# MSR-TRACK: MITIGATING SEARCH REPRESENTATION DILUTION IN INFRARED ANTI-UAV TRACKING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Reliable infrared anti-UAV tracking is difficult because aerial targets occupy few pixels, provide weak appearance cues, and are frequently obscured by clutter, thermal distractors, rapid motion, or temporary disappearance. Although adaptive local trackers enlarge the search area to recover targets, resizing the larger crop to a fixed input shrinks the target's relative scale and admits additional background. We call this effect search representation dilution. To address it, we introduce MSR-Track, which enhances search representations at complementary token and spatial stages. Lightweight Token Refinement (LTR) recalibrates search tokens before template–search interaction without altering their number or spatial order. The Multi-Scale Response Enhancement (MSRE) head aggregates fine-scale evidence and broad context to reinforce weak UAV responses before bounding-box prediction. MSR-Track obtains 62.5% AUC, 86.1% Precision, and 82.2% Normalized Precision on Anti-UAV410, exceeding our FocusTrack-SRA by 2.3, 2.9, and 2.7 points while adding little computation. Tests on Anti-UAV300 and CST Anti-UAV demonstrate cross-dataset generalization under unseen distributions.

## 1 INTRODUCTION

Infrared anti-UAV tracking aims to continuously localize UAV targets in thermal infrared videos (Jiang et al., 2023; Huang et al., 2024). The task is difficult because long-range UAVs often occupy only a few pixels and provide weak texture and shape cues. Background clutter, thermal distractors, abrupt motion, occlusion, and temporary disappearance further increase tracking drift, making the setting more challenging than generic object tracking (Bertinetto et al., 2016; Li et al., 2019; Danelljan et al., 2019; Bhat et al., 2019). Because the next search crop is centered on the current prediction, a localization error can propagate to later frames.

Transformer trackers commonly crop a local search region around the previous target position and model template–search relations through attention (Chen et al., 2021; Yan et al., 2021; Ye et al., 2022; Cai et al., 2023; Cui et al., 2023). This local paradigm is efficient but can miss the target after abrupt displacement or out-of-view disappearance. Once the target falls entirely outside the crop, no target pixels remain for local matching. FocusTrack therefore adjusts the local search region according to estimated target presence, improving coverage and recovery while avoiding continuous global search (Wang et al., 2025).

Search enlargement, however, changes the composition of the input. Once the enlarged crop is resized to the same fixed resolution, the tiny UAV occupies a smaller relative area and substantially more background and pseudo-target patterns enter the search image. We call this condition *search representation dilution*. It does not imply that enlargement necessarily reduces overall accuracy: the coverage benefit may remain dominant. Rather, it creates a background-dominant representation in which weak target evidence is harder to distinguish. With a fixed token-grid size, each search token in a larger crop represents a wider region of the original scene, increasing the need for discriminative features. Figure 1 examines how the performance gap varies across fixed search factors. Both trackers benefit from broader coverage, but the advantage of MSR-Track grows from 1.6 to 3.0 AUC points as the search factor increases, motivating representation enhancement without sacrificing the adaptive local-search pipeline. Our comparisons with FocusTrack-SRA therefore keep the search policy and input resolution unchanged.

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.

The dilution effect follows from the crop-and-resize operation. If the linear extent of a search crop grows from factor $f_1$ to $f_2$ while the input resolution stays fixed, an approximately unchanged target occupies only $f_1/f_2$ of its former normalized width and height, and roughly $(f_1/f_2)^2$ of its former area. At the same time, more background structures enter the input. For infrared UAVs that cover very few pixels, this change can move useful evidence toward the spatial resolution limit of the backbone. The goal is therefore not simply to search farther, but to preserve discriminative target evidence while doing so. This challenge intensifies as the search factor increases. We address it by refining search tokens and strengthening spatial responses.

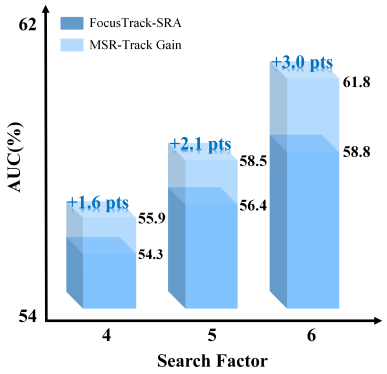



Figure 1: Search-factor study. The MSR-Track gain grows from 1.6 to 3.0 AUC points as broader crops introduce more background.

We propose **MSR-Track**, which strengthens the search representation at two complementary stages while retaining the baseline interface. A Lightweight Token Refinement (LTR) module first aggregates the search sequence into a compact latent set and injects its information back into every original search token through position-wise gated residual fusion. Unlike token pruning or merging, LTR preserves the token count and spatial order required by dense localization. A Multi-Scale Response Enhancement (MSRE) head then combines fine-scale evidence with coarse contextual cues immediately before prediction, strengthening sparse target responses without introducing multiple prediction levels. LTR acts before template–search interaction, where it reduces background-dominant token interference; MSRE acts after transformer encoding, where it reinforces spatially weak responses for localization. The two modules therefore address the same failure mechanism at different representation stages rather than duplicating one operation.

Our contributions are threefold:

- We characterize *search representation dilution* in adaptive local anti-UAV tracking: search enlargement improves coverage but reduces the target's relative proportion in a fixed-resolution input and introduces substantially more background content.
- We introduce MSR-Track, combining shape-preserving token refinement and lightweight multi-scale spatial enhancement at complementary stages without changing the dense prediction interface.
- Extensive experiments on Anti-UAV410, together with module ablations, search-factor analysis, parameter sensitivity, attribute evaluation, efficiency analysis, qualitative results, and direct cross-dataset transfer, verify the proposed design.

