An Occlusion-Aware Tracker With Local-Global Features Modeling in UAV Videos

Qiuyu Jin, Yuqi Han, Wenzheng Wang, Linbo Tang, Jianan Li, and Chenwei Deng

***Abstract*—Recently, tracking with unmanned aerial vehicle (UAVs) platforms has played significant roles in Earth observation tasks. However, target occlusion remains a challenging factor during the continuous tracking procedure. In particular, incomplete local appearance features can mislead the tracking network to produce inaccurate size and position estimations when the target is occluded. Furthermore, the tracking network lacks sufficient occlusion supervision information, which may lead to template degradation during template updating. To address these challenges, in this article, we design an occlusion-aware tracker with local-global features modeling, which contains two key components, namely the feature intrinsic association module (FIAM) and the feature verification module (FVM). Specifically, the FIAM divides the local features into blocks and utilizes the transformer network to explore the relative relationships among each subblock, which supplements the damaged local target features and assists the modeling for global target features. In addition, the FVM establishes a correlation measurement network between the target and the template. To precisely evaluate the occlusion status, masked samples with occlusion exceeding 50% are selected as negative samples for independent training, which ensures the purity of the target template. Qualitative and quantitative experiments are conducted on publicly available datasets, including UAV20 L, UAV123, and LaSOT. Qualitative and quantitative experiments have demonstrated the effectiveness of the proposed tracking algorithm over the other state-of-the-art trackers in occlusion scenarios.**

***Index Terms*—Local-global feature modeling, object tracking, occlusion awareness, UAV.**

I. Introduction

The rapid advancement of Earth Observation technology has progressively posed a challenge in effectively tracking objects of interest within observation images. Visual target tracking is extensively utilized in aerial imagery [1], including military reconnaissance [2], [3], [4] and aerial photography [5], [6], [7], [8] to ensure the continuous maintenance of the target at the center of the field of view. The objective of object tracking is to accurately predict the position of subsequent video sequences for a specified region by considering the initial frame's object as the designated target. Visual target tracking algorithms can be broadly categorized into correlation filtering-based algorithms [9], [10], [11], [12], [13], [14], [15], [16] and Siamese network-based algorithms [17], [18], [19], each belonging to different schools of algorithm design. In the correlation filtering method, a filter template is manually constructed using handcrafted features and employed to perform correlation operations with candidate regions. The target position is determined based on the maximum output response. However, due to the utilization of complex optimization strategies and reliance on handcrafted features, trackers based on correlation filtering face challenges in improving robustness in dynamic and complex environments. On the other hand, Siamese network trackers based on contrastive learning achieve a favorable balance between accuracy and efficiency, attracting attention from researchers as they lead an emerging trend in the field of visual tracking [20].

While the majority of target tracking networks primarily focus on short-term tracking, it is imperative to address the challenges associated with long-term target tracking to accurately detect and locate targets. Long-term target tracking encounters greater variations in targets and more frequent occlusions compared to short-term tracking [21]. Existing methods effectively tackle the issue of target changes by incorporating spatio-temporal context information [22], [23]; however, they face challenges such as tracking drift or losing the target when dealing with frequent occlusions. Siamese network trackers utilize contrastive learning to establish the correlation between the target and template. Robust feature modeling and precise matching information are crucial for accurate tracking. We have observed that occlusion significantly affects both feature modeling and template updating in the network. In **feature modeling**, when partial occlusion occurs, the network exclusively models nonoccluded local appearance information, potentially leading to tracking drift toward a specific subpart of the target. In **template updating**, deciding whether to update the template usually relies on assessing the correlation between current target features and historical templates. When a large area of occlusion affects the target, discriminative local features and template responses may still exist; however, updating using such incomplete samples containing substantial background can introduce noise-induced degradation in templates.

This article proposes a long-term tracking network with anti-occlusion to address the challenges of feature loss and template degradation in scenarios where targets are occluded. In **feature**

Manuscript received 27 October 2023; revised 27 January 2024; accepted 13 February 2024. Date of publication 21 February 2024; date of current version 4 March 2024. This work was supported by National Natural Science Foundation of China (NSFC) under Grant 62171040. *(Corresponding author: Yuqi Han.)*

Qiuyu Jin, Yuqi Han, Wenzheng Wang, and Chenwei Deng are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: yuqi_han@bit.edu.cn).

Linbo Tang is with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China, and also with the Advanced Technology Research Institute, Beijing Institute of Technology, Jinan 100048, China.

Jianan Li is with the Key Laboratory of Photoelectronic Imaging Technology and System, Ministry of Education of China, Beijing Institute of Technology, Beijing 10081, China.

Digital Object Identifier 10.1109/JSTARS.2024.3368035

modeling stage, it is crucial for the tracking network to carefully consider potential associations among appearance features, which act as valuable clues for filling in missing parts using unoccluded features. To accomplish this goal, we divide the convolutional neural network-based appearance features into multiple subblocks and use a transformer network to capture inherent associations within each subblock. Positive masked samples are used for training purposes to enhance the network's ability to learn and complete these features. In addition, by leveraging the transformer's ability to capture long-range dependencies, we establish feature associations across different templates over an extended temporal scale, enabling the network to comprehensively grasp the temporal dynamics of target appearance changes. In **template updating** stage, the discriminator needs to consider not only the correlation between the target and the template but also whether the current state of the target is sufficiently complete compared to the template. To address this, we introduce an additional feature verification module (FVM), which is a network designed to calculate both target features and template features' inner product. The training objective of this module differs from that of the tracking network; it utilizes highly masked samples as negative examples to encourage accurate judgment regarding occlusion extent based on feature loss. The learning objectives of both tracker and verification are complementary, aiming to comprehensively evaluate tracking confidence and occlusion degree while discarding potential noise pollution in participating in template updating. The experimental results on the UAV20 L, UAV123, and LaSOT datasets demonstrate superior overall tracking performance. Furthermore, the proposed method has a total computational complexity of less than 6G Flops, which satisfies real-time requirements with limited computing resources.

The main contributions of this article are as follows.

1) An intrinsic association module is advocated, which utilizes the attention mechanism to extract relationships between deep feature subblocks, ensuring to supplement the occluded global information.
2) An FVM is proposed to assess the occlusion status by evaluating the similarity between the potential target feature and the historical template feature.
3) The experimental results on the UAV123 [48], UAV20L [48], and LaSOT [51] datasets demonstrate that the proposed modules significantly improve tracking performance in occlusion scenarios.

The rest of this article is organized as follows. Section II gives a brief review of the related works. The implementation details for the proposed tracker are illustrated in Section III. Subsequently, experimental validation, including the qualitative and quantitative ones, is shown in Section IV. Finally, Section V concludes this article.

II. Related Works

A. Aerial Video Object Tracking

Target tracking algorithms have been extensively employed in aerial video in RS imagery, including generative tracking algorithms such as Camshift and optical flow methods, as well as discriminative tracking algorithms. Discriminative tracking algorithms compare the difference between background information and target models to extract target information. Notably, representative methods include correlation filtering-based target tracking and deep learning-based target tracking.

For example, Ye et al. [24] utilized multiple regularized correlation filters to effectively mitigate response variations and achieve adaptive channel weight distribution, thereby enhancing the adaptability of target appearance changes over extended time scales. SAT [12] learns the features of both the target and its surrounding patches simultaneously, guiding the filter toward more reliable regions suitable for tracking. Subsequently, some of the researchers [13], [25] introduced several spatio-temporal context-aware tracking algorithms based on discriminative correlation filter (DCF) that enable accurate discrimination between objects and backgrounds in long-term aerial videos by learning spatio-temporal context weights. Furthermore, Yu et al. [26] proposed a Siamese network that employs conditional generative adversarial networks (GAN) to estimate global motion information in UAV videos and generate accurate motion predictions. The fast DCF tracker and the precise deep learning method are integrated to mitigate cumulative drift during tracking [16], enabling a reliable update of the target template with high confidence. Fu et al. [27] combining diverse semantic information, enhanced the flow of information and significantly contributed to robust aerial tracking.

