MambaVLT: Time-Evolving Multimodal State Space Model for Vision-Language Tracking

Xinqi Liu[1†], Li Zhou[1†], Zikun Zhou[2*], Jianqiu Chen[1], and Zhenyu He[1,*]
[1]Harbin Institute of Technology, Shenzhen [2]Pengcheng Laboratory
{xqliu01,lizhou.hit,zhouzikunhit,jianqiuer}@gmail.com zhenyuhe@hit.edu.cn

Abstract

The vision-language tracking task aims to perform object tracking based on various modality references. Existing Transformer-based vision-language tracking methods have made remarkable progress by leveraging the global modeling ability of self-attention. However, current approaches still face challenges in effectively exploiting the temporal information and dynamically updating reference features during tracking. Recently, the State Space Model (SSM), known as Mamba, has shown astonishing ability in efficient long-sequence modeling. Particularly, its state space evolving process demonstrates promising capabilities in memorizing multimodal temporal information with linear complexity. Witnessing its success, we propose a Mamba-based vision-language tracking model to exploit its state space evolving ability in temporal space for robust multimodal tracking, dubbed MambaVLT. In particular, our approach mainly integrates a time-evolving hybrid state space block and a selective locality enhancement block, to capture contextual information for multimodal modeling and adaptive reference feature update. Besides, we introduce a modality-selection module that dynamically adjusts the weighting between visual and language references, mitigating potential ambiguities from either reference type. Extensive experimental results show that our method performs favorably against state-of-the-art trackers across diverse benchmarks.

1. Introduction

Single object tracking involves localizing a target in a video based on provided reference information, which may be an initial bounding box [53], a language specification [31], or a combination of both [31]. This technology has diverse applications, including video surveillance, robotics, and autonomous vehicles. Tracking by initial bounding box [14, 33, 52] is an extensively studied tracking task.

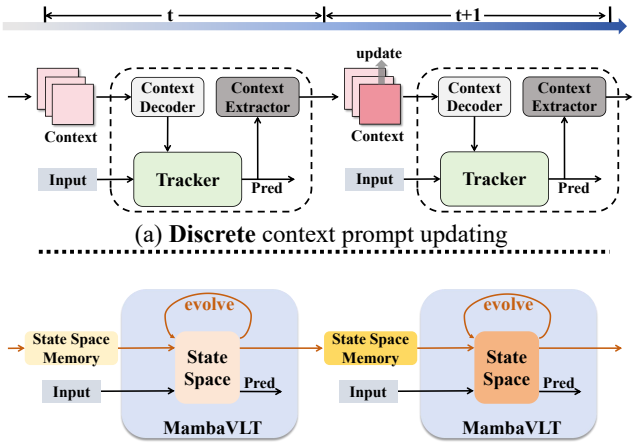

Figure 1. Illustration of two ways for capturing temporal context information. (a) Vision-language tracker with discrete context prompt. (b) Our MambaVLT with continuous time-evolving state space for temporal information transmission.

A common solution is cropping a template based on the bounding box in the first frame as a reference and accordingly locating the target in subsequent frames. Nonetheless, solely relying on the visual template without direct semantics may lead to ambiguity [46]. Recently, tracking by natural language specification [31] and tracking by both language and box specification [31] have been proposed to address this issue. These approaches incorporate language descriptions as references, facilitating natural human-computer interaction. Previous studies [37, 41, 58] have made significant progress under various reference settings. However, they are still constrained by their limited ability to capture long-term temporal information and adaptively update the reference information as the tracker operates on a video.

Typically, the appearance and motion mode of the target keep varying in the video. Previous works [37, 41, 58] have introduced different methods to adapt to these tempo-

[†]Xinqi Liu and Li Zhou contribute equally. [*]Zikun Zhou and Zhenyu He are Corresponding authors.

ral variations, which can be summarized as a discrete approach to extracting contextual features, as shown in Figure 1(a). It can be divided into two steps: (1) generating a context prompt by a context extractor based on the bounding box prediction; and (2) decoding target information from the context prompt with a context decoder. This approach extracts and updates context prompts discretely without explicit cross-frame correlation and highly depends on the accuracy of predictions, which may result in error accumulation and insufficient modeling of the target varying patterns. Furthermore, in vision-language tasks, there are multimodal references, yet most methods focus only on updating visual references, lacking an effective approach for jointly updating language and visual information.

Recently, the state space models, advanced by the LSSL [17], S4 [16], GSS [39], and S4D [18] have demonstrated exceptional performance in long-sequence modeling. Particularly, Mamba [15], which uses selective variables to autoregressively model the sequential evolving neural states, has been noticed as a compelling alternative to Transformers on large-scale data. Yet, the utilization of state space for temporal multimodal feature modeling and updating is still under-investigated.

To jointly retain temporal information and update reference features adaptively, we explore the evolving process of Mamba's state space, in where MambaVLT memorizes long-term historical target features, and by which the model selectively updates the reference features. As shown in Figure 1(b), our model captures temporal information through a continuously evolving state-space memory. Since Mamba autoregressively processes input sequences with the state space, the final state space in each layer inherently contains global features. Based on this observation, we design a state space memory and a state space evolving strategy to retain long-term multimodal information throughout the whole video. The evolved state space will be utilized to update the reference features adaptively. Compared with the method in Figure 1(a), this approach not only enables context-aware multimodal modeling and reference feature updating but also provides a more elegant solution without extra network components.

Overall, MambaVLT introduces a time-evolving multimodal fusion module, which integrates a Hybrid Multimodal State Space (HMSS) block and a Selective Locality Enhancement (SLE) block. Each HMSS block consists of the temporal state space evolving mechanism and a modality-guided bidirectional scan. The temporal state space will transmit historical target features as shown in Figure 1(b), while the modality-guided scan can dynamically update and fuse various modality features through different scan orders. After the HMSS module performs global modeling of cross-frame information, the SLE module will enhance the intra-modal dependency and inter-modal correlation of the current tracking frame through global receptiveness. Afterward, we present a modality-selection module to dynamically weigh the different modality reference features for search region feature refining to distinguish the reliability of different references at various time stamps.

