A SIMPLIFIED DINOV3-DRIVEN DISCRIMINATIVE FRAMEWORK WITH SIAMESE TEMPLATE FUSION FOR VISUAL TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual object tracking requires representations that separate target from background with high precision while remaining cheap enough for real-time deployment. On UAV platforms, model size and latency constraints make this trade-off especially acute. Lightweight vision transformers achieve the required speed but their limited capacity weakens discriminative power, whereas self-supervised architectures such as DINOv3 provide rich semantic hierarchies that remain underexplored in tracking and, lacking an official tiny variant, cannot be deployed directly on constrained devices. We present DinoSDTrack, which reconciles these two regimes. First, we introduce a simplified, unified Siamese-discriminative framework in which the Siamese template prior is fused into a discriminative tracking head within a one-stream architecture, enabling target-background separation to be learned jointly rather than assembled from independent modules. Second, we show that DINOv3-Base fine-tuned on tracking objectives supplies task-adapted representations, and we formulate a task-adaptive distillation scheme that transfers these representations into a custom DINOv3-Tiny, preserving discriminative accuracy under strict real-time budgets. Experiments on six general tracking benchmarks and six UAV benchmarks show that DinoSDTrack attains state-of-the-art accuracy while maintaining real-time inference speed, indicating that task-adapted self-supervised hierarchies can be effectively compressed for discriminative tracking. Code will be released.

1 INTRODUCTION

Visual object tracking is the task of estimating the state of a target throughout a video given only its initial state in the first frame. Before one-stream methods became popular Ye et al. (2022); Cui et al. (2024), two paradigms dominated the field: Discriminative Correlation Filters (DCF) and Siamese tracking Javed et al. (2023). With the introduction of Transformers Waswani et al. (2017), discriminative trackers Yu et al. (2021); Wang et al. (2021a); Mayer et al. (2022) and Siamese trackers Yan et al. (2021a); Chen et al. (2021); Gao et al. (2022) both achieved superior performance. Their pipelines share feature learning, relation modeling, and a tracking head, and are categorized into two-stream and one-stream architectures depending on whether these stages are jointly conducted. Two-stream trackers typically rely on CNN or hierarchical Transformer backbones, whereas ViT Dosovitskiy et al. (2021) has become the preferred backbone of one-stream frameworks Ye et al. (2022); Chen et al. (2022a). The two paradigms embody complementary inductive biases: discriminative methods optimize target-specific predictors for target-background separation, while Siamese methods rely on stable template-search similarity matching. We refer to Appendix A.1 for a formal comparison of the two paradigms. Most one-stream trackers follow Siamese logic, jointly encoding template and search patches in a single Transformer. JDTrack He et al. (2025) recently demonstrated the strong capacity of one-stream discriminative pipelines, yet it is a derivative of ToMP Mayer et al. (2022) and cannot naturally inject the Siamese template prior into discriminative learning. How to unify stable template-guided matching with strong background resistance remains open, motivating our first objective: fusing discriminative supervision into the template-matching pipeline. Robustness is equally governed by appearance degradation: most trackers remain vulnerable to partial occlusion and distractors that pure 2D appearance matching cannot fully handle,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and recent work improves this by editing the model online with geometry-aware cues under null-space constraints Chen et al. (2026). Efficiency becomes most acute on UAV platforms, where strict model size and latency budgets meet motion blur, viewpoint changes, small objects, and severe occlusion Li et al. (2024); Wu et al. (2025); Zhang et al. (2026). Early DCF trackers such as KCF Henriques et al. (2015) ran at hundreds of FPS but relied on handcrafted features, while SiamFC Bertinetto et al. (2016) and lightweight CNN designs Yan et al. (2021b); Borsuk et al. (2022) introduced learning-based real-time tracking. Recent tiny ViT-based UAV trackers Li et al. (2023; 2024); Xue et al. (2025); Wu et al. (2025) follow the OSTrack paradigm Ye et al. (2022) and advance the performance-efficiency frontier through token pruning, adaptive layer activation, and occlusion-aware training Zhang et al. (2026). Nevertheless, structural simplification itself weakens discriminative capacity in complex scenarios Yang et al. (2026b). Teacher-guided distillation compensates for this loss without inference overhead Yang et al. (2026b), but existing schemes inherit two limitations: hand-crafted teacher-student layer assignments and naive base-to-tiny transfer, causing semantic mismatch and accuracy degradation in deeper layers Wang et al. (2026). Task-tailored pre-training Wu et al. (2023); Liang et al. (2025) is an alternative yet computationally expensive, and recent analyses warn that apparent gains often stem from increased inputs and parameters rather than from the mechanism itself Xue et al. (2026). Self-supervised architectures offer a third route: DINOv3 Siméoni et al. (2025) provides rich semantic hierarchies but remains underexplored in tracking and, lacking an official Tiny variant, cannot be deployed directly on constrained devices. This motivates our second objective: adapt the strong self-supervised representation to the tracking task, then compress it under real-time budgets through task-adaptive distillation. We propose **DinoSDTrack**, a lightweight real-time tracker that reconciles these two regimes; the overall pipeline is illustrated in Figure 1. First, we introduce a simplified, unified Siamese-discriminative framework in which the Siamese template prior is fused into the discriminative learning of a one-stream architecture, enabling target-background separation to be learned jointly rather than assembled from independent modules. A guided template-fusion module refines the template features and injects them into classification tokens without extra Transformer computation, and we combine the prediction heads of ToMP Mayer et al. (2022) and OSTrack Ye et al. (2022) to design an optimized branch. Second, we fine-tune DINOv3-Base on tracking objectives and formulate a task-adaptive distillation scheme that transfers the resulting representations into a custom DINOv3-Tiny. Unlike naive base-to-tiny layer matching Wang et al. (2026), our distillation operates on task-adapted representations, avoiding capacity mismatch in deep layers while preserving discriminative accuracy under strict real-time budgets. Extensive experiments on six general tracking benchmarks and six UAV benchmarks show that DinoSDTrack attains state-of-the-art accuracy while maintaining real-time inference speed, indicating that task-adapted self-supervised hierarchies can be effectively compressed for discriminative tracking. Our main contributions are summarized as follows: (1) We propose a simplified, unified Siamese-discriminative paradigm that fuses Siamese template matching with discriminative learning in a one-stream architecture, enhancing target-background separation without assembling the two paradigms from independent modules. (2) We present a task-adaptive DINOv3 distillation pipeline that transfers task-fine-tuned DINOv3-Base representations into a custom DINOv3-Tiny, enabling strong self-supervised semantics to be deployed on resource-constrained UAV platforms. (3) Extensive experiments on six general tracking benchmarks and six UAV benchmarks show that DinoSDTrack achieves state-of-the-art accuracy while maintaining high real-time inference speed, and we validate the design with ablations over lightweight ViT backbones and the distilled DINOv3-Tiny variant.

2 Related Work

Two-Stream Tracking Frameworks. Two-stream trackers decouple feature extraction from cross-branch relation modeling and are grouped into Siamese matching and discriminative correlation filter (DCF) trackers. Classic pipelines stack ResNet He et al. (2016) features with Transformer correlation modeling: TranT Chen et al. (2021) and STARK Yan et al. (2021a) pioneer Transformer-based fusion with static and dynamic templates, while AiA-Track Gao et al. (2022) and CSWinTT **?** strengthen cross-branch interaction. In the discriminative branch, DiMP Bhat et al. (2019) establishes the first end-to-end deep DCF architecture, TrDiMP Wang et al. (2021a) and DTT Yu et al. (2021) migrate Transformer blocks into DiMP, and ToMP Mayer et al. (2022) designs a Transformer-based target predictor that models training-test correlation. GOT-Edit Chen et al. (2026) further extends this line with online model editing under null-space constraints, absorbing geometry-aware cues while preserving semantic discrimination.

One-Stream Tracking Frameworks. One-stream trackers unify feature extraction and relation modeling within a single ViT Dosovitskiy et al. (2021) encoder and have become the mainstream paradigm Ye et al. (2022); Chen et al. (2022a); Cui et al. (2024). OSTrack Ye et al. (2022) contributes candidate early elimination and the widely used center-based prediction head. Subsequent works strengthen one-stream capacity through token partition and attention masking Gao et al. (2023), historical target cues Cai et al. (2024), learnable query tokens Zheng et al. (2024); Xu et al. (2025); Xie et al. (2024), Vision Mamba layers Kang et al. (2025); Xie et al. (2025), and video-level temporal propagation Zheng et al. (2024); Kang et al. (2025); Xie et al. (2024). JDTrack He et al. (2025) validates the strong capacity of one-stream discriminative tracking, yet as a derivative of ToMP Mayer et al. (2022) it cannot naturally inject the Siamese template prior into discriminative learning. MUTrack Wu et al. (2026) balances short-term adaptability and long-term identity with a unified memory bank, while MoDTrack Yang et al. (2026a) enhances motion modeling with a learnable motion dictionary.

