MUTrack: A Memory-Aware Unified Representation Framework for Visual Tracking

Weijing Wu[1,2], Qihua Liang[1,2*], Bineng Zhong[1,2*], Xiaohu Tang[1,2], Yufei Tan[1,2,3], Ning Li[1,2], Yuanliang Xue[4]

[1]Key Laboratory of Education Blockchain and Intelligent Technology Ministry of Education, Guangxi Normal University, Guilin 541004, China
[2]University Engineering Research Center of Educational Intelligent Technology, Guangxi Normal University, Guilin, 541004, China
[3]Guangxi Key Laboratory of Brain-inspired Computing and Intelligent Chips, School of Electronic and Information Engineering, Guangxi Normal University, Guilin, 541004, China
[4]Xi'an Research Institute of High Technology, Xi'an 710025, China
wuweijing182@163.com, qhliang@gxnu.edu.cn, bnzhong@gxnu.edu.cn, tangxiaohu@gxnu.edu.cn
jeffrey.yf.tan@gxnu.edu.cn, ningli65536@mailbox.gxnu.edu.cn, xyl_507@outlook.com

Abstract

Building a unified target representation that simultaneously achieves short-term adaptability and long-term stability is crucial for robust visual tracking. However, existing trackers typically face an inherent trade-off. Methods primarily relying on short-term appearance and motion cues achieve rapid adaptation, but they often struggle with long-term identity consistency. Conversely, trackers that emphasize extensive temporal context provide strong robustness, yet this approach can compromise their short-term adaptability. To bridge this gap, we propose a novel tracker, **MUTrack**, which comprehensively integrates both long-term and short-term memories into a unified target representation for more robust tracking. Specifically, we design a unified memory bank that stores and manages long-term memory for maintaining long-term identity consistency, and short-term memory for adapting to instantaneous appearance changes. To fully leverage the complementary nature of both long-term and short-term temporal information, we introduce a perception interaction module that dynamically fuses these memory types through deep and bidirectional interactions, enabling mutual refinement where one guides the other. This ultimately generates a highly adaptive target representation, which effectively balances adaptability to instantaneous changes with robustness against long-term identity drift. Extensive experiments on GOT10k, TrackingNet, LaSOT, LaSOT$_{ext}$, NfS, and OTB100 consistently demonstrate that MUTrack achieves SOTA performance.

Introduction

Visual object tracking (VOT) (Bertinetto et al. 2016) is a fundamental computer vision task that predicts a target's position in videos given an initial state. Earlier tracking approaches (Chen et al. 2021, 2022; Hu et al. 2023; Fu et al. 2022; Hu et al. 2024) have primarily relied on an initial static template (illustrated as the initial memory in Fig. 1(a)), leading to limited use of contextual cues. This reliance is unreliable when the target undergoes appearance or scale variations. To overcome this, many trackers introduce short-term memory (as shown in Fig. 1(a)). For instance, subsequent studies (Yang and Chan 2018; Zhang et al. 2019; Yan et al. 2021) introduced template update mechanisms to adapt to target appearance changes. However, these models depend on manual parameters, resulting in limited adaptability in dynamic scenarios. Some methods addressed these challenges by designing appearance token propagation mechanisms (Bai et al. 2024; Xie et al. 2024) or by incorpo-

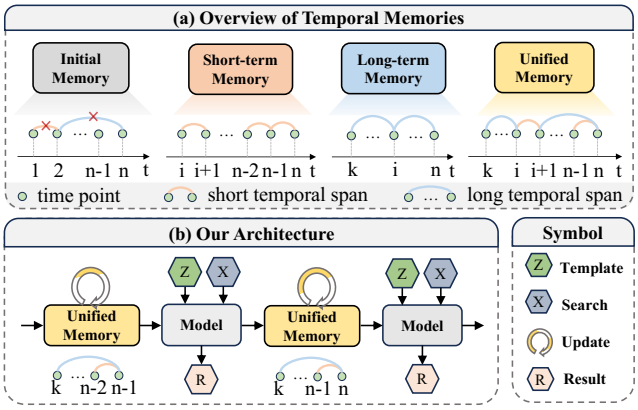

Figure 1: (a) The top part illustrates representative tracking strategies based on different temporal memories: initial memory-based trackers (Ye et al. 2022) typically rely on a non-adaptive initial template for target matching; short-term memory-based trackers (Bai et al. 2024) emphasize recent target features for rapid adaptation; while long-term memory-based trackers (Zheng et al. 2024) focus on preserving appearance continuity over extended time spans. In contrast, unified memory-based trackers integrate these complementary approaches to construct a more complete and balanced target representation across time. For this illustration, it is assumed that $k \leq i$. (b) The bottom part presents an overview of our proposed tracking framework.

*Corresponding Author

rating motion cues (Bai et al. 2024; Xue et al. 2024), notably improving tracking robustness and performance. Although these methods effectively utilize short-term memory, error accumulation during occlusions remains a critical challenge, often causing irreversible degradation in tracking accuracy. Some trackers (Zheng et al. 2024) have independently integrated long-term memory (illustrated as the long-term memory in Fig. 1(a)), enriching the target's representation. Nevertheless, trackers relying predominantly on long-term memory often struggle to adapt to sudden appearance changes, making them less effective with motion blur and severe target deformation. Consequently, it is essential to simultaneously exploit both long-term and short-term memory. **A critical question arises: How can we efficiently capture and exploit both types of information to enable robust tracking?**

To achieve this, we propose a novel tracking framework named **MUTrack**. As illustrated in Fig.1(b), the MUTrack stems from two core components: the Unified Memory Bank(UMB) and the Perception Interaction Module(PIM). The UMB is conceptualized as an advanced system for managing long-term memory(LTM) and short-term memory(STM). Far from being a simple repository, the UMB actively manages these crucial feature streams. For generating the LTM, we employ our innovative Sparse Multi-scale Extractor(SME), a lightweight module integrated within the UMB. The SME efficiently processes a diverse reference frame pool through multi-scale convolutional operations and specialized sampling to construct a potent LTM, which captures robust, context-rich information of the target. Simultaneously, it maintains an STM, typically derived from a queue capturing recent appearance evolution data, that precisely captures the target's most current visual state. This dual-stream management ensures that both enduring and transient aspects of the target are available as rich feature inputs, which then serve as inputs to the PIM.

The PIM serves as the very dynamic core for the intelligent and adaptive fusion of these complementary memories. Recognizing that simple concatenation or superficial mixing is often insufficient for robust tracking, the PIM is therefore designed to facilitate deep, hierarchical, and bidirectional interactions between LTM and STM. In this synergistic, bidirectional interaction, stable LTM provides historical context to clarify and validate recent STM features. Concurrently, the STM, representing the current state, acts as a dynamic query to extract the most relevant long-term patterns from LTM. The result is a synergistic, holistic, and robust unified representation that is ultimately far more powerful than the simple sum of its parts, especially in challenging scenarios.