## 2 Related Work

**Transformer and anti-UAV tracking.** Transformer trackers model template–search relations with attention (Chen et al., 2021; Yan et al., 2021; Cai et al., 2023; Cui et al., 2023). Earlier designs often separated feature extraction and relation modeling, whereas unified architectures concatenate template and search tokens and process them in a shared backbone for efficient dense localization (Chen et al., 2022; Ye et al., 2022; Cui et al., 2022). These models are effective when the local crop contains a clear target, but infrared anti-UAV tracking adds extremely small objects, weak texture, thermal distractors, and complex backgrounds (Jiang et al., 2023; Huang et al., 2024). The difficulty is amplified after abrupt motion or disappearance because a fixed local crop may no longer cover the target.

Long-term approaches address this problem through broad or global re-detection (Huang et al., 2020; Voigtlaender et al., 2020), but repeatedly processing large image regions is expensive and exposes the tracker to many distractors. FocusTrack instead enlarges the local search region according to predicted target presence (Wang et al., 2025). MSR-Track adopts its SRA-only variant as the

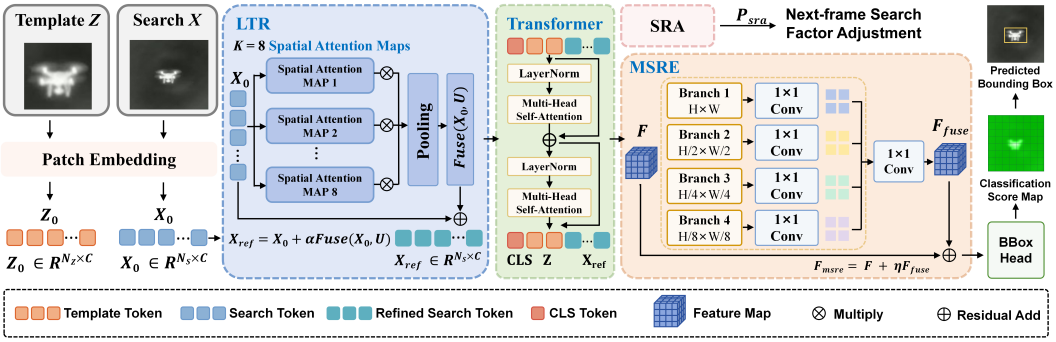



Figure 2: MSR-Track architecture. LTR refines the complete search-token grid before transformer interaction; MSRE enhances the resulting 2D search map before the original bounding-box head. SRA retains the baseline next-frame search-factor adjustment.

baseline and retains the same adaptive search policy. Our focus is complementary: once a broader crop has been selected, we improve the representation that reaches template–search interaction and dense prediction.

**Multi-scale tiny-target representation.** Multi-scale and contextual modeling combine fine target details with broader context in tiny-object and infrared small-target detection (Lin et al., 2017a; Gong et al., 2021; Dai et al., 2021; Liu et al., 2024), while spatial–temporal modeling has been explored for sequential infrared detection (Tong et al., 2024). Fine-resolution features are sensitive to sparse target responses, whereas coarser receptive fields help distinguish a target from structured thermal clutter. Conventional feature pyramids obtain this complementarity from several backbone stages, which introduces lateral connections and changes the downstream feature interface. MSRE instead operates only on the final search map. Every scale-specific branch is restored to the original resolution, fused once, and added residually before the unchanged dense prediction head. It is therefore a response enhancement module rather than a new multi-level detector.

**Token refinement.** Token pruning, merging, and reorganization reduce redundant computation by discarding or combining visually redundant tokens (Rao et al., 2021; Liang et al., 2022; Bolya et al., 2023); latent-token methods instead aggregate visual evidence into a compact representation for global reasoning (Ryoo et al., 2021). These ideas are attractive for a background-dominant search region, but direct token removal is risky in tracking: a tiny UAV may occupy only one or a few tokens, and the prediction head expects a regular spatial grid. LTR therefore uses latent tokens only as an internal refinement basis. Their information is written back to the complete sequence through learned gates, so no search location is deleted or reordered. This distinction lets LTR exploit global aggregation without sacrificing the dense correspondence required by localization.

# 3 Method

## 3.1 Overall Framework

Given a template patch $z$ and a search patch $x$, MSR-Track follows the FocusTrack-SRA local tracking pipeline (Wang et al., 2025), as illustrated in Figure 2. Patch embedding (PE) produces template and search tokens, LTR refines only the search sequence, the shared transformer backbone $\mathcal{B}$ performs template–search interaction, SRA predicts target presence for the next-frame search-factor adjustment, and MSRE refines the decoded search map before prediction. The complete pipeline is:

$$\begin{aligned} Z_0 &= \mathrm{PE}(z), \qquad X_0 = \mathrm{PE}(x), \qquad X_0^{\mathrm{ref}} = \mathcal{R}(X_0), \\ [c_L, Z_L, X_L] &= \mathcal{B}\big([c_0; Z_0; X_0^{\mathrm{ref}}]\big), \qquad \hat{y}_{\mathrm{sra}} = \mathcal{S}(c_L, X_L), \\ F &= \mathrm{Unflatten}(X_L), \qquad F_{\mathrm{msre}} = \mathcal{M}(F), \qquad (P, \hat{B}) = \mathcal{H}(F_{\mathrm{msre}}). \end{aligned} \tag{1}$$

Here, $Z_0 \in \mathbb{R}^{N_z \times C}$ and $X_0 \in \mathbb{R}^{N_s \times C}$. With a $128 \times 128$ template, $256 \times 256$ search image, and patch size 16, $N_z = 64$ and $N_s = 256$. The classification token is $c_0$, $\hat{y}_{\mathrm{sra}} \in [0, 1]$ is the predicted target-presence probability, $F \in \mathbb{R}^{C \times H \times W}$ with $H = W = 16$, $P$ is the score map, and $\hat{B}$ is the predicted box. Both proposed modules preserve the dimensions expected by the baseline backbone and head.

