Object Tracking Network Based on Deformable Attention Mechanism

Kexin Chen
ckx0225@163.com
Baojie Fan*
jobfbj@gmail.com
xiaobin Guo
roaneguo@gmail.com

Nanjing University
of Posts and
Telecommunications, China

Abstract

Recently, many Transformer-based algorithms have emerged in the field of object tracking. Thanks to the full Attention mechanism in the Transformer structure, these tracking algorithms have achieved competitive results, but such model parameters are often bloated compared with CNN-based. In this paper, we focus on the characteristics of the object tracking task, explore novel interaction between template frames and search frames, and propose DeTrack. The modified model uses a combination of an encoder module based on deformable attention mechanism and an encoder module based on self-attention mechanism for feature interaction. The deformable attention-based encoder can precisely track the target location without focusing on all the pixels, which reduces the number of model parameters and effectively improves the model accuracy. We have achieved state-of-the-art performance on LaSOT, TrackingNet, GOT-10K and VOT2020.

1 Introduction

Generic object tracking is a fundamental but challenging task in computer vision. Given the annotation of the first frame, the tracker needs to locate the target state in each subsequent frame. It has huge potential and a large range of applications such as visual surveillance and augmented reality. To the contrary of object detection, target feature is only captured at the inference phase, which means that no prior information, including class and surroundings, about the object. From our research, facing a complicated environment(e.g., blur, scale variation, color shift, and fast motion, etc.), most existing trackers couldn't handle these scenes excellently.

Recent object tracking networks can be grouped into three categories: Siamese-based object tracking networks, discriminative object tracking networks and Transformer-based object tracking networks. The representative algorithms of Siamese-based object tracking networks are SiamFC[1], SiamRPN[18], SiamRPN++[19], etc. These algorithms determine the target position by calculating the similarity metric between the template frame and the search frame, which has a good real-time performance but the model is not robust enough. Discriminative object tracking networks include ATOM[9], DiMP[2], PrDiMP[10],

etc. These algorithms take into account the background information of the target to improve the robustness of the model but lack real-time performance. The trackers described above all boil down to a convolution-based model. With the rise attention of the Transformer, more and more Transformer-based work is occurring in the field of computer vision, such as DETR[4] in object detection, TransT[5] and STARK[37] in object tracking. STARK is inspired by the DETR algorithm for object detection and designed an end-to-end target tracking algorithm including Encoder module,Decoder module, and one query embedding, which has achieved competitive results on major datasets.

It is worth considering that the object detection task is class-specific, while the object tracking task is target-specific. In DETR[4], 100 object queries are set up to match 80 classes in the coco dataset through training, while in STARK-S version a query is set up but does not really work. In the STARK-ST version the query is used to provide a basis for model updates, and as discussed in the recently proposed SwinTrack[20], the query setting is not necessary for the target tracking task.

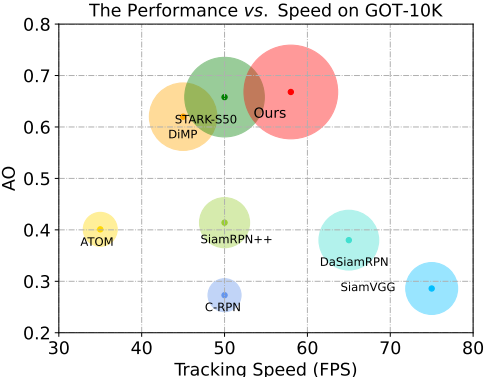

Figure 1: Comparison with state-of-the-arts on GOT-10K. We visualize the Success performance with respect to the FPS tracking speed. Better viewed in color.

