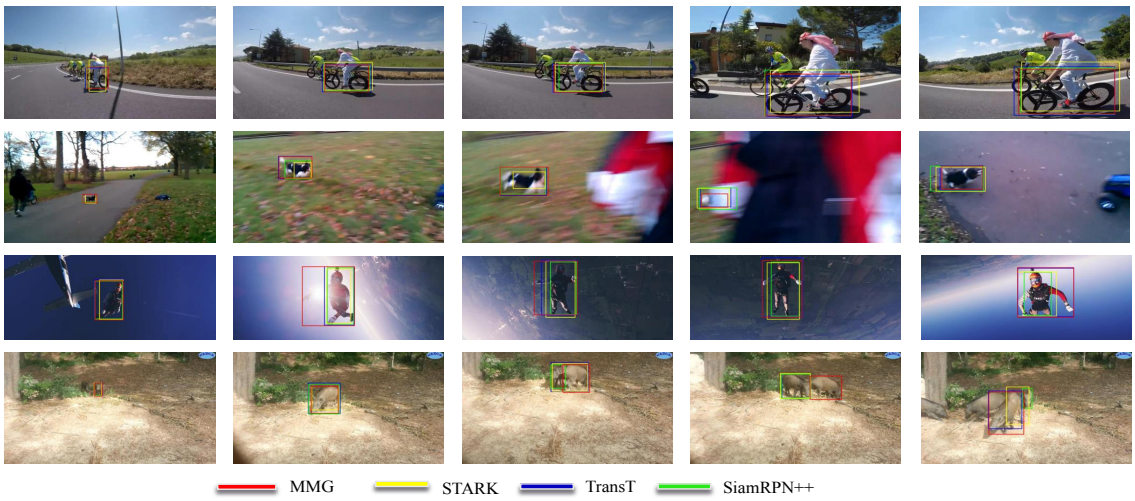

Fig. 5. We present qualitative tracking results produced by Stark, TransT, SiamRPN++, and our baseline trackers MMG on 4 challenging sequences (bicycle-7, dog-1, preson-10, pig-2) from LaSOT, as shown from top to bottom. The results demonstrate that our improved tracker significantly enhances long-term tracking performance compared to the other trackers.

TABLE II

COMPARISON WITH STATE-OF-THE-ARTS ON THREE BENCHMARKS: LASOT$_{ext}$ [37] AND UAV123 [38] AUC(%) SCORES ARE REPORTED. THE BEST TWO RESULTS ARE SHOWN IN RED AND BLUE FONTS.

	STARK [3]	TransT [4]	KeepTrack [39]	SwinTrack [27]	OSTrack [17]	CTTrack [7]	DETRack [33]	ROMTrack [31]	F-BDMTrack [32]	Ours
LaSOT$_{ext}$	-	-	48.2	47.6	47.4	-	-	48.9	47.9	49.2
UAV123	68.2	68.1	68.2	-	68.3	68.8	68.7	-	69.0	69.4

TABLE III

STATE-OF-THE-ART COMPARISON ON VOT2020 [11]. THE BEST TWO RESULTS ARE SHOWN IN RED AND BLUE FONTS. OUR TRACKERS USE ALPHA-REFINE [40] TO PREDICT MASKS.

	SiamMask [41]	D3S [42]	AlphaRaf [40]	OceanPlus [24]	RPT [43]	DualTFR [44]	STARK [3]	ToMP [45]	MixFormer [16]	Ours
EAO	0.321	0.439	0.305	0.491	0.53	0.528	0.505	0.497	0.528	0.541
Accuracy	0.624	0.699	0.492	0.685	0.7	0.755	0.759	0.75	0.734	0.778
Robustness	0.648	0.769	0.745	0.842	0.869	0.836	0.817	0.798	0.857	0.84

overlap between the classes in the provided training and test sets. To ensure adherence to the prescribed protocol, our model was trained using the train set and the results were evaluated on the official server. As depicted in Tab. I, MMG exceeds OSTrack [17] by 2.4% in AO. MMG showcases an SR0.5 score of 83.0%, surpassing OSTrack by 2.6%. Similarly, with an SR0.75 score of 70.2%, MMG outperforms OSTrack by 2.0%.These results highlight the excellent visual feature extraction and feature alignment capabilities of our tracker.

