FeatTracker: Short- and Long-Range Temporal Feature Consistency for Robust Underwater Object Tracking

Jiaqing Li[1,2], Bin Lin[2†], Chaocan Xue[3], Wu Ai[2], and Qingping Zheng[1†]

[1] School of Informatics, Xiamen University, China
[2] School of Mathematics and Statistics, Guilin University of Technology, China
[3] School of Computer Science and Engineering, Guangxi Normal University, China

Abstract. Existing underwater object tracking (UOT) methods can be broadly divided into two categories: frame-level trackers and video-level trackers. Frame-level trackers process each frame independently, using enhancement or adaptation techniques to handle underwater distortions. However, this often results in inconsistent feature styles and weakened temporal correlations. Video-level trackers attempt to improve temporal consistency by modeling dependencies across frames through autoregressive mechanisms. Despite this, they still struggle with persistent feature degradation and tracking drift caused by challenging underwater conditions. To overcome these limitations, we propose FeatTracker, a feature-level tracking framework that enforces both short- and long-range temporal feature consistency, effectively preserving semantic integrity and ensuring temporal stability for robust underwater object tracking. The framework integrates two key innovations: the Short-Range Temporal Feature Consistency module, which combines diffusion-like feature enhancement with a dynamic memory pool to mitigate degradation and preserve local temporal coherence; and the Long-Range Temporal Feature Consistency module, which leverages wavelet decomposition to separate stable structural components from transient details within historical tokens, effectively preventing the loss of trajectory details. Extensive experiments demonstrate that FeatTracker achieves state-of-the-art performance across four UOT benchmarks. Our code and models are available at `https://github.com/fishgfish/FeatTracker`.

Keywords: Underwater Object Tracking · Temporal Feature Consistency · Single Object Tracking

1 Introduction

Underwater object tracking (UOT) aims to uniquely identify and follow objects in video sequences across diverse underwater environments, given arbitrary target queries [1, 5, 16]. Compared with open-air tracking, UOT is substantially

† Corresponding authors: Bin Lin (linbin@glut.edu.cn) and Qingping Zheng (zhengqingping@xmu.edu.cn)

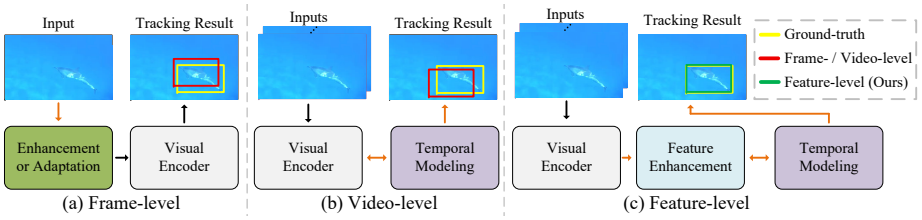

Fig. 1: Comparison of three tracking paradigms: (a) Frame-level trackers typically induce inconsistent feature styles and weak temporal correlations; (b) Video-level trackers still struggle with persistent feature degradation, which can lead to tracking drift; (c) Feature-level tracker (Ours), enhances features after the visual encoder to better adapt to underwater degradations. It enforces spatiotemporal alignment through short- and long-range consistency modules (SRTFC and LRTFC), propagating refined features to maintain temporal coherence and improve tracking robustness.

more challenging due to severe color distortion, low visibility, and the presence of visually similar distractors [2, 35]. Existing frame-level approaches primarily rely on enhancement-based preprocessing or domain adaptation to mitigate degradations [24, 36]. However, treating frames independently leads to inconsistent feature representations and weak temporal coherence (Fig. 1(a)). Moreover, the persistent visual degradation and scarcity of labeled underwater data further limit the effectiveness of these enhancement and adaptation techniques.

To overcome the temporal inconsistency of frame-level approaches, recent video-level trackers employ autoregressive modeling and sequence-to-sequence learning to capture long-range frame dependencies beyond simple frame-pair matching [29, 31, 37]. Nevertheless, as shown in Fig. 1(b), these methods still encounter challenges in underwater scenarios, including persistent feature degradation, color distortion, and low visibility, which induce feature drift and unstable temporal alignment. Off-the-shelf video-level trackers frequently struggle to adequately correct feature-level color shifts caused by underwater distortions, resulting in drift and degraded target representation. Moreover, their limited capacity to enforce stable cross-frame coherence leads to accumulated errors over time, reducing tracking robustness in complex underwater scenarios.

A fundamental question is whether feature-level enhancement can simultaneously mitigate complex underwater degradations while preserving temporal consistency for robust tracking. In underwater scenarios, the accuracy critically hinges on the diversity and robustness of the features extracted by the visual encoder, as these features form the basis for reliable target localization and discrimination. As illustrated in Fig. 1(c), enhancing features after the visual encoder stage enables the model to better adapt to complex underwater degradations by enriching semantic representations and suppressing noise early on. Moreover, directly propagating raw encoder features across frames often amplifies accumulated noise and inconsistencies inherent in degraded inputs. By instead propagating enhanced features that have been refined for robustness and discriminability, it can maintain stable, temporally coherent representations that are more resilient to environmental distractors over time.

To enhance feature quality while preserving temporal coherence, we propose **FeatTracker**, a feature-level tracking framework. FeatTracker extends video-level tracking to a feature-level framework by enforcing both short- and long-range temporal consistency through spatiotemporal alignment in feature space. FeatTracker integrates two complementary modules: *the Short-Range Temporal Feature Consistency* (SRTFC) module, which addresses temporal inconsistencies in appearance features by refining the current frame feature through a diffusion-like enhancement process and incorporating a short-range memory mechanism to maintain local temporal coherence. This combination stabilizes feature evolution across consecutive frames, effectively improving robustness in underwater tracking; and the *Long-Range Temporal Feature Consistency* (LTRFC) module, which leverages wavelet decomposition to disentangle stable structural components from transient details, thereby mitigating the loss of target trajectory details during temporal propagation. Together, these modules enable FeatTracker to achieve effective spatiotemporal feature alignment, preserving semantic fidelity and significantly boosting tracking reliability in challenging underwater environments. Our main contributions can be summarized as follows:

- We propose FeatTracker, a feature-level tracking framework that addresses the limitations of existing frame-level and video-level trackers. FeatTracker effectively mitigates feature degradation and temporal incoherence, enabling robust performance.
- We design the SRTFC module, integrating diffusion-like feature enhancement with dynamic feature memory to mitigate underwater feature degradation, preserve local temporal coherence, and prevent fragmentation resulting from frame-independent processing.
- We propose the LTRFC module, designed to preserve target trajectory details during temporal propagation by leveraging wavelet decomposition to separate low-frequency structural information from high-frequency fine details within historical tokens over time.
- Our proposed FeatTracker achieves precise spatiotemporal feature alignment and sets new state-of-the-art performance across four UOT benchmarks, significantly outperforming PUTracker (e.g., on UW-COT220, achieving 61.50 vs. 60.38 in AUC and 58.35 vs. 57.24 in Precision).