When the Transformer structure was applied to computer vision tasks, researchers in a wide range of computer vision tasks realised that the Transformer structure could break through the various limitations of convolution, and that the structure's self-attention mechanism could break the spatial limitations of convolution operations by capturing correlations between pixels at longer distances in the same image. Whereas traditional convolutional neural networks for feature capture is limited to a fixed size of the convolution kernel. As the self-attention mechanism considers each pixel value on the feature map, it increases the computational complexity and requires more attention to the target itself rather than the global picture for computer vision tasks. Therefore, this paper focuses on the characteristics of the target tracking task and explores novel target interaction methods, using a combination of a deformable attention-based encoder module and a self-attention encoder module for feature interaction. The deformable attention based encoder can precisely track the target location without focusing on all pixels. The model discards the decoder structure and query settings, which reduces the number of model parameters and effectively improves the model accuracy. Through the novel feature interaction, we have designed a new deformable attention-based tracker, DeTrack. It has demonstrated the effectiveness of our approach through extensive experiments.

The main innovations of DeTrack are summarised below:

- Exploring a new paradigm for tracking tasks and reducing the number of model parameters by removing the Decoder structure from the Transformer-based model.
- Feature interactions are first performed using an encoder structure with deformable attention mechanism, and then the results are further enhanced using an encoder with self attention mechanism. The proposed algorithm achieves the best combination of the two.

2 Related Work

2.1 Single Object Tracking

Siamese-based trackers: The pioneering work, Fully convolutional Siamese network[1] learns the similarity metric between the target template and the search region, which adopt offline training strategy achieving SOTA results while with a real-time FPS. Due to its big success, many researchers have been exploring this direction and have proposed many improved methods. SiamRPN[18] contains the Siamese subnetwork for feature extraction and the region proposal networks, composed of a classification branch and a bounding box regression branch. SiamRPN++[19] adopts ResNet[15] as the feature extraction network, which removes the stride of the last two blocks and increases the expansion of the convolution to increase the receptive field. However, anchor-based trackers introduce many hyperparameters which need to be carefully designed. With the emergence of anchor-free trackers, these problems have been resolved to vary degrees. SiamFC++[36] addresses existing object tracker issues, and proposes four guidelines based on anchor-free target estimation. they add assess quality branch and used various loss joint training, which greatly improved the tracking performance.

Discriminative online learning based trackers: Unlike Siamese-based trackers, Discriminative trackers make full use of the target and background appearance information to learn an adaptive filter online, which is crucial for distinguishing the target from the background and the robustness improvement. CFNet[31] combines the Correlation Filter layer with the Siamese network, which benefits from end-to-end training and achieves good performance under shallow features. Discriminative online learning methods need to learn as good as possible target representation by only a few images, which is similar to meta-learning methods. Recently, Bhat et al.[2] designed DiMP tracker based on the meta-learning architecture, which predicts discriminative target model weights and then employed for coarse localization. FCOT[7] fused the low-resolution score and the high-resolution score map to predict the target center, which improves the discrimination of similar objects.

2.2 Vision Transformers

Originally, Transformer[32] was applied for processing machine translation tasks in NLP[12, 28] tasks and achieved state-of-the-art results. Recently, it has been witnessed a rapid boost in computer vision tasks due to the increasing research of transformer architecture, which viewed as the potential replacement of traditional convolutional neural networks such as ResNet. ViT[13] divides the input image into patches and then maps these patches into embeddings, which can be viewed as tokens in NLP tasks. Swin Transformer[22] proposes a hierarchical Transformer with shifted windows, where limiting self-attention computation within. Despite being effective, the shifted windows may have uneven sizes. PVT[35] proposes anothor solution, introducing pyramid structure into Transformer so that it can be seamlessly connected to various downstream tasks. Twins[6] inspired by the widely-used separable depthwise convolutions and proposed spatially separable self-attention for transformer backbone design. Referring to the idea of DETR[4], STARK[37] introduced Transformer into object tracking and achieved success. However, we observe that the encoder and decoder of its feature interaction part are redundant. Meanwhile, our idea of deformable attention comes from Deformable DETR[41]. By applying it to object tracking, the performance of the tracker is improved while reducing the model parameters.