Moreover, to analyze the effectiveness of state space memory and its ability to memorize long-term target information, we design a new tracking paradigm called semi-reference-free tracking, which aims to track without reference data input from the second search image in a video. To conclude, the main contributions of our work are:

- We introduce MambaVLT, the first Mamba-based vision-language tracker, which is able to exploit the temporal information and update the reference features effectively and efficiently.
- We present a time-evolving multimodal fusion module that not only memorizes long-term target information for cross-frame information modeling and reference feature updating but also enhances the internal multimodal correlation of the current tracking frame.
- We conduct extensive experiments on the TNL2K [46], LaSOT [8], OTB99 [31], and MGIT [24] benchmarks, which demonstrate the effectiveness of our MambaVLT.

2. Related Work

2.1. Vision-language Tracking

In the vision-language tracking task, there are three different reference settings: only target bounding box, only natural language, or both of them. Visual reference can provide more direct guidance, while natural language description can reveal details about the appearance and changes of the object over time.

Tracking by Initial Bounding Box aims to continuously track a target throughout a video sequence based on the initial bounding box provided in the first frame. Siamese-based trackers [2, 26, 27, 49, 59] utilize Siamese networks to extract visual features and locate targets by a matching module. To learn the historical changes of the target, Some trackers [3, 6, 13, 23, 40, 44, 50, 60] use previous prediction results for template update. TransT [4] introduces the Transformer architecture for visual tracking and achieves promising results. In addition, OSTrack [52] and MixFormer [5] construct simplified one-stream tracking pipelines with superior performance. However, tracking based on purely visual inference may lead to ambiguity in object identification. To make full use of temporal context, ODTrack [57] and LMTrack [48] have made significant improvements in tracking based on long-term contextual association strategy.

Tracking by Natural Language Specification presents a unique approach with a more natural human-computer interaction way, which specifies the target with purely natural

language description. Li *et al* [31] first defines this task and validates its effectiveness. They propose a paradigm conducting TNL task with separate grounding and tracking models, which was adopted by following works [30, 46, 51]. JointNLT [58] proposes a Transformer-based framework that unifies visual grounding and tracking and outperforms state-of-the-art algorithms. QueryNLT [41] then presents a context-aware fusion of visual and language references through query interactions to address the misalignment between language and visual information.

Tracking by Language and Box Specification specifies the target with a bounding box and natural language description. The work of Li *et al* [31] demonstrates that combining natural language description and initial target bounding box can enhance the tracking performance. SNLT [11] and VLT [20] utilize natural language as an extra enhancement to aid visual features for tracking. Besides, both JointNLT and QueryNLT demonstrate great performance in this task. MMTrack [56] proposes an autoregressive tracking framework to simplify the model and achieves promising performance. Recently, UVLTrack [37] proposes a unified Transformer-based architecture that can simultaneously model the above three reference settings. However, due to the inherent computation mechanisms of CNN and Transformer, the aforementioned methods struggle to learn long-range temporal information.

2.2. State Space Models

State space models (SSMs) [12, 15, 16, 43] have gained much attention because of their promising potential in long sequence modeling. Initially, the Structured State Space Sequence Model (S4) [16] was proposed to model long-range dependencies in linear complexity. Based on S4, subsequent works including S5 [43], H3 [12], and Mamba [15] were proposed to improve the ability and efficiency of the model. Especially, Mamba outperforms Transformers in several long sequence NLP tasks with linear scalability due to its data-dependent selective state space mechanism and hardware implementation.

For the strong potential of Mamba in long sequence modeling, a series of outstanding works [22, 34, 42, 47, 54, 61] have been proposed in the visual domain. Vim [61] and Vmamba [34] adapt Mamba to visual classification tasks and released reliable pretrained models. They employ multidirectional scans to model the visual data. For video modeling, VideoMamba [28] applies S6 by concatenating 1D image sequences in temporal order. MambaIR [19] is the first to transfer the S6 model to the image restoration field. MTMamba [32] designed a Mamba-based dual-stream architecture for multi-task learning. CoupledMamba [29] conducts multimodal fusion with coupling state chains of different modalities.

Leveraging the autoregressive computation manner of Mamba, we propose a time-evolving mechanism to retain long-term target information, based on which a time-evolving multimodal fusion module is introduced to adaptively update reference features in the tracking. Meanwhile, a modality-selection module is designed to weigh the vision-language features for search region feature refining.

3. MambaVLT

3.1. Preliminaries: SSM and Mamba

State Space Model (SSM). SSM is a continuous system that maps input sequence $x(t) \in \mathbb{R}$ to output sequence $y(t) \in \mathbb{R}$ with hidden state space $h(t) \in \mathbb{R}^N$, which can be formulated as follows:

$$h'(t) = \mathbf{A}h(t) + \mathbf{B}x(t), \quad y(t) = \mathbf{C}h(t). \tag{1}$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are state transition matrices. The discrete counterpart, *i.e.*, discrete SSM, utilizes zero-order hold discretization with a timescale parameter $\mathbf{\Delta}$ to transform continuous parameters $\mathbf{A}$ and $\mathbf{B}$ into discrete parameters $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$:

$$\begin{aligned} \overline{\mathbf{A}} &= \exp(\mathbf{\Delta A}), \\ \overline{\mathbf{B}} &= (\mathbf{\Delta A})^{-1}(\exp(\mathbf{\Delta A}) - \mathbf{I}) \cdot \mathbf{\Delta B}. \end{aligned} \tag{2}$$

Selective State Space Model (Mamba). The above SSMs are still static systems for various inputs because of their data-independent parameters, which limits their ability to dynamically model sequences. To this end, Mamba generates $\mathbf{A}_i, \mathbf{B}_i, \mathbf{\Delta}_i$ based on the i^{th} input x_i. The selective state space model can be written as:

$$\begin{aligned} \overline{\boldsymbol{A}}_i &= \exp\left(\Delta_i \boldsymbol{A}\right), \\ \overline{\boldsymbol{B}}_i &= \Delta_i \boldsymbol{B}_i, \\ \boldsymbol{h}_i &= \overline{\boldsymbol{A}}_i \boldsymbol{h}_{i-1} + \overline{\boldsymbol{B}}_i x_i, \\ y_i &= \boldsymbol{C}_i \boldsymbol{h}_i + \boldsymbol{D} x_i. \end{aligned} \tag{3}$$

