Learning Dynamic-Sensitivity Enhanced Correlation Filter With Adaptive Second-Order Difference Spatial Regularization for UAV Tracking

Yu-Feng Yu, *Member, IEEE*, Zhongsen Chen, Yang Zhang, Chuanbin Zhang, and Weiping Ding, *Senior Member, IEEE*

***Abstract*— Discriminative correlation filter (DCF)-based tracking algorithms continue to advance in the field of UAV tracking due to their computational efficiency. The idea of integrating the advantages of historical information and response adjustments into the CF tracking framework is continuously being developed. However, maintaining the stability of mobile video tracking in highly dynamic environments is extremely challenging. This difficulty arises from frequent changes in targets and backgrounds, as well as the stochastic noise generated by the photon-counting process in sensors. In addition, the inconsistent rates of these changes are often overlooked and require further scrutiny. In this paper, we propose a dynamic sensitivity enhanced correlation filter with adaptive second-order difference spatial regularization to address the issue of inconsistent motion rates in dynamic videos. We use the non-local means algorithm to denoise template images before feature extraction, improving the discriminative power of target contours. Then, we incorporate the proposed dynamic-sensitivity error method into CF learning and employ a novel adaptive second-order difference spatial regularization to simultaneously optimize the filter coefficients and spatial regularization weights. This regularization effectively works in synergy with the dynamic-sensitivity error strategy. Furthermore, an additional ADMM optimizer is introduced to derive the solution, thereby improving the convergence and computational efficiency of the algorithm. This algorithm supports the adjustment of filter updates in dynamic environments by balancing consistency with previous filter templates and flexibility to accommodate rapid target changes. By conducting extensive experiments on three challenging UAV tracking databases, we compare the proposed model with existing models. The experimental results demonstrate our superior performance. Code is released at: https://github.com/Johnsonirene/LDECF.**

***Index Terms*— Correlation filters, dynamic sensitivity, UAV tracking, non-local means, second-order difference weights.**

Received 26 August 2024; revised 20 December 2024; accepted 21 January 2025. Date of publication 4 February 2025; date of current version 5 May 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62006056 and Grant U2433216, in part by the Natural Science Foundation of Guangdong Province under Grant 2024A1515012040, in part by the Science and Technology Planning Project of Guangzhou under Grant 2024A03J0401, in part by the National Key Research and Development Plan of China under Grant 2024YFE0202700, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20231337, and in part by the Innovation Project of Guangdong Province University under Grant 2024KQNCX023. The Associate Editor for this article was Y. Wiseman. *(Corresponding author: Weiping Ding.)*

Yu-Feng Yu and Zhongsen Chen are with the Department of Statistics, Guangzhou University, Guangzhou 511370, China (e-mail: yuyufeng220@163.com; gdcfqzl@163.com).

Yang Zhang is with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yc47958@um.edu.mo).

Chuanbin Zhang is with the School of Computer Science and Software, Zhaoqing University, Zhaoqing 526061, China (e-mail: chuanbinzhang@hotmail.com).

Weiping Ding is with the School of Artificial Intelligence and Computer Science, Nantong University, Nantong 226019, China, and also with the Faculty of Data Science, City University of Macau, Macau, China (e-mail: dwp9988@163.com).

Digital Object Identifier 10.1109/TITS.2025.3533953

I. Introduction

UNNAMED aerial vehicles (UAVs) equipped with visual tracking systems harness this technology for a variety of applications, such as monitoring, remote sensing [1] and so on. These systems facilitate autonomous object detection and tracking by first identifying a target and then following it consistently through video frames. They leverage data from the initial frames to continuously update and refine the location of the object. This functionality not only enhances the UAVs' operational efficiency but also broadens their utility in fields such as traffic congestion monitoring [2], autonomous driving vehicles [3], aerial cinematography [4] and human-computer interaction [5] thereby expanding their impact across multiple domains.

In recent years, although advanced visual tracking methods have been continuously proposed, there remains considerable room for development due to persistent tracking challenges. While UAV visual tracking builds upon general visual tracking principles, it presents unique technical demands and application challenges, such as highly dynamic backgrounds, drastic viewpoint changes, and uncertain flight conditions. Trackers based on deep learning and discriminative correlation filters remain research hotspots. With advancements in mobile onboard computing capabilities, deep learning-based UAV tracking methods have become increasingly popular. The Siamese network-based SiamFC [6] pioneers the first paradigm in deep learning object tracking methods, addressing object tracking through a similarity learning approach. In addition, some deep learning-based trackers employ the tracking-learning-detection (TLD) [7] method to follow targets in video streams. The TLD approach enhances the robustness and accuracy of the tracking system by simultaneously performing tracking and detection, with mutual learning between these processes. However, in practical UAV applications, the complex post-processing and hyperparameter tuning of deep

learning frameworks can significantly affect onboard operational speed. While CNN-based trackers are renowned for their high accuracy, they often require substantial computational resources, making them less suitable for scenarios demanding high efficiency. To mitigate inefficiencies, lightweight CNN-based trackers have been developed for UAV tracking. These trackers employ filter pruning [8], [9], [10], [11] to reduce parameters, thereby significantly enhancing both accuracy and efficiency. In recent years, transformers have demonstrated immense potential in the visual domain [12], with a range of vision transformer-based methods achieving impressive performance on public UAV tracking benchmarks. However, the accuracy of these methods heavily depends on large training datasets, as extensive training is essential for optimal results. Consequently, discriminative correlation filter (DCF-based) methods remain a trusted choice in resource-constrained environments, where they shine by balancing efficiency with practicality. Correlation filter trackers based on deep features combine traditional correlation filter tracking techniques with deep learning technology. This method first utilizes deep networks to extract image features and then employs correlation filters for real-time tracking. This approach have demonstrated superior performance on multiple standard tracking datasets, yet they continue to face computationally intensive challenges. Due to power limitations and payload constraints, most UAVs use a single CPU as the computing platform, which significantly restricts processing speed. Consequently, achieving robust tracking without compromising high efficiency remains a difficult task for these methods.

Compared to deep learning-based trackers, discriminative correlation filter (DCF) trackers primarily address ridge regression problems to obtain filters capable of identifying objects from their surroundings. The core design principle is to use a target template to efficiently locate the target's position in the current frame through correlation operations, rather than performing a global detection across the entire image. Due to their high computational efficiency on a single CPU and robust performance, DCF trackers have found extensive application in UAV visual tracking, meeting the demands of real-time tracking. In existing research, efforts are mainly focused on enhancing target feature representation [13], [14], [15], [16], [17], improving the discriminative power of filters [18], [19], [20], [21], [22] and adjusting the precision of response maps [22], [23], [24], [25], [26] to achieve stable tracking in complex scenarios. However, most DCF-based trackers inadequately utilize historical information and overlook the fact that the rate of change between the target and the background is not always consistent. To address this issue, we exploit the dynamic-sensitivity error in response levels between frames and propose a learning dynamic-sensitivity enhanced correlation filter (LDECF) model. As illustrated in Fig. 1, the forward block and backward block respectively establish the forward-sensitive error $\Delta_f R$ and the historically-sensitive backtracking error $\Delta_b R$. There is a significant difference between the two within the same frame, and their differences between consecutive frames are also inconsistent. Therefore, incorporating the residuals of both into CF learning proves to be effective.

Maintaining the stability of mobile video tracking in highly dynamic environments is exceedingly challenging. This difficulty arises not only from the frequent changes in targets and backgrounds and the stochastic noise generated by the photon counting process in sensors but also from the inconsistent rates of these changes. Such variability severely tests the adaptability and response speed of tracking algorithms. The target's movement speed may suddenly accelerate or decelerate, viewpoint changes may shift from gradual to abrupt, and lighting conditions may quickly transition from extremely bright to dim. This inconsistency demands that tracking algorithms precisely capture the target's immediate state and swiftly adapt to these changes to maintain continuous and accurate tracking. Formally, this problem can be defined as follows: achieving stable target tracking in highly dynamic video environments requires addressing two main challenges. First, inconsistent rates of change arise from frequent and unpredictable variations in both the target and the background, such as abrupt shifts in target speed or environmental conditions. These variations challenge the algorithm's adaptability and real-time responsiveness, potentially compromising tracking accuracy, as shown by the abrupt changes in frames 38 and 75 in Fig. 3. Second, stochastic noise from photon counting in sensors adds further complexity, as illustrated by the noisy features in Fig. 2, where the contour appears suboptimal. Together, these challenges present significant obstacles to robust and reliable tracking under dynamic conditions. Existing UAV tracking methods face substantial limitations in handling such complex environments. Traditional DCF-based algorithms often struggle to adapt effectively to sudden changes in target movement and background, as their update mechanisms are designed primarily for relatively static or moderately dynamic scenarios. Moreover, most existing methods integrate deep features to enhance tracking performance, which hampers real-time tracking in realistic dynamic scenarios. Consequently, these methods often exhibit lower accuracy and efficiency when applied to highly dynamic environments with frequent, abrupt changes in speed, angle, or lighting. To address this issue, we explore denoising strategy to effectively eliminate noise caused by complex environments and comprehensively utilize inter-frame information to reflect changes in the response map.

In this paper, we first use the non-local means (NLM) denoising algorithm [27] to enhance the discriminative power of target contours. As illustrated in Fig. 2, the feature map of a small UAV target within the same frame exhibits significantly clearer contour after NLM denoising compared to the directly extracted feature. Additionally, the interference of target edges in the tracking response map is effectively suppressed, resulting in a higher real-time overlap rate. Furthermore, we conduct an in-depth analysis of the inconsistency in response level changes between forward tracking and historical backtracking in highly dynamic environments. This inconsistency reflects the variation gap between forward tracking and historical backtracking of the filter across consecutive frames. Based on this, we incorporate the inconsistency of response level

