2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Progressive Representation Learning for Real-Time UAV Tracking

Changhong Fu[1*], Xiang Lei[2], Haobo Zuo[3], Liangliang Yao[1], Guangze Zheng[3], and Jia Pan[3]

***Abstract*— Visual object tracking has significantly promoted autonomous applications for unmanned aerial vehicles (UAVs). However, learning robust object representations for UAV tracking is especially challenging in complex dynamic environments, when confronted with aspect ratio change and occlusion. These challenges severely alter the original information of the object. To handle the above issues, this work proposes a novel progressive representation learning framework for UAV tracking, *i.e.*, PRL-Track. Specifically, PRL-Track is divided into coarse representation learning and fine representation learning. For coarse representation learning, two innovative regulators, which rely on appearance and semantic information, are designed to mitigate appearance interference and capture semantic information. Furthermore, for fine representation learning, a new hierarchical modeling generator is developed to intertwine coarse object representations. Exhaustive experiments demonstrate that the proposed PRL-Track delivers exceptional performance on three authoritative UAV tracking benchmarks. Real-world tests indicate that the proposed PRL-Track realizes superior tracking performance with 42.6 frames per second on the typical UAV platform equipped with an edge smart camera. The code, model, and demo videos are available at https://github.com/vision4robotics/PRL-Track.**

I. Introduction

Robust visual object tracking is fundamental for intelligent unmanned aerial vehicle (UAV) applications, *e.g.*, task planning [1], biodiversity protection [2], and target localization [3]. During the above extensive applications, UAV trackers aim to predict the location of the object in subsequent frames, starting from the initial position in the first frame. Driven by large-scale datasets with manual annotations, Siamese trackers [4]–[7] have shown promising performance by adopting convolutional neural networks (CNNs) to learn object representations. However, when encountered with complex dynamic environments, *e.g.*, aspect ratio change and occlusion, these trackers struggle to obtain robust object representations due to limited representation capabilities of lightweight CNNs like AlexNet [8]. Although trackers with deeper backbones, *e.g.*, ResNet [9], can better learn object representations, they fail to meet the real-time requirement constrained by limited computational resources on UAVs. ***Hence, robust object representations for UAV tracking are far from sufficient in complex dynamic environments.***

One promising approach is to explore multi-scale features oriented to UAV tracking tasks [4]. Specifically, convolutional operations are adopted to aggregate multi-scale features from different layers, which contribute to alleviating feature degradation due to occlusion during UAV tracking. However, with limited receptive fields of convolutional kernels, CNNs lack the modeling ability of long-range dependencies [10]. Consequently, it is challenging to capture global context information between multi-scale features. Recently, Vision Transformer (ViT) [11] has exhibited tremendous potential in modeling long-range dependencies by virtue of attention mechanisms. The introduction of ViT into the Siamese trackers addresses the shortcomings of traditional CNN-based trackers in learning global information. Moreover, the intrinsic global modeling capability of ViT proves to be advantageous in tackling appearance variations, *e.g.*, aspect ratio change [12]. Nonetheless, compared with CNN, ViT tends to ignore local spatial information, which decreases the discriminability of image objects [13]. Besides, the quadratic computational complexity and memory cost of the attention mechanisms are obstacles to its wide deployment on embedded processors in UAVs, which have limited computing resources. ***Therefore, how to extract more reliable information and then generate robust object representations for UAV tracking is worth exploring carefully.***

To fully exploit the global context information and local spatial information, integrating CNNs and ViT represents a promising complementary coupling. Given the strength of CNNs in fast convergence and filtering redundant information [14], [15], they are well-suited for extracting object local information from images to form coarse object representations. Subsequently, ViT utilizes coarse object representations to refine and enhance the understanding of global context information, thereby generating robust fine object representations. However, considering the distinctions

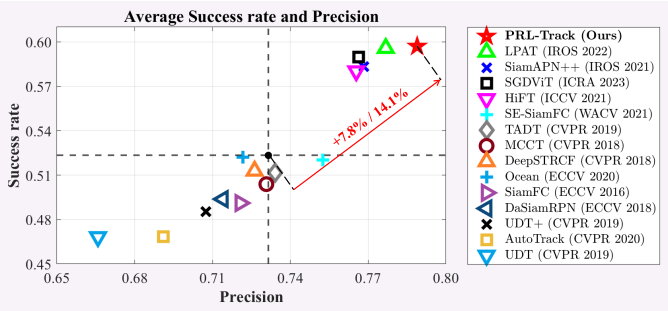

Fig. 1. Overall comparison of the proposed PRL-Track with other 14 state-of-the-art (SOTA) trackers on the combination of UAV tracking benchmarks. PRL-Track achieves more robust performance than other 14 SOTA trackers. Specifically, PRL-Track surpasses the average precision and success rate of the 14 trackers (**black dot**) by **7.8%** and **14.1%**, respectively.

*Corresponding author
[1]C. Fu and L. Yao are with the School of Mechanical Engineering, Tongji University, Shanghai 201804, China. *Email: changhongfu@tongji.edu.cn*
[2]X. Lei is with the School of Software Engineering, Tongji University, Shanghai 201804, China.
[3]H. Zuo, G. Zheng, and J. Pan are with the Department of Computer Science, the University of Hong Kong, Hong Kong, China.

in feature space between the plain CNNs and ViT, directly concatenating them leads to performance degradation [16], [17]. ***Therefore, how to effectively integrate CNNs and ViT for real-time UAV tracking is a problem worth exploring.***

This work proposes a novel progressive representation learning framework, namely PRL-Track, which consists of CNN-based coarse representation learning and ViT-based fine representation learning. Leveraging the complementary strengths of the CNNs and ViT, PRL-Track can learn robust fine object representations, achieving satisfactory performance when encountering challenges such as occlusion and aspect ratio change during UAV tracking. Fig. 1 highlights the impressive performance of PRL-Track in UAV tracking, outperforming other 14 state-of-the-art (SOTA) trackers in terms of average precision and success rate. The main contributions of this work are as follows:

- A novel progressive representation learning framework dubbed PRL-Track, is proposed to learn robust fine object representations for UAV tracking via a coarse-to-fine perspective, thus improving tracking performance.
- An innovative appearance-aware regulator is developed to mitigate appearance interference and extract useful information from shallow features for coarse representation learning. Besides, a convenient semantic-aware regulator is designed to capture semantic information and promote the concentration of deep features.
- A new hierarchical modeling generator is proposed to augment the comprehension of contextual information by fusing coarse object representations for fine representation learning, further generating robust fine object representations for UAV tracking.
- Comprehensive evaluations confirm that PRL-Track achieves SOTA performance, validating the power of the proposed framework. Real-world tests conducted on the typical UAV platform demonstrate the superior efficiency and robustness of PRL-Track in practical scenarios.