Efficient and UAV Tracking. UAV tracking demands compact, low-latency trackers for edge platforms, yet most conventional frameworks are optimized for high-performance GPUs Lin et al. (2024); Cai et al. (2025). Traditional DCF trackers Henriques et al. (2015); Huang et al. (2019b); Li et al. (2020); Danelljan et al. (2017) achieve CPU real-time inference on handcrafted features; CNN-based Siamese designs Yan et al. (2021b); Borsuk et al. (2022); Zong et al. (2025) and hierarchical Transformer trackers Kang et al. (2023); Chen et al. (2022b); Cao et al. (2021) improve multi-level aerial modeling. Tiny ViT-based trackers Li et al. (2023; 2024); Xue et al. (2025); Wu et al. (2025) further achieve a promising performance-speed balance through token halting, adaptive layer activation, and occlusion-aware training, with OCTrack Zhang et al. (2026) modeling occlusion as a clustered spatial process. However, these methods uniformly adopt the Siamese template-search paradigm, and structural simplification itself weakens discriminative capacity in complex scenarios Yang et al. (2026b). **Knowledge Distillation and Temporal Learning.** Knowledge distillation is a standard route to compress trackers: EATrack Yang et al. (2026b) shows that teacher-guided dual-branch distillation compensates for the representation loss of lightweight students without inference overhead. Yet existing distillation schemes inherit two limitations—hand-crafted teacher-student layer assignments and naive base-to-tiny transfer—causing semantic mismatch and accuracy degradation in deeper layers Wang et al. (2026). Task-tailored pre-training Wu et al. (2023); Liang et al. (2025) is an alternative but computationally expensive, and LETrack Xue et al. (2026) warns that apparent gains from added temporal components often stem from increased inputs and parameters. DINOv3 Siméoni et al. (2025) provides rich self-supervised semantic hierarchies but remains under-explored in tracking and lacks an official Tiny variant for deployment. These observations motivate our task-adaptive distillation, which fine-tunes DINOv3-Base on tracking objectives and transfers the task-adapted representations into a custom DINOv3-Tiny.

3 Proposed Approach

This section presents DinoSDTrack, a one-stream discriminative tracker that unifies Siamese template matching and discriminative learning under a DINOv3 encoder. The framework takes three image inputs, injects discriminative priors through prompt encoding, fuses the Siamese template prior into the classification token via the Guided Spatial Weighted Token Fusion (GSWTF) module, and adopts an optimized tracking head. To meet real-time budgets, a task-adaptive distillation scheme transfers a task-fine-tuned DINOv3-Base teacher into a custom DINOv3-Tiny student.

3.1 One-Stream DINOv3 Discriminative Transformer Encoder

As illustrated in Figure 1, DinoSDTrack takes three inputs: a template image $z_t \in \mathbb{R}^{3\times128\times128}$ that encodes the initial target appearance, a training image $x_{tr} \in \mathbb{R}^{3\times256\times256}$ paired with its ground-truth bounding box, and a test image $x_{te} \in \mathbb{R}^{3\times256\times256}$ for online localization. The tracking task is to learn a mapping $f : (z_t, x_{tr}, x_{te}) \mapsto (S, b_t)$, where $S \in \mathbb{R}^{1\times16\times16}$ is the target score map and $b_t = [x, y, w, h]$ is the predicted bounding box. Unlike conventional two-stream discriminative trackers that decouple ResNet feature extraction from an independent Transformer predictor Mayer et al. (2022); Yu et al. (2021), DinoSDTrack uses DINOv3 Siméoni et al. (2025) as the unified one-stream backbone, so that feature extraction and discriminative target modeling are jointly optimized while the Siamese template prior is naturally integrated. A shared patch embedding layer projects the three images into visual tokens with DINOv3's native 16×16 patches and stride 16, producing

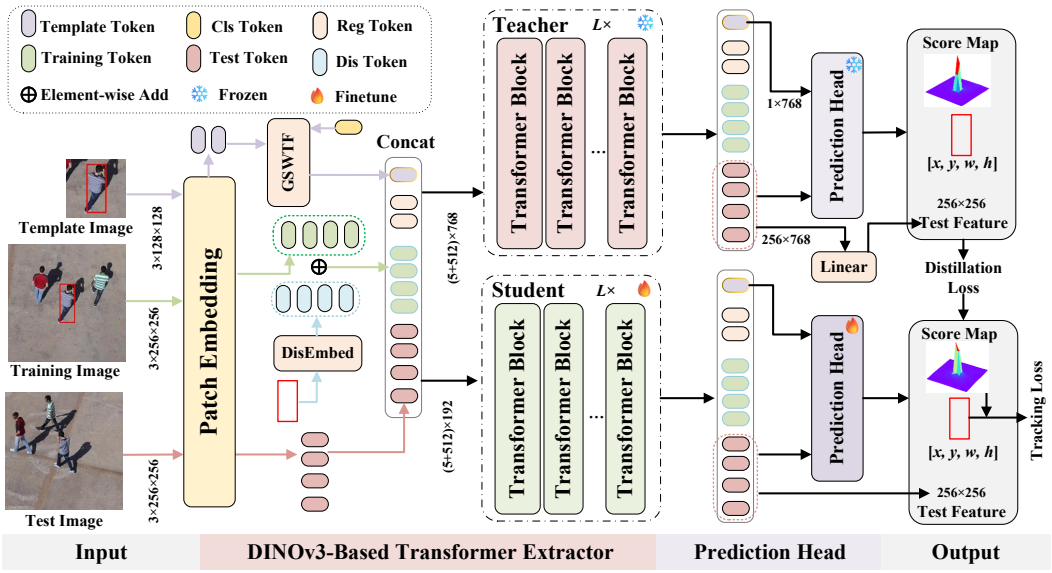


Figure 1: Framework of DinoSDTrack. A template frame, a training frame, and a test frame are jointly encoded by a one-stream DINOv3 Transformer. Discriminative prompt encoding injects target priors into the training tokens, and GSWTF refines the template tokens into the classification token. A frozen task-adapted DINOv3-Base teacher transfers its test-frame representations, score maps, and bounding boxes to the custom DINOv3-Tiny student through task-adaptive distillation. Best viewed in color.

64 template tokens, 256 training tokens, and 256 test tokens. The training tokens are augmented by the discriminative prompt encoding described in Section 3.2, and the classification token is refined by GSWTF in Section 3.3. The full token sequence is constructed as

$$e_{\text{tokens}} = \text{Concat}\big(e^{t}_{cls};\ e_{reg};\ e^{d}_{tr};\ e_{te}\big) \in \mathbb{R}^{(5+512)\times D}, \tag{1}$$

where e^{t}_{cls} is the template-refined classification token, $e_{reg} \in \mathbb{R}^{4\times D}$ are the DINOv3 register tokens, e^{d}_{tr} is the discriminative training token sequence, and e_{te} is the test token sequence. The template tokens are consumed by GSWTF and do not enter the sequence. The token sequence is processed by L stacked DINOv3 Transformer blocks with Rotary Position Embedding (RoPE) e_{pos}, which encodes relative positions without relying on DINOv3's absolute position weights:

$$\begin{aligned} z_0 &= e_{\text{tokens}} + e_{pos}, \\ z^{att}_{l} &= z_{l-1} + \text{MHSA}\big(\text{LayerNorm}(z_{l-1})\big), \\ z^{mlp}_{l} &= z^{att}_{l} + \text{MLP}\big(\text{LayerNorm}(z^{att}_{l})\big), \\ e^{enc}_{\text{tokens}} &= \text{LayerNorm}\big(z^{mlp}_{L}\big), \quad l = 1, \ldots, L. \end{aligned} \tag{2}$$

Each block follows the standard pre-LayerNorm Transformer design with residual connections. Our framework supports two DINOv3 variants with identical topology and block design, differing only in the embedding dimension ($D = 192$ for DINOv3-Tiny and $D = 768$ for DINOv3-Base) and the number of attention heads ($h = 3$ and $h = 12$, respectively). This consistent topology is what makes the task-adaptive distillation in Section 3.4 structurally aligned between teacher and student.

