

SENTRY : SAM2-Enhanced Neighbor-Aware and Temporally Reasoned Memory for Visual Tracking

Mohamad Alansari^{1*} , Yonathan Michael^{1*} , Hasan AlMarzouqi¹ ,
Muzammal Naseer^{1,2} , Naoufel Werghi¹ , and Sajid Javed¹ 

¹Khalifa University, Abu Dhabi, UAE ²University of Western Australia, Australia
{100061914,100053679,hasan.almarzouqi,muhammadmuzammal.naseer,
naoufel.werghi,sajid.javed}@ku.ac.ae
<https://hamadya.github.io/SENTRY/>

Abstract. We revisit the memory update mechanism in SAM2-based visual object tracking and identify confidence-only mask selection as the dominant cause of drift under occlusion, rapid motion, and distractors. We introduce SENTRY, a training-free, plug-and-play, refine-before-write module that validates each memory update for short-horizon temporal consistency before committing it. SENTRY aggregates diverse segmentation hypotheses per frame, backtracks them into short tracklets, and uses neighbor-aware cycle-consistent matching against recent trajectories to favor temporally and geometrically consistent masks. It leaves the base architecture untouched, replacing confidence-driven writes with consistency-validated ones. For fair evaluation, we re-evaluate major open-source SAM2-based trackers across all available scales and datasets, filling gaps in prior reports. Integrated into five strong baselines, SENTRY delivers consistent gains across nine benchmarks, achieving new zero-shot SOTA on LaSOT, LaSOT_{ext}, GOT-10k, VOT20, VOT22, and DiDi. Despite these checks, the SAM2-L version runs at 32.8 FPS on an A100, and across compatible hosts adds only about 0.4–0.6 GB VRAM. Our results provide the first unified all-scale evaluation of SAM2-based trackers and show that enforcing temporal validity at write time stabilizes memory-augmented tracking without retraining.

Keywords: Segment Anything · Temporal Verification · Visual Object Tracking

1 Introduction

Memory-based architectures have become central to visual object tracking (VOT) by maintaining feature histories that help recover from occlusion, motion blur, and appearance changes. Instead of processing frames independently [2, 8, 36, 41, 50, 62, 67, 69, 75], recent trackers build temporal associations by querying a memory bank of past representations, improving robustness and identity preservation [16, 31–34, 72, 78].

* Equal contribution.

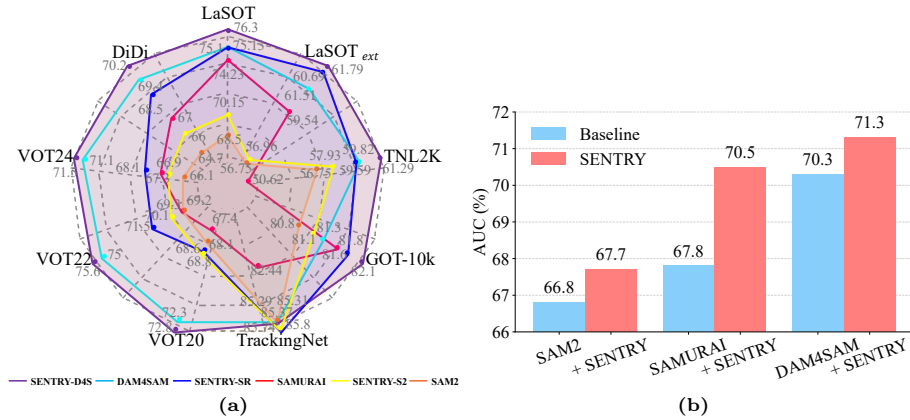


Fig. 1: SENTRY performs favorably against SAM2-based variants across major tracking benchmarks. We denote SENTRY applied to SAM2, SAMURAI, and DAM4SAM as SENTRY-S2, SENTRY-SR, and SENTRY-D4S. (a) AUC per benchmark; (b) average AUC across [22, 23, 47, 60].

SAM2 [51] follows this trend with streaming memory and multiple mask hypotheses per frame. Its main weakness is confidence-driven mask selection: the highest-confidence mask is chosen without verifying short-term temporal consistency. Under occlusion, fast motion, or distractors, this cue is often unreliable, causing incorrect masks to be written into memory and triggering drift (Fig. 2(a)). Existing extensions attempt to mitigate this with motion filtering [71] or distractor checks [58], but they still rely heavily on SAM2’s confidence-based mask selection. Motion filters can break under nonlinear trajectories, and heuristic thresholds often fail in cluttered scenes, leading to similar error accumulation (Figs. 2b,c). To our knowledge, *current SAM2-based trackers lack a systematic, trajectory-level mechanism to verify short-term temporal consistency before a memory write*.

We address this gap with **SAM2-Enhanced Neighbor-aware and Temporally Reasoned memory** (SENTRY), a training-free inference-time memory-admission module that reduces incorrect memory writes. SENTRY targets a different interface from prior motion- or distractor-aware extensions: the memory-write decision. Rather than only improving current-frame prediction or score, it verifies which candidate mask should become future evidence in the autoregressive memory stream. SENTRY replaces confidence-driven selection with a refine-before-write mechanism that explicitly tests each hypothesis for short-horizon temporal plausibility. For every candidate, SENTRY (i) propagates it backward over a fixed window, (ii) applies neighbor-aware checks that penalize identity swaps with nearby instances or distractors, and (iii) evaluates cycle-consistency by requiring that forward-backward trajectories remain aligned with the target’s recent motion. This trajectory-level verification, absent in existing SAM2-style architectures, adds a lightweight temporal reasoning layer without modifying the backbone or memory. Because SENTRY operates entirely outside the backbone and memory module, it integrates cleanly with existing SAM2-based variants

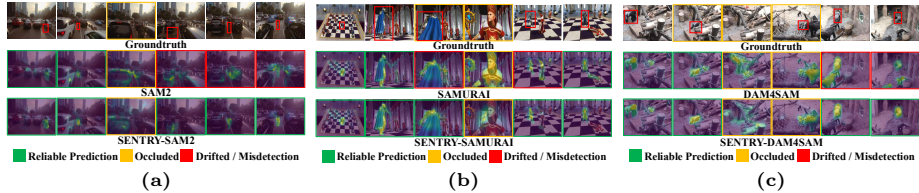


Fig. 2: Qualitative comparison under occlusion, abrupt motion, and distractor interference. We visualize decoder attention to highlight spatial focus during tracking. (a) SAM2 [51] loses target identity after occlusion, (b) SAMURAI [71] drifts under rapid motion, and (c) DAM4SAM [58] misidentifies distractors due to heuristic filtering. The last row shows SENTRY maintaining accurate localization and consistent masks.

and substantially improves robustness when confidence cues are unreliable (see the bottom row in Fig. 2). Our contributions are:

1. We propose SENTRY, a training-free refine-before-write module that enforces short-horizon temporal consistency before any memory update in SAM2-based trackers. It replaces confidence-driven selection with a unified mechanism that evaluates multi-source mask candidates through backward re-tracking and neighbor-aware trajectory matching.
2. To the best of our knowledge, we conduct the first unified, all-scale re-evaluation of five open-source SAM2-based trackers across nine benchmarks and four model sizes, establishing consistent and reproducible baselines.

Experiments across nine benchmarks including LaSOT [23], LaSOT_{ext} [22], TNL2K [60], GOT-10k [28], TrackingNet [47], VOT20/22 [31, 32], VOTS24 [33], and DiDi [58], demonstrate that integrating SENTRY into strong baselines yields consistent improvements and new zero-shot SOTA performance (Fig. 1), while preserving real-time operation (Sec. 4.4, App. J).

2 Related Work

2.1 Memory-Based Visual Object Tracking

Memory networks are widely used in VOT and video object segmentation (VOS) to maintain temporal coherence under appearance change, occlusion, and motion. Key-value memory designs such as STM [49] and FEELVOS [59] established long-term feature propagation, later refined through kernelized or decomposed attention [17, 55, 73] and transformer-based memory structures [16, 72, 78]. These ideas were further explored in segmentation-based tracking frameworks such as STMTrack [25], and inspired memory-augmented single- and multi-object trackers including RTracker [29], MeMOT [6], and multimodal variants like MemVLT [24]. Despite architectural diversity, memory-based VOS trackers typically update memory using frame-level confidence, similarity metrics, or simple schedules. When predictions are unreliable due to occlusion, motion blur, or distractors, incorrect masks or features may enter memory and cause drift. Existing works explore restricted or hierarchical updates [16, 78], but they still assume the selected representation is temporally reliable. Explicit short-horizon verification of a mask’s temporal consistency before committing it to memory has received little attention.

2.2 SAM2 and Its Extensions

SAM2 [51] generalizes memory-based reasoning to a promptable segmentation-tracking framework but retains a confidence-driven memory update, making it vulnerable to high-confidence but temporally inconsistent masks. Extensions stabilize this process from different directions: motion-based priors in SAMU-RAI [71], distractor filtering in DAM4SAM [58], long-term memory tree in SAM2Long [21], and improved scoring or hierarchical features in SAMITE [66] and HiM2SAM [10]. However, they do not explicitly validate competing candidate masks through short-horizon backtracking before the selected prediction becomes future tracking state. Across SAM2-based variants [10, 21, 51, 58, 66, 71], temporal consistency is treated mainly as a current-frame prediction, scoring, or trajectory-selection problem rather than as an explicit memory-admission decision at the write interface. We address this with a training-free mechanism that checks short-term consistency before storage, complementing existing SAM2 extensions without modifying their architectures or memory rules. For plug-in evaluation, we focus on five open-source SAM2-based trackers that operate in an online, frame-by-frame setting [10, 51, 58, 66, 71]. We report SAM2Long [21] as a standalone baseline rather than applying SENTRY to it, because SAM2Long performs constrained memory-tree inference over multiple branches, whereas SENTRY is a pre-write admission rule for a single online memory stream. Combining them would require redesigning branch expansion, pruning, or trajectory scoring, making it a new hybrid method rather than a controlled plug-in ablation.

