ODTrack: Online Dense Temporal Token Learning for Visual Tracking

Yaozong Zheng[1,2], Bineng Zhong[1,2*], Qihua Liang[1,2], Zhiyi Mo[3], Shengping Zhang[4], Xianxian Li[1,2]

[1]Key Laboratory of Education Blockchain and Intelligent Technology, Ministry of Education, Guangxi Normal University
[2]Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University
[3]Guangxi Key Laboratory of Machine Vision and Intelligent Control, Wuzhou University
[4]Harbin Institute of Technology
yaozongzheng@stu.gxnu.edu.cn, bnzhong@gxnu.edu.cn, qhliang@gxnu.edu.cn
zhiyim@gxuwz.edu.cn, s.zhang@hit.edu.cn, lixx@gxnu.edu.cn

Abstract

Online contextual reasoning and association across consecutive video frames are critical to perceive instances in visual tracking. However, most current top-performing trackers persistently lean on sparse temporal relationships between reference and search frames via an offline mode. Consequently, they can only interact independently within each image-pair and establish limited temporal correlations. To alleviate the above problem, we propose a simple, flexible and effective video-level tracking pipeline, named **ODTrack**, which densely associates the contextual relationships of video frames in an online token propagation manner. ODTrack receives video frames of arbitrary length to capture the spatio-temporal trajectory relationships of an instance, and compresses the discrimination features (localization information) of a target into a token sequence to achieve frame-to-frame association. This new solution brings the following benefits: 1) the purified token sequences can serve as prompts for the inference in the next video frame, whereby past information is leveraged to guide future inference; 2) the complex online update strategies are effectively avoided by the iterative propagation of token sequences, and thus we can achieve more efficient model representation and computation. ODTrack achieves a new *SOTA* performance on seven benchmarks, while running at real-time speed. Code and models are available at https://github.com/GXNU-ZhongLab/ODTrack.
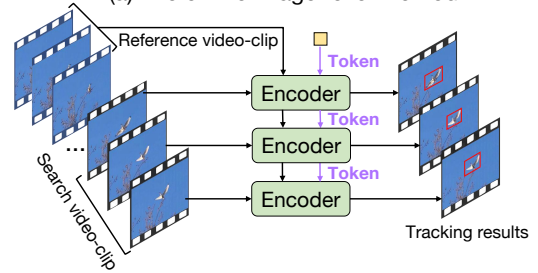
Introduction

Visual tracking aims to uniquely identify and track an object within a video sequence by using arbitrary target queries. In the visual world, objects rarely exist in isolation but rather within a larger and dynamic context. Therefore, visual perception is a complex process that involves interpreting and understanding the surrounding environment of an object. In such a situation, equipping a model with the ability to perform online contextual reasoning and establish associations presents a challenge in the field of visual tracking.

Despite this challenge, a significant number of current tracking methods overlook this problem and instead rely on the offline image-pair matching to localize instances in the current frame. As shown in Fig.1(a), these offline methods(Bertinetto et al. 2016; Li et al. 2019; Chen et al. 2021;
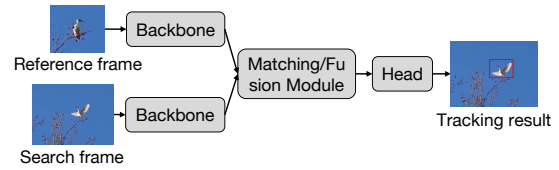
Figure 1: Comparison of tracking methods. (a) The offline image level tracking methods(Li et al. 2019; Chen et al. 2021) based on sparse sampling and image-pair matching. (b) Our online video-level tracking method based on video sequence sampling and temporal token propagation.

Yan et al. 2021a; Ye et al. 2022; Cui et al. 2022) typically follow a three-phase process: (*i*) extracting features by sampling two video frames (i.e., reference and search frames); (*ii*) propagating the initial target information from the reference to the search frame through a matching/fusion module; and (*iii*) utilizing a bounding box prediction head to output the localization results. Most trackers have performed well under this paradigm, but still exhibit the following drawbacks: (1) *The sampling frames are sparse* (i.e., using only one reference frame and one search frame). Although visual tracking inherently contains rich temporal data, this simple sampling strategy falls short in accurately representing the motion state of an object, posing a significant challenge for trackers to comprehend dynamic video content, and (2) *The target information is matched offline and limited to image-pair level*, preventing the association of the targets across video frames. Traditional feature matching/fusion methods(Chen et al. 2020; Zhang et al. 2020; Guo et al. 2021; Xie et al. 2022) focus on the appearance similar-

ity of object, without considering the property that tracking instance rely on continuous cross-frame associations.

To incorporate temporal information into the model, some approaches commonly design online updating techniques, such as updating templates(Yan et al. 2021a; Cui et al. 2022) and updating model parameters(Bhat et al. 2019). Despite being successful, these methods still rely on sparse sampling frames (i.e., reference, search, and update frames) and do not effectively explore how information is propagated online across search frames. This inspired us to think: *can our visual tracking algorithm densely associate and perceive an object in a video streaming context?*

The answer is affirmative. Unlike conventional approaches that rely on offline image-pair matching with sparse sampling frames, this paper proposes **ODTrack**, a novel video-level framework for visual tracking that capitalizes on video stream modeling. Specifically, we reformulate object tracking as a token sequence propagation task that densely associates the contextual relationships of across video frames in an auto-regressive manner, as shown in Fig.1(b). To overcome the limitations of traditional image-pair sampling strategy and explore the rich temporal dependencies, we extend the model's input from image-pair to the level of a video stream. Under this new input paradigm, we design two simple yet effective temporal token propagation attention mechanism that captures the spatio-temporal trajectory relationships of the target instance using an online token propagation manner, thus allowing the processing of video-level inputs of arbitrary length. Notably, we treat each video sequence as a continuous sentence, enabling us to employ language modeling for a comprehensive contextual understanding of the video content. This novel approach significantly distinguishes our tracker from traditional methods (Yan et al. 2021a; Ye et al. 2022; Cui et al. 2022) and greatly strengthens its ability to understand the spatio-temporal trajectory of target instance.