2 Related Work

Visual Object Tracking. Visual object tracking (VOT) methods fall into three main categories: discriminative correlation filter (DCF)-based approaches [4, 11, 14], which are fast but use hand-crafted features that struggle with underwater color distortion; Siamese network (SN)-based methods [3, 9, 19], which leverage deep features but are limited by early convolutional designs; and vision transformer (ViT)-based trackers like Mixformer [10] and OSTrack [34], which combines feature extraction and interaction in a unified encoder for better fusion of search and template features. Recent ViT-based trackers use autoregressive methods to improve temporal modeling: AQATrack [31] uses learnable queries

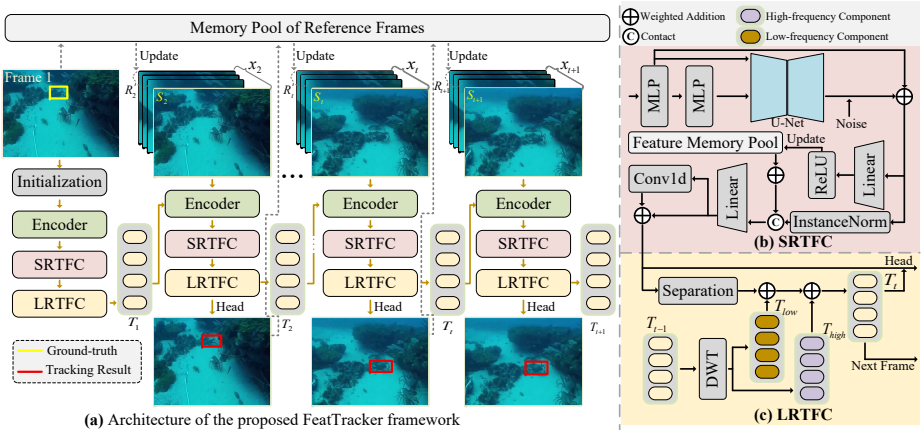

Fig. 2: Architecture of the proposed FeatTracker framework integrating complementary temporal consistency mechanisms. **(a)** illustrates the overall framework. **(b)** Short-Range Temporal Feature Consistency (SRTFC) enforces local temporal coherence by applying diffusion-based feature enhancement coupled with a dynamic feature memory.**(c)** Long-Range Temporal Feature Consistency (LRTFC) enhances discrimination against distractors and occlusions using wavelet decomposition, capturing both structural persistence and transient details.

to track appearance changes in sliding windows; ARTrack [29] treats tracking as autoregressive coordinate sequence prediction; and ODTrack [37] applies token propagation for dense context association.

Underwater Object Tracking. Underwater object tracking relies on general models for initialization to boost learning efficiency and performance due to the scarcity of large-scale annotated underwater datasets. For example, SiamFCA [26] integrates attention and image enhancement to correct color casts; UOSTrack [22] uses underwater-open-air hybrid training with Kalman filter-based motion prediction; ATCTRack [24] enhances trajectory correction and input images; OKTrack [35], based on OSTrack, applies omni-knowledge distillation for cross-domain feature alignment; PUTrack [36] employs domain-specific prompts for adaptive underwater information; and U2WTrack [32] fuses RGB and sonar images with separate enhancement strategies to handle underwater degradation. These methods are mostly frame- or video-level, limiting their robustness in challenging underwater conditions.

3 Method

3.1 Overview of FeatTracker

Given a sequence of underwater visual clips $\mathcal{V} = \{v_t\}_{t=1}^{L}$ of length L, where each frame v_t is a three-channel RGB image of height H and width W, and the first frame is annotated with a ground truth bounding box $\mathcal{B} = [x_c, y_c, w, h]$, where (x_c, y_c) and (w, h) denote the bounding box center coordinates and dimensions,

respectively, the goal of underwater object tracking is to accurately detect and localize the target object in all subsequent frames starting from the second frame, referred to as the search frame S. At each time step $t \geq 2$, S_t denotes the current search frame, and R_t represents the set of reference frames sampled from a memory pool containing historical search results.

Let $x_t = (S_t, R_t)$ denote the model input at time step t, where $R_t = \{R_t^1, R_t^2, R_t^3\}$ consists of three reference frames. The input x_t is first partitioned into non-overlapping patches of size $P \times P$ and then processed by a stack of Vision Transformer (ViT) [12] layers $\mathcal{E}$ to capture spatial and contextual relationships, producing feature representations $\mathbf{X} \in \mathbb{R}^{D \times N}$, where D is the token dimension and $N = HW/P^2$. Following ODTrack [37], the resulting features $\mathbf{X}$ are passed into a lightweight dual-branch regression head $\mathcal{H}(\mathbf{X})$, which predicts confidence scores through a classification branch and outputs bounding box coordinates through a regression branch. The model is trained using a composite loss function:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_r \mathcal{L}_{reg} + \lambda_g \mathcal{L}_{GIoU}, \tag{1}$$

where $\lambda_r = 5$ and $\lambda_g = 2$. Here, $\mathcal{L}_{cls}$ employs focal loss [23] to mitigate class imbalance, and $\mathcal{L}_{reg}$ combines L_1 loss and Generalized IoU (GIoU) loss [27] to achieve precise target localization.

Tracking accuracy in underwater scenarios is fundamentally constrained by degradations in visual features $\mathbf{X}$, which weaken both the discriminative cues required for target localization and the regression capability of the predictor $\mathcal{H}$. To enhance feature quality while maintaining temporal coherence, we propose **FeatTracker**, a feature-level tracking framework that enforces both short- and long-range temporal consistency. As illustrated in Fig. 2, FeatTracker integrates a *Short-Range Temporal Feature Consistency (SRTFC)* module that stabilizes feature evolution across adjacent frames, and a *Long-Range Temporal Feature Consistency (LTRFC)* module that leverages wavelet-based decomposition to preserve discriminative object characteristics over extended sequences.