3 Method

3.1 Preliminaries

SAM2 [51] consists of an image encoder, a prompt encoder, a memory module, and a mask decoder. **Image Encoder.** A hierarchical ViT backbone (Hiera [53]) produces patch tokens for each frame, which serve as visual features queried by the memory-augmented decoder. **Prompt Encoder.** User prompts (points, boxes, or masks) are encoded into tokens. In tracking, only the first frame is prompted; subsequent frames rely solely on memory-guided decoding. **Memory Module.** The memory bank $\mathbf{Mem}^t$ stores embeddings of the initialization mask and a fixed-length queue of recent frame-mask pairs. Temporal cues are added to the recent frames to capture their order, while the initialization frame is left unaltered to serve as a fixed supervised anchor. Each memory token encodes frame features modulated by the predicted mask, and the queue is updated in FIFO order. Incorrect writes to $\mathbf{Mem}^t$ propagate errors forward, causing drift, one of the key failure modes that SENTRY addresses. **Mask Decoder.** The decoder attends to $\mathbf{Mem}^t$ and the current frame features to produce three candidate hypotheses $\tilde{\mathcal{M}}_{\text{dec}}^t = \{\tilde{\mathcal{M}}_1^t, \tilde{\mathcal{M}}_2^t, \tilde{\mathcal{M}}_3^t\}$, each with a predicted IoU score IoU_i^t and occlusion score $\hat{o}_i^t$. SAM2 selects the mask with the highest IoU_i^t and writes it to memory; however, this selection relies solely on instantaneous confidence and often yields temporally inconsistent memory updates.

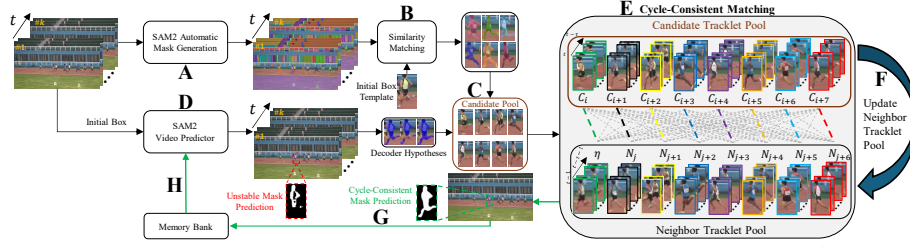


Fig. 3: Overview of the SENTRY framework. Per-frame candidates combine Automatic Mask Generation (AMG) proposals (A–B) and decoder hypotheses (D) into a joint set (C). Each candidate is evaluated through cycle-consistent, neighbor-aware bipartite matching in trajectory space (E–F). The most consistent mask is written to memory (G–H) following the baseline update schedule.

3.2 SENTRY

SENTRY is a training-free, architecture-agnostic module adding a refine-before-write step to SAM2-based trackers. Unlike prior works that impose temporal consistency through training objectives [16, 72], SENTRY performs explicit consistency validation at inference time, preventing erroneous memory updates online. Instead of relying on the highest-confidence decoder mask, SENTRY selects the mask with the highest short-horizon temporal consistency before it is written to memory. This makes temporal reasoning part of the tracker state transition: the admitted mask becomes the memory state used for future predictions, rather than a post-hoc correction of the current frame. As shown in Fig. 3, SENTRY consists of three stages: (i) candidate generation from Automatic Mask Generation (AMG) (A–B) and decoder (D) to form $\mathcal{C}^t$ (C); (ii) cycle-consistent, neighbor-aware reasoning (E–F); and (iii) memory update (G–H). Algorithm 1 summarizes the refine-before-write procedure. Intuitively, SENTRY first collects multiple plausible masks for the target, then re-tracks each candidate backward over a short time window to test which one aligns best with the target’s recent motion before memory writing.

Candidate Mask Generation. SAM2 generates two sources of per-frame hypotheses: (1) three decoder masks with predicted IoU and occlusion scores, and (2) a set of target-agnostic proposals from AMG. SENTRY filters and regularizes these hypotheses into a compact, diverse candidate set that preserves plausible target states.

Decoder hypotheses. The decoder outputs masks $\tilde{\mathcal{M}}_{\text{dec}}^t = \{\tilde{\mathcal{M}}_i^t\}_{i=1}^3$ with corresponding IoU scores $\text{Io}\hat{\text{U}}_i^t$ and occlusion scores $\hat{o}_i^t$. SENTRY retains hypotheses that are (i) close to the best decoder prediction and (ii) not flagged as occluded:

$$\mathcal{C}_{\text{dec}}^t = \left\{ \tilde{\mathcal{M}}_i^t \mid \text{Io}\hat{\text{U}}_i^t > \alpha \max_j \text{Io}\hat{\text{U}}_j^t \wedge \hat{o}_i^t > 0 \right\}, \quad (1)$$

where $\alpha \in [0, 1]$ controls tolerance to decoder uncertainty.

AMG proposals. AMG generates N^t target-agnostic proposals $\{\tilde{\mathcal{M}}_{\text{AMG},j}^t\}$. Let the image regions covered by these masks be $\mathcal{R}_j^t$. To identify proposals relevant to the target, SENTRY computes cosine similarity between proposal features and a

first-frame template $\mathcal{T}^0$, defined as the image region covered by the initialization mask $\mathcal{M}^0$. Let $f(\cdot)$ denote image encoder features averaged over the region. The proposal template similarity is:

$$s_j = \frac{f(\mathcal{R}_j^t) \cdot f(\mathcal{T}^0)}{\|f(\mathcal{R}_j^t)\|_2 \|f(\mathcal{T}^0)\|_2}. \quad (2)$$

We keep proposals that match the target relatively well:

$$\mathcal{C}_{\text{AMG}}^t = \left\{ \tilde{\mathcal{M}}_{\text{AMG},j}^t \mid s_j \geq \beta \max_k s_k \right\}, \quad (3)$$

with $\beta \in [0, 1]$ controlling tolerance to appearance variation. Both thresholds (α and β) are fixed globally across all datasets (see App. A).

Candidate filtration. The union $\hat{\mathcal{C}}^t = \mathcal{C}_{\text{dec}}^t \cup \mathcal{C}_{\text{AMG}}^t$ may contain redundant or overlapping hypotheses. Each mask is projected to its minimum-enclosing bounding box, and Soft-NMS [5] is applied in box space to suppress near duplicates while preserving spatial diversity. The remaining boxes are mapped back to their masks, yielding the refined candidate set $\hat{\mathcal{C}}^t$.

Kalman motion prior. To maintain robustness under heavy occlusion, SENTRY includes a motion-only fallback derived from a constant-acceleration Kalman filter. Using the center and scale of the previous refined mask $\mathcal{M}^{t-1}$, the filter predicts a new box center b_κ^t . A motion-derived mask $\mathcal{M}_\kappa^t$ is produced by translating $\mathcal{M}^{t-1}$ so its bounding box aligns with b_κ^t . Its shape is fixed since it is used only as a fallback. This prior is appended to the candidate pool, $\mathcal{C}^t = \hat{\mathcal{C}}^t \cup \{\mathcal{M}_\kappa^t\}$, but excluded from Soft-NMS, since its role is to provide a motion-only fallback rather than compete with appearance-derived proposals.

Neighbor Reasoning and Matching. Each candidate is evaluated through short-horizon temporal reasoning. SENTRY builds backward-propagated candidate tracklets, forward-propagated neighbor tracklets for recent distractors, and performs cycle-consistent matching to select the mask with the strongest temporal support. Neighbor-aware means that candidates whose trajectories align strongly with distractor motions are penalized during matching, reducing the chance of drift in cluttered scenes.

Candidate tracklet pool. A mask that appears plausible at frame t can still be inconsistent with the target’s recent motion. To evaluate temporal reliability, SENTRY builds a short backward tracklet for each candidate $\mathcal{M}_i^t \in \mathcal{C}^t$. The mask $\mathcal{M}_i^t$ is used as a dense prompt in SAM2’s promptable segmentation, and the previous τ frames ($I^{t-1}, \dots, I^{t-\tau}$) are re-segmented to obtain $\xi_i^t = (\mathcal{M}_i^t, \mathcal{M}_i^{t-1}, \dots, \mathcal{M}_i^{t-\tau})$. These newly generated backward segmentations form the candidate tracklet pool, $\mathbf{P}_c^t = \{\xi_i^t \mid \mathcal{M}_i^t \in \mathcal{C}^t\}$. For consistency evaluation, SENTRY maintains a reference trajectory $\eta^t = (\mathcal{M}^t, \dots, \mathcal{M}^{t-\tau})$ constructed from previously selected refined masks. This trajectory serves as a temporal anchor against which all candidate tracklets are evaluated. The temporal consistency length τ is fixed to 10 for all datasets, with runtime analysis in Sec. 4.4.

Neighbor tracklet pool. To provide context during ambiguity, SENTRY maintains a small set of distractor tracklets. This pool encodes *what the target is not*, giving contextual cues when candidates become ambiguous or visually similar. After selecting $\bar{\mathcal{M}}^t$, the remaining candidate tracklets from frame t are retained as neighbors (i.e., masks not chosen for update). Each neighbor is updated one step forward by re-segmenting I^{t+1} using its last mask as a prompt, keeping all sequences aligned within the τ -frame window. The resulting pool $\mathbf{P}_n^{t+1}$ serves as a negative motion context: candidates whose trajectories align with these distractors are penalized during cycle-consistent matching. The pool requires no explicit pruning; neighbor tracklets naturally expire once they fall outside the τ -frame window. This keeps the pool small, prevents unbounded growth, and avoids long-term drift.

Cycle-consistent matching. Given candidate and neighbor tracklets, SENTRY evaluates each hypothesis by comparing its backward-propagated trajectory against the forward-aligned target and neighbor tracklets over the same temporal window. Following the notion of cycle consistency used in [14], a candidate is considered reliable if tracking it backward produces a trajectory that overlaps well with the target’s recent forward track; conversely, candidates whose trajectories align more strongly with neighbor motion are treated as distractors. For every candidate tracklet $\xi_i^t \in \mathbf{P}_c^t$ and neighbor or target tracklet $\zeta_j^t \in \mathbf{P}_n^t \cup \{\eta^t\}$, we compute a trajectory similarity score:

$$\text{Sim}(\xi_i, \zeta_j) = \frac{1}{\tau} \sum_{k=t-\tau}^{t-1} \text{IoU}(b_i^k, b_j^k), \quad (4)$$

where b_i^k and b_j^k are the corresponding boxes at frame k . These similarities form a bipartite matrix over candidates and neighbors/target. A complete bipartite graph is constructed with weights $\text{Sim}(\xi_i, \zeta_j)$, and the Hungarian algorithm [48] is applied to obtain a one-to-one assignment that maximizes total trajectory agreement. This matching acts as a motion-consistency filter: candidates that are temporally aligned with the target are favored, while those that match neighbor trajectories are suppressed. Although final selection uses candidate–target similarity, the Hungarian assignment implicitly penalizes candidates aligned with distractor motion. This assignment serves purely as a consistency filter; the final mask is determined solely by candidate–target similarity s_i under the threshold rules.

