Single Target Tracking with Contextual Prompts in Scene Understanding

Xuelin Liu, Huiyu Zhou, Junyu Dong, *Member, IEEE*, Jingjing Xiao*, and Xinghui Dong*, *Member, IEEE*

***Abstract*—This paper presents a single object tracker that incorporates scene understanding, referred to as SU-STTrack, which explicitly encodes the contextual information by the given or Multi-modal Large Language Model (MLLM) generated descriptions of the tracking scene. SU-STTrack merges the linguistic contextual prompts encoded using a pre-trained LLM and visual features using a dual attention mechanism to strengthen tracking robustness and adaptability, where channel attention blocks are fixed within a multi-modality feature fusion process. The experience replay strategy is further proposed, to maintain long-term tracking performance by periodically refreshing the tracking template with accumulated experiences, preventing the model from catastrophic forgetting. SU-STTrack enables both vision-only and vision-language tracking tasks to share the same parameters. Through extensive experiments on vision-language data sets (TNL2k, LaSOT and LaSOT$_{ext}$) and vision-only data sets (UAV123, NfS and OTB100), our tracker achieves 0.569 on TNL2K, 0.628 on the challenging LaSOT, 0.528 on LaSOT$_{ext}$, 0.646 on UAV123, 0.603 on NfS and 0.718 on OTB100 in terms of the Area Under the Curve (AUC) metric with the inference speed of 36.3 FPS. SU-STTrack normally surpasses state-of-the-art methods and shows superior generalization ability across diverse and challenging tracking scenarios[1].**

***Impact Statement*—SU-STTrack not only offers a solution for both the vision-only and vision-language tracking tasks, but also improves tracking performance using the proposed dual attention mechanism and experience replay strategy. Experimental results derived on multiple data sets show the superior performance and generalization ability of SU-STTrack. This study advances the field of object tracking, which provides a new approach that leverages the language and vision information effectively, and has the potential to be applied to various practical scenarios, such as surveillance, robotics, surgical instrument tracking and autonomous driving, improving the accuracy and reliability of tracking systems. However, two critical considerations should be paid attention to, including the risk which is dependent on the generated text may propagate bias, and the ethical concerns, such as a lack of accountability, which also requires mitigation.**

***Index Terms*—Single Object Tracking, Scene Understanding, Template Update, Large Language Models (LLMs).**

I. Introduction

OBJECT tracking [1]–[3] is a vital task of computer vision aimed at tracking the movement of a target in video frames given its initial location in the first frame, where the temporal information should be encoded. It has been applied to robotics [4], medical imaging [5], military [6] and autonomous vehicles [7]. Primarily, the tracking approaches can be categorized into two classes, i.e., generative tracking methods and discriminative tracking methods. For generative tracking methods, they rely on the prior knowledge of the target by constructing a detailed model, such as its statistical distribution characteristics of the appearance of the target. Representative methods include strategies which are designed based on sparse representation [8] and model construction with graph theory [9] [10]. These methods delve deeply into the intrinsic feature of the target, offering a bottom-up solution to tracking tasks.

In contrast to the generative methods, discriminative tracking methods, such as the correlation filtering algorithm [11], adopt a more direct strategy for simplifying the complex tracking issue into a straightforward binary classification problem, which aims to differentiate between the target and the background [12]. The advantage of these methods lies in their ability to leverage powerful classifiers, e.g., Convolutional Neural Networks (CNNs) and Siamese Networks [13], [14], to enhance the accuracy and robustness of object tracking.

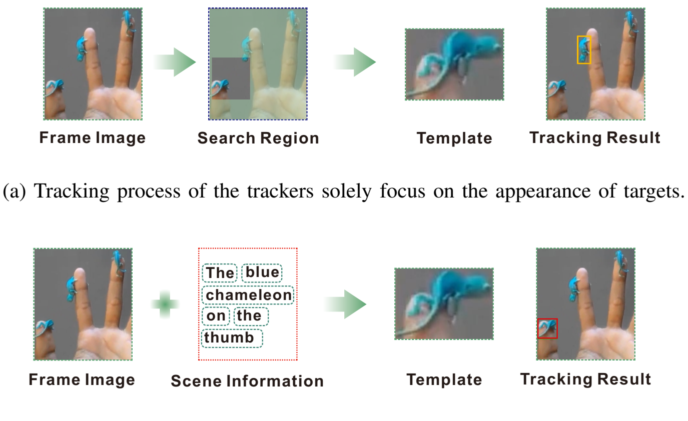
As computer vision and machine learning techniques continue to advance, the two types of methods have also been evolving and integrating. For instance, generative methods can draw on classification ideas from discriminative methods to improve the generalization ability of the model. Similarly, discriminative methods can incorporate appearance modeling techniques from generative methods to enhance adaptability to complex scenes. Despite the remarkable performance can be achieved using these approaches, they are still mainly focused on the analysis of the target itself and heavily rely on the heuristic assumptions of image regions, e.g., appearance consistency and motion consistency [15]. These assumptions may be insufficient when the target has dramatic appearance changes, occlusions and unpredictable motions.

Moreover, those methods tend to overlook the critical interactions between the target and its surrounding environment (see Fig. 1(a)). Although some recent work [16] paid attention to the importance of scene information, their focus was limited to spatial relationships of surroundings, which ignored the more complex and dynamic interplay between the target and the scene. It should be noted that the semantic information

This study was supported by the National Natural Science Foundation of China (NSFC) (No. 42176196) (Corresponding authors: Xinghui Dong and Jingjing Xiao).

X. Liu, J. Dong and X. Dong are with the State Key Laboratory of Physical Oceanography and the Faculty of Information Science and Engineering, Ocean University of China, Qingdao, 266100, China. (e-mail: liuxuelin6654@stu.ouc.edu.cn, dongjunyu@ouc.edu.cn, xinghui.dong@ouc.edu.cn). H. Zhou is with the School of Computing and Mathematical Sciences, University of Leicester, LE1 7RH Leicester, U.K. (e-mail: hz143@leicester.ac.uk). J. Xiao is with the Bio-Med Informatics Research Centre & the Clinical Research Centre, The Second Affiliated Hospital of the Army Medical University, Chongqing, 400037, China. (shine636363@sina.com).

[1] The code and models are available at https://indtlab.github.io/projects/SU-STTrack.



(a) Tracking process of the trackers solely focus on the appearance of targets.

(b) Tracking process of the proposed SU-STTrack with scene understanding.

Fig. 1: Comparison between the existing trackers and SU-STTrack. Here, (a) illustrates the traditional trackers, which solely rely on the pixel-based template similarity and disregard the contextual semantic information within the scene, while (b) demonstrates the proposed tracker that captures contextual prompts from the surrounding by extracting linguistic features.

could be helpful in tracking [15]. For instance, a car driving on a road will exhibit predictable motion patterns aligned with the constraints of the road, such as maintaining a straight path or making smooth turns. In contrast, an athlete performing in a gymnasium is likely to demonstrate highly irregular movements, such as jumps, spins and abrupt direction changes. In other words, the semantic context of the scene imposes inherent constraints on the possible movements of the target.

To address the above issues, we therefore introduce a novel scene understanding tracker, namely, SU-STTrack, which encodes contextual prompts using a pre-trained Large Language Model (LLM) in both the vision-language and vision-only scenarios (see Fig. 1(b)). This tracker utilizes not only visual features but also scene descriptions for vision-language tasks. For vision-only tasks, SU-STTrack generates captions using a pre-trained Multi-modal Large Language Model (MLLM) to grasp scene context, merging it with visual cues. It derives a global semantic representation from descriptions or captions and merges visual and linguistic features through two Channel Attention Blocks (CABs) before passing them to the text and image encoders. A corner-based prediction head forecasts tracking outcomes, while an experience replay system refreshes templates with historical and new images, enabling the tracker to review the past data and adjust to scene variations. The proposed SU-STTrack achieves a proper trade-off between the tracking accuracy and the speed (see Fig. 2). To our knowledge, single target tracking has not been performed in such a manner.

The contributions of this study can be summarized in three aspects below.

1) We introduce a novel end-to-end framework built upon contextual prompts, referred to as SU-STTrack. With powerful cross-modal learning, this framework achieves excellent performance in both the visual-only and visual-language tracking tasks.