II. Related Work

A. UAV Tracking

Siamese trackers [4], [18], [19] have gained popularity and promoted the development of UAV applications owing to their remarkable tracking performance. These trackers utilize a CNN-based backbone to extract features of both the template patch and search patch, followed by a correlation-based network to calculate the similarity between them. Compared with correlation filter-based trackers [20], [21], fully CNN-based trackers further exploit the local spatial information, thus improving tracking performance. As a pioneer, SiamFC [18] introduces the Siamese framework into object tracking for similarity matching. Inspired by the region proposal network, SiamRPN [22] combines a Siamese network with regression and classification branches, achieving efficient classification and accurate prediction. However, trackers with fully CNN-based architecture lack effective long-range dependency modeling, which means they often struggle to capture global context information. Thus, it is difficult to ensure reliable tracking in complex dynamic environments. To address this issue, ViT [11] has been introduced into object tracking, owing to its high representational capacity for global context information. ViT integrates global contextual information by decomposing the image into fixed-size blocks and processing them with Transformer architecture. TransT [23] proposes a ViT-based feature fusion model for object tracking, achieving promising performance. HiFT [12] introduces a ViT structure optimized for efficient multi-feature fusion, thereby augmenting tracking robustness. SGDViT [24] designs a saliency-guided dynamic ViT to capture similarity and incorporate information. However, the attention mechanism in ViT often ignores local feature details and object spatial structures [13]. Therefore, a promising approach to overcome these limitations is the integration of CNNs and ViT, leveraging the strengths of both architectures in the context of UAV tracking. CNNs can capture local spatial information, which contributes to maintaining accuracy in the rapid environments in which UAVs operate. By integrating this with ViT's ability to model global context, the framework can better understand broader scene dynamics, enabling more stable tracking across wide fields of view.

B. Representation Learning

Representation learning aims to acquire object representations that facilitate the utilization of reliable information when constructing classifiers or predictors [25]. Deep neural networks (DNNs) are commonly employed to extract object representations in visual tasks [8], [26]. Compared with conventional hand-crafted representations, DNNs tend to learn more comprehensive representations [27]. Previous works on representation learning have yielded notable frameworks and methodologies. UniFormer [16] designs a concise unified framework and integrates the strength of CNNs and the ViT, realizing efficient spatiotemporal representation learning. EsViT [28] formulates an efficient self-supervised ViT for representation learning, achieving superior transfer performance in downstream tasks. MARLIN [29] employs a facial video masked autoencoder to learn generic and robust facial representations. HRNet [27] proposes to uphold high-resolution object representations throughout the entire workflow, thereby ensuring the reliability of object representations. Despite the rapid development mentioned above, object representation learning via a coarse-to-fine perspective for real-time UAV tracking has not been investigated yet. Besides, most existing tracking methods [12], [18] struggle to maintain excellent performance in dynamic environments due to limited computing resources and challenges, such as partial occlusion and aspect ratio change. Consequently, an effective progressive representation learning framework for UAV tracking is urgently needed.

III. Proposed Method

As depicted in Fig. 2, the proposed PRL-Track is divided into coarse representation learning and fine representation learning. The coarse representation learning generates coarse

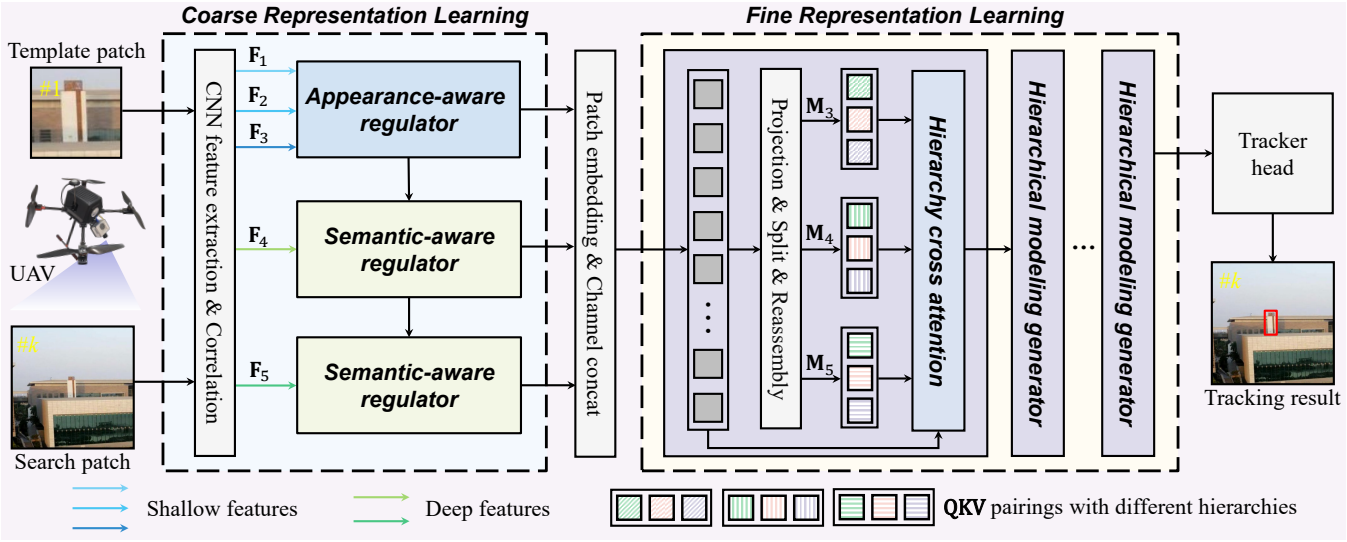

Fig. 2. Illustration of the proposed progressive representation learning framework for UAV tracking. In the coarse representation learning, the appearance-aware regulator and semantic-aware regulator are employed to generate coarse object representations, which highlight different features of the image. In the fine representation learning, the coarse object representations are first patched, then projected, split, and reassembled to obtain $\mathbf{M}_3$, $\mathbf{M}_4$, and $\mathbf{M}_5$ respectively, followed by fusion via hierarchical cross-attention. Best viewed in color (Image frames are from UAV123 [30]).