Let $s_i = \text{Sim}(\xi_i^t, \eta^t)$ denote each candidate’s similarity to the target trajectory. We select the candidate with the highest s_i if it exceeds a reliability threshold θ_{rel} ($\max_i s_i \geq \theta_{\text{rel}}$). When no strong match exists but at least one candidate is temporally plausible ($\max_i s_i \geq \theta_{\text{min}}$), the highest-scoring candidate is used. If all candidates exhibit weak or inconsistent motion ($\max_i s_i < \theta_{\text{min}}$), SENTRY falls back to the Kalman prior $\mathcal{M}_K^t$, which provides a motion-only estimate during severe occlusion or appearance failure. The selected mask $\bar{\mathcal{M}}^t$ is then written to memory under SAM2’s default update schedule.

The thresholds θ_{rel} and θ_{min} are fixed for all datasets (see App. A). IoU-based trajectory similarity is used instead of appearance distance because appearance

cues become unreliable under occlusion, clutter, or lighting changes. Backward propagation uses each candidate mask as a segmentation prompt, so appearance information is already implicitly embedded in the resulting tracklet; using geometric motion as the comparison metric yields a more stable and robust signal [14].

Algorithm 1: SENTRY: Refine-Before-Write Tracking

Require: Video frames $\{I_t\}_{t=0}^T$; initial mask $\bar{\mathcal{M}}^0$; memory bank $\mathbf{Mem}^0$; image encoder $f(\cdot)$; sampled-point grid $\mathcal{G}$; backtrack horizon τ

- 1: **Init:**
- 2: $\mathbf{Mem}^0 \leftarrow \mathbf{Mem}^0 \cup \{\bar{\mathcal{M}}^0\}$
- 3: $\mathbf{P}_n^0 \leftarrow \emptyset$
- 4: **for** $(I^t) \in (I^T)$ **do:**
- 5: $F^t \leftarrow f(I^t)$
- 6: $\mathcal{C}_t^{\text{dec}} \leftarrow \text{DecodeCandidates}(F^t, \mathbf{Mem}^{t-1})$
- 7: $\mathcal{C}_{\text{AMG}}^t \leftarrow \text{DecodeFromGrid}(F^t, \mathcal{G})$
- 8: $\hat{\mathcal{C}}^t \leftarrow \text{Soft-NMS}(\mathcal{C}_t^{\text{dec}} \cup \mathcal{C}_{\text{AMG}}^t)$
- 9: $\mathcal{M}_\kappa^t \leftarrow \text{MaskFromKF}(\text{BBBox}(\bar{\mathcal{M}}^{t-1}))$
- 10: $\mathcal{C}^t \leftarrow \hat{\mathcal{C}}^t \cup \{\mathcal{M}_\kappa^t\}$
- 11: $\mathbf{P}_c^t \leftarrow \{\text{BacktrackPoints}(\mathcal{M}_i^t, \tau) \mid \mathcal{M}_i^t \in \mathcal{C}^t\}$
- 12: $\eta^t \leftarrow \text{CycleConsistentMatch}(\mathbf{P}_c^t, \mathbf{P}_n^{t-1})$
- 13: $\mathbf{P}_n^t \leftarrow \mathbf{P}_c^t \setminus \eta^t$
- 14: $\mathbf{Mem}^t \leftarrow \text{UpdateMemory}(\mathbf{Mem}^{t-1}, \eta^t)$
- 15: $\bar{\mathcal{M}}^t \leftarrow \text{SelectOutputMask}(\eta^t, \mathcal{M}_\kappa^t)$
- 16: **end for**

4 Experiments

We integrate SENTRY into five SAM2-based trackers: SAM2 [51], SAMURAI [71], DAM4SAM [58], SAMITE [66], and HiM2SAM [10] (the latter two are evaluated in App. K). We denote SENTRY applied to SAM2, SAMURAI, and DAM4SAM as SENTRY-S2, SENTRY-SR, and SENTRY-D4S. In App. D, we further report a small generality study on non-SAM memory trackers and other SAM-family trackers, suggesting that the benefit stems from verified memory writes rather than SAM2-specific internals. We evaluate all models across nine standard tracking benchmarks covering long-term and short-term bounding box tracking (LaSOT, LaSOT_{ext}, TNL2K, GOT-10k, TrackingNet) and the VOT/DiDi protocols. Each dataset’s official metrics are used. Comprehensive tables and per-scale results are provided in App. G and App. H, and ablation studies on scalability, candidate generation, and the temporal consistency window τ appear in Sec. 4.4. We additionally evaluate VOS benchmarks under the first-frame-mask protocol in App. I, where SENTRY improves each corresponding host tracker. For fair comparison, we reproduce any missing baseline results using each method’s official implementation under the same evaluation protocol. In the quantitative tables, the top three entries in each column are marked using **red** for best, **green** for second, and **blue** for third. Relative improvements of each SENTRY variant over its baseline are reported as percentage-point gains, shown in ($\uparrow$).

Table 1: SENTRY provides consistent improvements across baselines and datasets.

Method	LaSOT [23]			LaSOT _{ext} [22]			TNL2K [60]			GOT-10k [28]			TrackingNet [47]		
	S	NP	P	S	NP	P	S	NP	P	AO	SR _{0.50}	SR _{0.75}	S	NP	P
DiffusionTrack [63]	70.8	79.8	76.7	—	—	—	56.4	72.5	57.3	74.8	85.4	72.0	83.8	88.2	82.1
HIPTrack [7]	72.7	82.9	79.5	53.0	64.3	60.6	—	—	—	77.4	88.0	74.5	84.5	89.1	83.8
AQATrack ₂₅₆ [64]	71.4	81.9	78.6	51.2	62.2	58.9	57.8	59.4	—	73.8	83.2	72.1	83.8	88.6	83.1
ARPTTrack ₂₅₆ [38]	72.6	81.4	78.5	52.0	62.9	58.7	—	—	—	77.7	87.3	74.3	85.5	90.0	85.3
SPMTTrack-B [8]	74.9	84.0	81.7	—	—	—	62.0	79.7	66.7	76.5	85.9	76.3	86.1	90.2	85.6
UVLTrack-B [43]	69.4	—	74.9	49.2	—	55.8	62.7	—	65.4	—	—	—	83.4	—	82.1
QueryNLT [56]	59.9	69.6	63.5	—	—	—	57.8	75.6	58.7	—	—	—	—	—	—
DUTrack ₃₈₄ [36]	74.1	84.9	82.9	52.5	63.6	60.5	65.6	83.2	71.9	77.8	—	—	—	—	—
ManoNLT [41]	66.6	77.3	71.0	—	—	—	66.5	90.9	69.9	—	—	—	—	—	—
CLDTracker [2]	74.0	83.9	81.1	53.1	64.8	60.6	61.5	82.2	64.3	77.5	85.4	75.6	85.1	89.7	84.9
MemNLT [24]	72.9	85.7	80.5	52.1	63.3	59.8	63.3	80.9	67.4	—	—	—	—	—	—
RTracker-L [29]	74.7	84.5	—	54.9	65.5	62.7	60.6	—	63.7	77.9	87.0	76.9	—	—	—
Zero-shot Method															
SAM2-L [51]	68.5	76.1	73.6	56.8	71.1	67.0	56.7	75.4	62.5	80.8	91.3	75.5	85.3	91.3	88.2
SAMURAI-L [71]	74.2	82.7	80.2	61.0	73.9	72.2	50.6	67.5	54.2	81.7	92.2	76.9	85.3	88.2	85.0
DAM4SAM-L [58]	75.1	83.3	81.1	60.9	75.3	72.2	59.8	79.8	66.8	81.1	91.4	77.2	85.3	90.9	87.4
SENTRY-S2L	70.2 (1.7)	77.2 (1.1)	74.5 (0.9)	57.0 (0.2)	71.7 (0.6)	67.1 (0.1)	57.9 (1.2)	76.9 (1.5)	64.1 (1.6)	81.1 (0.3)	91.4 (0.1)	76.5 (1.0)	85.7 (0.4)	91.9 (0.6)	88.9 (0.7)
SENTRY-SRL	75.1 (0.9)	82.7 (0.4)	80.4 (0.2)	61.5 (0.5)	75.0 (1.1)	72.9 (0.7)	59.6 (9.0)	78.8 (11.3)	66.4 (12.2)	81.8 (0.1)	92.3 (0.1)	77.1 (0.2)	85.8 (0.5)	91.1 (2.9)	88.1 (3.1)
SENTRY-D4S-L	76.3 (1.2)	84.7 (1.4)	82.4 (1.3)	61.8 (0.9)	76.6 (1.3)	73.8 (1.6)	61.3 (1.5)	81.3 (1.5)	68.3 (1.5)	82.1 (1.0)	92.6 (1.2)	78.2 (1.0)	85.9 (0.6)	91.5 (0.7)	87.9 (0.5)

4.1 SOTA comparison on box benchmarks

Results on LaSOT, LaSOT_{ext}, TNL2K, GOT-10k, and TrackingNet are summarized in Tab. 1. Full results across all trackers and model scales appear in App. G (Tab. 19). Attribute-level analyses are shown in Figs. 4(a)–(c) and in App. Tabs. 20, 21, and 22. SENTRY’s gains follow each dataset’s difficulty profile: smaller improvements on short or clean sequences (TrackingNet), moderate improvements on heterogeneous or appearance-diverse benchmarks (GOT-10k, LaSOT_{ext}), and larger improvements on long or rapidly changing sequences where baseline errors accumulate more (LaSOT, TNL2K).