Equation 1 also makes the division of responsibility explicit. SRA operates on the encoded representation and controls the search factor used for the next frame; it is not modified by our method. LTR changes the content, but not the geometry, of the search-token sequence that enters the backbone. MSRE changes the response quality, but not the shape, of the decoded search map that enters the head. This interface preservation allows the pretrained backbone, the adaptive local-search mechanism, and the center-based prediction head to be retained. It also prevents a wider crop from being compensated by an uncontrolled increase in input resolution or by a second global-search branch.

### 3.2 Lightweight Token Refinement

Most search tokens in infrared anti-UAV tracking describe background, yet pruning them risks removing the few tokens occupied by the target and breaks the regular grid. LTR instead forms $K$ latent summaries from all search tokens:

$$A = \mathrm{Softmax}_{N_s}\big(g(X_0)^\top\big) \in \mathbb{R}^{K \times N_s}, \qquad U = AX_0 \in \mathbb{R}^{K \times C}. \tag{2}$$

The token-wise network $g : \mathbb{R}^C \to \mathbb{R}^K$ is a two-layer MLP with hidden dimension 32 and GELU, and the softmax normalizes each latent token over the $N_s$ search positions. We use $K = 8$. The latent set $U$ summarizes the search sequence but never replaces it.

The normalization direction is important. Each row of $A$ defines one distribution over spatial search positions, allowing different latent tokens to collect complementary evidence from the same dense sequence. Because the aggregation is learned from the current search input, the latent set can emphasize a weak target neighborhood in one frame and suppress a structured distractor in another. Unlike a fixed global average, the $K$ summaries retain several distinct modes of the search content; unlike hard selection, they remain differentiable and do not remove candidate locations.

LTR writes the aggregated information back through position-wise gates:

$$\begin{aligned} G &= \sigma(h_f(\mathrm{LN}_f(X_0))) \in \mathbb{R}^{N_s \times K}, \\ X_0^{\mathrm{ref}} &= X_0 + \alpha\, G\, \mathrm{LN}_u(U). \end{aligned} \tag{3}$$

Here, $h_f : \mathbb{R}^C \to \mathbb{R}^K$ is another two-layer GELU MLP, $\sigma$ is the sigmoid, $\mathrm{LN}_f$ and $\mathrm{LN}_u$ are layer normalization, and $\alpha$ is a learnable residual scale. Gates are independent rather than normalized across the $K$ latent tokens. The mapping $\mathcal{R} : \mathbb{R}^{N_s \times C} \to \mathbb{R}^{N_s \times C}$ thus preserves both token count and spatial order.

For search position $i$, the $i$-th row of $G$ determines which latent summaries should refine that position. Independent sigmoid gates are used instead of a softmax across latent tokens because several summaries may be simultaneously useful, while all gates may be small when the original token is already reliable. The residual path protects the baseline representation and lets the learned scale $\alpha$ regulate the magnitude of the update. The additional aggregation and redistribution both scale as $\mathcal{O}(N_s K C)$, with $K \ll N_s$; the transformer still processes the original $N_s$ search tokens. LTR therefore targets representation quality rather than obtaining speed through token deletion.

### 3.3 Multi-Scale Response Enhancement

After transformer encoding, $X_L$ is reshaped to $F \in \mathbb{R}^{C \times H \times W}$. Even when SRA maintains target coverage, an enlarged crop can produce sparse UAV responses amid extensive background activations. For each branch $s \in \{1, \ldots, S\}$, MSRE applies average-pooling downsampling, a branch-specific $1 \times 1$ projection, batch normalization, ReLU $\delta$, and bilinear upsampling:

$$F_s = \mathrm{Up}_s\big(\delta\big(\mathrm{BN}_s\big(\mathrm{Conv}^s_{1\times1}(\mathrm{Down}_s(F))\big)\big)\big), \quad r_s = 2^{s-1}. \tag{4}$$

$\mathrm{Down}_s$ uses kernel and stride $r_s$; $\mathrm{Up}_s$ returns the branch to $H \times W$ without corner alignment. The first branch is identity in spatial resampling. Each projection maps $C$ channels to $C_b = 128$. With

$S = 4$, the pre-upsampling resolutions are $H \times W$, $H/2 \times W/2$, $H/4 \times W/4$, and $H/8 \times W/8$. Fine branches retain local evidence; coarse branches provide context for suppressing structured distractors.

Average pooling is used as a parameter-free spatial summarizer. The $s = 1$ branch retains the original spatial sampling and is therefore most responsive to a small, localized activation. Increasing $r_s$ progressively enlarges the effective context observed by a branch, which helps determine whether an isolated response is consistent with its neighborhood or belongs to an extended background structure. A separate $1 \times 1$ projection and normalization are learned for every branch so that scale-specific responses are calibrated before fusion. No additional anti-aliasing filter or prediction head is introduced.

The restored branches are concatenated and projected back to $C$ channels, followed by residual refinement:

$$F_{\text{fuse}} = \delta\Big(\text{BN}\Big(\text{Conv}_{1\times1}^{\text{fuse}}([F_1; \ldots; F_S])\Big)\Big),$$
$$F_{\text{msre}} = F + \eta F_{\text{fuse}}. \tag{5}$$

For $S = 4$, the fusion layer maps $4C_b = 512$ channels to $C = 768$; $\eta$ is learnable. Consequently, $\mathcal{M} : \mathbb{R}^{C\times H\times W} \to \mathbb{R}^{C\times H\times W}$ strengthens multi-scale responses without changing the baseline prediction interface.