TrackingNet. With an extensive collection of over 30K videos and more than 14 million dense bounding box annotations, TrackingNet [9] samples videos from YouTube, encompassing real-life object categories and scenes. The results, demonstrated in Tab. I, illustrate MMG's superior performance, achieving a leading AUC score of 84.0%. This achievement marks a significant improvement of 0.3% over the prior best results with the same input resolution.

LaSOT$_{ext}$. LaSOT$_{ext}$ [37] is a recently released LaSOT extension, which consists of 150 additional videos from 15 new categories. The experimental results, as shown in Tab. II, indicate that the previous state-of-the-art tracker ROMTrack achieved an AUC of 48.9 while operating at approximately 62 fps by designing a complex association network in the same input resolution. In contrast, our proposed MMG exhibits superior performance and operates at a speed of 95 fps.

UAV123. UAV123 [38] is an extensive dataset with 123 sequences having an average length of 915 frames, captured from low-altitude UAVs. Our performance results on the UA123 dataset are presented in Tab. II, where MMG stands out, outperforming all other trackers.

VOT2020. The VOT2020 benchmark, curated by Kristan et al. [11], comprises a corpus of 60 intricate video sequences, each laden with challenges germane to visual tracking. VOT employs binary segmentation masks as the canonical ground truth for evaluation. In the context of segmentation mask prediction, we augment our MMG framework with the Alpha-Refine [40] module. As shown in Tab. III, our MMG approach achieves 0.541 EAO outperforming

on mask evaluations.

TABLE IV

ABLATION EXPERIMENTS ON DECODER TYPES, WHERE SELF-DECODER REPRESENTS THE SELF-DECODING COARSE RECONSTRUCTION OF THE IMAGE, ONLINE2SEARCH REPRESENTS THE FINE RECONSTRUCTION FROM THE SEARCH IMAGE, AND SEARCH2ONLINE REPRESENTS THE FINE RECONSTRUCTION OF THE TEMPLATE IMAGE.

Decoder Type	Self-Decoder	Online2 Search	Search2 Online	LaSOT	GOT10K
	-	-	-	68.7	70.8
	✓	-	-	69.1	72.2
	✓	✓	-	70.0	73.1
	✓	-	✓	69.5	72.8
	-	✓	✓	69.7	73.4
	✓	✓	✓	**70.3**	**73.5**

TABLE V

AN APPROPRIATE MASK RATIO CAN BRING STRONGER FEATURE ALIGNMENT CAPABILITIES TO TRACKING. THE DEFAULT MASK RATIO IS 75%.

Masking ratio	0%	25%	50%	75%
LaSOT	68.8	69.4	69.9	**70.3**
GOT10K	71.4	72.6	73.1	**73.5**

C. Ablation Study and Analysis

Impact of decoder components on tracker performances. Our decoder includes a self-decoder and a guided decoder, and its structure is shown in Figure 4. Table IV shows the different combinations of the decoder, the self-decoding of the dynamic template image and the search image, the coarsely reconstructed dynamic template image, and the original template stitching together to guide the fine reconstruction of the search image, and the coarsely reconstructed search image and the original template. Stitching together guides the fine reconstruction of dynamic template images. The results highlight the crucial role of our guided decoder in visual feature alignment. While the pure self-decoder enhances self-information aggregation of the target, the guided decoder fuses target status across different frames. It guides the tracker in recognizing the target in diverse scenes, thereby enhancing tracker robustness. and guide the tracker to recognize the target in various scenes to enhance the robustness of the tracker.