3.2 Discriminative Prompt Encoding

Unlike Siamese trackers that rely on template similarity alone, the training frame retains complete background clutter, and the discriminative prior must explicitly separate the target from the background. As shown in Figure 2(a), given the training image x_{tr} and its bounding box $b_t = [x, y, w, h]$, we derive two supervision maps: a Gaussian label map $G_c(b_t)$ for foreground-background discrimination, and a regression distance map $R_d(b_t)$ that encodes pixel-to-box edge offsets:

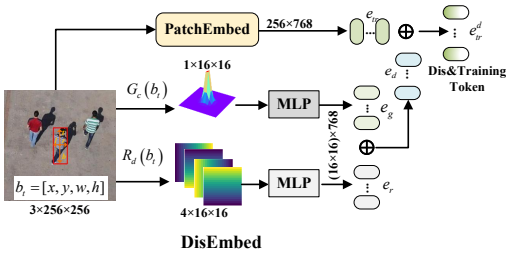
$$y_g = G_c(b_t), \quad y_g \in \mathbb{R}^{1\times h_f \times w_f}, \tag{3}$$

$$d_{\text{ltrb}} = R_d(b_t) = [l, t, r, b] \in \mathbb{R}^{4\times h_f \times w_f}, \tag{4}$$

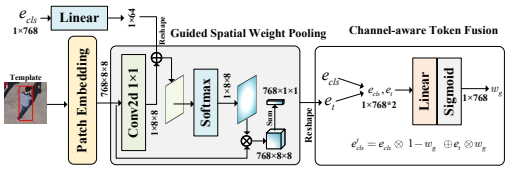
where $[l, t, r, b]$ are normalized distances to the four box edges and $[h_f, w_f] = [16, 16]$ is the feature map size. A lightweight Conv1d MLP encodes y_g into the Gaussian prompt e_g, and a parallel MLP encodes d_{ltrb} into the spatial prompt e_r. The two prompts are combined with learnable scale factors and fused with the training tokens:

$$e_d = \alpha\, e_g \oplus \beta\, e_r, \quad e^d_{tr} = e_{tr} \oplus e_d, \tag{5}$$

where $\oplus$ denotes element-wise addition, $e_{tr} \in \mathbb{R}^{256\times D}$ is the training token sequence, and the learnable scales α and β let the encoder balance the classification prior and the regression prior during training.



(a) Discriminative prompt encoding (DisEmbed). The Gaussian and LTRB maps are encoded by MLPs into prompts and fused with the training tokens.



(b) Guided Spatial Weighted Token Fusion (GSWTF). Dimensions are shown for the DINOv3-Base variant. The student applies the same module with $D = 192$.

Figure 2: Module details of DinoSDTrack. (a) Discriminative prompt encoding injects target priors into the training tokens. (b) GSWTF refines the template tokens into the classification token. Best viewed in color.

3.3 Guided Spatial Weighted Token Fusion (GSWTF)

Template priors are handled in three main ways in the literature. Static template matching concatenates the template tokens into the encoder and relies on implicit attention for refinement Ye et al. (2022); Chen et al. (2022a), so background clutter inside the template crop propagates into every frame and the template influence is unconstrained. Dynamic-template and update-based methods Yan et al. (2021a); Zheng et al. (2024); Kang et al. (2025) refresh the template online, but they introduce update decisions or extra memory and remain vulnerable to drift. Attention-based fusion Gao et al. (2022; 2023); He et al. (2025) strengthens cross-branch interaction, yet often at the cost of additional Transformer layers. Within the discriminative line, ToMP Mayer et al. (2022) converts the template into a target-specific filter through a dedicated predictor, which is computationally heavy and lives outside the one-stream encoder. GSWTF addresses these limitations from within the token space. Instead of passing raw template tokens into the sequence, it compresses them into a compact target feature through query-guided spatial pooling, so that background inside the template crop is suppressed before fusion. As shown in Figure 2(b), the classification token serves as a query that highlights the target within the template. We project e_{cls} into the template grid, combine it with the template feature map, and compute spatial attention weights via a 1×1 convolution followed by Softmax over all template positions:

$$w_{sp} = \text{Softmax}\Big(\mathcal{F}_{sp}\big(\text{Proj}(e_{cls}) \oplus e_{tp}\big)\Big), \tag{6}$$

where $e_{tp} \in \mathbb{R}^{64\times D}$ are the template tokens, Proj maps the classification token to the template grid, and $\mathcal{F}_{sp}$ is a 1×1 convolution. The target template feature is obtained by weighted pooling:

$$e_t = \sum_i w^{(i)}_{sp}\, e^{(i)}_{tp}, \tag{7}$$

which aggregates target-consistent information into a compact representation $e_t \in \mathbb{R}^D$. To inject the template prior while preserving the original query information, we fuse e_{cls} and e_t through learned gating:

$$w_g = \sigma\Big(\text{Linear}\big([e_{cls}, e_t]\big)\Big), \quad e^t_{cls} = e_{cls} \otimes (1 - w_g) \oplus e_t \otimes w_g, \tag{8}$$

where σ is the Sigmoid function and $\otimes$ denotes element-wise multiplication. The refined classification token e_{cls}^{t} aggregates the Siamese template prior into the discriminative pipeline. The refinement is performed once at the encoder input, and no template tokens enter the Transformer sequence afterward, so the template overhead remains constant and minimal. GSWTF introduces only linear layers, a 1×1 convolution, and pooling, adding negligible computational cost while enabling the encoder to capture stronger foreground-background separation.

3.4 Task-Adaptive DINOv3 Knowledge Distillation

DINOv3 provides no official Tiny variant, which prevents its rich self-supervised semantics from being deployed on resource-constrained platforms. To close this gap, we first fine-tune DINOv3-Base on tracking objectives so that its representations are adapted to the tracking task, then freeze it as the teacher and distill the task-adapted representations into our custom DINOv3-Tiny student. This task-adaptive scheme differs from naive base-to-tiny layer matching, which suffers from representational capacity mismatch and accuracy degradation in deeper layers Wang et al. (2026). Both teacher and student share the one-stream architecture, and they produce test features, score maps, and bounding boxes in parallel:

$$\begin{aligned} \big[e_{te}, S, b_t\big]^{\text{teacher}} &= \mathcal{T}_{\text{enc}}^{\text{teacher}}\big(e_{\text{tokens}}\big), \\ \big[e_{te}, S, b_t\big]^{\text{student}} &= \mathcal{T}_{\text{enc}}^{\text{student}}\big(e_{\text{tokens}}\big). \end{aligned} \tag{9}$$

The teacher test features are projected from 768 to 192 dimensions with a linear layer to match the student. The distillation loss combines feature alignment, score consistency, and box regression:

$$\begin{aligned} \mathcal{L}_{\text{feat}} &= 1 - \frac{1}{BM} \sum_{b=1}^{B} \sum_{m=1}^{M} \frac{e_{te}^{\mathrm{T}}\, e_{te}^{\text{linear}}}{\|e_{te}\|\, \|e_{te}^{\text{linear}}\|}, \\ \mathcal{L}_{\text{score}} &= \mathrm{KL}\Big(\mathrm{Softmax}(S_s/T) \,\Big\|\, \mathrm{Softmax}(S_t/T)\Big), \\ \mathcal{L}_{\text{box}} &= \big\|b_t^{\text{student}} - b_t^{\text{teacher}}\big\|_1, \\ \mathcal{L}_{\text{distill}} &= \lambda_{\text{feat}}\mathcal{L}_{\text{feat}} + \lambda_{\text{score}}\mathcal{L}_{\text{score}} + \lambda_{\text{box}}\mathcal{L}_{\text{box}}, \end{aligned} \tag{10}$$

where B is the batch size, $M = 256$ is the token sequence length, $T = 2.0$ is the temperature, and the weighting coefficients are set to $\lambda_{\text{feat}} = 0.2$, $\lambda_{\text{score}} = 0.3$, and $\lambda_{\text{box}} = 0.6$. Cosine similarity aligns feature semantics, KL divergence preserves score distribution, and L1 loss anchors box-level localization, so that the compact student retains the discriminative accuracy of the task-adapted teacher under strict real-time budgets.

3.5 Training Details and Loss Function

We train DinoSDTrack on the training splits of LaSOT Fan et al. (2019), TrackingNet Muller et al. (2018), GOT-10k Huang et al. (2019a), and COCO Lin et al. (2014). The teacher DINOv3-Base is first fine-tuned on tracking objectives, after which it is frozen and the DINOv3-Tiny student is trained with the tracking loss and the distillation loss jointly. We sample 60k video sub-sequences and train for 300 epochs on dual NVIDIA RTX 4090 GPUs. The input size is 256 with a search scale factor of 4, and the template size is 128 with a scale factor of 2. We adopt the AdamW Loshchilov & Hutter (2019) optimizer with a base learning rate of 1×10^{-4}, a backbone learning rate of 2×10^{-5}, a weight decay of 1×10^{-4}, and a learning rate decay factor of 0.2 at epochs 150 and 250. The tracking loss follows OSTrack Ye et al. (2022) and consists of Focal loss for the Gaussian score map and joint L1 and GIoU loss for bounding box regression:

$$\mathcal{L}_{\text{track}} = \lambda_{\text{focal}}\, \mathcal{L}_{\text{focal}}(S, \bar{S}) + \lambda_{\text{l1}}\, \mathcal{L}_{\text{l1}}(\hat{b}, \bar{b}) + \lambda_{\text{giou}}\, \mathcal{L}_{\text{giou}}(\hat{b}, \bar{b}), \tag{11}$$

where $\lambda_{\text{focal}} = 1$, $\lambda_{\text{l1}} = 5$, and $\lambda_{\text{giou}} = 2$. The student is optimized by $\mathcal{L}_{\text{track}} + \mathcal{L}_{\text{distill}}$, with Focal loss stabilizing the discriminative prompt branch and L1 and GIoU loss refining the precise localization of the hybrid prediction head.