In addition, Cui et al. [28] utilized spatio-temporal background, object appearance models, and motion vectors to provide occlusion information for driving reinforcement learning actions under complete occlusion, thereby improving tracking accuracy while maintaining speed. Moreover, Feng et al. [29] proposed an improved siamRPN++ method based on clustering and frame difference techniques that refined the response map by reducing background noise and preserving vehicle motion information, thus enhancing fuzzy vehicle tracking performance in optical remote sensing videos (ORSV) scenarios. Despite achieving high performance levels as demonstrated by the aforementioned discriminative trackers, there still exist significant challenges related to feature modeling and template updating when dealing with more frequent occlusion scenes encountered during long-term remote sensing video object tracking tasks.

B. Feature Modeling

Feature modeling plays a pivotal role in the Siamese network tracker as it determines the robustness of both template and target features. The early Siamese network trackers [30] predominantly rely on convolutional neural networks based on AlexNet [31]. Subsequent advancements, such as siamRPN++ [32] and SiamDW [33], explore deeper and wider backbone network structures for feature extraction, incorporating ResNet, Mobilenet, and other networks [34], [35], [36], [37] to extract more comprehensive target appearance features. Inspired by NAS research ideas, Lighttrack [38] encodes the search space of backbone networks into a network set and selects paths within this set guided by an evolutionary controller to discover high-efficiency networks that represent the desired features. On the other hand, SiamGAT [39] employs a complete bipartite graph to establish part-part correspondence between targets and search areas while utilizing graph attention mechanisms to propagate target information from template features to search features. This approach enhances local feature perception. However, these tracking networks primarily emphasize local feature information while neglecting global information during the feature modeling stage.

Furthermore, certain tracking algorithms [40], [41] attempt to leverage the inherent structure of pure Transformer networks [42], [43] for capturing global information. However, these approaches suffer from inherent limitations. First, these trackers adopt the ViT framework [44], [45], which preprocesses input images into a series of flat slices, resulting in nonoverlapping segments that undermine local neighborhood relationships and potentially discard discriminative information through segmentation. Next, while ViT-based features possess extensive global context due to their long modeling capability, they cannot effectively capture fine-grained details necessary for tracking small targets from aerial perspectives.

Therefore, we initially employed a manually designed CNN backbone network with a large receptive field to depict the appearance characteristics of the target. Building upon this, we further partition the local appearance into multiple subblocks and leverage the transformer network to explore potential internal correlations among each feature block. During training, we utilize numerous masked positive samples to facilitate the transformer network in acquiring the ability to infer missing information about the target, thereby effectively enhancing the robustness of target features in occlusion scenarios.

C. Template Updating

Template updating is a crucial approach in tracking networks to address the challenge of continuous target and environmental changes during long-term tracking. However, template updating can be a double-edged sword in complex tracking tasks. While offline tracking networks without template updating struggle to adapt to significant target variations, they also avoid introducing noise information from occlusions into the initial template. Some tracking networks employ template updating that considers interframe correlation information during the tracking process and utilizes dynamic templates to capture the diversity and temporal evolution of the target state, which proves advantageous for long-term tracking. Nevertheless, when occlusion occurs, damaged dynamic templates may lead to reduced matching success rates within the tracking network.

To address the issue of timing decisions for template updating, methods such as RTMDNet [46] utilize classification confidence to determine the optimal timing for template updating, while ATOM [47] employs response graphs of targets and templates to decide whether or not to reject updates. LTMU [22] assesses target states by synthesizing geometry, appearance, and other relevant information. However, these approaches may face challenges when dealing with occluded targets since response scores based on tracker output do not necessarily decrease immediately under extensive occlusion. This is because the training objective of the tracker aims to enhance the robustness of features, ensuring a high response even in the presence of damaged features. However, this training goal may pose a potential risk of template contamination when occlusion occurs.

To assess the extent of template feature damage in cases of occlusion, we have independently developed an FVM external to the tracking network. The role of this verifier is to minimize the correlation between the target and template when occlusion occurs, enabling us to determine whether there is significant occlusion affecting the target area. This approach helps prevent additional background information from being incorporated into the template during occlusion, thereby enhancing the reliability of our template.

III. Proposed Method

We propose an occlusion-aware tracking network that incorporates local and global features modeling. The overall architecture of the tracker is illustrated in Fig. 1. The network follows the Siamese network paradigm for tracking and introduces a feature intrinsic association module (FIAM) after the convolutional neural network. In addition, an FVM is incorporated into the state estimation header.

The network employs a three-head input network configuration, in which a lightweight CNN is utilized to extract depth features of a search region S_{Region}, an initial template T_{init}, and a dynamic template T_{dynamic}. FIAM initially conducts vectorization on the deep features, then models the spatial correlation of each local feature vector using cross-attention (CAttn) and self-attention (SAttn). During the training phase, a significant number of masked samples are used to facilitate learning about cosine correlation between local feature vectors in the attention module of the transformer network and assigning weights to individual feature vectors. This allows for the effective enhancement of global appearance features in occluded scenarios by leveraging unoccluded appearance features, thus improving overall robustness. FVM introduces a verifier to address the normalized cosine correlation between the current and historical template features during state estimation of the tracking network. Specifically, this verifier is trained separately using masked samples with occlusion greater than 50% as negative examples, enabling it to assess whether the target feature is complete based on template features that possess complete information. This approach ensures a robust and consistent historical template pool mitigating noise pollution caused by severely damaged target features.

Moreover, we avoid using the encoder–decoder Transformer network architecture in the design process of FIAM and FVM to alleviate the excessive computational burden on the tracker. The tracking network maintains a computational complexity of 5.3G FLOPs throughout, with a parameter size of 15 M, thereby meeting the deployment requirements for limited computing power.

A. Feature Intrinsic Association Module

We observe that the convolutional neural network effectively captures the local appearance information of the target through

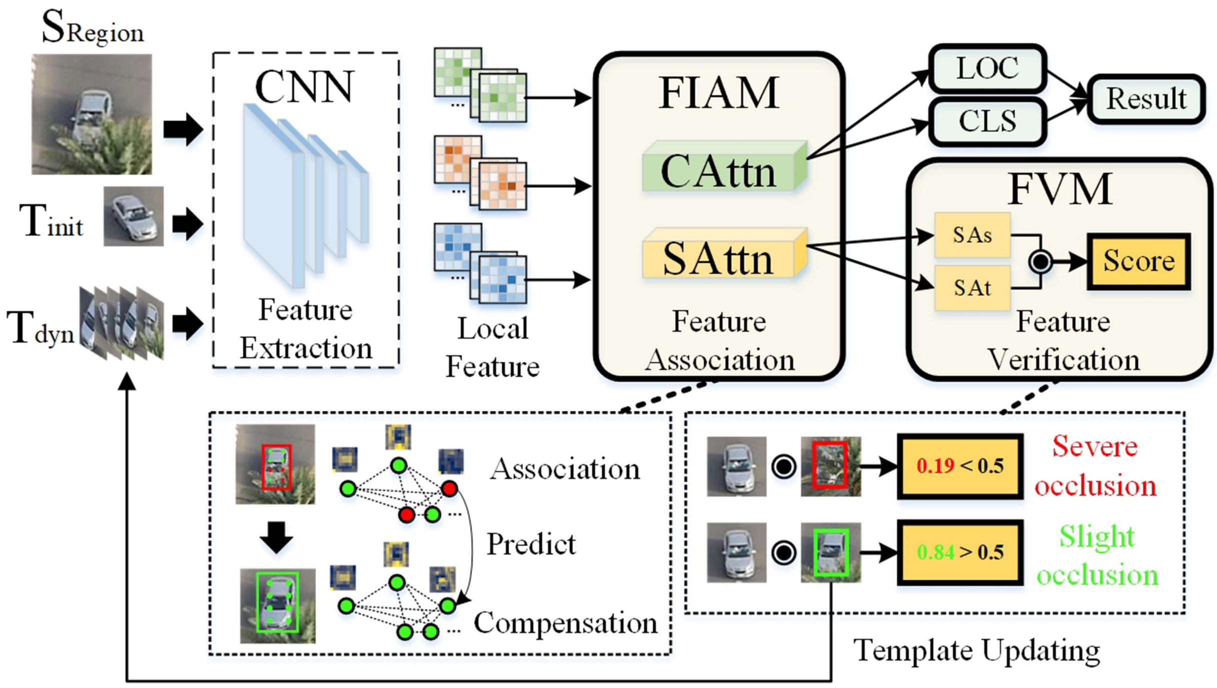


Fig. 1. Main framework of our tracking network. The CNN represents a lightweight feature extraction module. FIAM denotes a feature intrinsic association module, which is utilized to compensate for damaged appearance features and restore global features. The FVM is an additional module used to assess the extent of feature corruption and update the template when the score exceeds a predefined threshold.

a sliding window but fails to exploit the intrinsic association between deep feature subblocks represented by each local observation window. Consequently, during subsequent feature matching, the tracking network tends to prioritize local features with high weights rather than global features. While classification and regression can generally recover complete position and size information of the target in tracking processes, occlusion may cause drift in tracking results toward a specific subpart of the target.