LaSOT [23]. All SENTRY variants outperform their SAM2-based baselines across all metrics. SENTRY-D4S gains 1.6% S, 1.7% NP, and 1.6% P over DAM4SAM; SENTRY-SR improves SAMURAI by 1.2% S and 0.2% P; and SENTRY-S2 surpasses SAM2 by 2.5% S, 1.4% NP, and 1.2% P. LaSOT’s long sequences amplify small mask errors into drift, making it a setting where SENTRY’s stability-focused updates yield substantial improvements. **LaSOT_{ext} [22].** SENTRY-D4S sets a new SOTA on LaSOT_{ext}, improving over DAM4SAM by 1.5% S, 1.7% NP, and 2.2% P. Other variants show consistent gains: SENTRY-S2 adds 0.4% S, 0.8% NP, and 0.1% P, while SENTRY-SR improves SAMURAI by 0.8% S, 1.5% NP, and 1.0% P. LaSOT_{ext} contains many small objects and unseen categories, which frequently destabilize baseline predictions; SENTRY mitigates these cases, resulting in consistent gains. **TNL2K [60].** All SENTRY variants outperform their SAM2-based baselines. SENTRY-D4S gains 2.5% S, 1.9% NP, and 2.2% P over DAM4SAM; SENTRY-S2 adds 2.1% S, 2.0% NP, and 2.6% P; and SENTRY-SR yields the largest boost of 17.8% S, 16.7% NP, and 22.5% P over SAMURAI. TNL2K contains frequent appearance shifts, occlusions, and strongly non-linear motion, which disrupt SAMURAI’s Kalman filter-based linear update model; in contrast, SENTRY’s refine-before-write mechanism validates each memory update via temporal consistency rather than motion assumptions, yielding its largest gains specifically on this benchmark. **GOT-10k [28].** SENTRY-D4S achieves the best overall performance and sets a new SOTA. SENTRY-SR ranks second on AO and SR_{0.5}, and third on SR_{0.75}. Operating fully zero-shot, SENTRY-D4S even outperforms fine-tuned competitors, highlighting the scalability of reliability-aware refinement. Rela-

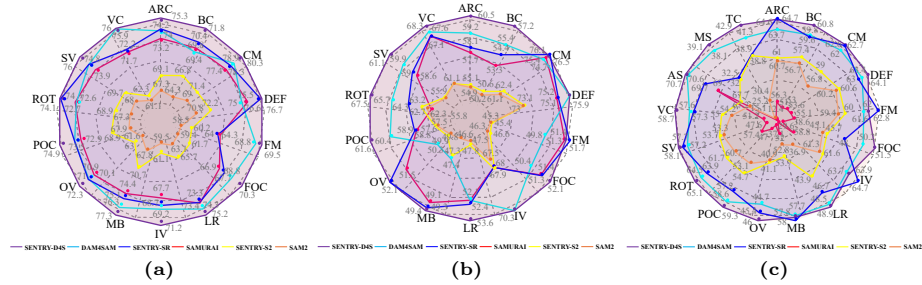


Fig. 4: AUC of attributes on (a) LaSOT [23], (b) LaSOT_{ext} [22], and (c) TNL2K [60].

tive gains over baselines include 1.2% AO, 1.3% SR_{0.5}, and 1.3% SR_{0.75} for DAM4SAM; 0.4%, 0.1%, and 1.3% for SAM2; and 0.1%, 0.1%, and 0.3% for SAMURAI. GOT-10k’s category and appearance diversity induces mild inconsistencies that SENTRY stabilizes. **TrackingNet** [47]. All SENTRY variants improve over their respective baselines. Integrating SENTRY into DAM4SAM yields 0.7% S, 0.7% NP, and 0.6% P; applying it to SAM2 adds 0.5% S, 0.7% NP, and 0.8% P; and enhancing SAMURAI provides the largest gains of 0.6% S, 3.3% NP, and 3.6% P. TrackingNet’s short, clean sequences leave limited room for correction, so SENTRY primarily smooths small prediction fluctuations, producing modest but consistent gains. **Attribute-Wise Performance.** Across all datasets (Figs. 4(a)–(c)), SENTRY delivers its strongest gains in conditions that destabilize per-frame masks, such as BC, motion changes, LR, and occlusions. These patterns are consistent across SAM2, SAMURAI, and DAM4SAM, reflecting the shared sensitivity of SAM2-style memory updates to noisy frames. Differences emerge at the dataset level. On LaSOT, the long sequences magnify issues like MB and CM, making drift suppression the dominant source of improvement. On LaSOT_{ext}, many targets are small or visually scarce, so gains are driven by better handling of LR views, weak cues, and occlusions. On TNL2K, sudden appearance changes and non-linear motion are the main challenges, and SENTRY’s filtering stabilizes these rapid transitions. Across benchmarks, SENTRY consistently removes unreliable updates, but the specific attributes that benefit most align with each dataset’s characteristic failure modes.

4.2 SOTA comparison on VOT benchmarks

We evaluate SENTRY on VOT20 [32], VOT22 [31], and VOTS24 [33], with results in Tabs. 2–4 and Figs. 5(a)–(c). We also assess DiDi [58], which focuses on visually similar non-negligible distractors, with results in Tab. 5 and Fig. 5(d). Additional results for model scales and baselines are in Appendices H and K. **VOT20** [32]. All SENTRY variants outperform their SAM2-based baselines, with relative gains up to 1.2% in *Q*, 3.4% in *Acc*, and 0.7% in *Rob*. Most VOT20 failures stem from unstable masks during fast motion and occlusions, which trigger resets; SENTRY filters these unreliable updates, reducing reset events and improving all metrics. SENTRY-D4S obtains the highest scores, setting a new SOTA. **VOT22** [31]. All SENTRY variants again improve over their baselines, with gains up to 2.1% in *Q*, 3.1% in *Acc*, and 0.8% in *Rob*. VOT22 includes

Table 2: Comparison on the VOT20 [32].

Method	VOT20		
	Q	Acc	Rob
ODTrack [77]	0.605	0.761	0.902
MixViT-L+AR [19]	0.584	0.755	0.890
SeqTrack-L [11]	0.561	-	-
MixFormer-L [18]	0.555	0.762	0.855
RPT [44]	0.530	0.700	0.869
OceanPlus [76]	0.491	0.685	0.842
AlphaRef [70]	0.482	0.754	0.777
AFOD [13]	0.472	0.713	0.795
<i>Zero-shot Method</i>			
SAM2-L [51]	0.681	0.778	0.941
SAMURAI-L [71]	0.674	0.741	0.942
DAM4SAM-L [58]	0.723	0.796	0.961
SENTRY-S2-L	0.690 (0.9↑)	0.785 (0.7↑)	0.949 (0.8↑)
SENTRY-SR-L	0.686 (1.2↑)	0.775 (3.4↑)	0.947 (0.5↑)
SENTRY-D4S-L	0.732 (0.9↑)	0.804 (0.8↑)	0.970 (0.9↑)

Table 4: Comparison on the VOTS24 [33].

Method	VOTS24 [33]		
	Q	Acc	Rob
S3-Track [37]	0.722	0.784	0.889
DMAOT-SAM [74]	0.653	0.794	0.780
HQ-DMAOT [74]	0.639	0.754	0.790
DMAOT [74]	0.636	0.751	0.795
LY-SAM [33]	0.631	0.765	0.776
Cutie-SAM [15]	0.607	0.756	0.730
AOT [72]	0.550	0.698	0.767
LORAT [39]	0.536	0.725	0.784
<i>Zero-shot Method</i>			
SAM2-L [51]	0.661	0.791	0.790
SAMURAI-L [71]	0.673	0.776	0.851
DAM4SAM-L [58]	0.711	0.793	0.864
SENTRY-S2-L	0.669 (0.8↑)	0.793 (0.2↑)	0.795 (0.5↑)
SENTRY-SR-L	0.681 (0.8↑)	0.782 (0.6↑)	0.856 (0.5↑)
SENTRY-D4S-L	0.715 (0.4↑)	0.794 (0.1↑)	0.867 (0.3↑)

Table 3: Comparison on the VOT22 [31].

Method	VOT22		
	Q	Acc	Rob
MS-AOT [72]	0.673	0.781	0.944
DiffusionTrack [63]	0.634	-	-
DAMTMask [31]	0.624	0.796	0.891
MixFormer-M [18]	0.589	0.799	0.878
OSTrackSTS [75]	0.581	0.775	0.867
Linker [68]	0.559	0.772	0.861
SRATransTS [31]	0.547	0.743	0.866
TransT_M [12]	0.542	0.743	0.865
GDFormer [31]	0.538	0.744	0.861
TransLL [31]	0.530	0.735	0.861
LWL_B2S [4]	0.516	0.736	0.831
D3Sv2 [42]	0.497	0.713	0.827
<i>Zero-shot Method</i>			
SAM2-L [51]	0.692	0.779	0.946
SAMURAI-L [71]	0.693	0.744	0.951
DAM4SAM-L [58]	0.750	0.797	0.971
SENTRY-S2-L	0.701 (0.9↑)	0.787 (0.8↑)	0.953 (0.7↑)
SENTRY-SR-L	0.715 (2.1↑)	0.775 (3.1↑)	0.957 (0.6↑)
SENTRY-D4S-L	0.759 (0.9↑)	0.805 (0.8↑)	0.979 (0.8↑)

Table 5: Comparison on the DiDi [58].