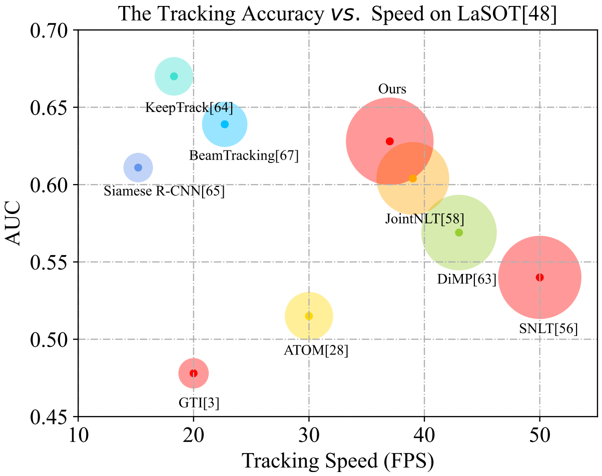


Fig. 2: Comparison between the proposed SU-STTrack and state-of-the-art trackers on LaSOT [17] in terms of the tracking accuracy (AUC) and speed.

2) We design a dual attention mechanism that enables the network to account for both low-level texture features and high-level semantic features and eliminates redundancy when fusing linguistic and visual features.
3) We propose an experience replay-based template update method, which allows the tracker to learn more historical information and avoid catastrophic forgetting.

The remainder of this paper is organized as follows. We first review the relevant work in Section II. The proposed approach is then introduced in Section III in detail. In Section IV, we report the experimental setup and results. Finally, we draw our conclusion in Section V.

II. Related Work

A. Vision-Only Trackers

Vision-only trackers operate solely on video frames to predict the subsequent positions given the initial state. In particular, semantic tracking has gained attention as it helps capture contextual cues beyond raw pixel information. While Convolutional Neural Networks (CNNs) have been used in this context, Transformer-based trackers have recently emerged as a powerful alternative [18] due to their ability to capture long-range dependencies and a larger receptive field, making them particularly effective for semantic tracking tasks.

1) Semantic Tracking: Traditional single-object tracking methods mainly rely on low-level visual features, such as color histogram, texture and shape [19], [20]. In contrast, semantic tracking leverages high-level semantic information to enhance tracking performance. This information typically comes from pre-trained deep learning models to extract the category and contextual information of targets [9], [21]. By integrating semantic information into the tracking process, trackers can better understand the essence of the target region and the background, thereby improving the robustness and accuracy of tracking. For instance, Xiao *et al.* [15] combined the recognition of target categories in the online tracking process using inter-supervised networks, distinguishing targets from the background by capturing general and distinctive features.

The GOTURN tracker [22] used CNNs to learn generic object tracking, which showed that deep features could enhance tracking robustness across different scenarios. SiamRPN++ [23] utilized a more powerful feature backbone, allowing the tracker to maintain high precision in complex and clustered scenes. On the other hand, ATOM [24] achieved higher accuracy and more robust target tracking by combining semantic information and maximizing the overlap rate. In addition, D3S [25] proposed a deformable Siamese network that enhanced tracking performance using spatial supervision information. LSSiam [26] resorted to adding a smaller network for learning local semantic features, incorporating a classification branch into the classical Siamese framework.

2) Transformer-Based Tracking: By capturing broader contextual information, transformers enable more accurate tracking in complex environments [27], as they excel at capturing long-range dependencies through self-attention mechanisms. For instance, Wang *et al.* [5], Zhao *et al.* [28] and Chen *et al.* [29] utilized the encoder-decoder mechanism to replace traditional cross-correlation operations. This enabled a deep analysis of interdependencies between the target object and the global scene context. TransT [29] introduced the transformer architecture in a hybrid model that combined it with traditional tracking methods, demonstrating its potential in real-time tracking tasks. STARK [30] further optimized the application of the transformer, achieving end-to-end tracking and capturing long-range dependencies in both spatial and temporal dimensions. Recent research, such as FFTrack [31], has achieved breakthrough progress in single object tracking tasks. FFTrack [31] combined high-frequency and low-frequency features with two-stage frequency fusion. In [32], CoTracker was introduced which tracked points jointly using the token proxies fed into Transformer.

B. Vision-Language Trackers

The concept of Vision-Language (VL) tracking was first introduced by Li *et al.* [33], which facilitated the development of subsequent studies. Li *et al.* [33] designed a unified local-global-search framework from the perspective of cross-modality retrieval. In recent years, key advancements have further developed vision-language trackers. Zhao *et al.* [34] designed a fusion module based on a transformer network and proposed proxy tokens, enabling the model to leverage textual information more effectively. Zhang *et al.* [35] introduced a multi-modality alignment module and achieved feature integration in a unified backbone to improve semantic guidance.

Feng *et al.* [11] designed a generally applicable module named SNLT and SNLT-RPN for all Siamese trackers, promising to improve vision-language tracker performance in the future. Chen *et al.* [36] present a sequence-to-sequence learning framework named Seqtrack, which cast visual tracking as a sequence generation problem and predicts object bounding boxes in an autoregressive fashion, similarly, MMTrack [37] cast vision-language tracking as a token generation task. By leveraging that textual semantic information, these trackers can handle cases where visual similarity alone is insufficient, such as when targets are visually similar or undergo appearance changes. Therefore, in our study, we encode that linguistic information as contextual prompts in our tracker.

C. Template Update

Early siamese-based trackers, such as SiamFC [38] and SiamRPN [39], relied solely on a static initial template extracted from the first frame. These trackers lacked adaptation to variations in the target appearance, which inevitably led to tracking drift in dynamic scenarios. To address this limitation, recent research was focused on adaptive template update mechanisms. For example, a confidence threshold-based update mechanism was used, such as TATrack [40]. A Transformer-based framework was also utilized, such as STARK [30], which captured long-term dependencies by means of historical information. Although template update mechanisms have drawn the attention of researchers, many tracking frameworks avoided them in order to maintain architectural simplicity, including OneTracker [41]. In contrast, our experience replay mechanism extends the temporal receptive field to the initial frame rather than using the latest frame.

III. Our Approach

In this section, we will introduce the proposed multi-modality tracking framework, i.e., SU-STTrack. This framework can be used for both the vision-language and vision-only tracking tasks. The pipeline of SU-STTrack is shown in Fig. 3.

A. Overall Network Architecture

The proposed SU-STTrack contains four modules, including contextual prompt-based feature extraction, adaptive multi-modality feature fusion, prediction and experience replay. The pipeline of SU-STTrack is displayed in Algorithm 1 and Fig. 3. The input of SU-STTrack is a frame image along with text descriptions or only a frame image. Then a search region is cropped from the image, which will be fed into the image encoder along with the template image. If text descriptions are not available, a pre-trained MLLM, i.e., OFA [42], will be used to generate a set of text descriptions. The descriptions help disambiguate the behavior of the target and the interactions in complex scenes, which tends to enhance tracking accuracy.

We first extract visual features and contextual prompts from both the search region and the template image and the text descriptions using an image encoder and a text encoder, respectively. In this study, we used the pre-trained BERT [43] and Resnet101 [44] as the text and image encoders, respectively. The contextual prompts serve as a bridge between scene understanding and target representation. They encapsulate the semantic information from the text descriptions, guiding feature extraction to align with the dynamic interplay between the target and the surrounding environment.

The visual features and contextual prompts are then fused using a dual attention mechanism that we design, to generate the more discriminative features. Within the dual attention mechanism, we first use a Channel Attention Block (CAB)