object representations, obtaining the local spatial information of the object. Building upon this foundation, the fine representation learning generates robust fine object representations for UAV tracking. With the coarse-to-fine progressive perspective, the proposed framework ensures tracking performance in complex dynamic environments, such as occlusion and aspect ratio change.

A. Coarse Representation Learning

For coarse representation learning, the CNN-based backbone is first utilized to extract multi-scale features. The features extracted by the shallow layers of CNNs tend to include a mass of appearance information. Instead, the features extracted by the deep layers of CNNs tend to enrich semantic information. Therefore, the appearance-aware regulator and the semantic-aware regulator are proposed to process shallow features and deep features, respectively.

1) Appearance-aware regulator (AR): The AR is utilized to learn appearance information such as color, edge, and shape from the shallow features.

As depicted in Fig. 3(a), a branch called the Gating controller (GC) serves as a switch, determining the activation of related information. Specifically, the features of the first layer $\mathbf{F}_1$ and the second layer $\mathbf{F}_2$ are the inputs of the GC. Then the convolutional operation (Conv) is employed to achieve cross-channel information integration, where the kernel size in Conv is 1×1. The intermediate results $\mathbf{I}_1$ and $\mathbf{I}_2$ before concatenation (Concat) are generated as follows:

$$\begin{aligned} \mathbf{I}_1 &= \mathrm{Pooling}(\mathrm{Norm}(\mathrm{Conv}(\mathbf{F}_1))) \ , \\ \mathbf{I}_2 &= \mathrm{Conv}(\mathbf{F}_2) \ , \end{aligned} \tag{1}$$

where Norm denotes batch normalization, which helps stabilize and accelerate the training process. Besides, the Pooling operation is employed to ensure dimensional alignment.

Subsequently, a weight map $\boldsymbol{\alpha}_c$ can be obtained after Concat and Conv, which is followed by a rectified linear unit activate function (ReLU):

$$\boldsymbol{\alpha}_c = \mathrm{ReLU}(\mathrm{Conv}(\mathrm{Concat}(\mathbf{I}_1, \mathbf{I}_2))) \ . \tag{2}$$

Finally, the weight map $\boldsymbol{\alpha}_c$ is employed for element-wise multiplication with the features of the third layer $\mathbf{F}_3$, followed by a residual connection. Then the output of AR, *i.e.*, $\mathbf{W}_3$, can be obtained as:

$$\mathbf{W}_3 = \mathrm{CNR}(\mathbf{F}_3 + \boldsymbol{\alpha}_c \cdot \mathbf{F}_3) \ , \tag{3}$$

where the CNR represents the combination operations of

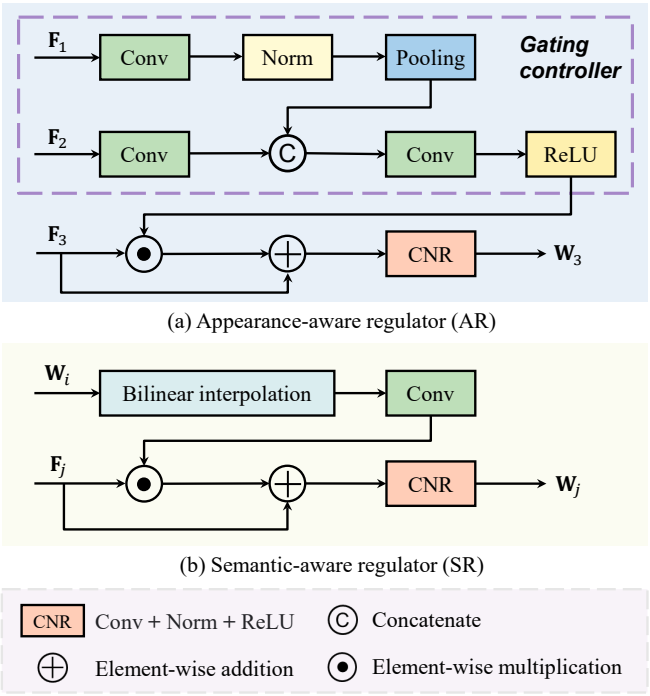

Fig. 3. Structure of the proposed AR (above) and SR (below). The AR is designed to mitigate appearance inference, while the SR is designed to capture semantic information.

Conv, Norm, and ReLU. Additionally, residual connections and activation functions are utilized to speed up network learning and avoid the vanishing gradient problem.

***Remark 1**:* The GC is employed to control the flow of features, thereby improving the quality of the object representations. In the learning process, 1×1 Conv can adaptively retain effective information or filter out redundant information, thus enhancing object representations.

2) Semantic-aware regulator (SR): The SR is designed to learn semantic information from the deep features, *i.e.*, the features from the fourth and fifth layers.