The main contributions of this work are as follows.

- We propose a novel video-level tracking pipeline, called ODTrack. In contrast to existing tracking approaches based on sparse temporal modeling, we employ a token sequence propagation paradigm to densely associate contextual relationships across video frames.

- We introduce two temporal token propagation attention mechanisms that compress the discriminative features of the target into a token sequence. This token sequence serves as a prompt to guide the inference of future frames, thus avoiding complex online update strategies.

- Our approach achieves a new state-of-the-art tracking results on seven visual tracking benchmarks, including LaSOT, TrackingNet, GOT10K, LaSOT$_{ext}$, VOT2020, TNL2K, and OTB100.

Related Work

Traditional Tracking Framework. The current popular trackers(Bertinetto et al. 2016; Li et al. 2019; Chen et al. 2021; Ye et al. 2022) are dominated by the Siamese tracking paradigm, which achieves tracking by image-pair matching.

To improve the accuracy and robustness of trackers, several different approaches are proposed, such as prediction head networks (Li et al. 2018; Chen et al. 2020; Zhang et al. 2020), cross-correlation modules (Han et al. 2021; Liao et al. 2020; Chen et al. 2021), powerful backbone (Chen et al. 2022; Cui et al. 2022) and attention mechanisms (Guo et al. 2021; Yu et al. 2020). In recent years, the introduction of the transformer (Vaswani et al. 2017) enables trackers (Yan et al. 2021a; Xie et al. 2022; Cui et al. 2022; Ye et al. 2022) to explore more powerful and deeper feature interactions, resulting in significant advances in tracking algorithm development. However, most of these methods are designed based on offline mode and sparse image-pair strategy. With this design paradigm, the tracker struggles to accurately comprehend the object's motion state in the temporal dimension and can only resort to traditional Siamese similarity for appearance modeling. In contrast to these approaches, we reformulate object tracking as a token sequence propagation task and aim to extend Siamese tracker to efficiently exploit target temporal information in an auto-regressive manner.

Temporal Modelling in Visual Tracking. Multi-object tracking algorithms(Meinhardt et al. 2022; Zeng et al. 2022) typically involve the recognition and association of individual objects in a video, making the study of trajectory information a common practice. However, there exists a relatively limited amount of research exploring the utilization of spatio-temporal trajectory information in single-object tracking algorithms.

To explore temporal cues within the Siamese framework, several online update methods are carefully designed. UpdateNet(Zhang et al. 2019) introduces an adaptive updating strategy, which utilizes a custom network to fuse accumulated templates and generate a weighted updated template feature for visual tracking. DCF-based trackers(Danelljan et al. 2019; Bhat et al. 2019; Danelljan, Gool, and Timofte 2020) excel at updating model parameters online using sophisticated optimization techniques, thereby improving the robustness of the tracker. STMTrack(Fu et al. 2021) and TrDiMP(Wang et al. 2021a) employ attention mechanism to effectively extract contextual information along the temporal dimension. STARK(Yan et al. 2021a) and Mixformer(Cui et al. 2022) specifically design target quality branch for updating template frame, which aids in improving the tracking results. Recently, there has been a gradual surge in research attention towards modeling temporal context from various perspectives. TCTrack (Cao et al. 2022) introduces an online temporal adaptive convolution and an adaptive temporal transformer that aggregates temporal contexts at two levels containing feature extraction and similarity map refinement. VideoTrack (Xie et al. 2023) designs a new tracker based on video transformer and uses a simple feedforward network to encode temporal dependencies. ARTrack (Xing et al. 2023) presents a new time-autoregressive tracker that estimates the coordinate sequence of an object progressively.

Nevertheless, the above tracking algorithms still suffer from the following limitations: (1) The optimization process is complex, involving the design of specialized loss functions(Bhat et al. 2019), multi-stage training strategies(Yan
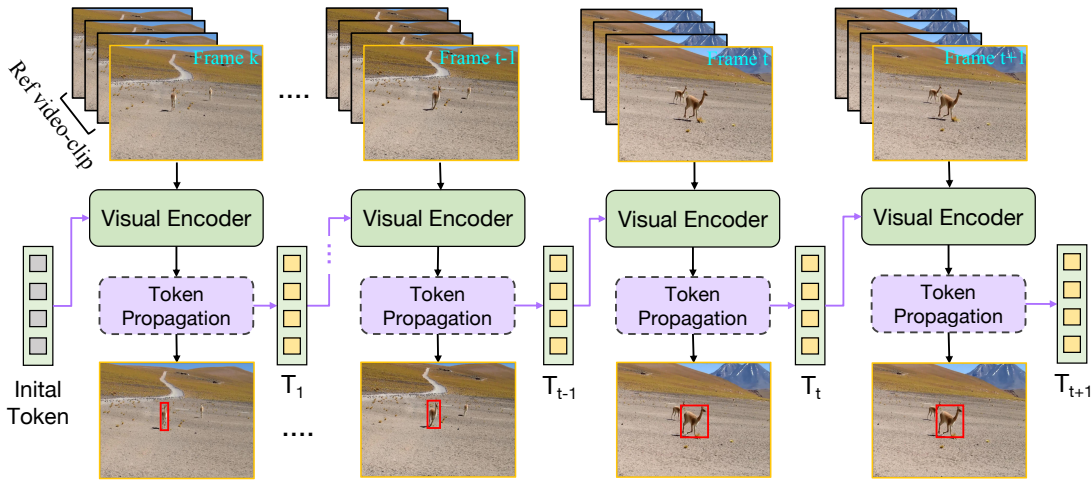
Figure 2: ODTrack Framework Architecture. The ODTrack pipeline takes video clips, consisting of reference and search frames, of arbitrary length as input. Then, the model utilizes a temporal token propagation attention mechanism to generate a temporal token for each video frame. These temporal tokens are subsequently propagated to the following frames in an auto-regressive manner, enabling cross-frame propagation of target trajectory information.

