Multi-Scale Enhanced Features Correlation Filters Learning With Dual Second-Order Difference for UAV Tracking

Yu-Feng Yu, *Member, IEEE*, Yang Zhang, Long Chen, *Senior Member, IEEE*, Pengfei Ge, and C. L. Philip Chen, *Fellow, IEEE*

***Abstract*—Currently, most Discriminative Correlation Filters (DCF) algorithms used for Unmanned Aerial Vehicle (UAV) target tracking primarily focus on improving the tracking model. However, in UAV tracking, the tracked targets are typically small in size and frequently undergo scale variations, making singular improvements to tracking models less effective. As a response to this challenge, we propose a novel feature preprocessing approach. Specifically, for the extracted Histogram of Oriented Gradients (HOG) and Color Names (CN) features, we simulate their transformations at different scales and electively enhance the target region features based on global features to obtain multi-scale enhanced features. By implementing these procedures, the tracker improves its ability to recognize targets and exhibits increased adaptability in challenging tracking scenarios. Furthermore, in contrast to the conventional approach used by most UAV algorithms, which unidirectionally incorporate historical filters into model updates to prevent filter divergence, we introduce dual second-order difference terms that correspond to features and filters. This integration enables a more effective fusion of historical information with current frame data, thereby enhancing the robustness of the filtering process. Extensive experiments are conducted to evaluate the proposed tracker against other state-of-the-art trackers using the DTB70, UAV123@10fps, and UAVDT datasets. The experimental results affirm the effectiveness of our approach.**

***Index Terms*—Multi-scale enhanced features, dual second-order difference, discriminative correlation filters, UAV tracking.**

I. Introduction

SINCE the advent of the machine learning and deep learning waves, numerous research directions have emerged, with the investigation of target tracking algorithms standing out as a prominent theme. In the industrial sector, the application prospects of target tracking hold immense potential, such as in the context of path tracking for autonomous driving vehicles [1], [2] and multi-target vehicle tracking systems [3], [4]. Beyond ground transportation, in the aviation domain, as unmanned aerial vehicles (UAVs) become widely prevalent globally, research in the field of computer vision pertaining to UAV tracking has remained incessant.

Looking ahead, the integration of intelligent vehicles and UAV tracking technology has the capacity to enhance the efficiency and convenience of urban transportation systems. The synergy between smart cars and UAV technology could yield several benefits for urban traffic management. UAVs could play a pivotal role in enhancing driving safety by assisting in the monitoring of the surrounding environment for autonomous vehicles. These UAVs can be deployed for real-time traffic surveillance [5], [6], identifying traffic accidents and congestion, strengthening inter-vehicle communication [7] and relaying up-to-the-minute traffic information to smart vehicles. This, in turn, would aid vehicles in selecting the most optimal routes for their journeys, especially in avoiding traffic hazards [8].

The current state of UAV target tracking primarily involves two mainstream algorithms, namely Discriminative Correlation Filters (DCF) and Deep Learning methods. Tracking algorithms based on DCF typically begin by extracting hand-crafted multi-channel features, such as Histogram of Oriented Gradients (HOG) and Color Names (CN), for the target region based on initial target position in the initial frame. Subsequently, filters are trained based on the extracted features. In the subsequent frames, filters are applied to the image, and they slide across the image, computing response values at each position. These response values signify the confidence level of the target's presence at that location. The position with the maximum response value is considered the predicted target location. The algorithm iteratively repeats these steps to achieve continuous target tracking.

Manuscript received 1 December 2023; revised 4 January 2024; accepted 13 January 2024. Date of publication 17 January 2024; date of current version 29 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62006056, in part by Guangdong Basic and Applied Basic Research Foundation under Grants 2022A1515011595 and 2020A1515111149, in part by the Science and Technology Planning Project of Guangzhou under Grant 2024A03J0401, in part by the Science and Technology Development Fund, Macao S.A.R under Grant 0046/2023/RIA1, in part by the University of Macau and the University of Macau Development Foundation under Grant MYRG-GRG2023-00106-FST-UMDF, and in part by the Talent Special Projects of School-level Scientific Research Programs under Guangdong Polytechnic Normal University under Grant 2021SDKYA066. *(Corresponding author: Long Chen.)*

Yu-Feng Yu and Yang Zhang are with the Department of Statistics, Guangzhou University, Guangzhou 511370, China (e-mail: yuyufeng220@163.com; 2112164137@e.gzhu.edu.cn).

Long Chen is with the Department of Computer and Information Science, University of Macau, Macau 999078, China (e-mail: longchen@umac.mo).

Pengfei Ge is with the School of Mathematics and Systems Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China (e-mail: gepengfei@gpnu.edu.cn).

C. L. Philip Chen is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: philipchen@scut.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIV.2024.3355171.

Digital Object Identifier 10.1109/TIV.2024.3355171

At present, DCF employs the Alternating Direction Method of Multipliers (ADMM) for iterative solutions due to its rapid convergence and provision of relatively accurate solutions. Beyond its prominent role in machine learning, ADMM exhibits extensive applicability across diverse domains including image processing [9], [10], financial sectors [11], [12], among others.

Deep tracking algorithms can be further categorized into two types. The first type of deep algorithm still follows the tracking framework based on DCF but replaces hand-crafted features with deep features extracted using deep neural networks. For example, Fully Convolutional Networks Tracking (FCNT) [13] employs VGG-Net to extract features from the third, fourth, and fifth layers of Convolutional Neural Network (CNN). Certainly, nowadays, network frameworks related to feature extraction have become more diversified, such as multi-modal feature constraints [14], multiscale attention feature extraction [15] and top-down view features [16]. The second type of deep algorithm utilizes the structure of deep neural networks. The renowned Siamese Fully Convolutional Network (SiamFC) [17] introduces a Siamese network, where one network is responsible for processing the target image, while the other network handles target candidate regions. These two networks learn to compare the similarity between the target and candidate regions across different frames. The candidate region with the highest similarity to the target image is considered the predicted target location.

Due to the reliance on expensive GPU hardware and the substantial time required for pre-training models, most deep tracking algorithms are unable to achieve real-time UAV tracking. Conversely, excellent trackers based on DCF strike a balance between tracking accuracy and speed. In UAV tracking scenarios, targets are often small in size and prone to sudden changes due to the high-speed motion of UAV. To address this challenge, many advanced trackers incorporate information from past frames into the model's updates to mitigate filter degradation issues. However, few researchers have focused on the preprocessing of extracted multi-channel features. Typically, they directly input hand-crafted features into the model for learning and updating. In our view, if we can preprocess the extracted global hand-crafted features to moderately enhance the features of the target region, the filter's learning from the features will be more efficient, making it easier to differentiate between the target and the background.

In this paper, we present a feature preprocessing method. During the feature extraction step, we obtain global hand-crafted features from the target region and the background region. Subsequently, we enhance the overall global features based on the mean and standard deviation of the global features, resulting in enhanced global features. Following this, leveraging the mean and standard deviation of the enhanced global features, we individually enhance the features in the target region to emphasize the contrast between the target and background. As a result of the potential for frequent changes in the scale of the target during the UAV tracking process, the scale of the target region features may also vary accordingly. To simulate these variations in the scale of target region features during tracking, we utilize different scale factors to perform size scaling on the target region features. Then, we apply the features at the current scale to the corresponding regions in the global features, resulting in enhanced features for the target region. Finally, we combine the enhanced global features and the enhanced target region features with appropriate weighting to obtain the ultimate input features.

Furthermore, to address the issue of filter degradation that may occur during rapid tracking, we incorporate a dual second-order difference term into the model. Specifically, we perform convolution calculations on the differences in features and filter values between adjacent frames within a window of three consecutive frames. In this manner, during the model's learning phase, the filter for the current frame is adjusted based on the past frames' features and filters, preventing sudden noise disturbances such as occlusions and drift from causing filter divergence. The proposed Multi-Scale Enhanced Features Correlation Filters (MSEFCF) tracker makes the following contributions:

- To enhance target recognition and adapt to the continuously changing target scales, we propose a novel feature preprocessing method. Specifically, for the extracted HOG and CN features, we simulate their transformations at different scales, with a particular emphasis on highlighting the target region within the feature area. This contributes to the enhancement of the tracker's performance, making it more suitable for complex tracking environments.
- Differing from other UAV tracking algorithms, we introduce dual second-order difference terms that correspond to features and filters within the tracking model. This implies a more effective fusion of historical information with the current frame data, simultaneously reducing the likelihood of filter divergence, thereby enhancing the robustness of the filter.
- We conduct an extensive series of experiments, testing our proposed method on multiple datasets, including DTB70, UAV123@10fps, and UAVDT. These experimental results offer compelling evidence, confirming the effectiveness and performance improvements achieved by our proposed approach.

The following sections of this paper are organized as follows: Section II introduces relevant DCF methods and deep tracking methods. Section III furnishes an elaborate exposition of our MSEFCF tracker model. In Section IV, we present an extensive analysis of the experimental results. Lastly, in Section V, we encapsulate the key conclusions derived from our research.

II. Related Work

A. DCF-Based Method

The DCF methods have undergone rapid development in the past decade since the introduction of the Minimum Output Sum of Squared Errors (MOSSE) [18] tracker, which initially uses a regression formula for filter training. The Kernel Correlation Filter (KCF) [19] employs circulant matrices to sample the image for acquiring additional learning samples, while utilizing kernel functions to compute the correlation between the target and candidate regions. This approach reduces computational overhead and enhances processing speed. Given the boundary effects resulting from circulant sampling, Spatially Regularized Correlation Filters (SRDCF) [20] introduces a penalty matrix

to apply higher penalty coefficients to the peripheral regions of features. Building upon the SRDCF model, Spatial-Temporal Regularized Correlation Filters (STRCF) [21] introduces a temporal regularization term from a time-based perspective. It utilizes the differences between the filters of the previous frame and the current frame to constrain the update of the current frame's filter. This helps prevent filter degradation. Bilateral Weighted Regression Ranking (BWRR) [22] tracker enhances data fidelity by introducing a bilateral constraint, mitigating loss in filter learning rows and columns. Weighted matrices adaptively penalize substantial data loss, preventing model degradation during learning. The revised approach addresses imbalances in positive and negative samples, converting the original correlation filter regression into a regression-with-ranking problem.