As illustrated in Fig. 3(b), the SR takes the outputs from the previous layer $\mathbf{W}_i$ and the feature of current layer $\mathbf{F}_j$ as inputs. This enables the SR to dynamically integrate contextual information from both shallow and deep features. Then, the outputs of the two SRs used in the coarse representation learning, *i.e.*, $\mathbf{W}_4$ and $\mathbf{W}_5$, can be obtained as:

$$\begin{aligned}\mathbf{W}_4 &= \mathrm{CNR}(\mathbf{F}_4 + \mathbf{F}_4 \cdot \mathrm{Conv}(\mathrm{BLI}(\mathbf{W}_3))) \ , \\ \mathbf{W}_5 &= \mathrm{CNR}(\mathbf{F}_5 + \mathbf{F}_5 \cdot \mathrm{Conv}(\mathrm{BLI}(\mathbf{W}_4))) \ ,\end{aligned} \tag{4}$$

where the BLI denotes bilinear interpolation, ensuring the alignment of feature dimensions. Notably, the first equation corresponds to the SR depicted in the upper part of Fig. 2, focusing on refining the features from the fourth layer $\mathbf{F}_4$. Instead, the second equation corresponds to the SR depicted in the lower part of Fig. 2, which primarily enhances the features from the fifth layer $\mathbf{F}_5$.

***Remark 2**:* The SR is utilized to extract useful information from deep features and transmit them to the fine representation learning. By leveraging appearance information from the AR, the SR significantly improves scene interpretation capability, which is beneficial for UAV tracking.

B. Fine Representation Learning

For fine representation learning, the hierarchical modeling generator (HMG) is designed to fuse the interaction information between coarse object representations. The coarse object representations generated during the previous process are first divided into patches, followed by concatenation along the channel dimension.

As shown in Fig. 4, the token $\mathbf{X}$ aggregated by coarse object representations is decomposed into **QKV** pairings with different hierarchies, namely $\mathbf{M}_3$, $\mathbf{M}_4$, and $\mathbf{M}_5$. Then they are intertwined in the ViT feature space by performing cross-attention after the interaction operation. This strategy enables the model to capture the relationship between coarse object features at different hierarchies, thereby improving the model's representation ability.

Specifically, the process begins by decomposing the input $\mathbf{X}$ into query ($\widehat{\mathbf{Q}}$), key ($\widehat{\mathbf{K}}$), and value ($\widehat{\mathbf{V}}$) vectors via linear projection. For the query vectors ($\widehat{\mathbf{Q}}$), further splitting is conducted at the channel level, yielding $\mathbf{Q}_3$, $\mathbf{Q}_4$, and $\mathbf{Q}_5$. Similar operations are performed for the $\widehat{\mathbf{K}}$ and $\widehat{\mathbf{V}}$, respectively. From level 3 to level 5, the corresponding query, key, and value pairs at each tier are utilized to reassembly **QKV** pairings, which can be represented as follows:

$$\mathbf{M}_i = \mathrm{Concat}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) \ , \quad \text{for } i = 3, 4, 5 \ . \tag{5}$$

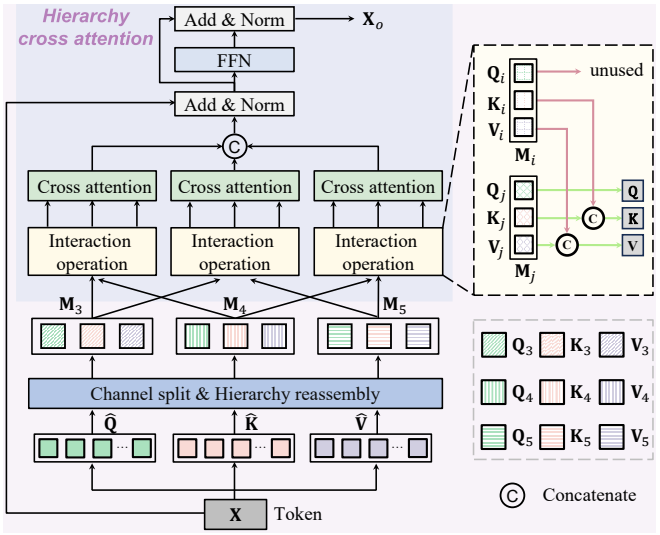

Fig. 4. Detailed workflow of the proposed HMG. With interaction operation and cross-attention, the **QKV** pairings with different hierarchies, *i.e.*, $\mathbf{M}_3$, $\mathbf{M}_4$, and $\mathbf{M}_5$, can communicate with each other. Best viewed in color.

Within the proposed HMG, the hierarchy cross-attention is designed to enhance the interaction between different hierarchy representations. To establish hierarchical connections, interaction operations are performed between $\mathbf{M}_3$ and $\mathbf{M}_4$, as well as between $\mathbf{M}_3$ and $\mathbf{M}_5$, and between $\mathbf{M}_4$ and $\mathbf{M}_5$. During the interaction operation between $\mathbf{M}_i$ and $\mathbf{M}_j$, the keys $\mathbf{K}_i$ from $\mathbf{M}_i$ and $\mathbf{K}_j$ from $\mathbf{M}_j$ are concatenated, as well as the values $\mathbf{V}_i$ and $\mathbf{V}_j$, which can be expressed as:

$$\begin{aligned}\mathbf{K}_{ij} &= \mathrm{Concat}(\mathbf{K}_i, \mathbf{K}_j) \ , \\ \mathbf{V}_{ij} &= \mathrm{Concat}(\mathbf{V}_i, \mathbf{V}_j) \ ,\end{aligned} \tag{6}$$

where $i < j$, $\mathbf{K}_{ij}$ denotes the concatenated key from $\mathbf{M}_i$ and $\mathbf{M}_j$, while $\mathbf{V}_{ij}$ represents the concatenated value.

Then, cross-attention mechanisms are utilized to integrate information, which can be represented as follows:

$$\begin{aligned}\mathbf{H}_{att}^{34} &= \mathrm{Softmax}\left(\frac{\mathbf{Q}_4 \cdot [\mathbf{K}_3, \mathbf{K}_4]^T}{\sqrt{d}}\right) \cdot [\mathbf{V}_3, \mathbf{V}_4] \ , \\ \mathbf{H}_{att}^{35} &= \mathrm{Softmax}\left(\frac{\mathbf{Q}_5 \cdot [\mathbf{K}_3, \mathbf{K}_5]^T}{\sqrt{d}}\right) \cdot [\mathbf{V}_3, \mathbf{V}_5] \ , \\ \mathbf{H}_{att}^{45} &= \mathrm{Softmax}\left(\frac{\mathbf{Q}_5 \cdot [\mathbf{K}_4, \mathbf{K}_5]^T}{\sqrt{d}}\right) \cdot [\mathbf{V}_4, \mathbf{V}_5] \ ,\end{aligned} \tag{7}$$

where d represents the dimension of the concatenated key. Besides, $\mathbf{H}_{att}^{34}$, $\mathbf{H}_{att}^{35}$, and $\mathbf{H}_{att}^{45}$ are the attention maps of hierarchical representations, respectively.