To address this issue, we propose the construction of a FIAM, which aims to capture the inherent association relationship among deep feature subblocks. First, we employ vectorization techniques to represent the deep features and utilize an attention mechanism to model their mutual relationships and weights in space. To better simulate target occlusion scenarios, we introduce a large number of positive sample pairs randomly covered with masks during the training phase of our tracking network. The training methodology serves two primary objectives: 1) it effectively mitigates overfitting by avoiding excessive focus on specific features; 2) it actively promotes the network to exploit internal associations between feature subblocks to compensate for missing feature information and enhance the robustness of global features.

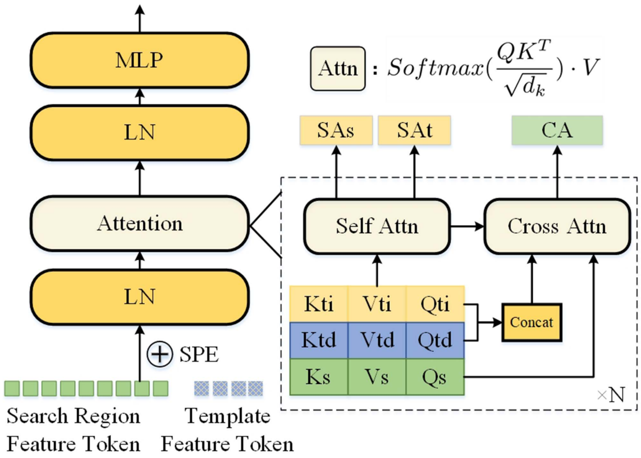


Fig. 2. Structure diagram of FIAM. After applying layer normalization (LN) and sinusoidal position encoding (SPE), the corresponding Q, K, and V matrices of the feature token are obtained through embedding. These matrices are utilized to represent the adaptive relationships between features in the core operation module-attention network. Furthermore, the attention network consists of a self-attention (SAttn) module and a cross-attention (CAttn) module. The final result of relation mapping is outputted by a multilayer perceptron (MLP).

The structure and process of FIAM are illustrated in Fig. 2. The multiscale deep features extracted by the CNN network are converted into a sequence of feature vectors, known as feature tokens. To preserve their positional integrity within the image, we utilize sinusoidal position embedding (SPE) to encode each vector. The size of the search region feature token is $(H_s/S, W_s/S, C)$, while the size of the template feature token is $(H_t/S, W_t/S, C)$. Here, S represents the downsampling scale of the backbone to the original image, and C represents the number of channels in the feature map. The FIAM incorporates position encoding information into both search region feature tokens and template tokens in a high-dimensional space. As a result, it obtains query matrices, key matrices, and value matrices for each Token, including Q_{ti}, K_{ti}, V_{ti}, Q_{td}, K_{td}, V_{td}, Q_s, K_s, V_s. The self-attention module based on search region features is referred to as SA_s; similarly, SA_t denotes the self-attention module based on template features. In addition, the CA signifies the cross-attention module that operates on both search region feature tokens and template feature tokens. We concatenate Q_{ti} and Q_{td}, K_{ti} and K_{td}, V_{ti} and V_{td} along the channel direction to obtain Q_t, K_t, and V_t respectively. The QKV matrices with subscript t represent the subfeature map matrices of the template.

$$
\begin{aligned}
SA_s &= \text{Softmax}\left(\frac{[Q_s][K_s]^T}{\sqrt{d_k}}\right) \cdot [V_s] \\
SA_t &= \text{Softmax}\left(\frac{[Q_{t_i}; Q_{t_d}][K_{t_i}; K_{t_d}]^T}{\sqrt{d_k}}\right) \cdot [V_{t_i}; V_{t_d}] \\
CA &= \text{Softmax}\left(\frac{[Q_s][K_t]^T}{\sqrt{d_k}}\right) \cdot [V_t]. \qquad (1)
\end{aligned}
$$

The matrices Q, K, and V in the aforementioned expression correspond to the query matrix, key matrix, and value matrix. The symbol d_k represents the dimensions of the query matrix, key matrix, and value matrix, which are the same constant values. The t_i and t_d indicate that the matrix belongs to the initial template and dynamic template, while the s denote that the matrix belongs to the search region.

Let *P* denote the outcome of the operation performed on the query matrix and the key matrix

$$
\text{Softmax}\left(\frac{Q_x K_y^T}{\sqrt{d_k}}\right) = P_{xy}. \qquad (2)
$$

The SA is approximately solved as depicted in (3). The SA resolves the inner product correlation between each feature token and all others, thereby characterizing the angular disparity of local feature vectors within the feature space. In scenarios where the target is partially occluded, besides leveraging CNN-modeled appearance depth features, the intrinsic correlations can provide valuable cues to guide the network in completing missing local features. This assists in modeling based on the global features of the target, ensuring the integrity of its appearance characteristics and mitigating tracking drift caused by damaged local features. Moreover, we sample N dynamic templates $\{t_1, t_2, \ldots, t_N\}$ and utilize self-attention to establish association among multiple templates in SA_t. Specifically, the subfeature vectors Token of the N templates are concatenated along the batch dimension to obtain Q_{td}, K_{td}, and V_{td} with spatio-temporal information. The SA retains spatio-temporally encoded global features of N templates, enabling the network to perceive long-term changes in target appearance and enhance robustness in long-term target tracking

$$
SA = SA_s + SA_t \triangleq P_{ss} \cdot V_s + P_{tt} \cdot V_t. \qquad (3)
$$

The cross-attention expression CA is approximately solved as depicted in (4). On one hand, the CA utilizes the appearance feature Token constructed by CNN to address the local feature correlation between the template and the target. On the other hand, within the multilayer cross-attention solution, the CA leverages the global modeling outcomes of the SA to tackle global feature correlation. Integrating local and global feature correlations effectively enhances the accuracy of feature matching.

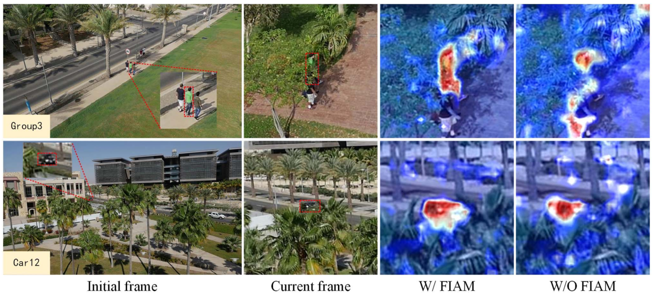


Fig. 3. Visualization results of different attention weights on Group3 and Car12 of UAV123.

$$
CA \triangleq P_{st} \cdot V_t. \qquad (4)
$$

The working mechanism of FIAM is intuitively depicted in Fig. 3. Including FIAM in the network enables the construction of global features by leveraging local subfeature relationships, thereby enhancing attention toward the complete target in occlusion scenarios and ultimately improving tracking robustness. Conversely, without FIAM, the network primarily focuses on unoccluded parts of the target, which may lead to tracking drift or errors in scale estimation.

B. Feature Verification Module

It is worth noting that the confidence scores based on similar responses to local appearance features may not immediately decrease when the large occlusion covers the target. This phenomenon can be attributed to the tracking network's ability to use local information to complete feature matching due to masked positive samples during training. As long as the discriminative features of the target remain unaffected by occlusion, the tracking result will still exhibit a relatively high response. While this phenomenon has for the tracking task itself, it poses potential risks in determining the timing of template updating. The tracking network needs to update the dynamic template based on the current target state; however, extensive occlusion introduces noise information that potentially contaminates the historical template pool.