et al. 2021a), and manual update rules(Yan et al. 2021a), and (2) Although they explore temporal information to some extent, they fail to investigate *how temporal cues propagate across search frames*. In this work, we introduce a new dense context propagation mechanism from a token propagation perspective, which offers a solution to circumvent intricate optimization processes and training strategies. Further, we propose a new baseline approach, called ODTrack, focused on unlocking the potential of temporal modeling through the propagation of target motion/trajectory information.

Approach

We introduce ODTrack, a new video-level framework that employs token sequence propagation for visual tracking, as shown in Fig.2. This section first describes the concept of video-level visual object tracking, followed by the introduction of temporal token propagation attention mechanism and how they are trained in a new design paradigm.

Question Formulation

To provide a comprehensive understanding of our ODTrack framework, it is pertinent to first offer a review of previously prominent image-pair matching tracking methodologies(Bertinetto et al. 2016; Chen et al. 2021; Ye et al. 2022).

Given a pair of video frames, i.e., a reference frame $R \in \mathbb{R}^{3 \times H_r \times W_r}$ and a search frame $S \in \mathbb{R}^{3 \times H_s \times W_s}$, the mainstream visual trackers Ψ are formulated as $B \leftarrow \Psi : \{R, S\}$, where B denotes the predicted box coordinates of the current search frame. If Ψ is a conventional convolutional siamese tracker(Li et al. 2019; Chen et al. 2020, 2021), it undergoes three stages, namely feature extraction, feature fusion, and bounding box prediction. Whereas if Ψ is a transformer tracker(Ye et al. 2022; Cui et al. 2022; Chen et al. 2022), it consists solely of a backbone and a prediction head network, where the backbone integrates the processes of feature extraction and fusion.

Specifically, the transformer tracker receives a series of non-overlapping image patches (the resolution of each image patch is $p \times p$) as input. This means that a 2D reference-search image pair needs to pass through a patch embedding layer to generate multiple 1D image token sequences $\{f_r \in \mathbb{R}^{D \times N_r}, f_s \in \mathbb{R}^{D \times N_s}\}$, where D is the token dimension, $N_r = H_r W_r / p^2$, and $N_s = H_s W_s / p^2$. These 1D image tokens are then concatenated and loaded into a L-layer transformer encoder for feature extraction and relationship modeling. Each transformer layer δ contains a multi-head attention and a multi-layer perceptron. Here, we formulate the forward process of the lth transformer layer as follows:

$$f_{rs}^l = \delta^l(f_{rs}^{l-1}), l = 1, 2, ..., L \qquad (1)$$

where f_{rs}^{l-1} denotes the concatenated token sequence of the reference-search image pair generated from the $(l-1)^{th}$ transformer layer, and f_{rs}^l represents the token sequence generated by the current l^{th} transformer layer.

By adopting the modeling approach mentioned above, we can construct a concise and elegant tracker to achieve per-frame tracking. Nevertheless, this modeling approach has a clear drawback. The created tracker solely focuses on intra-frame target matching and lacks the ability to establish inter-frame associations necessary for tracking object across a video stream. Consequently, this limitation hinders the research of video-level tracking algorithms.

In this work, we aim to alleviate this challenge and propose a new design paradigm for video-level tracking algorithms. First, we extend the inputs of the tracking framework from the image-pair level to the video level for temporal modeling. Then, we introduce a new temporal token/prompt T designed to propagate information about the appearance, spatio-temporal location and trajectory of the target instance in a video sequence. Formally, we formulate video-level

tracking as follows:

$$B \leftarrow \Psi : \{R_1, R_2, ..., R_k, S_1, S_2, ..., S_n, T\} \qquad (2)$$

where $\{R_1, R_2, ..., R_k\}$ denotes the reference frames of length k, and $\{S_1, S_2, ..., S_n\}$ represents the search frames of length n. Our video-level tracking framework receives video clip of arbitrary length to model spatio-temporal trajectory relationships of the target object. We describe the proposed core module in more detail in the next section.

Video-Level Tracking Pipeline

Fig.2 gives an overview of our ODTrack framework. In this section, our focus lies in constructing a video-level tracking pipeline. Theoretically, we model the entire video as a continuous sequence, and decode the localization of target frame by frame in an auto-regressive manner. Firstly, we present a novel video sequence sampling strategy designed specifically to meet the input requirements of the video-level model. Subsequently, to capture the spatio-temporal trajectory information of the target instance within the video sequences, we introduce two simple yet effective temporal token propagation attention mechanisms.

Video Sequence Sampling Strategy   Most existing trackers (Yan et al. 2021a; Cui et al. 2022; Ye et al. 2022) commonly sample image-pairs within a short-term interval, such as 50, 100, or 200 frame intervals. However, this sampling approach poses a potential limitation as these trackers fail to capture the long-term motion variations of the tracked object, thereby constraining the robustness of tracking algorithms in long-term scenarios.

To obtain richer spatio-temporal trajectory information of the target instance from long-term video sequences, we deviate from the traditional short-term image-pair sampling method and propose a new video sequence sampling strategy. Specifically, we establish a larger sampling interval and randomly extract multiple video frames within this interval to form video clips $\{R_1, R_2, ..., R_k, S_1, S_2, ..., S_n\}$ of any lengths. Although this sampling approach may seem simplistic, it enables us to approximate the content of the entire video sequence. This is crucial for video-level modeling.