3.2 Short-Range Temporal Feature Consistency

To address the temporal inconsistency of appearance features caused by underwater disturbances (e.g., scattering, illumination variation, and color attenuation), we introduce the Short-Range Temporal Feature Consistency (SRTFC) module. As illustrated in Fig. 2**(c)**, the SRTFC module refines the current frame feature $\mathbf{X}_t$ through a diffusion-like enhancement process and integrates a short-range memory mechanism to maintain local temporal coherence and stabilize feature evolution across consecutive frames, thereby enhancing robustness in underwater object tracking.

Diffusion-like Feature Enhancement. Visual features extracted by ViTs in underwater environments are highly susceptible to degradations such as scattering, color attenuation, and low contrast, which significantly impair tracking accuracy. To improve robustness under such domain shifts, feature perturbation

mechanisms have been widely explored in domain generalization, as they encourage adaptation to evolving feature distributions [20]. Motivated by diffusion-based formulations, which enable iterative noise-guided refinement and demonstrate a stronger capacity to model realistic distribution shifts than flow-based models or denoising autoencoders, we introduce a controlled diffusion-like feature perturbation strategy. This mechanism progressively injects structured noise into the feature space to enhance robustness and temporal stability while preserving semantic consistency.

Given the current feature $\mathbf{X}_t$, we first process it with a multi-layer perceptron (MLP) consisting of a linear projection, GELU activation, and layer normalization to obtain enriched representations $\mathbf{H}_t$. The sequence feature $\mathbf{H}_t$ is then averaged along the token dimension to produce a global representation, which is further transformed by another MLP to generate a condition vector $\mathbf{C}_t$ that captures global contextual information. Based on this condition, we simulate a controlled diffusion-like noise injection process that gradually perturbs the feature representation, thereby improving robustness to underwater degradation and enhancing temporal consistency during tracking.

Specifically, we introduce a lightweight U-Net (4.8M parameters) to perform context-aware noise modulation in the feature space. The U-Net function f_θ predicts a condition-dependent noise residual based on the current feature $\mathbf{H}_t$ and the global context vector $\mathbf{C}_t$, thereby adaptively regulating the perturbation applied to $\mathbf{H}_t$. During training, we construct a diffusion-like process with T timesteps to progressively simulate controlled perturbations under different noise intensities. At inference time, this multi-step process is approximated with a single forward pass for computational efficiency. The diffusion-like feature enhancement at diffusion step $i \in \{0, \ldots, T\}$ is formulated as:

$$\mathbf{H}_t^{i-1} = \frac{1}{\sqrt{\alpha_i}} \left(\mathbf{H}_t^i - \frac{1-\alpha_i}{\sqrt{1-\gamma_i}} f_\theta(\mathbf{C}_t^i, \mathbf{H}_t^i) \right) + \lambda_i \beta_i \epsilon_i \tag{2}$$

where α_i, γ_i, and β_i are parameters of the diffusion-like schedule controlling the noise level at step i, $\epsilon_i \sim \mathcal{N}(0, \mathbf{I})$ is Gaussian noise introducing stochastic refinement, and i denotes the diffusion timestep controlling the noise intensity. The U-Net output is given by $\mathbf{H}'_t = f_\theta(\mathbf{H}_t, \mathbf{C}_t)$.

Finally, the noise-enhanced feature $\mathbf{H}'_t$ is adaptively fused with the original sequence feature $\mathbf{H}_t$ to produce the final output:

$$\widetilde{\mathbf{X}}_t = \mathbf{g} \odot \mathbf{H}'_t + (1 - \mathbf{g}) \odot \mathbf{H}_t, \tag{3}$$

where the fusion weight $\mathbf{g}$ is generated by an MLP followed by a sigmoid activation: $\mathbf{g} = \sigma\left(\mathbf{W}_g \cdot [\mathbf{H}_t \oplus \mathbf{H}'_t]\right)$, with $\mathbf{W}_g \in \mathbb{R}^{2C \times C}$ representing a learnable weight matrix, $\oplus$ denoting concatenation along the feature dimension, and $\sigma(\cdot)$ the sigmoid function. This fusion mechanism balances information from both the original and enhanced features, yielding a refined representation $\widetilde{\mathbf{X}}_t$.

Short-Range Feature Memory Pool. To generate short-range temporal consistent features $\widetilde{\mathbf{X}}'_t$, we leverage both the diffusion-corrected current feature $\widetilde{\mathbf{X}}_t$

and a dynamic memory bank $\mathbf{M} = \{\widetilde{\mathbf{X}}_{t-1}, \widetilde{\mathbf{X}}_{t-2}, \ldots, \widetilde{\mathbf{X}}_{t-K}\}$ that stores historical features to enforce temporal consistency. Here, K denotes the memory size. The memory is updated at each time step by replacing the oldest entry with $\widetilde{\mathbf{X}}_t$, ensuring it retains the most relevant recent features. To fuse historical features with the current one, the cosine similarity between $\widetilde{\mathbf{X}}_t$ and each stored feature $\widetilde{\mathbf{X}}_k \in \mathbf{M}$ is computed as: $\cos(\widetilde{\mathbf{X}}_t, \widetilde{\mathbf{X}}_k) = \frac{\widetilde{\mathbf{X}}_t \cdot \widetilde{\mathbf{X}}_k}{\|\widetilde{\mathbf{X}}_t\| \, \|\widetilde{\mathbf{X}}_k\|}$. We select the top-$K$ most similar features based on this metric, and compute their fusion weights ω_k via a softmax function:

$$\omega_k = \frac{\exp\Big(\cos(\widetilde{\mathbf{X}}_t, \widetilde{\mathbf{X}}_k)\Big)}{\sum_{j=1}^{K} \exp\Big(\cos(\widetilde{\mathbf{X}}_t, \widetilde{\mathbf{X}}_j)\Big)}. \tag{4}$$

The fused feature vector $\widetilde{\mathbf{X}}_t^{\text{fused}}$ is then computed as a weighted sum: $\widetilde{\mathbf{X}}_t^{\text{fused}} = \sum_{k=1}^{K} \omega_k \widetilde{\mathbf{X}}_k$. This memory-based fusion effectively aligns the current feature with temporally consistent historical information, thereby improving feature coherence across frames.