Table 1: Comparison results of our DinoSDTrack against state-of-the-art trackers on LaSOT, LaSOText, TrackingNet, GOT-10k, NFS, and TNL2K. Evaluation metrics include Precision (P), Normalized Precision (NP), AUC, Success Rate (SR), and AO (%). **Bold red** and **blue** values represent the top two performances, respectively, with image search region dimensions marked for each tracker.

Trackers	Source	LaSOT			LaSOText			TrackingNet			GOT-10k			NFS	TNL2K
		AUC	NP	P	AUC	NP	P	AUC	NP	P	AO	$SR_{0.5}$	$SR_{0.75}$	AUC	AUC
$OSTrack_{256}$ Ye et al. (2022)	ECCV'22	69.1	78.7	75.2	47.4	57.3	53.3	83.1	87.8	82.0	71.0	80.4	68.2	64.7	54.3
$ARTrack_{256}$ Wei et al. (2023)	CVPR'23	70.4	79.5	76.6	46.4	56.5	52.3	84.2	88.7	83.5	73.5	82.2	70.9	64.3	57.5
GRM_{256} Gao et al. (2023)	CVPR'23	69.9	79.3	75.8	-	-	-	84.0	88.7	83.3	73.4	82.9	70.4	65.6	-
$F\text{-}BDMTrack_{256}$ Yang et al. (2023)	ICCV'23	69.9	79.4	75.8	47.9	57.9	54.0	83.7	88.3	82.6	72.7	82.0	69.9	66.0	-
$ROMTrack_{256}$ Cai et al. (2023)	ICCV'23	69.3	78.8	75.6	48.9	59.3	55.0	83.6	88.4	82.7	72.9	82.9	70.2	**68.0**	-
$LoRAT\text{-}B_{224}$ Lin et al. (2024)	ECCV'24	71.7	80.9	77.3	50.3	61.6	57.1	83.5	87.9	82.1	72.1	81.8	70.7	-	58.8
$ARTrackV2_{256}$ Bai et al. (2024)	CVPR'24	71.6	80.2	77.2	50.8	61.9	57.7	84.9	89.3	84.5	75.9	85.4	72.7	-	-
$AQATrack_{256}$ Xie et al. (2024)	CVPR'24	71.4	81.9	78.6	51.2	62.2	58.9	83.8	88.6	83.1	73.8	83.2	72.1	-	57.8
$EVPTrack_{224}$ Shi et al. (2024)	AAAI'24	70.4	80.9	77.2	48.7	59.5	55.1	83.5	88.3	-	73.3	83.6	70.7	-	57.5
$LMTrack_{256}$ Xu et al. (2025)	AAAI'25	69.8	79.2	76.3	49.0	59.6	55.8	84.2	89.0	82.8	76.3	87.1	73.9	-	-
$JDTrack_{288}$He et al. (2025)	MM'25	73.3	82.7	79.2	50.4	60.7	57.5	84.0	87.7	83.2	**78.2**	87.8	**78.1**	-	-
$ARPTrack_{256}$ Liang et al. (2025)	CVPR'25	72.6	81.4	78.5	52.0	**62.9**	58.7	85.5	90.0	**85.3**	77.7	87.3	74.3	-	-
$MoDTrack_{256}$ Yang et al. (2026a)	AAAI'26	72.9	82.6	79.1	**52.4**	**62.9**	**60.5**	85.0	89.6	84.4	74.0	84.1	71.9	-	**59.7**
$MUTrack_{256}$ Wu et al. (2026)	AAAI'26	72.3	82.9	79.5	51.2	62.5	58.8	85.5	**90.4**	85.2	78.1	**88.3**	77.5	-	-
$GOT\text{-}Edit_{252}$ Chen et al. (2026)	ICLR'26	**73.9**	**84.5**	**81.7**	-	-	-	**85.6**	**90.1**	-	76.4	-	75.8	-	-
DinoSDTrack-B_{256}	**Ours**	**74.3**	**83.8**	**81.3**	**53.2**	**63.7**	**60.8**	**85.7**	89.8	**86.0**	**79.3**	**89.4**	**79.4**	**69.2**	**60.8**

4 EXPERIMENTS

4.1 IMPLEMENTATION DETAILS

We evaluate DinoSDTrack on six general tracking benchmarks and six UAV tracking benchmarks. All test experiments run on an NVIDIA GeForce RTX 4070 GPU. For the general benchmarks we report the task-adapted DINOv3-Base variant DinoSDTrack-B, and for the UAV benchmarks we report the distilled DINOv3-Tiny student DinoSDTrack-T, which preserves discriminative accuracy under strict real-time budgets.

4.2 COMPARISON WITH THE STATE OF THE ART

General Tracking Benchmarks. Table 1 compares DinoSDTrack-B with state-of-the-art trackers on six general tracking benchmarks at a unified input resolution of 256. **LaSOT** Fan et al. (2019). DinoSDTrack-B achieves the best AUC of 74.3%, leading the recent ICLR'26 tracker GOT-Edit (73.9%) and the one-stream discriminative tracker JDTrack (73.3%), and outperforming all one-stream Siamese trackers including MUTrack (72.3%) and ARPTrack (72.6%). This demonstrates that discriminative learning and Siamese template fusion reinforce each other within a one-stream encoder. **LaSOText** Fan et al. (2021). On this generalization benchmark, DinoSDTrack-B attains the best AUC (53.2%), Normalized Precision (63.7%), and Precision (60.8%), gaining +5.8% AUC over OSTrack Ye et al. (2022) and clear margins over the strongest recent competitors MoDTrack and ARPTrack. The strong results on 15 unseen object categories validate the generalization of DINOv3-driven discriminative representations. **TrackingNet** Muller et al. (2018). DinoSDTrack-B reaches the highest AUC (85.7%) and Precision (86.0%), outperforming MUTrack, MoDTrack, and ARTrackV2, with a competitive Normalized Precision of 89.8%. **GOT-10k** Huang et al. (2019a). Following the official protocol with training only on the GOT-10k training split, DinoSDTrack-B achieves the best AO (79.3%), SR0.5 (89.4%), and SR0.75 (79.4%), surpassing JDTrack and MUTrack on all three overlap metrics, which verifies the effectiveness of our framework under the restricted-data protocol. **NFS and TNL2K** Kiani Galoogahi et al. (2017); Wang et al. (2021b). DinoSDTrack-B obtains 69.2% and 60.8% AUC on NFS and TNL2K, ranking first on both benchmarks. Consistent performance across these small and medium-scale benchmarks confirms the robustness and generalization of our framework.

UAV Tracking Benchmarks. Table 2 compares the distilled DinoSDTrack-T with 9 state-of-the-art UAV trackers on six benchmarks using AUC and Precision. **UAV123 and UAV123@10fps** Mueller et al. (2016). DinoSDTrack-T ranks first on UAV123 with 69.0% AUC, surpassing EATrack-DeiT (68.1%) and LETrack (68.0%); its 86.5% Precision trails only EATrack-DeiT (89.0%) and LETrack (87.0%). On UAV123@10fps, it leads both metrics (68.9% AUC / 86.5% Precision), outperforming LETrack and EATrack-DeiT (both 67.4% AUC) by +1.5% and raising Precision by +0.6% over LETrack, which verifies robustness to low-frame-rate target motion. **DTB70** Li & Yeung (2017). Our tracker achieves 66.4% AUC, tied for second with Aba-ViTrack and ORTrack-DeiT behind EATrack-DeiT (66.5%), and 84.2% Precision, remaining competitive under blur and jitter thanks to DINOv3 self-supervised features. **UAVDT** Du et al. (2018). OCTrack leads this small-object

benchmark (63.0% AUC / 85.0% Precision). DinoSDTrack-T reaches 59.0% AUC / 78.1% Precision, outperforming HCAT (54.3%), TCTrack++ (52.6%), and HiT-Small (46.6%) by clear margins and staying on par with mainstream ViT trackers, which validates the center predictor for small-target localization. **UAVTrack112 and UAVTrack112-L** Fu et al. (2021). DinoSDTrack-T attains the highest AUC of 69.8% and 69.3%, surpassing LETrack (69.6% / 68.2%) and all other competitors. In Precision, LETrack remains the strongest (85.8% / 84.8%), while our tracker obtains 81.9% / 81.4%, ranking third on both, demonstrating stable long-term target representation and scale adaptability.

Table 2: AUC and Precision (%) comparisons of DinoSDTrack-T with state-of-the-art trackers on six UAV benchmarks.