The branch outputs are concatenated only after being restored to the same $H \times W$ grid, so each fused location remains aligned with the dense prediction coordinates. The $1 \times 1$ fusion projection learns how much fine and coarse evidence to combine without introducing an additional spatial convolution after upsampling. As in LTR, a residual connection and learnable scale $\eta$ allow the enhancement to be introduced gradually. This is especially useful for tiny targets: the fine branch protects a response that may cover only one cell, while the coarse branches supply enough context to reject broad thermal structures that can otherwise dominate the score map.

### 3.4 Training Objective

MSR-Track introduces no new annotations or loss terms beyond those used by FocusTrack-SRA. Following FocusTrack-SRA, the objective is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{track}} + \lambda_{\text{sra}}\mathcal{L}_{\text{sra}},$$
$$\mathcal{L}_{\text{track}} = \lambda_f\mathcal{L}_{\text{focal}} + \lambda_1\mathcal{L}_1 + \lambda_g\mathcal{L}_{\text{giou}}, \qquad \mathcal{L}_{\text{sra}} = \text{BCE}(y_{\text{sra}}, \hat{y}_{\text{sra}}), \tag{6}$$

where $(\lambda_f, \lambda_1, \lambda_g) = (1, 5, 2)$, $\mathcal{L}_{\text{focal}}$ is focal classification loss (Lin et al., 2017b), and $\mathcal{L}_{\text{giou}}$ is generalized IoU loss (Rezatofighi et al., 2019). Training is stage-wise: we first train the MSRE-enhanced tracker, then insert LTR and optimize its parameters with Equation 6; the inference pipeline remains unchanged. The first stage lets the spatial prediction path adapt to multi-scale response fusion under the established tracking objective. The second stage then learns token refinement on top of this stable localization pathway. Both stages use the same bounding-box and target-presence supervision, so the method requires neither segmentation masks nor auxiliary labels. At test time, LTR and MSRE are executed in a single forward pass and SRA continues to update the local search factor exactly as in the baseline.

## 4 Experiments

### 4.1 Experimental Settings

**Datasets and protocol.** We evaluate mainly on Anti-UAV410 (Huang et al., 2024), a thermal-infrared benchmark containing 410 sequences and more than 438K annotated frames. Its official split contains 200 training, 90 validation, and 120 test sequences, with many targets smaller than 50 pixels and some below 10 pixels. These properties make it suitable for evaluating representation quality under small target size, background clutter, scale variation, fast motion, and occlusion. All Anti-UAV410 results use the same test protocol unless stated otherwise. For cross-dataset generalization, the Anti-UAV410-trained model is evaluated directly on Anti-UAV300 (Jiang et al., 2023) and CST Anti-UAV (Xie et al., 2025), without target-set fine-tuning, parameter adaptation, or architectural changes.

**Baseline reproduction.** We independently reproduced FocusTrack-SRA (Wang et al., 2025) with SRA enabled and ATM disabled. On Anti-UAV410, our reproduction achieves 60.2% AUC, compared with the 62.3% reported for this variant in the original paper. Our reproduced baseline and MSR-Track were trained and evaluated under the same experimental setup, including the training schedule, random seed, and evaluation protocol. Under this setup, MSR-Track achieves 62.5% AUC, a 2.3-point improvement over our reproduced baseline. The cause of the 2.1-point reproduction gap has not yet been established.

**Metrics.** We report success AUC, center Precision, and Normalized Precision. Success evaluates the intersection-over-union between predicted and ground-truth boxes over a range of overlap thresholds, and its area under the curve summarizes overall tracking accuracy. Precision is the percentage of frames whose center-location error falls within 20 pixels. Normalized Precision divides center error by target size before thresholding, reducing the bias introduced by scale differences. We use all three metrics because a tracker can obtain a plausible overlap while still exhibiting unstable center localization on extremely small boxes.

**Implementation and comparison policy.** The $128 \times 128$ template and $256 \times 256$ search sizes match FocusTrack-SRA (Wang et al., 2025), our baseline because it retains adaptive search-region adjustment while offering a favorable accuracy–efficiency trade-off. Complete FocusTrack is additionally reported as a stronger but slower reference. Unless varied in the sensitivity study, LTR uses $K = 8$ latent tokens and MSRE uses $S = 4$ branches with branch width $C_b = 128$. Training and evaluation use one NVIDIA RTX A6000 GPU. All component ablations start from the same FocusTrack-SRA configuration and retain the same input sizes and evaluation code; thus, differences within an ablation table reflect the inserted representation module rather than a changed search resolution or metric implementation.

### 4.2 Comparison with Representative Trackers

Table 1: Comparison on the Anti-UAV410 test set. Input denotes search-image size. Best results are bold.