***Remark 3**:* The fine representation learning accepts purified coarse object representations and focuses on information fusion across various hierarchical representations. Excluding low-level queries in cross-attention streamlines the integration of relevant information across different levels of representation, thereby reducing computational costs.

Subsequently, $\mathbf{H}_{att}^{34}$, $\mathbf{H}_{att}^{35}$, and $\mathbf{H}_{att}^{45}$ are concatenated along the channel, followed by residual connection to the input $\mathbf{X}$, which can be expressed as:

$$\mathbf{W}_c = \mathrm{Norm}(\mathrm{Concat}(\mathbf{H}_{att}^{34}, \mathbf{H}_{att}^{35}, \mathbf{H}_{att}^{45}) + \mathbf{X}) \ . \tag{8}$$

Finally, the processed $\mathbf{W}_c$ further undergoes adjustments through a feed-forward network (FFN) and Norm. Thereby, the output of the HMG, denoted as $\mathbf{X}_o$, can be expressed as:

$$\mathbf{X}_o = \mathrm{Norm}(\mathrm{FFN}(\mathbf{W}_c) + \mathbf{W}_c) \ . \tag{9}$$

Remark 4*:* The strategic integration of cross-attention mechanisms facilitates precise interaction and effective fusion of diverse hierarchical features. Moreover, by iteratively fusing coarse object representations, the proposed HMG gradually captures both local and global information for improving performance in complex dynamic environments.

IV. Experiments

A. Implementation Details

The proposed PRL-Track is trained using Python 3.8 and PyTorch 1.13.1 on 2 NVIDIA A100 GPUs for 70 epochs. The backbone of PRL-Track is initialized using AlexNet [8], which has been pre-trained on ImageNet [31]. The learning rate initiates at 5×10^{-4}, rises to 10^{-2}, and subsequently decreases to 10^{-4} in log space. Additionally, the template patch is limited to dimensions of $127 \times 127 \times 3$, while the search patch is constrained to $287 \times 287 \times 3$. The training dataset are COCO [32], GOT-10K [33], and LaSOT [34].

B. Evaluation Metrics

The one-pass evaluation (OPE) metrics [30] are essential for assessing tracking performance, including precision and success rate. Specifically, the precision is measured by the Euclidean distance between the center of the predicted box and the ground truth, which is denoted as the center location error (CLE). The precision plot is drawn by counting the percentage of frames within a certain threshold of CLE. In the general evaluation, the threshold for tracker ranking is set to 20 pixels. The success rate is computed through the intersection over union (IoU) of the ground truth with the predicted box. The success plot is drawn by counting the percentage of frames whose IoU exceeds a predetermined threshold. Meanwhile, the area under the curve (AUC) is computed to rank trackers.

C. Overall Performance

In this section, PRL-Track is tested on three challenging and authoritative UAV tracking benchmarks with other 14 existing SOTA trackers including LPAT [36], SGDViT [24], HiFT [12], SiamAPN++ [4], SiamFC [18], DeepSTRCF [38], Ocean [39], DaSiamRPN [40], SE-SiamFC [35], MCCT [21], AutoTrack [20], TADT [41], UDT+ [6], and UDT [6]. Notably, all Siamese trackers use the same lightweight backbone, *i.e.*, AlexNet [8], for a fair comparison.

1) ***UAVTrack112****:* UAVTrack112 [37] is specifically constructed for UAV tracking, encompassing 112 sequences that introduce challenges for real-world evaluations. It encompasses common challenges [30] encountered in UAV tracking, including aspect ratio change, similar objects, partial occlusion, and so on. The results shown in Fig. 5 demonstrate the remarkable performance of PRL-Track, attaining precision (**0.786**) and success rates (**0.602**).

2) ***UAVTrack112_L****:* UAVTrack112_L [37] consists of 45 long-term tracking sequences and includes over 60K frames in total. Fig. 5 demonstrates that PRL-Track yields the best performance compared with other SOTA trackers. In the precision, PRL-Track leads the pack with a remarkable score of **0.803**, surpassing LPAT (0.760) and SGDViT (0.743), which trail behind in second and third place, respectively. Similarly, PRL-Track achieves the top success rate of **0.597**, outperforming LPAT (0.566) and SGDViT (0.554).

Remark 5*:* In this work, UAVTrack112_L is utilized to validate the long-term tracking performance of the proposed PRL-Track. The experimental results indicate that PRL-Track performs exceptionally well on long sequences, providing a more stable and sustained tracking capability.

3) ***UAV123****:* UAV123 [30] consists of 123 challenging sequences with a combined total of over 112K frames. These sequences involve demanding aerial scenarios, encompassing occlusion, illumination variation, and low-resolution challenges. Performance evaluation on UAV123 offers valuable insights into the advancement of aerial visual tracking. As shown in Fig. 5, PRL-Track stands out from other trackers with a success rate (**0.791**) and precision (**0.593**).

D. Attribute-Based Comparison

The robustness of PRL-Track in handling complex UAV tracking challenges is evaluated through attribute-based comparisons. Specifically, the attributes of aspect ratio change (ARC), partial occlusion (POC), scale variation (SV), and viewpoint change (VC) are considered during the evaluation process. As illustrated in TABLE I, PRL-Track performs the best in all four attributes compared with the other 5 SOTA trackers. Notably, PRL-Track achieves superior performance in the ARC, surpassing the second-best performance by **4.5%** in precision, and achieving **4.1%** increase in success rate. This substantial improvement demonstrates that the proposed PRL-Track can exploit the global connection of multi-scale features, thereby better adapting to scenarios where the scale of the tracking object changes. Additionally, when confronted with partial occlusion, the ViT-based HMG utilizes purified object representations for global modeling, mitigating the impact of object feature degradation caused by occlusion. Moreover, when encountering scale variation, the progressive process of coarse-to-fine exploration can generate more discriminative object representations to keep reliable tracking.