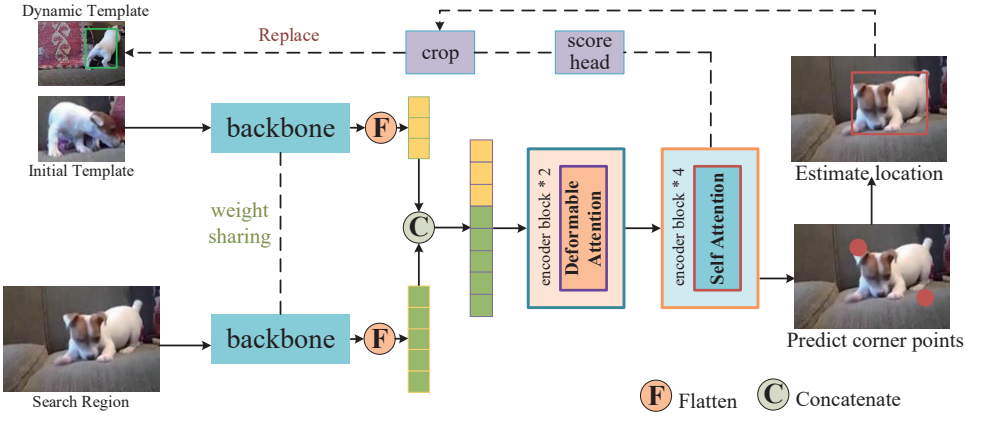

Figure 2: A diagram of the framework of our model. The dotted line part is our online update version improved on the basis of DeTrack, named DeTrack-T.

3 Approach

In this section, we propose a object tracker based on deformable attention mechanism, named DeTrack,as shown in Figure 2. Our tracker consists of the following components: the feature extraction network, the feature interaction module, and the corner point prediction module.

Our tracker is based on an improvement of the STARK-S50. The backbone network extracts the image features of the template frame and the search frame using common network parameters. The extracted template features and search features are then flattened and concatenated together by dimension and fed into the target fusion perception module to form the target perception feature map. Finally, the target position is predicted directly by the prediction head.

Backbone: Our tracking model shows good compatibility with various backbone networks. Since it is an improvement based on STARK, the backbone network is not the focus of the proposed method, so the setting of the backbone network in this paper is consistent with STARK. In addition, this paper only improves the algorithm on ResNet50.The specific structure of the backbone network is: the last layer of the ResNet is removed to accommodate the requirements of the tracking task for sensory field and step size. After the template image z containing the target and the search image x are fed into the backbone network to extract features, the respective feature maps are obtained and are denoted as $f_z \in \mathbb{R}^{C\times\frac{H_z}{S}\times\frac{W_z}{S}}$ and $f_x \in \mathbb{R}^{C\times\frac{H_x}{S}\times\frac{W_x}{S}}$,where C represents the number of feature map channels, H_z,W_z represents the height and width of the template image, similarly H_x,W_x represents the height and width of the search image, and S represents the network step size

Target Perception Module: The proposed novel interaction is a combination of a deformable encoder based on a deformable attention mechanism and an encoder based on a self-attentive mechanism. The deformable encoder based on the deformable attention mechanism has a strong perceptual capability, allowing for a more precise tracking of the target location without focusing on all pixels, and a more focused focus on the pixels around the target.For a given input feature map $f \in \mathbb{R}^{C\times H\times W}$,The deformable attention mechanism can be expressed as follows:

$$DA\left(Z_q, p_q, x\right) = \sum_{m=1}^{M} W_m \left[\sum_{k=1}^{K} A_{mqk} \cdot W_m^{'} x\left(p_q + \Delta p_{mqk}\right)\right] \tag{1}$$

where Z_q represents the feature query sequence, P_q represents the two-dimensional reference points, m is the index of the multi-attention head, k represents the sampling point index, and K denotes the total number of sampling points ($K \ll HW$). Δp_{mqk} ($\Delta p_{mqk} \in \mathbb{R}^2$) and A_{mqk}($A_{mqk} \in [0,1]$) denote the sampling offset and attention weight of the m attention head at the k sample point position, respectively. Both Δp_{mqk} and A_{mqk} are obtained from the feature query sequence Z_q. The feature query sequence is fed into a linear projection operator for the 3MK channels, with the first 2MK channels encoding the sampled reference point offset Δp_{mqk} , while the remaining one MK channel performs a softmax operation to obtain the attention weights.For the feature perception module, we employ two deformable encoders based on a deformable attention mechanism followed by four encoders based on a self-attentive mechanism.