Different Masking ratio. Based on research conducted on masking strategies in MAT, the grid-wise strategy yields less favorable results than the random masking strategy. Therefore, we have explored the impact of different random masking ratios on visual feature alignment. The effects of mask ratio on tracker performance are summarized in Table V. Following the findings of MAE, we default to a mask ratio of 75%. Experimental results have corroborated its suitability as the most effective mask ratio.

Online Template Updating. We evaluated to assess the impact of guided decoding using dynamic templates, and the experimental results are presented in Table VI. The findings reveal that solely relying on online updates has a limited impact on the tracker's performance. In contrast, employing dynamic templates for guidance significantly enhances the tracker's feature alignment capabilities.

TABLE VI

ABLATION FOR THE ONLINE TEMPLATE UPDATING COMPONENT. **ONLINE** DENOTES UPDATING THE TEMPLATE AT A FIXED UPDATE INTERVAL. **SCORE** REPRESENTS THE ONLINE TEMPLATE IS ONLY UPDATED WITH HIGH CONFIDENT SAMPLES.

	Online	Score	LaSOT	GOT10K
MMG	-	-	69.9	72.8
MMG	✓	-	68.2	72.1
MMG	✓	✓	**70.3**	**73.5**

V. CONCLUSIONS

In this study, we introduce a tracker featuring an encoder-decoder architecture that facilitates mutual guidance for feature alignment learning. We incorporate an additional learning objective into the tracker to equip the encoder with visual feature extraction and alignment capabilities. This approach addresses a limitation of previous methods, which focused solely on learning the target without considering its morphological transformations in the video sequence. Furthermore, our correlative masked decoder can be integrated into other transformer-based trackers, effectively enhancing tracking performance without compromising speed. Our propose MMG framework has undergone extensive evaluation across six common tracking benchmarks and demonstrates significant improvements compared to existing tracking devices.

REFERENCES

[1] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, "Siamese box adaptive network for visual tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2020.

[2] Z. Song, J. Yu, Y.-P. P. Chen, and W. Yang, "Transformer tracking with cyclic shifting window attention," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2022.

[3] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, "Learning spatio-temporal transformer for visual tracking," in *Proceedings of the IEEE/CVF international conference on computer vision(ICCV)*, 2021.

[4] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu, "Transformer tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2021.

[5] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2022.

[6] H. Zhao, D. Wang, and H. Lu, "Representation learning for visual object tracking by masked appearance transfer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2023.

[7] Z. Song, R. Luo, J. Yu, Y.-P. P. Chen, and W. Yang, "Compact transformer tracker with correlative masked modeling," in *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*, 2023.

[8] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling, "Lasot: A high-quality benchmark for large-scale single object tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2019.

[9] M. Muller, A. Bibi, S. Giancola, S. Alsubaihi, and B. Ghanem, "Trackingnet: A large-scale dataset and benchmark for object tracking in the wild," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[10] L. Huang, X. Zhao, and K. Huang, "Got-10k: A large high-diversity benchmark for generic object tracking in the wild," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[11] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, J.-K. Kämäräinen, M. Danelljan, L. Č. Zajc, A. Lukežič, O. Drbohlav *et al.*, "The eighth visual object tracking vot2020 challenge results," in *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, 2020.

[12] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *Computer Vision–ECCV 2016 Workshops(ECCVW)*. Springer.

[13] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2019.

[14] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, 2020.

[15] S. Gao, C. Zhou, C. Ma, X. Wang, and J. Yuan, "Aiatrack: Attention in attention for transformer visual tracking," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, 2022.

[16] Y. Cui, C. Jiang, L. Wang, and G. Wu, "Mixformer: End-to-end tracking with iterative mixed attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2022.

[17] B. Ye, H. Chang, B. Ma, S. Shan, and X. Chen, "Joint feature learning and relation modeling for tracking: A one-stream framework," in *Computer Vision–ECCV 2022: 17th European Conference(ECCV)*, 2022.