Remark 6*:* The promising results demonstrate that the proposed PRL-Track can learn robust object representations to tackle the challenging scenarios mentioned above. Moreover, these robust object representations contribute to the effectiveness of long-term tracking.

E. Ablation Study

To demonstrate the effectiveness of each representation learning within PRL-Track, detailed studies conducted on UAVTrack112_L are presented in this section. To ensure fairness, each variant of the tracker is configured with the same settings (including training strategy and parameter configurations) except for the studied module.

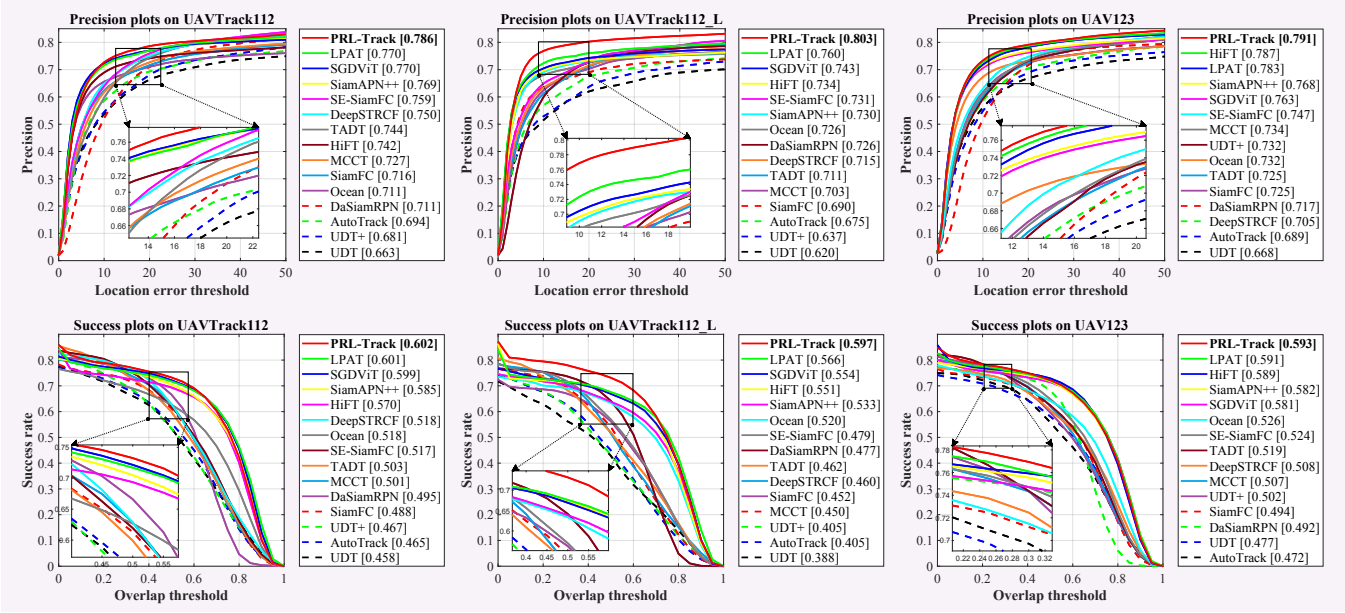

Fig. 5. Overall performance of PRL-Track and SOTA trackers on UAVTrack112 [37], UAVTrack112_L [37], and UAV123 [30]. The experimental results showcase the superior performance of the proposed PRL-Track on all benchmarks.

1) Clarification of symbol: First, the symbols used in TABLE II are explained. This work considers the model with only feature extraction and regression & classification network as Baseline. FLP represents the fine representation learning. AR and SR represent different components used in the coarse representation learning. PRL-Track denotes the full version of the proposed progressive representation learning framework.

2) Result analysis: As presented in the TABLE II, integrating FLP directly into the Baseline significantly improved its performance, improving precision by about 10.09% and success rate by 13.16%. This is attributed to the hierarchy modeling generator, which facilitates the integration of features across various scales. However, combining the SR and FLP can lead to performance degradation due to appearance interference from shallow features. On the other hand, combining the AR and FLP enhances tracking precision by 13.11%. Furthermore, adopting the Baseline+AR+SR+FLP configuration yields the best performance, showcasing improvement in precision by **15.71%** and in success rate by **17.29%** compared to the Baseline. All the aforementioned results verify the efficiencies of the coarse representation learning (AR+SR) and FLP in improving object representation exploration for UAV tracking.

F. Qualitative Evaluation

As shown in Fig. 6, the visualization comparison results between PRL-Track and the other 4 SOTA trackers demonstrate the robustness of PRL-Track in complex dynamic environments. When encountering similar objects during the tracking process, the two learning processes within PRL-Track produce discriminative object representations, enabling stable and reliable tracking. In contrast, SE-SiamFC [35] is disrupted by similar objects, leading to tracking failure. Furthermore, as observed from the second row of Fig. 6, only PRL-Track completes the re-detection task and achieves tracking restoration after a brief out-of-view period. Finally, in the common scenario of occlusion encountered in UAV tracking, PRL-Track also exhibits superior performance. Owing to the robust fine object representations, the proposed PRL-Track achieves reliable tracking performance.

TABLE I

COMPARATIVE EVALUATION OF 6 SOTA TRACKERS ON UAVTRACK112_L BASED ON ATTRIBUTES. THE BEST TWO PERFORMANCES ARE HIGHLIGHTED IN RED AND GREEN, RESPECTIVELY.