The application of deformable attention is shown in Figure 3, we flatten and concatenate the features of the template and the search region after extracting features from the backbone. Deformable attention enables the encoder to pay more attention on the features of the target itself and its surroundings, as well as the target feature interaction between the template and the search region. Meanwhile, less weight is assigned to the attention of the features between the background and the background (the dotted line in the figure).

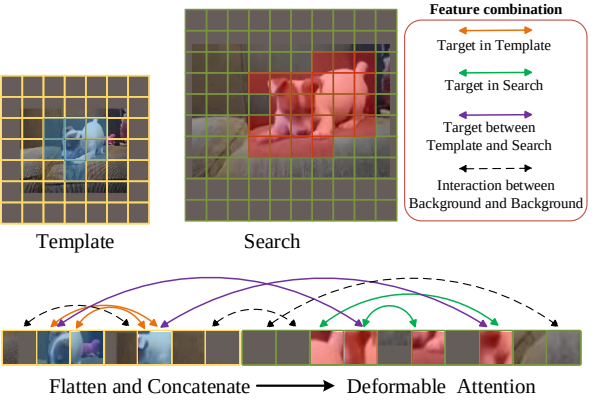

Figure 3: The application process of deformable attention.

Prediction Head: This part is consistent with the STARK prediction model, but differs from STARK in that STARK performs a dot-multiplication operation between the feature map output by the decoder and the feature map output by the encoder in order to further enhance the features, and since the proposed algorithm in this paper discards the decoder, we directly use the features output by the feature-aware module for corner-point prediction.We also adopt a joint optimization network with L_1 loss and GIoU loss, and the loss function can be expressed as:

$$L = \lambda_1 L_{GIoU}\left(y, \hat{y}\right) + \lambda_2 L_1\left(y, \hat{y}\right) \tag{2}$$

where $y, \hat{y}$ represents the true value of the target frame and the predicted value of the network, respectively, and λ_1, λ_2 is the value of the respective loss weights. The λ_1, λ_2 are set to 5 and 2, respectively.

Template Online Update: For the large-scale deformation and appearance changes of the target during the tracking process, the online update of the template is necessary. We concatenate template features $z \rightarrow (B \times H_z W_z \times C)$ and search features $x \rightarrow (B \times H_x W_x \times C)$ after feature extraction. Where B is the batchsize, HW is the product of height and width, and C is the number of channels.

$$\begin{aligned} F &= SA(DA(Concat(z,x))) \\ z',x' &= DeConcat(F) \\ p_i &= MLP(x') \end{aligned} \tag{3}$$

Refer to Equation 3, after deformable attention (DA) and self-attention (SA), we deconcat the encoded features F. We only take the search feature $x' \rightarrow (B \times H_{x'} W_{x'} \times C)$ and feed it into a 3-layer MLP. The output channel of the last layer of the MLP is set to 1, and the output is activated with a sigmoid function to obtain the confidence score p_i of the template update. We utilize the cross-entropy loss, as follows:

$$L_{score} = y_i \log p_i + (1 - y_i) \log (1 - p_i) \tag{4}$$

y_i is the ground-truth box label, and p_i is the predicted confidence score. During inference, the update threshold is set to 0.5, and the template is updated every 50 frames.

We also adopt two-stage training. In the first stage, only the prediction bounding bboxes are trained, the version DeTrack. Confidence scores for DeTrack-T are additionally trained in the second stage. Unlike STARK-S50[37] and STARK-ST50[37], we only take the search features output by the encoder as input to the subsequent prediction head.