Temporal Token Propagation Attention Mechanism   Instead of employing a complex video transformer (Xie et al. 2023) as the foundational framework for encoding video content, we approach the design from a new perspective by utilizing a simple 2D transformer architecture, i.e., 2D ViT (Dosovitskiy et al. 2021).

To construct an elegant instance-level inter-frame correlation mechanism, it is imperative to extend the original 2D attention operations to extract and integrate video-level features. In our approach, we design two temporal token attention mechanisms based on the concept of *compression-propagation*, namely *concatenated token attention mechanism* and *separated token attention mechanism*, as shown in Fig.3(left). The core design involves injecting additional information into the attention operations, such as more video sequence content and temporal token vectors, enabling them to extract richer spatio-temporal trajectory information of the target instance.
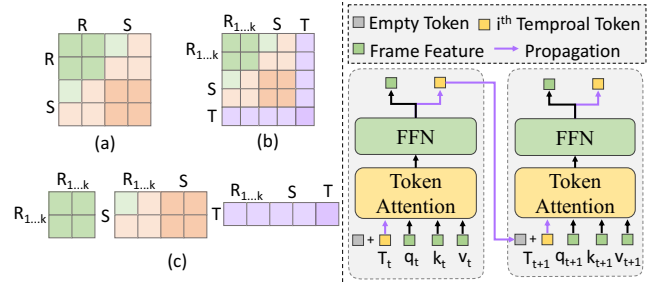

Figure 3: Left: the architecture of temporal token propagation attention mechanism. Right: illustration of online token propagation. (a) Original reference-search attention mechanism, (b) and (c) Different variants of the proposed temporal token propagation attention mechanisms. R is a single reference frame, $R_{1...k}$ denotes the reference frames of length k, S represents the current search frame, and T is the temporal token sequence of current video frames.

In Fig.3(a), the original attention operation commonly employs an image pair as inputs, where the process of modeling their relationships can be represented as $f = \text{Attn}([R, S])$. In this paradigm, the tracker can only engage in independent interactions within each image pair, establishing limited temporal correlations. In Fig.3(b), the proposed concatenated token attention mechanism extends the input to the aforementioned video sequence, enabling dense modeling of spatio-temporal relationships across frames. Inspired by the contextual nature of language formed through concatenation, we apply the concatenation operation to establish context for video sequences as well. Its formula can be represented as:

$$
\begin{aligned}
f_t &= \text{Attn}([R_1, R_2, ..., R_k, S_t, T_t]) \\
&= \sum_{s''t''} V_{s''t''} \cdot \frac{\exp\langle q_{st}, k_{s''t''} \rangle}{\sum_{s't'} \exp\langle q_{st}, k_{s't'} \rangle}
\end{aligned} \qquad (3)
$$

Where T_t is the temporal token sequence of tth video frame. $[\cdots, \cdots]$ denotes concatenation among tokens. q_{st}, k_{st} and v_{st} are spatio-temporal linear projections of the concatenated feature tokens.

It is worth noting that we introduce a temporal token for each video frame, with the aim of storing the target trajectory information of the sampled video sequence. In other words, we *compress* the current spatio-temporal trajectory information of the target into a token vector, which is used to *propagate* to the subsequent video frames.

Once the target information is extracted by the temporal token, we propagate the token vector from t^{th} frame to $(t+1)^{th}$ frame in an auto-regressive manner, as shown in Fig.3(right). Firstly, the tth temporal token T_t is added to the $(t+1)^{th}$ empty token T_{empty}, resulting in an update of the content token T_{t+1} for $(t+1)^{th}$ frame, which is then propagated as input to the subsequent frames. Formally, the propagation process is:

$$
\begin{aligned}
T_{t+1} &= T_t + T_{empty} \\
f_{t+1} &= \text{Attn}([R_1, R_2, ..., R_k, S_{t+1}, T_{t+1}])
\end{aligned} \qquad (4)
$$

Method	Resolution	Params	FLOPs	Speed	Device
SeqTrack-B	384×384	89M	148G	$11 fps$	2080Ti
ODTrack-B	384×384	92M	73G	$32 fps$	2080Ti

Table 1: Comparison of model parameters, FLOPs, and inference speed.

In this new design paradigm, we can employ temporal tokens as prompts for inferring the next frame, leveraging past information to guide future inference. Moreover, our model implicitly propagates appearance, localization, and trajectory information of the target instance through online token propagation. This significantly improves tracking performance of the video-level framework.

On the other hand, as illustrated in Fig.3(c), the proposed separated token attention mechanism decomposes attention operation into three sub-processes: self-information aggregation between reference frames, cross-information aggregation between reference and search frames, and cross-information aggregation between temporal token and video sequences. This decomposition improves the computational efficiency of the model to a certain extent, while the token propagation aligns with the aforementioned procedures.

Discussions with Online Update. Most previous tracking algorithms combine online updating methods to train a spatio-temporal tracking model, such as adding an extra score quality branch(Yan et al. 2021a) or an IoU prediction branch(Danelljan et al. 2019). They typically require complex optimization processes and update decision rules. In contrast to these methods, we avoid complex online update strategies by utilizing online iterative propagation of token sequences, enabling us to achieve more efficient model representation and computation.