3.2. Overall Framework

As shown in Figure 2, the proposed MambaVLT is capable of jointly modeling different modality reference settings including the initial bounding box, natural language, or both. Firstly, we utilize a separate vision and language encoder for preliminary feature extraction. The input language description l will be projected to language feature $F_l \in \mathbb{R}^{N_l \times C}$ with pretrained Mamba-based text encoder [15]. In particular, we use a template video clip to capture the appearance changes of the target explicitly. For the template video clip $z \in \mathbb{R}^{L \times 3 \times H_z \times W_z}$ and search region $x \in \mathbb{R}^{3 \times H_x \times W_x}$, they will be processed by shared Vmamba-based visual encoder [34] to obtain template feature $F_z \in \mathbb{R}^{N_z \times \mathrm{C_v}}$ and search region feature $F_x \in \mathbb{R}^{N_x \times \mathrm{C_v}}$. L is the number of frames in the template

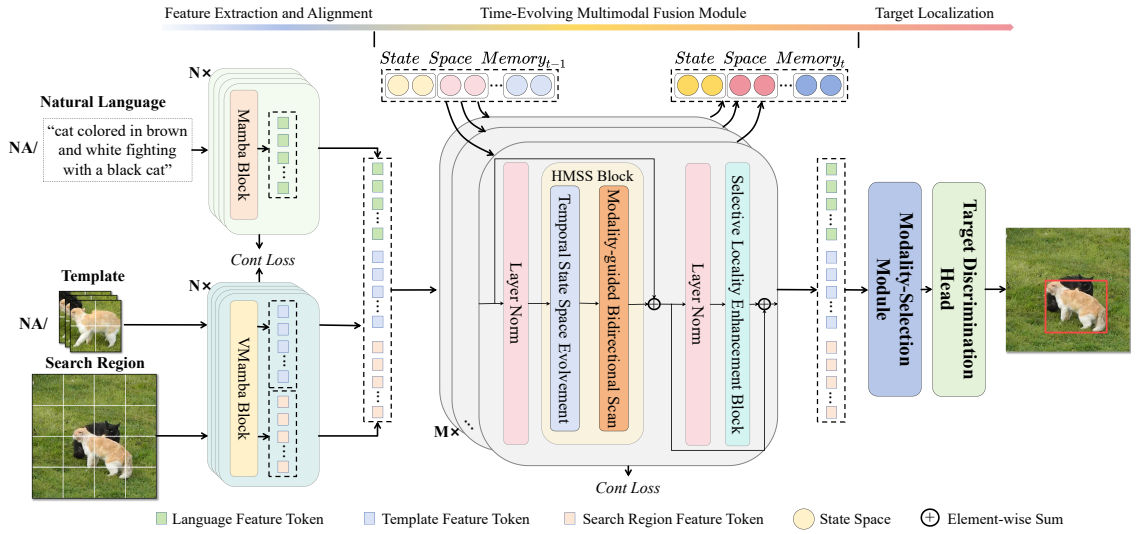

Figure 2. Overview of the MambaVLT. Given various modality reference settings, features are initially extracted and aligned, then forwarded to the time-evolving multimodal fusion module. Subsequently, these features are input into the localization module to obtain precise localization information. MambaVLT performs temporal information-aware vision-language tracking with adaptive reference feature updating. Note that 'NA' indicates when the corresponding reference is not provided.

video clip. Then, the language, template, and search region features will be concatenated to a unified 1D sequence G for the time-evolving multimodal fusion, which will capture historical target information for reference features update and unified multimodal modeling.

Then the Time-Evolving Multimodal Fusion Module is introduced to model multimodal features in temporal space, in which the Hybrid Multimodal State Space Block will model long-term target information and further update the reference features. With the updated reference features, the modality-selection module dynamically weighs and selects them to alleviate potential ambiguities arising from either reference type. Finally, the target discrimination head fully exploits the target and background information embedded in the search region feature to locate the target accurately. Additionally, the prediction head calculates a confidence score for each prediction to update the template video clip.

Moreover, we propose an intra-video and inter-video multimodal contrastive loss to align multimodal features during the feature fusion stages. We firstly extract reference token T with mean pooling operation based on reference features to calculate token-wise similarity s_i with positive samples and negative samples to enhance the discriminative ability of features:

$$s^i = \frac{\mathbf{T}\left(f^i\right)^\top}{\|\mathbf{T}\|_2 \left\|f^i\right\|_2}. \tag{4}$$

For intra-video contrastive loss $\mathcal{L}_w$, the positive sample is the target center token of the search region feature, and negative samples are N_w^n most similar tokens in the search region background. For inter-video contrastive loss $\mathcal{L}_o$, the positive sample is the same as intra-video loss, while the negative samples are N_o^n target center tokens from search regions of other video sequences.

3.3. Time-Evolving Multimodal Fusion Module

Temporal information is crucial for dynamically adapting to target variations in vision-language tracking. Previous Transformer-based models mainly retain context information in a discrete manner. For continuous long-term target feature retention, MambaVLT presents a Time-evolving Multimodal Fusion (TEMF) module by harnessing the potential of the state space to enable unified feature modeling and adaptive reference information updating.