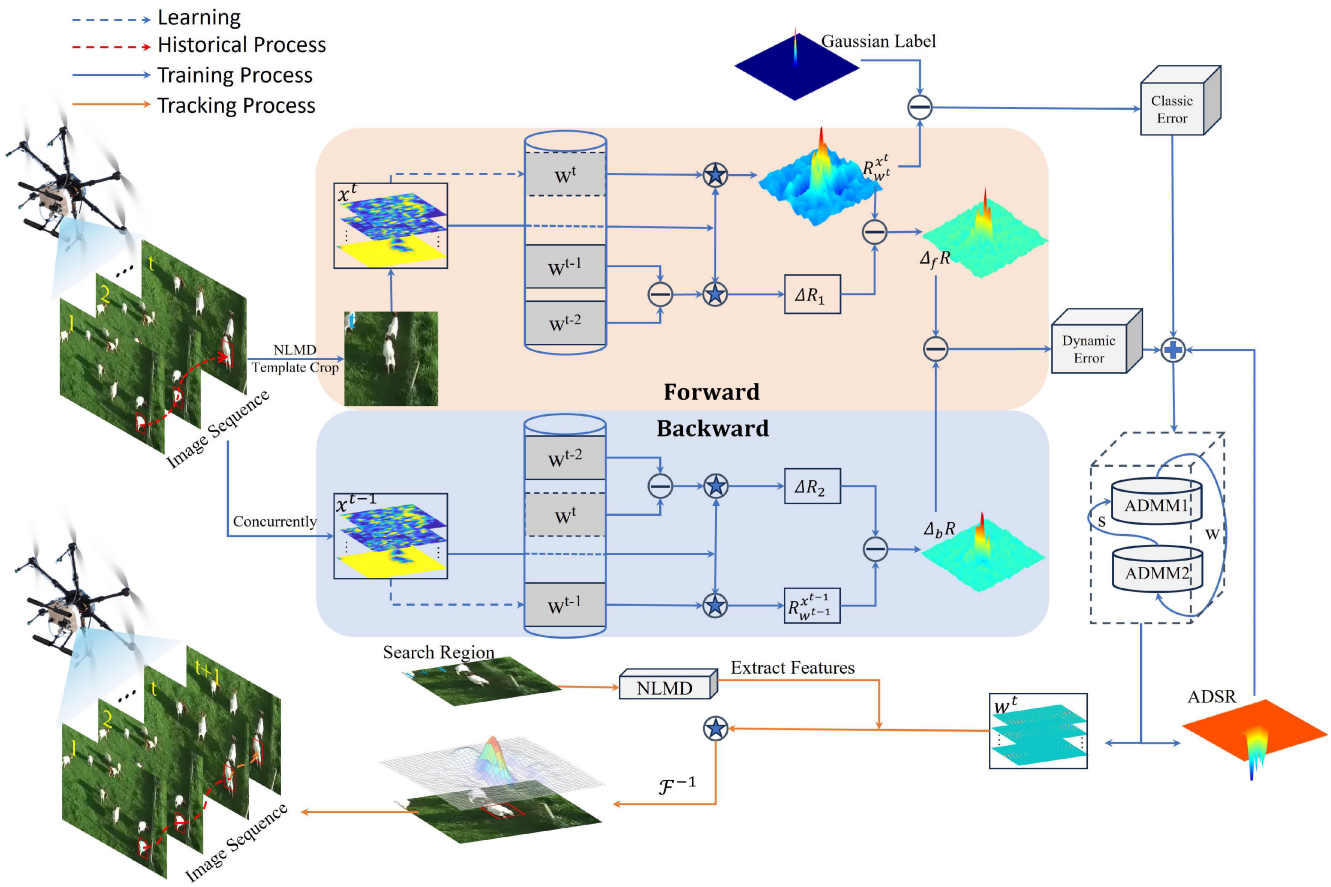


Fig. 1. A flowchart of the proposed LDECF tracker. In contrast to most advanced DCF-based methods, we first use an image denoising method (NLMD) before feature extraction to enhance the discriminative power of target contours, which is particularly effective in low-resolution images. Furthermore, we consider incorporating an additional error term into CF learning, namely the dynamic-sensitivity error, which reflects the dynamic residuals between the forward and backward phases of the filter's response to the appearance template across frames. To better integrate this error term, we propose an adaptive second-order difference spatial regularization (ADSR) to refine the fixed spatial regularizer, effectively working in synergy with the dynamic-sensitivity error strategy. Additionally, an ADMM method is introduced to solve this spatial regularizer, improving both convergence and computational efficiency.

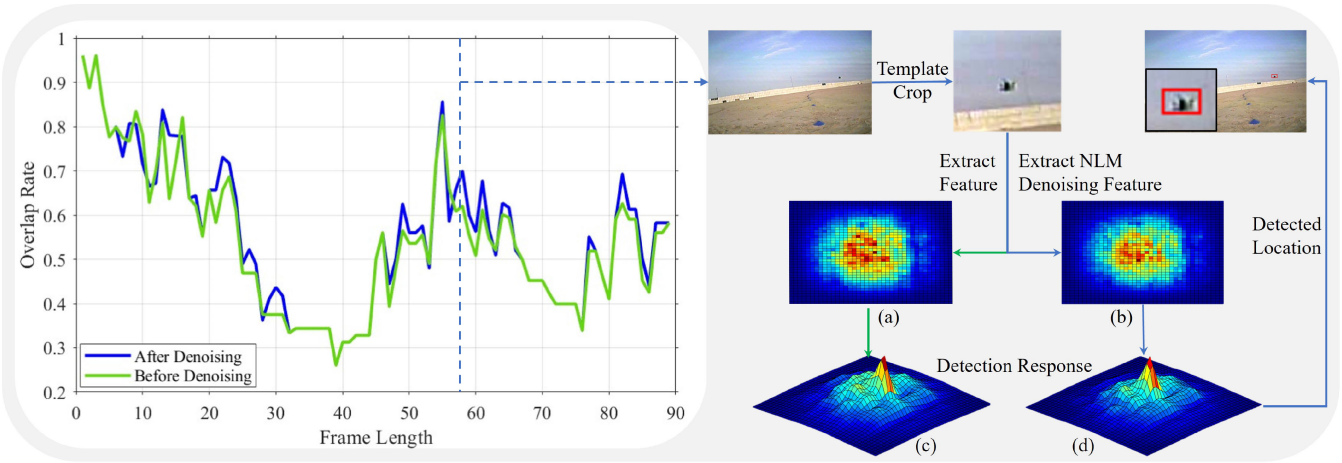


Fig. 2. Comparison of feature maps and detection response maps before and after NLM denoising, along with the real-time overlap rate of the corresponding sequence, shows that the feature contour obtained after NLM denoising is noticeably clearer than one directly extracted. Additionally, in the tracking response map, interference at the target edges is effectively suppressed, resulting in an enhanced real-time overlap rate.

changes into CF learning, introducing a dynamic-sensitivity error optimization problem and a second-order difference spatial regularization. The proposed method allows the filter to be constrained by the discriminative power of previous filters when the response level changes are small, preventing degradation. When changes are significant, the filter becomes more flexible, accommodating fluctuations in dynamic-sensitivity error and accelerating template updates to adapt to new

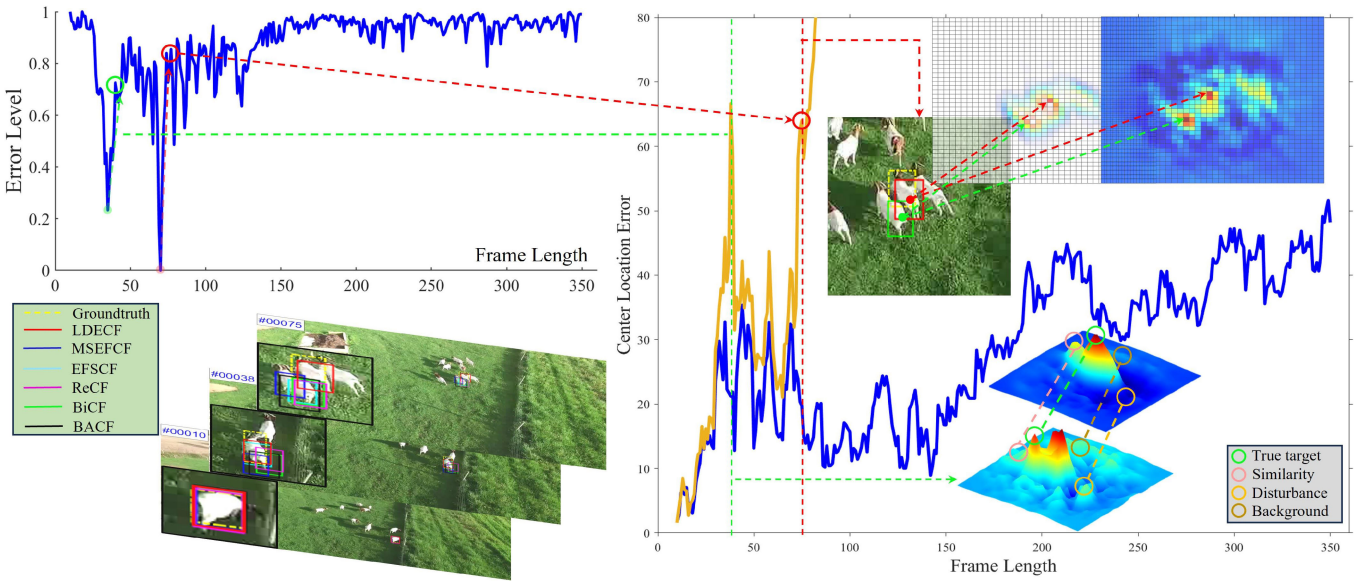


Fig. 3. Illustration of our dynamic-sensitivity error in an efficiently working way (Sequence Sheep1, DTB70). At low levels of error fluctuation, our model exhibits robustness similar to standard models. However, in cases of significant changes, our approach allows for faster updates of the filter template within short sequences of consecutive frames to better adapt to these changes. When the error level increases substantially, our model strengthens the penalty mechanism to reduce the impact of external disturbances, thereby enhancing overall robustness.

dynamic environments. This enhances the generalization capability to significant appearance changes.

In this paper, our goal is to address the inconsistency between rapid forward tracking and historical backtracking changes. Our contributions are summarized as follows:

- We introduce a denoising method before feature extraction to enhance target clarity and reduce edge interference in response maps, achieving lower computational costs than deep feature methods.
- We propose a new method that detects and adapts to response level inconsistencies between forward and backward tracking, termed dynamic-sensitivity error, improving tracking stability in dynamic environments without added computation.
- An adaptive second-order difference spatial regularization term is proposed, optimized by an additional ADMM, to synergize with the dynamic-sensitivity strategy, enhancing model convergence and computational efficiency.
- Extensive experiments on three UAV datasets demonstrate that the proposed LDECF tracker outperforms other state-of-the-art trackers in tracking performance.

An outline of the subsequent sections of this paper is provided as follows: Section II reviews the literature on tracking with discriminative correlation filters, along with an overview of the latest methods in the field. Section III presents our LDECF tracker model. In Section IV, an extensive experimental analysis is shown. Finally, Section V provides a discussion, followed by the conclusion in Section VI.