[18] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever, "Generative pretraining from pixels," in *International conference on machine learning*, 2020.

[19] P. Gao, T. Ma, H. Li, Z. Lin, J. Dai, and Y. Qiao, "Convmae: Masked convolution meets masked autoencoders," *arXiv preprint arXiv:2205.03892*, 2022.

[20] H. Wang, Y. Tang, Y. Wang, J. Guo, Z.-H. Deng, and K. Han, "Masked image modeling with local multi-scale reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2023.

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems(NIPS)*, 2017.

[23] Y. Xu, Z. Wang, Z. Li, Y. Yuan, and G. Yu, "Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines," in *Proceedings of the AAAI conference on artificial intelligence(AAAI)*, 2020.

[24] Z. Zhang, H. Peng, J. Fu, B. Li, and W. Hu, "Ocean: Object-aware anchor-free tracking," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 2020.

[25] M. Danelljan, L. V. Gool, and R. Timofte, "Probabilistic regression for visual tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2020.

[26] F. Xie, C. Wang, G. Wang, Y. Cao, W. Yang, and W. Zeng, "Correlation-aware deep tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2022.

[27] L. Lin, H. Fan, Z. Zhang, Y. Xu, and H. Ling, "Swintrack: A simple and strong baseline for transformer tracking," *Advances in Neural Information Processing Systems(NIPS)*, 2022.

[28] B. Chen, P. Li, L. Bai, L. Qiao, Q. Shen, B. Li, W. Gan, W. Wu, and W. Ouyang, "Backbone is all your need: a simplified architecture for visual object tracking," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, 2022.

[29] B. Yan, Y. Jiang, P. Sun, D. Wang, Z. Yuan, P. Luo, and H. Lu, "Towards grand unification of object tracking," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXI*, 2022.

[30] K. He, C. Zhang, S. Xie, Z. Li, and Z. Wang, "Target-aware tracking with long-term context attention," in *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*, 2023.

[31] Y. Cai, J. Liu, J. Tang, and G. Wu, "Robust object modeling for visual tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV)*, 2023.

[32] D. Yang, J. He, Y. Ma, Q. Yu, and T. Zhang, "Foreground-background distribution modeling transformer for visual object tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV)*, 2023.

[33] Q. Wei, G. Zeng, and B. Zeng, "Efficient training for visual tracking with deformable transformer," *arXiv preprint arXiv:2309.02676*, 2023.

[34] S. Gao, C. Zhou, and J. Zhang, "Generalized relation modeling for transformer tracking," 2023, pp. 18 686–18 695.

[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 2014.

[36] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[37] H. Fan, H. Bai, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, M. Huang, J. Liu, Y. Xu *et al.*, "Lasot: A high-quality large-scale single object tracking benchmark," *International Journal of Computer Vision*, 2021.

[38] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for uav tracking," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 2016.

[39] C. Mayer, M. Danelljan, D. P. Paudel, and L. Van Gool, "Learning target candidate association to keep track of what not to track," in *Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV)*, 2021.

[40] B. Yan, X. Zhang, D. Wang, H. Lu, and X. Yang, "Alpha-refine: Boosting tracking performance by precise bounding box estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2021.

[41] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition(CVPR)*, 2019.

[42] A. Lukezic, J. Matas, and M. Kristan, "D3s-a discriminative single shot segmentation tracker," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2020.

[43] Z. Ma, L. Wang, H. Zhang, W. Lu, and J. Yin, "Rpt: Learning point set representation for siamese visual tracking," in *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, 2020.

[44] F. Xie, C. Wang, G. Wang, W. Yang, and W. Zeng, "Learning tracking representations via dual-branch fully transformer networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV)*, 2021.

[45] C. Mayer, M. Danelljan, G. Bhat, M. Paul, D. P. Paudel, F. Yu, and L. Van Gool, "Transforming model prediction for tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, 2022.