Method	DiDi [58]		
	Q	Acc	Rob
ODTrack [77]	0.608	0.740	0.809
Cutie [15]	0.575	0.704	0.776
AOT [72]	0.541	0.622	0.852
AQATrack [64]	0.535	0.693	0.753
SeqTrack [11]	0.529	0.714	0.718
KeyTrack [45]	0.502	0.646	0.748
TransT [12]	0.465	0.669	0.678
<i>Zero-shot Method</i>			
SAM2Long [21]	0.646	0.719	0.883
SAM2-L [51]	0.649	0.720	0.887
SAMURAI-L [71]	0.680	0.722	0.930
DAM4SAM-L [58]	0.694	0.727	0.944
SENTRY-S2-L	0.660 (1.1↑)	0.729 (0.9↑)	0.897 (1.0↑)
SENTRY-SR-L	0.693 (1.3↑)	0.734 (1.2↑)	0.943 (1.3↑)
SENTRY-D4S-L	0.705 (1.1↑)	0.737 (1.0↑)	0.956 (1.2↑)

stronger appearance changes and heavier distractor pressure than VOT20, increasing the risk of incorrect memory writes; SENTRY suppresses these errors and yields consistent improvements across all methods. SENTRY-D4S again sets a new SOTA. **VOTS24 [33]**. SENTRY variants surpass their baselines, with gains up to 0.8% in *Q*, 0.6% in *Acc*, and 0.5% in *Rob*. VOTS24’s long-term protocol exposes trackers to gradual drift and extended occlusion periods, where errors accumulate. SENTRY reduces these accumulation points and provides steady performance gains. SENTRY-D4S attains the highest *Acc* and ranks second in *Q* and *Rob*, closely trailing the competition-tuned S3-Track [37] while outperforming all zero-shot and several supervised competitors. **DiDi [58]**. SENTRY variants improve over their baselines, achieving gains up to 1.3% in *Q*, 1.2% in *Acc*, and 1.3% in *Rob*. DiDi is explicitly designed to induce failures through dense distractors, where baseline trackers often commit erroneous memory updates. SENTRY mitigates these errors and stabilizes mask updates across all variants, with SENTRY-D4S achieving the highest overall scores and setting a new SOTA.

4.3 Qualitative Results

Tracking Result. Fig. 6 shows how SENTRY mitigates key failure modes of existing SAM2-based variants. Row 1: SAM2 loses identity after heavy occlusion due to ambiguous mask selection, whereas SENTRY-S2 disambiguates and refines the correct mask before memory writing. Row 2: under abrupt scene change, SAMURAI’s Kalman prior diverges, while SENTRY-SR restores identity through neighbor-guided matching. Row 3: In an underwater scene with haze and distractors, DAM4SAM mis-selects background objects due to heuristic thresholds, while SENTRY-D4S chooses the candidate maintaining short-

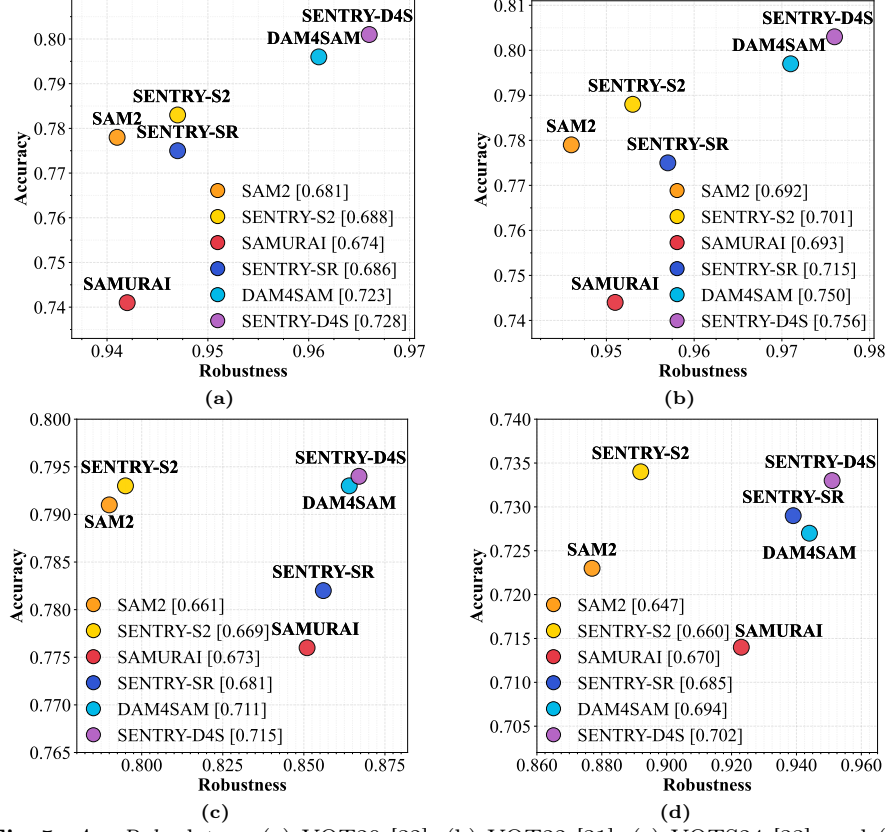


Fig. 5: *Acc-Rob* plot on (a) VOT20 [32], (b) VOT22 [31], (c) VOTS24 [33], and (d) DiDi [58] for SAM2 and SENTRY variants. The Q is given at each label.

term trajectory coherence. Row 4: after an extreme long-term occlusion (≈ 100 frames), all methods fail to recover the tiny target, highlighting persistent re-identification limits.

4.4 Ablation Study

Scalability across SAM2 sizes and base frameworks. We integrate SENTRY across four SAM2 model scales (T/S/B/L). Tab. 10 summarizes average S improvements over five benchmarks, with full per-metric, per-scale results in Appendices G and H (Tabs. 19 and 23). SENTRY improves all base frameworks and remains largely backbone-agnostic: SAM2 gains 0.6–0.8 across scales, indicating memory reliability rather than capacity is the main limitation; SAMURAI benefits more strongly (3.1–5.4 on average, with +8.8/+13.8/+11.8/+9.0 on TNL2K) due to SENTRY removing failures caused by its linear-motion update model. DAM4SAM gains 1.0–2.8 on average, reflecting improved stability in distractor-heavy settings. Notably, the S-scale variant is consistently the weakest among DAM4SAM models, indicating stronger dependence on feature capacity; SENTRY compensates by filtering low-confidence distractor-aligned updates before

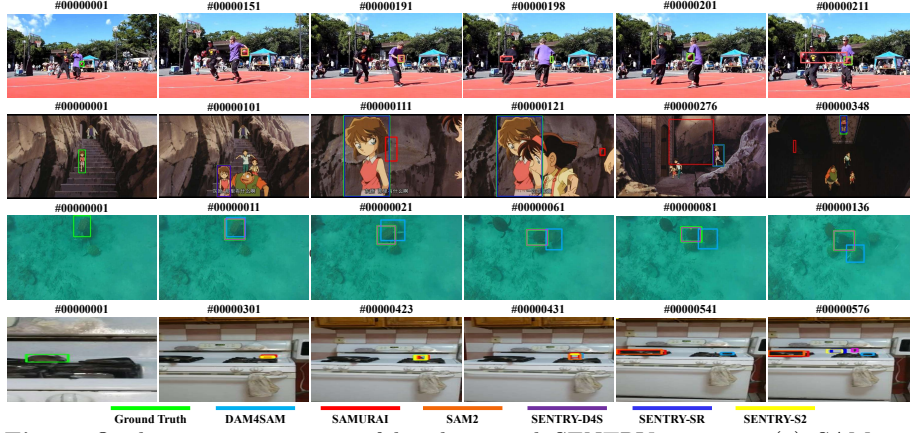


Fig. 6: Qualitative comparison of baselines and SENTRY variants: (1) SAM2 vs. SENTRY-S2, (2) SAMURAI vs. SENTRY-SR, (3) DAM4SAM vs. SENTRY-D4S. (4) Failure case: all methods lose the target after a long out-of-view period (100 frames) with a tiny target.

they enter memory. **Candidate mask generation.** Tab. 6 compares candidate-generation strategies within SENTRY. Detection-segmentation pipelines such as GroundingDINO [40] with SAM/SAM2 [30, 51] produce spatially precise but frame-independent masks, resulting in poor temporal consistency. Conversely, SAM2’s native propagation preserves identity but offers no mechanism for correcting drift once it accumulates. Building on SAM2, we introduce four complementary modules evaluated under the same selection policy: decoder hypotheses ($\mathcal{C}_{\text{dec}}^t$) to retain multiple plausible masks, AMG proposals ($\mathcal{C}_{\text{AMG}}^t$) for expanded spatial coverage, Soft-NMS to reduce redundancy among overlapping hypotheses, and a Kalman prior ($\mathcal{M}_{\kappa}^t$) to stabilize motion. As shown in Tab. 6, each addition yields incremental S gains of 0.5, 0.1, 0.1, and 0.1, leading to a 0.9% average improvement over SAM2 across benchmarks. **Temporal consistency length τ .** We analyze the effect of varying the temporal window τ , which defines how many past frames contribute to matching. As shown in Tab. 7, increasing τ improves robustness by using richer appearance and motion context, but overly long windows may introduce outdated features and temporal noise. Performance saturates around $\tau=10$. **Runtime analysis.** We evaluate runtime on an NVIDIA A100 GPU. SENTRY-S2-L achieves 32.8 FPS compared to 44.0 FPS for SAM2-L (Tab. 9), adding only $\sim 7\text{--}8$ ms per frame ($\sim 25\%$ overhead) while retaining real-time performance. Across SENTRY-compatible large-model hosts, the supplementary runtime and memory study reports a 16.6–25.5% FPS drop and 0.4–0.6 GB additional VRAM (App. J). The overhead comes from three components: (i) candidate construction (AMG/decoder aggregation, template similarity, and Soft-NMS), (ii) short-horizon reasoning (backward re-segmentation and Hungarian matching), and (iii) Kalman prior. Removing the Kalman prior or AMG/decoder inputs reduces candidate load; disabling Soft-NMS increases latency due to redundant proposals; and w/o neighbor pool restricts matching to the target trajectory only (no distractor modeling). **Neighbor tracklet**

pool. Removing the neighbor tracklet pool $\mathbf{P}_n^t$ reduces matching to target-only consistency checks, ignoring distractor motion. This causes drift under clutter. Incorporating NP provides contextual negatives, penalizing candidates aligned with distractor trajectories and yielding temporal stability (Tab. 8). **Scalabil-**

Table 6: Incremental contribution of candidate-generation modules on [22,23,28,60]. **Table 7:** Effect of τ on performance. **Table 8:** Effect of $\mathbf{P}_n^t$ on [23]. Δ denotes average S gain over the SAM2-L.