We have developed an additional verifier to assess the extent of compromise in feature integrity, as illustrated in Fig. 4. This module takes the global feature from feature modeling as input, which includes both the global feature SA_s from the search region and the global feature SA_t from the template. We perform a reverse mapping of the bounding box from the tracker's output to its corresponding position in SA_s and extract a portion of SA_s with spatial resolution matching that of Sat, denoted as SA'_s. Subsequently, we compute the inner product correlation between the target global feature SA'_s and the template global feature SA_t. To provide a more intuitive measure of target feature completeness, we introduce a score token V_{Score} for independent training of this verifier after completing tracker training. $Q_{SA'_s}$ and K_{SA_t} are obtained through global feature encoding, with their correlation established by the inner product. The resulting

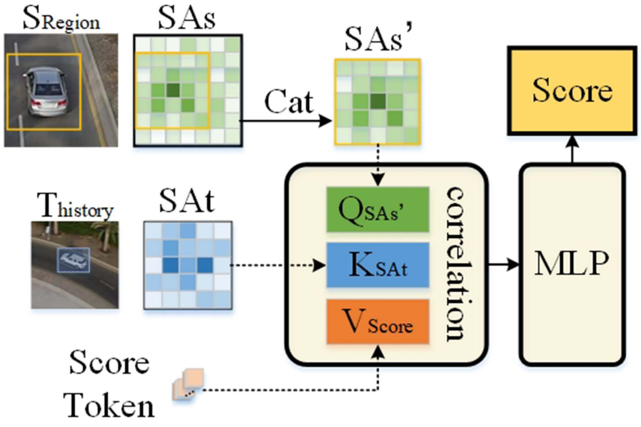


Fig. 4. Structure diagram of FVM based on integrity discrimination.

inner product correlation is then mapped onto the score Token, followed by a multilayer perceptron (MLP) output to generate the final result. The expression is as follows:

$$\text{Score} = \text{MLP}\left(\text{Softmax}(\frac{[Q_{SA'_s}][K_{SA_t}]^T}{\sqrt{d_k}}) \cdot [V_{\text{Score}}]\right) \quad (5)$$

The learning objectives of the verifier network and the tracker network differ. Specifically, while the tracking network considers all masked samples as positive samples to enhance its ability to reason the global information from locally damaged features, the verifier network treats samples with a mask range greater than half as negative samples. This ensures that the feature verification network can effectively determine whether target features are complete or incomplete, thereby reducing the introduction of noise during template updating. The feature verification network is trained using a standard cross-entropy loss function

$$L_{\text{Score}} = y_i \log(p_i) + (1 - y_i)\log(1 - p_i) \quad (6)$$

where y_i represents the groundtruth label, and p_i denotes the predicted score.

Similar to the cross-attention representation in the Transformer structure, we propose that $[Q_{SA'_s}][K_{SA_t}]^T$ effectively captures the inner product correlation of high-dimensional feature embeddings, thereby reflecting the angular disparity among features within a class. The minimal intraclass disparity of global features can be considered as an indication of the completeness of the target feature. Furthermore, since the FIAM has comprehensively modeled the global features, the FVM does not impose significant computational burdens, seamlessly aligning with platform deployment requirements.

C. Other Key Modules

1) Feature Extraction Module: The feature extraction backbone network has been redesigned as shown in Fig. 5 to overcome the limitations of computing power. It is important to note that while native ResNet or MobileNet are better classification tasks, accurate target localization requires higher demands on the target tracking task. Taking inspiration from NAS in LightTrack, we have replaced the 3 × 3 small-size convolution kernels with a significant number of 7×7 and 5 × 5 large-size convolution kernels to expedite the acquisition of a sufficiently large receptive field and enhance the positioning accuracy of the tracker.

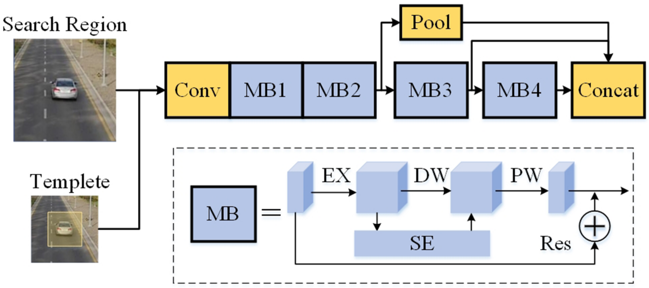


Fig. 5. Lightweight backbone for efficient feature extraction. Conv represents an ordinary conventional operation. MB incorporates the architectural components from MobileNet, which consist of Expansion (EX), Depth-Wise convolution (DW), Point-Wise linear (PW), Squeeze Excitation (SE), and residual connection (Res).

Furthermore, to ensure that the final output feature map encompasses both high-level semantic information and low-level details, such as the texture and color of the target object, we employ a concatenation approach for multilevel feature maps. This enhances the accuracy of tracking small targets. After max-pooling downsampling (Pool in Fig. 5), the low-level feature map maintains the same spatial resolution as the high-level feature map. A 1×1 convolution kernel is used to adjust the channel weights of each feature map, and then these maps are concatenated (Concat in Fig. 5) along the channel dimension. The resulting output of the feature map can be expressed as follows.

$$E_{\text{all}}(\cdot) = \sum \alpha_i \otimes \eta_i(\cdot) \quad (7)$$

where α_i represents the scaling factor for channel weights of each layer's feature map, while $\otimes$ denotes the convolution operation.

2) State Estimation Module: Our tracking state estimation module is inspired by the STARK tracking network and uses a corner-based fully convolutional module that generates two probability maps representing the top-left and bottom-right corners of the object bounding box. By utilizing a fully convolutional network for feature matching, we can obtain both probability distributions for each corner as well as for the overall bounding box of our target. During training, we employ a combination of L1 loss and GIOU loss to train our tracking network end-to-end while also incorporating a set of hyperparameters for weight adjustment

$$L_{\text{all}} = \gamma_1 L_1 + \gamma_2 L_{\text{giou}} \quad (8)$$

where γ_1 and γ_2 represent the weight ratio in the loss function.

IV. Experiments

In this subsection, we conduct quantitative and qualitative experiments on three publicly available tracking datasets (UAV20 L, UAV123, and LaSOT) to compare the proposed method with other state-of-the-art tracking networks and validate the effectiveness of both the FIAM and the feature validator module.

A. Dataset and Evaluation Metrics

UAV20L [48]: As the number of frames within a sequence increases in long-term tracking, the challenges associated with changes in object position, size, perspective, illumination, and other factors become progressively more demanding. This intensifies the complexity of visual tracking. The dataset comprises 20 extensive sequences, with the largest one of over 5000 frames and an average length of close to 3000 frames per sequence. Consequently, UAV20 L is employed to assess the performance of long-term tracking.

UAV123 [48]: The UAV123 dataset has been selected to evaluate the adaptability of the long-term tracking network to various challenges in short-term tracking. This dataset encompasses diverse tracking scenarios, including illumination changes, fast motion, occlusion, target deformation, background interference, and scale variations. It comprises 123 video clips and over 100 000 frames of images.

LaSOT [51]: The dataset is a large-scale, high-quality dataset for long-term tracking of single objects, consisting of 1400 challenge videos: 1120 for training and 280 for testing. With over 3.52 million frames of manually labeled images covering more than 70 categories, this dataset enables comprehensive and robust evaluation of tracking networks.

The tracking network is assessed using the one-pass evaluation across all the aforementioned datasets, with accuracy and success rate serving as the evaluation metrics. Specifically, the success rate is measured through IoU, where the success plot illustrates the proportion of frames that exceed a predetermined threshold. The area-under-the-curve of this plot is employed for ranking purposes. Furthermore, accuracy is gauged by center location error, which quantifies the disparity between estimated and ground truth bounding boxes. The accuracy plot shows the percentage of scenes in which said disparity falls below the thresholds. In the following section, a score at 20 pixels is utilized for ranking.

B. Implementation Details

The training process of the tracking network comprises two stages: 1) tracker training and 2) verifier training. The entire process is trained end-to-end using four datasets: 1) COCO [49]; 2) GOT-10K [50]; 3) LaSOT [51], and 4) TrackingNet [52]. During the tracker training phase, we assign hyperparameter weights γ_1 and γ_2 to be 5 and 2 in the loss function. The training phase requires 800 epochs, with the global learning rate decaying from 0.0004 to 0.00001. It is worth noting that the feature modeling network's learning rate is set at only 0.1 times the global learning rate. As for the selection of positive and negative samples, we chose sample pairs consisting of a 128×128 template image patch and a 288×288 search field image patch from the dataset. Subsequently, we randomly applied a mask coverage rate of less than 75% on the 288×288 search field, considering mask sample pairs containing the target in the same sequence as positive samples. Sample pairs that do not contain the target are considered negative samples. During the verifier training phase, we freeze the parameters of the tracker network and independently train the verifier network for 200 epochs. To decrease the correlation response between the template feature and the target feature during occlusion, masked samples with over 50% coverage are utilized as negative samples, while the learning rate is set to 0.001 and decayed to 0.00001 in gradient fashion.