- We introduce MUTrack, a unified memory-based tracking framework designed to achieve robust tracking by constructing a unified and adaptive target representation.
- Our framework's two core components are the UMB, which captures rich target contextual information across diverse temporal spans, and the PIM, which bidirectionally fuses and refines this information into a highly adaptive unified representation.

Related Work

Initial Memory-Based Tracking. Early object tracking approaches primarily relied on an initial template to represent and track targets. Many offline Siamese trackers (Bertinetto et al. 2016; Li et al. 2019) exemplify this by treating tracking as a matching problem between such an initial static template and the search region. Despite their remarkable progress, these methods primarily capture local similarity, often overlooking valuable global context. Recent works (Chen et al. 2021; Zheng et al. 2023; Xue et al. 2025; Ge et al. 2025b; Li et al. 2024; Zheng et al. 2022; Ge et al. 2024, 2025a; Wang, Li, and Ge 2025; Shi et al. 2025b,a; Wang et al. 2025, 2024b,a; Zheng et al. 2025) leverage transformer-based architectures to enhance the modeling of complex dependencies in visual tracking. For instance, TransT (Chen et al. 2021) presents a transformer-based architecture utilizing ego-context and cross-feature augmentation for robust feature fusion. While these approaches significantly increase accuracy, their inherent dependence on a non-adaptive initial appearance model limits their adaptability in highly dynamic or cluttered environments.

Short-Term Memory-Based Tracking. To enhance tracker adaptability to rapid appearance variations, various prominent methods focus on two main categories: appearance evolution modeling (e.g., template updates and target appearance propagation) and motion modeling. AQATrack (Xie et al. 2024) introduce novel end-to-end learnable mechanisms to dynamically model target appearance evolution over time, avoiding predefined hyperparameters for update frequency or importance weighting. Beyond appearance modeling, incorporating motion cues also significantly improves short-term tracking robustness. ARTrackV2 (Bai et al. 2024) explicitly integrate crucial temporal motion information. However, despite their notably improved adaptability, these short-term memory methods still often suffer from significant error accumulation during target occlusions due to their inherently limited temporal span, frequently leading to a severe and irreversible degradation in overall tracking accuracy or complete loss of the target in prolonged occlusions or extended out-of-view scenarios.

Long-Term Memory-Based Tracking. To overcome short-term adaptation limits, long-term memory mechanisms have been explored to retain historical information. While STMTrack (Fu et al. 2021) uses spatio-temporal memory, its separate backbones increase complexity. More recently, some trackers (Zheng et al. 2024; He et al. 2023; Hu et al. 2025; Zeng et al. 2025; Hu et al. 2024) leverage multiple templates to capture target evolution and select high-quality exemplars for robust tracking. However, this heavy multi-template reliance can lead to significant template drift in challenging scenarios, ultimately causing target loss.

In this work, we aim to efficiently capture and robustly leverage both the LTM and the STM, thereby enabling real-time tracking while consistently maintaining reliable target representation across diverse challenging scenarios.

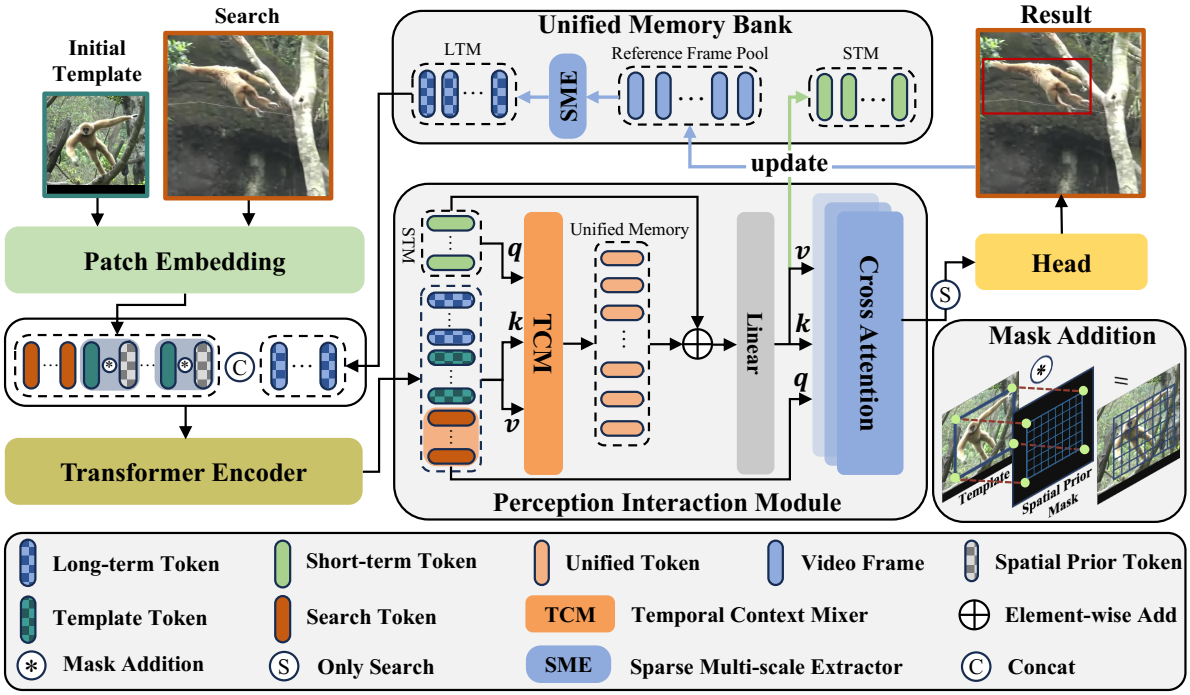

Figure 2: **Overview of our proposed MUTrack**. The architecture comprises four main components: a Transformer Encoder for robust visual feature extraction; the UMB, with the SME, for generating and managing the LTM and STM; the PIM for deep, bidirectional LTM-STM interaction and refinement into a unified representation; and a prediction head for final target localization. Green and blue bold arrows indicate how STM is refreshed with recent features and LTM is generated by SME from an updated reference pool.