Method / Component	LaSOT	LaSOT _{ext}	TNL2K	GOT-10k	Δ
SAM2 (Baseline) [51]	68.5	56.8	56.7	80.8	-
Candidate Mask Generation Baselines					
GroundingDINO & SAM [30,40]	68.2	56.4	56.4	80.5	0.3↓
GroundingDINO & SAM2 [40,51]	69.0	56.7	56.6	80.9	0.1↑
SAM [30]	69.0	56.8	57.0	80.9	0.2↑
Incremental Modules on SAM2					
+ C_{dec}	69.8	57.0	57.5	80.9	0.6↑
+ C_{AMG}	69.9	57.0	57.8	80.9	0.7↑
+ Soft-NMS	70.0	57.0	57.8	81.0	0.8↑
+ $\mathcal{M}^c$	70.2	57.0	57.9	81.1	0.9↑

τ	LaSOT	LaSOT _{ext}	TNL2K
1	68.5	56.8	56.7
3	69.1	56.8	57.1
5	69.9	56.9	57.5
10	70.2	57.0	57.9
20	70.0	56.9	57.7
30	69.2	56.8	57.7

	w/o $\mathbf{P}_n^t$
S	65.8
NP	73.1
P	70.4
with $\mathbf{P}_n^t$	
S	70.2
NP	77.2
P	74.5

Table 9: Runtime analysis.

Variant	FPS ↑	Overhead (%) ↓
SAM2-L	44.0	
SENTRY-S2-L	32.8	25.5
w/o Kalman prior	34.7	21.1
w/o AMG proposals	35.4	19.5
w/o decoder hypotheses	36.9	16.1
w/o neighbor pool	38.7	12.0
w/o candidate filtration	30.1	31.6

ity. Across all scales (T/S/B/L) and all three base trackers, SENTRY consistently improves robustness (Tab. 10). Gains hold on both short- and long-term benchmarks, despite differences in dataset difficulty and model capacity. SAM2 benefits most at small and large scales; SAMURAI sees the strongest overall lift, especially on TNL2K and TrackingNet; and DAM4SAM shows targeted gains, with S-scale improving notably on GOT-10k and TrackingNet. Fig. 7 corroborates this.

Table 10: Average relative improvements (Δ) of SENTRY over each base framework across model sizes (T/S/B/L). Mean Δ across all bases: SAM2 (0.7), SAMURAI (4.5), DAM4SAM (1.5). Full per-metric results are provided in App. G, Tab. 19.

(a) SAM2 → SENTRY					(b) SAMURAI → SENTRY					(c) DAM4SAM → SENTRY				
Dataset	T	S	B	L	Dataset	T	S	B	L	Dataset	T	S	B	L
LaSOT	0.6↑	1.5↑	1.0↑	1.7↑	LaSOT	2.6↑	1.4↑	2.5↑	0.9↑	LaSOT	1.1↑	1.1↑	2.0↑	1.2↑
LaSOT _{ext}	0.7↑	1.0↑	0.7↑	0.2↑	LaSOT _{ext}	2.0↑	2.4↑	2.4↑	2.0↑	LaSOT _{ext}	1.0↑	1.5↑	2.4↑	1.1↑
TNL2K	1.2↑	1.2↑	1.3↑	1.3↑	TNL2K	8.8↑	13.8↑	11.8↑	9.0↑	TNL2K	1.7↑	1.4↑	1.5↑	1.5↑
GOT-10k	0.3↑	0.1↑	0.1↑	0.3↑	GOT-10k	2.1↑	2.4↑	2.1↑	0.2↑	GOT-10k	0.7↑	5.3↑	0.3↑	0.8↑
TrackingNet	0.3↑	0.4↑	0.4↑	0.4↑	TrackingNet	6.3↑	7.0↑	5.7↑	3.4↑	TrackingNet	0.3↑	4.9↑	0.1↑	0.6↑
Δ	0.6↑	0.8↑	0.7↑	0.8↑	Δ	4.4↑	5.4↑	4.9↑	3.1↑	Δ	1.0↑	2.8↑	1.3↑	1.0↑

5 Conclusion

We revisited memory update mechanisms in SAM2-based VOT and exposed confidence-only mask selection as a key source of drift under occlusion, motion, and distractors. We introduced SENTRY, a training-free, plug-and-play, refine-before-write module that validates temporal consistency before committing memory updates. By replacing confidence-driven writes with consistency-validated ones, SENTRY stabilizes tracking without retraining or architectural changes and consistently improves all-scale evaluation performance across baselines and benchmarks.

Acknowledgements

This work was supported in part by the Khalifa University Center for Autonomous Robotic Systems (KUCARS) under Award RC1-2018-KUCARS; in part by Silal Innovation Oasis through projects under grants 8475000023, 8475000024, 8475000025, and 8475000026; and in part by Khalifa University of Science and Technology through the Faculty Start-Ups under Project ID KU-INT-FSU-2005-8474000775.

References

1. Alansari, M., Abdul Hay, O., Alansari, S., Javed, S., Shoufan, A., Zweiri, Y., Werghi, N.: Drone-person tracking in uniform appearance crowd: A new dataset. *Scientific Data* **11**(1), 15 (2024)
2. Alansari, M., Javed, S., Ganapathi, I.I., Alansari, S., Naseer, M.: Cldtracker: A comprehensive language description for visual tracking. *Information Fusion* **124**, 103374 (2025). <https://doi.org/https://doi.org/10.1016/j.inffus.2025.103374>, <https://www.sciencedirect.com/science/article/pii/S1566253525004476>
3. Asanomi, T., Nishimura, K., Bise, R.: Multi-frame attention with feature-level warping for drone crowd tracking. In: 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 1664–1673 (2023). <https://doi.org/10.1109/WACV56688.2023.00171>
4. Bhat, G., Lawin, F.J., Danelljan, M., Robinson, A., Felsberg, M., Van Gool, L., Timofte, R.: Learning what to learn for video object segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) *Computer Vision – ECCV 2020*. pp. 777–794. Springer International Publishing, Cham (2020)
5. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-nms – improving object detection with one line of code. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Oct 2017)
6. Cai, J., Xu, M., Li, W., Xiong, Y., Xia, W., Tu, Z., Soatto, S.: Memot: Multi-object tracking with memory. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 8090–8100 (June 2022)
7. Cai, W., Liu, Q., Wang, Y.: Hiptrack: Visual tracking with historical prompts. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 19258–19267 (June 2024)
8. Cai, W., Liu, Q., Wang, Y.: 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 (CVPR)*. pp. 16871–16881 (June 2025)
9. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Coll-Vinent, D.S., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., HAZRA, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollar, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: SAM 3: Segment anything with concepts. In: *The Fourteenth International Conference on Learning Representations* (2026), <https://openreview.net/forum?id=r35clVtGzw>

10. Chen, R., Sun, G., Li, Y., Qin, J., Benini, L.: Him2sam: Enhancing sam2 with hierarchical motion estimation and memory optimization towards long-term tracking. In: Kittler, J., Xiong, H., Yang, J., Chen, X., Lu, J., Lin, W., Yu, J., Zheng, W. (eds.) *Pattern Recognition and Computer Vision*. pp. 276–291. Springer Nature Singapore, Singapore (2026)
11. Chen, X., Peng, H., Wang, D., Lu, H., Hu, H.: Seqtrack: Sequence to sequence learning for visual object tracking. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 14572–14581 (June 2023)
12. Chen, X., Yan, B., Zhu, J., Wang, D., Yang, X., Lu, H.: Transformer tracking. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 8126–8135 (2021)
13. Chen, Y., Xu, J., Yu, J., Wang, Q., Yoo, B., Han, J.J.: Afod: Adaptive focused discriminative segmentation tracker. In: Bartoli, A., Fusiello, A. (eds.) *Computer Vision – ECCV 2020 Workshops*. pp. 666–682. Springer International Publishing, Cham (2020)
14. Chen, Y.H., Wang, C.Y., Yang, C.Y., Chang, H.S., Lin, Y.L., Chuang, Y.Y., Liao, H.Y.M.: Neighbortrack: Single object tracking by bipartite matching with neighbor tracklets and its applications to sports. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. pp. 5139–5148 (2023). <https://doi.org/10.1109/CVPRW59228.2023.00542>
15. Cheng, H.K., Oh, S.W., Price, B., Lee, J.Y., Schwing, A.: Putting the object back into video object segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 3151–3161 (June 2024)
16. Cheng, H.K., Schwing, A.G.: Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) *Computer Vision – ECCV 2022*. pp. 640–658. Springer Nature Switzerland, Cham (2022)
17. Cheng, H.K., Tai, Y.W., Tang, C.K.: Rethinking space-time networks with improved memory coverage for efficient video object segmentation. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) *Advances in Neural Information Processing Systems*. vol. 34, pp. 11781–11794. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/61b4a64be663682e8cb037d9719ad8cd-Paper.pdf
18. Cui, Y., Jiang, C., Wang, L., Wu, G.: Mixformer: End-to-end tracking with iterative mixed attention. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 13608–13618 (2022)
19. Cui, Y., Jiang, C., Wu, G., Wang, L.: Mixformer: End-to-end tracking with iterative mixed attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **46**(6), 4129–4146 (2024). <https://doi.org/10.1109/TPAMI.2024.3349519>
20. Ding, H., Liu, C., He, S., Jiang, X., Torr, P.H., Bai, S.: Mose: A new dataset for video object segmentation in complex scenes. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. pp. 20167–20177 (2023). <https://doi.org/10.1109/ICCV51070.2023.01850>
21. Ding, S., Qian, R., Dong, X., Zhang, P., Zang, Y., Cao, Y., Guo, Y., Lin, D., Wang, J.: Sam2long: Enhancing sam 2 for long video segmentation with a training-free memory tree. *arXiv preprint arXiv:2410.16268* (2024)
22. Fan, H., Bai, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Harshit, Huang, M., Liu, J., et al.: Lasot: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision* **129**(2), 439–461 (2021)

23. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C., Ling, H.: Lasot: A high-quality benchmark for large-scale single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
24. Feng, X., Li, X., Hu, S., Zhang, D., Wu, M., Zhang, J., Chen, X., Huang, K.: Memvlt: Vision-language tracking with adaptive memory-based prompts. In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) Advances in Neural Information Processing Systems. vol. 37, pp. 14903–14933. Curran Associates, Inc. (2024), https://proceedings.neurips.cc/paper_files/paper/2024/file/1af3e0bf5905e33789979f666c31192d-Paper-Conference.pdf
25. Fu, Z., Liu, Q., Fu, Z., Wang, Y.: Stmtrack: Template-free visual tracking with space-time memory networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13774–13783 (June 2021)
26. Hong, L., Chen, W., Liu, Z., Zhang, W., Guo, P., Chen, Z., Zhang, W.: Lvos: A benchmark for long-term video object segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 13480–13492 (October 2023)
27. Hong, L., Liu, Z., Chen, W., Tan, C., Feng, Y., Zhou, X., Guo, P., Li, J., Chen, Z., Gao, S., Zhang, W., Zhang, W.: Lvos: A benchmark for large-scale long-term video object segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **48**(1), 946–961 (2026). <https://doi.org/10.1109/TPAMI.2025.3611020>
28. Huang, L., Zhao, X., Huang, K.: 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 (2021)
29. Huang, Y., Li, X., Zhou, Z., Wang, Y., He, Z., Yang, M.H.: Rtracker: Recoverable tracking via pn tree structured memory. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19038–19047 (2024). <https://doi.org/10.1109/CVPR52733.2024.01801>
30. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollar, P., Girshick, R.: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4015–4026 (October 2023)
31. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Kämäräinen, J.K., Chang, H.J., Danelljan, M., Zajc, L.Č., Lukežič, A., Drbohlav, O., Björklund, J., Zhang, Y., Zhang, Z., Yan, S., Yang, W., Cai, D., Mayer, C., Fernández, G., Ben, K., Bhat, G., Chang, H., Chen, G., Chen, J., Chen, S., Chen, X., Chen, X., Chen, X., Chen, Y., Chen, Y.H., Chen, Z., Cheng, Y., Ciaramella, A., Cui, Y., Džubur, B., Dasari, M.M., Deng, Q., Dhar, D., Di, S., Nardo, E.D., Du, D.K., Dunnhofer, M., Fan, H., Feng, Z., Fu, Z., Gao, S., Gorthi, R.K., Granger, E., Gu, Q.H., Gupta, H., He, J., He, K., Huang, Y., Jangid, D., Ji, R., Jiang, C., Jiang, Y., Lawin, F.J., Kang, Z., Kiran, M., Kittler, J., Lai, S., Lan, X., Lee, D., Lee, H., Lee, S., Li, H., Li, M., Li, W., Li, X., Li, X., Li, X., Li, Z., Lin, L., Ling, H., Liu, B., Liu, C., Liu, S., Lu, H., Cruz, R.M.O., Ma, B., Ma, C., Ma, J., Ma, Y., Martinel, N., Memarmoghadam, A., Micheloni, C., Moallem, P., Nguyen-Meidine, L.T., Pan, S., Park, C., Paudel, D., Paul, M., Peng, H., Robinson, A., Rout, L., Shan, S., Simonato, K., Song, T., Song, X., Sun, C., Sun, J., Tang, Z., Timofte, R., Tsai, C.Y., Gool, L.V., Verma, O.P., Wang, D., Wang, F., Wang, L., Wang, L., Wang, L., Wang, L., Wang, Q., Wu, G., Wu, J., Wu, X., Xie, F., Xu, T., Xu, W., Xu, Y., Xu, Y., Xue, W., Xun, Z., Yan, B., Yang, D., Yang, J., Yang, W., Yang,

- X., Yang, Y., Yang, Y., Yang, Z., Ye, B., Yu, F., Yu, H., Yu, J., Yu, Q., Yu, W., Ze, K., Zhai, J., Zhang, C., Zhang, C., Zhang, K., Zhang, T., Zhang, W., Zhang, Z., Zhang, Z., Zhao, J., Zhao, S., Zheng, F., Zheng, H., Zheng, M., Zhong, B., Zhu, J., Zhu, X., Zhuang, Y.: The tenth visual object tracking vot2022 challenge results. In: Karlinsky, L., Michaeli, T., Nishino, K. (eds.) *Computer Vision – ECCV 2022 Workshops*. pp. 431–460. Springer Nature Switzerland, Cham (2023)
32. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R., Kämäräinen, J.K., Danelljan, M., Zajc, L.Č., Lukežič, A., Drbohlav, O., He, L., Zhang, Y., Yan, S., Yang, J., Fernández, G., Hauptmann, A., Memarmoghadam, A., García-Martín, Á., Robinson, A., Varfolomeiev, A., Gebrehiwot, A.H., Uzun, B., Yan, B., Li, B., Qian, C., Tsai, C.Y., Micheloni, C., Wang, D., Wang, F., Xie, F., Lawin, F.J., Gustafsson, F., Foresti, G.L., Bhat, G., Chen, G., Ling, H., Zhang, H., Cevikalp, H., Zhao, H., Bai, H., Kuchibhotla, H.C., Saribas, H., Fan, H., Ghanei-Yakhdan, H., Li, H., Peng, H., Lu, H., Li, H., Khaghani, J., Bescos, J., Li, J., Fu, J., Yu, J., Xu, J., Kittler, J., Yin, J., Lee, J., Yu, K., Liu, K., Yang, K., Dai, K., Cheng, L., Zhang, L., Wang, L., Wang, L., Van Gool, L., Bertinetto, L., Dunnhofer, M., Cheng, M., Dasari, M.M., Wang, N., Wang, N., Zhang, P., Torr, P.H.S., Wang, Q., Timofte, R., Gorthi, R.K.S., Choi, S., Marvasti-Zadeh, S.M., Zhao, S., Kasaei, S., Qiu, S., Chen, S., Schön, T.B., Xu, T., Lu, W., Hu, W., Zhou, W., Qiu, X., Ke, X., Wu, X.J., Zhang, X., Yang, X., Zhu, X., Jiang, Y., Wang, Y., Chen, Y., Ye, Y., Li, Y., Yao, Y., Lee, Y., Gu, Y., Wang, Z., Tang, Z., Feng, Z.H., Mai, Z., Zhang, Z., Wu, Z., Ma, Z.: The eighth visual object tracking vot2020 challenge results. In: Bartoli, A., Fusiello, A. (eds.) *Computer Vision – ECCV 2020 Workshops*. pp. 547–601. Springer International Publishing, Cham (2020)
 33. Kristan, M., Matas, J., Tokmakov, P., Felsberg, M., Zajc, L.Č., Lukežič, A., Tran, K.T., Vu, X.S., Björklund, J., Chang, H.J., Fernández, G., Attari, M., Chan, A., Chen, L., Chen, X., Collins, J., Cui, Y., Devarapu, G.S.M., Du, Y., Fan, H., Fan, W.C., Feng, Z., Gao, M., Gorthi, R.K.S., Goyal, R., Han, J., Hatuwal, B., He, Z., Hu, X., Huang, X., Huang, Y., Jiang, D., Kang, B., Kannappan, P., Kittler, J., Lai, S., Li, N., Li, X., Li, X., Liang, C., Lin, L., Ling, H., Liu, T., Liu, Z., Lu, H., Luo, Y., Miao, D., Mogollon, J., Pang, Z., Pochimireddy, J.R., Prutyanov, V., Rahmon, G., Romanov, A., Shi, L., Siam, M., Sigal, L., Sivapuram, A.K., Solovyev, R., Kazemi, E.S., Toubal, I.E., Wan, J., Wang, L., Wang, X., Wang, Y., Wang, Y.X., Wang, Z., Wu, G., Wu, Q., Wu, X., Xia, Z., Xie, J., Xu, C., Xu, T., Xu, Y., Xue, C., Yang, C., Yang, J., Yang, M.H., Yu, C., Yu, K., Zhang, C., Zhang, J., Zhang, Z., Zheng, F., Zheng, Y., Zhong, B., Zhou, J., Zhou, J., Zhou, Y., Zhou, Z., Zhu, G., Zhu, J., Zhu, X., Zunin, V.: The second visual object tracking segmentation vots2024 challenge results. In: Del Bue, A., Canton, C., Pont-Tuset, J., Tommasi, T. (eds.) *Computer Vision – ECCV 2024 Workshops*. pp. 357–383. Springer Nature Switzerland, Cham (2025)
 34. Kristan, M., Matas, J., Danelljan, M., Felsberg, M., Chang, H.J., Čehovin Zajc, L., Lukežič, A., Drbohlav, O., Zhang, Z., Tran, K.T., Vu, X.S., Björklund, J., Mayer, C., Zhang, Y., Ke, L., Zhao, J., Fernández, G., Al-Shakarji, N., An, D., Arens, M., Becker, S., Bhat, G., Bullinger, S., Chan, A.B., Chang, S., Chen, H., Chen, X., Chen, Y., Chen, Z., Cheng, Y., Cui, Y., Deng, C., Dong, J., Dunnhofer, M., Feng, W., Fu, J., Gao, J., Han, R., Hao, Z., He, J.Y., He, K., He, Z., Hu, X., Huang, K., Huang, Y., Jiang, Y., Kang, B., Lan, J.P., Lee, H., Li, C., Li, J., Li, N., Li, W., Li, X., Li, X., Liu, P., Liu, Y., Lu, H., Luo, B., Luo, P., Ma, Y., Miao, D., Micheloni, C., Palaniappan, K., Park, H., Paul, M., Peng, H., Qian, Z., Rahmon, G., Scherer-Negenborn, N., Shao, P., Shin, W., Kazemi, E.S., Song, T.,