We instantiate two types of trace models: 1) Model-L and 2) Model-S. By modifying the hidden feature layer dimension of FIAM, the network exhibits distinct parameters and computational complexity, as illustrated in Table I.

C. Quantitative Evaluation

On the UAV20 L dataset, we compared our proposed network with eight advanced convolutional neural network-based tracking networks, namely SiamGAT [39], LightTrack [38], SiamBAN [53], SiamRPN++ [32], SiamRPN [54], SiamCAR [55], Ocean [56], and SiamDW [33]. These tracking networks encompass models focusing solely on spatial information (e.g., LightTrack and SiamRPN++) as well as those incorporating interframe relationships (e.g., Ocean). All tracking network results were obtained from published data [20]. The success rate of the proposed tracking network reaches 70.6%, as presented in Table II, surpassing that of SiamGAT by 8.6%. The leading results on the long-term tracking dataset demonstrate that incorporating a robust antiocclusion capability into the tracking network effectively mitigates drift error accumulation and significantly enhances the success rate of the tracking task. Furthermore, this outcome further emphasizes the significant improvement achieved through the integration of global information into the tracking network to enhance its anti-occlusion ability.

To evaluate the robustness of the long-term tracking network in diverse environments, we further assess its performance on the UAV123 tracking dataset. The experimental results are presented in Fig. 6. It is worth noting that all methods utilize the lightweight configuration of this tracking network, which aligns well with real-world UAVs earth observation application scenarios. Our proposed algorithm achieves a success rate of 69.2% and a precision rate of 90.3%. Compared to LightTrack, another lightweight tracking network, our approach demonstrates an improvement of 4.8% in success rate and 8.1% in precision rate. Furthermore, our method exhibits performance enhancements of 5.2% in success rate and 7.0% in precision rate compared to SaimGAT.

We conduct additional tests on challenging problems, such as background clutter, scale changes, partial occlusion, and complete occlusion, that are prone to occur in long-term tracking tasks. The proposed tracker significantly enhances the success rate and precision rate. With the help of the template updating mechanism, we effectively adapt to target scale changes in long-term tracking scenarios. Furthermore, the FVM ensures that the template remains unaffected by noise information caused by occlusions during tracking, thereby establishing a more stable

TABLE I
IMPORTANT PARAMETERS OF THE MODEL, ALONG WITH THEIR CORRESPONDING UTILIZATION OF COMPUTATIONAL AND STORAGE RESOURCES

Model	Hidden dims	Backbone params (M)	Backbone flops (G)	Head params (M)	Head flops (G)	Total flops (G)
Model-L	256	6.066	2.769	6.464	2.537	5.306
Model-S	96	1.917	0.875	0.891	0.349	1.224

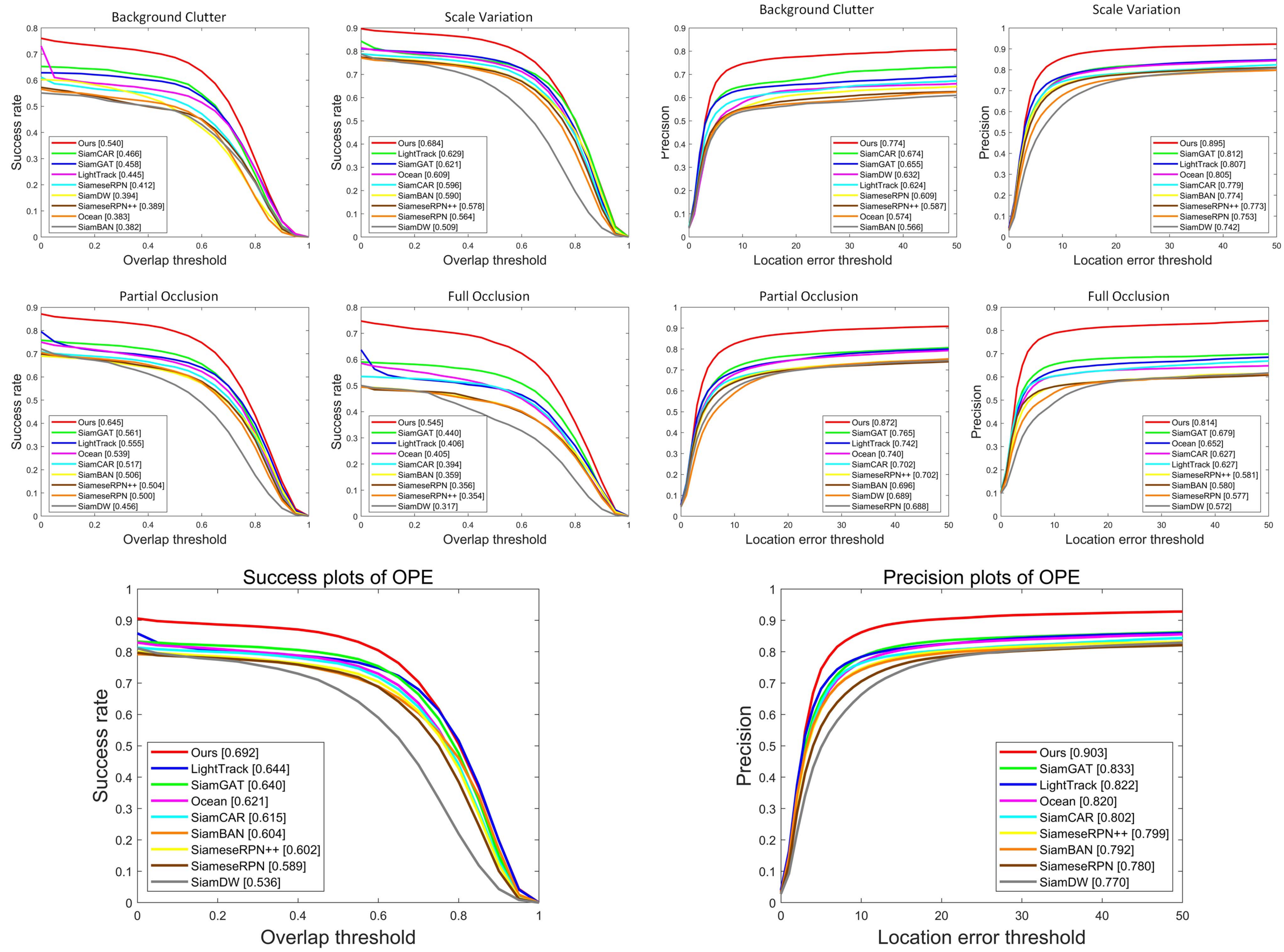


Fig. 6. Performance of the nine tracking networks is comprehensively evaluated on UAV123 while also presenting evaluation results for common scenarios encountered in long-term tracking, including Background Clutter, Scale Variation, Partial Occlusion, and Full Occlusion. Our tracking network surpasses others in terms of both the overall evaluation metric and each subevaluation metric.

TABLE II
COMPARISONS ON UAV20 L TEST SET

Tracker	Success rate	Precision rate
Ours	70.6	92.8
SiamGAT	62.0	79.6
LightTrack	62.0	79.1
SiamBAN	56.4	73.6
SiamRPN++	54.7	72.3
SiamRPN	52.4	67.4
SiamCAR	52.3	68.7
Ocean	44.4	63.0
SiamDW	40.7	63.2

template pool and endowing the tracking network with anti-occlusion capabilities. In addition, our modeling method based on intrinsic feature association exhibits strong resistance when targets are disturbed by background clutter since it enables effective utilization of local features to complement global features and enhance overall robustness.