II. Related Work

A. Tracking Methods With ViTs

With the success of Transformers [28] in natural language processing tasks, many researchers have explored their application to computer vision tasks, leading to the development of various new architectures. DETR [29] is the first to introduce the Transformer model into vision tasks, while ViT [30] is the first to apply Transformers directly on non-overlapping image patches for image classification. Following ViT, numerous research teams extended and improved the model, making it suitable for a broader range of vision tasks. DeiT [31] introduces distillation techniques into the training process, optimizing the training pipeline and reducing the need for large-scale pre-training. To lower the computational complexity of ViT, Swin Transformer [32] performes self-attention locally within non-overlapping windows and introduces a shifted window mechanism to enable cross-window connections. To further accelerate ViT, many lightweight ViT variants have been proposed in recent years, including low-rank methods, model compression, and hybrid designs [33], [34], [35]. However, low-rank and quantization techniques often trade accuracy for efficiency, while pruning-based ViTs usually require complex pruning ratio selection and fine-tuning. Hybrid ViT structures using CNNs at the input level introduce certain advantages but significantly restrict input image size, limiting the ability to process images of varying dimensions. As ViTs have gained popularity in vision tasks, efficient ViTs based on conditional computation have recently begun exploring adaptive inference, dynamically adjusting the computational load based on input complexity to allocate resources more effectively during inference. For instance, DynamicViT [36] temporarily disables certain tokens by introducing control gates and using the Gumbel-softmax technique. A-ViT [37] adopts a similar approach to Adaptive Computation Time (ACT), eliminating the need for additional stop sub-networks and achieving significant improvements in efficiency, accuracy, and token importance allocation. A-ViT has also been applied in Aba-ViTrack [38], reducing inference time through adaptive and background-aware

token computation. Furthermore, AVTrack [39] introduces an adaptive computation framework designed to selectively activate Transformer blocks for real-time UAV target tracking. In practice, training a Vision Transformer model from scratch typically requires large-scale, high-quality training data and consumes substantial computational resources and time. In contrast, DCF-based methods do not require complex offline training and instead adapt rapidly to target changes through online update strategies. This approach effectively balances efficiency and performance, maintaining its practicality and competitiveness in resource-limited UAV visual tracking applications.

B. Tracking Methods With CFs

The minimum output sum of squared errors (MOSSE) [40] tracker is a pioneering method based on correlation filters (CFs). Its high computational efficiency is derived from the use of fast Fourier transform (FFT), which converts operations from the spatial domain to the Fourier domain, transforming complex cyclic correlations into element-wise operations. Since MOSSE's expression in the spatial domain results in ridge regression with a linear kernel, Henriques et al. [41] leverage the cyclic shifts of image patches to construct training samples, forming a circulant matrix. They propose an efficient kernelized correlation filter tracker (KCF) to exploit the strong discriminative power of nonlinear kernels. To address the boundary effects caused by cyclic sampling, Danelljan et al. [42] introduce a smooth spatial regularization factor into the regularizer to suppress boundary effects. In [43], BACF uses negative samples generated by real shifts, encompassing a larger search area and true background, rather than the traditional CF methods' negative samples generated by cyclic shifts of positive samples. An ADMM-based method is also proposed to enable the filter to utilize multi-channel features, significantly enhancing computational efficiency.

Some studies use new correlation filter learning strategies to improve tracking performance. Li et al. [18] consider temporal consistency to make the current filter more similar to the filter from the previous frame. This method can limit abrupt changes in the filter but also restricts the generalization capability of the learned filter. Yu et al. [44] integrate both first-order and second-order data-fitting terms into the DCF framework, enhancing the discriminative power of the learned correlation filter. Wang et al. [45] introduce a feature and expert pool to select the most reliable experts for tracking features in different frames and propose a multi-cue collaborative tracking scheme (MCCT). MCCT achieves better performance at the cost of increased computational complexity. In [46], a novel adaptive spatial regularized correlation filter (ASRCF) model is proposed to concurrently refine the filter parameters and spatial regularization factors. Zhang et al. [47] introduce a regularization term to learn the environmental residuals between two adjacent frames, achieving excellent performance. He et al. [22] propose prediction-based context regularization to enhance tracker robustness against complex environments.

C. Multi-Feature Methods

To fully exploit the information of objects for better representation, many successful tracking algorithms use robust handcrafted features (HOG [48] and CN [49]) on top of gray-scale features to maintain model robustness or to learn discriminative object representations. In [50], the authors propose the staple tracker, which combines HOG and CN features to enhance overall performance. Subsequently, the CSVMF [17] tracker introduces a multi-feature fusion strategy by integrating intensity, HOG, CN, and saliency features to better model the appearance of target objects in challenging tracking scenarios. In [15], a spatial domain adaptive feature selection method is proposed to improve robustness against background interference and noise over a broader search area. In [13], global feature-based selective enhancement of target area features is proposed to obtain multi-scale enhanced features, making it more suitable for complex tracking environments.

Some researchers also attempt to combine DCF models with deep visual features, enabling DCF-based trackers to achieve state-of-the-art performance [51], [52], [53]. However, when trackers utilize some powerful and complex features (especially those extracted from deep networks), this strategy significantly increases the computational load and reduces tracking speed.

D. Normalizing Response Bias Methods

Huang et al. [26] propose a new method to suppress anomalies occurring during the detection process, mitigating any potential noise during the training phase. However, this approach might limit the algorithm's adaptability to atypical target behaviors, as excessive noise suppression could sometimes lead to inadequate responses to real-world changes. Lin et al. [54] introduce bidirectional incongruity into CF learning. Lin et al. [25] introduce a response-weighted background residual term, enabling CF to selectively learn the background based on response strength. Zhang et al. [23] optimize input features with Gaussian denoising and introduce a context-based method to integrate continuously weighted dynamic response maps from both temporal and spatial perspectives. Zhang et al. [14] introduce a feature block-aware correlation filter method that separates target and background features and employs channel reliability weights for accurate target positioning. The idea of incorporating historical information and response maps into the tracking framework has proven to be effective.

When incorporating historical information, it is crucial to ensure its accuracy and relevance. Unlike most methods mentioned above, we initially apply the NLM algorithm for image denoising to optimize features, which incurs less computational overhead than incorporating deep features. Subsequently, we integrate information from consecutive frames in forward tracking and historical backtracking without incurring additional computational costs. Depending on the dynamic environment, the filter's update should balance constraints from previous filter templates with the flexibility to adapt to rapid changes, rather than merely suppressing anomalies.

By analyzing the dynamic-sensitivity error process of forward tracking and historical backtracking, we find that there is a discrepancy in the dynamic-sensitivity error of consecutive frame response levels, with inconsistent levels of change. Compared to the forward-backward error in [55] and the BiCF method in [54], which emphasize global error minimization or bidirectional incongruity, the proposed dynamic-sensitivity error focuses on the unique inconsistency change rate between appearance models and correlation filters across consecutive frames. Additionally, a novel adaptive second-order difference spatial regularization is employed to simultaneously optimize filter coefficients and spatial regularization weights, working in synergy with the dynamic-sensitivity error strategy. The proposed improvement enhances generalization capabilities without sacrificing tracking speed, making it suitable for robust real-time UAV tracking.

III. Proposed Method

To address the challenges posed by stochastic noise and dynamic motion rates in object tracking, we propose a learning dynamic-sensitivity enhanced correlation filter (LDECF) model, which includes the following components: 1) The employment of the NLM algorithm to denoise images in areas of interest. Unlike Gaussian denoising, NLM denoising considers repeated structures within the image, thereby better preserving image details and edges. 2) A dynamic-sensitivity error is introduced into the CF learning strategy, allowing the model to capture inconsistencies in response level change rates between forward tracking and historical backtracking across consecutive frames, thereby improving its ability to adapt to dynamic changes in both the target and background. 3) An adaptive second-order difference spatial weighting strategy is proposed, which suppresses abrupt changes caused by background noise or irrelevant information, and synergizes seamlessly with the dynamic-sensitivity error. 4) A correlational framework that captures and supports the nuances of inter-frame response levels is proposed. Our LDECF not only leverages the intrinsic characteristics of the correlation model to facilitate sudden shifts in response levels between training templates and matching frames but also pre-treats images with NLM denoising to preserve edge information and image structure before feature extraction. Different from the bidirectional incongruity discussed in Section II, our LDECF embraces both inconsistencies and abrupt changes in response levels between consecutive frames while addressing the limitations of fixed spatial regularizer.

A. Non-Local Means Denoising

In the realm of digital imaging, each pixel's triplet of color values is invariably subject to perturbations. These perturbations arise from the stochastic nature of photon detection within image sensors. Moreover, a myriad of external environmental influences can introduce noise during the critical stages of data acquisition, transmission, and processing, thereby degrading image quality. This vulnerability underscores the persistent challenges in preserving image fidelity throughout the digital workflow.

Given these challenges, it is crucial to employ an effective preprocessing strategy that includes robust denoising techniques. In this context, we advocate for implementing the non-local means [27] denoising algorithm before the feature extraction stage of the correlation filter algorithm to improve the tracker's overall recognition ability by capitalizing on the inherent self-similarity found within images. This approach posits that disparate regions of an image may exhibit highly similar local structures and textural features, even if these regions are spatially distant. Unlike traditional denoising methods that only consider immediate pixel neighborhoods, NLM algorithm extends its search across the entire image to identify and leverage these similarities, thereby achieving more comprehensive noise reduction. This broader scope not only enhances the effectiveness of denoising but also preserves the structural and textural integrity of the image.

For each pixel p in the image, with a neighborhood window of size $(2f+1)\times(2f+1)$, defining a search window of size $(2r+1)\times(2r+1)$. Within this search window, finding all regions similar to the local neighborhood window of the current pixel (assumed to be located at q). For the neighborhood windows of size $(2f+1)\times(2f+1)$ located at p and q, the squared Euclidean distance between the two blocks can be calculated as:

$$d^2(B(p,f),B(q,f)) = \frac{1}{3(2f+1)^2} \times \sum_{i=1}^{3}\sum_{j\in B(0,f)} (I_i(p+j)-I_i(q+j))^2, \tag{1}$$

where $B(p,f)$ and $B(q,f)$ are neighborhoods of size $(2f+1)\times(2f+1)$, centered at p and q respectively. I_i is the i-th channel of the image. Based on the calculated distances, a weight w [27] is assigned to each neighborhood pair (p,q) by using the exponential kernel:

$$w(p,q) = \exp\left(-\frac{\max(d^2-2\sigma^2,0)}{h^2}\right), \tag{2}$$

where σ denotes the standard deviation of the noise and $h=k\sigma$ is a filtering parameter that controls the degree of denoising. The value of k decreases as the size of the patch increases. Finally, the value of each pixel p is updated using a weighted average:

$$I_{\text{denoised}}(p) = \frac{\sum_{q\in S} w(p,q)\cdot I(q)}{\sum_{q\in S} w(p,q)}, \tag{3}$$

where S includes all pixels within the search window, and $I(q)$ is the original pixel value at position q.

Fig. 2 illustrates the process of tracking a small target, showing the feature maps before (a) and after (b) applying NLM denoising. The denoised feature map (b) reveals a clearer target contour, highlighting the effectiveness of NLM in enhancing target features amidst complex background noise. Furthermore, the response map in (d), learned from the denoised features, exhibits a higher peak and clearer boundaries at the target location compared to (c). It is evident that the denoising process allows for more effective feature extraction.