Attributes / Trackers	Aspect Ratio Change		Partial Occlusion		Scale Variation		Viewpoint Change	
	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
SE-SiamFC [35]	0.699	0.442	0.770	0.480	0.718	0.465	0.442	0.673
SiamAPN++ [4]	0.700	0.511	0.725	0.517	0.718	0.522	0.495	0.681
HiFT [12]	0.712	0.528	0.760	0.557	0.721	0.541	0.491	0.657
SGDViT [24]	0.719	0.536	0.762	0.560	0.731	0.543	0.514	0.695
LPAT [36]	0.735	0.541	0.802	0.589	0.749	0.557	0.502	0.690
PRL-Track (Ours)	**0.780**	**0.582**	**0.819**	**0.607**	**0.795**	**0.591**	**0.542**	**0.738**

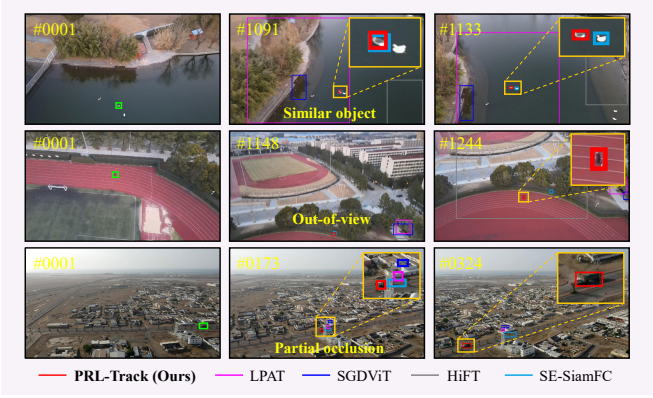

Fig. 6. Qualitative comparison of the proposed PRL-Track with other 4 SOTA trackers on three challenging UAV tracking sequences: duck1_2 and jogging2 from UAVTrack112 [37], and truck2 from UAV123 [30]. The green box in the first frame of each sequence represents the tracking object.

V. Real-World Tests

To demonstrate the real-world applicability of PRL-Track, extensive testing is conducted on a typical UAV platform, as shown in Fig. 7. Specifically, the UAV platform is equipped with an NVIDIA Jetson Orin NX 16GB-based edge smart camera. During the testing phase, the edge smart camera exhibits the following average utilization rates: RAM usage is at 32.67%, while GPU and CPU record average utilizations of 28.81% and 14.15%, respectively. The experimental results from several of these tests are shown in Fig. 7. These sequences present a variety of challenges, including fast motion, partial occlusion, and illumination variation.

In Test 1, the tracked object engages in a basketball game on the court, characterized by rapid and frequent movements. Additionally, due to shooting actions, bodily deformation occurs intermittently. Nonetheless, the PRL-Track consistently maintains a high level of tracking precision in such dynamic scenarios. The Test 2 and Test 3 sequences focus on tracking cars during steady flights, including scenarios with partial occlusion and illumination variation. When encountering occlusion, minor fluctuations are observed in the tracking results but quickly restore stability. Furthermore, the Test 3 sequence highlights the performance of PRL-Track over extended durations, showcasing its robustness in long-term tracking scenarios. Finally, the proposed PRL-Track remains a speed exceeding **42.6** frames per second, demonstrating its superior tracking speed. The experiment results in real-world tests underscore the ability of PRL-Track to learn object representations and achieve stable tracking.

TABLE II

Ablation Study of the Proposed Framework on UAVTrack112_L. Δ Shows Improvement over Baseline.

Trackers	Prec.	$\Delta_{Prec.}$ (%)	Succ.	$\Delta_{Succ.}$ (%)
Baseline	0.694	-	0.509	-
Baseline+FLP	0.764	+10.09	0.576	+13.16
Baseline+SR+FLP	0.777	+11.96	0.567	+11.39
Baseline+AR+FLP	0.785	+13.11	0.577	+13.36
PRL-Track	**0.803**	**+15.71**	**0.597**	**+17.29**

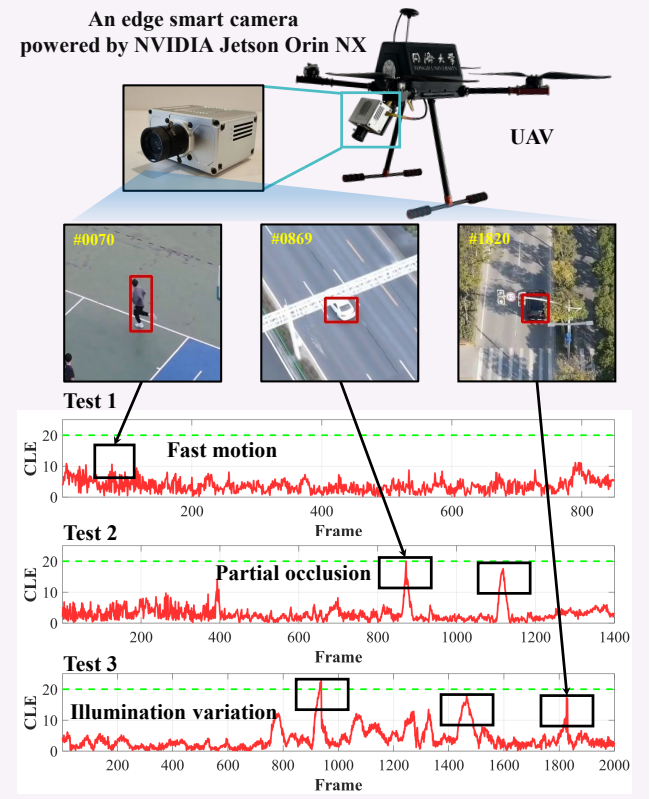

Fig. 7. Visualization of real-world tests: the red bounding boxes denote tracking results. The center location error (CLE) score below 20 is deemed reliable in the real-world test.

VI. Conclusions

In this work, a novel progressive representation learning framework, *i.e.*, PRL-Track, is proposed to extract robust object representations for UAV tracking. In the proposed PRL-Track, two CNN-based regulators are utilized to create coarse object representations. Furthermore, the ViT-based hierarchical modeling generator is adopted to exploit coarse object representations. This progressive learning process empowers the tracker, *i.e.*, PRL-Track, to generate robust object representations, thereby better addressing the challenges in complex UAV scenarios. Extensive experiments, including challenging real-world tests, demonstrate that PRL-Track has achieved outstanding performance. We are convinced that our framework can promote further research in UAV tracking and foster related practical applications.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 62173249) and the Natural Science Foundation of Shanghai (No. 20ZR1460100).