To further enhance temporal coherence, style-adaptive normalization is applied to $\widetilde{\mathbf{X}}_t$, which is then aligned with the fused feature $\widetilde{\mathbf{X}}_t^{\text{fused}}$ via linear interpolation, producing the style-transferred features $\widetilde{\mathbf{X}}_t^{\text{transferred}}$. Temporal smoothing is subsequently performed using a learnable 1D convolution with kernel size 3 and padding 1, parameterized by $\mathbf{K} \in \mathbb{R}^3$, generating the smoothed features $\widetilde{\mathbf{X}}_t^{\text{smooth}}$. The final output is obtained through a weighted sum:

$$\widetilde{\mathbf{X}}_t' = \alpha \widetilde{\mathbf{X}}_t^{\text{transferred}} + (1 - \alpha) \widetilde{\mathbf{X}}_t^{\text{smooth}}, \tag{5}$$

where $\alpha = 0.7$ balances style fidelity and temporal smoothness. It yields features that are both style-consistent and temporally coherent across frames.

3.3 Long-Range Temporal Feature Consistency

To encode the target trajectory information in the sampled video sequence, we introduce a temporal token for each frame. Building upon the online dense temporal token learning framework proposed by ODTrack [37], we initialize an empty token $\mathbf{T}_{\text{empty}} \in \mathbb{R}^{1 \times D}$ as the starting point. This token is then propagated in an auto-regressive manner from the $(t-1)^{th}$ frame to the t^{th} frame. To mitigate the loss of target trajectory details during temporal propagation, we introduce a Long-Range Temporal Feature Consistency (LTRFC) module that employs a wavelet transform to decompose features, enabling the distinction and preservation of object characteristics across frames and improving temporal consistency over extended sequences.

Low and High Frequent Token Decomposition. The propagation over time often leads to noise accumulation and loss of fine details, particularly in challenging environments such as underwater scenes, where long-term tracking may gradually degrade subtle yet critical features [37]. To mitigate this issue, we incorporate wavelet decomposition into the propagation pipeline to explicitly

disentangle feature representations into low-frequency components, capturing stable global structures, and high-frequency components, preserving fine-grained details. This design is grounded in multiresolution analysis theory [25,33], where the wavelet transform hierarchically separates a signal into coarse structural information (e.g., object contours) and transient detail variations (e.g., textures).

Specifically, the token sequences are zero-padded along the token dimension to a minimum length of 4. The previous token $\mathbf{T}_t$ is reshaped into a 2D tensor to enable spatial processing. We then apply a 2D discrete wavelet transform (DWT) with the Haar wavelet, decomposing $\mathbf{T}_t^{2D}$ into one low-frequency component $\mathbf{T}_{\text{low}}$ and three high-frequency components $\mathbf{T}_{\text{LH}}, \mathbf{T}_{\text{HL}}, \mathbf{T}_{\text{HH}}$. Formally, this process is expressed as:

$$\mathbf{T}_t \xrightarrow{\text{reshape}} \mathbf{T}_t^{2D} \xrightarrow{\text{2D-DWT}} \{\mathbf{T}_{\text{low}}, \mathbf{T}_{\text{LH}}, \mathbf{T}_{\text{HL}}, \mathbf{T}_{\text{HH}}\}, \tag{6}$$

where $\mathbf{T}_{\text{low}}$ captures the coarse, structural information of the features, while $\mathbf{T}_{\text{LH}}, \mathbf{T}_{\text{HL}}, \mathbf{T}_{\text{HH}}$ encode high-frequency details along vertical, horizontal, and diagonal orientations. This separation improves the temporal consistency and robustness of features, especially in changing environments or with visual noise.

Long-Range Temporal Token Propagation. To enhance the temporal token propagation, we start by concatenating the low-frequency component $\mathbf{T}_{\text{low}}$, the high-frequency components $\mathbf{T}_{\text{LH}}, \mathbf{T}_{\text{HL}}, \mathbf{T}_{\text{HH}}$, and the current token $\mathbf{T}_t$ along the feature dimension, forming the combined tensor $\mathbf{T}'_t = [\mathbf{T}_{\text{low}}, \mathbf{T}_{\text{LH}}, \mathbf{T}_{\text{HL}}, \mathbf{T}_{\text{HH}}, \mathbf{T}_t] \in \mathbb{R}^{1\times 5D}$. This tensor is then fed into a linear layer followed by a sigmoid activation to generate fusion weights: $\mathbf{G} = \sigma\left(\mathbf{W}_t\mathbf{T}_t + \mathbf{b}\right)$, where $\mathbf{W}_t \in \mathbb{R}^{5D\times 2D}$ and $\mathbf{b} \in \mathbb{R}^{1\times 2D}$ are learnable parameters, and $\sigma(\cdot)$ denotes the sigmoid function. The output $\mathbf{G} \in \mathbb{R}^{1\times 2D}$ is then split along the feature dimension into two gating vectors: $\mathbf{G}_1, \mathbf{G}_2 \in \mathbb{R}^{1\times D}$.

The high-frequency components are summed to form a consolidated high-frequency representation $\mathbf{T}_{\text{HF}} = \mathbf{T}_{LH} + \mathbf{T}_{HL} + \mathbf{T}_{HH}$. This is then modulated by the first weight tensor $\mathbf{G}_1$ through element-wise multiplication: $\mathbf{T}_{\text{high}} = \mathbf{T}_{\text{HF}} \odot \mathbf{G}_1$. Similarly, the low-frequency component $\mathbf{T}_{low}$ and the current token $\mathbf{T}_t$ are combined using the second weight tensor $\mathbf{G}_2$:

$$\mathbf{T}'_{\text{low}} = \mathbf{T}_{low} \odot \mathbf{G}_2 + \mathbf{T}_t \odot (\mathbf{1} - \mathbf{G}_2), \tag{7}$$

where $\mathbf{1}$ denotes a tensor of ones with the same dimensions as $\mathbf{G}_2$. The results are concatenated along the feature dimension to form $\mathbf{T}_{\text{fused}} = [\mathbf{T}'_{\text{low}}, \mathbf{T}_{\text{high}}] \in \mathbb{R}^{1\times 2D}$. Finally, this fused tensor is projected back to the feature dimension D through a linear layer: $\mathbf{T}_{\text{out}} = \mathbf{W}_p\mathbf{T}_{\text{fused}} + \mathbf{b}_p$, where $\mathbf{W}_p \in \mathbb{R}^{2D\times 1}$ is a learnable weight and $\mathbf{b}_p \in \mathbb{R}^D$ is a bias term. The output $\mathbf{T}_{\text{out}} \in \mathbb{R}^{1\times D}$ serves as the propagated token for the next frame, ensuring temporal coherence.