Consequently, the resulting response map achieves greater accuracy in locating and tracking the target, leading to a more promising real-time tracking overlap rate. This approach is particularly well-suited for scenarios involving target tracking in dynamic and low-resolution environments.

B. Dynamic-Sensitivity Error Modeling

Since BiCF [54] effectively leverages the core idea of forward-backward error [55] by incorporating bidirectional incongruity into CF learning. Inspired by this, we conduct an in-depth analysis of the inconsistency in response level changes between forward-sensitive tracking and historically-sensitive backtracking in highly dynamic environments, which reveals the filter performance fluctuations caused by target or background changes between consecutive frames. In contrast to BiCF, which primarily focuses on bidirectional incongruity between two consecutive frames, we consider the multi-frame historical information for immediate response adjustments and introduce this inconsistency in response level changes into CF learning.

Fig. 1 illustrates how to construct dynamic-sensitivity error, which include forward-sensitive tracking error $\Delta_f R$ and historically-sensitive backtracking error $\Delta_b R$. In the training process, we use a three-stage time point response analysis to train enhanced filters with dynamic-sensitivity. Suppose x^t preprocessed by the NLM denoising algorithm is extracted from the t th frame. Theoretically, the degree and rate of change in response for the current feature x^t at consecutive time points should be consistent with the previous moment. In other words, the object and environment are expected to remain stable. However, in real tracking scenarios, due to sudden changes, occlusions, and other factors, there is a significant difference between the response $R_{w^t}^{x^t}$ of the filter w^t to the feature x^t and the responses ($R_{w^{t-1}}^{x^t}$ and $R_{w^{t-2}}^{x^t}$) of nearby filters to the same feature, even abruptly. That is

$$R_{w^t}^{x^t} - R_{w^{t-1}}^{x^t} \neq R_{w^{t-1}}^{x^t} - R_{w^{t-2}}^{x^t}, \tag{4}$$

where $R_{w^{t-i}}^{x^t} = w^{t-i} * x^t, i = 0, 1, 2$. In forward-sensitive tracking, the filter should prioritize the state of the current frame t, requiring a comparison of the response in frame t with the responses of the two preceding frames. This process computes the actual difference between the current frame's response and the neighboring frame values, quantifying the deviation of the current response from its neighboring responses. Therefore, the forward-sensitive tracking error $\Delta_f R$ is defined as follows:

$$\begin{aligned}\Delta_f R &= (R_{w^t}^{x^t} - R_{w^{t-1}}^{x^t}) - (R_{w^{t-1}}^{x^t} - R_{w^{t-2}}^{x^t}) \\ &= R_{w^t}^{x^t} - (2 \cdot R_{w^{t-1}}^{x^t} - R_{w^{t-2}}^{x^t}).\end{aligned} \tag{5}$$

Simultaneously, the filter should not only capture the current state of the target but also vividly paint a picture of its historical trajectory changes. In the retrospective response of the target's position, the historical filter's response $R_{w^{t-1}}^{x^{t-1}}$ with the historical features x^{t-1} should mirror the response from the prior moment's filter $R_{w^t}^{x^{t-1}}$, while also taking into account even earlier response such as $R_{w^{t-2}}^{x^{t-1}}$. The equation can be expressed by replacing t with $t-1$ in Eq. (4). In historically-sensitive backtracking, the filter should prioritize the state of the current frame $t-1$, requiring a comparison of its response with those of the two adjacent frames. This leads to the definition of the historically-sensitive backtracking error $\Delta_b R$ as follows:

$$\Delta_b R = 2 \cdot R_{w^{t-1}}^{x^{t-1}} - (R_{w^{t-2}}^{x^{t-1}} + R_{w^t}^{x^{t-1}}), \tag{6}$$

where $R_{w^{t-i}}^{x^{t-1}} = w^{t-i} * x^{t-1}, i = 0, 1, 2$. Our extensive experiments have revealed a notable inconsistency between the forward-sensitive tracking error and the historically-sensitive backtracking error, with abrupt changes sometimes occurring between consecutive frames. This phenomenon not only highlights the asymmetry inherent in the tracking process but also underscores the complexity of the target's dynamic behavior. In response to these challenges and to accurately capture and adapt to the disturbances caused by such discrepancies, the dynamic-sensitivity error is proposed and defined as:

$$\epsilon = \left\| \Delta_f R - \Delta_b R \right\|^2 \tag{7}$$

By introducing the new dynamic-sensitivity error metric ϵ, which quantifies the inconsistent rates of environmental changes across consecutive frames, the filter's sensitivity and responsiveness to rapid changes can be effectively enhanced.

Fig. 3 visually illustrates the working mechanism of the dynamic-sensitivity error. Throughout the tracking process, the Error Level undergoes two abrupt changes. The first occurs when the target changes from stationary to rapidly accelerating, notably while the camera is in motion. At this moment, our LDECF successfully adapts to this high-dynamic environment, whereas some advanced trackers experience tracking drift. The second abrupt change happens when the running target blends into a similar group and then begins to gradually decelerate. Our LDECF successfully distinguishes the true target, which exhibits similar dynamic movement as before, from a relatively stationary target (walking on the left). In this situation, other sophisticated trackers make tracking mistakes.

C. Adaptive Second-Order Difference Spatial Regularization

Excessive filter sensitivity, as captured by the dynamic-sensitivity strategy, can result in heightened responsiveness to minor variations or background noise, leading to false positives or tracking inaccuracies. Consequently, the algorithm may erroneously track irrelevant background changes or noise instead of the target object, severely compromising tracking performance, particularly in complex and dynamic environments.Consequently, the algorithm may erroneously track irrelevant background changes or noise instead of the target object, severely compromising tracking performance, particularly in complex and dynamic environments. Acknowledging the potential challenge of the proposed dynamic-sensitivity and drawing inspiration from the rearranged form of Equations (5) and (6), which align with the second-order difference, we introduce an adaptive second-order difference spatial regularization (ADSR) to improve the fixed spatial regularizer. The proposed spatial regularization mitigates abrupt fluctuations in filter weights caused by noise or abnormal

changes in the current frame by referencing the weight update trend from historical frames ($s - 2s^{t-1} + s^{t-2}$). Designed to complement the dynamic-sensitivity strategy, its effectiveness is demonstrated in the ablation study in the subsection IV-E. This spatial regularizer constrains the spatial distribution of filter responses, effectively suppressing activations in noisy or less relevant regions, thereby reducing the risk of overfitting. Furthermore, the second-order difference spatial regularization regulates variations in filter coefficients, ensuring smoother spatial responses. By dynamically adjusting filter weights based on the target's appearance in the current frame, this approach enhances the tracker's robustness. The adaptive nature of this strategy, achieved through dynamically updated regularization weights, overcomes the limitations of fixed filter coefficients in responding to target variations.

D. Proposed LDECF Tracker

Before presenting our model, this section provides a brief overview of the baseline tracker, BACF [43], to facilitate a better understanding of our approach. Given D channel vectorized samples $x^d \in \mathbb{R}^N$ $(d = 1, 2, \ldots, D)$ and a vectorized ideal response $y \in \mathbb{R}^N$, the BACF minimizes the objective function as:

$$\mathcal{E}(w) = \frac{1}{2}\left\| y - \sum_{d=1}^{D} Bx^d * w^d \right\|_2^2 + \sum_{d=1}^{D} \|w^d\|_2^2. \tag{8}$$

where x^d represents the feature map of the object block in the d-th channel, $B \in \mathbb{R}^{M\times N}$ (where $M \ll N$) is a binary cropping matrix used to select the central M elements of each channel's input vector sample x^d. $w^d \in \mathbb{R}^M$ is the correlation filter learned in the d-th channel. $*$ denotes a correlation operator.

The proposed LDECF tracker is designed to keenly reflect the degree and speed of target state changes, with a focus on the impact of dynamic variations between consecutive frames. In SRDCF, the spatial regularizer remains consistent throughout the tracking process and across different objects. This one-size-fits-all approach struggles to accommodate the unique characteristics and appearance variations of each object. Additionally, the ASRCF method, in light of the above discussion, proposes switching from a fixed spatial regularization term to an adaptive spatially regularization term, allowing the adaptive spatial weight s to closely align with the reference weight s^{t-1}. This effectively prevents model degradation. However, these adaptive spatial weights may not fully adapt to current and previous target states. In this paper, we introduce an new adaptive second-order difference spatial weighting strategy, designed to work in tandem with dynamic-sensitivity errors, thereby precisely capturing fluctuations of the dynamic-sensitivity error between consecutive frames. The adaptive second-order difference spatial regularization term is effectively integrated into the correlation filter (CF) learning framework to enhance the model's adaptability and sensitivity to temporal variations. This strategy helps the filter maintain target continuity by capturing smooth transitions typically observed in multi-frame tracking, while suppressing abrupt changes from background noise or irrelevant information, thereby enhancing tracking stability. Combining the dynamic-sensitivity error and the ADSR term, the objective function is defined as follows:

$$\begin{aligned}\mathcal{E}(w, s) = \frac{1}{2}\left\| y - \sum_{d=1}^{D} Bx_d^t * w_d^t \right\|_2^2 + \frac{\lambda_1}{2}\sum_{d=1}^{D} \|s \odot w_d\|_2^2 \\ + \frac{\lambda_2}{2}\|s - \Delta s\|_2^2 + \frac{\gamma}{2}\sum_{d=1}^{D}\epsilon_d\end{aligned} \tag{9}$$

where $\Delta s = 2s^{t-1} - s^{t-2}$, the adaptive second-order difference spatial weight s on the filter w_d that needs to be optimized, and λ_1 represents the regularization parameter. The third term adjusts the spatial weights s, with λ_2 denoting the regularization parameter, attempting to consider changes between previous reference weights s^{t-1} and s^{t-2}. This not only introduces prior knowledge of s but also more flexibly learns the impact of spatial changes, preventing performance degradation of the model.