The TEMF module mainly consists of a Hybrid Multimodal State Space (HMSS) block and a Selective Locality Enhancement (SLE) block. Given a unified multimodal sequence G, the HMSS block first captures long-term temporal information by the time-evolving state space, based on which it models multimodal features and updates target reference information by a modality-guided bidirectional scan. After global cross-frame feature modeling, the SLE module performs a sliding window scan with a selective map A_l to enhance multimodal features of the current time stamp through a global receptiveness. Formally,

$$G' = \boldsymbol{\phi_{SLE}}\left(\left(\boldsymbol{\phi_{HMSS}}\left(G\right)\right)\right). \tag{5}$$

Hybrid Multimodal State Space Block. As shown in Figure 3, the Hybrid Multimodal State Space (HMSS) block integrates the temporal hybrid state space evolving mech-

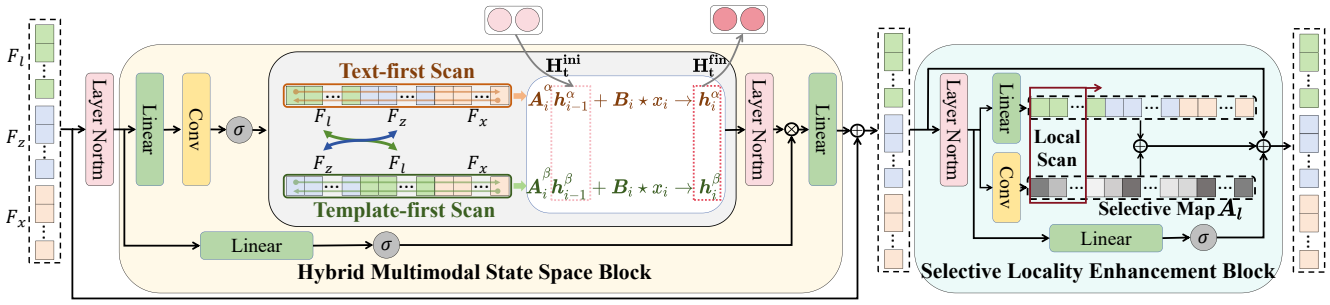

Figure 3. Overall pipeline of the Hybrid Multimodal State Space Block. The multimodal feature includes language feature F_l, template feature F_z and search region feature F_X. The Hybrid Multimodal State Space block is for time-evolving global modeling and reference feature updating. Then, the Selective Locality Enhancement block will enhance the features of the current tracking frame. $\mathbf{H}_t^{ini}$ and $\mathbf{H}_t^{fin}$ denote the initial state space and final state space. local scan represents the linear attention scan. $\boldsymbol{A_l}$ represents the global selective map.

anism for temporal information retention and a modality-guided directional scan for dynamic features update.

In the temporal hybrid state space evolving mechanism, we construct a multi-level state space memory $SS = \{\{\mathbf{H}_{t-1}^{fin_i,\alpha}, \mathbf{H}_{t-1}^{fin_i,\beta}\} | i \in 1, 2, ..., M\}$ to store different final state space $\mathbf{H}^{fin}$ of TEMF modules, where α and β denote text-first and template-first scan. M is the number of TEMF modules. Since Mamba processes sequences with the state space autoregressively, the final state space will inherently contain global information of processed tokens. As the state space memory processes each frame of the video sequence and updates the template video clip, the state space memory evolves temporally and memorizes long-term target information naturally. At the prior of each HMSS module, we derive the initial state space from the state space memory and a learnable state space $\mathbf{H}^l$:

$$\mathbf{H}_t^{ini} = a\mathbf{H}^l + (1-a)\mathbf{H}_{t-1}^{fin}. \tag{6}$$

where $\mathbf{H}_{t-1}^{fin}$ represents the final state space of the last time stamp in state space memory. a is the trade-off parameter.

Captured by the temporal target information by multi-level state space memory, the HMSS block will perform a modality-guided bidirectional scan to adaptively update reference features and fuse multimodal information, which is based on the prior insight that different scan orders in Mamba will influence the modality feature. As shown in Figure 3, the HMSS block will conduct bidirectional scans based on text-first order α and template-first order β, in which search region feature will always be placed at the end of the sequence to gather reference information. By changing the order of the text and template features, they will serve as guiding information to direct the update and fusion of the features respectively. Different from previous multi-direction scan methods [28, 61] which use completely different parameters for multidirectional scan, HMSS block mainly utilizes shared parameters including $\overline{\boldsymbol{B}}, \boldsymbol{C}$ and $\boldsymbol{D}$ to reduce parameter redundancy and model the overall perception of target information. We use distinct parameters $\overline{\boldsymbol{A}}$ as state space update gates in different scan orders to adaptively update reference features and model search region features. This process can be formulated as:

$$\begin{aligned} \boldsymbol{h}_i^{\alpha} &= \overline{\boldsymbol{A}}_i^{\alpha} \boldsymbol{h}_{i-1}^{\alpha} + \overline{\boldsymbol{B}}_i \circ x_i, \\ \boldsymbol{h}_i^{\beta} &= \overline{\boldsymbol{A}}_i^{\beta} \boldsymbol{h}_{i-1}^{\beta} + \overline{\boldsymbol{B}}_i \star x_i, \\ y_i &= (\boldsymbol{C}_i \circ \boldsymbol{h}_i^{\alpha} + \boldsymbol{D} \circ x_i + \boldsymbol{C}_i \star \boldsymbol{h}_i^{\beta} + \boldsymbol{D} \star x_i)/2. \end{aligned} \tag{7}$$

The $\overline{\boldsymbol{B}}, \boldsymbol{C}$ and $\boldsymbol{D}$ are control gates for overall feature extraction. $\overline{\boldsymbol{A}}_i^{\alpha}$ and $\overline{\boldsymbol{A}}_i^{\beta}$ are utilized for language-guided α and template-guided β autoregressive feature update. $\circ$ and $\star$ denote the Hadamard product performed in the text-first and template-first scanning.

Selective Locality Enhancement Block. After HMSS performs global modeling of multimodal information on cross-frame temporal dimension, we introduce a Selective Locality Enhancement (SLE) block to enhance features of the current time stamp. Linear attention is known for reducing the computation cost of softmax attention to O(N). Inspired by previous linear attention work [1, 21] and the carefully designed architecture of Mamba, we propose a 1D local scan method with global selective receptiveness. In classic linear attention, only top layers have access to nearly global representation [1] which is critical to multimodal correlation modeling. To address this issue, as shown in the right part of Figure 3, we introduce a global selective map $\boldsymbol{A}_l$ to enhance the global perception capability of the SLE block. $\boldsymbol{A}_l$ is obtained by performing a convolution operation on the output of the HMSS block to extract the inherent global selective information in the HMSS block. Then the sequence added with a global selective map will be enhanced by linear attention scan $\boldsymbol{\gamma}$. Moreover, the SLE block employs several Mamba-like control gates including $\boldsymbol{B}_l$ and $\boldsymbol{D}_l$ to process the sequence. The SLE block can be formulated as:

$$\begin{aligned} \boldsymbol{h}_t &= \boldsymbol{A}_l + \boldsymbol{B}_l G, \\ G' &= \boldsymbol{\gamma}(\boldsymbol{h}_l) + \boldsymbol{D}_l G. \end{aligned} \tag{8}$$