The algorithms described above have achieved excellent tracking results in traditional tracking datasets such as OTB [23] and TC128 [24] at the time. However, with the proliferation of UAV, their performance on UAV tracking datasets often falls short of expectations. This is primarily due to the fact that traditional datasets typically feature relatively simple tracking scenarios with minimal variations in target appearance and motion characteristics. In contrast, UAV tracking datasets present more challenging conditions. In UAV tracking scenarios, targets may undergo frequent changes in scale, making it harder for traditional algorithms to adapt effectively. Additionally, due to the high-speed motion of drones, images can be more susceptible to blurring, further increasing the tracking difficulty. Therefore, the discrepancy in performance can be attributed to the inherent complexity of UAV tracking datasets, characterized by significant scale variations and potential image blurring caused by the fast-moving platform, which traditional algorithms are less equipped to handle.

In recent years, many UAV tracking algorithms have incorporated historical information into the model's learning process to address more challenging scenarios in UAV tracking. Aberrance Repressed Correlation Filters (ARCF) [25] introduces a regularization term involving the difference between the convolution results of the features and filters from the previous frame and the current frame. This regularization term is incorporated to constrain aberrations. Multi-Regularized Correlation Filter (MRCF) [26] constructs second-order difference terms for response maps to enable smooth variations in response. Robust Correlation Filter Learning (RCFL) [27] constructs a continuously weighted dynamic response map to refine the learning of the current frame filter. Compared to the majority of algorithms that employ fixed and singular Gaussian class labels, Exploiting Response Reasoning for Correlation Filters (ReCF) [28] establishes adaptive class labels to address the overfitting issue in filter updates. Robust Spatial-Temporal Correlation Filter Tracker (TRBCF) [29] introduces a novel multi-feature fusion mechanism to emphasize the distinct contributions of feature channels to the model. By extracting authentic background patches to introduce updates to the model, TRBCF enhances the discriminability between the target and the background. Motivated by the inspiration drawn from the utilization of historical information in these UAV algorithms and the observed deficiency in feature preprocessing, we propose the MSEFCF tracker.

B. *Deep Tracking Methods*

The initial deep tracking methods primarily employs the DCF model, but with the substitution of deep features for hand-crafted features. Representative algorithms in this category include FCNT, which utilize the deep network VGG-Net for feature extraction. The advantage of this approach lies in the ability to capture different types of information from various pooling layers.

Another type of deep tracking algorithm is entirely built upon a deep neural network framework. Robust Visual Tracking Using Very Deep Generative Model (RTDG) [30] utilizes a convolutional neural network (CNN) to construct the model, encompassing both feature extraction and a binary classifier network. To enhance tracking performance, they integrate a Generative Adversarial Network (GAN) into the CNN, improving training stage performance through an adversarial learning process. Multi-Expert Visual Tracking (MEVT) [31] employs deep convolutional neural networks to extract features from objects with diverse attributes, constructing them into an ensemble. Each element within the ensemble is utilized for tracking targets across all frames. Ultimately, the highest-scoring element from the ensemble is selected for prediction in each frame. Zhang et al. [32] adopt a U-Net network to remove noise and develop a learnable data augmentation method for object tracking. Scale Equivariance Improves Siamese Tracking (SE-SiamFC) [33] aims to enhance Siamese networks with intrinsic scale equivariance to capture the natural variations of the target. Unsupervised Deep Representation Tracking (LUDT) [34] establishes a robust tracker by exclusively leveraging unlabeled videos for end-to-end learning. This tracker demonstrates the capability to forwardly locate a target object across consecutive frames and retroactively trace it back to its initial position in the first frame. Autoregressive Visual Tracking (ARTrack) [35] introduces a novel self-regressive framework that conceptualizes the tracking task as the interpretation of a coordinate sequence. By progressively estimating the target trajectory, it employs a temporal autoregressive approach to simulate the sequential evolution of the target's motion trajectory. This method is straightforward, simplifying the intricate network architectures of mainstream deep tracking models.

In addition, over the past two years, with the popularity of transformer-based models, many scholars have applied them to the field of visual tracking. Models such as Hierarchical Feature Transformer for Aerial Tracking (HiFT) [36] and Learning Spatio-Temporal Transformer for Visual Tracking (STARK) [37] have been developed to establish global spatio-temporal feature relationships between target objects and background regions, enabling end-to-end search for targets. Such processing eliminates the need for any post-processing steps and enhances the model's capability for target recognition.

Deep learning algorithms have demonstrated significant potential in the field of object tracking, enabling higher tracking precision. However, such performance relies on extensive datasets for pretraining deep neural networks and necessitates costly GPU hardware to provide sufficient computational resources. This drawback is quite evident, as deep learning algorithms typically exhibit lower real-time performance in object

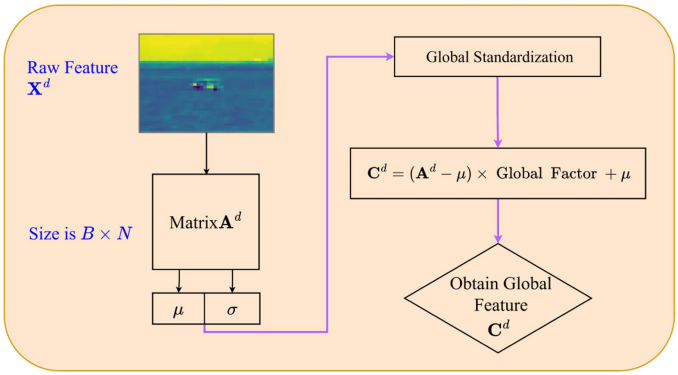


Fig. 1. Perform global processing on the extracted raw features.

tracking, implying that they cannot meet the requirements of applications in which rapid responsiveness is essential, such as real-time UAV tracking.

III. Methodology

A. Baseline Model

The MSEFCF model we propose is built upon the baseline Background-Aware Correlation Filters (BACF) [38] model. To better comprehend our approach, this subsection provides a brief overview of the BACF model.

$$E(\mathbf{F}) = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \mathbf{F}_t^d * \mathbf{P}\mathbf{X}_t^d - \mathbf{Y}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{F}_t^d\right\|_2^2 \quad (1)$$

In the loss function of the BACF model, $\mathbf{X}_t \in \mathbb{R}^{B\times N\times D}$ represents the extracted multi-channel features. There are a total of D channels, and the size of each feature channel is $B \times N$. $\mathbf{Y} \in \mathbb{R}^{B\times N}$ is a pre-defined label matrix composed of Gaussian functions. The central region of the matrix is set to 1, while the remaining regions are set to 0. $\mathbf{F}_t \in \mathbb{R}^{B\times N\times D}$ is the filter that the model aims to solve, and its dimensions are identical to those of the multi-channel feature $\mathbf{X}_t$. Each feature channel corresponds to a dedicated filter. $\mathbf{P}$ is a binary matrix used to crop the central part of features in order to suppress boundary effects. The symbol $*$ denotes convolution operation, and λ represents a hyperparameter.

B. Multi-Scale Enhanced Feature

In this section, we introduce the feature preprocessing method that we have proposed. For each frame in a video sequence, tracking algorithms extract features comprising D channels, denoted as $\mathbf{X} \in \mathbb{R}^{B\times N\times D}$. As illustrated in Fig. 1, the features of the d-th channel $\mathbf{X}^d$ can be regarded as matrix $\mathbf{A}^d$. By calculating the mean μ and standard deviation σ of matrix $\mathbf{A}^d$, we perform global standardization on matrix $\mathbf{A}^d$ using (2), resulting in the generation of matrix $\mathbf{C}^d$. Global Factor is used to control the degree of variation.

$$\mathbf{C}^d = \left(\mathbf{A}^d - \mu\right) \times \text{ Global Factor } + \mu \quad (2)$$

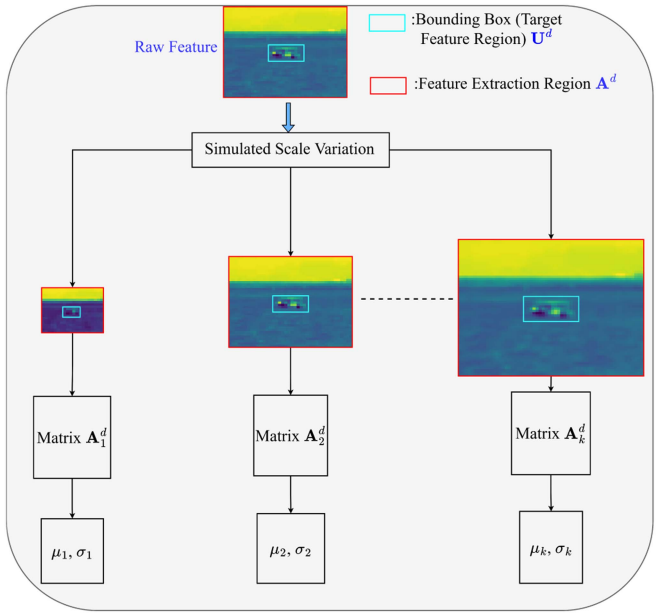


Fig. 2. Simulate various scale variations on the extracted raw features.

In the context of real-time drone tracking, the target frequently undergoes scale changes, so we simulate the transformations of feature matrix $\mathbf{A}^d$ at k different scales. In the context illustrated in Fig. 2, during the tracking process, the algorithm annotates the bounding box, representing the target region, in each frame. The features of the target region are denoted as the matrix $\mathbf{U}^d$. However, the feature extraction step not only captures features from the target region but also extracts features from the background region. Therefore, in each frame, the algorithm ultimately extracts features encompassing both the target and background regions, denoted as the matrix $\mathbf{A}^d$. The raw feature matrix $\mathbf{A}^d$ undergoes scale transformations, denoted as $\mathbf{A}_j^d, j = 1, 2, 3, \ldots, k$. Subsequently, we compute the mean μ_j and standard deviation σ_j of $\mathbf{A}_j^d$.

Next, we consider how to integrate features $\mathbf{A}_j^d$ at different scales. Furthermore, given that the size of the target in drone tracking is typically small while the background region is often large, we aim to emphasize the integration on the target region to accentuate the contrast between the target and the background, thereby enhancing the tracker's discriminative capabilities.