Prediction Head and Loss Function

For the design of the prediction head network, we employ conventional classification head and bounding box regression head to achieve the desired outcome. The classification score map $\mathbb{R}^{1 \times \frac{H_s}{p} \times \frac{W_s}{p}}$, bounding box size $\mathbb{R}^{2 \times \frac{H_s}{p} \times \frac{W_s}{p}}$, and offset size $\mathbb{R}^{2 \times \frac{H_s}{p} \times \frac{W_s}{p}}$ for the prediction are obtained through three sub-convolutional networks, respectively. We adopt the focal loss(Lin et al. 2017) as classification loss L_{cls}, and the L_1 loss and $GIoU$ loss(Rezatofighi et al. 2019) as regression loss. The total loss L can be formulated as:

$$L = L_{cls} + \lambda_1 L_1 + \lambda_2 L_{GIoU} \qquad (5)$$

where $\lambda_1 = 5$ and $\lambda_2 = 2$ are the regularization parameters. Since we use video segments for modeling, the task loss is computed independently for each video frame, and the final loss is averaged over the length of the search frames.

Experiments

Implementation Details

Training. We use ViT-Base (Dosovitskiy et al. 2021) model as the visual encoder, and its parameters are initialized with MAE(He et al. 2022) pre-training parameters. The training
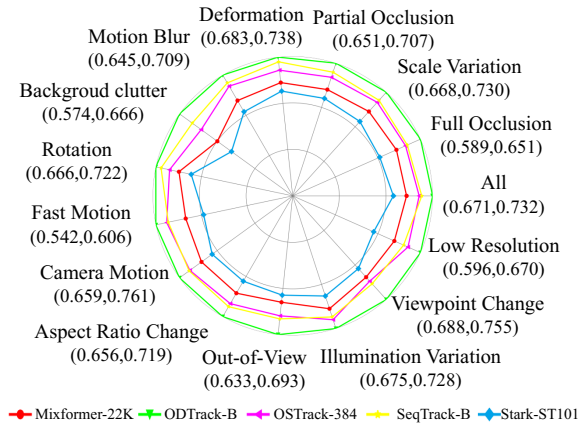

Figure 4: AUC scores of different attributes on LaSOT.

data includes LaSOT (Fan et al. 2019), GOT-10k (Huang, Zhao, and Huang 2021), TrackingNet (Müller et al. 2018), and COCO (Lin et al. 2014). In terms of input data, we take the video sequence including three reference frames with 192×192 pixels and two search frames with 384×384 pixels as the input to the model. We employ the AdamW to optimize the network parameters with initial learning rate of 1×10^{-5} for the backbone, 1×10^{-4} for the rest, and set the weight decay to 10^{-4}. We set the training epochs to 300 epochs. $60,000$ image pairs are randomly sampled in each epoch. The learning rate drops by a factor of 10 after 240 epochs. The model is conducted on a server with two 80GB Tesla A100 GPUs and set the batch size to be 8.

Inference. To align with the training setting, we incorporate three reference frames at equal intervals into our tracker during the inference phase. Concurrently, search frames and temporal token vectors are input frame-by-frame. Further, we conduct comparative experiments in model parameters, FLOPs and inference speed, as shown in Tab.1. The proposed ODTrack is tested on a 2080Ti, and it runs at 32 fps.

Comparison with the SOTA

GOT10K. GOT10K is a large-scale tracking dataset that contains more than 10,000 video sequences. The GOT10K benchmark proposes a protocol, which the trackers only use its training set for training. We follow the protocol to train our framework. As shown in Tab.2, the proposed method outperforms previous trackers and exhibits very competitive performance (77.0% AO) when compared to the previous best-performing tracker ARTrack (75.5% AO). These results demonstrate that one benefit of our ODTrack comes from the video-level sample strategy, which is design to release the potential of video-level modeling framework.

LaSOT. LaSOT is a large-scale long-term tracking benchmark that includes 1120 sequences for training and 280 sequences for testing. As shown in Tab.2, compared to most other tracking algorithms, our ODTrack-B achieves a new state-of-the-art result. For example, compared with the latest ARTrack, our method achieves 0.5%, 1.8%, and 1.5% gains in terms of AUC, P_{Norm} and P score, respectively. Furthermore, Fig.4 shows the results of attribute evaluation, demon-

Method	GOT10K*			LaSOT			TrackingNet			LaSOT_ext		
	AO	$SR_{0.5}$	$SR_{0.75}$	AUC	P_{Norm}	P	AUC	P_{Norm}	P	AUC	P_{Norm}	P
SiamFC (Bertinetto et al. 2016)	34.8	35.3	9.8	33.6	42.0	33.9	57.1	66.3	53.3	23.0	31.1	26.9
ATOM (Danelljan et al. 2019)	55.6	63.4	40.2	51.5	57.6	50.5	70.3	77.1	64.8	37.6	45.9	43.0
SiamPRN++ (Li et al. 2019)	51.7	61.6	32.5	49.6	56.9	49.1	73.3	80.0	69.4	34.0	41.6	39.6
DiMP (Bhat et al. 2019)	61.1	71.7	49.2	56.9	65.0	56.7	74.0	80.1	68.7	39.2	47.6	45.1
SiamRCNN (Voigtlaender et al. 2020)	64.9	72.8	59.7	64.8	72.2	-	81.2	85.4	80.0	-	-	-
Ocean (Zhang et al. 2020)	61.1	72.1	47.3	56.0	65.1	56.6	-	-	-	-	-	-
STMTrack (Fu et al. 2021)	64.2	73.7	57.5	60.6	69.3	63.3	80.3	85.1	76.7	-	-	-
TrDiMP (Wang et al. 2021a)	67.1	77.7	58.3	63.9	-	61.4	78.4	83.3	73.1	-	-	-
TransT (Chen et al. 2021)	67.1	76.8	60.9	64.9	73.8	69.0	81.4	86.7	80.3	-	-	-
Stark (Yan et al. 2021a)	68.8	78.1	64.1	67.1	77.0	-	82.0	86.9	-	-	-	-
SBT-B (Xie et al. 2022)	69.9	80.4	63.6	65.9	-	70.0	-	-	-	-	-	-
Mixformer (Cui et al. 2022)	70.7	80.0	67.8	69.2	78.7	74.7	83.1	88.1	81.6	-	-	-
TransInMo (Guo et al. 2022)	-	-	-	65.7	76.0	70.7	81.7	-	-	-	-	-
OSTrack (Ye et al. 2022)	73.7	83.2	70.8	71.1	81.1	77.6	83.9	88.5	83.2	50.5	61.3	57.6
AiATrack (Gao et al. 2022)	69.6	80.0	63.2	69.0	79.4	73.8	82.7	87.8	80.4	47.7	55.6	55.4
SeqTrack (Chen et al. 2023)	74.5	84.3	71.4	71.5	81.1	77.8	83.9	88.8	83.6	50.5	61.6	57.5
GRM (Gao, Zhou, and Zhang 2023)	73.4	82.9	70.4	69.9	79.3	75.8	84.0	88.7	83.3	-	-	-
VideoTrack (Xie et al. 2023)	72.9	81.9	69.8	70.2	-	76.4	83.8	88.7	83.1	-	-	-
ARTrack (Xing et al. 2023)	75.5	84.3	74.3	72.6	81.7	79.1	<u>85.1</u>	89.1	84.8	51.9	62.0	58.5
ODTrack-B	<u>77.0</u>	**87.9**	<u>75.1</u>	<u>73.2</u>	<u>83.2</u>	<u>80.6</u>	<u>85.1</u>	<u>90.1</u>	<u>84.9</u>	<u>52.4</u>	<u>63.9</u>	<u>60.1</u>
ODTrack-L	**78.2**	<u>87.2</u>	**77.3**	**74.0**	**84.2**	**82.3**	**86.1**	**91.0**	**86.7**	**53.9**	**65.4**	**61.7**