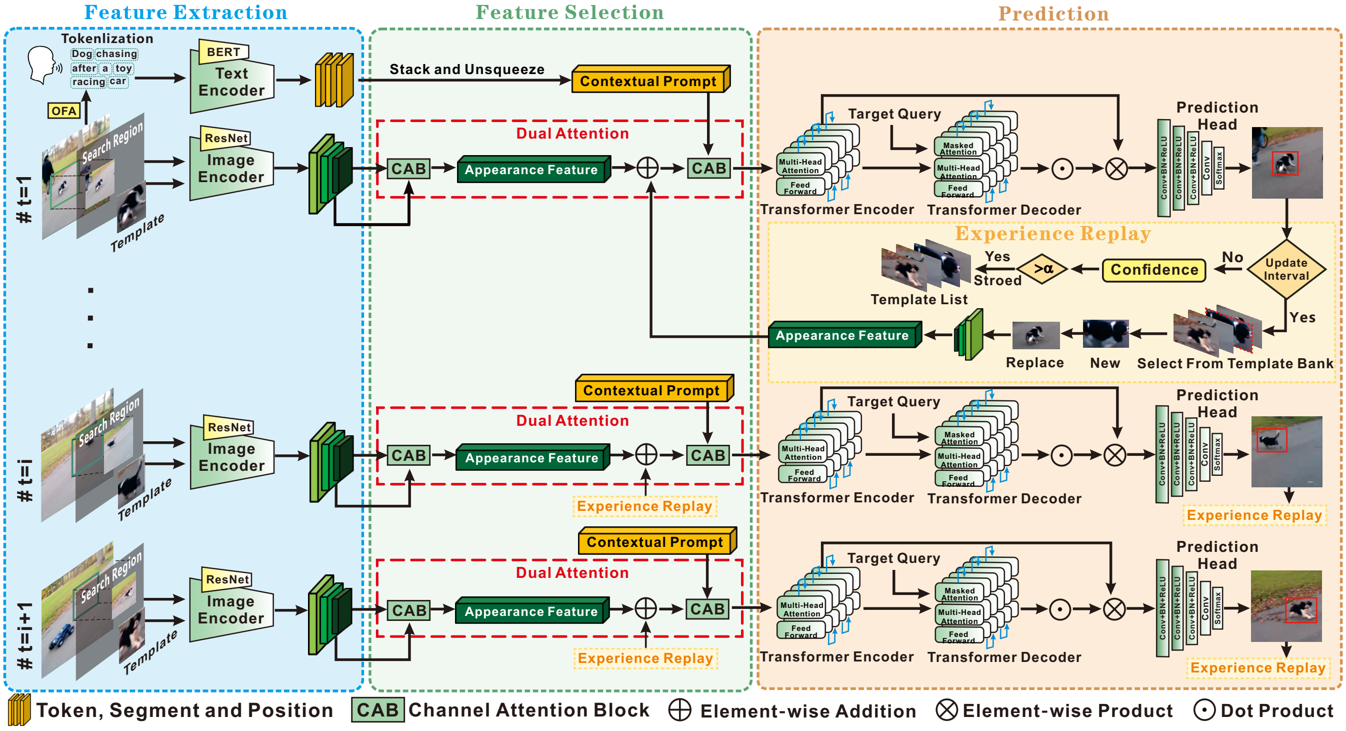


Fig. 3: The pipeline of the online tracking using the proposed SU-STTrack. Specifically, SU-STTrack combines visual features and linguistic contextual prompts by means of a dual attention mechanism. The features fused are then sent to an encoder-decoder network. The output is fed into a fully convolutional network and the result is the tracking prediction.

to fuse the deep and shallow visual features. Then visual features are fused with linguistic features using a second CAB. The features fused are fed into a Transformer encoder. The output together with a target query are sent to the decoder. In this situation, contextual prompts provide the decoder with crucial semantic cues, ensuring that the tracking predictions are boosted by the scene context.

Furthermore, the corner-based bounding box prediction head is used to predict tracking results. In addition, we propose an experience replay strategy that stores confident tracking results as new template candidates. When updating the template, historical templates are randomly selected to be merged with the new template.

B. Contextual Prompt-Based Feature Extraction

For vision-language data sets, we use the Bidirectional Encoder Representations from Transformers (BERT) [43] encoder to extract frame-wise linguistic features which are used as contextual prompts. Whether these prompts are explicitly provided or are generated using a captioning model, they encapsulate the critical scene-level information that complements the visual features extracted using the image encoder. The input text T is tokenized into subwords and embedded into a set of vectors. These vectors are then passed through multiple Transformer layers, to produce a sequence of contextualized embeddings. This process can be expressed as:

$$H = \text{BERT}(T) = [h_1, h_2, \ldots, h_n], \tag{1}$$

where H represents the sequence of hidden states at the final layer of BERT, and h_i denotes the hidden state corresponding to the i^{th} token. To obtain a fixed-dimensional representation of the entire text, we use the hidden state of the classification token, which is designed to capture the aggregate information of the sequence:

$$f^l = H_{CLS}. \tag{2}$$

The linguistic feature vector $f^l \in \mathbb{R}^{1\times B\times C}$ is used as contextual prompts in conjunction with visual features for subsequent frame tracking. To achieve shape alignment during cross-modal feature fusion, we also expand text prompt vectors in order to match the shape of image features $f^v \in \mathbb{R}^{WH\times B\times C}$.

Regarding vision-only data sets, we generate text descriptions using a caption generator model, namely, OFA [42], as contextual prompts from the visual content. OFA is a versatile MLLM, which is capable of handling multiple vision and language tasks, including image captioning. The text descriptions generated for the given initial frame I from the sequences are produced by the OFA [42] model, which summarizes the visual content. The descriptions are then fed into the BERT encoder, to extract linguistic features, following the same process as described above and ensuring the consistent feature extraction across both vision-language and vision-only data sets.

Given a frame, we use the pre-trained ResNet-101 model as the image encoder to extract visual features, which capture both low-level and high-level characteristics of the target and its surroundings. Both the visual features and contextual prompts are sent to the feature fusion module built on top of

Algorithm 1 Online Tracking Algorithm at Time t

```
 1: Input:
 2:     Case 1: Image $I$ and text description $D$
 3:     Case 2: Only image $I$
    Initialization: Crop search region $S$ and initialize template $T$.
 4: Feature Extraction:
 5:     if Case 2 then
 6:         Generate description: $D \leftarrow \text{TextGenerator}(I)$
 7:     end if
 8:     Extract linguistic features: $f^l \leftarrow \text{TextEncoder}(D)$ (Equation (2))
 9:     Extract template features: $f^t \leftarrow \text{ImageEncoder}(T)$
10:     Extract search features: $f^s \leftarrow \text{ImageEncoder}(S)$
11: Feature Fusion:
12:     Fuse high- and low-level features: $f^t \leftarrow \text{CAB}(f^t)$
13:     Fuse high- and low-level features: $f^s \leftarrow \text{CAB}(f^s)$
14:     Combine visual features: $f^v \leftarrow f^t + f^s$
15:     Fuse visual and linguistic features: $f \leftarrow \text{CAB}(f^v, f^l)$
16: Bounding Box Prediction:
17:     Predict corner: $(\hat{x}_{tl}, \hat{y}_{tl)}, (\hat{x}_{br}, \hat{y}_{br})$ (Equations (9-12))
18: Template Update:
19:     Update the template bank with confident results (Algorithm 2)
20: Repeat: Perform steps 3 to 19 for all frames in the sequence.
```

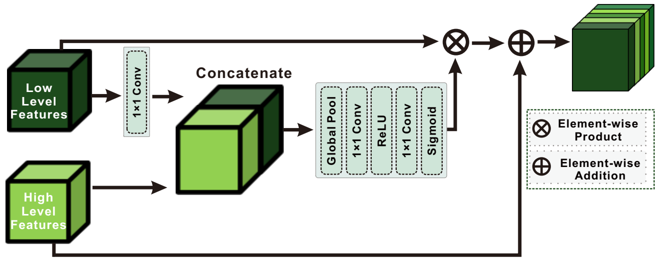


Fig. 4: Architecture of the proposed Channel Attention Block (CAB).

the dual attention mechanism.

C. Adaptive Multi-modality Feature Fusion

It is known that low-level features contain abundant spatial details and essential discriminative cues but lack the semantic information. In contrast, high-level features encode rich semantic representation while suffering from the reduced spatial precision [23], [24], [45]. To jointly exploit both the low-level and high-level features, we design a Channel Attention Block (CAB), as shown in Fig. 4. These features are fused using a CAB in order to obtain a compact image representation. In addition, we enrich the tracking model with the additional contextual information by fusing the visual features and the linguistic contextual prompts using a seconod CAB, enhancing its ability to understand and track the target.

Specifically, the CAB first applies a 1×1 convolution in order to reshape the low-level features f_{Low}. This operation aligns these features with the dimensionality of the high-level features f_{High}. The low-level features reshaped are then concatenated with the high-level features, forming a set of composite feature maps. Channel attention weights are further computed using a sequence of operations, including the global average pooling, a 1×1 convolution, a ReLU activation, a second 1×1 convolution and the Sigmoid function, which can be formulated as:

$$f_{Concat} = \text{Concat}(\text{Conv}_{1\times1}(f_{Low}), f_{High}), \quad (3)$$

$$\omega = \text{Sigmoid}(\text{Conv}_{1\times1}(\text{ReLU}(\text{Conv}_{1\times1}(\text{GAP}(f_{Concat}))))). \quad (4)$$

The weights ω are applied to the low-level features, producing a series of weighted feature maps. Finally, the high-level features and the weighted low-level features are fused using the addition operation, which can be expressed as:

$$f^v = \omega \otimes f_{Low} + f_{High}. \quad (5)$$

In essence, the CAB dynamically emphasizes key features by assigning channel-wise attention weights. Thereby, it selectively enhances the important components of the feature maps.

As a global semantic embedding, linguistic features provide contextual scene constraints that are critical for the scenarios where target movement is influenced by its surrounding environment. Therefore, we extend the CAB in order to incorporate textual scene descriptions into the visual feature space. In other words, the CAB can also be utilized for combining the linguistic and visual features except being used to fuse the low-level and high-level visual features. This operation can be formulated as:

$$f = \omega' \otimes f^l + f^v. \quad (6)$$

As a result, an overall feature representation can be obtained across the two modalities, which encodes not only the visual characteristics but also the scene understanding information.