As observed in Fig. 3, the raw feature matrix $\mathbf{A}^d$, after the initial scale transformation, is represented as $\mathbf{A}_1^d$. We extract the target region from $\mathbf{A}_1^d$ and represent it as matrix $\mathbf{U}_1^d$. We enhance $\mathbf{U}_1^d$ using (3) to generate $\mathbf{O}_1^d$. Subsequently, we substitute the matrix $\mathbf{O}_1^d$ for $\mathbf{U}_1^d$ at the corresponding position in the original matrix $\mathbf{A}_1^d$ to create a new feature matrix $\mathbf{L}_1^d$. We then use interpolation to rescale $\mathbf{L}_1^d$ to match the size of $\mathbf{A}^d$, resulting in $\widetilde{\mathbf{L}}_1^d$. Finally, we apply (4) to perform a weighted fusion between $\widetilde{\mathbf{L}}_1^d$ and $\mathbf{C}^d$ to achieve the desired combination.

$$\mathbf{O}_1^d = \left(\mathbf{U}_1^d - \mu_1\right) \cdot \frac{\sigma}{\sigma_1} + \mu_1 \quad (3)$$

$$\mathbf{V}_1^d = \alpha \cdot \mathbf{C}^d + (1-\alpha) \cdot \widetilde{\mathbf{L}}_1^d \quad (4)$$

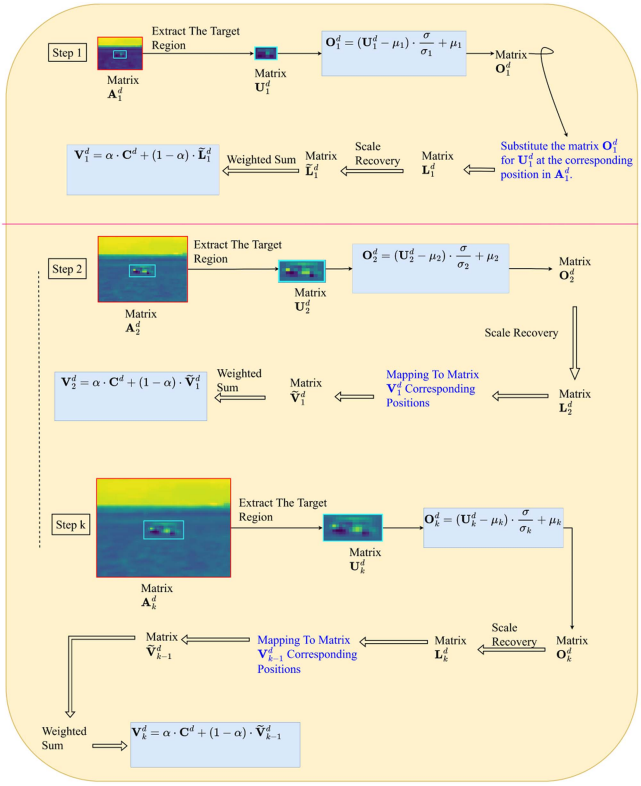


Fig. 3. Enhance the target region within the extracted raw features and perform feature fusion at multiple scales.

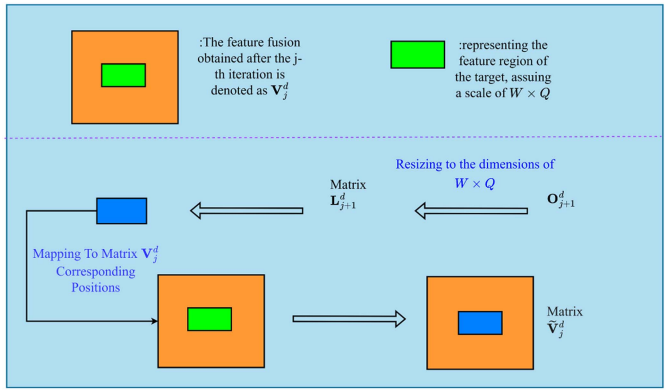


Fig. 4. Mapping Method in Feature Fusion at Multiple Scales: This approach encompasses the restoration of the target matrix's scale under simulated scale variations. It involves the replacement of the target matrix at its original scale position.

Likewise, we perform the extraction of the region of target from the feature matrix $\mathbf{A}_2^d$, which has undergone the second scale transformation. Subsequently, we apply (5) to enhance $\mathbf{U}_2^d$, resulting in the matrix $\mathbf{O}_2^d$. However, considering the scale variations in the features, the critical aspect for successful tracking lies in the tracker's ability to accurately identify the regions where the target features have experienced scale changes. Hence, we directly undertake scale restoration on Matrix $\mathbf{O}_2^d$ to obtain a matrix $\mathbf{L}_2^d$ of the same dimensions as the matrix $\mathbf{U}^d$. We map $\mathbf{L}_2^d$ to the corresponding positions within $\mathbf{V}_1^d$, corresponding to the target feature regions, resulting in the matrix $\widetilde{\mathbf{V}}_1^d$. The specific mapping method is illustrated in Fig. 4. Finally, we combine $\widetilde{\mathbf{V}}_1^d$ with the global feature $\mathbf{C}^d$ through a weighted fusion process to obtain the feature representation $\mathbf{V}_2^d$ by (6).

$$\mathbf{O}_2^d = \left(\mathbf{U}_2^d - \mu_2\right) \cdot \frac{\sigma}{\sigma_2} + \mu_2 \tag{5}$$

$$\mathbf{V}_2^d = \alpha \cdot \mathbf{C}^d + (1-\alpha) \cdot \widetilde{\mathbf{V}}_1^d \tag{6}$$

Subsequently, for the feature matrix $\mathbf{A}_j^d$ that has undergone the j-th scale transformation, we extract the region of interest, denoted as $\mathbf{U}_j^d$. We then enhance $\mathbf{U}_j^d$ using (7), resulting in the matrix $\mathbf{O}_j^d$. Following this, we perform scale restoration directly on $\mathbf{O}_j^d$ to obtain a matrix $\mathbf{L}_j^d$ of the same dimensions as $\mathbf{U}^d$. We map $\mathbf{L}_j^d$ to the corresponding positions within $\mathbf{V}_{j-1}^d$, where the target feature regions are located, producing the matrix $\widetilde{\mathbf{V}}_{j-1}^d$. Finally, we employ (8) to perform a weighted fusion between $\widetilde{\mathbf{V}}_{j-1}^d$ and the global feature $\mathbf{C}^d$ to obtain the feature representation $\mathbf{V}_j^d$.

$$\mathbf{O}_j^d = \left(\mathbf{U}_j^d - \mu_j\right) \cdot \frac{\sigma}{\sigma_j} + \mu_j \tag{7}$$

$$\mathbf{V}_j^d = \alpha \cdot \mathbf{C}^d + (1-\alpha) \cdot \widetilde{\mathbf{V}}_{j-1}^d \tag{8}$$

Until undergoing k scale changes, $\mathbf{V}_k^d$ represents the new features obtained after feature preprocessing, and we denote it with the symbol $\mathbf{M}^d$. We substitute the features $\mathbf{M}^d$ obtained through feature preprocessing for the original features $\mathbf{X}^d$ as input to update the model's filters $\mathbf{F}^d$ during training.

After the multi-scale fusion of multi-channel features, the resulting new feature $\mathbf{M}^d$ contains information about potential scale changes in the target. In practical tracking scenarios, regardless of whether the target scales up or down, the filter updates become more accurate. The majority of UAV tracking algorithms lack adaptability to target scale changes. When the target undergoes scale changes, the filters are unable to learn effectively and promptly for features undergoing significant alterations. This leads to real-time tracking boxes generated by these algorithms not matching the actual size of the target, thereby reducing tracking accuracy and even resulting in tracking failures. Our method enhances the tracker's sensitivity to target scale changes, allowing the filters to adaptively adjust according to target scale variations, thereby improving the tracker's ability for target identification. This enhancement contributes to the robustness of the tracker during the tracking process.

C. Dual Second-Order Difference Regularization

In order to address the issue of filter mutation and degradation during model updates, it has been empirically demonstrated that learning from historical information is effective. Equation (9) below represents the temporal regularization term introduced in the STRCF [21] model. $\mathbf{F}_t$ denotes the filters that require updating at the current time frame t, while $\mathbf{F}_{t-1}$ represents the filters updated in the previous frame. STRCF aims to minimize the difference between $\mathbf{F}_t$ and $\mathbf{F}_{t-1}$ as much as possible to

impose constraints on the model's updates.

$$\|\mathbf{F}_t - \mathbf{F}_{t-1}\|_2^2 \tag{9}$$

Inspired by STRCF, the Context-Based Spatial Via Multi-feature Fusion (CSVMF) [39] suggests that when there is a significant disparity between the filter to be updated in the current frame and the one from the previous frame, the combination of these filters through a first-order difference term may not adequately adapt to the current and previous target states. Therefore, CSVMF introduces the following second-order difference term.

$$\|\Delta\mathbf{F}_t - \Delta\mathbf{F}_{t-1}\|_2^2 \tag{10}$$

where $\Delta\mathbf{F}_t = \mathbf{F}_t - \mathbf{F}_{t-1}$ and $\Delta\mathbf{F}_{t-1} = \mathbf{F}_{t-1} - \mathbf{F}_{t-2}$. $\mathbf{F}_{t-2}$ represents the filters from the two previous frames in the current frame.

Although CSVMF effectively addresses the challenge of how to better integrate filters from different frames, we believe that relying solely on historical filter information to constrain the update of the current frame's filter is not sufficient. While the second-order difference regularization term of the filter effectively captures variations in the filter, feature-based information is equally crucial. This is because the features of the target may undergo significant changes even in adjacent frames. Therefore, we extend the second-order difference term proposed by CSVMF into the following dual second-order difference term:

$$\sum_{d=1}^{\mathrm{D}} \left\| \left(\Delta\mathbf{F}_t^d - \Delta\mathbf{F}_{t-1}^d\right) * \mathbf{P}\left(\Delta\mathbf{X}_t^d - \Delta\mathbf{X}_{t-1}^d\right) \right\|_2^2 \tag{11}$$

In the context of the baseline model BACF, $\mathbf{P}$ represents the cropping matrix proposed. $\Delta\mathbf{X}_t^d = \mathbf{X}_t^d - \mathbf{X}_{t-1}^d$, and $\Delta\mathbf{X}_{t-1}^d = \mathbf{X}_{t-1}^d - \mathbf{X}_{t-2}^d$. Here, $\mathbf{X}_t^d$ represents the d-th feature channel at frame t, $\mathbf{X}_{t-1}^d$ denotes the d-th feature channel at frame $t-1$, and $\mathbf{X}_{t-2}^d$ signifies the d-th feature channel at frame $t-2$.