4 Experiments

4.1 Implementation details

The algorithm proposed in this chapter is based on Python and the deep learning framework Pytorch on 8 Nvidia 2080ti GPUs. The backbone network is ResNet50, trained with pre-training parameters from ImageNet[30], and the step size of the fourth layer of the backbone network is set to 1 to accommodate the tracking task step size of 16. The encoder part is composed of a 2-layer deformable encoder based on the deformable attention mechanism and a 4-layer self-attentive encoder based on the self-attentive mechanism, both of which use 8-head attention. The algorithm is trained using the training set part of the TrackingNet[25], LaSOT[14], GOT-10K[16], and COCO[21] datasets, with a template image input size of 128×128 and a search image input size of 320×320. Data enhancement strategies such as rotation, flip and blur are applied to the input images. The model was trained using the AdamW optimiser for 500 epochs, with the weight decay coefficient set to 10e-4 and the batchsize set to 30. The first three network parameters of the backbone network were frozen, and the fourth layer was trained with the learning rate of the backbone network set to 10e-5 and the rest of the model set to 10e-4. After 400 epochs, the learning all dropped by a factor of ten. The model testing phase, which consisted only of forward passing of the algorithm and changing the coordinates predicted by the model from the search region to the original image, did not involve any additional post-processing similar to cosine windows.

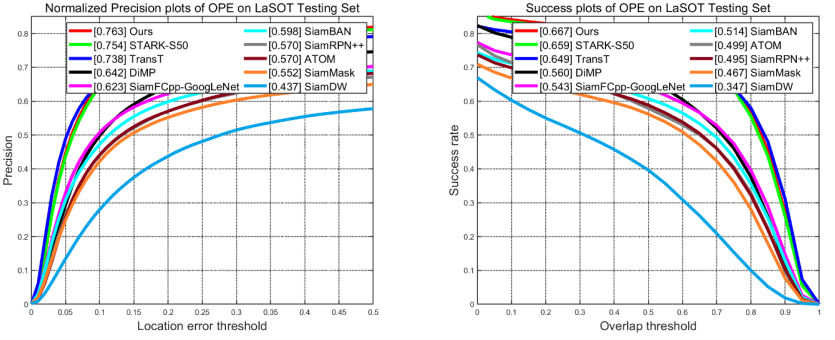

Figure 4: Comparison among existing State-of-the-art trackers on the LaSOT dataset. Better viewed with zooming in.

4.2 Comparison with the state of the art

Results on LaSOT Benchmark: LaSOT[14] is a recently proposed benchmark for evaluating single object tracking, which constructs following the five principles of large-scale, high-quality, dense annotation, long-term tracking, category balance, comprehensive labeling and covers various object categories in different contexts, including 70 object categories. Most categories were selected from ImageNet 1,000 categories. Eventually, the researchers formed a large-scale data set by collecting 1,400 sequences and 3.52 million frames of YouTube[29] videos. The average video length of LaSOT is 2512 frames (that is, 30 frames and 84 seconds per second). The shortest video contains 1000 frames (33 seconds), and the longest video contains 11397 frames (378 seconds). This Benchmark measures the performance of the tracker through three indicators of normalized precision, precision and success. In addition, Figure 4 shows the normalized precision and success rate of the proposed DeTrack compared with other advanced algorithms such as DiMP, TransT and STARK-S50. At the same time, as can be seen from Table 1, our online updated version DeTrack-T has comprehensively surpassed STARK-ST50 in terms of success rate, normalized precision and precision. The visualization results of representative trackers are shown in Figure 5.

Results on TrackingNet Benchmark: TrackingNet[25] is a large scale short term target tracking dataset that applies the target detection dataset YouTubeBB to target tracking, filling the gap of a very large dataset for target tracking and enriching the challenges that target trackers have to face. The dataset is rich in target categories and contains a total of approximately 30,000 videos, of which the test set contains 511 video sequences, using the same one-time evaluation (OPE) approach as the OTB2015 dataset, with the main evaluation criteria being success rate and normalisation criteria, as shown in Table 1. Compared to STARK-ST50, our proposed DeTrack-T improves the AUC, P_{Norm} and P metrics by 1.0%, 1.1% and 1.1%, respectively. The performance on such a large dataset as TrackingNet demonstrates the strong model generalization capability of our proposed tracker.