D. Prediction

The prediction module consists of an encoder, a decoder and a fully convolutional prediction head. Both the encoder and decoder comprise a set of Transformer blocks, which are able to capture the long-range dependencies.

In terms of the encoder, multi-modality feature maps f depicted in Equation (6) are first flattened. The features flattened are then processed in order to capture the long-range dependencies and enrich them with the global contextual information. As a result, the localization ability of the model trained can be improved. Given that V corresponds to the processed features, K represents the template features, Q represents the query and $\sqrt{d_k}$ is the scaling factor, the multi-head attention mechanism used in the encoder can be expressed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \quad (7)$$

With regard to the decoder, a single query (Q') is used to predict tracking results while the key and value pairs (K', V') represent the features extracted from the template and search region by the encoder. In addition to the self-attention mechanism, each decoder block contains an encoder-decoder attention mechanism [27], which can be formulated as:

$$\text{EncDecAttention}(Q', K', V') = \text{softmax}\left(\frac{Q'K'^{T}}{\sqrt{d_{k'}}}\right)V'. \quad (8)$$

This mechanism allows the target query to visit all positions within the template and search region patches. Hence, the more comprehensive information can be captured.

For the purpose of predicting the bounding box surrounding a target, the prediction head computes the similarity score between the features extracted from the search region and the output of the decoder. To emphasize the discriminative areas, the score, which reflects the confidence level, is element-wise multiplied with the search region features. The processed features are then passed through a fully convolutional sub-network, which comprises four sequential customized convolutional layers. Each layer captures hierarchical feature representations. Within the layer, batch normalization ensures the stable and efficient training and the ReLU function improves the ability of the model to learn complex patterns by introducing the non-linearity. Following the last layer, a convolutional layer and the softmax function are used to convert the feature maps into a set of probabilities. Finally, the coordinates of the corners of the bounding box are predicted on top of the probabilities. The prediction operation can be illustrated as:

$$\hat{x}_{tt} = \iint_{\Omega} x \cdot P_{tl}(x, y)\, dx\, dy, \quad (9)$$

$$\hat{y}_{tt} = \iint_{\Omega} y \cdot P_{tl}(x, y)\, dx\, dy, \quad (10)$$

$$\hat{x}_{br} = \iint_{\Omega} x \cdot P_{br}(x, y)\, dx\, dy, \quad (11)$$

$$\hat{y}_{br} = \iint_{\Omega} y \cdot P_{br}(x, y)\, dx\, dy, \quad (12)$$

where $\hat{x}_{tl}$ denotes the x-coordinate of the top-left corner, $P_{tl}(x, y)$ is the probability produced by the prediction head and Ω stands for the search region.

E. Experience Replay

For the purpose of enhancing the robustness and adaptability of the tracking operation, we introduce an experience replay strategy by periodically updating the tracking template based on accumulated experiences, as illustrated in Algorithm 2. Specifically, the bounding box region given is used as the initial template for the first frame. At a regular interval, T_E, the confidence score of the tracking region predicted is computed using a fully convolutional sub-network that we described in Section III-D. If the confidence score exceeds a predefined threshold τ, the result is considered reliable, which can be treated as a new template. This template is then cropped from the original image and is appended to a list of candidate templates for future use.

At each template update interval, T_U, we randomly select a historical template from the candidate list and combine it with the current template using a weighted multiplication operation. This operation can be formulated as:

Algorithm 2 Experience Replaying Strategy

```
 do
     if t % $T_E$ == 0 and confidence_t $\geq \tau$ :
         E = E.append(result_t)
     if t % $T_U$ == 0 and $|E| \geq \lambda$ :
         new_template = random choice(E)
          $\text{template}_{t+1} = \alpha \cdot \text{template}_t + \beta \cdot \text{New\_template}$
while $1 < t < T$
```

$$T_{t+1} = \alpha \cdot T_t + \beta \cdot \text{Random}(E), \quad (13)$$

where T_t represents the confident tracking result at the current frame, E denotes the pool of historical templates, and α and β stand for the weighting coefficients which control the balance between the historical templates and the new representation. Specifically, α adjusts the emphasis on the newly captured template, which ensures the adaptability to the latest scene context. On the other hand, β controls the weight assigned to the historical templates, which represents the reliance of the tracker on the past observations. The weighted multiplication operation enhances the robustness of the tracker, which ensures that the historical information is seamlessly integrated with the latest tracking results.

IV. Experiments

In this section, we first report the experimental setup in which the details of the training and inference stages are introduced. Then we compare the proposed tracker with nine vision-language and 14 vision-only baseline methods on six publicly available data sets. Finally, a series of ablation experiments are conducted in order to examine the effectiveness of the components of the proposed approach.

A. Experimental Setup

Our tracker was implemented using Python 3.6 and PyTorch 1.7. The offline training was conducted on a workstation with two NVidia 3090 Graphics Processing Units (GPUs). The inference process was run on a single NVidia 3090 GPU. In total, four training data sets were utilized, including LaSOT [17], RefCOCO [46], TNL2K [47] and OTB99-Lang [33]. With regard to the image and text encoders, the pre-trained ResNet [44] and BERT [43] were used, respectively. The images and phrases contained in the training sets of LaSOT, OTB99-Lang, RefCOCO and TNL2K were used to train the proposed SU-STTrack.

Following the existing study [30], we conducted the offline training operation in two stages, in which the network was trained for 500 epochs and 50 epochs, respectively. Within the first stage, both the feature fusion module and the Transformer network were trained while the image and text encoders and the prediction head were kept frozen. During the second stage, only the prediction head was trained and the other modules of the entire network were kept frozen. The batch size was set to 32. We employed the Adam optimizer with the weight decay of 10^{-4}. The initial learning rate was set to 10^{-4} and was decreased to 10^{-5} after 400 epochs. We utilized horizontal flip and brightness jittering for data augmentation. The resolutions of the search and template images were set to 384×384 pixels and 192×192 pixels, respectively. The dynamic templates were updated when the update interval of 200 had been reached by default, where the template with the highest score was selected as an online sample. The confidence threshold was set to 0.6.

The One Pass Evaluation (OPE) scheme was utilized, in which the tracker was initialized on the first frame and performance was evaluated until the last frame did not require re-initialization [48]. The overlap, also known as the success plot, measures the Intersection over Union (IoU) between the bounding box predicted and the ground-truth bounding box across all the frames in a sequence [48]. The Area Under the Curve (AUC) [48] metric can be computed on top of the IoU values. We used AUC, precision [48] and normalized precision [49] to measure the performance of the tracking task.

B. Main Experiments

We applied the proposed SU-STTrack and existing trackers to the vision-language and vision-only tracking tasks. The quantitative results are reported in Table I. It can be seen that our method outperformed its counterparts on both the TNL2K [47] and LaSOT$_{\text{ext}}$ [50] data sets while performed slightly worse than VLT$_{\text{TT}}$ [51] and TransVLT [34] on the LaSOT [17] data set in the vision-language tracking task. In addition, the visualization of the results obtained using our method and six baselines are shown in Fig. 5. As can be observed, SU-STTrack was able to maintain robustness when suffering interference from motion blur, out of view, viewpoint change, illumination variation and occlusion.

1) Vision-Language Tracking Task: Regarding the vision-language tracking task, we evaluated the proposed SU-STTrack together with ten baselines on three challenging language-guided tracking benchmarks, including TNL2K [47], LaSOT [17] and LaSOT$_{\text{ext}}$ [50].

TNL2K: As a large-scale benchmark for natural language tracking tasks, the TNL2K [47] data set contains 2,000 diverse video sequences. In total, over 1.2 million frames are included in this data set. Each frame was annotated with both bounding boxes and corresponding English sentences. The TNL2K data set poses challenges in both the visual and linguistic understanding due to its diversity in scene categories and object appearances. As shown in Table I, our method achieved improvements of 0.011 in AUC and 0.124 in normalized precision compared to TransVLT [34]. Also, the proposed SU-STTrack performed comparably to the state-of-the-art vision-language tracker, JointNLT [57]. These results show the advantage of integrating the linguistic data with the visual characteristics, which enables the model trained to better capture the semantic information from the global scene.

LaSOT: The LaSOT [17] data set was collected for the long-term tracking task with difficult attributes, such as rapid motion, occlusions and background clutter. This data set consists of 280 videos with an average sequence length of 2,448 frames. It provides extensive test videos for the trackers designed for prolonged sequences. LaSOT [17] also includes fine-grained text descriptions that enrich the challenges of the data set. As presented in Table I, our approach outperformed the majority of the baselines, except VLT$_{\text{TT}}$ [51] and TransVLT [34]. In particular, SU-STTrack achieved an increase of 0.016 over JointNLT [57] and an increase of 0.088 over SNLT [54] in terms of the AUC metric.