By adopting this approach, the model not only considers the temporal changes of the filter during the learning process but also takes into account the spatial variations of the features. Furthermore, the second-order difference terms for both the filter and features serve as flexible, adaptive mutual constraints, preventing either term from dominating the learning process and thereby maintaining a balance. Consequently, the tracking performance of the model becomes more robust to variations in target appearance and scene conditions.

Therefore, by replacing the raw features $\mathbf{X}_t^d$ with the multi-scale enhanced features $\mathbf{M}_t^d$, the resulting tracking model can be referred to as the MSEFCF tracker. The specific formula for the loss function is as follows:

$$\begin{aligned}\varepsilon = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \mathbf{P}\mathbf{M}_t^d * \mathbf{F}_t^d - \mathbf{Y}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{F}_t^d\right\|_2^2 \\ + \frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{P}\left(\Delta\mathbf{M}_t^d - \Delta\mathbf{M}_{t-1}^d\right) * \left(\Delta\mathbf{F}_t^d - \Delta\mathbf{F}_{t-1}^d\right)\right\|_2^2\end{aligned} \tag{12}$$

where λ and θ denote hyperparameters.

D. ADMM Solution for Model

In order to simplify the computation in (12), we use $\mathbf{Z}_t^d$ to represent $\mathbf{P}\mathbf{M}_t^d$ and employ $\mathbf{W}$ to denote $\mathbf{P}(\Delta\mathbf{M}_t^d - \Delta\mathbf{M}_{t-1}^d)$. So, (12) can be rewritten as (13).

$$\begin{aligned}\varepsilon = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \mathbf{Z}_t^d * \mathbf{F}_t^d - \mathbf{Y}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{F}_t^d\right\|_2^2 \\ + \frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{W} * \left(\mathbf{F}_t^d - 2\mathbf{F}_{t-1}^d + \mathbf{F}_{t-2}^d\right)\right\|_2^2\end{aligned} \tag{13}$$

Let $\mathbf{R} = 2\mathbf{F}_{t-1}^d - \mathbf{F}_{t-2}^d$. Equation (13) can be further simplified as:

$$\begin{aligned}\varepsilon = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \mathbf{Z}_t^d * \mathbf{F}_t^d - \mathbf{Y}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{F}_t^d\right\|_2^2 \\ + \frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{W} * \left(\mathbf{F}_t^d - \mathbf{R}\right)\right\|_2^2\end{aligned} \tag{14}$$

By introducing an auxiliary variable $\mathbf{G} \in \mathbb{R}^{B\times N\times D}$ that satisfies $\mathbf{F} = \mathbf{G}$, the augmented Lagrangian form of (14) can be expressed as follows:

$$\begin{aligned}\varepsilon = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \mathbf{Z}_t^d * \mathbf{F}_t^d - \mathbf{Y}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{G}_t^d\right\|_2^2 \\ + \frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\mathbf{W} * \left(\mathbf{F}_t^d - \mathbf{R}\right)\right\|_2^2 + \frac{\gamma}{2}\sum_{d=1}^{D}\left\|\mathbf{F}_t^d - \mathbf{G}_t^d\right\|_2^2 \\ + \sum_{d=1}^{D}\mathrm{Tr}\left(\left(\mathbf{F}_t^d - \mathbf{G}_t^d\right)^T \mathbf{S}^d\right)\end{aligned} \tag{15}$$

Let $\mathbf{S}$ be a 3D array with dimensions $B \times N \times D$, for d ranging from 1 to D. This array represents the Lagrange multipliers. Additionally, the symbol γ is employed to denote the step size.

According to the Alternating Direction Method of Multipliers (ADMM), we define $\mathbf{H} = \frac{1}{\gamma}\mathbf{S}$. Additionally, to simplify the computational complexity, we apply Parseval's theorem to transform (15) into the complex domain, resulting in (16). $\cdot$ represents the dot product.

$$\begin{aligned}\varepsilon = \frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \hat{\mathbf{Z}}_t^d \cdot \hat{\mathbf{F}}_t^d - \hat{\mathbf{Y}}\right\|_2^2 + \frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{G}}_t^d\right\|_2^2 \\ + \frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{W}} \cdot \left(\hat{\mathbf{F}}_t^d - \hat{\mathbf{R}}\right)\right\|_2^2 \\ + \frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{F}}_t^d - \hat{\mathbf{G}}_t^d + \hat{\mathbf{H}}_t^d\right\|_2^2\end{aligned} \tag{16}$$

The (16) can be decomposed into a series of sub-problems for iterative solution.

$$\hat{\mathbf{F}}^{(l+1)} = \underset{\hat{\mathbf{F}}}{\mathrm{argmin}}\frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}} \hat{\mathbf{Z}}_t^d \cdot \hat{\mathbf{F}}_t^d - \hat{\mathbf{Y}}\right\|_2^2$$

$$+\frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{W}}\cdot\left(\hat{\mathbf{F}}_t^d-\hat{\mathbf{R}}\right)\right\|_2^2 +\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{F}}_t^d-\hat{\mathbf{G}}_t^d+\hat{\mathbf{H}}_t^d\right\|_2^2 \tag{17}$$

$$\hat{\mathbf{G}}^{(l+1)}=\underset{\hat{\mathbf{G}}}{\operatorname{argmin}}\frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{G}}_t^d\right\|_2^2+\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{F}}_t^d-\hat{\mathbf{G}}_t^d+\hat{\mathbf{H}}_t^d\right\|_2^2 \tag{18}$$

$$\hat{\mathbf{H}}^{(l+1)}=\hat{\mathbf{H}}^{(l)}+\hat{\mathbf{F}}^{(l+1)}-\hat{\mathbf{G}}^{(l+1)} \tag{19}$$

Solving F :

$$J(\hat{\mathbf{F}})=\frac{1}{2}\left\|\sum_{d=1}^{\mathrm{D}}\hat{\mathbf{Z}}_t^d\cdot\hat{\mathbf{F}}_t^d-\hat{\mathbf{Y}}\right\|_2^2 +\frac{\theta}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{W}}\cdot\left(\hat{\mathbf{F}}_t^d-\hat{\mathbf{R}}\right)\right\|_2^2 +\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{F}}_t^d-\hat{\mathbf{G}}_t^d+\hat{\mathbf{H}}_t^d\right\|_2^2 \tag{20}$$

To obtain the closed-form solution for $\hat{\mathbf{F}}$, we set the partial derivatives of (20) with respect to $\hat{\mathbf{F}}$ equal to zero.

$$\hat{\mathbf{F}}=\left(\hat{\mathbf{Z}}_t^T\hat{\mathbf{Z}}_t+\theta\hat{\mathbf{W}}^T\hat{\mathbf{W}}+\gamma\mathbf{I}\right)^{-1}\mathbf{q} \tag{21}$$

where $\mathbf{q}=\hat{\mathbf{Z}}_t\hat{\mathbf{Y}}+\theta\hat{\mathbf{W}}^T\hat{\mathbf{W}}\hat{\mathbf{R}}+\gamma(\hat{\mathbf{G}}-\hat{\mathbf{H}})$.

Solving G :

$$J(\hat{\mathbf{G}})=\frac{\lambda}{2}\sum_{d=1}^{\mathrm{D}}\left\|\hat{\mathbf{G}}_t^d\right\|_2^2+\frac{\gamma}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{F}}_t^d-\hat{\mathbf{G}}_t^d+\hat{\mathbf{H}}_t^d\right\|_2^2 \tag{22}$$

By equating the derivative of (22) with respect to the variable $\hat{\mathbf{G}}$ to zero, we can derive the closed-form solution as follows:

$$\hat{\mathbf{G}}^{(l+1)}=\frac{\gamma\left(\hat{\mathbf{F}}+\hat{\mathbf{H}}\right)}{(\lambda+\gamma)\mathbf{I}} \tag{23}$$

Solving H :

$$\hat{\mathbf{H}}^{(l+1)}=\hat{\mathbf{H}}^{(l)}+\hat{\mathbf{F}}^{(l+1)}-\hat{\mathbf{G}}^{(l+1)} \tag{24}$$

Updating step size :

$$\gamma^{(l+1)}=\min\left(\gamma^{\max},\beta\gamma^{(l)}\right) \tag{25}$$

The parameter γ has an upper limit, denoted as $\gamma^{\max}$, which represents the maximum allowable value. Furthermore, the scaling factor is indicated by β.

Model Update :

$$\hat{\mathbf{M}}_t^{\text{model}}=(1-\eta)\hat{\mathbf{M}}_{t-1}^{\text{model}}+\eta\hat{\mathbf{M}}_t \tag{26}$$

where η denotes the learning rate.

IV. Experiments

A. Datasets

DTB70 [40] is a dataset comprising 70 distinct video sequences, designed primarily for research on motion modeling, particularly in challenging scenarios. The predominant tracked objects in these video sequences are humans and cars, and they are captured using specially crafted camera motion patterns. These videos are sourced from YouTube, offering a wide variety of scenes and target appearances, thus providing a rich diversity of data for tracking algorithm research. UAV123@10fps [41] is a dataset comprising 123 high-definition video sequences captured from low-altitude aerial perspectives. The total number of frames in this dataset exceeds 110,000, with the majority of video sequences containing over 1000 frames. This dataset encompasses a multitude of challenging tracking scenarios, imposing exceptionally high demands on the performance of tracking algorithms. UAVDT [42] encompasses 14 intricate scenarios, totaling approximately 80,000 frames within video sequences. This dataset offers support for tasks related to object detection, as well as single-object and multi-object tracking. In contrast to other unmanned aerial vehicle datasets, it innovatively introduces tracking scenarios involving diverse weather conditions and vehicle categories.

B. Implementation Detail

The MSEFCF tracker proposed in this study is experimentally evaluated on on a laptop featuring an i7-12700H CPU and operates within the MATLAB 2021a environment. The configuration of parameters plays a crucial role in influencing the performance of the tracker. Consequently, this paper sets the number of iterations for the AMDD algorithm to 2, thereby simulating scale variations of 0.5, 1, and 1.5. The initial step size γ is fixed at 0.001, with β set to 10, and the maximum step size $\gamma^{\max}$ capped at 10000. The parameter θ is set to 0.07, and the learning rate η is 0.0298. The regularization parameter, represented by λ, is specified as 0.001.