Our Method

Overview

The overall architecture of MUTrack is shown in Fig. 2. It consists of four components: a Transformer Encoder for robust visual feature extraction, the UMB for adaptively storing and managing memory representations, the PIM for the LTM and the STM interaction, and a prediction head for final tracking output. First, the initial template and the search region are effectively embedded into patch tokens via a patch embedding operation. Meanwhile, the SME, as an integrated component of the UMB, operates on the reference frame pool to comprehensively generate the LTM through a specialized sampling and convolutional processing pipeline. The initial template feature, search feature, and the LTM are then spatially concatenated and fed into the Transformer Encoder for joint representation learning. The resulting enhanced features are adaptively fused with the STM via the PIM, forming a comprehensive unified memory. This memory is injected into search features, guiding the prediction head for robust target localization.

Unified Memory Bank

The UMB is a fundamental component of MUTrack that efficiently manages two complementary memory streams: the LTM and the STM. This unique and powerful dual memory design is key to its efficacy, as it enables our framework to maintain stable historical representations for long-term consistency while effectively and continuously adapting to immediate appearance changes in dynamic environments.

Sparse Multi-scale Extractor. To efficiently extract the LTM, we introduce the SME, a lightweight module. As illustrated in Fig. 4, this module is designed to extract compact and expressive multi-scale features from a carefully selected reference frames(detailed next), enabling downstream modules to better understand the target's historical appearance while minimizing computational cost.

To enable efficient tracking during the inference stage, $\boldsymbol{\mathcal{R}}_{\text{pool}}$ is dynamically constructed by storing target patches cropped from each predicted bounding box. Let $\{\mathcal{R}_i\}_{i=1}^{m}$ denote a subset sampled from $\boldsymbol{\mathcal{R}}_{\text{pool}}$ for subsequent processing. Each reference frame $\mathcal{R}_i \in \mathbb{R}^{3\times H_z\times W_z}$ is independently passed through three convolutional layers with kernel sizes 18×18, 16×16, and 14×14, respectively, yielding three feature maps:

$$\boldsymbol{\mathcal{R}}^{(j)} = \text{Concat}(\mathcal{R}_1^{(j)}, \ldots, \mathcal{R}_m^{(j)}), j \in \{14, 16, 18\}, \quad (1)$$

The extracted features are first concatenated to yield a multi-scale representation:

$$\boldsymbol{\mathcal{R}}_{\text{multi}} = \text{Concat}(\boldsymbol{\mathcal{R}}^{(18)}, \boldsymbol{\mathcal{R}}^{(16)}, \boldsymbol{\mathcal{R}}^{(14)}), \quad (2)$$

To further enhance the semantic expressiveness of the aggregated multi-frame features while alleviating the quadratic

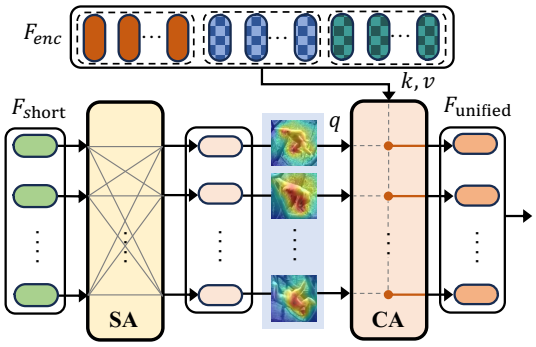

Figure 3: The structure of the Temporal Context Mixer. SA and CA represent self-attention and cross-attention, respectively. The symbols maintain the same meaning with Fig. 2.

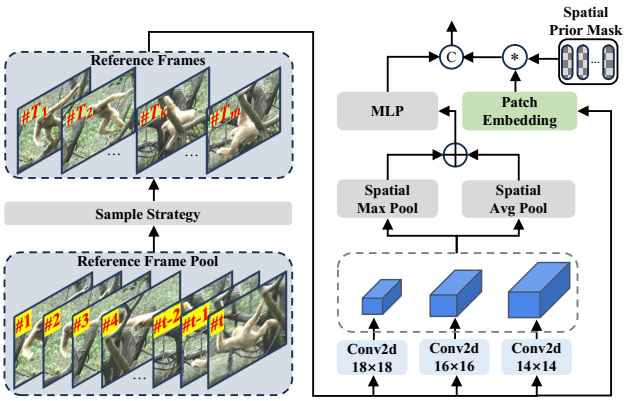

Figure 4: The structure of the Sparse Multi-scale Extractor. The symbols maintain the same meaning with Fig. 2.

computational cost of the subsequent Transformer Encoder, both max pooling and average pooling are applied over $\boldsymbol{\mathcal{R}}_{\text{multi}}$. The results are summed to preserve rich contextual information, and the fused feature is then passed through a Multi-Layer Perceptron (MLP) to produce a compact representation:

$$\boldsymbol{\mathcal{F}}_{\text{multi}} = \text{MLP}(\text{MaxPool}(\boldsymbol{\mathcal{R}}_{\text{multi}}) + \text{AvgPool}(\boldsymbol{\mathcal{R}}_{\text{multi}})), \tag{3}$$

In the meantime, we conduct the next step as follows:

$$\mathcal{R}_i^{\text{mask}} = \mathcal{R}_i^{\text{patch}} + \mathcal{P}_i^{\text{mask}}, \quad i = 1, 2, \ldots, m, \tag{4}$$

Here, $\mathcal{R}_i^{\text{patch}} \in \mathbb{R}^{C \times \frac{H_z}{16} \times \frac{W_z}{16}}$ denotes the patch-level features extracted from the i-th reference frame $\mathcal{F}_i$. As depicted in the "Mask Addition" component of Fig. 2, the spatial prior mask $\mathcal{P}_i^{\text{mask}} \in \mathbb{R}^{C \times \frac{H_z}{16} \times \frac{W_z}{16}}$ encodes the predicted target location in a dense spatial format and is added element-wise to the patch features $\mathcal{R}_i^{\text{patch}}$, yielding the enhanced template features $\boldsymbol{\mathcal{F}}_{\text{ref}}$ that explicitly incorporate target-aware positional information.

$$\boldsymbol{\mathcal{F}}_{\text{ref}} = \text{Concat}(\mathcal{R}_1^{\text{mask}}, \ldots, \mathcal{R}_m^{\text{mask}}), \tag{5}$$

$$\boldsymbol{\mathcal{F}}_{\text{long}} = \text{Concat}(\boldsymbol{\mathcal{F}}_{\text{multi}}, \boldsymbol{\mathcal{F}}_{\text{ref}}), \tag{6}$$

Subsequently, $\boldsymbol{\mathcal{F}}_{\text{multi}}$, a compact and contextual representation, is concatenated along with $\boldsymbol{\mathcal{F}}_{\text{ref}}$, forming a comprehensive long-range representation $\boldsymbol{\mathcal{F}}_{\text{long}}$, which serves as part of the visual input to the subsequent Transformer Encoder.