LaSOT$_{\text{ext}}$: The LaSOT$_{\text{ext}}$ [50] data set is an extended version of LaSOT [17] by introducing additional video sequences and text descriptions, which makes it more challenging. This data set is particularly useful for testing the generalization capability of trackers across diverse conditions due to the great differences between the training and testing sets. As depicted in Table I, SU-STTrack achieved a gain of 0.044 in terms of the AUC metric, compared to the state-of-the-art VLT$_{\text{TT}}$ [51] method, in such challenging vision-language tracking scenarios, showing its adaptability to novel environments.

2) Vision-Only Tracking Task: We also assessed SU-STTrack along with 14 baselines on three vision-only benchmarks, including NfS [60], OTB [48] and UAV123 [59], for the vision-only tracking task.

UAV123: The UAV123 [59] data set was captured using an Unmanned Aerial Vehicle (UAV). This data set consists of 123 fully high-resolution video sequences with an average length of 915 frames per video, 10 object categories and 12 challenging attributes. Using the UAV123 data set, as reported in Table I, the AUC, precision and normalized precision values that SU-STTrack achieved were 0.646, 0.856 and 0.806, respectively. In contrast to SiamRPN++ [23], our method outperformed it by 0.033 in terms of the AUC metric.

NfS: The NfS [60] data set was collected for high-frame-rate visual tracking tasks, in which fast-moving targets were captured with challenging attributes. We used the 30 FPS version of the Need for Speed subset. As shown in Table I, SU-STTrack gained an improvement of 0.013 in terms of the AUC metric compared with ATOM [24].

OTB100: Due to the comprehensiveness and the challenges that the OTB100 [48] data set presents, it has been widely utilized for tracking tasks. As reported in Table I, our SU-STTrack performed the best among the 14 trackers with the AUC, precision and normalized precision values of 0.718, 0.904 and 0.872, respectively. In particular, the proposed SU-STTrack outperformed the recent state-of-the-art ToMP [69] and SeqTrack [36] methods by the margins of 0.017 and 0.022, respectively, with regard to the AUC metric.

3) Performance Analysis: We analyzed the performance of SU-STTrack in terms of inference speed (Frames Per Second or FPS), computational complexity (Floating-Point Operations Per Second or FLOPs) and model size (number of parameters). As shown in Table II, our SU-STTrack processed video sequences at an impressive speed of 36.3 FPS, which outperformed five state-of-the-art trackers. Furthermore, our model has an FLOPs of 20.4G and a total of 47.2 million parameters. It is suggested that our SU-STTrack achieved a proper balance between computational complexity and model size. In contrast, the two lightweight models, i.e., SiamRPN++

TABLE I: Comparison between the proposed SU-STTrack with ten and fourteen baselines on the vision-language and vision-only data sets, respectively. The top three results are highlighted in the **bold**, underlined and *italic* fonts, in turn.

Tracker Type	Method	Source	TNL2K [47]			LaSOT [17]			LaSOT$_{ext}$ [50]		
			AUC (↑)	*P* (↑)	P_{Norm} (↑)	*AUC* (↑)	*P* (↑)	P_{Norm} (↑)	*AUC* (↑)	*P* (↑)	P_{Norm} (↑)
Vision-Language	Wang *et al.* [52]	arxiv2018	-	-	-	0.277	0.304	-	-	-	-
	Feng *et al.* [53]	WACV2020	0.250	0.270	0.340	0.500	0.560	-	-	-	-
	SNLT [54]	CVPR2021	-	-	-	0.540	0.574	0.636	-	-	-
	GTI [55]	TCSVT2021	-	-	-	0.478	0.476	-	-	-	-
	TNL2K-II [47]	CVPR2021	0.420	0.420	0.500	0.513	0.554	-	-	-	-
	Li *et al.* [56]	CVPRW2022	0.440	0.450	0.520	0.530	-	0.560	-	-	-
	VLT$_{TT}$ [51]	NeurPS2022	0.547	0.553	0.718	**0.673**	**0.715**	**0.802**	0.484	0.543	0.599
	TransVLT [34]	PRL2023	0.558	-	*0.616*	0.660	0.698	-	-	-	-
	JointNLT [57]	CVPR2023	**0.569**	**0.581**	0.736	0.604	0.636	0.735	-	-	-
	QueryNLT [58]	CVPR2024	0.533	0.530	0.704	0.542	0.550	0.625	-	-	-
	SU-STTrack	Ours	**0.569**	0.572	**0.740**	*0.628*	*0.660*	*0.754*	**0.528**	**0.565**	**0.613**
Tracker Type	Method	Source	UAV123 [59]			NfS [60]			OTB100 [48]		
			AUC (↑)	*P* (↑)	P_{Norm} (↑)	*AUC* (↑)	*P* (↑)	P_{Norm} (↑)	*AUC* (↑)	*P* (↑)	P_{Norm} (↑)
Vision-Only	SiamRPN [39]	CVPR2018	-	-	-	-	-	-	0.637	0.851	-
	DeepSCDCF [61]	CVPR2018	-	-	-	-	-	-	0.631	-	-
	ATOM [24]	CVPR2019	0.650	-	-	0.590	-	-	0.671	-	-
	SiamRPN++ [23]	CVPR2019	0.613	0.807	-	-	-	-	0.696	0.914	-
	DiMP [62]	ICCV2019	0.654	0.856	-	0.620	-	-	0.684	-	-
	KeepTrack [63]	ICCV2019	0.697	-	-	0.664	-	-	0.701	-	-
	Siamese R-CNN [64]	CVPR2020	0.649	-	-	0.639	-	-	0.701	-	-
	TransT [29]	CVPR2021	0.691	-	-	0.657	-	-	0.694	-	-
	OSTrack-256 [65]	ECCV2022	0.683	-	-	0.647	-	-	-	-	-
	BeamTracking [66]	TIP2022	0.668	0.773	-	-	-	-	0.653	0.886	-
	AiATrack [67]	ECCV2022	0.706	-	-	0.679	-	-	0.696	-	-
	Mixformer-L [68]	CVPR2022	0.695	0.910	-	-	-	-	-	-	-
	TOMP101 [69]	CVPR2022	0.669	-	-	0.667	-	-	0.701	-	-
	SeqTrack-B384 [36]	CVPR2023	0.692	0.900	0.848	0.676	0.829	0.850	0.696	0.908	0.858
	SU-STTrack	Ours	0.646	0.856	0.806	0.603	0.716	0.740	0.718	0.904	0.872

TABLE II: Comparison between the proposed SU-STTrack and six baselines in terms of the inference speed, computational complexity and number of parameters.

Tracker	Speed (FPS)	FLOPs (G)	Params (M)
JointNLT [57]	**39.0** [57]	34.9 [57]	153.0 [57]
SiamRPN++ [23]	35.0 [23]	48.9 [23]	54.0 [23]
SeqTrack-B384 [36]	15.0 [36]	148.0 [36]	89.0 [36]
ATOM [24]	30.0 [24]	-	-
TOMP101 [69]	19.6 [69]	-	-
KeepTrack [63]	18.3 [63]	-	-
SU-STTrack (Ours)	36.3	**20.4**	**47.2**

[23] and SeqTrack-B384 [36], used 54.0 and 89.0 million parameters, respectively. By referring to the results reported in Table I, it has been demonstrated that SU-STTrack achieved a good trade-off between accuracy and inference speed, computational complexity and model size. In other words, our method produced the promising tracking accuracy without using excessive parameters and sacrificing the inference speed.

4) Cross-Domain Generalization: To further evaluate the cross-domain generalization capability of our pre-trained model, we conducted an additional experiment on the UVOT400 [72] underwater tracking data set. In contrast to the training data set that we used, this data set was captured in a more challenging unseen domain, characterized by turbid water, low contrast and dynamic lighting conditions. As shown in Table III, our pre-trained model achieved an AUC value of 48.2%, which was higher than the values produced by six baselines. It is indicated that our SU-STTrack has a good cross-domain generalization capability.

5) Limitations: Although SU-STTrack was designed in order to accommodate both vision-language and vision-only data sets, it may struggle with vision-only data sets due to the robustness of the text description generation operation. Specifically, the image captioning model, i.e., OFA [42], was originally introduced for the purpose of generating a caption for the visual content of an image. This model normally serves as a proxy for natural language annotations. However, the captions generated may not consistently capture the semantic complexity of the scene or accurately represent the fine-grained details of a target. Consequently, the captions generated may result in ambiguous or inaccurate descriptions, particularly in the cases where the appearance of an target is subtle or the surrounding is intricate.

Furthermore, the use of pre-trained Large Language Models (LLMs) may introduce risks of propagating biases. Given that multiple models process the output iteratively, minor errors may compound into a significant deviation, particularly in the Single Object Tracking (SOT) area because noisy captions usually confuse the target representation. Due to the reliance on single-source human evaluations, image captioning models also tend to treat subjective assessments as the objective truth.