ϵ_d represents the dynamic-sensitivity error of the d-th channel, which can be calculated using the following formula:

$$\begin{aligned}\epsilon_d &= \left\| \Delta_f R_d - \Delta_b R_d \right\|_2^2 \\ &= \left\| (R_{w_d^t}^{x_d^t} - R_1) - (2 \cdot R_{w_d^{t-1}}^{x_d^{t-1}} - R_2) \right\|_2^2 \\ &= \left\| B\left((x_d^t * w_d^t - R_1) - (2 \cdot x_d^{t-1} * w_d^{t-1} - R_2) \right) \right\|_2^2 \\ &= \left\| M * (B^T w_d^t - \Delta R_d) \right\|_2^2.\end{aligned} \tag{10}$$

where $M = x_d^t + x_d^{t-1}$, $R_1 = 2 \cdot x_d^t * w_d^{t-1} - x_d^t * w_d^{t-2}$, $R_2 = x_d^{t-1} * w_d^{t-2} + x_d^{t-1} * w_d^t$ and $\Delta R_d = B^T(2w_d^{t-1} - w_d^{t-2})$. Therefore, Eq. (9) can be reformulated as:

$$\begin{aligned}\mathcal{E}(w, s) = \frac{1}{2}\left\| y - \sum_{d=1}^{D} Bx_d^t * w_d^t \right\|_2^2 + \frac{\lambda_1}{2}\sum_{d=1}^{D} \|s \odot w_d\|_2^2 \\ + \frac{\lambda_2}{2}\|s - \Delta s\|_2^2 + \frac{\gamma}{2}\sum_{d=1}^{D}\left\| M * (B^T w_d^t - \Delta R_d) \right\|_2^2.\end{aligned} \tag{11}$$

E. Optimization Through ADMM

Note that $\mathcal{E}(w, s)$ can be decomposed into D error terms $\mathcal{E}_d (d = 1, 2, \ldots, D)$ for optimization, and in this work, the d-th channel is selected for the derivation of the following model. To optimize the filter w_d^t, we introduce an auxiliary variable $g_d^t = B^T w_d^t \in \mathbb{R}^N$. The original optimization problem is decomposed into several more manageable subproblems. These subproblems can be iteratively solved using the ADMM method. Accordingly, we can rewrite $\mathcal{E}_d$ in a form with equality constraints:

$$\begin{aligned}\mathcal{E}_d(g_d, w_d, s) = \frac{1}{2}\left\| y - x_d^t * g_d^t \right\|_2^2 + \frac{\lambda_1}{2}\|s \odot w_d\|_2^2 \\ + \frac{\lambda_2}{2}\|s - \Delta s\|_2^2 + \frac{\gamma}{2}\left\| M * (g_d^t - \Delta R_d) \right\|^2, \\ \text{s.t.} \quad g_d^t = B^T w_d^t, \quad d = 1, 2, \ldots, D\end{aligned} \tag{12}$$

According to the Parseval's theorem, we can transform Eq. (12) for expression and processing in the Fourier domain to improve computational efficiency:

$$\begin{aligned}\mathcal{E}_d(\hat{g_d}, w_d, s) = &\frac{1}{2}\left\|\hat{y} - \hat{x}_d^t \odot \hat{g}_d^t\right\|_2^2 + \frac{\lambda_1}{2}\|s \odot w_d\|_2^2 \\ &+ \frac{\lambda_2}{2}\|s - \Delta s\|_2^2 + \frac{\gamma}{2}\left\|\hat{M} \odot (\hat{g}_d^t - \Delta\hat{R}_d)\right\|_2^2, \\ \text{s.t.}\quad &\hat{g}_d^t = \sqrt{N}FB^T g_d^t, \quad d = 1, 2, \ldots, D\end{aligned} \tag{13}$$

where the superscript $\hat{}$ is used to denote the discrete Fourier transform (DFT) of a signal and its complex conjugate. The DFT matrix $F \in \mathbb{C}^{N\times N}$ transforms a real vector $v \in \mathbb{R}^{N\times 1}$ into its frequency domain representation, that is $\hat{v} = \sqrt{N}Fv$.

Furthermore, Eq. (13) can be expressed in the augmented Lagrangian form:

$$\begin{aligned}\mathcal{L}(\hat{g_d}, w_d, \hat{\zeta}_d, s) = \mu &\left\|\hat{g}_d^t - \sqrt{N}FB^T w_d + \frac{1}{\mu}\hat{\zeta}_d\right\|_2^2 \\ &+ \mathcal{E}_d(\hat{g_d}, w_d, s),\end{aligned} \tag{14}$$

here $\hat{\zeta}_d \in \mathbb{C}^N$ is the Lagrange multiplier in the dth channel, and μ is a penalty parameter. Next, we decompose the original optimization problem into three subproblems and use the ADMM method to solve these subproblems:

$$\begin{cases} w_d^{(i+1)} = \arg\min\limits_{w} \lambda_1\|s \odot w_d\|_2^2 \\ \qquad + \dfrac{\mu}{2}\left\|\hat{g}_d^t - \sqrt{N}F^T B w_d + \hat{\zeta}_d\right\|_2^2, \\ \hat{g_d}^{(i+1)} = \arg\min\limits_{\hat{g}} \dfrac{1}{2}\left\|\hat{y} - \hat{x}_d^t \odot \hat{g}_d^t\right\|_2^2 \\ \qquad + \dfrac{\gamma}{2}\left\|\hat{M} \odot (\hat{g}_d^t - \Delta\hat{R}_d)\right\|_2^2 \\ \qquad + \mu\left\|\hat{g}_d^t - \sqrt{N}FB^T w_d + \dfrac{1}{\mu}\hat{\zeta}_d\right\|_2^2, \\ s^{(i+1)} = \arg\min\limits_{s} \dfrac{\lambda_1}{2}\|s \odot w_d\|_2^2 + \dfrac{\lambda_2}{2}\|s - \Delta s\|_2^2. \end{cases} \tag{15}$$

where i represents the current number of iterations. The detailed iterative solution for each subproblem is as follows.

Solving w_d: Taking the derivative with respect to w_d and setting it to zero, we obtain the optimal solution for $w_d^{(i+1)}$:

$$w_d^{(i+1)} = \frac{\mathcal{F}^{-1}\left(\mu\hat{g}_d^t + \hat{\zeta}_d\right)}{\frac{\lambda_1}{N}(s \odot s) + \mu}, \tag{16}$$

here the division operator indicates element-wise division.

Solving $\hat{g_d}$: Taking the derivative with respect to $\hat{g_d}$ and setting it to zero, the optimal solution for $\hat{g_d}^{(i+1)}$ is obtained as follows:

$$\hat{g_d}^{(i+1)} = \frac{(\hat{x}_d^t)^T y + \gamma\hat{M}^T\hat{M}\Delta\hat{R}_d + \mu\hat{w}_d - \hat{\zeta}_d}{(\hat{x}_d^t)^T\hat{x}_d^t + \gamma\hat{M}^T\hat{M} + \mu}, \tag{17}$$

Solving s: Fixing w_d, $\hat{g}_d^t$, and $\hat{\zeta}_d$, to enhance model convergence, we additionally employ an ADMM solver to solve s. An auxiliary variable $q = s$ is introduced, and the original subproblem s is divided into two additional subproblems:

$$\begin{cases} s^{(i+1)} = \arg\min\limits_{s} \dfrac{\lambda_1}{2}\|W_d s\| + \dfrac{\mu_s}{2}\|s - q + u\|_2^2, \\ q^{(i+1)} = \arg\min\limits_{q} \dfrac{\lambda_2}{2}\|q - \Delta s\|_2^2 + \dfrac{\mu_s}{2}\|s - q + u\|_2^2 \end{cases} \tag{18}$$

where $W_d = \text{diag}(w_d) \in \mathbb{R}^{N\times N}$. Using a similar method to solve for s, the analytical solution is obtained as:

$$s^{(i+1)} = \frac{\mu_s(q - u)}{\lambda_1 w_d \odot w_d + \mu_s}. \tag{19}$$

For the subproblem q, the analytical solution can also be derived:

$$q^{(i+1)} = \frac{\lambda_2\Delta s + \mu_s(s + u)}{\lambda_2 + \mu_s}. \tag{20}$$

Update the Lagrange Multiplier $\hat{\zeta}_d$ and u:

$$\hat{\zeta}_d^{(i+1)} = \hat{\zeta}_d^{(i)} + \mu\left(\hat{g_d}^{(i+1)} - \hat{w}_d^{(i+1)}\right), \tag{21}$$

$$u_d^{(i+1)} = u_d^{(i)} + \mu_s\left(s^{(i+1)} - q_d^{(i+1)}\right), \tag{22}$$

where $\hat{w}_d^{(i+1)} = \sqrt{N}FB^T w_d^{(i+1)}$. In the i-th iteration of ADMM, the parameter μ is typically updated as follows:

$$\mu^{(i+1)} = \min(\mu_{\max}, \beta\mu^{(i)}). \tag{23}$$

Online Updating: To better adapt to the dynamic changes of the target, we employ an online adaptive scheme to enhance the robustness of the filter, as follows:

$$\hat{x}_{d,\text{model}} = (1 - \eta)\hat{x}_{d,\text{model}}^{t-1} + \eta\hat{x}_d. \tag{24}$$

Here $\hat{x}_{d,\text{model}}$ and $\hat{x}_{d,\text{model}}^{t-1}$ respectively represent the appearance models of the current and previous frames, and η is the online adaptation rate. The optimization details of our LDECF in the d-th channel of the t-th frame are summarized in Algorithm 1.

Algorithm 1 LDECF Tracker

Input: Image: I_t,
Previous two filters: w_d^{t-1}, w_d^{t-2},
Feature of the $t-1$-th frame: x_d^{t-1},
Spatial regularization weights: s
1: Extract feature x_d^t after performing NLM from I_t,
2: Initialize variables $\hat{g}_d^t(0)$, $w_d^t(0)$, and $\hat{\zeta}_d$,
3: **for** iteration $i = 1$
4: Update $w_d^{(i+1)}$ via Eq. (16),
5: Update $\hat{g_d}^{(i+1)}$ via Eq. (17),
6: Update $s^{(i+1)}$ via Eq. (19),
7: Update the appearance model using Eq. (24),
Ouput: Current filter in the t-th frame: w_d^t

F. Computational Complexity Analysis

For each pixel p in the image, it needs to be compared with all pixels within the search region, and the weighted Euclidean distance between their neighborhoods must be calculated. Consequently, the total computational cost in NLM denoising is $O(WHS^2P^2)$, where $WH = N$ represents the image size,

TABLE I

ATTRIBUTE BASED DP/AUC OF THE PROPOSED LDECF AND 17 OTHER TRACKERS ON THE UAV123@10FPS DATABASE. THE TOP THREE METHODS IN EACH ATTRIBUTE ARE DENOTED BY DIFFERENT COLORS: RED, GREEN AND BLUE. (↑) INDICATES OUTPERFORMANCE AGAINST BASELINE