Reference Frame Pool. To capture the temporal evolution of the target over extended periods, we maintain a reference frame pool $\boldsymbol{\mathcal{R}}_{\text{pool}}$, which is dynamically updated during tracking by storing template patches cropped from the predicted bounding boxes at each frame. To ensure efficient inference, we directly store each cropped template without applying confidence-based filtering. This design eliminates additional computation while relying on our sampling strategy to maintain the overall quality and robustness of the selected templates. We explore three distinct sampling strategies for selecting frames from the pool: (1) *Random* sampling, which randomly selects frames; (2) *Recent-Consecutive* sampling, which prioritizes the most recent consecutive frames; and (3) *Uniform* sampling, which selects frames at regular intervals to ensure broad temporal coverage.

Short-Term Memory Queue. To capture recent appearance dynamics, a First-In-First-Out (FIFO) queue mechanism maintains the STM. This design enables the tracker to adapt rapidly to immediate appearance changes while preserving temporal continuity. As depicted in Fig. 2, output features from the MLP (shown in gray) update the STM at each time step.

Formally, we define the STM queue at time step t as $\mathcal{Q}_t = \{\boldsymbol{q}^{t-n_t+1}, \boldsymbol{q}^{t-n_t+2}, \ldots, \boldsymbol{q}^t\}$, where $n_t \leq k$ denotes the current queue length. Here, k is a predefined maximum capacity that dictates the temporal span of the STM. Each element $\boldsymbol{q}^j \in \mathbb{R}^{N_s \times C}$ is a short-term feature representing the target's appearance from a previous frame. At the beginning of tracking (i.e., for $t = 0$), the queue elements are initialized to zeros vector.

At each time step t, after obtaining the current aggregated feature $\boldsymbol{\mathcal{F}}_{\text{agg}} \in \mathbb{R}^{N_s \times C}$ (which is the output of MLP_{agg} as discussed in the Temporal Context Mixer), we define the new STM element for the current time step as $\boldsymbol{q}^{t+1} = \boldsymbol{\mathcal{F}}_{\text{agg}}$. The update procedure is then defined as:

$$\mathcal{Q}_{t+1} = \begin{cases} \{\boldsymbol{q}^{t-n_t+2}, \boldsymbol{q}^{t-n_t+3}, \ldots, \boldsymbol{q}^t, \boldsymbol{q}^{t+1}\}, & \text{if } n_t = k \\ \{\boldsymbol{q}^{t-n_t+1}, \boldsymbol{q}^{t-n_t+2}, \ldots, \boldsymbol{q}^t, \boldsymbol{q}^{t+1}\}, & \text{if } n_t < k \end{cases} \tag{7}$$

This update mechanism ensures the queue $\mathcal{Q}_{t+1}$ always retains the $\min(t+1, k)$ most recent features by dequeuing the oldest $\boldsymbol{q}^{t-n_t+1}$ before enqueuing the newest $\boldsymbol{q}^{t+1}$ when capacity k is reached. This robust, dynamic memory consistently provides the Perception Interaction Module with a relevant temporal window of recent appearance dynamics, enabling the tracker to adapt to rapid changes and gradual evolutions while maintaining robustness.

Transformer Encoder

To retain fine-grained spatial details in visual representation learning, we adopt HiViT (Zhang et al. 2023) as our Transformer encoder. Unlike the conventional ViT (Dosovitskiy et al. 2020), which directly applies 16×16 patch embed-

dings, HiViT projects the search image and the template image into token sequences $\mathcal{F}_{\mathrm{x}} \in \mathbb{R}^{N_x \times C}$ and $\mathcal{F}_{\mathrm{z}} \in \mathbb{R}^{N_z \times C}$, respectively. This transformation is performed through a three-stage hierarchical downsampling pipeline, consisting of an initial 4×4 patch embedding layer followed by two successive 2×2 merging layers. The number of resulting tokens is given by $N_x = \frac{H_x W_x}{16^2}$, $N_z = \frac{H_z W_z}{16^2}$. Furthermore, the LTM $\mathcal{F}_{\mathrm{long}} \in \mathbb{R}^{N_l \times C}$, extracted by the SME, is concatenated with the search feature $\mathcal{F}_{\mathrm{x}}$ and the template feature $\mathcal{F}_{\mathrm{z}}$ to form the encoder input:

$$\mathcal{F}_{\mathrm{enc}} = \mathrm{Concat}(\mathcal{F}_{\mathrm{z}}, \mathcal{F}_{\mathrm{x}}, \mathcal{F}_{\mathrm{long}}). \tag{8}$$

Perception Interaction Module

The PIM plays a crucial role in MUTrack. As illustrated in Fig. 2, the PIM serves as a bridge that connects the STM, which captures continuous and dense appearance evolution, with the stable visual features extracted by the Transformer Encoder, which are derived from the initial template, current search region, and the LTM. This fusion yields an adaptive and comprehensive target representation that balances robustness and adaptability, and is subsequently injected into the search region to enhance its feature representation.

Temporal Context Mixer. As illustrated in Fig. 3, the goal of this stage is to integrate the STM $\mathcal{F}_{\mathrm{short}} \in \mathbb{R}^{k \times N_s \times C}$ with the stable visual features $\mathcal{F}_{\mathrm{enc}}$ extracted by the Transformer Encoder, thereby forming a unified and temporally-aware target representation. We first refine the STM $\mathcal{F}_{\mathrm{short}}$ using self-attention to capture intra-memory dependencies. Subsequently, cross-attention is applied between the refined STM $\mathcal{F}'_{\mathrm{short}} \in \mathbb{R}^{k \times N_s \times C}$ and the stable visual features $\mathcal{F}_{\mathrm{enc}}$ to generate a unified temporal representation $\mathcal{F}_{\mathrm{unified}}$. To further process and compact these features, a linear projection is applied to $\mathcal{F}_{\mathrm{unified}} \in \mathbb{R}^{k \times N_s \times C}$, mapping it into the final memory representation $\mathcal{F}_{\mathrm{mem}} \in \mathbb{R}^{N_s \times C}$.

$$\mathcal{F}'_{\mathrm{short}} = \mathrm{SelfAttn}(\mathcal{F}_{\mathrm{short}}, \mathcal{F}_{\mathrm{short}}, \mathcal{F}_{\mathrm{short}}), \tag{9}$$

$$\mathcal{F}_{\mathrm{unified}} = \mathrm{CrossAttn}(\mathcal{F}'_{\mathrm{short}}, \mathcal{F}_{\mathrm{enc}}, \mathcal{F}_{\mathrm{enc}}), \tag{10}$$

$$\mathcal{F}_{\mathrm{mem}} = \mathrm{Linear}(\mathcal{F}_{\mathrm{unified}}), \tag{11}$$

This final memory representation $\mathcal{F}_{\mathrm{mem}}$ is then deeply fused with the search feature $\mathcal{F}_{\mathrm{x}}$ via a multi-layer cross-attention mechanism. The process yields a context-enhanced feature for prediction, denoted as $\mathcal{F}_{\mathrm{out}}$.