Results on GOT-10K Benchmark: GOT-10K[16] is a recently proposed large high-diversity benchmark for generic object tracking. It contains more than 10,000 videos composed of five categories: Animals, Artifact, Person, Natural Object, and Part, which can be further subdivided into 563 target categories and no overlap in object classes between train split data and test split data. For a fair comparison, it should be ensured that trackers are evaluated with

Tracker	Source	LaSOT[14]			TrackingNet[25]			GOT-10K[16]		
		AUC	P_{Norm}	P	AUC	P_{Norm}	P	AO	$SR_{0.75}$	$SR_{0.5}$
DeTrack-T	Ours	67.6	77.8	71.9	82.3	87.4	79.0	69.0	63.4	78.8
DeTrack	Ours	66.7	76.3	71.3	81.2	86.3	78.0	68.1	62.7	77.5
STARK-ST50[37]	ICCV2021	66.4	76.3	71.2	81.3	86.1	78.1	68.0	62.3	77.7
STARK-S50[37]	ICCV2021	65.9	75.4	-	80.3	85.1	-	67.2	61.2	76.1
KeepTrack[24]	ICCV2021	67.1	77.2	70.2	-	-	-	-	-	-
DTT[33]	ICCV2021	60.1	-	-	79.6	85.0	78.9	63.4	51.4	74.9
TransT[5]	CVPR2021	64.9	73.8	69.0	81.4	86.7	80.3	67.1	60.9	76.8
TrDiMP[33]	CVPR2021	63.9	-	61.4	78.4	83.3	73.1	67.1	58.3	77.7
TrSiam[33]	CVPR2021	62.4	-	60.0	78.1	82.9	72.7	66.0	57.1	76.6
KYS[3]	ECCV2020	55.4	63.3	-	74.0	80.0	68.8	63.6	51.5	75.1
Ocean-online[40]	ECCV2020	56.0	65.1	56.6	-	-	-	61.1	47.3	72.1
Ocean-offline[40]	ECCV2020	52.6	-	52.6	-	-	-	59.2	-	69.5
PrDiMP50[11]	CVPR2020	59.8	68.8	60.8	75.8	81.6	70.4	63.4	54.3	73.8
SiamAttn[39]	CVPR2020	56.0	64.8	-	75.2	81.7	-	-	-	-
SiamFC++[36]	AAAI2020	54.4	62.3	54.7	75.4	80.0	70.5	59.5	47.9	69.5
DiMP50[2]	ICCV2019	56.9	65.0	56.7	74.0	80.1	68.7	61.1	49.2	71.7
SiamRPN++[19]	CVPR2019	49.6	56.9	49.1	73.3	80.0	69.4	51.7	32.5	61.6
ECO[8]	ICCV2017	32.4	33.8	30.1	55.4	61.8	49.2	31.6	11.1	30.9
MDNet[26]	CVPR2016	39.7	46.0	37.3	60.6	70.5	56.5	29.9	9.9	30.3
SiamFC[1]	ECCVW2016	33.6	42.0	33.9	57.1	66.3	55.3	34.8	9.8	35.3

Table 1: State-of-the-art comparison on LaSOT, TrackingNet, and GOT-10K. The best two results are shown in red and blue, respectively.

	STM [27]	SiamMasK [34]	Ocean [40]	D3S [23]	AlphaRef [17]	STARK-ST50 +AR[37]	**DeTrack +AR**	**DeTrack-T +AR**
EAO (↑)	0.308	0.321	0.430	0.439	0.482	0.505	0.473	0.512
Accurary (↑)	0.751	0.624	0.693	0.699	0.754	0.759	0.760	0.763
Robustness (↑)	0.574	0.648	0.754	0.769	0.777	0.817	0.763	0.825

Table 2: Comparison of tracking results on VOT2020[17]. The two best results are marked in red and blue font.

the protocol using the same testing data. This benchmark evaluates the performance of the tracker through three indicators of success plots, average overlap (AO) and success rate(SR), where SR refers to the accuracy of successful tracking under a certain AO threshold, and two thresholds of 0.5 and 0.75 are taken. Following the official protocol training on the GOT-10K train split dataset and evaluating the GOT-10K test split dataset. As shown in Table 1 and Figure 1, the proposed algorithm is compared with SiamFC, SiamRPN++ and the recent Transformer-based trackers, such as TrDiMP and STARK-ST50. The proposed DeTrack-T outperforms the excellent STARK-ST50 on GOT-10K by more than 1%.