References

[1] J. He, Z. Sun, N. Cao, D. Ming, and C. Cai, "Target Attribute Perception Based UAV Real-Time Task Planning in Dynamic Environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 888–895.

[2] J. Pak, B. Kim, C. Ju, S. H. You, and H. I. Son, "UAV-Based Trilateration System for Localization and Tracking of Radio-Tagged Flying Insects: Development and Field Evaluation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 4981–4988.

[3] D. R. McArthur, Z. An, and D. J. Cappelleri, "Pose-Estimate-Based Target Tracking for Human-Guided Remote Sensor Mounting with A UAV," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 10 636–10 642.

[4] Z. Cao, C. Fu, J. Ye, B. Li, and Y. Li, "SiamAPN++: Siamese Attentional Aggregation Network for Real-Time UAV Tracking," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 3086–3092.

[5] C. Fu, K. Lu, G. Zheng, J. Ye, Z. Cao, B. Li, and G. Lu, "Siamese Object Tracking for Unmanned Aerial Vehicle: A Review and Comprehensive Analysis," *Artificial Intelligence Review*, vol. 56, no. Suppl 1, pp. 1417–1477, 2023.

[6] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised Deep Tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1308–1317.

[7] C. Fu, L. Yao, H. Zuo, G. Zheng, , and J. Pan, "SAM-DA: UAV Tracks Anything at Night with SAM-Powered Domain Adaptation," in *Proceedings of the IEEE International Conference on Advanced Robotics and Mechatronics (ICARM)*, 2024, pp. 1–8.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, vol. 25, 2012, pp. 1097–1105.

[9] 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 (CVPR)*, 2016, pp. 770–778.

[10] J. Fang, H. Lin, X. Chen, and K. Zeng, "A Hybrid Network of CNN and Transformer for Lightweight Image Super-Resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022, pp. 1102–1111.

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020, pp. 1–22.

[12] Z. Cao, C. Fu, J. Ye, B. Li, and Y. Li, "HiFT: Hierarchical Feature Transformer for Aerial Tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 15 457–15 466.

[13] Z. Peng, Z. Guo, W. Huang, Y. Wang, L. Xie, J. Jiao, Q. Tian, and Q. Ye, "Conformer: Local Features Coupling Global Representations for Recognition and Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 9454–9468, 2023.

[14] L. Zhang, Y. Dong, and Y. Wu, "Multi-Layer CNN Features Aggregation for Real-Time Visual Tracking," in *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2018, pp. 2404–2409.

[15] M. Zhao, S. Zhong, X. Fu, B. Tang, and M. Pecht, "Deep Residual Shrinkage Networks for Fault Diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4681–4690, 2020.

[16] K. Li, Y. Wang, J. Zhang, P. Gao, G. Song, Y. Liu, H. Li, and Y. Qiao, "UniFormer: Unifying Convolution and Self-Attention for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12 581–12 600, 2023.

[17] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do Vision Transformers See Like Convolutional Neural Networks?" in *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, vol. 34, 2021, pp. 12 116–12 128.

[18] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-Convolutional Siamese Networks for Object Tracking," in *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, 2016, pp. 850–865.

[19] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning Dynamic Siamese Network for Visual Object Tracking," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1763–1771.

[20] Y. Li, C. Fu, F. Ding, Z. Huang, and G. Lu, "AutoTrack: Towards High-Performance Visual Tracking for UAV with Automatic Spatio-Temporal Regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 923–11 932.

[21] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-Cue Correlation Filters for Robust Visual Tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4844–4853.

[22] 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 (CVPR)*, 2018, pp. 8971–8980.

[23] 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 (CVPR)*, 2021, pp. 8126–8135.

[24] L. Yao, C. Fu, S. Li, G. Zheng, and J. Ye, "SGDViT: Saliency-Guided Dynamic Vision Transformer for UAV Tracking," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 3353–3359.

[25] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[26] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object Detection in 20 Years: A Survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[27] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep High-Resolution Representation Learning for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 10, pp. 3349–3364, 2020.

[28] C. Li, J. Yang, P. Zhang, M. Gao, B. Xiao, X. Dai, L. Yuan, and J. Gao, "Efficient Self-Supervised Vision Transformers for Representation Learning," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022, pp. 1–27.

[29] Z. Cai, S. Ghosh, K. Stefanov, A. Dhall, J. Cai, H. Rezatofighi, R. Haffari, and M. Hayat, "MARLIN: Masked Autoencoder for Facial Video Representation Learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 1493–1504.

[30] M. Mueller, N. Smith, and B. Ghanem, "A Benchmark and Simulator for UAV Tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 445–461.

[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, 2015.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014, pp. 740–755.

[33] L. Huang, X. Zhao, and K. Huang, "GOT-10K: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1562–1577, 2019.

[34] 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 (CVPR)*, 2019, pp. 5374–5383.

[35] I. Sosnovik, A. Moskalev, and A. W. Smeulders, "Scale Equivariance Improves Siamese Tracking," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 2765–2774.

[36] C. Fu, W. Peng, S. Li, J. Ye, and Z. Cao, "Local Perception-Aware Transformer for Aerial Tracking," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 12 122–12 129.

[37] C. Fu, Z. Cao, Y. Li, J. Ye, and C. Feng, "Onboard Real-Time Aerial Tracking with Efficient Siamese Anchor Proposal Network," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–13, 2021.

[38] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4904–4913.

[39] Z. Zhang, H. Peng, J. Fu, B. Li, and W. Hu, "Ocean: Object-Aware Anchor-Free Tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 771–787.

[40] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-Aware Siamese Networks for Visual Object Tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 101–117.

[41] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-Aware Deep Tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1369–1378.