Trackers	SV (↑)	ARC (↑)	LR (↑)	FM (↑)	POC (↑)	IV (↑)	VC (↑)	CM (↑)	SOB(↑)
BACF	0.525/0.456	0.478/0.396	0.431/0.293	0.407/0.332	0.467/0.387	0.430/0.369	0.491/0.409	0.532/0.483	0.605/0.528
STRCF	0.580/0.494	0.524/0.413	0.509/0.334	0.488/0.377	0.559/0.454	0.493/0.406	0.537/0.438	0.602/0.526	0.630/0.555
EFSCF	0.581/0.505	0.532/0.433	0.515/0.351	0.479/0.379	0.560/0.466	0.494/0.403	0.526/0.441	0.592/0.526	0.624/0.557
ECO_HC	0.587/0.506	0.558/0.449	0.527/0.346	0.487/0.390	0.556/0.462	0.507/0.430	0.548/0.459	0.609/0.544	0.637/0.566
LADCF	0.608/0.516	0.561/0.445	0.549/0.370	0.503/0.389	0.564/0.461	0.481/0.410	0.548/0.457	0.637/0.551	0.653/0.554
DRCF	0.604/0.520	0.553/0.442	0.522/0.354	0.520/0.386	0.591/0.500	0.556/0.442	0.566/0.464	0.610/0.531	0.620/0.546
MSCF	0.606/0.513	0.567/0.446	0.521/0.343	0.517/0.401	0.582/0.477	0.543/0.424	0.578/0.469	0.598/0.524	0.650/0.553
CACF	0.608/0.521	0.579/0.461	0.519/0.368	0.352/0.352	0.570/0.479	0.560/0.445	0.574/0.467	0.595/0.529	0.692/0.585
MSEFCF	0.640/0.547	0.605/0.482	0.533/0.388	0.539/0.412	0.571/0.497	0.567/0.452	0.593/0.471	0.653/0.574	0.672/0.592
BiCF	0.619/0.534	0.578/0.466	0.534/0.371	0.484/0.376	0.591/0.502	0.581/0.494	0.581/0.496	0.625/0.555	0.705/0.609
ReCF	0.621/0.529	0.584/0.465	0.543/0.359	0.512/0.406	0.590/0.499	0.592/0.501	0.581/0.491	0.632/0.562	0.687/0.598
ARCF	0.623/0.529	0.580/0.461	0.561/0.394	0.516/0.378	0.582/0.498	0.552/0.467	0.573/0.472	0.610/0.530	0.657/0.573
AutoTrack	0.629/0.535	0.598/0.476	0.532/0.372	0.525/0.407	0.584/0.496	0.550/0.472	0.588/0.480	0.647/0.564	0.664/0.569
RBSCF	0.629/0.535	0.583/0.462	0.554/0.393	0.503/0.386	0.562/0.483	0.554/0.452	0.597/0.474	0.631/0.552	0.655/0.572
IBRI	0.631/0.529	0.592/0.461	0.559/0.393	0.547/0.383	0.595/0.498	0.548/0.450	0.590/0.469	0.623/0.526	0.656/0.568
EMCF	0.644/0.545	0.608/0.479	0.572/0.395	0.510/0.391	0.583/0.494	0.572/0.478	0.602/0.489	0.646/0.569	0.678/0.576
RCFL	0.647/0.546	0.624/0.489	0.573/0.390	0.552/0.438	0.609/0.507	0.593/0.466	0.607/0.485	0.655/0.570	0.705/0.593
LDECF	0.663/0.563	0.635/0.502	0.593/0.419	0.552/0.438	0.612/0.512	0.613/0.489	0.614/0.492	0.677/0.587	0.707/0.604

S^2 is the area of the search window, and P^2 is the area of the neighborhood window. In practice, an accelerated NLM algorithm, as demonstrated in [56], can be implemented to reduce the cost to $O(NS^2)$. The computation of Eq. (16) has a complexity bounded by $O(DN\log(N))$, where D represents the number of channels, and $O(N\log(N))$ refers to the cost of computing the IFFT for a signal of length N. Eq. (17) is pixel-wise separable, with a complexity of $O(ND)$, since M and ΔR_d can be precomputed from historical frames, and B (cropping matrix) can be efficiently performed via a lookup table. In Eqs. (19 and 20), the computational complexities are $O(N)$. Hence, the overall computational cost of Eq. (11) is $O(AIDN\log(N))$, where I represents the number of ADMM iterations, and A comes from the additional cost incurred by ADMM when solving ADSR. Finally, the total cost of our algorithm is $O(NS^2 + AIDN\log(N))$. Compared to the baseline method (BACF) with a computational complexity of $O(IDN\log(N))$, the total computational complexity of our proposed method increases by a factor of A and includes the denoising cost.

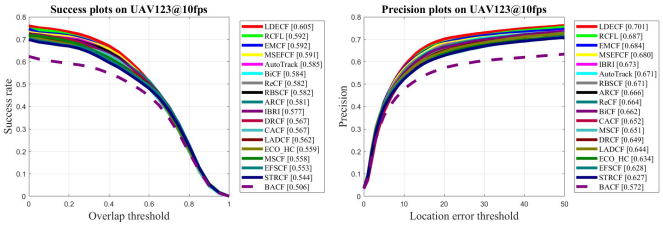


Fig. 4. Success plots and precision plots of the proposed LDECF and other trackers on the UAV123@10fps database.

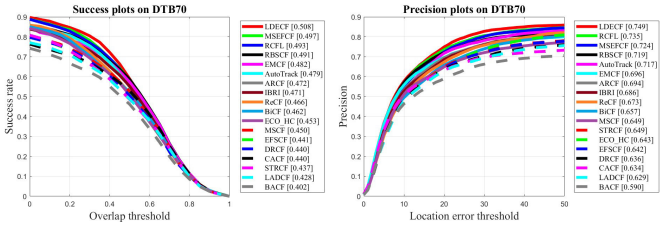


Fig. 5. Success plots and precision plots of the proposed LDECF and other trackers on the DTB70 database.

IV. EXPERIMENTS

In this section, the proposed LDECF tracker is evaluated through extensive experiments on three widely used drone tracking datasets, including UAV123@10fps [1], DTB70 [57], and UAVDT [58].

A. Dataset

The UAV123@10fps dataset, collected from a low-altitude perspective, consists of 123 video sequences with the original sampling rate reduced from 30fps to 10fps. The lack of intermediate frames results in greater object displacement and appearance changes between frames, thereby increasing the difficulty of tracking. DTB70 consists of 70 video sequences totaling 15,777 frames. Some videos were recorded using drones within university campuses and others were collected from YouTube to increase the diversity of target appearances and scenes. DTB70 is characterized by high diversity and low bias, providing a valuable resource for tracking algorithm research. UAVDT is specifically designed for vehicle tracking,

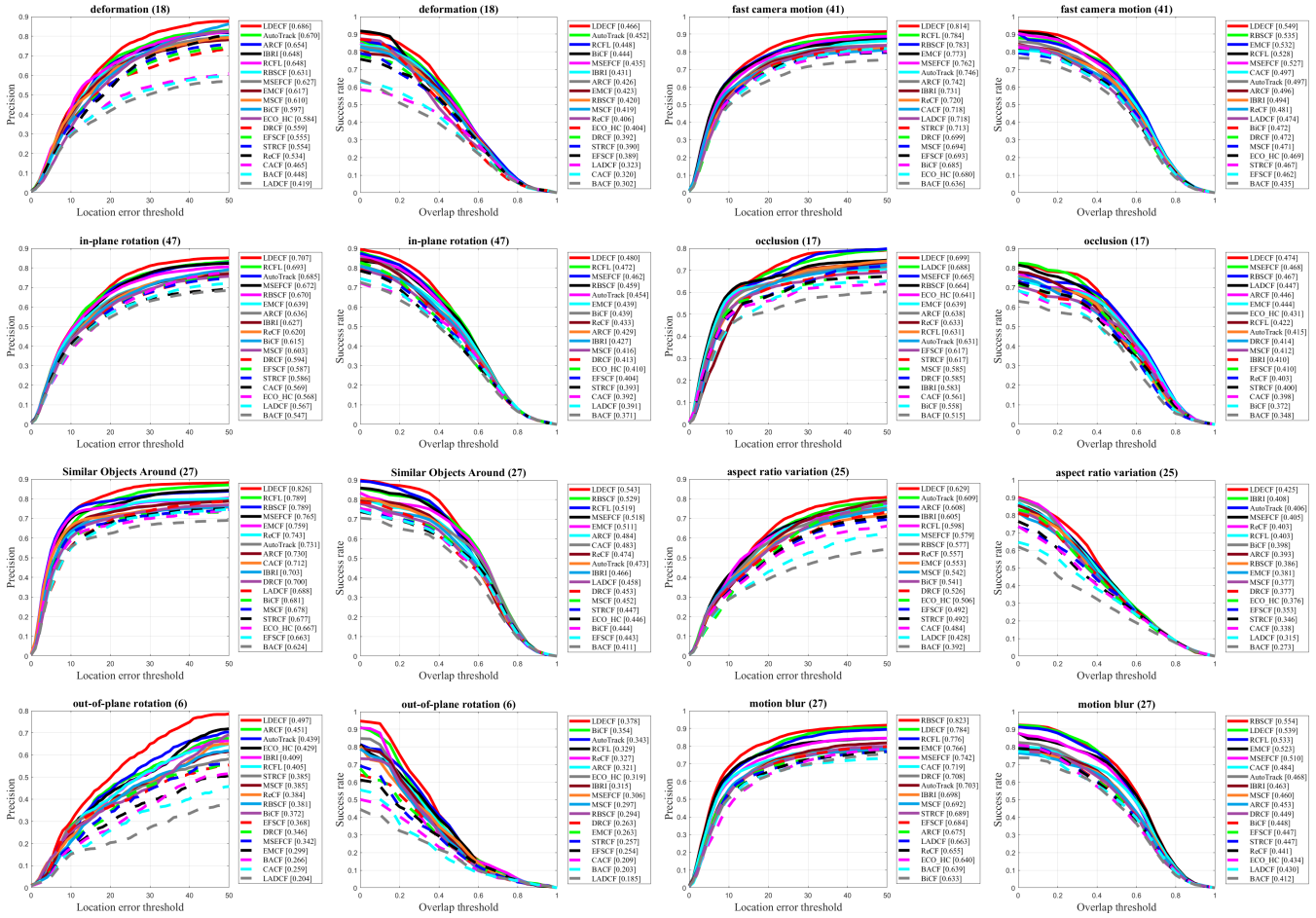


Fig. 6. Precision and success plots of our LDECF and 18 other state-of-the-art trackers using hand-crafted features in eight attributes on the DTB70 benchmark.

including 30 training sequences and 70 testing sequences. Compared to other drone datasets, UAVDT focuses mainly on vehicles facing complex new challenges, such as varying weather conditions, different flight altitudes, and camera angles.

B. Implementation Detail

The proposed LDECF tracker is implemented in the MATLAB R2022b environment on a computer equipped with an Intel i5-13500HX CPU (2.50 GHz). The one-pass evaluation (OPE) criterion is used to measure the tracking accuracy and success rate of the proposed LDECF, and frames per second (FPS) is adopted as a measure of tracking speed. The tracker utilizes handcrafted features including histogram of oriented gradients (HOG) with 31 channels, color naming (CN) with 10 channels, and gray-scale features with 1 channel to characterize the target. The sizes of the search window and neighborhood window for NLM denoising are set to 11 and 5, respectively, with the value of k being 0.03. The regularization parameters λ_1, λ_2, and γ are set to 1.2, 0.001, and 0.14, respectively. The alternating direction method of multipliers (ADMM) is used for optimization, the first and second initial penalty factors are set to 100 and 3, respectively. The step-size ratios are set to 50 and 10, and the maximum value is 10^5. The number of ADMM iterations is set to 3 and the online adaptation rate η is set to 0.0365.