4 Experiments

4.1 Implementation Details

Training. The model is based on the ViT-Base architecture [12] initialized with weights from MAE pre-training [17]. To address the scarcity of underwater

data, we adopt a composite training set comprising LaSOT [13], WebUOT-1M [35], HUT290 [36], and UVOT400 [1], implementing UOSTrack's [22] hybrid training strategy. The input with template frames resized to 128×128 pixels and search regions to 256×256 pixels. Optimization is performed using the AdamW optimizer with learning rates set to 1×10^{-5} for the backbone and 1×10^{-4} for other components, accompanied by 1×10^{-4} weight decay regularization. The learning rate drops by a factor of 10 after 240 epochs. Training spans 300 epochs with random batch sampling, executed on a server equipped with three 24 GB NVIDIA GeForce 4090 GPUs using a batch size of 5.

Inference. During inference, FeatTracker processes videos using its feature correction pipeline to maintain temporal coherence in dynamic underwater scenes. All tests are conducted on a single NVIDIA GeForce 4090 GPU. Notably, during the diffusion inference stage, the model adopts the optimal single-step configuration with the lightweight U-Net, achieving 32.35 FPS.

4.2 Comparison with the State-of-the-Arts

We compare FeatTracker against leading underwater trackers, including UOSTrack [22], OKTrack [35], and PUTrack [36] (specifically designed and trained for UOT), as well as representative visual object trackers such as OSTrack [34], ODTrack [37], MCITrack [18], and recent SAM/Mamba-based methods like DAM4SAM [28] and MambaLCT [21]. This comprehensive comparison evaluates performance across diverse underwater challenges.

Overall Performance Evaluation. As summarized in Table 1, our FeatTracker achieves state-of-the-art performance across all four UOT benchmarks, validating the effectiveness of the proposed feature-level framework. It attains the highest AUC scores on WebUOT-1M (67.62%), UW-COT220 (61.50%), VMAT (61.25%), and UTB180 (81.40%), and also leads in precision (P) on the latter three datasets. This consistent superiority highlights its enhanced robustness to diverse underwater degradations. While specialized UOT methods like UOSTrack* show strong results on WebUOT-1M (AUC 67.50%), they are outperformed by FeatTracker on other benchmarks. Among general-purpose trackers, DAM4SAM is a strong competitor, particularly on UTB180 (AUC 81.21%), and MCITrack performs well on VMAT (AUC 60.89%). Although FeatTracker's average FPS (32.35) is lower than some high-speed trackers (e.g., HIPTrack at 92.14 FPS), it maintains real-time capability and offers a favorable trade-off between high tracking accuracy and practical efficiency.

Attribute-Based Evaluation. Fig. 3 visually compares with ODTrack, PUTrack, and OKTrack on WebUOT-1M. FeatTracker maintains robust tracking under severe greenish distortion, low visibility, and similar distractors, where other methods lose the target early or confuse identities. It also successfully handles full occlusion, recovering accurately after the target reappears, while other trackers exhibit drift or scale errors. Collectively, these quantitative and

Table 1: Comparison of the area under the curve (AUC), precision (P) metrics, average FPS, and Params with state-of-the-art methods on four UOT benchmarks: WebUOT-1M, UW-COT220, VMAT, and UTB180. Asterisk (*) denotes trackers specifically designed for UOT, and they use the same training dataset. The top two performers in each metric category are highlighted in red and blue respectively.

Method	WebUOT-1M		UW-COT220		VMAT		UTB180		Avg. FPS	Params (M)
	AUC (%)	P (%)	AUC (%)	P (%)	AUC (%)	P (%)	AUC (%)	P (%)		
Generic Visual Trackers										
TransT [8]	54.60	49.84	49.34	41.62	43.52	52.94	57.61	50.79	44.46	-
GRM [15]	58.86	55.35	56.14	50.12	57.63	71.68	63.99	58.62	35.16	-
OSTRack [34]	58.50	54.35	53.85	47.26	52.78	65.92	62.51	55.90	73.91	92.1
DropTrack [30]	59.94	56.31	57.03	52.36	55.82	68.90	63.47	58.32	27.25	-
SeqTrack [7]	57.63	54.56	55.95	50.64	55.67	69.97	60.73	55.25	28.09	89.0
ARTrack [29]	62.44	59.69	60.02	55.89	58.67	74.17	68.80	65.02	37.31	181.0
HIPTrack [6]	61.58	58.01	57.90	53.63	55.83	68.96	66.14	60.65	92.14	120.4
ODTrack [37]	63.82	60.88	60.38	56.38	54.66	67.73	69.96	64.48	34.42	92.0
MCITrack [18]	62.02	60.44	60.10	57.24	60.89	74.92	65.37	63.68	24.28	88.0
MambaLCT [21]	62.52	59.66	60.02	55.28	56.16	72.07	67.31	60.42	28.82	72.0
DAM4SAM [28]	66.18	67.99	60.28	56.48	56.98	72.41	81.21	84.74	20.32	81.0
Underwater-Specific Trackers										
UOSTrack [22]	60.57	55.96	56.87	50.94	55.84	67.90	66.65	60.05	70.08	-
UOSTrack*	67.50	67.45	59.14	54.79	54.99	68.11	80.99	83.79	70.08	-
OKTrack [35]	62.05	59.13	59.17	53.82	53.82	66.83	69.24	66.46	69.78	92.1
OKTrack*	52.03	44.62	44.90	35.76	41.16	51.02	58.00	46.22	69.78	92.1
PUTrack [36]	54.06	49.56	47.05	38.54	43.14	54.12	73.12	70.47	28.17	173.7
PUTrack*	64.64	62.43	58.75	52.68	51.17	64.54	71.46	68.86	28.17	173.7
FeatTracker*	67.62	68.84	61.50	58.35	61.25	78.07	81.40	86.52	32.35	98.7

qualitative comparisons once again verify the robustness of the proposed method. Fig. 4 shows FeatTracker's advantage in addressing key underwater challenges on WebUOT-1M. It achieves 78.7% AUC under Light Brown color distortion and 72.2% AUC in Light Yellow conditions, demonstrating strong robustness to color variations. In low visibility scenarios, FeatTracker attains 62.9% AUC, while effectively handling similar distractors with 60.9% AUC. For full occlusion challenges, it achieves 47.7% AUC. Additionally, it shows notable strength against camouflage with 63.7% AUC and maintains 62.2% AUC during fast motion, confirming its ability to tackle diverse underwater complexities.