Trackers	Source	UAV123		UAV123@10fps		DTB70		UAVDT		UAVTrack112		UAVTrack112-L	
		AUC	Prec.	AUC	Prec.	AUC	Prec.	AUC	Prec.	AUC	Prec.	AUC	Prec.
HiT-Small Kang et al. (2023)	ICCV'23	63.8	80.6	64.3	80.9	58.9	74.8	46.6	61.6	63.6	78.0	59.8	74.6
Aba-ViTrack Li et al. (2023)	ICCV'23	66.4	86.4	65.5	85.0	**66.4**	**85.9**	59.9	**83.4**	-	-	64.2	81.1
AVTrack-DeiT Li et al. (2024)	ICML'24	66.8	84.8	65.8	83.2	65.0	84.3	58.7	82.1	65.4	80.3	62.7	78.2
ORTrack-DeiT Wu et al. (2025)	CVPR'25	66.4	84.3	-	-	**66.4**	**86.2**	60.1	**83.4**	-	-	-	-
SGLATrack-DeiT* Xue et al. (2025)	CVPR'25	66.9	84.9	65.5	82.6	65.1	84.4	59.9	81.9	67.5	**82.8**	64.0	79.2
OCTrack Zhang et al. (2026)	CVPR'26	66.9	84.9	-	-	66.0	85.7	**63.0**	**85.0**	-	-	-	-
EATrack-DeiT Yang et al. (2026b)	CVPR'26	**68.1**	**89.0**	**67.4**	85.4	**66.5**	85.2	**60.2**	82.7	-	-	-	-
LETrack Xue et al. (2026)	CVPR'26	68.0	**87.0**	**67.4**	**85.9**	65.2	85.0	-	-	**69.6**	**85.8**	**68.2**	**84.8**
DinoSDTrack-T	**Ours**	**69.0**	86.5	**68.9**	**86.5**	**66.4**	84.2	59.0	78.1	**69.8**	81.9	**69.3**	**81.4**

4.3 Ablation Study and Analysis

Following the benchmark-model assignment of the main tables, module-level ablations are conducted on the general benchmarks (LaSOT, NFS, TNL2K) with the DinoSDTrack-B model, while distillation-related ablations are conducted on the UAV benchmarks (UAV123, UAVTrack112) with the distilled DinoSDTrack-T model. All models are trained with the same protocol, and the final row of each table corresponds to the full model.

Component-wise Analysis. Table 3 decomposes the overall gain on the general benchmarks. Starting from a one-stream Siamese baseline that performs only template-search matching (70.1% on LaSOT), DisEmbed brings the largest gain (+1.9% on LaSOT), since discriminative supervision explicitly separates foreground from background before self-attention. The one-shot GSWTF further adds +1.1%, and the complete DinoSDTrack-B reaches 74.3% / 69.2% / 60.8% AUC on LaSOT / NFS / TNL2K, confirming that Siamese template fusion and discriminative learning are complementary within one encoder.

Table 3: Ablation of the unified paradigm on LaSOT, NFS, and TNL2K with DinoSDTrack-B.

Configuration	LaSOT	NFS	TNL2K
One-stream Siamese only	70.1	65.1	55.2
+ DisEmbed	72.0	67.5	58.0
+ GSWTF	73.1	68.4	59.5
DinoSDTrack-B	74.3	69.2	60.8

Table 4: Ablation of DisEmbed on LaSOT, NFS, and TNL2K. The best configuration is highlighted in gray.

#	DisEmbed	LaSOT	NFS	TNL2K
1	w/o e_{tr}	70.1	65.1	55.2
2	w/o e_g & e_r	66.2	63.0	52.5
3	w/ e_{tr} w/ e_g	70.6	66.4	56.8
4	w/ e_{tr} w/ e_r	71.3	67.0	57.5
5	w/ e_g & e_r	72.0	67.5	58.0

Table 5: Ablation of GSWTF on LaSOT, NFS, and TNL2K. "In-sequence" concatenates template tokens into the token sequence.

Fusion	LaSOT	NFS	TNL2K
DinoSDTrack-B + DisEmbed	72.0	67.5	58.0
In-sequence fusion	72.4	67.8	58.6
+ GSWTF (ours)	73.1	68.4	59.5

Table 6: Ablation of the tracking head on LaSOT, NFS, and TNL2K.

f_{shared}	f_{att}	LaSOT	NFS	TNL2K
-	-	73.0	68.3	59.8
-	✓	73.6	68.8	60.3
✓	-	73.2	68.5	60.0
✓	✓	74.3	69.2	60.8

Discriminative Embedding. Table 4 ablates DisEmbed on the DINOv3-Base backbone. Removing the discriminative branch entirely (#1) drops accuracy, and adding the training tokens without prompts (#2) is harmful, since uninformative tokens dilute self-attention. The Gaussian prompt e_g and the spatial prompt e_r each contribute independently (#3 and #4), and their fusion (#5) yields the best result, validating that foreground-background supervision and geometric offset cues are complementary.

Template Fusion. Table 5 ablates GSWTF under the DisEmbed-equipped framework on the general benchmarks. GSWTF consistently improves all three datasets (+1.1% / +0.9% / +1.5%). We further compare with the alternative of directly concatenating template tokens into the token sequence (in-sequence fusion), which extends the sequence from 512 to 576 tokens and reaches 72.4% / 67.8% / 58.6%. Our one-shot GSWTF achieves 73.1% / 68.4% / 59.5% instead, showing that one-shot refinement is both more accurate and more efficient than serial template injection, since raw template patches contain background noise that disturbs matching.

Tracking Head. Table 6 ablates the shared backbone f_{shared} and the attention modulation f_{att}. The shared backbone reduces parameters from 8.39M to 6.43M with negligible accuracy loss, and f_{att} brings a clear AUC gain (+1.1% / +0.7% / +0.8% on the shared-backbone setting), validating that the attention module strengthens target-aware representation while the shared backbone provides parameter-efficient refinement.

Distillation. Distillation ablations are conducted on the UAV benchmarks with the distilled student. Table 7 ablates the distillation losses with L_{feat} fixed. Removing L_{score} or L_{box} degrades accuracy progressively, with L_{box} as the dominant term (69.0→67.2 on UAV123), confirming that feature, score, and box-level knowledge are complementary. Table 8 further isolates the contribution of distillation and the role of student capacity. Training the Tiny student without a teacher drops to 66.5% / 66.8%. Distilling from DINOv3-Base recovers 69.0% / 69.8%, and scaling the teacher to DINOv3-Large yields only marginal gains (+0.3% / +0.4%). Directly fine-tuning the DINOv3-Base student (DinoSDTrack-B) reaches the highest accuracy (70.4% / 71.2%) but at roughly four times the inference cost, confirming that task-adaptive distillation compresses most of the Base model's discriminative capacity into the Tiny student under a strict real-time budget.

Table 7: Ablation of distillation losses on UAV123 and UAVTrack112.

L_{feat}	L_{score}	L_{box}	UAV123	UAVTrack112
✓	✓	-	67.2	65.6
✓	-	✓	67.9	66.7
✓	✓	✓	69.0	69.8

Table 8: Effect of distillation and student scale.

Configuration	UAV123	UAVTrack112
Tiny, no teacher	66.5	66.8
Tiny, DINOv3-Base teacher (DinoSDTrack-T)	69.0	69.8
Tiny, DINOv3-Large teacher	69.3	70.2
DinoSDTrack-B (Base, no distillation)	70.4	71.2

Table 9: Sensitivity of distillation loss weights on UAV123 and UAVTrack112. The default (0.2, 0.3, 0.6) is highlighted in gray.

λ_{feat}	λ_{score}	λ_{box}	UAV123 / UAVTrack112
0.1	0.3	0.6	68.7 / 69.5
0.3	0.3	0.6	68.6 / 69.4
0.2	0.2	0.6	68.8 / 69.6
0.2	0.4	0.6	68.7 / 69.5
0.2	0.3	0.4	68.2 / 68.9
0.2	0.3	0.8	68.5 / 69.3
0.2	0.3	0.6	69.0 / 69.8

We also examine sensitivity to the loss weights (Table 9). Deviating from the default (0.2, 0.3, 0.6) incurs up to a 0.9% AUC drop, and the default achieves the best accuracy on both datasets. Box-level supervision dominates the weighting, as lowering λ_{box} to 0.4 costs 0.8% / 0.9% on UAV123 / UAVTrack112 and raising it to 0.8 also degrades accuracy, whereas the feature- and score-level weights are comparatively robust (within 0.4%).

5 Conclusions

We presented DinoSDTrack, a lightweight tracker that unifies Siamese template matching and discriminative learning within a one-stream DINOv3 encoder. Discriminative prompt encoding injects Gaussian and spatial priors into the training tokens, and the GSWTF module refines the Siamese template into the classification token in a one-shot manner with negligible overhead. The hybrid prediction head combines template filtering with fine-grained center regression through a parameter-sharing backbone. To reconcile capacity and efficiency, we first fine-tune DINOv3-Base on tracking objectives and then distill the task-adapted representations into a custom DINOv3-Tiny student. Extensive experiments on six general tracking benchmarks and six UAV tracking benchmarks show that DinoSDTrack attains state-of-the-art accuracy while maintaining real-time inference speed, indicating that task-adapted self-supervised hierarchies can be effectively compressed for discriminative tracking.