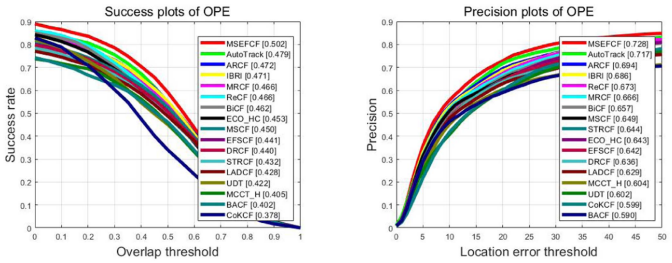
C. Quantitative Analysis of Experimental Results

1) Experimental Comparisons on the DTB70 Dataset: We conduct a comprehensive experimental comparison of our proposed MSEFCF tracker with sixteen other state-of-the-art trackers on the DTB70 dataset. The compared trackers are Visual Tracking With Automatic Spatio-Temporal Regularization (AutoTrack) [43], Disruptor-Aware Interval-Based Response Inconsistency for Correlation Filters (IBRI) [44], ARCF [25], Bidirectional Incongruity-Aware Correlation Filter (BiCF) [45], Enhanced Robust Spatial Feature Selection Correlation Filter (EFSCF) [46], Mutation Sensitive Correlation Filter (MSCF) [47], ReCF [28], MRCF [26], STRCF [21], Efficient Convolution Operators for Tracking (ECO) [48], Learning Adaptive Discriminative Correlation Filters (LADCF) [49], Dual Regularized Correlation Filter (DRCF) [50], BACF [38], Unsupervised Deep Tracking (UDT) [51], Co-trained Kernelized Correlation Filters (CokCF) [52], and Multi-Cue Correlation Filters (MCCT) [53].

TABLE I
BASED SUCCESS PLOT OF THE PROPOSED MSEFCF AND 16 TRACKERS IN DIFFERENT ATTRIBUTES ON THE DTB70 DATABASET

	MB	SOA	BC	OV	OPR	IPR	FCM	DEF	OCC	SV	ARV
MRCF	0.484	0.471	0.399	0.448	0.323	0.429	0.496	0.416	0.418	0.451	0.384
AutoTrack	0.468	0.473	0.394	0.407	0.343	0.454	0.497	0.452	0.415	0.493	0.405
IBRI	0.463	0.466	0.398	0.464	0.315	0.427	0.494	0.431	0.410	0.469	0.408
MSCF	0.460	0.452	0.376	0.414	0.297	0.416	0.471	0.419	0.412	0.450	0.377
ARCF	0.453	0.484	0.377	0.427	0.321	0.429	0.496	0.426	0.446	0.487	0.393
DRCF	0.449	0.453	0.365	0.430	0.263	0.472	0.533	0.392	0.414	0.446	0.377
BiCF	0.448	0.444	0.381	0.389	0.354	0.439	0.472	0.444	0.372	0.482	0.398
EFSCF	0.447	0.443	0.380	0.431	0.254	0.404	0.462	0.389	0.410	0.422	0.353
ReCF	0.441	0.474	0.378	0.364	0.327	0.433	0.481	0.406	0.403	0.496	0.403
STRCF	0.437	0.444	0.341	0.407	0.260	0.391	0.460	0.400	0.401	0.426	0.340
ECO	0.434	0.446	0.349	0.416	0.319	0.410	0.469	0.404	0.431	0.429	0.376
LADCF	0.430	0.458	0.350	0.452	0.323	0.391	0.474	0.443	0.447	0.425	0.315
UDT	0.385	0.387	0.300	0.407	0.326	0.389	0.434	0.407	0.370	0.464	0.414
CoKCF	0.352	0.373	0.347	0.364	0.247	0.369	0.376	0.355	0.393	0.303	0.338
MCCT	0.334	0.399	0.296	0.349	0.243	0.376	0.410	0.354	0.377	0.439	0.334
BACF	0.412	0.411	0.337	0.419	0.203	0.371	0.435	0.302	0.348	0.392	0.273
MSEFCF	0.510⇑	0.527⇑	0.401⇑	0.404	0.370 ⇑	0.469⇑	0.532⇑	0.460⇑	0.472 ⇑	0.508⇑	0.423⇑

The top three methods in each attribue are denoted by different colors: red, green and blue. ⇑ indicates outperformance against baseline.



(a) (b)

Fig. 5. Success plots (a) represent tracking success rates, and precision plots (b) depict a comparative analysis of the tracking accuracy between the proposed MSEFCF tracker and other state-of-the-art trackers on the DTB70 dataset.

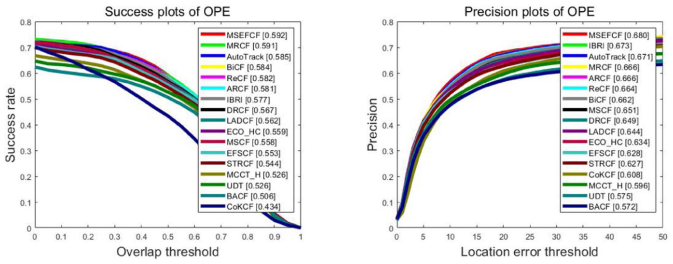
The tracking comparison results for a total of seventeen trackers are illustrated in Fig. 5. In the comparative analysis of tracking success rate and tracking accuracy, MSEFCF consistently demonstrates superior performance. According to the success plots, MSEFCF outperforms the second-ranked AutoTrack by 2.3% and surpasses the baseline model BACF by 10.0%. In the precision plots, MSEFCF demonstrates a 1.1% improvement over AutoTrack and surpasses the baseline model BACF by 13.8%

Furthermore, to offer a more comprehensive assessment of MSEFCF's performance, we have conducted comparisons between MSEFCF and these sixteen trackers across various scenario attributes, based on tracking success plots. These attributes include Motion Blur (MB), Similar Objects Around (SOA), Background Clutter (BC), Out Of View (OV), Out-of-Plane Rotation (OPR), In-Plane Rotation (IPR), Fast Camera Motion (FCM), Deformation (DEF), Occlusion (OCC), Aspect Ratio Variation (ARV), and Scale Variation (SV). As demonstrated in Table I, among the eleven attributes, except for the attribute OV, MSEFCF outperforms all others trackers in terms of performance across the remaining ten attributes, securing the top position. This indicates that MSEFCF exhibits excellent robustness and adaptability across various complex scenarios.

The abbreviation OV in the context of attributes stands for Out Of View, signifying scenarios where the target moves beyond the camera's field of view. During such instances, not only does the camera lose sight of the target, but there is also a complete absence of target information throughout the entire frame. This situation may persist for a few frames to several tens of frames. Consequently, real-time generation of the response map becomes inaccurate. This inaccuracy leads to imprecise localization of the tracking box, and the size of the tracking box may undergo unconstrained expansion or contraction. Our proposed approach relies on enhancing information based on the target box to simulate scale changes. Therefore, the efficacy of this method is significantly compromised in situations where the target is completely lost.

2) Experimental Comparisons on the UAV123@10fps Dataset: We conduct experimental comparisons between MSEFCF and the same sixteen tracking methods. These tracking methods include MRCF [26], AutoTrack [43], IBRI [44], MSCF [47], ARCF [25], DRCF [50], BiCF [45], EFSCF [46], ReCF [28], STRCF [21], ECO [48], LADCF [49], BACF [38], UDT [51], CokCF [52], and MCCT [53].

From Fig. 6, we can see that MSEFCF consistently achieves first place in both tracking success and tracking accuracy comparisons. In the success plots, MSEFCF surpasses the second-ranked MRCF by 0.1% and demonstrates a significant



(a) (b)

Fig. 6. Success plots (a) and precision plots (b) for the proposed MSEFCF tracker and sixteen other trackers on the UAV123@10fps database.

TABLE II
PERFORMANCE EVALUATION OF MSEFCF AND OTHER STATE-OF-THE-ART DEEP TRACKERS ON THE UAVDT BENCHMARK

Tracker	Venue	Prec	Succ	FPS
HDT	16'CVPR	0.596	0.303	9.0
ADNet	17'CVPR	0.683	0.429	7.5
DeepSTRCF	18'CVPR	0.667	0.437	6.8
MCCT	18'CVPR	0.671	0.437	7.9
TADT	19'CVPR	0.677	0.431	32.3
UDT	19'CVPR	0.674	0.442	73.3
UDT+	19'CVPR	0.696	0.415	56.9
MN_ECO	20'ACM	0.691	0.435	30.6
MN_MDNet	20'ACM	0.672	0.440	30.6
fECO	20'TIP	0.699	0.415	20.6
LUDT	21'JJCV	0.631	0.418	78.8
LUDT+	21'IJCV	0.701	0.406	59.4
MEVT	21'IS	0.691	0.448	3
SE-SiamFC	21'WACA	0.626	0.363	5.6
HiFT	21'ICCV	0.641	0.468	-
STARK-ST101	21'ICCV	0.704	0.469	30.2
BSTCF	23'AI	0.685	0.441	19
RTDG	23'JBD	0.728	0.458	3.2
MSEFCF	Ours	0.733	0.448	30

Trackers in each attribute are highlighted with different colors: red, green and blue, representing the top three methods.

enhancement of 8.6% when compared to the baseline. In the precision plots, MSEFCF exhibits a superiority of 0.7% over the second-ranked IBRI. Similarly, MSEFCF significantly outperforms the baseline, with an improvement of 10.8%.

To assess the tracking robustness of MSEFCF, we conduct performance comparisons between MSEFCF and these sixteen trackers across various scenario attributes, based on tracking success plots and precision plots. As depicted in Fig. 7, it becomes clear that MSEFCF exhibits outstanding performance across attributes such as Camera Motion (CM), Fast Motion (FM), Aspect Ratio Change (ARC), and Scale Variation (SV). Given that our proposed method is primarily designed to tackle fast motion in UAV tracking and alterations in the target's scale during motion, MSEFCF consistently delivers remarkable results in these attributes, serving as strong evidence for the efficacy of our approach.