C. Comparison of Trackers With Hand-Crafted Features

1) Overall performance: The proposed LDECF tracker is first compared with 17 state-of-the-art trackers using hand-crafted features, i.e., ECO [73], BACF [43], STRCF [18], ARCF [26], DRCF [74], IBRI [75], AutoTrack [76], BiCF [54], ReCF [77], LADCF [20], MSCF [78], EMCF [47], CACF [79], RCFL [23], RBSCF [25], EFSCF [15], and MSEFCF [13], on two UAV benchmarks in terms of overall performance. Fig. 4 and Fig. 5 present that our LDECF tracker obtains the outstanding performance compared with the other 17 hand-crafted feature-based trackers on two challenging UAV benchmarks.

In Fig. 4, our LDECF tracker achieves the highest DP score (0.701), surpassing the second highest tracker, RCFL (0.687), by 1.4%, and the third highest tracker, EMCF (0.684), by 1.7%. LDECF also leads in the AUC score (0.605), which is 1.3% higher than both RCFL and EMCF (0.592), and 1.4% higher than MSEFCF (0.591), which is in third place. Moreover, compared to the baseline method BACF (0.572/0.506), LDECF achieves a substantial improvement of 12.9% in the DP score and 9.9% in the AUC score.

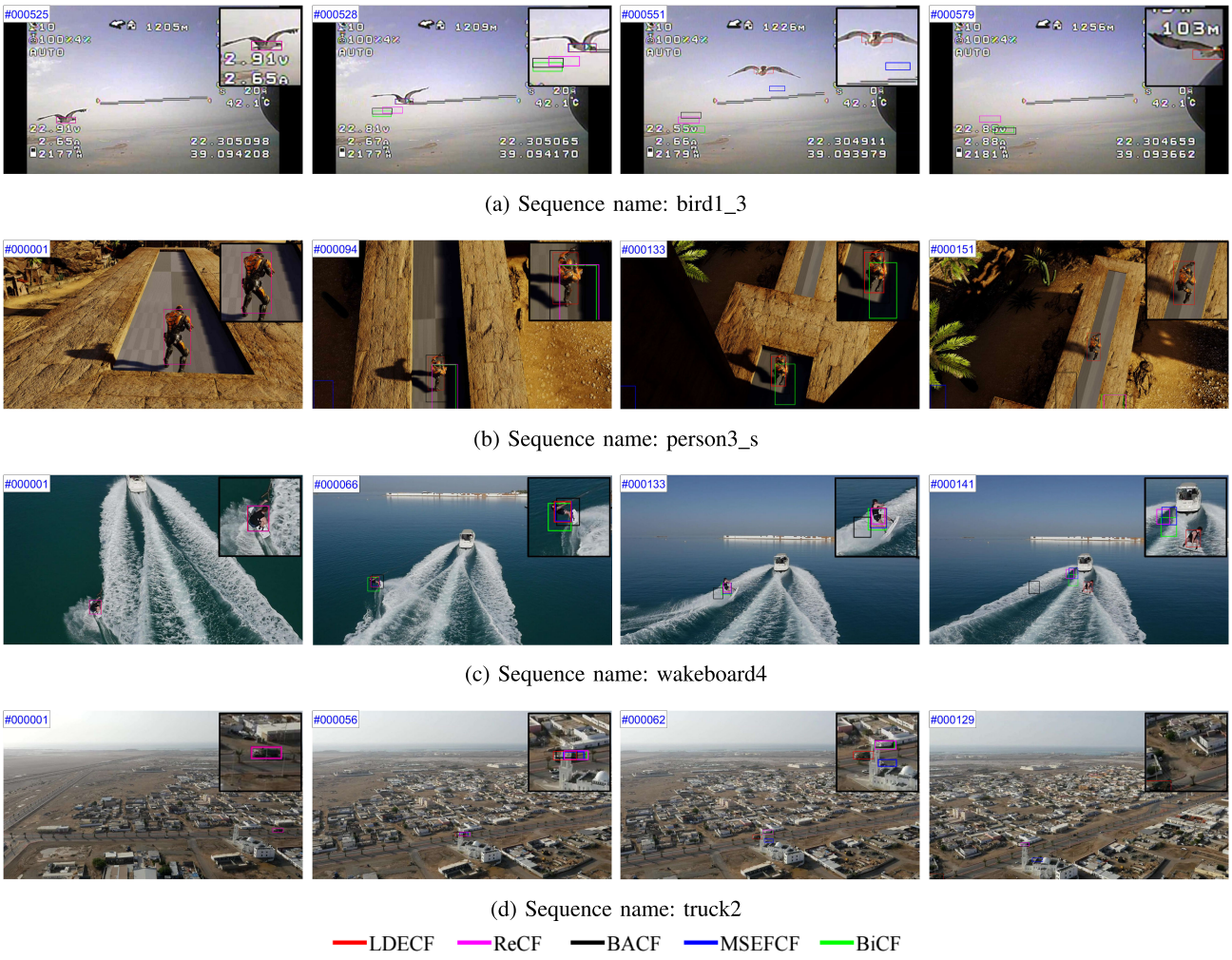


Fig. 7. Qualitative comparison using hand-crafted features trackers on four sequences of UAV123@10fps dataset.

As shown in Fig. 5, our LDECF tracker provides the best DP score (0.749), which exceeds 1.4% and 2.5% of the second best tracker RCFL (0.735) and the third best tracker MSEFCF (0.724). Moreover, the best AUC score is also obtained by our LDECF (0.508), surpassing MSEFCF (0.497) by 1.1% and RCFL (0.493) by 1.5%. Furthermore, compared to the baseline method BACF (0.590/0.402), LDECF achieves a substantial improvement of 15.9% in the DP score and 10.6% in the AUC score.

In summary, the LDECF tracker shows exceptional performance on two challenging UAV benchmarks, achieving a significant advantage over 17 other hand-crafted feature-based trackers. LDECF ranks first in both DP and AUC scores on the UAV123@10fps and DTB70 benchmarks. This is attributed to our synergistic strategy of dynamic-sensitivity error and adaptive second-order difference spatial regularization, which secured top performance across most attributes. Additionally, features enhanced by NLM denoising provide sharper target contours, further improving overall performance. A detailed comparison of performance across various attributes is further discussed in the next section. Compared with the baseline BACF, LDECF achieves substantial improvements in DP and AUC scores on both benchmarks, fully validating its effectiveness and superiority in overall performance.

2) Attribute-based performance: In this section, we further substantiate the effectiveness of our LDECF under dynamic conditions by comparing it with other tracking algorithms across nine challenging attributes on UAV123@10fps benchmark: scale variation (SV), aspect ratio change (ARC), low resolution (LR), fast motion (FM), partial occlusion (POC), illumination variation (IV), viewpoint change (VC), camera motion (CM), and similar objects (SOB). As demonstrated in Table I, our LDECF consistently outperforms competing methods, achieving the highest DP score across all evaluated attributes. Furthermore, in terms of AUC score, our LDECF surpasses other advanced trackers in six of the nine attributes, underscoring its robustness in complex scenarios.

We also conduct a comparative analysis against 17 other algorithms, focusing on challenging attributes within the DTB70 benchmark. As shown in Fig. 6, the evaluation spans eight specific attributes: fast camera motion (FCM), deformation (DEF), aspect ratio variation (ARV), in-plane rotation (IPR), out-of-plane rotation (OPR),occlusion (OCC), motion blur (MB) and similar objects around (SOA). Our findings reveal that the proposed LDECF consistently outperforms other trackers, achieving superior DP and AUC scores across all specified attributes, particularly in out-of-plane rotation, where its scores 4.6% higher in DP and 2.4% higher in AUC

TABLE II
PERFORMANCE COMPARISONS OF 20 ADVANCED TRACKERS USING DEEP FEATURES (MARKED BY †) AND LIGHTWEIGHT TRACKERS ON THE UAVDT BENCHMARK. TOP THREE IN PREC, SUCC, AND FPS ARE HIGHLIGHTED IN RED, GREEN, AND BLUE, RESPECTIVELY

	Method	Venue	Prec.	Succ.	GPU	CPU
DCF-based	BACF [43]	17'ICCV	0.706	0.442	-	51.9
	DeepSTRCF† [18]	18'CVPR	0.667	0.437	6.8	-
	MCCT† [45]	18'CVPR	0.671	0.437	7.9	-
	TADT† [59]	19'CVPR	0.677	0.431	32.3	-
	UDT† [60]	19'CVPR	0.674	0.442	73.3	-
	ASRCF† [46]	19'CVPR	0.700	0.437	14.1	-
	MN_ECO† [61]	20'ACM	0.691	0.435	30.6	-
	fECO† [62]	20'TIP	0.699	0.415	20.6	-
	MEVT† [63]	21'IS	0.691	0.448	3	-
	CCF† [64]	23'TCSVT	0.736	0.467	45	-
	BSTCF† [65]	23'AI	0.685	0.441	19	-
	LDECF	**Ours**	**0.744**	**0.448**	**-**	**44.9**
CNN-based	HiFT [51]	21'ICCV	0.652	0.475	160.3	-
	LUDT+ [66]	21'IJCV	0.701	0.406	59.4	-
	F-SiamFC++ [67]	22'IJCNN	0.794	0.555	255.4	51.6
	TCTrack [8]	22'CVPR	0.725	0.530	139.6	-
	ABDNet [68]	23'RAL	0.755	0.553	130.2	-
ViT-based	STARK-ST101 [69]	21'ICCV	0.704	0.469	30.2	-
	HiT [70]	23'ICCV	0.623	0.471	237.7	43.2
	BDTrack-ViT [71]	24'arXiv	0.789	0.573	235.2	56.9
	TATTrack-ViT [72]	24'TGRS	0.805	0.586	203.3	47.7

than the second-best tracker. This highlights the robustness and efficacy of our LDECF in managing complex tracking scenarios.

Most of challenging attributes generally lead to dynamic variations in object appearance and changes in the environment, e.g., SV, ARC/ARV, VC, IV, FM/FCM/CM, IPR, OPR and DEF, our LDECF yields substantial performance and surpasses the second-best result by a large margin. This is attributed to the dynamic-sensitivity error strategy of the response level. By fully leveraging historical information, our tracker effectively adapts to response variations between forward-sensitive tracking and historically-sensitive backtracking, therefore enhancing its adaptability in conditions with sudden environmental changes. Notably, LDECF also reduces interference from similar objects in motion, as our method effectively distinguishes variations in their movement rates, thus improving performance on the SOB and SOA attributes. Additionally, features enhanced by NLM denoising boost performance on the LR and PO attributes. For other attributes, such as OCC/FOC, OV, and BC, our LDECF does not outperform other advanced trackers. Firstly, for the OCC/FOC attribute, the tracker may struggle to utilize historical information during extended occlusions, limiting the effectiveness of the dynamic-sensitivity error strategy, especially when the target is completely obscured, which typically results in a significant performance drop. Secondly, for the OV attribute, most trackers face difficulty in re-locating the target once it exits the camera's field of view. Lastly, for the BC attribute, complex or dynamic background changes often mislead the tracking algorithm, causing it to shift focus from the target to incorrect areas. However, compared to the baseline method BACF, LDECF still achieves satisfying improvements in DP and AUC scores for these attributes.