Table 2: Comparison with state-of-the-arts on four popular benchmarks: GOT10K, LaSOT, TrackingNet, and LaSOT_ext. Where * denotes for trackers only trained on GOT10K. Best in bold, second best underlined.

strating that our tracker outperforms other tracking methods on multiple challenge attributes. These results show that the token propagation mechanism helps the model to learn trajectory information about the target instance and greatly improves target localization in long-term tracking scenarios.

TrackingNet. TrackingNet is a large-scale short-term dataset that provides a test set with 511 video sequences. As reported in Tab.2, compared with the high-preformance tracker SeqTrack, our method achieves good tracking results that outperform 1.2%, 1.3%, and 1.3% in terms of success, normalized precision and precision score, respectively. This demonstrates that our ODTrack exhibits strong generalization capabilities.

LaSOT_ext. LaSOT_ext is the extended version of LaSOT, which comprises 150 long-term video sequences. As reported in Tab.2, our method achieves the good tracking results that outperform most compared trackers. For example, our tracker gets a AUC of 52.4%, P_{Norm} score of 63.9%, and P score of 60.1%, outperforming the ARTrack by 0.5%, 1.9%, and 1.6%, respectively. There results meet our expectation that video-level modeling has more stable object localization capabilities in complex scenarios.

VOT2020. VOT2020(Kristan, Leonardis, and et.al 2020) contains 60 challenging sequences, and it uses binary segmentation masks as the groundtruth. We use Alpha-Refine (Yan et al. 2021b) as a post-processing network for ODTrack to predict segmentation masks. As shown in Tab.3, our ODTrack-B and -L achieve the best results with EAO of 58.1% and 60.5% on mask evaluations, respectively.

TNL2K and OTB100. We evaluate our tracker on TNL2K(Wang et al. 2021b) and OTB100(Wu, Lim, and Yang 2015) benchmarks. They include 700 and 100 video sequences, respectively. These results in Tab.4 show that the

Method	EAO (↑)	Accuracy (↑)	Robustness (↑)
SiamMask	0.321	0.624	0.648
Ocean	0.430	0.693	0.754
D3S	0.439	0.699	0.769
SuperDiMP	0.305	0.492	0.745
AlphaRef	0.482	0.754	0.777
STARK	0.505	0.759	0.819
SBT	0.515	0.752	0.825
Mixformer	0.535	<u>0.761</u>	0.854
SeqTrack-B	0.522	-	-
ODTrack-B	<u>0.581</u>	**0.764**	<u>0.877</u>
ODTrack-L	**0.605**	<u>0.761</u>	**0.902**

Table 3: State-of-the-art comparison on VOT2020.

ODTrack-B and -L achieve the best performance on TNL2K and OTB100 benchmarks, demonstrating the effectiveness of the temporal token propagation attention mechanism.

Ablation Study

Importance of token propagation. To investigate the effect of token propagation in Eq.4, we perform experiments whether propagating temporal token in Tab.5(a). *w/o Token* denotes the experiment employing video-level sampling strategy without token propagation. From the second and third columns, it can be observed that the absence of the token propagation mechanism leads to a decrease in the AUC score by 1.2%. This result indicates that token propagation plays a crucial role in cross-frame target association.

Different token propagation methods. We conduct experiments to validate the effectiveness of two proposed token propagation methods in the video-level tracking frame-

	ATOM	Ocean	DiMP	TransT	OSTrack	Mixformer	SeqTrack	ARTrack	**ODTrack-B**	**ODTrack-L**
TNL2K	40.1	38.4	44.7	50.7	55.9	-	56.4	59.8	<u>60.9</u>	**61.7**
OTB100	66.3	68.4	68.4	69.6	-	70.0	-	-	<u>72.3</u>	**72.4**

Table 4: Comparison with state-of-the-art methods on TNL2K and OTB100 benchmarks in AUC score.