- Stiefelhagen, R., Sun, R., Tang, C., Tang, Z., Toubal, I.E., Valmadre, J., van de Weijer, J., Van Gool, L., Vira, J., Vujasinović, S., Wan, C., Wan, J., Wang, D., Wang, F., Wang, F., Wang, H., Wang, L., Wang, S., Wang, Y., Wang, Z., Wu, G., Wu, J., Wu, Q., Wu, X., Xiao, A., Xie, J., Xu, C., Xu, M., Xu, T., Xu, Y., Yan, B., Yang, D., Yang, M.H., Yang, T., Yang, Y., Yang, Z., Yin, X., Yu, F., Yu, H., Yu, Q., Yu, W., Yuan, Y., Yuan, Z., Zhang, J., Zhang, L., Zhang, T., Zhao, G., Zhao, S., Zheng, Y., Zhong, B., Zhu, J., Zhu, X., Zhuang, Y., Zong, C., Zuo, K.: The first visual object tracking segmentation vots2023 challenge results. In: 2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 1788–1810 (2023). <https://doi.org/10.1109/ICCVW60793.2023.00195>
35. Li, C., Yang, T., Zhu, S., Chen, C., Guan, S.: Density map guided object detection in aerial images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2020)
 36. Li, X., Zhong, B., Liang, Q., Mo, Z., Nong, J., Song, S.: Dynamic updates for language adaptation in visual-language tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19165–19174 (June 2025)
 37. Li, X., Miao, D., He, Z., Wang, Y., Lu, H., Yang, M.H.: Learning spatial-semantic features for robust video object segmentation. In: The Thirteenth International Conference on Learning Representations (2025), <https://openreview.net/forum?id=EM93t94zEi>
 38. Liang, S., Bai, Y., Gong, Y., Wei, X.: Autoregressive sequential pretraining for visual tracking. In: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7254–7264 (2025). <https://doi.org/10.1109/CVPR52734.2025.00680>
 39. Lin, L., Fan, H., Zhang, Z., Wang, Y., Xu, Y., Ling, H.: Tracking meets lora: Faster training, larger model, stronger performance. In: Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., Varol, G. (eds.) Computer Vision – ECCV 2024. pp. 300–318. Springer Nature Switzerland, Cham (2025)
 40. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In: Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., Varol, G. (eds.) Computer Vision – ECCV 2024. pp. 38–55. Springer Nature Switzerland, Cham (2025)
 41. Liu, X., Zhou, L., Zhou, Z., Chen, J., He, Z.: Mambavlt: Time-evolving multimodal state space model for vision-language tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8731–8741 (June 2025)
 42. Lukežič, A., Matas, J., Kristan, M.: A discriminative single-shot segmentation network for visual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44**(12), 9742–9755 (2022). <https://doi.org/10.1109/TPAMI.2021.3137933>
 43. Ma, Y., Tang, Y., Yang, W., Zhang, T., Zhang, J., Kang, M.: Unifying visual and vision-language tracking via contrastive learning. *Proceedings of the AAAI Conference on Artificial Intelligence* **38**(5), 4107–4116 (Mar 2024). <https://doi.org/10.1609/aaai.v38i5.28205>, <https://ojs.aaai.org/index.php/AAAI/article/view/28205>
 44. Ma, Z., Wang, L., Zhang, H., Lu, W., Yin, J.: Rpt: Learning point set representation for siamese visual tracking. In: Bartoli, A., Fusiello, A. (eds.) *Computer Vision*

- ECCV 2020 Workshops. pp. 653–665. Springer International Publishing, Cham (2020)
- 45. Mayer, C., Danelljan, M., Paudel, D.P., Van Gool, L.: Learning target candidate association to keep track of what not to track. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 13444–13454 (October 2021)
- 46. Meethal, A., Granger, E., Pedersoli, M.: Cascaded zoom-in detector for high resolution aerial images. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 2046–2055 (2023). <https://doi.org/10.1109/CVPRW59228.2023.00198>
- 47. Muller, M., Bibi, A., Giancola, S., Alsubaihi, S., Ghanem, B.: Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
- 48. Munkres, J.: Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics* **5**(1), 32–38 (1957). <https://doi.org/10.1137/0105003>, <https://doi.org/10.1137/0105003>
- 49. Oh, S.W., Lee, J.Y., Xu, N., Kim, S.J.: Video object segmentation using space-time memory networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019)
- 50. Qin, H., Xu, T., Li, T., Chen, Z., Feng, T., Li, J.: Must: The first dataset and unified framework for multispectral uav single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16882–16891 (June 2025)
- 51. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollar, P., Feichtenhofer, C.: SAM 2: Segment anything in images and videos. In: The Thirteenth International Conference on Learning Representations (2025)
- 52. Ren, W., Kang, D., Tang, Y., Chan, A.B.: Fusing crowd density maps and visual object trackers for people tracking in crowd scenes. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5353–5362 (2018). <https://doi.org/10.1109/CVPR.2018.00561>
- 53. Ryali, C., Hu, Y.T., Bolya, D., Wei, C., Fan, H., Huang, P.Y., Aggarwal, V., Chowdhury, A., Poursaeed, O., Hoffman, J., Malik, J., Li, Y., Feichtenhofer, C.: Hiera: A hierarchical vision transformer without the bells-and-whistles. In: Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J. (eds.) Proceedings of the 40th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 202, pp. 29441–29454. PMLR (23–29 Jul 2023), <https://proceedings.mlr.press/v202/ryali23a.html>
- 54. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 618–626 (2017). <https://doi.org/10.1109/ICCV.2017.74>
- 55. Seong, H., Hyun, J., Kim, E.: Kernelized memory network for video object segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 629–645. Springer International Publishing, Cham (2020)
- 56. Shao, Y., He, S., Ye, Q., Feng, Y., Luo, W., Chen, J.: Context-aware integration of language and visual references for natural language tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19208–19217 (June 2024)

57. Sun, P., Cao, J., Jiang, Y., Yuan, Z., Bai, S., Kitani, K., Luo, P.: Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 20993–21002 (June 2022)
58. Videnovic, J., Lukezic, A., Kristan, M.: A distractor-aware memory for visual object tracking with sam2. In: Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR). pp. 24255–24264 (June 2025)
59. Voigtlaender, P., Chai, Y., Schroff, F., Adam, H., Leibe, B., Chen, L.C.: Feelvos: Fast end-to-end embedding learning for video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
60. Wang, X., Shu, X., Zhang, Z., Jiang, B., Wang, Y., Tian, Y., Wu, F.: 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 (2021)
61. Wen, L., Du, D., Zhu, P., Hu, Q., Wang, Q., Bo, L., Lyu, S.: Detection, tracking, and counting meets drones in crowds: A benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7812–7821 (June 2021)
62. Wu, Y., Wang, X., Yang, X., Liu, M., Zeng, D., Ye, H., Li, S.: Learning occlusion-robust vision transformers for real-time uav tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 17103–17113 (June 2025)
63. Xie, F., Wang, Z., Ma, C.: Diffusiontrack: Point set diffusion model for visual object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19113–19124 (June 2024)
64. Xie, J., Zhong, B., Mo, Z., Zhang, S., Shi, L., Song, S., Ji, R.: Autoregressive queries for adaptive tracking with spatio-temporal transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19300–19309 (June 2024)
65. Xiong, Y., Zhou, C., Xiang, X., Wu, L., Zhu, C., Liu, Z., Suri, S., Varadarajan, B., Akula, R., Iandola, F., Krishnamoorthi, R., Soran, B., Chandra, V.: Efficient track anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 11513–11524 (October 2025)
66. Xu, Q., Zhu, L., Liu, C., Lin, G., Long, C., Li, Z., Zhao, R.: Samite: Position prompted sam2 with calibrated memory for visual object tracking. arXiv preprint arXiv:2507.21732 (2025)
67. Xue, C., Zhong, B., Liang, Q., Zheng, Y., Li, N., Xue, Y., Song, S.: Similarity-guided layer-adaptive vision transformer for uav tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6730–6740 (June 2025)
68. Xun, Z., Di, S., Gao, Y., Tang, Z., Wang, G., Liu, S., Li, B.: Linker: Learning long short-term associations for robust visual tracking. *IEEE Transactions on Multimedia* **26**, 6228–6237 (2024). <https://doi.org/10.1109/TMM.2023.3347644>
69. Yan, B., Peng, H., Fu, J., Wang, D., Lu, H.: Learning spatio-temporal transformer for visual tracking. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10448–10457 (2021)
70. Yan, B., Zhang, X., Wang, D., Lu, H., Yang, X.: Alpha-refine: Boosting tracking performance by precise bounding box estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5289–5298 (June 2021)

71. Yang, C.Y., Huang, H.W., Chai, W., Jiang, Z., Hwang, J.N.: Samurai: Motion-aware memory for training-free visual object tracking with sam 2. *IEEE Transactions on Image Processing* **35**, 970–982 (2026). <https://doi.org/10.1109/TIP.2026.3651835>
72. Yang, Z., Wei, Y., Yang, Y.: Associating objects with transformers for video object segmentation. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) *Advances in Neural Information Processing Systems*. vol. 34, pp. 2491–2502. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/147702db07145348245dc5a2f2fe5683-Paper.pdf
73. Yang, Z., Yang, Y.: Decoupling features in hierarchical propagation for video object segmentation. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) *Advances in Neural Information Processing Systems*. vol. 35, pp. 36324–36336. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/eb890c36af87e4ca82e8ef7bcb6a6a284-Paper-Conference.pdf
74. Yang, Z., Yang, Y.: Decoupling features in hierarchical propagation for video object segmentation. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) *Advances in Neural Information Processing Systems*. vol. 35, pp. 36324–36336. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/eb890c36af87e4ca82e8ef7bcb6a6a284-Paper-Conference.pdf
75. Ye, B., Chang, H., Ma, B., Shan, S., Chen, X.: Joint feature learning and relation modeling for tracking: A one-stream framework. In: *European conference on computer vision*. pp. 341–357. Springer (2022)
76. Zhang, Z., Peng, H., Fu, J., Li, B., Hu, W.: Ocean: Object-aware anchor-free tracking. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) *Computer Vision – ECCV 2020*. pp. 771–787. Springer International Publishing, Cham (2020)
77. Zheng, Y., Zhong, B., Liang, Q., Mo, Z., Zhang, S., Li, X.: Odtrack: Online dense temporal token learning for visual tracking. *Proceedings of the AAAI Conference on Artificial Intelligence* **38**(7), 7588–7596 (Mar 2024). <https://doi.org/10.1609/aaai.v38i7.28591>, <https://ojs.aaai.org/index.php/AAAI/article/view/28591>
78. Zhou, J., Pang, Z., Wang, Y.X.: Rmem: Restricted memory banks improve video object segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 18602–18611 (June 2024)