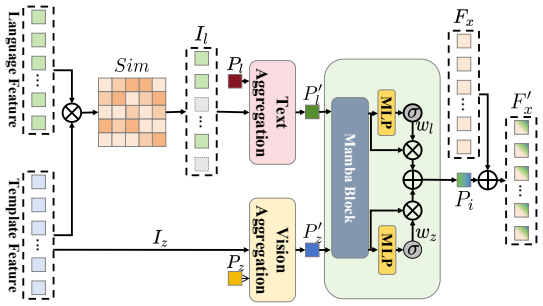

Figure 4. Overview of modality-selection module. w_l and w_z represents the weights of language invariant clue P_l and template invariant clue P_z.

$\boldsymbol{B}_l, \boldsymbol{D}_l$ denote the input gate and residual gate. $\boldsymbol{\gamma}$ represents the sliding window linear attention scan to enhance the intra-modal dependency. $\boldsymbol{A}_l$ is responsible for extracting the selective information as a global map. In this manner, the SLE block can selectively enhance different reference features and search region features while maintaining linear computational complexity.

3.4. Modality-Selection Module

In the time-evolving multimodal fusion module, the search region feature dynamically interacts with multimodal temporal features in a different fixed order. However, in different tracking frames, the reliability of the language and template features may vary due to the target motion and appearance changes. Therefore, we further employ a Modality-selection module to selectively fuse multimodal reference features for search region feature refining. As shown in Figure 4, it will firstly extract invariant language information I_l and template information I_z. Because the template feature can naturally reflect the invariant target appearance feature, we compute the similarity matrix Sim between the language and template features, based on which we extract N language tokens with the highest visual similarity as the final invariant language information.

Subsequently, language and vision query decoders are introduced to aggregate the invariant language and vision target clue: P_l and P_z. A Mamba-based selective block is then employed to weigh the P'_l and P'_z for language and vision clues fusion. The selected invariant reference clue P_i will be used to refine the search region feature for more accurate target localization.

3.5. Training Objective

The contrastive loss is consist of intra-video contrastive loss $\mathcal{L}_{c_w}$ and inter-video contrastive loss $\mathcal{L}_{c_o}$. Formally,

$$\mathcal{L}_c^i = -\log\left(\frac{e^{s_c^p}}{e^{s_c^p} + \sum_{k=1}^{N_c^n} e^{s_c^{n_k}}}\right). \tag{9}$$

Token-wise similarity s is calculated based on Equation 4. A binary cross-entropy loss is employed for target score map loss $\mathcal{L}_{tgt}$, whose groundtruth is generated based on the bounding box. We utilize the same training objectives of center score map $\mathcal{L}_{cls}$ and bounding box loss $\mathcal{L}_{bbox} = \lambda_1\mathcal{L}_1 + \lambda_{giou}\mathcal{L}_{giou}$ as OSTrack [52]. λ denotes different loss weights. The whole training objectives can be summarized as:

$$\mathcal{L} = \lambda_{bbox}\mathcal{L}_{bbox} + \lambda_{tgt}\mathcal{L}_{tgt} + \lambda_{cls}\mathcal{L}_{cls} + \lambda_{c_w}\mathcal{L}_{c_w} + \lambda_{c_o}\mathcal{L}_{c_o}. \tag{10}$$

4. Experiments

4.1. Implementation Details

Network Configuration. For vision inputs, the template and search region size are set to be 128×128 and 256×256. In the grounding task, the search region is resized such that its long edge is equal to 256. For the tracking task, the template is cropped from the first image and the scale factor is 2. The search region is cropped based on the last prediction bounding box and the scale factor is 4. For language inputs, the maximum length is 48. We utilize the first 4 layers of Mamba-130m [15] as text encoder and the 4-stage Vmamba-tiny [34] variant as visual encoder, where the downsample of Vmamba is modified to $\frac{1}{16}$.

Training Details. We utilize the official training splits of OTB99 [31], LaSOT [8], TNL2K [46], MGIT [24], RefCOCOg-google [38], and GOT-10k [25] to train our model. We use Adam to optimize the model and the learning rate is 0.0005. The weight decay coefficient is 0.05. We train our model for 300 epochs. The batch size is 8. We utilize common data augmentation methods including horizontal flip, translation, and color jittering.

4.2. The Analysis of State Space

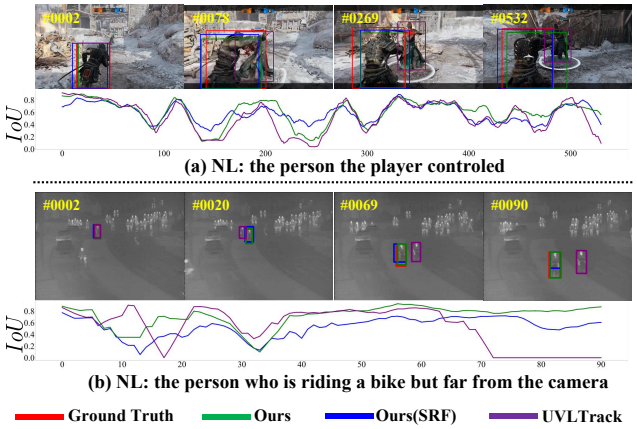

Figure 5. Qualitative comparison of NL&BBOX tracking task on two challenging sequences to analyze the effectiveness of state space. The line graphs represent the IoU of different trackers for each frame. The SRF means semi-reference-free tracking setting.

Table 1. Comparison of our method with state-of-the-art approaches on TNL2k, LaSOT and OTB99 datasets. The best and second-best results are highlighted in red and blue respectively.