Results on VOT2020 Benchmark: VOT2020[17] contains 60 video sequences with challenges such as occlusion, fast movement, and large-scale deformation. The proposed DeTrack also applies AlphaRef on the basis of updating templates to generate segmentation masks. As Table 2 shows, DeTrack-T achieved 0.512 on EAO, better than the previous STARK-ST50+AR.

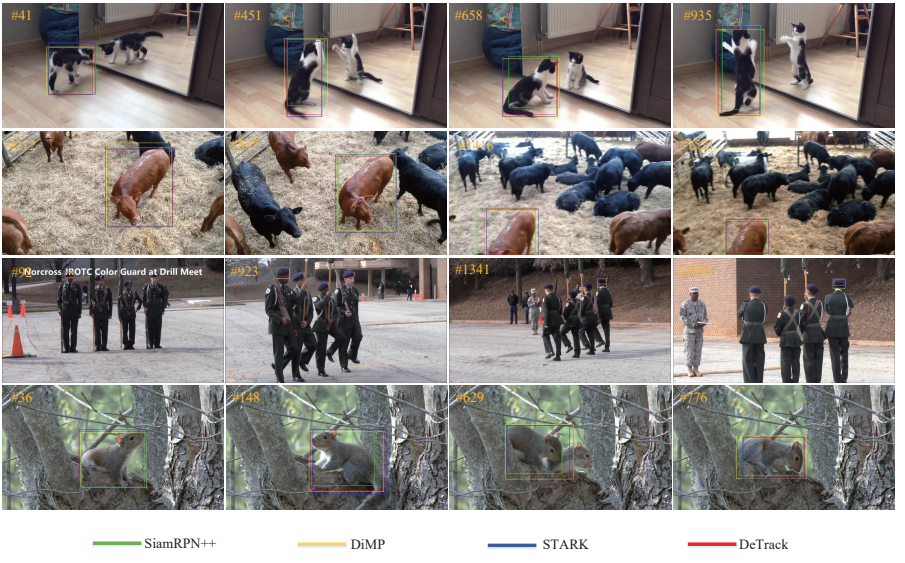

Figure 5: Comparison of visualization results of representative trackers on LaSOT.

4.3 Ablation study

This section provides experimental data to support the ideas presented in this chapter and designs experiments to verify the effectiveness of the target tracking algorithm for the deformable attention mechanism proposed in this paper.

We believe that the feature that distinguishes the target tracking task from other computer vision tasks is the feature interaction. The efficiency of the information exchange between the template frame and the search image frame is the key factor affecting the performance, while the overlay of encoder and decoder in STARK is to enhance the interaction features. There is no essential difference, we believe that this approach is too redundant.

As can be seen from Table 3, the performance of the tracker does not change significantly after we remove the decoder. At the same time, the parameters have been reduced by 4M, and the speed has been improved by 7 fps. After applying deformable attention to the feature interaction, the AUC and speed of the tracker are significantly improved. This is because with the addition of the deformable encoder module, the model focuses on the target information and gradually ignores the background. At the same time, the background information should not be completely ignored. Therefore, the combination of the two encoding methods gives better results in the experimental results.

Methods	AUC(%)	Params	Speed(fps)
STARK-S50	65.9	23M	50
Remove decoder	65.6	19M	57
DeTrack	**66.7**	**17M**	**59**

Table 3: Speed and parameter comparison on LaSOT.

Based on the above experiments, we tried to find novel feature interaction methods. As mentioned above, the self-attention mechanism-based encoder considers every pixel in the image, which also adds a significant computational burden to the model, whereas the target tracking task only requires the perception of the maximum response position of the target. In summary, we first use the deformable attention-based encoder for feature interaction, and

then access the self-attentive encoder. The deformable attention-based encoder is able to track the target location more accurately without focusing on all pixels, saving computational effort. And then the self attention encoder module is used to further enhance the target pixels sensed by the deformable encoder. Experiments were designed and validated for the combination of encoder modules, as shown in Table 4. The results show that the combination of deformable attention-based encoders and self-attention encoders for feature interaction improves the accuracy of the model by the combination of two deformable attention encoders and four self attention encoders.