AI USE STATEMENT

In this work, we used generative AI tools for polishing parts of the paper text and for LaTeX typesetting assistance. We have not used generative AI tools for collecting or producing experimental data or results, and the remaining required disclosure tasks are not applicable to this work.

REFERENCES

Yifan Bai, Zeyang Zhao, Yihong Gong, and Xing Wei. Artrackv2: Prompting autoregressive tracker where to look and how to describe. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19048–19057, 2024.

Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision Workshops*, pp. 850–865, 2016.

Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6182–6191, 2019.

Vasyl Borsuk, Roman Vei, Orest Kupyn, Tetiana Martyniuk, Igor Krashenyi, and Jiři Matas. Fear: Fast, efficient, accurate and robust visual tracker. In *European conference on computer vision*, pp. 644–663. Springer, 2022.

Wenrui Cai, Qingjie Liu, and Yunhong Wang. Hiptrack: Visual tracking with historical prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19258–19267, 2024.

Wenrui Cai, Qingjie Liu, and Yunhong Wang. Spmtrack: Spatio-temporal parameter-efficient fine-tuning with mixture of experts for scalable visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16871–16881, 2025.

Yidong Cai, Jie Liu, Jie Tang, and Gangshan Wu. Robust object modeling for visual tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9589–9600, 2023.

Ziang Cao, Changhong Fu, Junjie Ye, Bowen Li, and Yiming Li. Hift: Hierarchical feature transformer for aerial tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15457–15466, October 2021.

Ziang Cao, Ziyuan Huang, Liang Pan, Shiwei Zhang, Ziwei Liu, and Changhong Fu. Towards real-world visual tracking with temporal contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15834–15849, 2023.

Boyu Chen, Peixia Li, Lei Bai, Lei Qiao, Qiuhong Shen, Bo Li, Weihao Gan, Wei Wu, and Wanli Ouyang. Backbone is all your need: A simplified architecture for visual object tracking. In *European Conference on Computer Vision*, pp. 375–392. Springer, 2022a.

Shih-Fang Chen, Jun-Cheng Chen, I-Hong Jhuo, Yen-Yu Lin, et al. Got-edit: Geometry-aware generic object tracking via online model editing. In *Proceedings of the International Conference on Learning Representations*, 2026.

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*, pp. 8126–8135, 2021.

Xin Chen, Ben Kang, Dong Wang, Dongdong Li, and Huchuan Lu. Efficient visual tracking via hierarchical cross-attention transformer. In *European Conference on Computer Vision*, pp. 461–477. Springer, 2022b.

Yutao Cui, Cheng Jiang, Gangshan Wu, and Limin Wang. Mixformer: End-to-end tracking with iterative mixed attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46 (6):4129–4146, 2024.

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*, pp. 6638–6646, July 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. pp. 1–21, 2021.

Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *European conference on computer vision*, pp. 370–386, 2018.

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*, pp. 5374–5383, 2019.

Heng Fan, Hexin Bai, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Harshit, Mingzhen Huang, Juehuan Liu, et al. Lasot: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision*, 129:439–461, 2021.

Changhong Fu, Ziang Cao, Yiming Li, Junjie Ye, and Chen Feng. Onboard real-time aerial tracking with efficient siamese anchor proposal network. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2021.

Shenyuan Gao, Chunluan Zhou, Chao Ma, Xinggang Wang, and Junsong Yuan. Aiatrack: Attention in attention for transformer visual tracking. In *European Conference on Computer Vision*, pp. 146–164. Springer, 2022.

Shenyuan Gao, Chunluan Zhou, and Jun Zhang. Generalized relation modeling for transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18686–18695, 2023.

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*, pp. 770–778, 2016.

Xuedong He, Huiying Xu, Xinzhong Zhu, and Hongbo Li. High-performance discriminative tracking with spatio-temporal template fusion. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 709–718, 2025.

João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3): 583–596, 2015.

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, 2019a.

Ziyuan Huang, Changhong Fu, Yiming Li, Fuling Lin, and Peng Lu. Learning aberrance repressed correlation filters for real-time uav tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision)*, pp. 2891–2900, October 2019b.

Sajid Javed, Martin Danelljan, Fahad Shahbaz Khan, Muhammad Haris Khan, Michael Felsberg, and Jiri Matas. Visual object tracking with discriminative filters and siamese networks: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):6552–6574, 2023.

Ben Kang, Xin Chen, Dong Wang, Houwen Peng, and Huchuan Lu. Exploring lightweight hierarchical vision transformers for efficient visual tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9612–9621, October 2023.

Ben Kang, Xin Chen, Simiao Lai, Yang Liu, Yi Liu, and Dong Wang. Exploring enhanced contextual information for video-level object tracking. 2025.

Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Need for speed: A benchmark for higher frame rate object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1125–1134, 2017.

Shuiwang Li, Yangxiang Yang, Dan Zeng, and Xucheng Wang. Adaptive and background-aware vision transformer for real-time uav tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13989–14000, October 2023.

Siyi Li and Dit-Yan Yeung. Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Yiming Li, Changhong Fu, Fangqiang Ding, Ziyuan Huang, and Geng Lu. Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11923–11932, June 2020.

Yongxin Li, Mengyuan Liu, You Wu, Xucheng Wang, Xiangyang Yang, and Shuiwang Li. Learning adaptive and view-invariant vision transformer for real-time UAV tracking. In *International Conference on Machine Learning*, 2024.

Shiyi Liang, Yifan Bai, Yihong Gong, and Xing Wei. Autoregressive sequential pretraining for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7254–7264, June 2025.

Liting Lin, Heng Fan, Zhipeng Zhang, Yaowei Wang, Yong Xu, and Haibin Ling. Tracking meets lora: Faster training, larger model, stronger performance. In *European Conference on Computer Vision*, pp. 300–318, 2024.

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*, pp. 740–755, 2014.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. 2019.

Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8731–8740, 2022.

Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *European Conference on Computer Vision*, pp. 445–461. Springer, 2016.

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)*, pp. 300–317, 2018.

Liangtao Shi, Bineng Zhong, Qihua Liang, Ning Li, Shengping Zhang, and Xianxian Li. Explicit visual prompts for visual object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4838–4846, 2024.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025.

Guijie Wang, Tong Lin, Yifan Bai, Anjia Cao, Shiyi Liang, Wangbo Zhao, and Xing Wei. Fartrack: fast autoregressive visual tracking with high performance. *Proceedings of the International Conference on Learning Representations*, 2026.

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*, pp. 1571–1580, 2021a.

Xiao Wang, Xiujun Shu, Zhipeng Zhang, Bo Jiang, Yaowei Wang, Yonghong Tian, and Feng Wu. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13763–13773, 2021b.

A Waswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, A Gomez, L Kaiser, and I Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Xing Wei, Yifan Bai, Yongchao Zheng, Dahu Shi, and Yihong Gong. Autoregressive visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9697–9706, 2023.

Qiangqiang Wu, Tianyu Yang, Ziquan Liu, Baoyuan Wu, Ying Shan, and Antoni B Chan. Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14561–14571, 2023.

Weijing Wu, Qihua Liang, Bineng Zhong, Xiaohu Tang, Yufei Tan, Ning Li, and Yuanliang Xue. Mutrack: a memory-aware unified representation framework for visual tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 10772–10780, 2026.

You Wu, Xucheng Wang, Xiangyang Yang, Mengyuan Liu, Dan Zeng, Hengzhou Ye, and Shuiwang Li. Learning occlusion-robust vision transformers for real-time uav tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17103–17113, June 2025.

Jinxia Xie, Bineng Zhong, Zhiyi Mo, Shengping Zhang, Liangtao Shi, Shuxiang Song, and Rongrong Ji. Autoregressive queries for adaptive tracking with spatio-temporal transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19300–19309, 2024.

Jinxia Xie, Bineng Zhong, Qihua Liang, Ning Li, Zhiyi Mo, and Shuxiang Song. Robust tracking via mamba-based context-aware token learning. In *Proceedings of the AAAI conference on artificial intelligence*, 2025.

Chenlong Xu, Bineng Zhong, Qihua Liang, Yaozong Zheng, Guorong Li, and Shuxiang Song. Less is more: Token context-aware learning for object tracking. In *Proceedings of the AAAI conference on artificial intelligence*, 2025.

Chaocan Xue, Bineng Zhong, Qihua Liang, Yaozong Zheng, Ning Li, Yuanliang Xue, and Shuxiang Song. Similarity-guided layer-adaptive vision transformer for uav tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6730–6740, June 2025.

Chaocan Xue, Qihua Liang, Bineng Zhong, Yanting Zu, Yuanliang Xue, Haiying Xia, and Shuxiang Song. Toward low-cost yet effective temporal learning for uav tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 42538–42548, June 2026.