On the LaSOT dataset, we compare the aforementioned convolutional neural network-based tracking network with the proposed approach. By utilizing large-scale datasets like LaSOT, we can comprehensively evaluate the effectiveness of the tracking network in diverse environments. Fig. 7 visually illustrates the tradeoff between tracking performance and computational resource consumption for each tracking network. Considering practical constraints on UAVs platform performance, we advocate deploying tracking networks that consume less than 6G Flops of computing resources. The specific test values of the

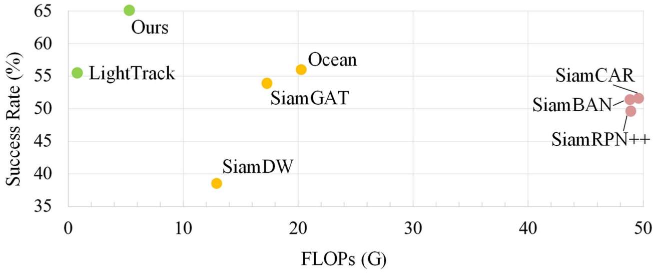


Fig. 7. Success rate and computational complexity of the proposed tracking network are compared with state-of-the-art tracking networks based on the LaSOT benchmark set. Different colors indicate the suitability of each tracking network for deployment on an embedded platform.

TABLE III
COMPARISONS ON LASOT TEST SET

Tracker	FLops (G)	SUC	PRE	PRE-n
Ours-L	5.31	65.1	68.8	74.6
Ours-S	1.22	60.7	62.9	69.9
SiamGAT	17.28	53.9	53	63.3
LightTrack	0.79	55.5	56.1	—
SiamBAN	48.84	51.4	52.1	59.8
SiamRPN++	48.92	49.6	49.1	56.9
Ocean	20.26	56	56.6	65.1
SiamCAR	49.6	51.6	52.4	61.0
SiamDW	12.9	38.5	38.9	48

TABLE IV
SUCCESS AND PRECISION RATE (IN PERCENTAGE) WITH RESPECT TO THE FRAME INTERVAL AND THE TEMPLATE FEATURES NUMBER PARAMETERS

Parameters	Frame interval				Template features number			
	50	100	200	400	1	2	3	5
SUC	68.1	68.6	69.2	67.9	65.1	67.6	69.2	68.8
PRE	88.9	89.4	90.3	88.1	85.6	88.0	90.3	89.8

tracking network are presented in Table III. Both our tracking network and LightTrack exhibit low computational requirements. However, compared to LightTrack, our proposed tracking network demonstrates a 9.6% improvement in success rate and an 12.7% improvement in precision rate. Consequently, the proposed tracking network demonstrates a superior performance-price ratio within the context of Earth observation applications.

D. Parametric Sensitivity

As demonstrated in Section IV-B, our tracker requires the configuration of numerous hyperparameters. Initially, we maintain consistency with SiamRPN++ by setting the common parameters γ_1, γ_1, and α_i to identical values for a fair comparison. Subsequently, we conduct experiments on the UAV123 dataset using different settings for the update frame interval, number of retained template features, and occlusion rate threshold. To ensure fairness in comparisons, all other parameters are kept constant when evaluating each specific parameter.

First, we conducted experiments on the configuration of the Frame Interval of the dynamic template and the Template Features Number, and the results are shown in Table IV. The findings indicate that performance significantly degrades when the Frame Interval is set to 50 or 400. Moreover, under a high-frequency update strategy, excessive attention to recent features and forgetting historical information exacerbate due to highly similar appearances within a period of time. This situation is not conducive to long-term tracking of targets by the network. Conversely, adopting a low-frequency update strategy may lead to target matching drift or tracking failure due to drastic changes in target scale and viewing angle. Furthermore, the performance experiences a significant decrease when only one or two template features are retained, suggesting that the tracking network relies on multiple template features to fully capture variations in target appearance over an extended period of time. This improvement plays a crucial role in enhancing the resilience of long-term target tracking.

TABLE V
SUCCESS AND PRECISION RATE (IN PERCENTAGE) WITH RESPECT TO THE OCCLUSION PERCENTAGE THRESHOLD

VALUE	0.2	0.35	0.5	0.75	0.9
SUC	66.8	67.8	69.2	69.1	68.4
PRE	87.2	87.9	90.3	89.9	88.8

Furthermore, as mentioned in Section III-B, our tracking network utilizes the FVM to evaluate the degree of damage to template appearance features and selects templates that exceed the predefined threshold for participation in template update. Consequently, we conducted experimental analysis on the configuration with the Occlusion Percentage Threshold, and the results are presented in Table V. The findings indicate that setting a low threshold (e.g., 0.2 or 0.35) poses the risk of introducing noise through template updates, thereby, diminishing tracking performance. Conversely, adopting a high threshold (e.g., 0.9) leads to excessively strict criteria that exclude mildly occluded samples from participating in template updates. This limitation hampers accurate perception of target state when significant appearance changes occur during prolonged mild occlusion periods, ultimately impairing tracking performance.

E. Qualitative Evaluation

In this section, a qualitative comparison is conducted between the proposed tracking network and eight other advanced trackers, as shown in Fig. 8. The UAV123 dataset is selected as the benchmark for comparison, from which three sets of representative image sequences with occluded targets are chosen: (a) Car7, (b) Group2, and (c) Group3.

In the Car7 sequence, the black car is frequently occluded by trees for short durations, with approximately 10 frames per occlusion. When the target experiences frequent occlusions, the verifier within the network detects these occlusions and triggers a brief locking state in the tracking network to prevent disturbances from incorrect responses, thereby minimizing significant drift in the tracking position. When the target reappears, the network performs new matching based on historical template characteristics.

In the Group2 sequence, the person remains occluded by the building for an extended period, lasting much longer than

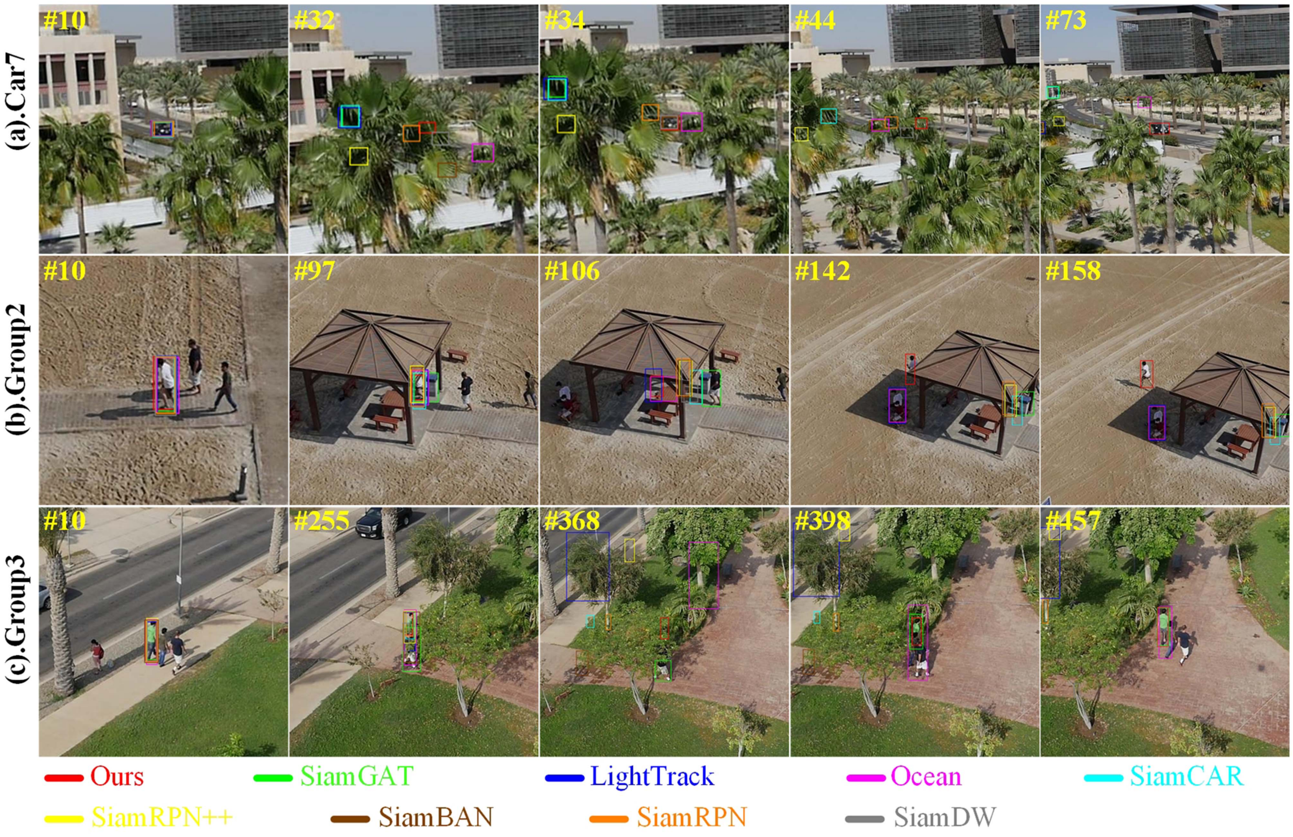


Fig. 8. Visual tracking results on UAV123.The figure consists of three subsequences, corresponding to challenges posed by frequent short-term occlusion, long-term complete occlusion, and masked coverage occlusion.