4.3 Improvement of Baseline

Module ablation experiments on the VMAT benchmark using ODTrack [37] as baseline (①) reveal progressive performance gains. The ② incorporating a Hybrid Training Strategy achieves 55.54% AUC, demonstrating 0.88% improvement through domain adaptation. The ③ adding a Diffusion-like Feature Enhancement, gains 1.58% AUC to 57.12%, resolving color distortion and low visibility. The ④ integrating Long-Range Temporal Feature Consistency shows 56.62% AUC, providing 0.96% improvement against distractors. Notably, this gain is achieved without increasing computational cost, as the FLOPs remain identical due to the shared backbone, and the LRTFC module introduces only negligible overhead through efficient wavelet-based processing. The ⑤ combining en-

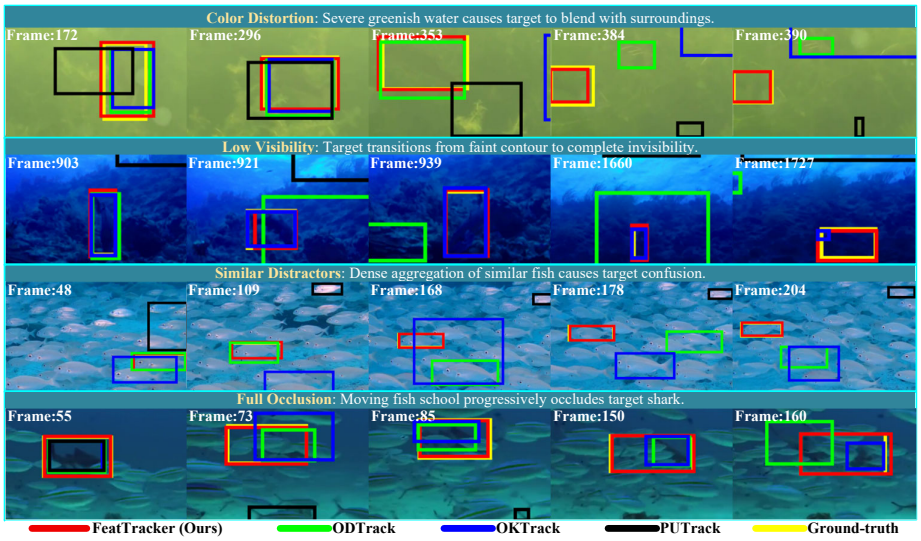

Fig. 3: Qualitative comparison results on WebUOT-1M demonstrate robustness against key underwater challenges, including color distortion, low visibility, similar distractors, and full occlusion.

hancement with Short-Range Temporal Feature Consistency, achieves 58.73% AUC, 1.61% gain through temporal coherence maintenance. The ⑥ merging diffusion correction with long-range consistency reaches 57.57% AUC. The full FeatTracker (⑦) achieves 61.25% AUC, +6.59 point gain, with 32.35 FPS. Although the DFE increases computation, it still maintains real-time tracking.

4.4 Ablation Studies

This section systematically validates the effectiveness of the proposed core modules through comprehensive ablation studies. Specifically, we compare the diffusion-like feature enhancement (DFE) with other feature enhancement methods, investigate the impact of the number of diffusion steps, evaluate different update strategies for the SRTFC module's memory, and examine various feature separation methods for the LRTFC module. Additionally, visualizations are provided to qualitatively demonstrate the modules' capabilities in enhancing feature robustness and temporal coherence.

Comparison of Various Feature Enhancement Methods. Table 4 demonstrates the superior performance of our proposed diffusion-like feature enhancement method on the VMAT dataset, achieving an AUC of 61.25% and precision of 78.07%, which outperforms both GAN-based (56.04% AUC, 70.98% precision) and Transformer-based (57.41% AUC, 72.07% precision) alternatives. This performance advantage stems from the progressive denoising mechanism that iteratively refines features through controlled noise injection and conditional

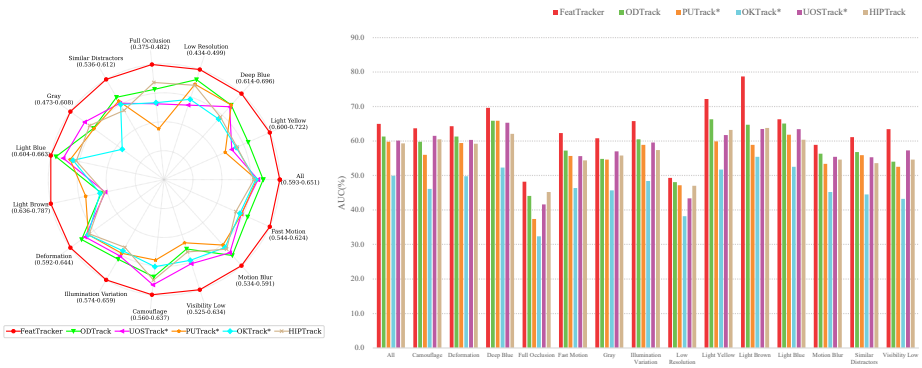

Fig. 4: AUC of different attributes on WebUOT-1M.

Table 2: We conduct a module ablation study on the VMAT benchmark. Top-2 performers in AUC and precision (P) metrics are highlighted in red and blue. Hybrid Training Strategy (HTS) refers to the replacement of training datasets. The Diffusion-like Feature Enhancement module (DFE), Short-Range Temporal Feature Consistency (SRTFC), and Long-Range Temporal Feature Consistency (LRTFC) constitute proposed modules with SRTFC dependent on DFE. The baseline is built upon ODTrack trained using the Hybrid Training Strategy.

	HTS	DFE	SRTFC	LRTFC	AUC (%)	P (%)	FPS	Params (M)	FLOPs (G)
①	✗	✗	✗	✗	54.66	67.73	34.4	92.1	73.0
②	✓	✗	✗	✗	55.54	68.86	34.4	92.1	73.0
③	✓	✓	✗	✗	57.12	71.65	33.1	96.9	76.9
④	✓	✗	✗	✓	56.62	70.17	34.4	92.1	73.0
⑤	✓	✓	✓	✗	58.70	73.70	32.3	98.7	78.1
⑥	✓	✓	✗	✓	57.57	71.80	33.1	96.9	76.9
⑦	✓	✓	✓	✓	61.25	78.07	32.3	98.7	78.1

guidance using a U-Net architecture, effectively addressing underwater degradation while maintaining temporal coherence via adaptive fusion. The method also ensures practical efficiency with 98.70M parameters and a processing speed of 32.35 FPS, preventing fragmentation issues common in frame-independent processing and making it suitable for challenging underwater tracking scenarios where feature consistency is paramount.