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*, pp. 10448–10457, 2021a.

Bin Yan, Houwen Peng, Kan Wu, Dong Wang, Jianlong Fu, and Huchuan Lu. Lighttrack: Finding lightweight neural networks for object tracking via one-shot architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15180–15189, June 2021b.

Dawei Yang, Jianfeng He, Yinchao Ma, Qianjin Yu, and Tianzhu Zhang. Foreground-background distribution modeling transformer for visual object tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10117–10127, 2023.

Hongtao Yang, Bineng Zhong, Qihua Liang, Xiantao Hu, Yufei Tan, Haiying Xia, and Shuxiang Song. Motion-aware object tracking via motion and geometry-aware cues. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11604–11612, 2026a.

Hongtao Yang, Bineng Zhong, Qihua Liang, Yaozong Zheng, Xiantao Hu, Yuanliang Xue, and Shuxiang Song. Dual-branch distilled transformer for efficient asymmetric uav tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13615–13625, June 2026b.

Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision*, pp. 341–357. Springer, 2022.

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*, pp. 9856–9865, 2021.

Jian Zhang, Xincheng Yu, and Yi Lin. Rethinking occlusion modeling for uav tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13563–13573, June 2026.

Yaozong Zheng, Bineng Zhong, Qihua Liang, Zhiyi Mo, Shengping Zhang, and Xianxian Li. Odtrack: Online dense temporal token learning for visual tracking. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 7588–7596, 2024.

Chengao Zong, Xin Chen, Jie Zhao, Yang Liu, Huchuan Lu, and Dong Wang. Enhancing the two-stream framework for efficient visual tracking. *IEEE Transactions on Image Processing*, 34: 5500–5512, 2025.

A Appendix

A.1 Tracking Paradigm

Siamese Paradigm. Siamese tracking leverages weight-sharing Siamese networks to encode template and search images separately, localizing targets via cross-feature matching. Modern one-stream variants integrate feature extraction and interaction into a single ViT using paired template-search inputs ($z \in \mathbb{R}^{3\times128\times128}$ and $x \in \mathbb{R}^{3\times256\times256}$). This paradigm can be formulated as:

$$[x, y, w, h] = \text{Head}\big(\text{ViT}(z, x)\big) \tag{12}$$

where $[x, y]$ denotes the top-left coordinate of the target bounding box, w and h represent its width and height, $\text{ViT}(\cdot)$ denotes the vision transformer with joint template-search feature learning, and $\text{Head}(\cdot)$ is the prediction head for bounding box regression. **Discriminative Paradigm**. The discriminative paradigm learns a target-specific convolutional filter f from a labeled training frame set $F_{\text{train}} = \{(x_j, y_j)\}_{j=1}^{n}$, where $x_j \in \mathbb{R}^{N\times C\times H\times W}$ are training image tensors of target and background regions, and $y_j \in \mathbb{R}^{N\times1\times h_f\times w_f}$ is the corresponding Gaussian target response map. The filter f is optimized for target-background discrimination by minimizing a data-driven generalized least-squares loss:

$$L(f) = \frac{1}{|F_{\text{train}}|} \sum_{(x,y)\in F_{\text{train}}} \|x * f - y\|^2 + \lambda\|f\|^2 \tag{13}$$

where $*$ denotes convolution and λ is the L_2 regularization coefficient for alleviating filter overfitting. In the ViT framework, f is integrated into feature learning via token embedding, encoded as a filter classification token f_{cls} and concatenated with training/test image tokens x_{tr}, x_{te}. The refined bounding box is predicted by fusing the joint feature output with the historical bounding box:

$$[x, y, w, h] = \text{Head}\big(\text{ViT}\big([f_{cls}, x_{tr}, x_{te}]\big)\big) \tag{14}$$

Fusing the two paradigms with CNN backbones is challenging, as it introduces high computation and conflicting optimization objectives. In contrast, the two can be seamlessly unified in our one-stream discriminative framework, since both Siamese matching and discriminative filtering generate

score maps and estimate state offsets within the same token space. We therefore formulate the unified architecture as:

$$[x, y, w, h] = \text{Head}\big(\text{ViT}\big([f_{cls}, z_t, x_{tr}, x_{te}]\big)\big) \tag{15}$$

Our core focus is on achieving more effective discriminative learning while incorporating the Siamese template to boost target sensitivity.

A.2 Optimized Tracking Prediction Head

After the encoder produces the refined classification token e^t_{cls} and the test tokens e_{te}, we build the prediction head by combining the discriminative filter paradigm of ToMP Mayer et al. (2022) with the fine-grained center regression of OSTrack Ye et al. (2022). Following ToMP, we generate a target-specific filter from the classification token and compute a discriminative response map:

$$w_f = \ell\big(\gamma(e^t_{cls})\big), \quad \mathcal{R} = f_{se} * w_f, \tag{16}$$

where $f_{se} = \gamma(e_{te})$ is the reshaped test feature map, $*$ denotes convolution, and ℓ reduces the channel dimension from D to 256. The template filter highlights target regions and suppresses background interference. We broadcast w_f into a spatial attention map and modulate the test features:

$$f^{att}_{te} = f_{se} \otimes \mathcal{A}(w_f), \tag{17}$$

where $\mathcal{A}(\cdot)$ expands the filter into a two-dimensional attention map. The modulated features pass through a shared convolutional backbone with channel transformations $256 \to 256 \to 128 \to 64 \to 32$, followed by three parallel 1×1 projection heads:

$$\begin{aligned} \boldsymbol{M}_{ctr} &= \sigma\Big(\text{Conv}^{32\to1}_{1\times1}\big(f_{\text{shared}}\big)\Big), \\ \boldsymbol{M}_{sz} &= \sigma\Big(\text{Conv}^{32\to2}_{1\times1}\big(f_{\text{shared}}\big)\Big), \\ \boldsymbol{M}_{off} &= \text{Conv}^{32\to2}_{1\times1}\big(f_{\text{shared}}\big), \end{aligned} \tag{18}$$

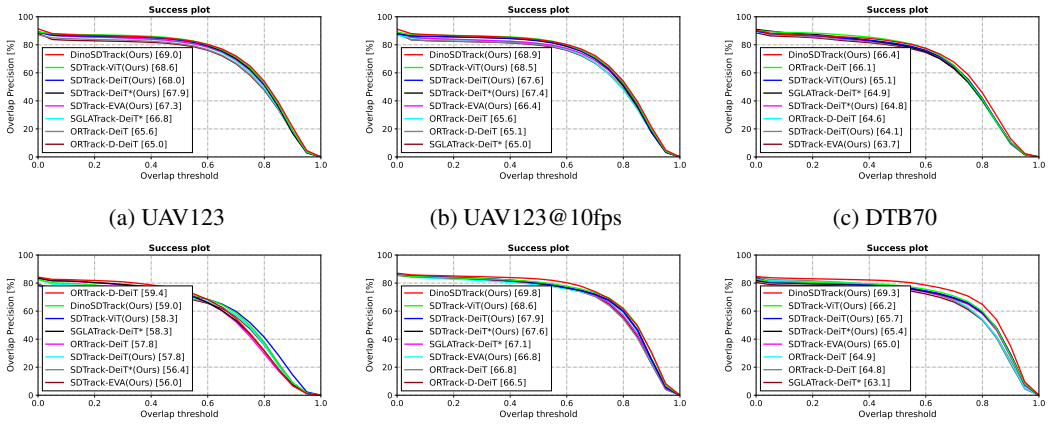
where $\boldsymbol{M}_{ctr}$ encodes the target center probability, $\boldsymbol{M}_{sz}$ predicts normalized width and height, and $\boldsymbol{M}_{off}$ compensates the grid coordinate deviation. We locate the peak position on $\boldsymbol{M}_{ctr}$, aggregate the size and offset predictions at that position, and map them back to the original image space with the patch stride, yielding the final normalized bounding box. The shared backbone is a parameter-sharing design rather than a scenario-specific component: all three tasks reuse one convolutional tower, which reduces redundancy for the lightweight UAV variant without sacrificing accuracy in the general setting. The integrated head inherits ToMP's template filtering for background suppression and OSTrack's multi-branch center regression for precise subpixel localization and adaptive size estimation.

A.3 Backbone Impacts

Figure 3 compares DinoSDTrack variants (diverse ViT-Tiny backbones and the distilled DINOv3-Tiny) with ORTrack and SGLATrack, with results of the compared trackers rerun from their official codes and models on the same hardware for a fair comparison. DinoSDTrack achieves the highest average AUC and Precision across all six benchmarks, and the DinoSDTrack-ViT variant tops UAV123 and UAV123@10fps in both metrics.