3) Experimental Comparison With Deep Tracking Models: We conduct an evaluation on the UAVDT dataset, comparing MSEFCF with 18 state-of-the-art deep learning-based trackers. These deep learning methods including Hedged Deep Tracking (HDT) [54], Action-Decision Networks (ADNet) [55], DeepSTRCF [21], MCCT [53], Target-Aware Deep Tracking(TADT) [56], UDT [51], UDT+ [51], Online Filtering Training Samples (MN_ECO) [57], MN_MDNet [57], Real-time Correlation Tracking via Joint Model Compression and Transfer (fECO) [58], LUDT [34], LUDT+ [34], Learning Background-Aware and Spatial-Yemporal Regularized Correlation Filters (BSTCF) [59], MEVT [31], SE-SiamFC [33], RTDG [30], HiFT [36] and STARK-ST101 [37].

According to the tracking results provided in Table II, our proposed MSEFCF tracker outperforms the second-ranked deep learning algorithm RTDG by 0.5% in terms of tracking success rate. While MSEFCF falls just short of the top three in tracking accuracy, it is crucial to note that MSEFCF achieves outstanding tracking performance and satisfactory tracking speed solely relying on cost-effective CPUs, without the need for expensive computing equipment. In contrast, most deep learning tracking methods struggle with speed due to the significant time required for model training, rendering them suboptimal in speed performance and often inadequate for real-time tracking demands.

4) Ablation Study: To validate the efficacy of our proposed method in enhancing the practical tracking performance of the tracker, we conduct ablation experiments on the MSEFCF tracker using two datasets: DTB70 and UAV123@10fps.

From Table III, it can be observed that when the features used in the Baseline are replaced with Multi-Scale Enhanced Features (MSEF) generated after feature preprocessing, the tracking success rates for the DTB70 and UAV123@10fps datasets improved by 8.9% and 7.7%, respectively, while the tracking accuracy improved by 13.1% and 10.4%, respectively. When the Baseline augmented with Dual Second-Order Difference (DSOD), the tracker exhibit an improvement in tracking accuracy by 8.4% and 6.8% for the DTB70 and UAV123@10fps datasets, respectively, with a further enhancement in tracking precision by 11.7% and 10.3%, respectively.

5) Parameter Analysis: Using the UAV123@10fps dataset, we perform a sensitivity analysis on the model's crucial parameter θ, as it controls the model's learning of the dual second-order difference regularization term.

As observed from Fig. 8, it can be seen that when the value of θ falls within the range of 0.01 to 0.07, the tracker's success rate and accuracy exhibit a gradual upward trend, reaching their peak at $\theta = 0.07$. In the range of θ values from 0.07 to 0.11, although the tracker's success rate and accuracy experience fluctuations, they demonstrate a gradual decline. Therefore, it is imperative to avoid setting θ too large, in order to prevent the filter from learning excessive historical information, which could lead to overfitting of the model.

6) Real-Time Tracking Analysis: Because the primary objective of our proposed method is to enhance the tracker's ability to better recognize small-sized targets and adapt more effectively to scale variations during fast target motion, we conduct real-time tracking comparisons between the MSEFCF tracker and existing methods such as MRCF, BACF, ARCF, and ReCF in corresponding scenarios.

Fig. 9(a) illustrates the real-time performance of five trackers in tracking the video sequence Basketball_1. The target in this

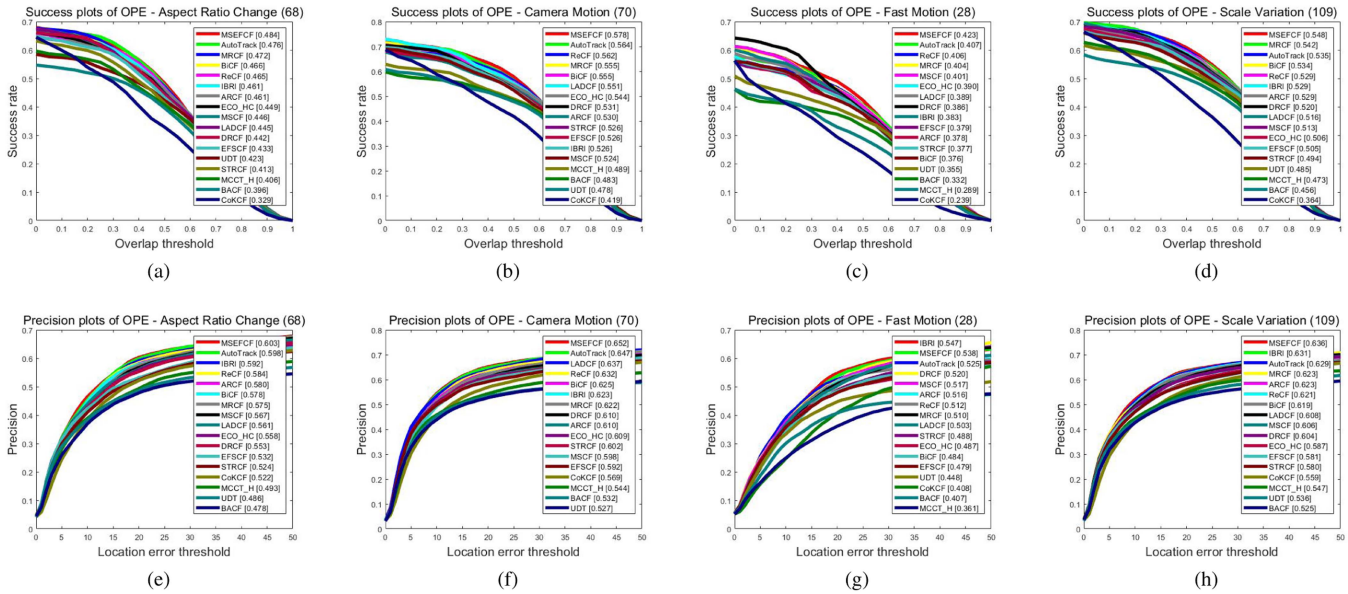


Fig. 7. Contrast the performance of MSEFCF against excellent trackers, particularly in challenging attributes.

TABLE III
ABLATION EXPERIMENTS BASED ON BASELINE

Tracker	Module		DTB70		UAV123@10fps		Average	
	MSEF	DSOD	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.
Baseline			0.402	0.590	0.506	0.572	0.454	0.581
Baseline+MSEF	✓		0.491	0.721	0.583	0.676	0.537	0.698
Baseline+DSOD		✓	0.486	0.707	0.574	0.675	0.530	0.691
MSEFCF	✓	✓	0.502	0.728	0.592	0.680	0.547	0.704

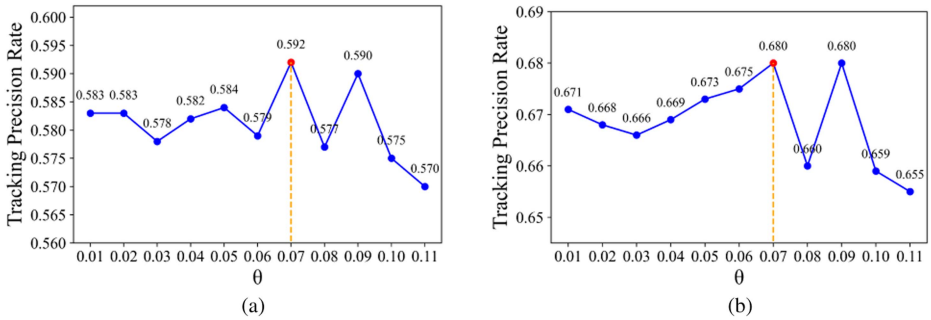


Fig. 8. (a) shows the impact of different θ values on tracking success rate and (b) demonstrates the influence of varied θ values on tracking precision rate.

sequence is of small size and exhibits attributes such as changing lighting conditions and interference from a similar background. When the target enters a crowd, all trackers, except our MSEFCF tracker, lose track of the target. Furthermore, from a quantitative perspective, we analyze the real-time variation of tracking accuracy for MSEFCF and the baseline method BACF during the tracking of the video sequence Basketball_1. As seen in Fig. 10, MSEFCF maintains a relatively stable tracking accuracy throughout the tracking process. In contrast, BACF rapidly loses the target's track early in the tracking due to occlusion, leading to a sharp drop in tracking accuracy to zero.

Fig. 9(b) illustrates the real-time performance of five trackers in tracking the video sequence Car4_1. During the tracking process, the target gradually reduces in size, causing the other four trackers to lose the target due to inadequate feature learning. Furthermore, from a quantitative perspective, we analyze the real-time variation in tracking accuracy for MSEFCF and the baseline method BACF during the tracking of the Car4_1 video sequence. As shown in Fig. 11, MSEFCF maintains a consistently stable real-time tracking accuracy, while BACF, although successful in continuous tracking, ultimately loses the target towards the end of the tracking, demonstrating the excellent robustness performance of the MSEFCF tracker.

Fig. 9(c) presents the real-time tracking performance of five trackers in a scenario where one UAV is tracking another while flying at high speed. It is evident that during the tracking process, the unmanned aircraft undergoes a 360-degree flip, during which BACF, ARCF, and ReCF lose track of the target. When the

(a) Basketball_1

(b) Car4_1

(c) ChasingDrones_1

(d) RaceCar1_1

(e) SnowBoarding4_1

Fig. 9. Performance of the MSEFCF tracker compared to four other trackers in real-time tracking.

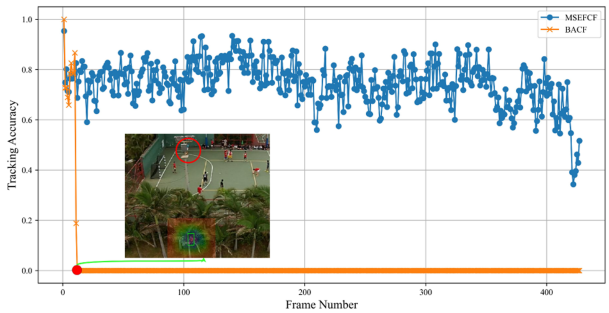


Fig. 10. Real-time tracking accuracy for the video sequence Basketball_1.

unmanned aircraft returns to its initial flight state, MRCF also loses track of the target. From a quantitative perspective, we analyze the real-time variation in tracking accuracy for MSEFCF and the baseline method BACF during the tracking of the ChasingDrones_1 video sequence. As shown in Fig. 12, MSEFCF's tracking accuracy experiences fluctuations during the UAV's flip, but it quickly adapts. In contrast, during the high-speed flip of the UAV, the camera lens experiences blurriness, causing BACF to lose track of the target. This contrast highlights the

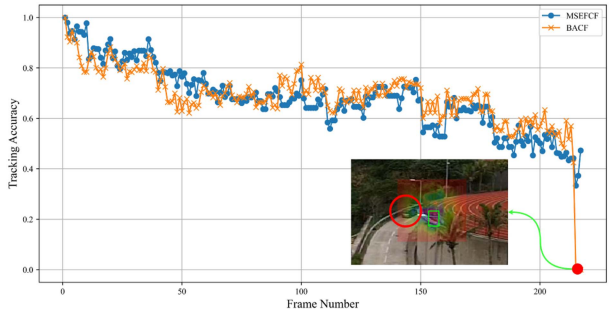


Fig. 11. Real-time tracking accuracy for the video sequence Car4_1.