The Optimal DFE's Steps ***($T = 700$).*** This investigation examines the impact of diffusion step selection on tracking performance, emphasizing the critical role of step calibration in balancing feature refinement and computational efficiency through the diffusion-like enhancement process described in Section 3.2. As quantitatively demonstrated in Table 3, optimal results are achieved at 700 steps with 61.25% AUC and 78.07% precision, where the U-Net architecture successfully estimates noise injection through the reverse diffusion process with sufficient feature reconstruction. Performance declines to 60.01% AUC at 600 steps due to inadequate refinement; shorter schedules fail to address color distortion,

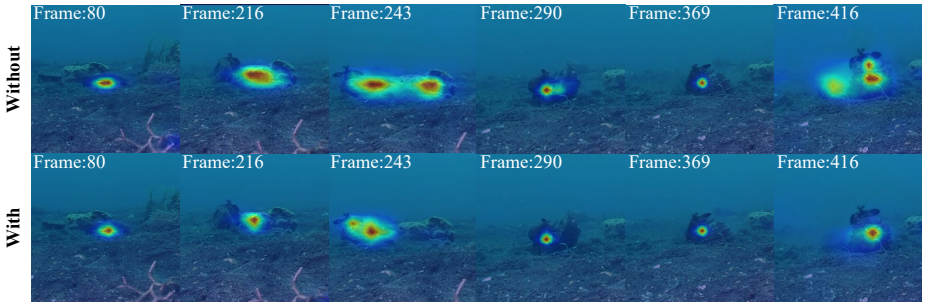

Fig. 5: Visualization comparison of the DFE's ablation study, with heatmaps derived from prediction head outputs.

Table 3: Comparison of DFE's diffusion steps.

Steps	600	700	800	900
AUC (%)	60.01	61.25	60.11	59.60
P (%)	75.92	78.07	76.07	75.55

Table 4: Comparison of different feature enhancements on VMAT.

Method	**AUC** (%)	**P** (%)	FPS	Params (M)
Diffusion-like	61.25	78.07	32.35	98.70
GAN	56.04	70.98	23.62	111.13
Transformer	57.41	72.07	28.23	100.52

whereas increasing to 800 and 900 steps reduces AUC to 60.11% and 59.60%, respectively, due to over-smoothing that erodes discriminative details. The visual analysis in Fig. 5 demonstrates that DFE counteracts underwater degradation by reducing non-target dispersion through noise injection and sharpening boundary focus via the adaptive fusion mechanism.

Effectiveness of FIFO Strategy in SRTFC. This analysis examines historical style update strategies for the SRTFC module's dynamic memory matrix, demonstrating that FIFO's rotating buffer replacement achieves optimal temporal coherence with 61.25% AUC by preserving recent degradation patterns while discarding obsolete data. This approach outperforms exponential moving averaging (60.33% AUC), which oversmooths temporal variations, global mean aggregation (59.24% AUC) that dilutes distinctive style characteristics, and random replacement (60.68% AUC), which introduces representation inconsistencies. The buffer mechanism effectively maintains style persistence across consecutive frames, providing robust feature alignment against underwater challenges, including color distortion and illumination variations, through selective retention of relevant stylistic information. As visualized in Fig. 6, the ablation study demonstrates SRTFC's efficacy in maintaining temporal coherence by stabilizing attention weights across consecutive frames and enhancing feature consistency through style-adaptive normalization and smoothing operations.

Effect of Haar Decomposition in LRTFC. This study tests four token separation methods in LRTFC on VMAT. *Haar* uses wavelet decomposition to split historical tokens into stable structures and transient details, clearly keeping complementary data. *Fourier* employs spectral masking to isolate frequency

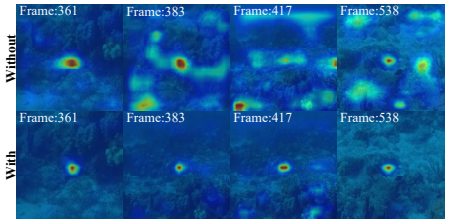

Fig. 6: Visualization comparison of the SRTFC's ablation study, with heatmaps derived from prediction head outputs.

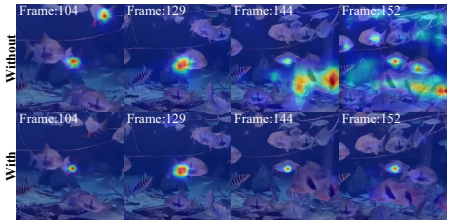

Fig. 7: Visualization comparison of the LRTFC's ablation study, with heatmaps derived from prediction head outputs.

Table 5: Comparison of SRTFC's update strategy.

Strategy	*FIFO*	*Ema*	*Mean*	*Random*
AUC (%)	61.25	60.33	59.24	60.68
P (%)	78.07	76.61	75.20	76.91

Table 6: Comparison of LRTFC's separation.

Method	*Haar*	*Fourier*	*Attention*	*Inception*
AUC (%)	61.25	60.71	60.76	60.31
P (%)	78.07	77.15	77.11	76.74

parts. *Attention* applies spatial attention for component separation. *Inception* uses fixed convolutional kernels to get multi-scale features. As shown in Table 6, *Haar* scores best with 61.25% AUC, topping *Fourier* by 0.54%, *Attention* by 0.49%, and *Inception* by 0.94%. These findings prove that clear separation of persistent structures and transient details via wavelet analysis best boosts discrimination against distractors and occlusion, solving issues in standard temporal methods. As visually demonstrated in Fig. 7, the ablation study illustrates LRTFC's capability to suppress background distractions and maintain feature integrity through wavelet-based structural preservation.

5 Conclusion

This paper presents FeatTracker, a feature-aligned consistency framework designed for UOT that achieves spatiotemporal feature alignment. It extends video-level tracking to a feature-level framework, introducing spatiotemporal alignment via diffusion-like enhancement and wavelet decomposition to mitigate underwater degradation. The framework integrates two core components: the Short-Range Temporal Feature Consistency module, which employs diffusion-like feature enhancement with dynamic style memory retention to resolve feature degradation while maintaining temporal coherence, and the Long-Range Temporal Feature Consistency module, which utilizes wavelet decomposition to enhance discrimination against distractors by separating historical tokens into stable structures and transient details. This approach addresses feature degradation while preserving spatiotemporal consistency, demonstrating competitive performance on four UOT benchmarks.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62502448 and in part by the Guangxi Natural Science Foundation under Grant No. 2026GXNSFAA00640132.