Tracker	Reference	TNL2K			LaSOT			OTB99		
		AUC	*Prec*	*N Prec*	*AUC*	*Prec*	*N Prec*	*AUC*	*Prec*	*N Prec*
SiamRPN++ [27]	BBOX	41.3	41.2	48.0	49.6	49.1	56.9	-	-	-
AutoMatch [55]	BBOX	47.2	43.5	-	58.3	59.9	67.4	-	-	-
TrDiMP [45]	BBOX	52.3	52.8	-	63.9	61.4	-	-	-	-
TransT [4]	BBOX	50.7	51.7	-	64.9	69.0	73.8	-	-	-
SwinTrack-B [33]	BBOX	-	57.1	-	69.6	74.1	78.6	-	-	-
OSTrack-256 [52]	BBOX	54.3	-	-	69.1	75.2	78.7	-	-	-
GRM [14]	BBOX	-	-	-	69.9	75.8	79.3	-	-	-
UVLTrack-B [37]	BBOX	62.7	65.4	-	69.4	74.9	-	69.3	90.1	84.3
Ours	BBOX	63.3	65.8	87.5	65.0	69.5	76.6	71.6	92.9	87.4
TNLS-II [31]	NL	-	-	-	-	-	-	25.0	29.0	-
RVTNLN [9]	NL	-	-	-	-	-	-	54.0	56.0	-
RTTNLD [10]	NL	-	-	-	28.0	28.0	-	54.0	78.0	-
GTI [51]	NL	-	-	-	47.8	47.6	-	58.1	73.2	-
TNL2K-1 [46]	NL	11.4	6.4	11.0	51.1	49.3	-	19.0	24.0	-
CTRNLT [30]	NL	14.0	9.0	-	52.0	51.0	-	53.0	72.0	-
JointNLT [58]	NL	54.6	55.0	70.6	56.9	59.3	64.5	59.2	77.6	-
DecoupleTNL [36]	NL	40.7	40.0	-	64.9	67.1	-	69.5	92.8	-
QueryNLT [41]	NL	53.3	53.0	70.4	54.2	55.0	62.5	61.2	81.0	73.9
UVLTrack-B [37]	NL	55.7	57.2	-	57.2	61.0	-	60.1	79.1	-
Ours	NL	58.4	58.9	80.9	55.8	57.2	63.7	58.9	79.2	72.0
TNLS-III [31]	NL&BBOX	-	-	-	-	-	-	55.0	72.0	-
RVTNLN [9]	NL&BBOX	25.0	27.0	34.0	50.0	56.0	-	67.0	73.0	-
RTTNLD [10]	NL&BBOX	25.0	27.0	33.0	35.0	35.0	-	61.0	79.0	-
SNLT [11]	NL&BBOX	27.6	41.9	-	54.0	57.6	-	66.6	80.4	-
TNL2K-2 [46]	NL&BBOX	41.7	42.0	50.0	51.0	55.0	-	68.0	88.0	-
JointNLT [58]	NL&BBOX	56.9	58.1	73.6	60.4	63.6	69.4	65.3	85.6	79.5
DecoupleTNL [36]	NL&BBOX	56.7	56.0	-	71.2	75.3	-	73.8	94.8	-
MMTrack [56]	NL&BBOX	58.6	59.4	75.2	70.0	75.7	82.3	70.5	91.8	-
QueryNLT [41]	NL&BBOX	57.8	58.7	75.6	59.9	63.5	69.6	66.7	88.2	82.4
UVLTrack-B [37]	NL&BBOX	63.1	66.7	-	69.4	75.9	-	69.3	89.9	-
Ours	NL&BBOX	66.5	69.9	90.9	66.6	71.0	77.3	72.2	94.4	88.1

Table 2. Comparison of our method with the latest approaches on the MGIT dataset based on the official reproduction results.

Tracker	Reference	MGIT		
		AUC	*Prec*	*N Prec*
PrDiMP [7]	BBOX	-	29.6	60.2
TransT [4]	BBOX	-	44.7	67.0
OSTrack [52]	BBOX	-	47.6	70.6
GRM [14]	BBOX	-	50.0	71.8
Ours	BBOX	65.7	51.6	72.9
Ours	NL	64.6	50.3	71.2
SNLT [11]	NL&BBOX	-	0.4	22.6
VLT_SCAR [20]	NL&BBOX	-	11.6	35.4
VLT_TT [20]	NL&BBOX	-	31.8	60.2
JointNLT [58]	NL&BBOX	-	44.5	78.6
Ours	NL&BBOX	69.9	58.9	78.0

To analyze the effectiveness of state space memory, we design a new tracking paradigm called **semi-reference-free** (SRF) tracking. In the semi-reference-free tracking, the reference data (language or initial bounding box) is used by tracker **only** in the first frame. The tracker needs to extract and retain the target information embedded within the reference data and subsequently locate the target in search regions solely through the retained target information, without relying on reference data.

As shown in Figure 5, in the NL&BBOX tracking task, we conduct qualitative comparisons on two sequences of different trackers to analyze the ability of the proposed state space evolving mechanism. The main challenges with these two sequences are target fast movement and distractors, respectively. As shown in the line plot, MambaVLT with the semi-reference-free setting generally outperforms UVLTrack with normal settings, which demonstrates that the proposed state space memory can efficiently extract target information from references and retain long-term target information during the tracking process. As shown in Figure 5(b), our model is capable of continuously tracking the target even with the semi-reference-free setting in challenging situations, which shows the discriminative ability of the state space memory.

4.3. Comparison with state-of-the-art trackers

Tracking by Initial Bounding Box (BBOX). In this section, we compare our method with state-of-the-art track-

ers using only initial Bounding Box for target specification on four datasets including TNL2K [46], LaSOT [8], OTB99 [31], and MGIT [24]. We utilize the Area Under the Curve (AUC) of the success plot and the tracking precision (Prec) as the main metrics to rank trackers. As shown in Table 1 and 2, our MambaVLT outperforms previous trackers in TNL2k, OTB99, and MGIT but performs less effectively on the LaSOT dataset. However, MambaVLT still achieves better results than some Transformer-based trackers including TransT and SwinTrack. MambaVLT outperforms the best trackers on the TNL2k and OTB99 datasets in terms of AUC by 0.6%, and 2.3%, respectively. It also surpasses the GRM [14] on the MGIT in terms of PRE by 1.6%.