TABLE VI
RESULTS OF DIFFERENT FEATURE MODELING AND TEMPLATE UPDATING STRATEGIES ON UAV123 DATASET

No.	Modules		Success rate	Precision rate
	Feature Modeling	Template Updating strategy		
#1	CNN	ST	65.1	85.6
#2	CNN	FFR	64.9	84.6
#3	CNN	TCR	66.5	87.1
#4	CNN	FVM	67.5	87.9
#5	CNN + FIAM	FVM	69.2	90.3

10 frames. The proposed tracking network transitions into a lock state and gradually expands the search area to ensure the successful recapture of the target and achieve stable tracking upon its reappearance.

In the Group3 sequence, only a partial amount of appearance information remains visible after the person is randomly occluded by branches and leaves. On one hand, this proposed tracking network utilizes local appearance feature clues to complete global features for maintaining tracking. On the other hand, the verifier network determines feature integrity to prevent degradation caused by an excessive amount of background information introduced into the history template.

However, we observed that the tracking performance is often affected by similar target interference during occlusion. Furthermore, long-term full occlusion may result in significant target variations and lead to inferior performance. In future, we will continue to optimize our approach to address such challenge.

F. Ablation Study

In this section, we present extensive analysis of the proposed submodules on the UAV123 dataset, which include the FIAM, the FVM, and a manually designed backbone feature extraction network.

We designed a range of template updating conditions to validate the efficacy of the FVM, and the comparative results are presented in Table VI. In Experiment #1, we employed a static template (ST) strategy where only the initial frame of target information was utilized as the template without any subsequent updates during tracking. The tracking achieved a

TABLE VII
Results of Testing Backbone on UAV123 Dataset

ResNet	Ours			Success rate	Precision rate
	MB4	MB3	MB2		
✓		-		65.2	86.3
-	✓	✗	✗	68.1	88.6
-	✓	✓	✓	69.2	90.3

success rate of 65.1% and a precision rate of 85.6%. In Experiment #2, we adopted a dynamic template approach that updated at a fixed frame rate (FFR), resulting in a success rate of 64.9% and a precision rate of 84.6%. By comparing the test outcomes between Experiment #1 and Experiment #2, it can be observed that updating the template at a fixed frame rate may introduce erroneous information and result in the degradation of the template, thereby reducing the robustness of the dynamic template compared to that of the static template. In Experiment #3, we utilize the tacking correlation response (TCR) of the tracking network as the criterion for template updating, with a response threshold set at 50%. The tracking success rate achieves 66.5%, and the precision rate reaches 87.1%. In Experiment #4, we employ the score from the FVM as the basis for template updating while maintaining a threshold of 50%. Consequently, the tracking success rate improves to 67.5%, and the precision rate improves to 87.9%. By comparing test results between Experiment #3 and Experiment #4, it is evident that establishing an independent verifier to assess the completeness of target appearance becomes imperative in effectively preventing template contamination caused by noise. Experiment #5 incorporates a FIAM after the CNN and utilizes the feature verification score (FVM) as the decision criterion for template updating. We still set the threshold at 50%. The tracking success rate is 69.2%, and the precision rate is 90.3%. Comparing the test results of Experiment #4 and Experiment #5, we observe that FIAM can construct more robust global features, leading to a further improvement in both tracking success rate and precision rate by 1.7% and 2.4%, respectively.

As stated in Section III-D, our tracking network utilizes a manually designed lightweight backbone network. To validate this backbone, we exclusively focus on evaluating it alongside the complete tracker, incorporating the FIAM and the FVM as configuration baselines. The corresponding results are presented in Table VII. First, we selected ResNet as the feature extraction network, which yields in a tracking success rate and precision rate of 65.2% and 86.3%. By employing a single-stage output feature map (MB4 in Fig. 5) from a lightweight backbone network, we achieved 2.9% and 2.3% improvement in the success rate and precision rate, respectively. These results demonstrate that our lightweight backbone network is more suitable for UAVs object tracking tasks compared to ResNet. Consequently, we have implemented a configuration that integrates multilayer feature maps (MB4 + MB3 + MB2 in Fig. 5.), which shows 1.1% and 1.7% improvement in the tracking success rate and precision rate. This experiment demonstrates the advantageous impact of fusing convolutional image features from multiple layers on enhancing target perception ability for tracking tasks. In addition, our lightweight backbone network only requires 0.35G Flops of computation, which is over 100 times less than that of ResNet, making it highly suitable for deployment on UAVs platforms.

V. Conclusion

This article presents an antiocclusion tracker for long-term Earth Observation tasks. Following the CNN backbone network, an intrinsic feature association module exploits the internal potential correlation of local appearance features to complement global features, thereby enhancing robustness in occlusion scenarios. In addition, an FVM supplements occlusion supervision information and carefully filters cases where the target is heavily occluded to ensure the purity of the template. Experimental results on public datasets, including UAV20 L, UAV123, and LaSOT, demonstrate the superior performance of the proposed tracking network, which could be deployed on UAVs computing platforms.

References

[1] Y. Han, H. Liu, Y. Wang, and C. Liu, “A comprehensive review for typical applications based upon unmanned aerial vehicle platform,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 9654–9666, 2022.

[2] S. Chen et al., “Vehicle tracking on satellite video based on historical model,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 7784–7796, 2022.

[3] X. Xu et al., “STN-Track: Multiobject tracking of unmanned aerial vehicles by swin transformer neck and new data association method,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 8734–8743, 2022.

[4] J. Shao, B. Du, C. Wu, and L. Zhang, “Tracking objects from satellite videos: A velocity feature based correlation filter,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7860–7871, Oct. 2019.

[5] M. Thomas, C. Kambhamettu, and C. A. Geiger, “Motion tracking of discontinuous sea ice,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 5064–5079, Dec. 2011.

[6] B. He, X. Zhao, Y. Chen, C. Liu, and X. Pang, “Application of feature tracking using K-nearest-Neighbor vector field consensus in sea ice tracking,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4326–4336, 2022.

[7] C. Li, G. Li, Z. Chen, X. Wang, and X. Cheng, “Matching vector filtering methods for sea ice motion detection using SAR imagery feature tracking,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6197–6202, 2022.

[8] Z. Li et al., “A Super-resolution algorithm based on hybrid network for multi-channel remote sensing images,” *Remote Sens.*, vol. 15, no. 14, 2023, Art. no. 3693.

[9] C. Zhong, J. Ding, and Y. Zhang, “Video SAR moving target tracking using joint kernelized correlation filter,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1481–1493, 2022.

[10] H. Zhang et al., “UAV tracking based on correlation filters with dynamic aberrance-repressed temporal regularizations,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 7749–7762, 2023.

[11] Y. Zhang and Y. Zheng, “Object tracking in UAV videos by multi-feature correlation filters with saliency proposals,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 5538–5548, 2023.

[12] Y. Han, C. Deng, B. Zhao, and D. Tao, “State-aware anti-drift object tracking,” *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4075–4086, Aug. 2019.

[13] Y. Han, C. Deng, B. Zhao, and B. Zhao, “Spatial-temporal context-aware tracking,” *IEEE Signal Process. Lett.*, vol. 26, no. 3, pp. 500–504, Mar. 2019.

[14] C. Deng, S. He, Y. Han, and B. Zhao, “Learning dynamic spatial-temporal regularization for UAV object tracking,” *IEEE Signal Process. Lett.*, vol. 28, pp. 1230–1234, 2021.