| Method | Input | AUC | P | $P_{\mathrm{Norm}}$ |
|---|---|---|---|---|
| TransT (Chen et al., 2021) | 256 | 48.2 | 67.7 | 64.1 |
| ETTrack (Blatter et al., 2023) | 256 | 41.5 | 59.7 | 54.8 |
| MixFormerV2-S (Cui et al., 2023) | 224 | 45.6 | 64.1 | 60.0 |
| GRM (Gao et al., 2023) | 256 | 42.3 | 58.5 | 55.1 |
| ARTrack (Wei et al., 2023) | 256 | 48.2 | 67.2 | 62.9 |
| JointNLT (Zhou et al., 2023) | 320 | 48.4 | 69.0 | 64.5 |
| SeqTrack (Chen et al., 2023) | 256 | 52.2 | 73.8 | 70.0 |
| PromptVT (Zhang et al., 2024) | 320 | 50.5 | 71.5 | 65.6 |
| STARK-ST101 (Yan et al., 2021) | 320 | 56.2 | 78.5 | 74.6 |
| TCTrack (Cao et al., 2022) | 287 | 41.1 | 60.4 | 56.0 |
| OSTrack (Ye et al., 2022) | 256 | 53.7 | 73.9 | 70.9 |
| ToMP50 (Mayer et al., 2022) | 288 | 54.1 | 73.8 | 70.2 |
| ToMP101 (Mayer et al., 2022) | 288 | 54.2 | 75.0 | 70.5 |
| SwinTrack-Tiny (Lin et al., 2022) | 224 | 53.0 | 71.4 | 68.1 |
| SwinTrack-Base (Lin et al., 2022) | 384 | 55.9 | 76.4 | 72.3 |
| AiATrack (Gao et al., 2022) | 320 | 58.6 | 82.3 | 78.0 |
| ROMTrack (Cai et al., 2023) | 256 | 54.7 | 74.5 | 71.7 |
| ZoomTrack (Kou et al., 2023) | 256 | 58.4 | 81.2 | 77.4 |
| MixFormerV2-B (Cui et al., 2023) | 288 | 58.7 | 80.5 | 76.8 |
| DropTrack (Wu et al., 2023) | 256 | 59.2 | 82.2 | 78.2 |
| FocusTrack-SRA (reproduced) (Wang et al., 2025) | 256 | 60.2 | 83.2 | 79.5 |
| FocusTrack (reproduced) (Wang et al., 2025) | 256 | 60.9 | 83.6 | 80.4 |
| **MSR-Track** | 256 | **62.5** | **86.1** | **82.2** |

As shown in Table 1, MSR-Track achieves 62.5% AUC, 86.1% Precision, and 82.2% Normalized Precision. Compared with our reproduced FocusTrack-SRA, it improves these metrics by 2.3, 2.9, and 2.7 points, respectively. It also exceeds our reproduced complete FocusTrack by 1.6, 2.5, and 1.8 points, respectively. The comparison includes transformer trackers with different input sizes as well as methods specifically retrained for Anti-UAV410. MSR-Track uses the same $256 \times 256$ search input as its baseline, so its improvement is not obtained by feeding a larger rasterized search image to the backbone.

The distinction from complete FocusTrack is also informative. Complete FocusTrack adds a substantially more expensive attention-to-mask pathway, whereas MSR-Track operates through compact token aggregation and a single final-map enhancement head. The result suggests that explicitly improving the representation inside the adaptive local crop can recover accuracy without requiring a second dense mask-prediction process. Figure 3 further shows a consistent ordering over overlap and center-error thresholds rather than a gain tied to one selected operating point.

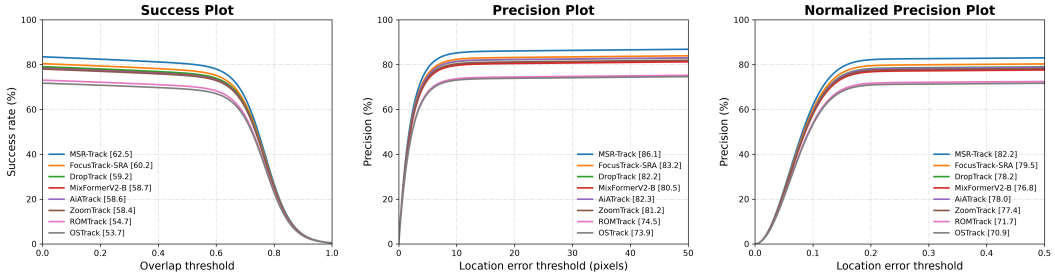



Figure 3: Success, Precision, and Normalized Precision plots on the Anti-UAV410 test set.

### 4.3 Direct Cross-Dataset Generalization

Every result in Table 2 uses Anti-UAV410-trained weights without target-set fine-tuning. On Anti-UAV300, MSR-Track improves FocusTrack-SRA by 0.9 AUC and 0.7 points on both precision metrics. The baseline is already strong on this dataset, yet the consistent change across all three measures indicates that the gain is not tied to one evaluation threshold.

Table 2: Direct cross-dataset generalization.

| Dataset | Method | AUC | P | $P_{\text{Norm}}$ |
|---|---|---|---|---|
| Anti-UAV300 | OSTrack (Ye et al., 2022) | 57.3 | 75.9 | 75.0 |
| | ROMTrack (Cai et al., 2023) | 61.6 | 82.3 | 81.0 |
| | DropTrack (Wu et al., 2023) | 65.3 | 87.4 | 86.1 |
| | ZoomTrack (Kou et al., 2023) | 68.1 | 91.3 | 89.6 |
| | FocusTrack-SRA (Wang et al., 2025) | 75.4 | 96.0 | 94.9 |
| | **MSR-Track** | **76.3** | **96.7** | **95.6** |
| CST Anti-UAV | FocusTrack-SRA (Wang et al., 2025) | 23.6 | 44.2 | 47.2 |
| | **MSR-Track** | **26.5** | **48.1** | **51.2** |

CST Anti-UAV presents a stronger distribution shift: absolute scores are markedly lower for both methods. MSR-Track nevertheless improves the same baseline by 2.9 AUC, 3.9 Precision, and 4.0 Normalized Precision points. This wider margin is compatible with the intended role of token aggregation and multi-scale context when test appearance differs from the training distribution. The low absolute CST scores also bound the claim: the proposed modules transfer better than the baseline under the same source-training and target-evaluation protocol, but they do not solve infrared domain generalization.