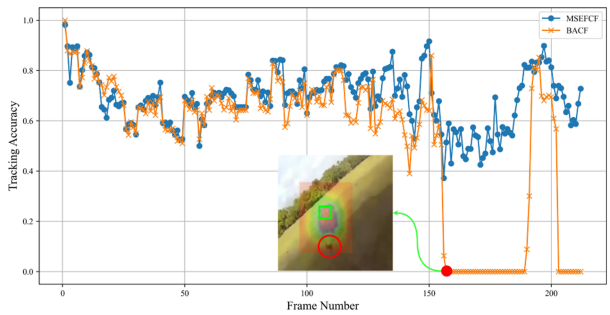


Fig. 12. Real-time tracking accuracy for the video sequence ChasingDrones_1.

advantage of trackers that have learned enhanced target features. Even when the target is of a small size, the filters can still effectively engage in adaptive learning.

Fig. 9(d) illustrates the tracking scenario where, in the first picture, the target is partially occluded by an obstacle. Subsequently, ReCF loses track of the target due to the occlusion. In the following tracking frames, the target is occluded for a second time, after which only MSEFCF successfully maintains tracking. This can be attributed to the inclusion of dual second-order difference terms in our model. By simultaneously learning historical features and filters, when the target is occluded in the current frame, the filters of the current frame do not deviate solely due to the incorrect learning of the current frame's features. Instead, through the fusion learning of historical features and filters, the filters of the current frame can effectively adapt and correct themselves. From a quantitative perspective, we analyze the real-time variation in tracking accuracy for MSEFCF and the baseline method BACF during the tracking of the RaceCar1_1 video sequence. As shown in Fig. 13, MSEFCF's tracking accuracy experiences significant fluctuations when the target encounters occlusion, but it exhibits exceptional adaptive adjustment capabilities. While BACF is able to maintain successful tracking after the first occlusion, it still loses track of the target when the second occlusion occurs.

According to Fig. 9(e), it is evident that the target undergoes significant scale variations. During these scale changes, MRCF, BACF, and ARCF lose track of the target one after the other. Although ReCF manages to track the target, it struggles to adapt well to the continuous scale variations, as indicated by its target bounding box. Our proposed MSEFCF tracker, on the other

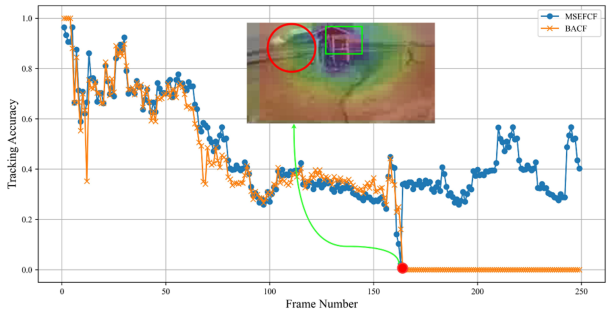


Fig. 13. Real-time tracking accuracy for the video sequence RaceCar1_1.

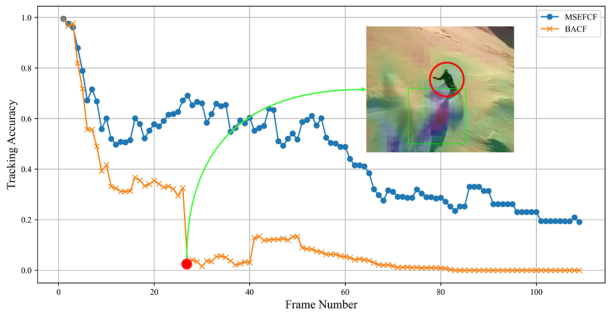


Fig. 14. Real-time tracking accuracy for the video sequence Car4_1.

hand, exhibits adaptive changes in the tracking box in response to the target's scale variations, showcasing the tracker's excellent scale adaptation capability through the learning of multi-scale enhanced features. In the quantitative analysis, Fig. 14 demonstrates the outstanding tracking performance of MSEFCF in such a challenging scenario.

V. Conclusion

In this paper, we propose a novel feature preprocessing method to address the challenges in target tracking for small-sized objects with frequent scale variations in UAV tracking. For the extracted multi-channel HOG+CN features, we not only simulate the changes in feature channels at different scales but also fuse features at different scales on the basis of considering feature global balance and specifically enhancing the target feature region. Through such processing, the tracker improves its target recognition capability. Additionally, to address the issues of degradation and variation in filters during the update process, we introduce dual second-order difference regularization terms related to learning historical features and filters. These changes facilitate the better fusion of historical features and filters in learning. On three authoritative datasets, namely DTB70, UAV123@10fps, and UAVDT, we conduct extensive experiments to compare our MSEFCF tracker with other state-of-the-art trackers, and the experimental results demonstrate the effectiveness of our proposed method.

Certainly, our method has its limitations. When the target moves beyond the frame during the tracking process, i.e., in Out-of-View (OV) scenarios, the performance of MSEFCF may exhibit instability. This is attributed to the model extracting inaccurate information for scale variations when the tracking box loses the target, potentially causing distortions in filter updates. If the frames with the lost target are of short duration, the model can be corrected using the dual second-order difference term. However, if the frames with the lost target extend beyond the correction range of the dual second-order difference term, filter updates may deviate in the wrong direction, leading to tracking failure. Therefore, in subsequent research, we will dedicate efforts to addressing this issue.

Furthermore, as autonomous driving technology continues to advance, there is a growing demand in the industry for increased precision in path tracking [60], [61], multi-task coordination of UAV [62] and visual control methods for tracking [63], [64]. Meeting these requirements is only possible through the development of advanced tracking algorithms. Therefore, the exploration of integrating ground traffic tracking and unmanned aerial vehicle tracking systems to enhance the convenience of the societal transportation system holds significant potential for further development. In addition, in future work, we plan to leverage insights from recent advancements in algorithm optimization [65], [66] to further decrease the computational complexity of the model, thereby achieving a faster tracking speed for the tracker.

References

[1] B. Li et al., "Adaptive pure pursuit: A real-time path planner using tracking controllers to plan safe and kinematically feasible paths," *IEEE Trans. Intell. Veh.*, vol. 8, no. 9, pp. 4155–4168, Sep. 2023.

[2] X. Zhou, Z. Wang, H. Shen, and J. Wang, "Robust adaptive path-tracking control of autonomous ground vehicles with considerations of steering system backlash," *IEEE Trans. Intell. Veh.*, vol. 7, no. 2, pp. 315–325, Jun. 2022.

[3] Z. Liu et al., "Robust multi-drone multi-target tracking to resolve target occlusion: A benchmark," *IEEE Trans. Multimedia*, vol. 25, pp. 1462–1476, 2023.

[4] H. Chen, Y. Liu, B. Zhao, C. Hu, and X. Zhang, "Vision-based real-time online vulnerable traffic participants trajectory prediction for autonomous vehicle," *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2110–2122, Mar. 2023.

[5] J. Sun, J. Shen, X. Wang, Z. Mao, and J. Ren, "Bi-Unet: A dual stream network for real-time highway surface segmentation," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1549–1563, Feb. 2023.

[6] P. Wang, C. Zhu, X. Wang, Z. Zhou, G. Wang, and Y. Wang, "Inferring intersection traffic patterns with sparse video surveillance information: An ST-GAN method," *IEEE Trans. Veh. Technol.*, vol. 71, no. 9, pp. 9840–9852, Sep. 2022.

[7] P. Wang, D. Li, Y. Zhang, and X. Chen, "UAV-assisted vehicular communication system optimization with aerial base station and intelligent reflecting surface," *IEEE Trans. Intell. Veh.*, early access, Oct. 13, 2023, doi: 10.1109/TIV.2023.3324385.

[8] M. M. Karim, Z. Yin, and R. Qin, "An attention-guided multistream feature fusion network for early localization of risky traffic agents in driving videos," *IEEE Trans. Intell. Veh.*, early access, May 11, 2023, doi: 10.1109/TIV.2023.3275543.

[9] J. Xu, Y. Zhao, H. Li, and P. Zhang, "An image reconstruction model regularized by edge-preserving diffusion and smoothing for limited-angle computed tomography," *Inverse Problems*, vol. 35, no. 8, 2019, Art. no. 085004.

[10] B. Gong, B. Schullcke, S. Krueger-Ziolek, U. Mueller-Lisse, and K. Moeller, "Sparse regularization for EIT reconstruction incorporating structural information derived from medical imaging," *Physiol. Meas.*, vol. 37, no. 6, 2016, Art. no. 843.

[11] Z.-R. Lai, L. Tan, X. Wu, and L. Fang, "Loss control with rank-one covariance estimate for short-term portfolio optimization," *J. Mach. Learn. Res.*, vol. 21, no. 1, pp. 3815–3851, 2020.

[12] Z.-R. Lai and H. Yang, "A survey on gaps between mean-variance approach and exponential growth rate approach for portfolio optimization," *ACM Comput. Surv.*, vol. 55, no. 2, pp. 1–36, 2022.

[13] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3119–3127.

[14] C. Shu and Y. Luo, "Multi-modal feature constraint based tightly coupled monocular visual-LiDAR odometry and mapping," *IEEE Trans. Intell. Veh.*, vol. 8, no. 5, pp. 3384–3393, May 2023.

[15] C. Sitaula, J. Aryal, and A. Bhattacharya, "A novel multiscale attention feature extraction block for aerial remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 2504705.

[16] T. Zhou, J. Chen, Y. Shi, K. Jiang, M. Yang, and D. Yang, "Bridging the view disparity between radar and camera features for multi-modal fusion 3D object detection," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1523–1535, Feb. 2023.