[15] B. Zhao et al., “Towards long–term UAV object tracking via effective feature matching,” *Electron. Lett.*, vol. 56, no. 20, pp. 1056–1059, 2020.

[16] Y. Han et al., “Boundary–aware vehicle tracking upon UAV,” *Electron. Lett.*, vol. 56, no. 17, pp. 873–876, 2020.

[17] Z. Cao, C. Fu, J. Ye, B. Li, and Y. Li, “SiamAPN: Siamese attentional aggregation network for real-time UAV tracking,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, 3086–3092.

[18] Y. Yu, Y. Xiong, W. Huang, and M. R. Scott, “Deformable siamese attention networks for visual object tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6728–6737.

[19] F. Tang and Q. Ling, “Ranking-based siamese visual tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8741–8750.

[20] C. Fu et al., “Siamese object tracking for unmanned aerial vehicle: A review and comprehensive analysis,” *Artif. Intell. Rev.*, vol. 56, pp. 1417–1477, 2023.

[21] L. Shi, Q. Zhang, B. Pan, J. Zhang, and Y. Su, “Global-local and occlusion awareness network for object tracking in UAVs,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 8834–8844, 2023.

[22] K. Dai, Y. Zhang, D. Wang, J. Li, H. Lu, and X. Yang, “High-performance long-term tracking with meta-updater,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6298–6307.

[23] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, “Learning spatio-temporal transformer for visual tracking,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10448–10457.

[24] J. Ye, C. Fu, F. Lin, F. Ding, S. An, and G. Lu, “Multi-regularized correlation filter for UAV tracking and self-localization,” *IEEE Trans. Ind. Electron.*, vol. 69, no. 6, pp. 6004–6014, Jun. 2022.

[25] D. Yuan et al., “Learning adaptive spatial-temporal context-aware correlation filters for UAV tracking,” *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 18, no. 3, pp. 1–18, 2022.

[26] H. Yu et al., “Conditional GAN based individual and global motion fusion for multiple object tracking in UAV videos,” *Pattern Recognit. Lett.*, vol. 131, pp. 219–226, 2020.

[27] C. Fu, Z. Cao, Y. Li, J. Ye, and C. Feng, “Onboard real-time aerial tracking with efficient siamese anchor proposal network,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5606913.

[28] Y. Cui, B. Hou, Q. Wu, B. Ren, S. Wang, and L. Jiao, “Remote sensing object tracking with deep reinforcement learning under occlusion,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5605213.

[29] J. Feng, B. Hui, Y. Liang, Q. Yao, and X. Zhang, “Improved siamRPN with clustering-based frame differencing for object tracking of remote sensing videos,” in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, 2021, pp. 4163–4166.

[30] L. Bertinetto et al., “Fully-convolutional siamese networks for object,” in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Springer International Publishing, 2016, pp. 850–865.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1106–1114.

[32] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, “SiamRPN: Evolution of siamese visual tracking with very deep networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4282–4291.

[33] Z. Zhang and H. Peng, “Deeper and wider siamese networks for real-time visual tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4591–4600.

[34] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.

[35] X. Zhang, X. Zhou, M. Lin, and J. Sun, “ShuffleNet: An extremely efficient convolutional neural network for mobile devices,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6848–6856.

[36] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[37] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[38] B. Yan, H. Peng, K. Wu, D. Wang, J. Fu, and H. Lu, “Finding lightweight neural networks for object tracking via one-shot architecture search,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Nashville, TN, USA. 2021, pp. 20–25.

[39] D. Guo, Y. Shao, Y. Cui, Z. Wang, L. Zhang, and C. Shen, “Graph attention tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9543–9552.

[40] B. Ye et al., “Joint feature learning and relation modeling for tracking: A one-stream framework,” in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2022, pp. 341–357.

[41] Y. Cui, C. Jiang, L. Wang, and G. Wu, “End-to-end tracking with iterative mixed attention,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 18–24.

[42] L. Jiao et al., “Transformer meets remote sensing video detection and tracking: A comprehensive survey,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 1–45, 2023.

[43] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[44] A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. 9th Int. Conf. Learn. Represent.*, 2021.

[45] N. Carion et al., “End-to-end object detection with transformers,” in *Proc. Eur. Conf. Comput. Vis.*, Cham: Springer International Publishing, 2020, pp. 213–229.

[46] I. Jung et al., “Real-time MDNet,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 83–98.

[47] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, “ATOM: Accurate tracking by overlap maximization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4660–4669.

[48] M. Mueller, N. Smith, and B. Ghanem, “A benchmark and simulator for UAV tracking,” in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Springer International Publishing, 2016, pp. 445–461.

[49] T. Y. Lin et al., “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis.*, Zurich, Switzerland, Springer International Publishing, 2014, pp. 740–755.

[50] L. Huang, X. Zhao, and K. Huang, “GOT-10 k: A large high-diversity benchmark for generic object tracking in the wild,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1562–1577, May 2021.

[51] H. Fan et al., “LaSOT: A high-quality benchmark for large-scale single object tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5374–5383.

[52] M. Muller et al., “TrackingNet: A large-scale dataset and benchmark for object tracking in the wild,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 300–317.

[53] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, “Siamese box adaptive network for visual tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6668–6677.

[54] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, “High performance visual tracking with siamese region proposal network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8971–8980.

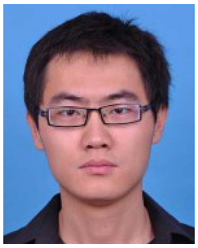
[55] D. Guo, J. Wang, Y. Cui, Z. Wang, and S. Chen, “SiamCAR: Siamese fully convolutional classification and regression for visual tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6269–6277.

[56] Z. Zhang et al., “Ocean: Object-aware anchor-free tracking,” in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K., Springer International Publishing, 2020, pp. 771–787.

Qiuyu Jin received the M.Sc. degree, in 2017, in instrument science and technology from the Beijing Institute of Technology, Beijing, China, where he is currently working toward the Ph.D. degree in electronic and information engineering focusing on intelligent image processing, visual object tracking, and machine learning.

Yuqi Han received the B.Eng. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2015, and the B.Sc. degree in economy from the National School of Development, Peking University, Beijing, in 2015. He received the Ph.D. degree in information and communication engineering from the School of Information and Electronics, Beijing Institute of Technology, in 2021.

From 2021, he was a Research Fellow with the Department of Computer Science and Technology, Tsinghua Univeristy.. He is currently an Assistant Professor with the School of Information and Electronics, Beijing Institute of Technology. His research interests include computer vision, remote sensing, and UAV.



Wenzheng Wang received the Ph.D. degree in information and communication engineering degree from the School of Electrical and Information Engineering, Beijing Institute of Technology, Beijing, China, in 2019.

He was a Postdoctoral Research Fellow with the School of Electronics Engineering and Computer Science, Peking University, Beijing. He is currently a tenure-track Assistant Professor with the Beijing Institute of Technology. His research interests include hyperspectral/optical imagery target detection and image analysis.

Linbo Tang was born in 1978. He received the Ph.D. degree in information and communication engineering from the School of Information and Electronics, Beijing Institute of Technology, Beijing, China, in 2005.

He has been engaged in Teaching and Research work with Radar Research Laboratory, Beijing Institute of Technology, Beijing, China. He has undertaken 863 and H863 projects. His research interests include image processing and real-time signal processing.

Jianan Li received the B.S. degree in optoelectronic information science and engineering, in 2013 and the Ph.D. degree in optical engineering, in 2019, both from School of Optics and Photonics, Beijing Institute of Technology, Beijing, China, where he is currently an Assistant Professor.

From 2015 to 2017, he worked as a joint training Ph.D. student at National University of Singapore. From 2017 to 2018, he worked as an intern at Adobe Research. His research interests mainly include computer vision and real-time image/video processing.

Chenwei Deng received the Ph.D. degree in signal and information processing from the Beijing Institute of Technology, Beijing, China, in 2009.

Since 2012, he has been an Associate Professor and then a Full Professor with the School of Information and Electronics, Beijing Institute of Technology. Prior to this, he was a Post-Doctoral Research Fellow at the School of Computer Engineering, Nanyang Technological University, Singapore. He has authored or co-authored over 50 technical papers in refereed international journals and conferences. He has co-edited one book. His current research interests include video coding, quality assessment, perceptual modeling, feature representation, object recognition, and tracking.