### 4.4 Component and Attribute Analysis

As shown in Table 3, the ablation isolates the two representation stages. LTR alone raises AUC from 60.2% to 61.7%, while MSRE alone reaches 61.9%. Their combination gives the best result on every metric: 62.5% AUC, 86.1% Precision, and 82.2% Normalized Precision. The gain beyond either single module supports the intended ordering: LTR improves evidence before template–search interaction, and MSRE further organizes the encoded response before box estimation.

The attribute analysis in Table 4 relates the overall gain to dynamic background clutter (DBC), scale variation (SV), fast motion (FM), and occlusion (OC). MSR-Track ranks first on all four attributes. Relative to FocusTrack-SRA, the gains are 3.8 points on DBC, 0.3 on SV, 1.1 on FM, and 3.1 on OC. The largest changes occur when misleading background responses are strong or target evidence is incomplete. This distribution agrees with the design: LTR recalibrates ambiguous search tokens, while MSRE compares sparse responses with broader context. Attribute evaluation cannot assign every corrected frame to one module, but it connects the full model's improvement to the failure modes motivating representation enhancement.

Table 3: Ablation of LTR and MSRE on Anti-UAV410.

| Baseline | LTR | MSRE | AUC | P | $P_{\text{Norm}}$ |
|---|---|---|---|---|---|
| ✓ | – | – | 60.2 | 83.2 | 79.5 |
| ✓ | ✓ | – | 61.7 | 85.2 | 81.4 |
| ✓ | – | ✓ | 61.9 | 85.5 | 81.7 |
| ✓ | ✓ | ✓ | **62.5** | **86.1** | **82.2** |

Table 4: Attribute-wise Precision@20px on Anti-UAV410.

| Method | DBC | SV | FM | OC |
|---|---|---|---|---|
| OSTrack | 60.4 | 53.6 | 62.0 | 41.2 |
| ROMTrack | 65.9 | 62.9 | 62.1 | 44.9 |
| DropTrack | 64.6 | 73.4 | 73.7 | 46.1 |
| AiATrack | 68.3 | 72.4 | 72.3 | 47.9 |
| MixFormerV2-B | 65.8 | 67.6 | 72.0 | 48.6 |
| FocusTrack-SRA | 65.1 | 77.1 | 76.6 | 46.2 |
| **MSR-Track** | **68.9** | **77.4** | **77.7** | **49.3** |

### 4.5 Capacity and Efficiency

Table 5: Sensitivity to LTR token count $K$ and MSRE branch count $S$.

| Module | Setting | AUC | P | $P_{\text{Norm}}$ |
|---|---|---|---|---|
| LTR | $K = 4$ | 60.9 | 84.6 | 80.7 |
| **LTR** | $\boldsymbol{K = 8}$ | **61.7** | **85.2** | **81.4** |
| LTR | $K = 12$ | 61.0 | 84.4 | 80.8 |
| LTR | $K = 16$ | 59.8 | 82.2 | 78.3 |
| MSRE | $S = 1$ | 60.6 | 83.9 | 79.9 |
| MSRE | $S = 2$ | 61.0 | 84.6 | 80.1 |
| MSRE | $S = 3$ | 61.1 | 84.8 | 80.8 |
| **MSRE** | $\boldsymbol{S = 4}$ | **61.9** | **85.5** | **81.7** |
| MSRE | $S = 5$ | 60.8 | 84.2 | 80.4 |

Table 6: Speed, computation, and accuracy on Anti-UAV410. FPS is measured on an NVIDIA RTX A6000.

| Tracker | FPS | MACs (G) | AUC | P |
|---|---|---|---|---|
| OSTrack | 133.4 | 21.5 | 53.7 | 73.9 |
| ROMTrack | 109.8 | 29.0 | 54.7 | 74.5 |
| ZoomTrack | 136.1 | 21.5 | 58.4 | 81.2 |
| DropTrack | 102.1 | 48.3 | 59.2 | 82.2 |
| FocusTrack-SRA | 139.3 | 29.1 | 60.2 | 83.2 |
| FocusTrack | 39.6 | 30.1 | 60.9 | 83.6 |
| **MSR-Track** | 131.9 | 29.3 | **62.5** | **86.1** |

For the sensitivity study in Table 5, we vary one module at a time. LTR peaks at $K = 8$ with 61.7% AUC: $K = 4$ under-summarizes heterogeneous search content, whereas 12 and 16 latent tokens reintroduce redundancy, with $K = 16$ falling below the baseline. MSRE improves from $S = 1$ through $S = 4$ and peaks at 61.9% AUC, 85.5% Precision, and 81.7% Normalized Precision; a fifth, very coarse branch degrades all metrics. We therefore use $K = 8$ and $S = 4$. The two controls are complementary rather than interchangeable: $K$ sets the number of global latent modes used for token recalibration, whereas $S$ sets the number of spatial contexts fused before prediction.

As shown in Table 6, the representation gain preserves the efficiency of adaptive local tracking. MSR-Track reaches 62.5% AUC at 131.9 FPS and 29.3 G MACs. Relative to FocusTrack-SRA under the same A6000 setup, it adds only 0.2 G MACs (about 0.7%), retains 94.7% of the measured throughput, and gains 2.3 AUC points. Our reproduced complete FocusTrack reaches 60.9% AUC at 39.6 FPS, while MSR-Track is more accurate and over three times faster in this setting. The generic trackers contextualize the operating point; the controlled efficiency claim is the comparison with the identically evaluated baseline because FPS remains hardware- and implementation-dependent.