Tracking by Language Specification (NL). We conduct experiments in tracking by language specification task across the four aforementioned benchmarks with state-of-the-art trackers. Our method achieves optimal performance on the TNL2k and MGIT datasets. However, the suboptimal results on the LaSOT and OTB99 datasets may be due to their limited capability to track targets based on ambiguous textual descriptions.

Tracking by Language and Bounding Box (NL&BBOX). We further evaluate MambaVLT in tracking by language and bounding box task with the latest trackers. The benchmarks are TNL2k, LaSOT, OTB99 and MGIT. MambaVLT shows more superior performance, achieving AUC improvements of 3.4% on the TNL2K. Besides, it improves the PRE metric by 14.4% on MGIT. Compared with previous transformer-based trackers, which utilize discrete context prompts, MambaVLT achieves better performance by introducing a continuous time-evolving state space memory to capture long-term multimodal temporal information.

4.4. Ablation Study

Table 3. Analysis of different components in MambaVLT

Variants	TNL2k					
	BBOX		NL		NL&BBOX	
	AUC	*Prec*	*AUC*	*Prec*	*AUC*	*Prec*
baseline	60.9	62.7	55.3	55.0	62.6	65.1
+THSS	62.1	64.2	56.8	57.6	64.5	67.3
+MgB	62.5	64.3	57.3	57.9	65.3	68.2
+SLE	63.1	65.2	57.5	58.3	65.7	69.0
+MS	63.3	65.8	58.4	58.9	66.5	69.9

In this section, we analyze the effectiveness of different main components in MambaVLT, which includes five variants of our model as shown in Table 3. The baseline is MambaVLT without time-evolving hybrid state space (THSS), modality-guided bidirectional scan (MgB), modality-selection (MS) module and selective locality enhancement (SLE) block. The time-evolving hybrid state space brings 1.2%, 1.5% and 1.9% AUC increase for BBOX, NL and NL&BBOX tasks, respectively, which shows the great ability of our time-evolving state space to capture long-term temporal information to adapt to target changes in vision-language tracking. Collaborating with the HMSS module and the SLE module, the modality-guided scan bidirectional scan improves model performance by dynamically modeling and updating reference features through text-first and template-first scans.

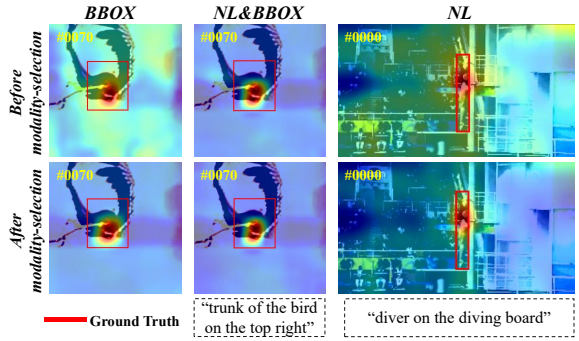

Figure 6. Visualization of the similarity between reference token and search region before and after the modality-selection module.

The introduction of the modality-selection module increases the ability of the model to weigh the importance of different modality information, further improving overall performance. Furthermore, Figure 6 shows the similarity between the reference token and search region feature before and after the modality-selection module, which indicates the refinement of the search region by the modality-selection module can enhance the discriminative ability of model in target localization. The performance increase after adding the selective locality enhancement block demonstrates that its capability to further enhance multimodal features in the current tracking frame.

5. Conclusion

In this work, we propose a Mamba-based vision-language tracking framework with a time-evolving state space to capture long-term continuous target information, based on which the proposed modality-guided bidirectional scan will model and update the multimodal features in a cross-frame manner. Besides, the selective locality enhancement block will enhance the features in the current tracking frame. Moreover, we present a modality-selection module to dynamically weigh the different modality reference features for search region feature refining. Our model achieves favorable performance against state-of-the-art algorithms on four vision-language tracking datasets.

6. Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62172126), and the Shenzhen Research Council (No. JCYJ20210324120202006).

References

[1] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 5

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. 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*, pages 850–865. Springer, 2016. 2

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6182–6191, 2019. 2

[4] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8126–8135, 2021. 2, 7

[5] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13608–13618, 2022. 2

[6] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Eco: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6638–6646, 2017. 2

[7] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7183–7192, 2020. 7

[8] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5374–5383, 2019. 2, 6, 8

[9] Qi Feng, Vitaly Ablavsky, Qinxun Bai, and Stan Sclaroff. Robust visual object tracking with natural language region proposal network. *arXiv preprint arXiv:1912.02048*, 1(7):8, 2019. 7

[10] Qi Feng, Vitaly Ablavsky, Qinxun Bai, Guorong Li, and Stan Sclaroff. Real-time visual object tracking with natural language description. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 700–709, 2020. 7

[11] Qi Feng, Vitaly Ablavsky, Qinxun Bai, and Stan Sclaroff. Siamese natural language tracker: Tracking by natural language descriptions with siamese trackers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5851–5860, 2021. 3, 7

[12] Daniel Y Fu, Tri Dao, Khaled K Saab, Armin W Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. *International Conference on Learning Representations*, 2022. 3

[13] Zhihong Fu, Qingjie Liu, Zehua Fu, and Yunhong Wang. Stmtrack: Template-free visual tracking with space-time memory networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13774–13783, 2021. 2

[14] Shenyuan Gao, Chunluan Zhou, and Jun Zhang. Generalized relation modeling for transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18686–18695, 2023. 1, 7, 8

[15] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 2, 3, 6

[16] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *International Conference on Learning Representations*, 2021. 2, 3

[17] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021. 2

[18] Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022. 2

[19] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *ECCV*, 2024. 3

[20] Mingzhe Guo, Zhipeng Zhang, Heng Fan, and Liping Jing. Divert more attention to vision-language tracking. *Advances in Neural Information Processing Systems*, 35:4446–4460, 2022. 3, 7

[21] Dongchen Han, Ziyi Wang, Zhuofan Xia, Yizeng Han, Yifan Pu, Chunjiang Ge, Jun Song, Shiji Song, Bo Zheng, and Gao Huang. Demystify mamba in vision: A linear attention perspective. *arXiv preprint arXiv:2405.16605*, 2024. 5