LaSOT	(a) Method				(b) Sequence Length				(c) Sample Rangle			
	Baseline	*w/o Token*	*Separate*	*Concatenation*	2	3	4	5	200	400	800	1200
AUC	70.1	71.0	72.2	**72.8**	72.8	**73.1**	72.5	72.0	72.8	**73.1**	73.0	73.0
P_{Norm}	80.2	81.1	82.3	**83.0**	83.0	**83.0**	82.9	82.1	83.0	**83.5**	83.3	83.1
P	76.9	78.0	79.2	**80.3**	80.3	**80.4**	79.9	79.3	80.3	**80.6**	80.4	80.1

Table 5: Ablation Studies of different token propagation designs on LaSOT benchmark.

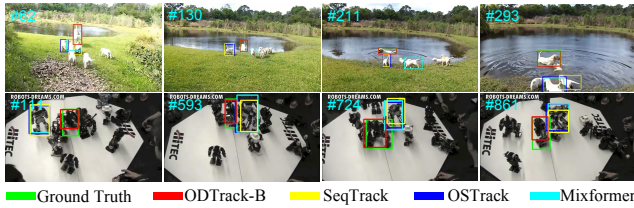Figure 5: Qualitative comparison results of our tracker with other three SOTA trackers on LaSOT benchmark.

Figure 6: The attention map of temporal token attention operation.

work in Tab.5(a). We can be observe that both the *separate* and *concatenation* methods achieve significant performance improvements, with the *concatenation* method showing slightly better results. This demonstrates the effectiveness of both attention mechanisms.

The length of search video-clip. As shown in Tab.5(b), we ablate the impact of search video sequence length on the tracking performance. When the length of video clip increases from 2 to 3, the AUC metric improves by 0.3%. However, continuous increment in sequence length does not lead to performance improvement, indicating that overly long search video clips impose a learning burden on the model. Hence, we should opt for an appropriate the length of search video clip.

The sampling range. To validate the impact of sampling range on algorithm performance, we conduct experiments on the sampling range of video frames in Tab.5(c). When the sampling range is expanded from 200 to 1200, there is a noticeable improvement in performance on the AUC metric, indicating that the video-level framework can learn target trajectory information from a larger sampling range.

Visualization and Limitation

Visualization. To intuitively show the effectiveness of the proposed method, especially in complex scenarios including similar distractors, we visualize the tracking results of our ODTrack and three advanced trackers on LaSOT dataset. As shown in Fig.5, due to its ability to densely propagate trajectory information of the target, our tracker far outperforms the latest tracker SeqTrack on these sequences.
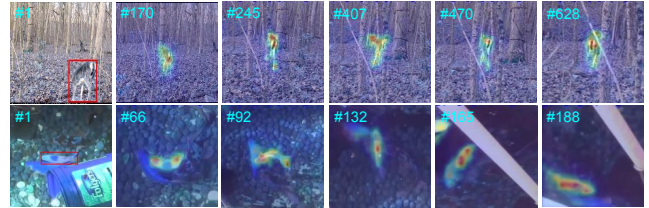
Furthermore, we visualize the attention map of temporal token attention operation, as shown in Fig.6. We can observe that the temporal token continuously propagate and attend to motion trajectory information of object, which aids our tracker in accurately localizing target instance.

Limitation. This work models the entire video as a sequence and decode the localization of instance frame by frame in an auto-regressive manner. Despite achieving remarkable results, our video-level modeling method is a global approximation due to constraints in GPU resources, and we are still unable to construct the framework in a cost-effective manner. A promising solution would involve improving the computational complexity and lightweight modeling of the transformer.

Conclusion

In this work, we present ODTrack, a new video-level framework for visual object tracking. We reformulate visual tracking as a token propagation task that densely associates the contextual relationships of across video frames in an auto-regressive manner. Furthermore, we propose a video sequence sampling strategy and two temporal token propagation attention mechanisms, enabling the proposed framework to simplify video-level spatio-temporal modeling and avoid intricate online update strategies. Extensive experiments show that our ODTrack achieves promising results on seven tracking benchmarks. We hope that this work inspires further research in video-level tracking modeling.

Acknowledgements

References

Bertinetto, L.; Valmadre, J.; Henriques, J. F.; Vedaldi, A.; and Torr, P. H. S. 2016. Fully-Convolutional Siamese Networks for Object Tracking. In *ECCV Workshops*, 850–865.

Bhat, G.; Danelljan, M.; Gool, L. V.; and Timofte, R. 2019. Learning Discriminative Model Prediction for Tracking. In *ICCV*, 6181–6190.

Cao, Z.; Huang, Z.; Pan, L.; Zhang, S.; Liu, Z.; and Fu, C. 2022. TCTrack: Temporal Contexts for Aerial Tracking. In *CVPR*, 14778–14788.

Chen, B.; Li, P.; Bai, L.; Qiao, L.; Shen, Q.; Li, B.; Gan, W.; Wu, W.; and Ouyang, W. 2022. Backbone is All Your Need: A Simplified Architecture for Visual Object Tracking. In *ECCV (22)*, 375–392.

Chen, X.; Peng, H.; Wang, D.; Lu, H.; and Hu, H. 2023. SeqTrack: Sequence to Sequence Learning for Visual Object Tracking. *CVPR*, abs/2304.14394.

Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; and Lu, H. 2021. Transformer Tracking. In *CVPR*, 8126–8135.

Chen, Z.; Zhong, B.; Li, G.; Zhang, S.; and Ji, R. 2020. Siamese Box Adaptive Network for Visual Tracking. In *CVPR*, 6667–6676.

Cui, Y.; Jiang, C.; Wang, L.; and Wu, G. 2022. MixFormer: End-to-End Tracking with Iterative Mixed Attention. In *CVPR*, 13598–13608.

Danelljan, M.; Bhat, G.; Khan, F. S.; and Felsberg, M. 2019. ATOM: Accurate Tracking by Overlap Maximization. In *CVPR*, 4660–4669.