### 4.6 Qualitative Evidence and Limitations

The examples in Figure 4 complement aggregate scores by showing tracking behavior under five difficult conditions. It compares all trackers on identical frames where UAV evidence is weak or a distractor is dominant.

Under strong building or thermal responses in Figure 4(a,b), competing trackers shift toward high-response background structures, whereas MSR-Track remains closer to the UAV. In (c,d), scale change leaves a small, unstable response; combining native-resolution evidence with broader context improves alignment. The fast-motion cases (e,f) contain several plausible nearby responses. LTR preserves candidate locations while recalibrating the dense sequence, avoiding irreversible pruning before correspondence is established.

In the occlusion cases (g,h), competing predictions are attracted to surrounding structures while MSR-Track remains nearer the annotation. The extremely tiny target in (i) summarizes the background-dominant regime. These observations agree with the stronger DBC, FM, and OC results in Table 4, although qualitative examples do not establish causality.

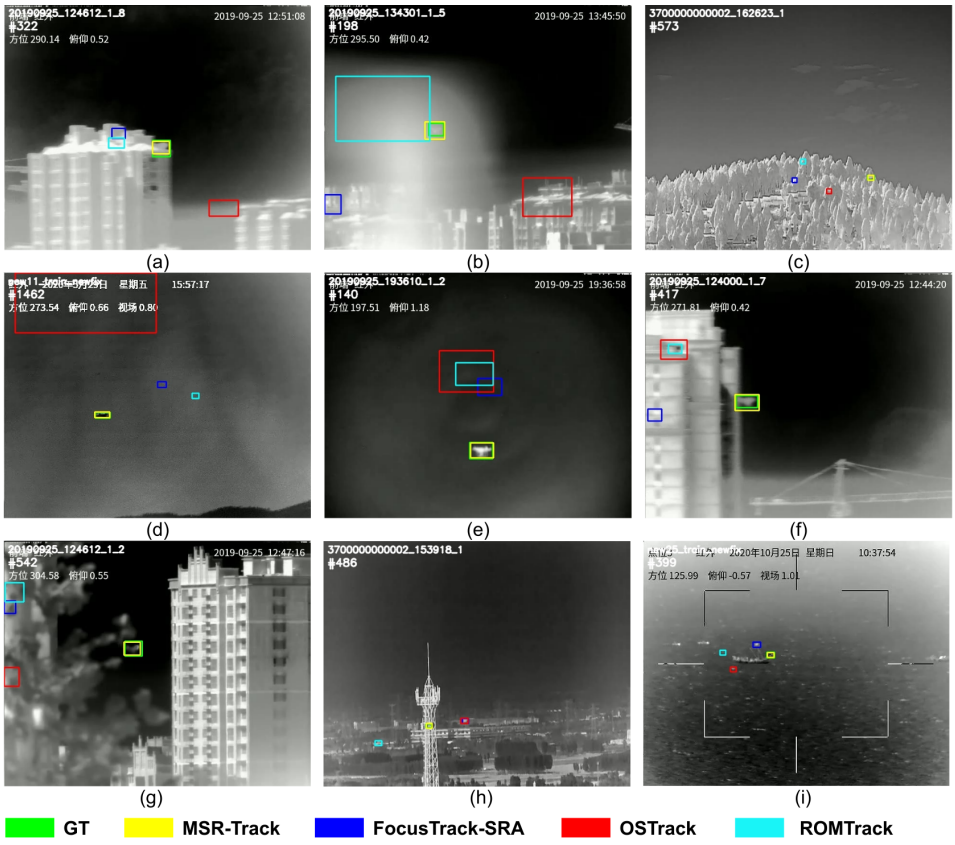



Figure 4: Representative Anti-UAV410 sequences. Green: ground truth; yellow: MSR-Track; blue: FocusTrack-SRA; red: OSTrack; cyan: ROMTrack. The examples cover dynamic background clutter, scale variation, fast motion, occlusion, and an extremely tiny target.

**Limitations.** MSR-Track remains a local tracker and cannot recover a target that stays outside the maximum search region for a prolonged period. A global re-detector could help, but would change the efficiency profile. Evaluation is limited to infrared anti-UAV data: direct transfer improves over the same baseline, yet low CST Anti-UAV scores reveal unresolved domain shift. Stage-wise training also adds optimization complexity, and the ablations do not exhaust every insertion layer or pooling design. Future work should study domain robustness, long-term recovery, and unified training while retaining adaptive local-search efficiency.

## 5 Conclusion

We presented MSR-Track for efficient infrared anti-UAV tracking. It addresses *search representation dilution* in adaptive local search: LTR refines the complete search-token grid before transformer interaction, while MSRE enhances multi-scale spatial responses before the unchanged prediction head. Search-factor analysis supports the motivation, component ablations verify complementarity, and main-benchmark, transfer, attribute, efficiency, and qualitative results demonstrate gains in accuracy and robustness. These findings show that weak target evidence inside the selected crop can be strengthened without discarding spatial tokens or forfeiting adaptive local-search efficiency.

### AI use statement

Generative AI was used primarily for language polishing, with limited assistance in drafting and LaTeX formatting. The authors reviewed all AI-assisted content and take responsibility for the final manuscript.

## REFERENCES


Luca Bertinetto, Jack Valmadre, Joao F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. In *ECCV Workshops*, pp. 850–865, 2016.

Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *ICCV*, pp. 6182–6191, 2019.

Philippe Blatter, Menelaos Kanakis, Martin Danelljan, and Luc Van Gool. Efficient visual tracking with exemplar transformers. In *WACV*, pp. 1571–1581, 2023.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *ICLR*, 2023.

Yidong Cai, Jie Liu, Jie Tang, and Gangshan Wu. Robust object modeling for visual tracking. In *ICCV*, pp. 9589–9600, 2023.