Head and Loss

The prediction head consists of three parallel branches built on lightweight convolutional layers, which respectively output a classification score map ($\in \mathbb{R}^{1 \times \frac{H_s}{p} \times \frac{W_s}{p}}$), a bounding box size map ($\in \mathbb{R}^{2 \times \frac{H_s}{p} \times \frac{W_s}{p}}$), and a center offset map ($\in \mathbb{R}^{2 \times \frac{H_s}{p} \times \frac{W_s}{p}}$). The training objective is a weighted sum of three distinct losses: the Focal Loss(Lin et al. 2017) (L_{cls}) for classification, and a combination of the L_1 loss and the Generalized IoU (GIoU) loss (Rezatofighi et al. 2019) for bounding box regression. The total loss L is defined as:

$$L = L_{cls} + \lambda_1 L_1 + \lambda_2 L_{GIoU}, \tag{12}$$

where the weights λ_1 and λ_2 are set to 5 and 2, respectively, to balance the regression terms.

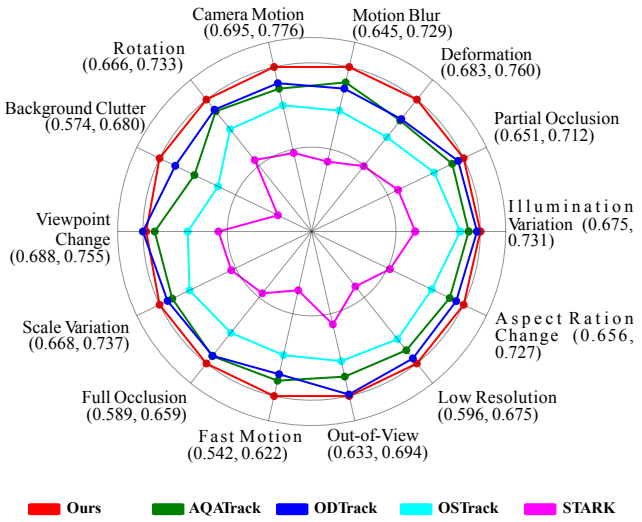

Figure 5: AUC scores of different attributes on LaSOT (Fan et al. 2019). Best viewed in color and by zooming in.

Model	MACs	Params	Speed
SeqTrack-256 (Chen et al. 2023)	66G	89M	40fps
MUTrack-256	39.2G	73.4M	69fps

Table 1: Comparison of model Params, FLOPs, and Speed on NVIDIA A100.

Experiment

Implementation Details

Model. We implemented two different variants of MUTrack models, described as follows:

- **MUTrack-256**. Template: 128×128 pixels; Search region: 256×256 pixels.
- **MUTrack-384**. Template: 192×192 pixels; Search region: 384×384 pixels.

Training. Our model was trained and tested on 4 NVIDIA A800 GPUs with 80GB of memory. We trained our model on GOT-10K (Huang, Zhao, and Huang 2021), LaSOT (Fan et al. 2019), COCO (Lin et al. 2014), and TrackingNet (Müller et al. 2018) to align with mainstream trackers. For the GOT-10K test set, we exclusively trained on its dataset as per official guidelines. We used AdamW (Loshchilov and Hutter 2017) as the optimizer, with an initial learning rate of 2×10^{-4} for 300 epochs (60k samples per epoch). The backbone's learning rate was 0.1 times the initial rate, and the learning rate decayed by a factor of 10 after 240 epochs.
Inference. During inference, the reference frame pool is first initialized with three identical initial template frames. Subsequently, a uniform sampling strategy within the SME selects three reference frames from this pool to extract the LTM. Concurrently, the short-term memory queue is initialized with zeros corresponding to its predefined maximum length. All other inference procedures mirror the architecture and operations employed during the training phase.

Method	Source	LaSOT			LaSOT$_{ext}$			GOT-10k*			TrackingNet		
		AUC	P$_{norm}$	P	AUC	P$_{norm}$	P	AO	SR$_{0.5}$	SR$_{0.75}$	AUC	P$_{norm}$	P
MUTrack-B256	Ours	**72.3**	82.9	**79.5**	51.2	62.5	58.8	**78.1**	**88.3**	**77.5**	**85.5**	**90.4**	**85.2**
MambaLCT-B256(Li et al. 2025)	AAAI25	71.8	**83.0**	79.4	51.6	**64.0**	59.0	74.8	85.4	72.1	84.3	89.2	83.9
TemTrack-B256(Xie et al. 2025)	AAAI25	72.0	82.1	79.1	**52.4**	63.3	**60.2**	74.9	84.8	71.7	84.3	88.8	83.5
ARTrackV2-B256 (Bai et al. 2024)	CVPR24	71.6	80.2	77.2	50.8	61.9	57.7	75.9	85.4	72.7	84.9	89.3	84.5
AQATrack-256(Xie et al. 2024)	CVPR24	71.4	81.9	78.6	51.2	62.2	58.9	73.8	83.2	72.1	83.8	88.6	83.1
SeqTrack-B256(Chen et al. 2023)	CVPR23	69.9	79.7	76.3	49.5	60.8	56.3	74.7	84.7	71.8	83.3	88.3	82.2
MixFormer-22k(Cui et al. 2022)	CVPR22	69.2	78.7	74.7	-	-	-	70.7	80.0	67.8	83.1	88.1	81.6
OSTrack-256(Ye et al. 2022)	ECCV22	69.1	78.7	75.2	47.4	57.3	53.3	71.0	80.4	68.2	83.1	87.8	82.0
TransT (Chen et al. 2021)	CVPR21	64.9	73.8	69.0	-	-	-	67.1	76.8	60.9	81.4	86.7	80.3
Ocean (Zhang et al. 2020)	ECCV 20	56.0	65.1	56.6	-	-	-	61.1	72.1	47.3	-	-	-
SiamRPN++(Li et al. 2019)	CVPR19	49.6	56.9	49.1	34.0	41.6	39.6	51.7	61.6	32.5	73.3	80.0	69.4
ECO (Danelljan et al. 2017)	ICCV 17	32.4	33.8	30.1	22.0	25.2	24.0	31.6	30.9	11.1	-	-	-
SiamFC (Bertinetto et al. 2016)	ECCVW16	33.6	42.0	33.9	23.0	31.1	26.9	34.8	35.3	9.8	-	-	-
Some Trackers with Higher Resolution													
SeqTrack-B384(Chen et al. 2023)	CVPR23	71.5	81.1	77.8	50.5	61.6	57.5	74.5	84.3	71.4	83.9	88.8	83.6
HIPTrack(Cai, Liu, and Wang 2024)	CVPR24	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-B384(Xie et al. 2024)	CVPR24	72.7	82.9	80.2	52.7	64.2	60.8	76.0	85.2	74.9	84.8	89.3	84.3
LoRAT-B378 (Lin et al. 2024)	ECCV24	72.9	81.9	79.1	53.1	64.8	60.6	73.7	82.6	72.9	84.2	88.4	83.0
ODTrack-B384 (Zheng et al. 2024)	AAAI24	73.2	83.2	80.6	52.4	63.9	60.1	77.0	87.9	75.1	85.1	90.1	84.9
ARTrackV2-B384 (Bai et al. 2024)	CVPR24	73.0	82.0	79.6	52.9	63.4	59.1	77.5	86.0	75.5	85.7	89.8	85.5
TemTrack-B256(Xie et al. 2025)	AAAI25	73.1	83.0	80.7	53.4	64.8	61.0	76.1	84.9	74.4	85.0	89.3	84.8
MambaLCT-B256(Li et al. 2025)	AAAI25	73.6	84.1	81.6	53.3	64.8	61.4	76.2	86.7	74.3	85.2	89.8	85.2
MUTrack-B384	Ours	**73.9**	**84.6**	**82.5**	**54.7**	**66.4**	**62.8**	**79.4**	**89.3**	**79.3**	**86.0**	**90.9**	**86.0**