References

1. Alawode, B., Dharejo, F.A., Ummar, M., Guo, Y., Mahmood, A., Werghi, N., Khan, F.S., Matas, J., Javed, S.: Improving underwater visual tracking with a large scale dataset and image enhancement (2023), arXiv preprint arXiv:2308.15816
2. Alawode, B., Guo, Y., Ummar, M., Werghi, N., Dias, J., Mian, A., Javed, S.: Utb180: A high-quality benchmark for underwater tracking. In: ACCV. pp. 3326–3342 (2022)
3. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: Fully-convolutional siamese networks for object tracking. In: ECCV 2016 Workshops. pp. 850–865 (2016)
4. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: CVPR. pp. 2544–2550 (2010)
5. Cai, L., McGuire, N.E., Hanlon, R., Mooney, T.A., Girdhar, Y.: Semi-supervised visual tracking of marine animals using autonomous underwater vehicles. IJCV **131**(6), 1406–1427 (2023)
6. Cai, W., Liu, Q., Wang, Y.: Hiptrack: Visual tracking with historical prompts. In: CVPR. pp. 19258–19267 (2024)
7. Chen, X., Peng, H., Wang, D., Lu, H., Hu, H.: Seqtrack: Sequence to sequence learning for visual object tracking. In: CVPR. pp. 14572–14581 (2023)
8. Chen, X., Yan, B., Zhu, J., Wang, D., Yang, X., Lu, H.: Transformer tracking. In: CVPR (2021)
9. Chen, Z., Zhong, B., Li, G., Zhang, S., Ji, R.: Siamese box adaptive network for visual tracking. In: CVPR. pp. 12108–12117 (2020)
10. Cui, Y., Jiang, C., Wang, L., Wu, G.: Mixformer: End-to-end tracking with iterative mixed attention. In: CVPR. pp. 13608–13618 (2022)
11. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: BMVC (2014)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
13. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C., Ling, H.: Lasot: A high-quality benchmark for large-scale single object tracking. In: CVPR (2019)
14. Galoogahi, H.K., Fagg, A., Lucey, S.: Learning background-aware correlation filters for visual tracking. In: ICCV (2017)
15. Gao, S., Zhou, C., Zhang, J.: Generalized relation modeling for transformer tracking. In: CVPR. pp. 18686–18695 (2023)
16. González-Sabbagh, S.P., Robles-Kelly, A.: A survey on underwater computer vision. ACM Comput. Surv. **55**(13s), 268:1–268:39 (2023)
17. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: CVPR. pp. 16000–16009 (2022)

18. Kang, B., Chen, X., Lai, S., Liu, Y., Liu, Y., Wang, D.: Exploring enhanced contextual information for video-level object tracking. In: AAAI. pp. 4194–4202 (2025)
19. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: CVPR. pp. 6974–6983 (2018)
20. Li, C., Zhang, D., Huang, W., Zhang, J.: Cross contrasting feature perturbation for domain generalization. In: ICCV. pp. 1327–1337 (2023)
21. Li, X., Zhong, B., Liang, Q., Li, G., Mo, Z., Song, S.: Mambalct: Boosting tracking via long-term context state space model. In: AAAI. pp. 4986–4994 (2025)
22. Li, Y., Wang, B., Li, Y., Liu, Z., Huo, W., Li, Y., Cao, J.: Underwater object tracker: Uostrack for marine organism grasping of underwater vehicles. Ocean Engineering **285**, 115449 (2023)
23. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: ICCV (2017)
24. Luo, X., Yuan, D., Shu, X., Liu, Q., Chang, X., He, Z.: Adaptive trajectory correction for underwater object tracking. IEEE TCSVT pp. 1–1 (2025)
25. Mallat, S.G.: A theory for multiresolution signal decomposition: The wavelet representation. IEEE TPAMI **11**(7), 674–693 (1989)
26. Mei, Y., Yan, N., Qin, H., Yang, T., Chen, Y.: Siamfca: A new fish single object tracking method based on siamese network with coordinate attention in aquaculture. Computers and Electronics in Agriculture **216**, 108542 (2024)
27. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. In: CVPR (2019)
28. Videnovic, J., Lukezic, A., Kristan, M.: A distractor-aware memory for visual object tracking with SAM2. In: CVPR (2025)
29. Wei, X., Bai, Y., Zheng, Y., Shi, D., Gong, Y.: Autoregressive visual tracking. In: CVPR. pp. 9697–9706 (2023)
30. Wu, Q., Yang, T., Liu, Z., Wu, B., Shan, Y., Chan, A.B.: Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks. In: CVPR. pp. 14561–14571 (2023)
31. Xie, J., Zhong, B., Mo, Z., Zhang, S., Shi, L., Song, S., Ji, R.: Autoregressive queries for adaptive tracking with spatio-temporal transformers. In: CVPR. pp. 19300–19309 (2024)
32. Yan, K., Qian, W., Bi, C., Liu, J.: United-modality underwater object tracking via distinct enhancing and high-dimensional fusion. Ocean Engineering **343**, 123224 (2026)
33. Yao, T., Pan, Y., Li, Y., Ngo, C.W., Mei, T.: Wave-vit: Unifying wavelet and transformers for visual representation learning. In: ECCV. pp. 328–345 (2022)
34. Ye, B., Chang, H., Ma, B., Shan, S., Chen, X.: Joint feature learning and relation modeling for tracking: A one-stream framework. In: ECCV. pp. 341–357 (2022)
35. Zhang, C., Liu, L., Huang, G., Wen, H., Zhou, X., Wang, Y.: Webuot-1m: Advancing deep underwater object tracking with a million-scale benchmark. In: NIPS. pp. 50152–50167 (2024)
36. Zhang, Q., Song, W., Liu, C., Zhang, M.: Putrack: Improved underwater object tracking via progressive prompting. IEEE Transactions on Industrial Informatics **21**(5), 3986–3995 (2025)
37. Zheng, Y., Zhong, B., Liang, Q., Mo, Z., Zhang, S., Li, X.: Odtrack: Online dense temporal token learning for visual tracking. In: AAAI. pp. 7588–7596 (2024)