Number		GOT-10K
Deformable Attention	Self Attention	AO
0	6	67.0
1	5	67.8
2	**4**	**68.1**
3	3	67.5
6	0	67.2

Table 4: Comparison of the results of the combination of encoder modules on GOT-10K.

5 Conclusion

This paper explores novel ways of interacting between template frame information and search frame information in terms of the characteristics of target tracking tasks. A combination of a deformable encoder based on a deformable attention mechanism and an encoder based on a self-attentive mechanism is used for feature interaction. The deformable encoder has a strong perceptual capability to more accurately track the target location without focusing on all the pixels. The self-attention encoder is used to further enhance the target pixels perceived by the deformable encoder and ultimately predict the target location directly, discarding the post-processing step in traditional target tracking algorithms. Finally, the analysis is validated and given on the datasets LaSOT, TrackingNet, GOT-10K and VOT2020.

Acknowledgements This work is supported by National Natural Science Foundation of China (No.61876092,U2013210), CAAI-Huawei MindSpore Open Fund(CAAIXSJLJJ-2021-003B).

*Corresponding author: Baojie Fan (email: jobfbj@gmail.com).

References

[1] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 850–865. Springer, 2016.

[2] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. *Proceedings of the IEEE International Conference on Computer Vision*, pages 6182–6191, 2019.

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Know your surroundings: Exploiting scene information for object tracking. In *European Conference on Computer Vision*, pages 205–221. Springer, 2020.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[5] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8126–8135, 2021.

[6] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021.

[7] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Fully convolutional online tracking. In *arXiv preprint arXiv:2004.07109*, 2020.

[8] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6638–6646, 2017.

[9] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2019.

[10] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7183–7192, 2020.

[11] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7183–7192, 2020.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[14] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin 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*, pages 5374–5383, 2019.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1562–1577, 2019.

[17] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, Ondrej Drbohlav, et al. The eighth visual object tracking vot2020 challenge results. In *European Conference on Computer Vision*, pages 547–601. Springer, 2020.

[18] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018.

[19] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019.

[20] Liting Lin, Heng Fan, Yong Xu, and Haibin Ling. Swintrack: A simple and strong baseline for transformer tracking. *arXiv preprint arXiv:2112.00995*, 2021.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[23] Alan Lukezic, Jiri Matas, and Matej Kristan. D3s-a discriminative single shot segmentation tracker. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7133–7142, 2020.

[24] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc 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*, pages 13444–13454, 2021.

[25] Matthias Muller, Adel Bibi, Silvio Giancola, Salman Alsubaihi, and Bernard Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 300–317, 2018.

[26] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4293–4302, 2016.

[27] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.

[28] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[29] Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5296–5305, 2017.

[30] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115 (3):211–252, 2015.

[31] Jack Valmadre, Luca Bertinetto, Joao Henriques, Andrea Vedaldi, and Philip HS Torr. End-to-end representation learning for correlation filter based tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2805–2813, 2017.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[33] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1571–1580, 2021.

[34] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 1328–1338, 2019.

[35] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021.

[36] Yinda Xu, Zeyu Wang, Zuoxin Li, Ye Yuan, and Gang Yu. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12549–12556, 2020.

[37] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10448–10457, 2021.

[38] Bin Yu, Ming Tang, Linyu Zheng, Guibo Zhu, Jinqiao Wang, Hao Feng, Xuetao Feng, and Hanqing Lu. High-performance discriminative tracking with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9856–9865, 2021.

[39] Yuechen Yu, Yilei Xiong, Weilin Huang, and Matthew R Scott. Deformable siamese attention networks for visual object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6728–6737, 2020.

[40] Zhipeng Zhang, Houwen Peng, Jianlong Fu, Bing Li, and Weiming Hu. Ocean: Object-aware anchor-free tracking. In *European Conference on Computer Vision*, pages 771–787. Springer, 2020.

[41] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2020.