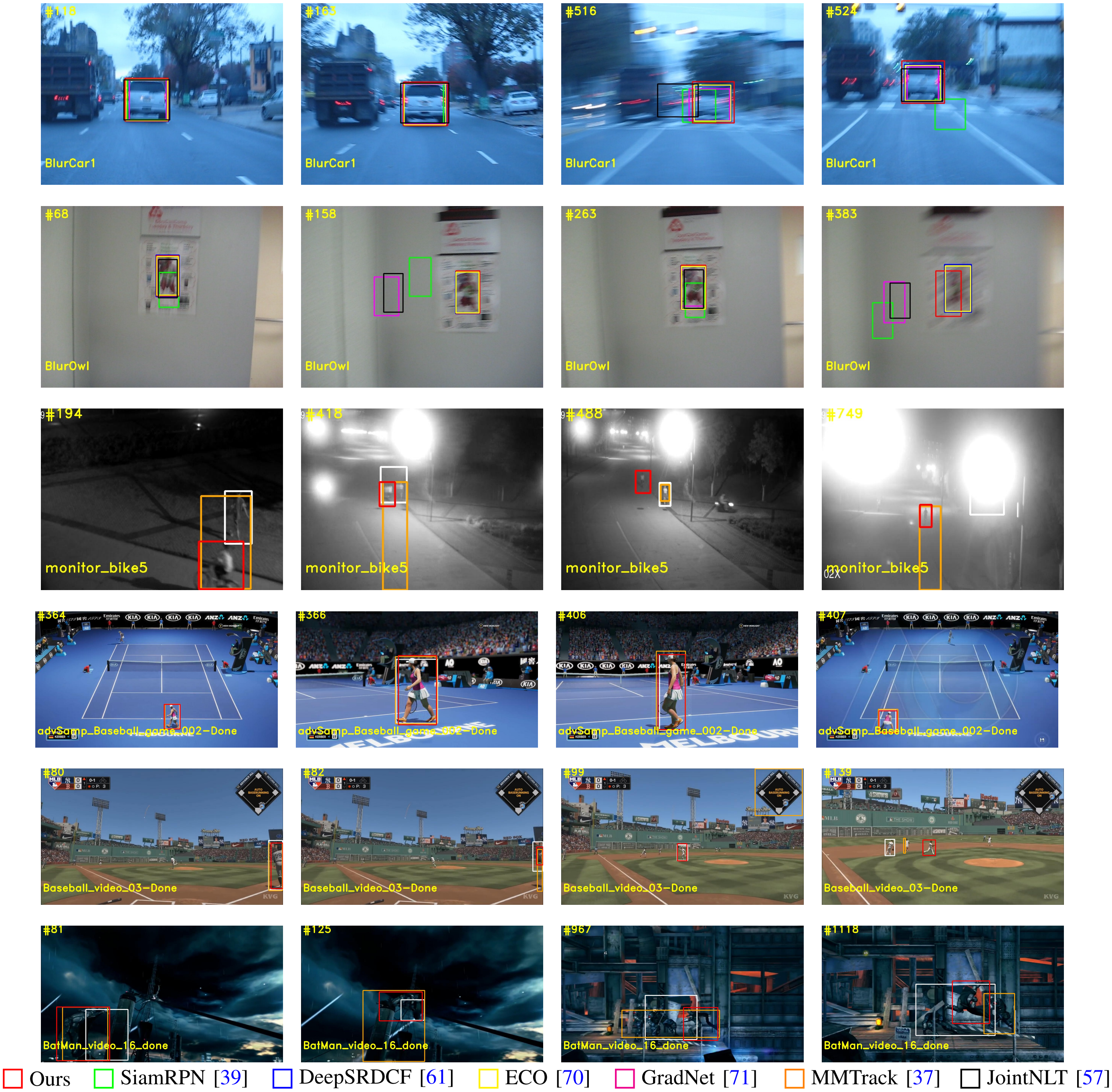


Fig. 5: Visualization of the results of our tracker and six state-of-the-art baseline trackers on the challenging sequences contained in the OTB100 [48] and TNL2k [47] data sets. Specifically, the challenges include motion blur (1st and 2nd rows), illumination variation and low resolution (3rd row), viewpoint change (4th row), out-of-view (5th row) and scale variation and full/partial occlusion (6th row).

Besides, there exists a risk of introducing noise or redundant information from the captioning model, which could impair the performance of the tracker, in particular, in challenging scenarios characterized by heavy occlusion or background clutter. Although the use of caption generation enables SU-STTrack to be applied to both the vision-language and vision-only tracking tasks, further improvements in the text description generation quality or use of alternative methods for enriching the visual information should be explored in the future, to effectively bridge the gap between both the tracking tasks.

As illustrated in Fig. 6, the initial textual description "the middle elephant in the last row" becomes invalid as both the elephant herd and camera perspective gradually shift from frame *11* in the sequence *Elephant_video_5*. The target position has moved to the leftmost column of the last row at frame *35*, rendering the initial description obsolete. Significant

TABLE III: Comparison between six pre-trained trackers and our pre-trained model on the UVOT400 [72] data set in terms of the AUC, precision and normalized precision metrics.

Metric	ATOM [24]	STARK [30]	DiMP-50 [62]	PrDiMP [73]	TrTr [28]	SiamRPN [39]	**Ours**
AUC (↑)	0.433	0.434	0.421	0.420	0.452	0.475	**0.482**
P (↑)	0.376	0.404	0.363	0.366	0.429	0.439	**0.477**
P_{Norm} (↑)	0.517	0.499	0.496	-	0.554	**0.579**	0.550

TABLE IV: The effect of contextual prompts and experience replay on the proposed SU-STTrack when the LaSOT [50] data set is used.

Contextual Prompts	Experience Replay	AUC	P	P_{Norm}
✗	✗	0.595	0.616	0.698
✓	✗	0.613	0.645	0.734
✗	✓	0.597	0.623	0.698
✓	✓	**0.628**	**0.659**	**0.754**

TABLE V: Effect of the dual attention mechanism on the proposed SU-STTrack when the LaSOT [50] data set is used.

Multi-level Attention	Multi-modality Attention	AUC	P	P_{Norm}
✗	✗	0.613	0.628	0.713
✓	✗	0.610	0.623	0.710
✗	✓	0.596	0.616	0.699
✓	✓	**0.628**	**0.659**	**0.754**

tracking drift occurs at frame *111* due to the interference from visually similar elephants and background color resemblance. Given the description "the third girl on the second column", the viewpoint begins rotating at frame *51* in sequence *CheerTeam_video_09*. While tracking remains robust during the rotation process, reconstructed spatial relationships after rotation cause the bounding box to be incorrectly associated with a non-target individual which matches the description.

C. Ablation Study

To investigate the effectiveness of the components of SU-STTrack, we conducted a series of ablation experiments. For simplicity, only the LaSOT [50] data set was utilized.

1) Effect of Contextual Prompts: For the purpose of examining the impact of contextual prompts on SU-STTrack, we conducted an ablation experiment by removing the text encoder branch while retaining the image encoder branch. Without contextual prompts, the model only relied on the visual cue. Hence, its ability to incorporate the broader scene context was impaired. As shown in Table IV, our method always performed better when contextual prompts were available than that it conducted without using these prompts, no matter whether the experience replay module was used or not.

2) Effect of the Dual Attention Mechanism: The proposed dual attention mechanism contains a multi-level CAB and a multi-modality CAB. To examine the impact of the dual attention mechanism on SU-STTrack, we further conducted an ablation experiment by removing the multi-level CAB and/or the multi-modality CAB. As reported in Table V, SU-STTrack with the dual attention mechanism performed better than that without the multi-level CAB and/or the multi-modality CAB.

3) Effect of Experience Relpay: We also investigated the effect of the experience replay strategy on our SU-STTrack through replacing the experience replay module by a simple template updating approach which only utilized the most recent frame. It can be obsered from Table IV that the absence of the proposed experience replay module led to a performance decline with or without the contextual prompts.

4) Effect of the Parameters of Experience Replay: The proposed experience replay strategy is a crucial component of SU-STTrack during the online tracking task. The two parameters α and β in Equation (13) are key to determining the degree of consideration of historical templates in the subsequent frames versus the newly captured templates. When α and β are set to 1 and 0, respectively, SU-STTrack solely relies on the template of the current frame, ignoring the historical information. On the other hand, SU-STTrack is only focused on the historical information and template update is ceased in the case that α and β are set to 0 and 1, respectively. Therefore, a trade-off between the two parameters is required for achieving tracking robustness. To investigate the impact of the two parameters on SU-STTrack, we tested different combinations of the α and β values. The results are shown in Fig. 7. It can be seen that the best result was produced when both α and β were set to 0.5, which balanced the use of the current and historical templates. In this case, the AUC value obtained was 0.628.