A.4 Model Complexity and Efficiency

Table 10 summarizes the model complexity and tracking speed of the two DinoSDTrack variants on an NVIDIA RTX 4070. The distilled Tiny student (DinoSDTrack-T) reduces the FLOPs by more than an order of magnitude (3.07G vs. 45.2G) and the parameters by roughly fourteen times (6.45M vs. 90.11M), while running more than three times faster (254.7 vs. 78.07 FPS), at a small accuracy cost. This enables real-time deployment of the full discriminative framework on resource-constrained UAV platforms.



(a) UAV123 (b) UAV123@10fps (c) DTB70

(d) UAVDT (e) UAVTrack112 (f) UAVTrack112-L

Figure 3: Success plots of DinoSDTrack and SOTA trackers across six UAV benchmarks. All results are obtained on the same hardware for fair comparison.

Table 10: FLOPs, parameters, and tracking speed of DinoSDTrack variants on an NVIDIA RTX 4070.

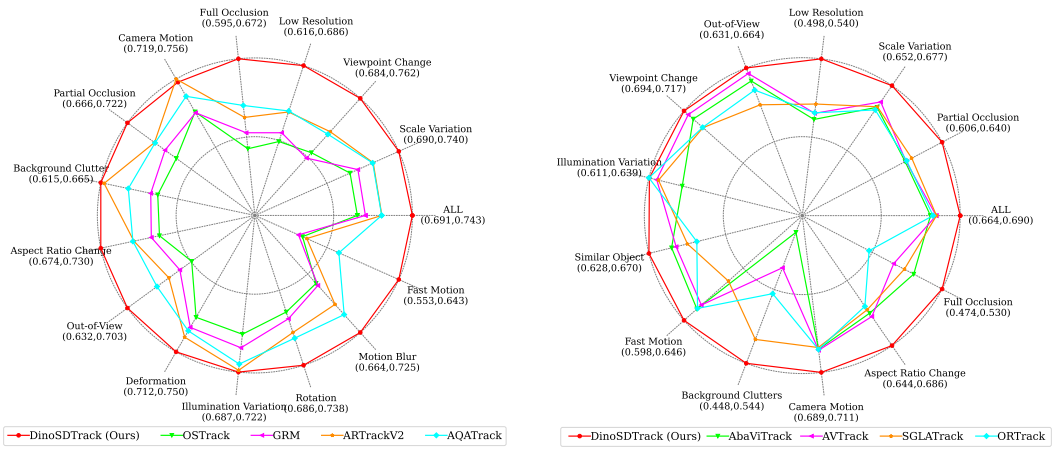
	FLOPs (G)↓	Params (M)↓	Speed (FPS)↑
DinoSDTrack-B	45.2	90.11	78.07
DinoSDTrack-T	3.07	6.45	254.7

A.5 Attribute-wise Comparison

Figure 4 compares DinoSDTrack (red) with the best-performing SOTA trackers across core tracking attributes on LaSOT and UAV123, using the Success metric. Values in parentheses denote (best SOTA, Ours). On LaSOT, DinoSDTrack achieves the highest Success on all assessed attributes, with the largest margins on Fast Motion and Background Clutter, where template matching alone typically struggles. On UAV123, the distilled DinoSDTrack-T consistently outperforms recent ViT-based UAV trackers across all twelve evaluated attributes, including Full Occlusion, Similar Object, and Background Clutters. The overall ALL success (0.743 on LaSOT and 0.690 on UAV123) matches the overall AUC reported in the main tables, confirming that the attribute-level gains aggregate into the overall improvements.

A.6 Qualitative Comparison

To qualitatively assess the tracking performance, we visualize the tracking results of our method (DinoSDTrack) and SOTA trackers across three representative UAV tracking scenarios: *uav_bird1_2* (703 frames), *uav_group1_1* (1333 frames), and *uav_truck1* (463 frames). The results are presented in Figure 5. In the *uav_bird1_2* sequence, the target is extremely small and subject to significant long-range perspective changes. AVTrack (yellow), SGLATrack (black), and ORTrack (blue) exhibit noticeable drift as the sequence progresses (e.g., at frame #0106), failing to maintain consistent alignment with the ground truth. In contrast, our DinoSDTrack (red) consistently adheres to the target's position and scale, demonstrating robust adaptability to long-range and small-target scenarios. For the *uav_group1_1* sequence, which involves pedestrian tracking with scale variation and occlusions, AVTrack and ORTrack exhibit inconsistent bounding box sizes (e.g., at frames #0930 and #1320), either overestimating or underestimating the target scale. SGLATrack achieves moderate performance but occasionally deviates during rapid scale changes. Our method accurately adjusts the bounding box scale to match the target's size across all frames, even during distractor and partial occlusion (e.g., frame #0930), showcasing superior distractor adaptation. In the *uav_truck1* sequence, characterized by motion blur and background clutter, AVTrack and ORTrack suffer from severe drift (e.g., at frames #0238 and #0400), where their bounding boxes either include exces-



(a) LaSOT (b) UAV123

Figure 4: Attribute-wise Success comparisons of DinoSDTrack (red) against the best SOTA tracker on LaSOT and UAV123. Values in parentheses denote (best SOTA, Ours).

sive background or fail to cover the entire target. SGLATrack shows some resistance to drift but still misaligns with the target's edges. Our DinoSDTrack maintains tight and accurate bounding boxes throughout the sequence, robustly filtering out background noise and motion artifacts, which validates its effectiveness in handling challenging environmental factors. Overall, the qualitative results demonstrate that our DinoSDTrack outperforms SOTA trackers in terms of target localization accuracy, scale adaptability, and robustness to complex UAV tracking scenarios.

A.7 Interaction Visualization

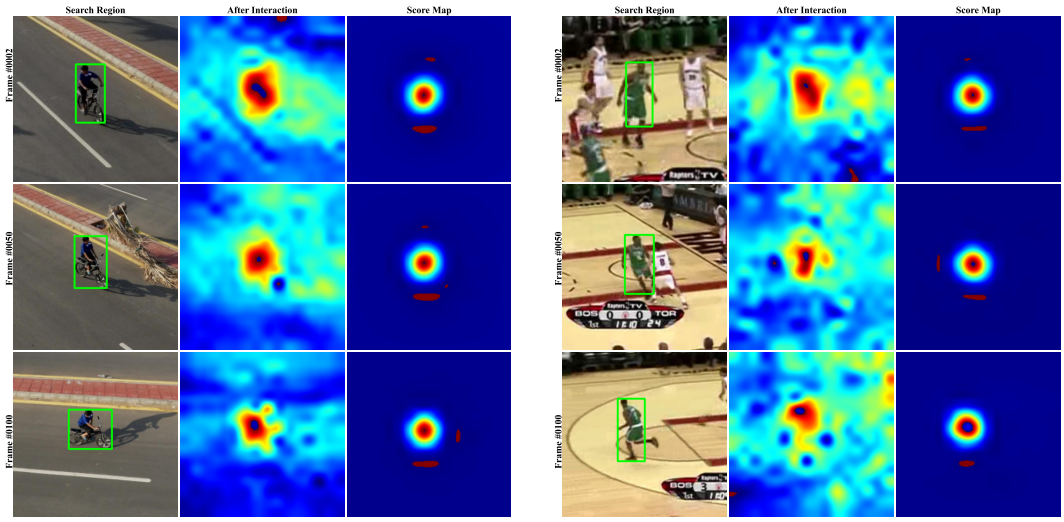
Figure 6 visualizes the intermediate representations of DinoSDTrack on two representative sequences, *Bike1* and *Basketball*. For each frame, we show the search region with the predicted bounding box, the feature response after encoder interaction, and the final score map. The heat responses remain tightly concentrated on the target across frames under background clutter and appearance changes, validating that the discriminative prompt encoding and the hybrid prediction head maintain precise target-background separation throughout the video.

A.8 Limitations

Despite the competitive performance and efficiency achieved by DinoSDTrack on UAV and general tracking benchmarks, our method still exhibits several limitations that deserve further exploration: **Modality Scope**. DinoSDTrack is currently developed for RGB-only UAV tracking and cannot directly support multi-modal inputs (e.g., thermal, depth) that are essential for UAV applications in low-light or adverse weather conditions. **Distillation Hyperparameters**. The loss weights for multi-component knowledge distillation are set empirically, without an automated or learnable tuning mechanism to adapt to different backbones and complex tracking scenarios. **Extreme Small Targets**. Although our method performs well on common UAV challenges, it still observes slight performance drops on densely populated aerial scenes, mainly due to the insufficient high-resolution feature preservation in the lightweight DINOv3-Tiny backbone. These limitations point to promising future directions, including extending our Siamese-discriminative framework to multi-modal UAV tracking, designing learnable distillation scheduling, and enhancing high-resolution feature representation for extreme small targets in UAV scenes.



Figure 5: Qualitative comparison of DinoSDTrack (red) with SOTA trackers on three representative UAV sequences.



(a) *bike1* (b) *Basketball*

Figure 6: Interaction visualization of DinoSDTrack. For each sequence, three frames are shown with the search region (with the predicted bounding box), the feature response after encoder interaction, and the final score map.