Ziang Cao, Ziyuan Huang, Liang Pan, Shiwei Zhang, Ziwei Liu, and Changhong Fu. TCTrack: Temporal contexts for aerial tracking. In *CVPR*, pp. 14798–14808, 2022.

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 *ECCV*, pp. 375–392, 2022.

Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *CVPR*, pp. 8126–8135, 2021.

Xin Chen, Houwen Peng, Dong Wang, Huchuan Lu, and Han Hu. SeqTrack: Sequence to sequence learning for visual object tracking. In *CVPR*, pp. 14572–14581, 2023.

Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. MixFormer: End-to-end tracking with iterative mixed attention. In *CVPR*, pp. 13608–13618, 2022.

Yutao Cui, Tianhui Song, Gangshan Wu, and Limin Wang. MixFormerV2: Efficient fully transformer tracking. In *NeurIPS*, volume 36, pp. 58736–58751, 2023.

Yimian Dai, Yiquan Wu, Fei Zhou, and Kobus Barnard. Asymmetric contextual modulation for infrared small target detection. In *WACV*, pp. 950–959, 2021.

Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ATOM: Accurate tracking by overlap maximization. In *CVPR*, pp. 4660–4669, 2019.

Shenyuan Gao, Chunluan Zhou, Chao Ma, Xinggang Wang, and Junsong Yuan. AiATrack: Attention in attention for transformer visual tracking. In *ECCV*, pp. 146–164, 2022.

Shenyuan Gao, Chunluan Zhou, and Jun Zhang. Generalized relation modeling for transformer tracking. In *CVPR*, pp. 18686–18695, 2023.

Yuqi Gong, Xuehui Yu, Yao Ding, Xiaoke Peng, Jian Zhao, and Zhenjun Han. Effective fusion factor in FPN for tiny object detection. In *WACV*, pp. 1160–1168, 2021.

Bin Huang, Jianan Li, Jun Chen, Gang Wang, Jian Zhao, and Tingfa Xu. Anti-UAV410: A thermal infrared benchmark and customized scheme for tracking drones in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):2852–2865, 2024.

Lianghua Huang, Xin Zhao, and Kaiqi Huang. GlobalTrack: A simple and strong baseline for long-term tracking. In *AAAI*, volume 34, pp. 11037–11044, 2020.

Nan Jiang, Kexin Wang, Xiaoke Peng, Xue Yu, Qiang Wang, Junliang Xing, Guohao Li, Guodong Guo, Qixiang Ye, Jianbin Jiao, Jian Zhao, and Zhenjun Han. Anti-UAV: A large-scale benchmark for vision-based UAV tracking. *IEEE Transactions on Multimedia*, 25:486–500, 2023.

Yutong Kou, Jin Gao, Bing Li, Gang Wang, Weiming Hu, Yizheng Wang, and Liang Li. ZoomTrack: Target-aware non-uniform resizing for efficient visual tracking. In *NeurIPS*, volume 36, pp. 50959–50977, 2023.

Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. SiamRPN++: Evolution of siamese visual tracking with very deep networks. In *CVPR*, pp. 4282–4291, 2019.

Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *ICLR*, 2022.

Liting Lin, Heng Fan, Zhipeng Zhang, Yong Xu, and Haibin Ling. SwinTrack: A simple and strong baseline for transformer tracking. In *NeurIPS*, volume 35, pp. 16743–16754, 2022.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pp. 2117–2125, 2017a.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pp. 2980–2988, 2017b.

Qiankun Liu, Rui Liu, Bolun Zheng, Hongkui Wang, and Ying Fu. Infrared small target detection with scale and location sensitivity. In *CVPR*, pp. 17490–17499, 2024.

Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *CVPR*, pp. 8731–8740, 2022.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *NeurIPS*, volume 34, pp. 13937–13949, 2021.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, pp. 658–666, 2019.

Michael S. Ryoo, A. J. Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. TokenLearner: Adaptive space-time tokenization for videos. In *NeurIPS*, volume 34, pp. 12786–12797, 2021.

Xiaozhong Tong, Zhen Zuo, Shaojing Su, Junyu Wei, Xiaoyong Sun, Peng Wu, and Zongqing Zhao. ST-Trans: Spatial-temporal transformer for infrared small target detection in sequential images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–19, 2024.

Paul Voigtlaender, Jonathon Luiten, Philip H. S. Torr, and Bastian Leibe. Siam R-CNN: Visual tracking by re-detection. In *CVPR*, pp. 6578–6588, 2020.

Ying Wang, Tingfa Xu, and Jianan Li. FocusTrack: A self-adaptive local sampling algorithm for efficient anti-uav tracking. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–14, 2025.

Xing Wei, Yifan Bai, Yongchao Zheng, Dahu Shi, and Yihong Gong. Autoregressive visual tracking. In *CVPR*, 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 *CVPR*, pp. 14561–14571, 2023.

Bin Xie, Congxuan Zhang, Fagan Wang, Peng Liu, Feng Lu, Zhen Chen, and Weiming Hu. CST Anti-UAV: A thermal infrared benchmark for tiny UAV tracking in complex scenes. In *ICCV Workshops*, pp. 6216–6225, 2025.

Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *ICCV*, pp. 10448–10457, 2021.

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

Ming Zhang, Qi Zhang, Weiming Song, Dong Huang, and Qi He. PromptVT: Prompting for efficient and accurate visual tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(8):7373–7385, 2024.

Li Zhou, Zikun Zhou, Kaige Mao, and Zhenyu He. Joint visual grounding and tracking with natural language specification. In *CVPR*, pp. 23151–23160, 2023.