[17] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 850–865.

[18] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2544–2550.

[19] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[20] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4310–4318.

[21] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4904–4913.

[22] H. Zhu, H. Peng, G. Xu, L. Deng, Y. Cheng, and A. Song, "Bilateral weighted regression ranking model with spatial-temporal correlation filter for visual tracking," *IEEE Trans. Multimedia*, vol. 24, pp. 2098–2111, 2022.

[23] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.

[24] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.

[25] Z. Huang, C. Fu, Y. Li, F. Lin, and P. Lu, "Learning aberrance repressed correlation filters for real-time UAV tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 2891–2900.

[26] 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.

[27] Y. Zhang, Y.-F. Yu, L. Chen, and W. Ding, "Robust correlation filter learning with continuously weighted dynamic response for UAV visual tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4705814.

[28] F. Lin, C. Fu, Y. He, W. Xiong, and F. Li, "ReCF: Exploiting response reasoning for correlation filters in real-time UAV tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 10469–10480, Aug. 2022.

[29] L. Chen and Y. Liu, "A robust spatial-temporal correlation filter tracker for efficient UAV visual tracking," *Appl. Intell.*, vol. 53, no. 4, pp. 4415–4430, 2023.

[30] E. R. AlBasiouny, A.-F. Attia, H. E. Abdelmunim, and H. M. Abbas, "Robust visual tracking using very deep generative model," *J. Big Data*, vol. 10, no. 1, pp. 1–26, 2023.

[31] S. Moorthy and Y. H. Joo, "Multi-expert visual tracking using hierarchical convolutional feature fusion via contextual information," *Inf. Sci.*, vol. 546, pp. 996–1013, 2021.

[32] Z. Zhang, W. Xue, Q. Liu, K. Zhang, and S. Chen, "Learnable diffusion-based amplitude feature augmentation for object tracking in intelligent vehicles," *IEEE Trans. Intell. Veh.*, early access, Oct. 25, 2023, doi: 10.1109/TIV.2023.3327501.

[33] I. Sosnovik, A. Moskalev, and A. W. Smeulders, "Scale equivariance improves siamese tracking," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2021, pp. 2765–2774.

[34] N. Wang, W. Zhou, Y. Song, C. Ma, W. Liu, and H. Li, "Unsupervised deep representation learning for real-time tracking," *Int. J. Comput. Vis.*, vol. 129, pp. 400–418, 2021.

[35] X. Wei, Y. Bai, Y. Zheng, D. Shi, and Y. Gong, "Autoregressive visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 9697–9706.

[36] Z. Cao, C. Fu, J. Ye, B. Li, and Y. Li, "HiFT: Hierarchical feature transformer for aerial tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15437–15446.

[37] 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. 10428–10437.

[38] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1144–1152.

[39] D. Elayaperumal and Y. H. Joo, "Robust visual object tracking using context-based spatial variation via multi-feature fusion," *Inf. Sci.*, vol. 577, pp. 467–482, 2021.

[40] S. Li and D.-Y. Yeung, "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 4140–4146.

[41] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 445–461.

[42] D. Du et al., "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 370–386.

[43] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11920–11929.

[44] C. Fu, J. Ye, J. Xu, Y. He, and F. Lin, "Disruptor-aware interval-based response inconsistency for correlation filters in real-time aerial tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6301–6313, Aug. 2021.

[45] F. Lin, C. Fu, Y. He, F. Guo, and Q. Tang, "BiCF: Learning bidirectional incongruity-aware correlation filter for efficient UAV object tracking," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 2365–2371.

[46] J. Wen, H. Chu, Z. Lai, T. Xu, and L. Shen, "Enhanced robust spatial feature selection and correlation filter learning for UAV tracking," *Neural Netw.*, vol. 161, pp. 39–54, 2023.

[47] G. Zheng, C. Fu, J. Ye, F. Lin, and F. Ding, "Mutation sensitive correlation filter for real-time UAV tracking with adaptive hybrid label," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 503–509.

[48] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6931–6939.

[49] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5596–5609, Nov. 2019.

[50] C. Fu, J. Xu, F. Lin, F. Guo, T. Liu, and Z. Zhang, "Object saliency-aware dual regularized correlation filter for real-time aerial tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8940–8951, Dec. 2020.

[51] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1308–1317.

[52] L. Zhang and P. N. Suganthan, "Robust visual tracking via co-trained kernelized correlation filters," *Pattern Recognit.*, vol. 69, pp. 82–93, 2017.

[53] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4844–4853.

[54] Y. Qi et al., "Hedged deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4303–4311.

[55] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Y. Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1349–1358.

[56] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1369–1378.

[57] J. Zhao, K. Dai, D. Wang, H. Lu, and X. Yang, "Online filtering training samples for robust visual tracking," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 1488–1496.

[58] N. Wang, W. Zhou, Y. Song, C. Ma, and H. Li, "Real-time correlation tracking via joint model compression and transfer," *IEEE Trans. Image Process.*, vol. 29, pp. 6123–6135, 2020.

[59] J. Zhang, Y. He, W. Feng, J. Wang, and N. N. Xiong, "Learning background-aware and spatial-temporal regularized correlation filters for visual tracking," *Appl. Intell.*, vol. 53, no. 7, pp. 7697–7712, 2023.

[60] G. Chen, X. Zhao, Z. Gao, and M. Hua, "Dynamic drifting control for general path tracking of autonomous vehicles," *IEEE Trans. Intell. Veh.*, vol. 8, no. 3, pp. 2527–2537, Mar. 2023.

[61] Z. Sun, J. Zou, D. He, and W. Zhu, "Path-tracking control for autonomous vehicles using double-hidden-layer output feedback neural network fast nonsingular terminal sliding mode," *Neural Comput. Appl.*, vol. 34, no. 7, pp. 5135–5150, 2022.

[62] Z. Han, M. Chen, S. Shao, H. Zhu, and Q. Wu, "Cooperative multi-task assignment of unmanned autonomous helicopters based on hybrid enhanced learning ABC algorithm," *IEEE Trans. Intell. Veh.*, early access, Sep. 25, 2023, doi: 10.1109/TIV.2023.3319110.
[63] H. Zhong, Y. Wang, Z. Miao, L. Li, S. Fan, and H. Zhang, "A homography-based visual servo control approach for an underactuated unmanned aerial vehicle in GPS-denied environments," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1119–1129, Feb. 2023.
[64] K. Samal, H. Kumawat, P. Saha, M. Wolf, and S. Mukhopadhyay, "Task-driven RGB-lidar fusion for object tracking in resource-efficient autonomous system," *IEEE Trans. Intell. Veh.*, vol. 7, no. 1, pp. 102–112, Mar. 2022.
[65] Z.-R. Lai, P.-Y. Yang, L. Fang, and X. Wu, "Short-term sparse portfolio optimization based on alternating direction method of multipliers," *J. Mach. Learn. Res.*, vol. 19, no. 1, pp. 2547–2574, 2018.
[66] Z.-R. Lai, C. Li, X. Wu, Q. Guan, and L. Fang, "Multitrend conditional value at risk for portfolio optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 23, 2022, doi: 10.1109/TNNLS.2022.3183891.

Yu-Feng Yu (Member, IEEE) received the Ph.D. degree in statistics from Sun Yat-Sen University, Guangzhou, China, in 2017. He is currently an Associate Professor with the Department of statistics, Guangzhou University, Guangzhou, China. From 2016 to 2017, he was a Visiting Scholar with the Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV, USA. From 2017 to 2018, he was a Senior Research Associate with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong. His research interests include image processing, statistical optimization, pattern recognition, machine learning, and computer vision.

Yang Zhang received the B.Sc. degree in applied mathematics from Guangzhou University, Guangzhou, China, where he is currently working toward the M.Sc. degree in statistics. His research interests include visual tracking and machine learning.

Long Chen (Senior Member, IEEE) received the B.S. degree in information sciences from Peking University, Beijing, China, in 2000, the M.S.E. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2003, the M.S. degree in computer engineering from the University of Alberta, Edmonton, AB, Canada, in 2005, and the Ph.D. degree in electrical engineering from The University of Texas at San Antonio, San Antonio, TX, USA, in 2010. From 2010 to 2011, he was a Postdoctoral Fellow with The University of Texas at San Antonio. He is currently an Associate Professor with the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include computational intelligence, Bayesian methods, and other machine learning techniques and their applications.

Pengfei Ge received the Ph.D. degree in statistics from the School of Mathematics, Sun Yat-sen University, Guangzhou, China, in 2020. He is currently an Assistant Professor with the School of Mathematics and Systems Science, Guangdong Polytechnic Normal University, Guangzhou, China. His research interests include pattern recognition and machine learning.

C. L. Philip Chen (Fellow, IEEE) received the M.S. degree in electrical engineering from the University of Michigan at Ann Arbor, Ann Arbor, MI, USA, in 1985, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1988. His research interests include cybernetics, systems, and computational intelligence. He is currently the Chair Professor and Dean of the College of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Being a Program Evaluator of the Accreditation Board of Engineering and Technology Education Baltimore, MD, USA, for computer engineering, electrical engineering, and software engineering programs, he successfully architects the University of Macaus Engineering and Computer Science programs, receiving accreditations from Washington/Seoul Accord through Hong Kong Institute of Engineers (HKIE), of which is considered as his utmost contribution in engineering/computer science education for Macau as the former Dean of the Faculty of Science and Technology. He is a Fellow of AAAS, IAPR, CAA, and HKIE, and a Member of Academia Europaea, European Academy of Sciences and Arts, and International Academy of Systems and Cybernetics Science. He was the recipient of the IEEE Norbert Wiener Award in 2018 for his contribution in systems and cybernetics, and machine learning, and the 2016 Outstanding Electrical and Computer Engineers Award from his alma mater, Purdue University (in 1988), after his graduation. He is also a highly cited Researcher by Clarivate Analytics in 2018 and 2019. He was the IEEE Systems, Man, and Cybernetics Society President from 2012 to 2013, Editor-in-Chief of IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS during 2014–2019, and he is an Associate Editor for IEEE TRANSACTIONS ON AI and IEEE TRANSACTIONS ON FUZZY SYSTEMS. From 2015 to 2017, he was the Chair of TC 9.1 Economic and Business Systems of International Federation of Automatic Control. He is also the Vice President of the Chinese Association of Automation.