Danelljan, M.; Gool, L. V.; and Timofte, R. 2020. Probabilistic Regression for Visual Tracking. In *CVPR*, 7181–7190.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; and Ling, H. 2019. LaSOT: A High-Quality Benchmark for Large-Scale Single Object Tracking. In *CVPR*, 5374–5383.

Fu, Z.; Liu, Q.; Fu, Z.; and Wang, Y. 2021. STMTrack: Template-Free Visual Tracking With Space-Time Memory Networks. In *CVPR*, 13774–13783.

Gao, S.; Zhou, C.; Ma, C.; Wang, X.; and Yuan, J. 2022. AiATrack: Attention in Attention for Transformer Visual Tracking. In *ECCV (22)*, 146–164.

Gao, S.; Zhou, C.; and Zhang, J. 2023. Generalized Relation Modeling for Transformer Tracking. *CVPR*, abs/2303.16580.

Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; and Shen, C. 2021. Graph Attention Tracking. In *CVPR*, 9543–9552.

Guo, M.; Zhang, Z.; Fan, H.; Jing, L.; Lyu, Y.; Li, B.; and Hu, W. 2022. Learning Target-aware Representation for Visual Tracking via Informative Interactions. In *IJCAI*, 927–934.

Han, W.; Dong, X.; Khan, F. S.; Shao, L.; and Shen, J. 2021. Learning To Fuse Asymmetric Feature Maps in Siamese Trackers. In *CVPR*, 16570–16580.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. B. 2022. Masked Autoencoders Are Scalable Vision Learners. In *CVPR*, 15979–15988.

Huang, L.; Zhao, X.; and Huang, K. 2021. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(5): 1562–1577.

Kristan, M.; Leonardis, A.; and et.al. 2020. The Eighth Visual Object Tracking VOT2020 Challenge Results. In *ECCV Workshops (5)*, volume 12539 of *Lecture Notes in Computer Science*, 547–601. Springer.

Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; and Yan, J. 2019. SiamRPN++: Evolution of Siamese Visual Tracking With Very Deep Networks. In *CVPR*, 4282–4291.

Li, B.; Yan, J.; Wu, W.; Zhu, Z.; and Hu, X. 2018. High Performance Visual Tracking With Siamese Region Proposal Network. In *CVPR*, 8971–8980.

Liao, B.; Wang, C.; Wang, Y.; Wang, Y.; and Yin, J. 2020. PG-Net: Pixel to Global Matching Network for Visual Tracking. In *ECCV*, 429–444.

Lin, T.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2017. Focal Loss for Dense Object Detection. In *ICCV*, 2999–3007.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. In *ECCV*, 740–755.

Meinhardt, T.; Kirillov, A.; Leal-Taixé, L.; and Feichtenhofer, C. 2022. TrackFormer: Multi-Object Tracking with Transformers. In *CVPR*, 8834–8844.

Müller, M.; Bibi, A.; Giancola, S.; Al-Subaihi, S.; and Ghanem, B. 2018. TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild. In *ECCV*, 310–327.

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I. D.; and Savarese, S. 2019. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In *CVPR*, 658–666.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *NIPS*, 5998–6008.

Voigtlaender, P.; Luiten, J.; Torr, P. H. S.; and Leibe, B. 2020. Siam R-CNN: Visual Tracking by Re-Detection. In *CVPR*, 6577–6587.

Wang, N.; Zhou, W.; Wang, J.; and Li, H. 2021a. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *CVPR*, 1571–1580.

Wang, X.; Shu, X.; Zhang, Z.; Jiang, B.; Wang, Y.; Tian, Y.; and Wu, F. 2021b. Towards More Flexible and Accurate Object Tracking With Natural Language: Algorithms and Benchmark. In *CVPR*, 13763–13773.

Wu, Y.; Lim, J.; and Yang, M. 2015. Object Tracking Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9): 1834–1848.

Xie, F.; Chu, L.; Li, J.; Lu, Y.; and Ma, C. 2023. VideoTrack: Learning to Track Objects via Video Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 22826–22835.

Xie, F.; Wang, C.; Wang, G.; Cao, Y.; Yang, W.; and Zeng, W. 2022. Correlation-Aware Deep Tracking. In *CVPR*, 8741–8750.

Xing, W.; Yifan, B.; Yongchao, Z.; Dahu, S.; and Yihong, G. 2023. Autoregressive Visual Tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9697–9706.

Yan, B.; Peng, H.; Fu, J.; Wang, D.; and Lu, H. 2021a. Learning Spatio-Temporal Transformer for Visual Tracking. In *ICCV*, 10428–10437.

Yan, B.; Zhang, X.; Wang, D.; Lu, H.; and Yang, X. 2021b. Alpha-Refine: Boosting Tracking Performance by Precise Bounding Box Estimation. In *CVPR*, 5289–5298. Computer Vision Foundation / IEEE.

Ye, B.; Chang, H.; Ma, B.; Shan, S.; and Chen, X. 2022. Joint Feature Learning and Relation Modeling for Tracking: A One-Stream Framework. In *ECCV (22)*, 341–357.

Yu, Y.; Xiong, Y.; Huang, W.; and Scott, M. R. 2020. Deformable Siamese Attention Networks for Visual Object Tracking. In *CVPR*, 6727–6736.

Zeng, F.; Dong, B.; Zhang, Y.; Wang, T.; Zhang, X.; and Wei, Y. 2022. MOTR: End-to-End Multiple-Object Tracking with Transformer. In *ECCV (27)*, 659–675.

Zhang, L.; Gonzalez-Garcia, A.; van de Weijer, J.; Danelljan, M.; and Khan, F. S. 2019. Learning the Model Update for Siamese Trackers. In *ICCV*, 4009–4018.

Zhang, Z.; Peng, H.; Fu, J.; Li, B.; and Hu, W. 2020. Ocean: Object-Aware Anchor-Free Tracking. In *ECCV*, 771–787.