V. Conclusion and Future Work

In this paper, we proposed a novel single-object tracking method with the linguistic contextual prompts encoded using a pre-trained Large Language Model (LLM). The method was referred to as SU-STTrack, which can be applied to both the vision-language and vision-only tracking tasks. To be specific, either the text descriptions given or that generated using a pre-trained Multi-modal Large Language Model (MLLM) were employed as the context information. SU-STTrack merged the contextual prompts extracted from the descriptions with visual features using a dual attention mechanism that we introduced on top of the Channel Attention Block (CAB), to strengthen tracking robustness and adaptability. In addition, an experience replay strategy was proposed for the sake of maintaining the long-term tracking performance by periodically refreshing the tracking template with accumulated experiences. This strategy

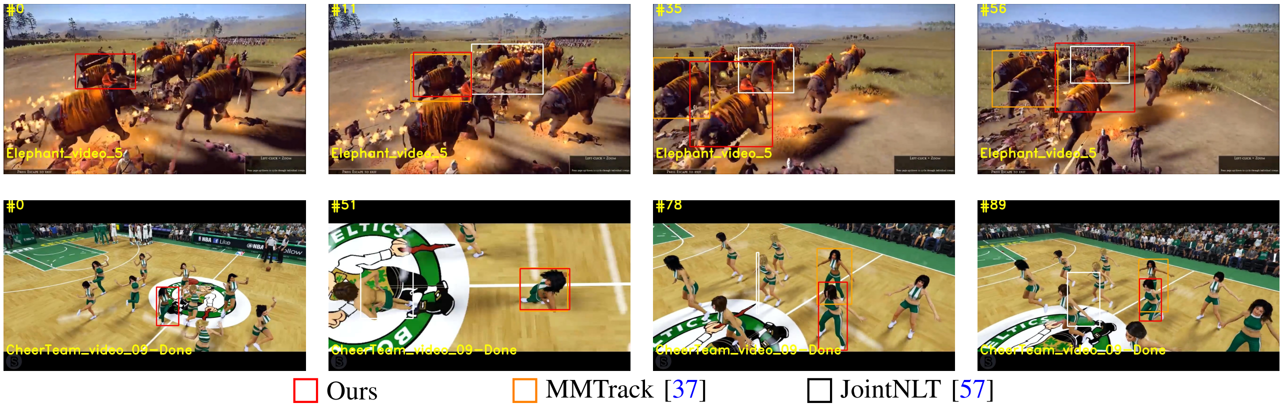


Fig. 6: Visualization of the results of our tracker and two state-of-the-art baseline trackers on the challenging sequences contained in the TNL2k [47] data set.

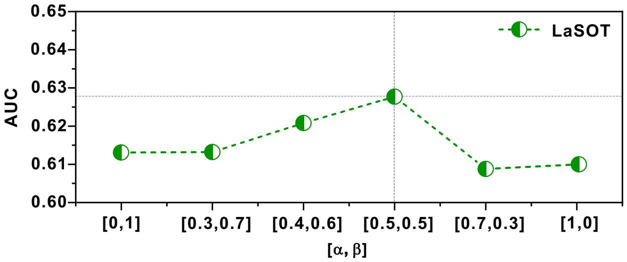


Fig. 7: Effect of different combinations of the α and β parameters used by the experience replay strategy on SU-STTrack when the LaSOT [50] data set is utilized.

is able to prevent the inference from catastrophic forgetting. Extensive experiments demonstrated the effectiveness of our SU-STTrack. The promising results should be due to the ability that our SU-STTrack manifested to integrate different modalities of features and leverage the historical information.

In our future work, we will explore the dynamic tracking strategy adaption approach based on reinforcement learning and extend the proposed SU-STTrack to multi-object tracking.

References

[1] Y. Wen, J. Huang, S. Sun, and X. Su, “Enclose and track a target of mobile robot with motion and field of view constraints based on relative position measurement,” *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 10, pp. 5110–5119, 2024.

[2] T. Xu, X.-J. Wu, X. Zhu, and J. Kittler, “Memory prompt for spatiotemporal transformer visual object tracking,” *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 8, pp. 3759–3764, 2024.

[3] P. Dwivedi, G. Routray, D. K. Jha, and R. M. Hegde, “Improving source tracking accuracy through learning-based estimation methods in sh domain: A comparative study,” *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 8, pp. 3974–3984, 2024.

[4] G. Chen, Y. Xu, X. Yang, H. Hu, H. Cheng, L. Zhu, J. Zhang, J. Shi, and X. Chai, “Target tracking control of a bionic mantis shrimp robot with closed-loop central pattern generators,” *Ocean Engineering*, vol. 297, p. 116963, 2024.

[5] Q. Wang, X. Du, D. Jin, and L. Zhang, “Real-time ultrasound doppler tracking and autonomous navigation of a miniature helical robot for accelerating thrombolysis in dynamic blood flow,” *ACS nano*, vol. 16, no. 1, pp. 604–616, 2022.

[6] W. Budiharto, E. Irwansyah, J. S. Suroso, and A. A. S. Gunawan, “Design of object tracking for military robot using pid controller and computer vision,” *ICIC Express Letters*, vol. 14, no. 3, pp. 289–294, 2020.

[7] Y. Hu, M. Wu, J. Kang, and R. Yu, “D-tracking: digital twin enabled trajectory tracking system of autonomous vehicles,” *IEEE Transactions on Vehicular Technology*, 2024.

[8] T. Zhang, C. Xu, and M.-H. Yang, “Robust structural sparse tracking,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 473–486, 2019.

[9] D. Guo, Y. Shao, Y. Cui, Z. Wang, L. Zhang, and C. Shen, “Graph attention tracking,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 9543–9552.

[10] J. Gao, T. Zhang, and C. Xu, “Graph convolutional tracking,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4649–4659.

[11] C. Fu, J. Xu, F. Lin, F. Guo, T. Liu, and Z. Zhang, “Object saliency-aware dual regularized correlation filter for real-time aerial tracking,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 12, pp. 8940–8951, 2020.

[12] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “High-speed tracking with kernelized correlation filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[13] Q. Shen, L. Qiao, J. Guo, P. Li, X. Li, B. Li, W. Feng, W. Gan, W. Wu, and W. Ouyang, “Unsupervised learning of accurate siamese tracking,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 8101–8110.

[14] X. Sun, G. Han, L. Guo, H. Yang, X. Wu, and Q. Li, “Two-stage aware attentional siamese network for visual tracking,” *Pattern Recognition*, vol. 124, p. 108502, 2022.

[15] J. Xiao, Q. Lan, L. Qiao, and A. Leonardis, “Semantic tracking: Single-target tracking with inter-supervised convolutional networks,” *arXiv preprint arXiv:1611.06395*, 2016.

[16] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, “Know your surroundings: Exploiting scene information for object tracking,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*. Springer, 2020, pp. 205–221.

[17] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. 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*, 2019, pp. 5374–5383.

[18] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu *et al.*, “A survey on vision transformer,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 1, pp. 87–110, 2022.

[19] H. He, Z. Chen, H. Liu, X. Liu, Y. Guo, and J. Li, “Practical tracking method based on best buddies similarity,” *Cyborg and Bionic Systems*, vol. 4, p. 0050, 2023.

[20] C. C. Mar, T. T. Zin, P. Tin, K. Honkawa, I. Kobayashi, and Y. Horii,

"Cow detection and tracking system utilizing multi-feature tracking algorithm," *Scientific reports*, vol. 13, no. 1, p. 17423, 2023.

[21] Y. Bai, Z. Zhao, Y. Gong, and X. Wei, "Artrackv2: Prompting autoregressive tracker where to look and how to describe," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 048–19 057.

[22] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 fps with deep regression networks," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 749–765.

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

[24] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "Atom: Accurate tracking by overlap maximization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4660–4669.

[25] A. Lukezic, J. Matas, and M. Kristan, "D3s-a discriminative single shot segmentation tracker," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7133–7142.

[26] Z. Liang and J. Shen, "Local semantic siamese networks for fast tracking," *IEEE Transactions on Image Processing*, vol. 29, pp. 3351–3364, 2020.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[28] M. Zhao, K. Okada, and M. Inaba, "Trtr: Visual tracking with transformer," *arXiv preprint arXiv:2105.03817*, 2021.

[29] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu, "Transformer tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8126–8135.

[30] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, "Learning spatio-temporal transformer for visual tracking," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 448–10 457.

[31] X. Hu, B. Zhong, Q. Liang, S. Zhang, N. Li, X. Li, and R. Ji, "Transformer tracking via frequency fusion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 2, pp. 1020–1031, 2024.