Table 2: Performance comparison with state-of-the-art trackers on LaSOT, LaSOT ext, GOT-10k, and TrackingNet test sets. For GOT-10k, an asterisk (*) denotes models trained exclusively on its training set. The top two results are highlighted with **bold** and underlined fonts, respectively.

	ARTrackV2-L384	HIPTrack-B384	ODTrack-B384	ARTrack-B384	SeqTrack-B384	DropTrack-B384	STARK	DiMP	ATOM	**MUTrack-B384**
NfS	68.4	68.1	-	66.8	66.7	-	-	-	-	**69.7**
OTB100	-	71.0	72.3	-	-	69.6	68.5	68.4	66.3	**72.8**

Table 3: Comparison with state-of-the-art methods on NfS and OTB100 benchmarks in AUC score. The top two results are highlighted with **bold** and underlined fonts, respectively.

#	Method	GOT-10K		LaSOT	
		AUC	SR$_{0.50}$	AUC	P$_{norm}$
1	basline	72.9	83.9	70.9	80.9
2	+ STM	74.8	84.9	71.4	81.5
3	+ spatial prior mask	75.6	86.8	71.7	81.7
4	+ LTM	**78.1**	**88.3**	**72.3**	**82.9**

Table 4: Ablation Study on Model Components

Encoder	AO	SR$_{0.50}$	SR$_{0.75}$
ViT-B	76.7	87.0	75.6
HiViT-B	**78.1**	**88.3**	**77.5**

Table 5: Ablation on different encoders on GOT-10k.

Comparison with the State-of-the-Art

We compare our MUTrack with SOTA trackers across six tracking benchmarks, as detailed in Tab. 2 and Tab. 3.

GOT-10k. GOT-10k(Huang, Zhao, and Huang 2021) is a large-scale tracking benchmark over 10,000 training and 180 test sequences. Its strict protocol requires trackers to use only its training data without overlapping object classes. Our MUTrack, trained exclusively on GOT-10k, outperforms mainstream trackers (Tab. 2). Even MUTrack-256 surpasses ARTrackV2-384, showing the benefits of LTM and STM feature enhancement.

LaSOT. LaSOT(Fan et al. 2019) is a key benchmark for long-term tracking evaluation, featuring 280 test videos with over 2,500 frames each to assess tracker robustness. As shown in Tab. 2, MUTrack-384 outperforms other mainstream trackers (e.g., ARTrackV2-384) across all metrics.

LaSOT$_{ext}$. LaSOT$_{ext}$(Fan et al. 2020) augments LaSOT with 150 new long-term sequences across 15 novel categories, introducing challenges like occlusions and similar distractors. As shown in Tab. 2, our tracker achieves competitive performance against state-of-the-art methods.

TrackingNet. TrackingNet(Müller et al. 2018) is a large-scale, diverse tracking dataset covering varied object classes and scenarios. Its 511-sequence test set with public ground truth provides a robust benchmark for tracker evaluation. Tab. 2 shows our MUTrack-384 achieves state-of-the-art performance, demonstrating strong generalization in real-world tracking.

OTB100 and NfS. OTB100(Wu, Lim, and Yang 2015) is a popular short-term tracking benchmark with 100 sequences covering 11 challenges like deformation and occlusion. NfS(Galoogahi et al. 2017) contains 100 videos (380K

GOT-10K	(a) Effect of the sampling strategies			(b) Effect of the number of reference frames				(c) Effect of the number of SME layers			
	Random	*Uniform*	*Recent-Consecutive*	2	3	4	5	2	3	4	5
AO	77.4	78.1	76.3	77.2	78.1	77.4	76.9	77.3	78.1	77.5	77.2
$SR_{0.50}$	87.9	88.3	87.7	87.9	88.3	88.1	87.3	88.0	88.3	88.2	87.5
$SR_{0.75}$	77.1	77.5	75.3	77.7	77.5	77.4	76.9	77.8	77.5	77.7	77.1

Table 6: Ablation studies on different components on the GOT-10K benchmark.

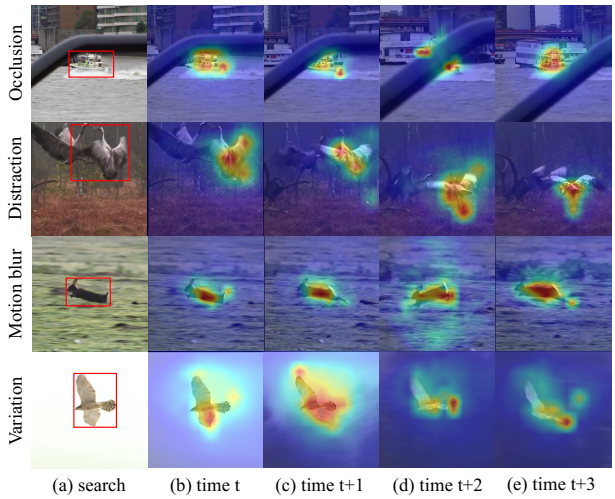

Figure 6: **Attention visualization.** (a) Search region(The red boxes represent ground truth of the target.) (b)-(e) show the attention maps from the last layer of the cross-attention in the Perception Interaction Module.

queue length	AO	$SR_{0.50}$	$SR_{0.75}$
2	76.7	86.8	76.4
4	**78.1**	**88.3**	**77.5**
8	77.2	87.9	77.7

Table 7: Effect of the short-term memory queue length.

frames). As shown in Tab.3, our tracker achieves SOTA performance, with MUTrack-B256 outperforming higher-resolution trackers due to LTM/STM feature enhancement.