[22] Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. Mambaad: Exploring state space models for multi-class unsupervised anomaly detection. *arXiv preprint arXiv:2404.06564*, 2024. 3, 1

[23] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):583–596, 2014. 2

[24] Shiyu Hu, Dailing Zhang, Xiaokun Feng, Xuchen Li, Xin Zhao, Kaiqi Huang, et al. A multi-modal global instance tracking benchmark (mgit): Better locating target in complex spatio-temporal and causal relationship. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 6, 8

[25] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1562–1577, 2019. 6

[26] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8971–8980, 2018. 2

[27] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4282–4291, 2019. 2, 7

[28] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. *arXiv preprint arXiv:2403.06977*, 2024. 3, 5

[29] Wenbing Li, Hang Zhou, Zikai Song, and Wei Yang. Coupled mamba: Enhanced multi-modal fusion with coupled state space model. *arXiv preprint arXiv:2405.18014*, 2024. 3

[30] Yihao Li, Jun Yu, Zhongpeng Cai, and Yuwen Pan. Cross-modal target retrieval for tracking by natural language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4931–4940, 2022. 3, 7

[31] Zhenyang Li, Ran Tao, Efstratios Gavves, Cees GM Snoek, and Arnold WM Smeulders. Tracking by natural language specification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6495–6503, 2017. 1, 2, 3, 6, 7, 8

[32] Baijiong Lin, Weisen Jiang, Pengguang Chen, Yu Zhang, Shu Liu, and Ying-Cong Chen. MTMamba: Enhancing multi-task dense scene understanding by mamba-based decoders. In *European Conference on Computer Vision*, 2024. 3

[33] Liting Lin, Heng Fan, Zhipeng Zhang, Yong Xu, and Haibin Ling. Swintrack: A simple and strong baseline for transformer tracking. *Advances in Neural Information Processing Systems*, 35:16743–16754, 2022. 1, 7

[34] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 3, 6, 1

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1

[36] Ding Ma and Xiangqian Wu. Tracking by natural language specification with long short-term context decoupling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14012–14021, 2023. 7

[37] Yinchao Ma, Yuyang Tang, Wenfei Yang, Tianzhu Zhang, Jinpeng Zhang, and Mengxue Kang. Unifying visual and vision-language tracking via contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4107–4116, 2024. 1, 3, 7, 2

[38] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 6

[39] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022. 2

[40] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4293–4302, 2016. 2

[41] Yanyan Shao, Shuting He, Qi Ye, Yuchao Feng, Wenhan Luo, and Jiming Chen. 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*, pages 19208–19217, 2024. 1, 3, 7

[42] Yuheng Shi, Minjing Dong, and Chang Xu. Multi-scale vmamba: Hierarchy in hierarchy visual state space model. *arXiv preprint arXiv:2405.14174*, 2024. 3

[43] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *International Conference on Learning Representations*, 2022. 3

[44] Mingjie Sun, Jimin Xiao, Eng Gee Lim, Bingfeng Zhang, and Yao Zhao. Fast template matching and update for video object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10791–10799, 2020. 2

[45] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1571–1580, 2021. 7

[46] Xiao Wang, Xiujun Shu, Zhipeng Zhang, Bo Jiang, Yaowei Wang, Yonghong Tian, and Feng Wu. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13763–13773, 2021. 1, 2, 3, 6, 7, 8

[47] Jiangwei Weng, Zhiqiang Yan, Ying Tai, Jianjun Qian, Jian Yang, and Jun Li. Mamballie: Implicit retinex-aware low light enhancement with global-then-local state space. *arXiv preprint arXiv:2405.16105*, 2024. 3

[48] Chenlong Xu, Bineng Zhong, Qihua Liang, Yaozong Zheng, Guorong Li, and Shuxiang Song. Less is more: Token context-aware learning for object tracking. *arXiv preprint arXiv:2501.00758*, 2025. 2

[49] Yinda Xu, Zeyu Wang, Zuoxin Li, Ye Yuan, and Gang Yu. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In *Proceedings of the AAAI conference on artificial intelligence*, pages 12549–12556, 2020. 2

[50] Tianyu Yang and Antoni B Chan. Learning dynamic memory networks for object tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pages 152–167, 2018. 2

[51] Zhengyuan Yang, Tushar Kumar, Tianlang Chen, Jingsong Su, and Jiebo Luo. Grounding-tracking-integration. *IEEE*

Transactions on Circuits and Systems for Video Technology, 31(9):3433–3443, 2020. 3, 7

[52] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision*, pages 341–357. Springer, 2022. 1, 2, 6, 7

[53] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13–es, 2006. 1

[54] Guozhen Zhang, Chunxu Liu, Yutao Cui, Xiaotong Zhao, Kai Ma, and Limin Wang. Vfimamba: Video frame interpolation with state space models. *arXiv preprint arXiv:2407.02315*, 2024. 3

[55] Zhipeng Zhang, Yihao Liu, Xiao Wang, Bing Li, and Weiming Hu. Learn to match: Automatic matching network design for visual tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13339–13348, 2021. 7

[56] Yaozong Zheng, Bineng Zhong, Qihua Liang, Guorong Li, Rongrong Ji, and Xianxian Li. Toward unified token learning for vision-language tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(4):2125–2135, 2023. 3, 7

[57] Yaozong Zheng, Bineng Zhong, Qihua Liang, Zhiyi Mo, Shengping Zhang, and Xianxian Li. Odtrack: Online dense temporal token learning for visual tracking. In *Proceedings of the AAAI conference on artificial intelligence*, pages 7588–7596, 2024. 2

[58] Li Zhou, Zikun Zhou, Kaige Mao, and Zhenyu He. Joint visual grounding and tracking with natural language specification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23151–23160, 2023. 1, 3, 7

[59] Zikun Zhou, Wenjie Pei, Xin Li, Hongpeng Wang, Feng Zheng, and Zhenyu He. Saliency-associated object tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9866–9875, 2021. 2

[60] Zikun Zhou, Jianqiu Chen, Wenjie Pei, Kaige Mao, Hongpeng Wang, and Zhenyu He. Global tracking via ensemble of local trackers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8761–8770, 2022. 2

[61] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In *Forty-first International Conference on Machine Learning*, 2024. 3, 5