[32] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, and C. Rupprecht, "Cotracker: It is better to track together," in *European Conference on Computer Vision*. Springer, 2024, pp. 18–35.

[33] Z. Li, R. Tao, E. Gavves, C. G. Snoek, and A. W. Smeulders, "Tracking by natural language specification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6495–6503.

[34] H. Zhao, X. Wang, D. Wang, H. Lu, and X. Ruan, "Transformer vision-language tracking via proxy token guided cross-modal fusion," *Pattern Recognition Letters*, vol. 168, pp. 10–16, 2023.

[35] C. Zhang, X. Sun, Y. Yang, L. Liu, Q. Liu, X. Zhou, and Y. Wang, "All in one: Exploring unified vision-language tracking with multi-modal alignment," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 5552–5561.

[36] X. Chen, H. Peng, D. Wang20, H. Lu, and H. Hu, "Seqtrack: Sequence to sequence learning for visual object tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 14 572–14 581.

[37] Y. Zheng, B. Zhong, Q. Liang, G. Li, R. Ji, and X. Li, "Towards unified token learning for vision-language tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[38] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *Proceedings of European Conference on Computer Vision Workshops, Part II 14*. Springer, 2016, pp. 850–865.

[39] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8971–8980.

[40] H. Wang, X. Liu, Y. Li, M. Sun, D. Yuan, and J. Liu, "Temporal adaptive rgbt tracking with modality prompt," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 6, 2024, pp. 5436–5444.

[41] L. Hong, S. Yan, R. Zhang, W. Li, X. Zhou, P. Guo, K. Jiang, Y. Chen, J. Li, Z. Chen, and W. Zhang, "Onetracker: Unifying visual object tracking with foundation models and efficient tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 19 079–19 091.

[42] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang, "Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework," *CoRR*, vol. abs/2202.03052, 2022.

[43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[45] Y. Gao, C. Li, Y. Zhu, J. Tang, T. He, and F. Wang, "Deep adaptive fusion network for high performance rgbt tracking," in *Proceedings of the IEEE/CVF International conference on computer vision workshops*, 2019, pp. 0–0.

[46] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, "Modeling context in referring expressions," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 69–85.

[47] X. Wang, X. Shu, Z. Zhang, B. Jiang, Y. Wang, Y. Tian, and F. 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*, 2021, pp. 13 763–13 773.

[48] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2411–2418.

[49] M. Muller, A. Bibi, S. Giancola, S. Alsubaihi, and B. Ghanem, "Trackingnet: A large-scale dataset and benchmark for object tracking in the wild," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[50] H. Fan, H. Bai, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, Harshit, M. Huang, J. Liu *et al.*, "Lasot: A high-quality large-scale single object tracking benchmark," *International Journal of Computer Vision*, vol. 129, pp. 439–461, 2021.

[51] M. Guo, Z. Zhang, H. Fan, and L. Jing, "Divert more attention to vision-language tracking," 2022.

[52] X. Wang, C. Li, R. Yang, T. Zhang, J. Tang, and B. Luo, "Describe and attend to track: Learning natural language guided structural representation and visual attention for object tracking," *arXiv preprint arXiv:1811.10014*, 2018.

[53] Q. Feng, V. Ablavsky, Q. Bai, G. Li, and S. Sclaroff, "Real-time visual object tracking with natural language description," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2020, pp. 700–709.

[54] Q. Feng, V. Ablavsky, Q. Bai, and S. Sclaroff, "Siamese natural language tracker: Tracking by natural language descriptions with siamese trackers," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 5847–5856.

[55] Z. Yang, T. Kumar, T. Chen, J. Su, and J. Luo, "Grounding-tracking-integration," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 9, pp. 3433–3443, 2020.

[56] Y. Li, J. Yu, Z. Cai, and Y. Pan, "Cross-modal target retrieval for tracking by natural language," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4931–4940.

[57] L. Zhou, Z. Zhou, K. Mao, and Z. He, "Joint visual grounding and tracking with natural language specification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 23 151–23 160.

[58] Y. Shao, S. He, Q. Ye, Y. Feng, W. Luo, and J. 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*, 2024, pp. 19 208–19 217.

[59] U. Benchmark, "A benchmark and simulator for uav tracking," in *European Conference on Computer Vision*, 2016.

[60] H. Kiani Galoogahi, A. Fagg, C. Huang, D. Ramanan, and S. Lucey, "Need for speed: A benchmark for higher frame rate object tracking," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1125–1134.

[61] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4904–4913.

[62] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6182–6191.

[63] C. Mayer, M. Danelljan, D. P. Paudel, and L. Van Gool, "Learning target candidate association to keep track of what not to track," in *Proceedings*

of the IEEE/CVF international conference on computer vision, 2021, pp. 13 444–13 454.

[64] P. Voigtlaender, J. Luiten, P. H. Torr, and B. Leibe, "Siam r-cnn: Visual tracking by re-detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6578–6588.

[65] B. Ye, H. Chang, B. Ma, S. Shan, and X. Chen, "Joint feature learning and relation modeling for tracking: A one-stream framework," in *Proceedings of European Conference on Computer Vision*, 2022, pp. 341–357.

[66] X. Wang, Z. Chen, B. Jiang, J. Tang, B. Luo, and D. Tao, "Beyond greedy search: Tracking by multi-agent reinforcement learning-based beam search," *IEEE Transactions on Image Processing*, vol. 31, pp. 6239–6254, 2022.

[67] S. Gao, C. Zhou, C. Ma, X. Wang, and J. Yuan, "Aiatrack: Attention in attention for transformer visual tracking," in *European Conference on Computer Vision*. Springer, 2022, pp. 146–164.

[68] Y. Cui, C. Jiang, L. Wang, and G. Wu, "Mixformer: End-to-end tracking with iterative mixed attention," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 13 608–13 618.

[69] C. Mayer, M. Danelljan, G. Bhat, M. Paul, D. P. Paudel, F. Yu, and L. Van Gool, "Transforming model prediction for tracking," 2022, pp. 8731–8740.

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

[71] P. Li, B. Chen, W. Ouyang, D. Wang, X. Yang, and H. Lu, "Gradnet: Gradient-guided network for visual object tracking," in *Proceedings of the IEEE/CVF International conference on computer vision*, 2019, pp. 6162–6171.

[72] B. Alawode, F. A. Dharejo, M. Ummar, Y. Guo, A. Mahmood, N. Werghi, F. S. Khan, J. Matas, and S. Javed, "Improving Underwater Visual Tracking With a Large Scale Dataset and Image Enhancement," vol. 14, 2023.

[73] M. Danelljan, L. V. Gool, and R. Timofte, "Probabilistic regression for visual tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7183–7192.

Xuelin Liu received a bachelor's degree in Computer Science and Technology from the Northeast Forestry University (NEFU), Herbin, Heilongjiang Province, China, in 2018. She is currently a postgraduate student at Ocean University of China (OUC) working toward his master's degree in Computer Science and Technology. Her research interests include deep learning and single-object tracking.

Huiyu Zhou received a B.Eng. degree in radio technology from the Huazhong University of Science and Technology, Wuhan, China, in 1990, the M.Sc. degree in biomedical engineering from the University of Dundee, Dundee, U.K., in 2002, and the Ph.D. degree in computer vision from Heriot-Watt University, Edinburgh, U.K., in 2006. He is currently a Full Professor at the School of Computing and Mathematical Sciences, University of Leicester, Leicester, U.K. His research interests include medical image processing, computer vision, intelligent systems, and data mining.

Junyu Dong received the B.Sc. and M.Sc. degrees from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, U.K., in 2003. He joined the Ocean University of China in 2004. He is currently a Professor and the Dean of the Faculty of Information Science and Engineering, at Ocean University of China. His research interests include computer vision, underwater image processing, and machine learning, with more than ten research projects supported by the NSFC, MOST, and other funding agencies.

Jingjing Xiao obtained her bachelor's and master's degrees from the School of Mechatronics Engineering and Automation at the National University of Defense Technology in China in 2010 and 2012, respectively. She further pursued her doctoral studies at the University of Birmingham in the United Kingdom. Currently, she serves as a director and senior engineer in the Bio-Med Informatics Research Centre & Clinical Research Centre, the Second Affiliated Hospital of the Army Medical University in China. Her current research interests include medical image processing and network medicine.

Xinghui Dong received a PhD degree from Heriot-Watt University, U.K., in 2014. He worked with the Centre for Imaging Sciences, at the University of Manchester, U.K., between 2015 and 2021. Then he joined Ocean University of China in 2021. He is currently a professor at the Ocean University of China. His research interests include computer vision, defect detection, texture analysis, and visual perception.