Ablation Studies

Study on different model components. We conducted comprehensive ablation studies on the GOT-10k dataset (Tab. 4). Our baseline achieved 72.9% AO. Incorporating Short-Term Memory (STM) significantly boosted performance to 74.8% AO. Adding a spatial prior mask further increased it to 75.6%. Our full model, integrating LTM, finally achieved 78.1% AO, confirming richer target representation from historical frames.

Study on different transformer encoder. We investigated the impact of different backbone networks on our memory-based framework (Tab. 5). HiViT significantly outperformed ViT as the backbone encoder. This gain is attributed to HiViT's hierarchical structure, which preserves finer spatial details through multi-stage downsampling, offering a multi-scale representation crucial for tracking.

Study on different sampling strategy. The sampling strategy for reference frames significantly impacts tracking performance. As shown in Tab. 6(a), uniform sampling achieves the best performance (78.1% AO) among random and recent-consecutive methods. This highlights the importance of maintaining a diverse yet temporally comprehensive representation of the target's appearance.

Study on the reference frame number. Tab. 6(b) investigates the impact of reference frame number. Performance peaks at 3 frames, showing a slight decline with more. This suggests that while minimal frames hinder appearance diversity, excessive frames introduce noise or redundancy.

Study on the number of SME layers. The Sparse Multi-scale Extractor efficiently processes the LTM. As shown in Tab. 6(c), performance peaks at SME layers of 3 and slightly declines with more, suggesting this depth is optimal for balancing feature richness and computational efficiency.

Study on the short-term memory queue length. The short-term memory queue length directly impacts temporal context for target adaptation. As Tab. 7 shows, a queue length of 4 yields optimal performance, indicating a moderate length optimally balances temporal context with recent, relevant appearance cues.

Attention visualization. PIM attention maps (Fig. 6) visually confirm MUTrack's robust target focus and the unified memory's effectiveness in handling various challenges. Our unified representation effectively addresses occlusions (first row) by using LTM's stable context to prevent attention drift and counter STM errors. It also enables target discrimination from distractors (second row), coping with motion blur (third row), and adapting to appearance variations (fourth row) while preserving identity. These visualizations confirm that integrating complementary LTM and STM information yields more reliable and adaptable tracking.

Conclusion

In this work, we presented MUTrack, a unified memory-based framework for robust visual tracking that constructs a highly adaptive, unified target representation. Its core lies in two synergistic components: a Unified Memory Bank managing long-term and short-term memories for rich temporal context; and a Perception Interaction Module that deeply fuses and refines their inherent complementarity. This synergistic design yields a highly adaptive target representation, expected to spur significant further research in video-level tracking and broader video understanding.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.U23A20383, 62472109 and 62466051), the Project of Guangxi Science and Technology (No.2025GXNSFAA069676, 2024GXNSFGA010001, and GuiKeFN2504240017) , the Guangxi "Young Bagui Scholar" Teams for Innovation and Research Project, the Research Project of Guangxi Normal University (No. 2025DF001).

References

Bai, Y.; Zhao, Z.; Gong, Y.; and Wei, X. 2024. Artrackv2: Prompting autoregressive tracker where to look and how to describe. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19048–19057.

Bertinetto, L.; Valmadre, J.; Henriques, J. F.; Vedaldi, A.; and Torr, P. H. 2016. Fully-convolutional siamese networks for object tracking. In *Computer vision–ECCV 2016 workshops: Amsterdam, the Netherlands, October 8-10 and 15-16, 2016, proceedings, part II 14*, 850–865. Springer.

Cai, W.; Liu, Q.; and Wang, Y. 2024. HIPTrack: Visual Tracking with Historical Prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 19258–19267.

Chen, B.; Li, P.; Bai, L.; Qiao, L.; Shen, Q.; Li, B.; Gan, W.; Wu, W.; and Ouyang, W. 2022. Backbone is all your need: A simplified architecture for visual object tracking. In *European conference on computer vision*, 375–392. Springer.

Chen, X.; Peng, H.; Wang, D.; Lu, H.; and Hu, H. 2023. Seqtrack: Sequence to sequence learning for visual object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14572–14581.

Chen, X.; Yan, B.; Zhu, J.; Wang, D.; Yang, X.; and Lu, H. 2021. Transformer Tracking. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Cui, Y.; Jiang, C.; Wang, L.; and Wu, G. 2022. Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13608–13618.

Danelljan, M.; Bhat, G.; Shahbaz Khan, F.; and Felsberg, M. 2017. Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6638–6646.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fan, H.; Bai, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Harshit, H.; Huang, M.; Liu, J.; Xu, Y.; Liao, C.; Lin, Y.; and Ling, H. 2020. LaSOT: A High-quality Large-scale Single Object Tracking Benchmark. *International Journal of Computer Vision,International Journal of Computer Vision*.

Fan, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Bai, H.; Xu, Y.; Liao, C.; and Ling, H. 2019. LaSOT: A High-quality Benchmark for Large-scale Single Object Tracking. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Fu, Z.; Fu, Z.; Liu, Q.; Cai, W.; and Wang, Y. 2022. SparseTT: Visual tracking with sparse transformers. *arXiv preprint arXiv:2205.03776*.

Fu, Z.; Liu, Q.; Fu, Z.; and Wang, Y. 2021. Stmtrack: Template-free visual tracking with space-time memory networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13774–13783.

Galoogahi, H. K.; Fagg, A.; Huang, C.; Ramanan, D.; and Lucey, S. 2017. Need for Speed: A Benchmark for Higher Frame Rate Object Tracking. In *2017 IEEE International Conference on Computer Vision (ICCV)*.

Ge, J.; Cao, J.; Chen, X.; Zhu, X.; Liu, W.; Liu, C.; Wang, K.; and Liu, B. 2025a. Beyond visual cues: Synchronously exploring target-centric semantics for vision-language tracking. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(5): 1–21.

Ge, J.; Cao, J.; Zhu, X.; Zhang, X.; Liu, C.; Wang, K.; and Liu, B. 2024. Consistencies are all you need for semi-supervised vision-language tracking. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 1895–1904.

Ge, J.; Zhang, X.; Cao, J.; Zhu, X.; Liu, W.; Gao, Q.; Cao, B.; Wang, K.; Liu, C.; Liu, B.; et al. 2025b. Gen4Track: A Tuning-free Data Augmentation Framework via Self-correcting Diffusion Model for Vision-Language Tracking. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 3037–3046.

He, K.; Zhang, C.; Xie, S.; Li, Z.; and Wang, Z. 2023. Target-aware tracking with long-term context attention. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 773–780.

Hu, X.; Tai, Y.; Zhao, X.; Zhao, C.; Zhang, Z.; Li, J.; Zhong, B.; and Yang, J. 2025. Exploiting multimodal spatial-temporal patterns for video object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 3581–3589.

Hu, X.; Zhong, B.; Liang, Q.; Zhang, S.; Li, N.; and Li, X. 2024. Toward modalities correlation for RGB-T tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(10): 9102–9111.

Hu, X.; Zhong, B.; Liang, Q.; Zhang, S.; Li, N.; Li, X.; and Ji, R. 2023. Transformer tracking via frequency fusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(2): 1020–1031.

Huang, L.; Zhao, X.; and Huang, K. 2021. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1562–1577.

Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; and Yan, J. 2019. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4282–4291.

Li, N.; Zhong, B.; Zheng, Y.; Liang, Q.; Mo, Z.; and Song, S. 2024. Robust tracking via combing top-down and bottom-up attention. *IEEE Transactions on Circuits and Systems for Video Technology*.

Li, X.; Zhong, B.; Liang, Q.; Li, G.; Mo, Z.; and Song, S. 2025. Mambalct: Boosting tracking via long-term context state space model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 4986–4994.

Lin, L.; Fan, H.; Zhang, Z.; Wang, Y.; Xu, Y.; and Ling, H. 2024. Tracking meets lora: Faster training, larger model, stronger performance. In *European Conference on Computer Vision*, 300–318. Springer.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollar, P. 2017. Focal Loss for Dense Object Detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*.

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

Loshchilov, I.; and Hutter, F. 2017. Decoupled Weight Decay Regularization. *Learning,Learning*.

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

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shi, L.; Liu, T.; Hu, X.; Hu, Y.; Yin, Q.; and Hong, R. 2025a. SwimVG: Step-wise Multimodal Fusion and Adaption for Visual Grounding. *arXiv preprint arXiv:2502.16786*.

Shi, L.; Zhong, B.; Liang, Q.; Hu, X.; Mo, Z.; and Song, S. 2025b. Mamba Adapter: Efficient Multi-Modal Fusion for Vision-Language Tracking. *IEEE Transactions on Circuits and Systems for Video Technology*.

Wang, B.; Li, W.; and Ge, J. 2025. R1-Track: Direct Application of MLLMs to Visual Object Tracking via Reinforcement Learning. arXiv:2506.21980.

Wang, J.; Liu, F.; Jiao, L.; Gao, Y.; Wang, H.; Li, S.; Li, L.; Chen, P.; and Liu, X. 2024a. Visual and language collaborative learning for RGBT object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*.

Wang, J.; Liu, F.; Jiao, L.; Wang, H.; Li, S.; Li, L.; Chen, P.; and Liu, X. 2024b. Multi-modal visual tracking based on textual generation. *Information Fusion*, 112: 102531.

Wang, J.; Liu, F.; Jiao, L.; Wang, H.; Li, S.; Li, L.; Chen, P.; Liu, X.; and Wang, X. 2025. FA3T: Feature-Aware Adversarial Attacks for Multi-modal Tracking. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 1376–1385.

Wu, Y.; Lim, J.; and Yang, M.-H. 2015. Object Tracking Benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1834–1848.

Xie, J.; Zhong, B.; Liang, Q.; Li, N.; Mo, Z.; and Song, S. 2025. Robust tracking via mamba-based context-aware token learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 8727–8735.

Xie, J.; Zhong, B.; Mo, Z.; Zhang, S.; Shi, L.; Song, S.; and Ji, R. 2024. Autoregressive Queries for Adaptive Tracking with Spatio-Temporal Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19300–19309.

Xue, C.; Zhong, B.; Liang, Q.; Xia, H.; and Song, S. 2024. Unifying Motion and Appearance Cues for Visual Tracking via Shared Queries. *IEEE Transactions on Circuits and Systems for Video Technology*.

Xue, C.; Zhong, B.; Liang, Q.; Zheng, Y.; Li, N.; Xue, Y.; and Song, S. 2025. Similarity-guided layer-adaptive vision transformer for UAV tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 6730–6740.

Yan, B.; Peng, H.; Fu, J.; Wang, D.; and Lu, H. 2021. Learning Spatio-Temporal Transformer for Visual Tracking. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Yang, T.; and Chan, A. B. 2018. *Learning Dynamic Memory Networks for Object Tracking*, 153–169.

Ye, B.; Chang, H.; Ma, B.; Shan, S.; and Chen, X. 2022. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision*, 341–357. Springer.

Zeng, F.; Zhong, B.; Xia, H.; Tan, Y.; Hu, X.; Shi, L.; and Song, S. 2025. Explicit context reasoning with supervision for visual tracking. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 8067–8076.

Zhang, L.; Gonzalez-Garcia, A.; Weijer, J. V. D.; Danelljan, M.; and Khan, F. S. 2019. Learning the Model Update for Siamese Trackers. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Zhang, X.; Tian, Y.; Xie, L.; Huang, W.; Dai, Q.; Ye, Q.; and Tian, Q. 2023. Hivit: A simpler and more efficient design of hierarchical vision transformer. In *The eleventh international conference on learning representations*.

Zhang, Z.; Peng, H.; Fu, J.; Li, B.; and Hu, W. 2020. Ocean: Object-aware anchor-free tracking. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 771–787. Springer.

Zheng, Y.; Zhong, B.; Liang, Q.; Li, G.; Ji, R.; and Li, X. 2023. Toward unified token learning for vision-language tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(4): 2125–2135.

Zheng, Y.; Zhong, B.; Liang, Q.; Mo, Z.; Zhang, S.; and Li, X. 2024. Odtrack: Online dense temporal token learning for visual tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7588–7596.

Zheng, Y.; Zhong, B.; Liang, Q.; Tang, Z.; Ji, R.; and Li, X. 2022. Leveraging local and global cues for visual tracking via parallel interaction network. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(4): 1671–1683.

Zheng, Y.; Zhong, B.; Liang, Q.; Zhang, S.; Li, G.; Li, X.; and Ji, R. 2025. Towards universal modal tracking with online dense temporal token learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.