3) Qualitative analysis: Extensive evaluations are conducted to demonstrate the performance of the proposed LDECF tracker relative to other algorithms (MSEFCF, ReCF, BiCF, and BACF) across four challenging sequences. These sequences (bird1_3, person3_s, and wakeboard4) exhibit highly dynamic variations, including scale variation (SV), aspect ratio change (ARC), fast motion (FM), viewpoint change (VC), and camera motion (CM), while truck2 frequently features partial occlusion. As illustrated in Fig. 7, all trackers experience tracking drift and fail to consistently track the target when it alters speed. This result highlights the challenges posed by rapid and variable target movements in tracking scenarios. Specifically, in the first video sequence, the target's sudden large acceleration leads to significant displacement, resulting in sequential tracking failures by BACF, BiCF, ReCF, and MSEFCF. In contrast, our LDECF maintains stability throughout these rapid movements.

In the second sequence, BACF struggles to adapt to a severe scale change early in the video. Subsequently, variations in the target's speed, along with the similarity in texture and color between the target and the background, begin to confuse the other trackers. This eventually leads to the loss of the target when the illumination conditions change, further complicating the tracking process. However, our LDECF consistently and precisely tracks the target, maintaining accurate location even amidst complex movements and challenging environments.

In the third video sequence, all trackers except LDECF fail to maintain tracking when the target undergoes a large occlusion while executing a rapid rotation stunt. Similarly, in the final video sequence, despite the target moving at a uniform speed, a significant occlusion occurs, and the other trackers subsequently lose the target. These scenarios underscore the robustness of our LDECF in handling complex tracking challenges. More representative real-time tracking results are visualized in Fig. 9 to demonstrate the discriminative capability of our LDECF against highly dynamic background and object.

D. Comparison With Trackers Using Deep Learning

In this section, we evaluate the tracking performance and efficiency of the proposed LDECF tracker by comparing it with 20 state-of-the-art trackers that utilize deep features or lightweight architectures on the UAVDT benchmark. These deep learning methods include DCF-based trackers such as DeepSTRCF [18], MCCT [45], TADT [59], UDT [60], ASRCF [46], MN_ECO, fECO [62], MEVT [63], CCF [64], and BSTCF [65]; CNN-based trackers including HiFT [51], LUDT+ [66], F-SiamFC++ [67], TCTrack [8], ABDNet [68], and DRCI [80]; and ViT-based trackers such as STARK-ST101 [69], HiT [70], BDTrack-ViT [71], and TATTrack-ViT [72]. Table II reports the DP and AUC scores of our LDECF tracker and the other trackers used for comparison, where the GPU and CPU speeds shown for CNN-based and

TABLE III
ABLATION STUDY OF THREE MODULES BASED ON BASELINE ON THE DTB70 BENCHMARK

Setting	Prec.	Succ.	BC	DEF	IPR	FCM	SV	SOA	ARV	MB	OCC
Baseline (BACF)	0.590	0.402	0.337	0.302	0.371	0.435	0.392	0.411	0.273	0.412	0.348
Baseline+DE	0.738	0.509	0.438	0.471	0.478	0.543	0.503	0.538	0.424	0.528	0.467
Baseline+ADSR	0.732	0.504	0.440	0.469	0.474	0.538	0.491	0.538	0.420	0.529	0.468
Baseline+DE+ADSR	0.744	0.505	0.448	0.469	0.476	0.545	0.491	0.542	0.425	0.531	0.474
Baseline+DE+NLMD	0.740	0.502	0.438	0.451	0.472	0.542	0.482	0.543	0.414	0.527	0.476
LDECF (ours)	0.749↑	0.508↑	0.464↑	0.466↑	0.480↑	0.549↑	0.489↑	0.543↑	0.425↑	0.539↑	0.476↑

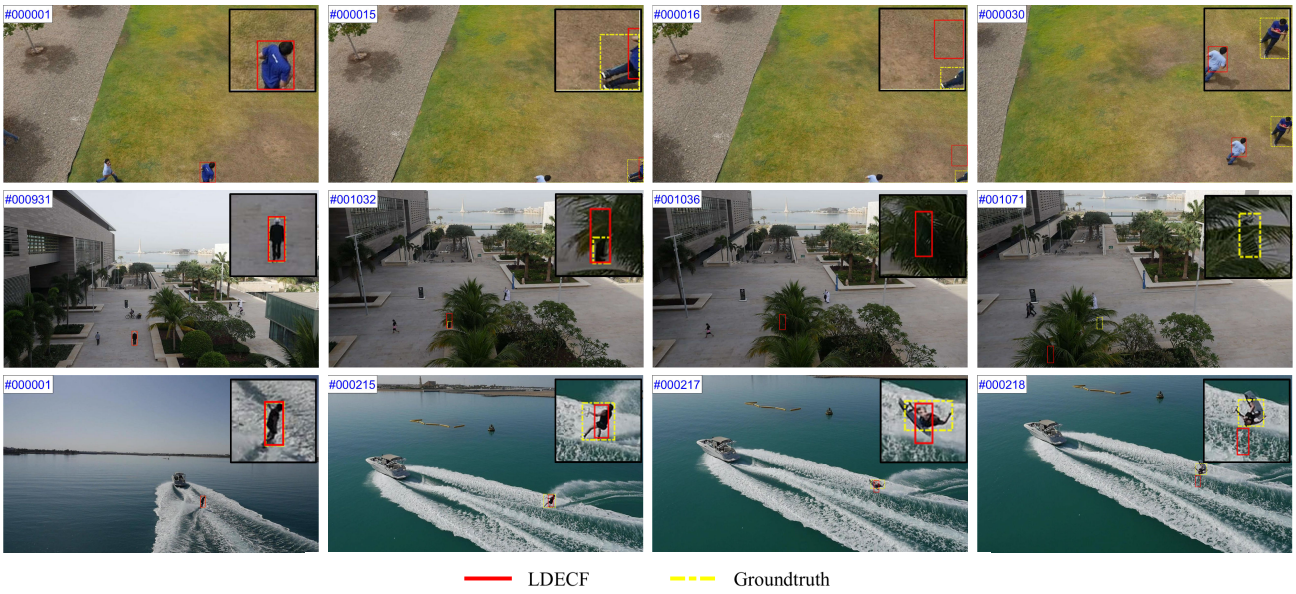


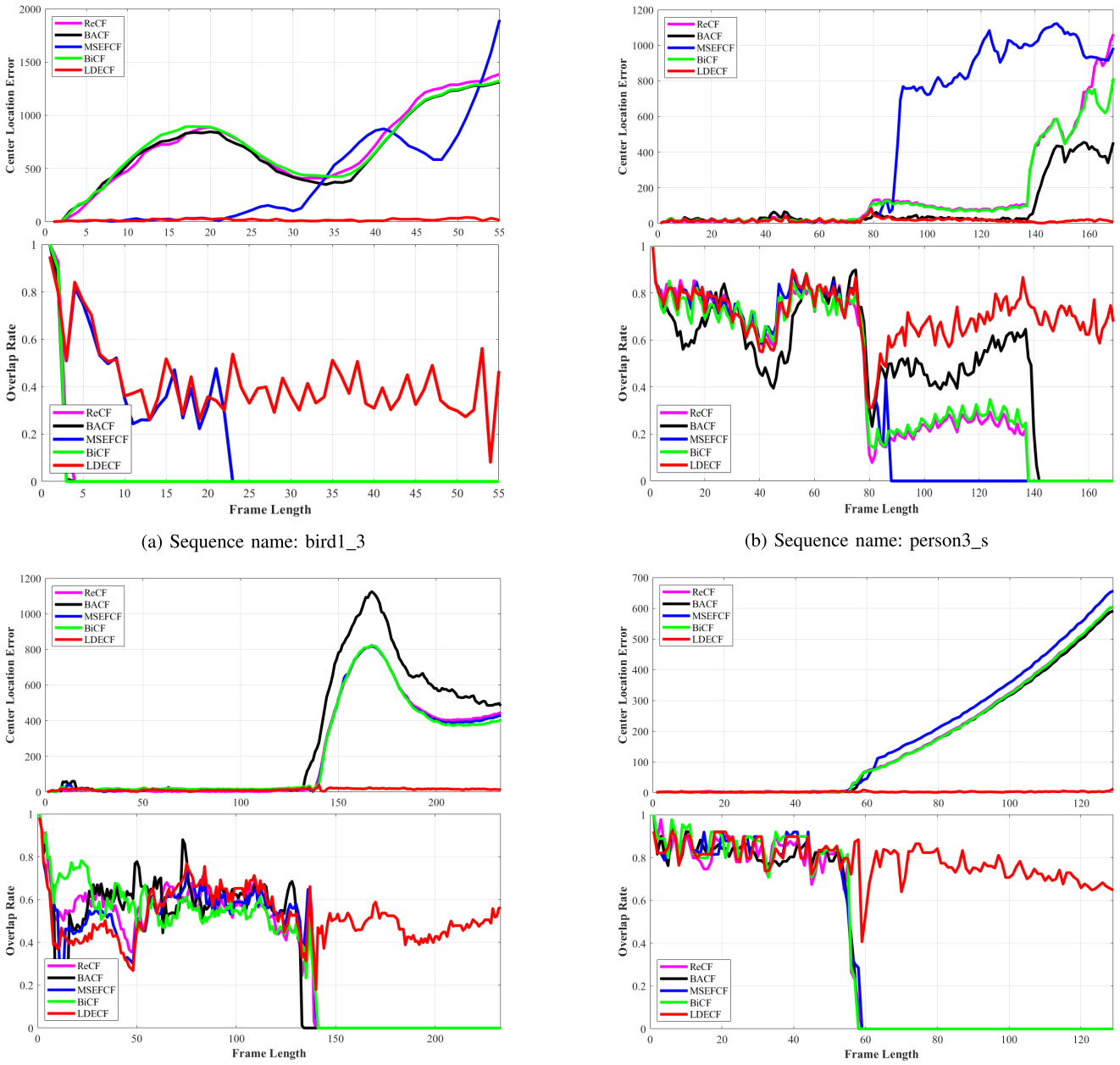
Fig. 8. Failure cases of the proposed LDECF tracker. The first, second, and third rows display sequences for person9, person19_3, and wakeboard2 from UAV123@10fps.

ViT-based trackers represent the average FPS across five UAV tracking benchmarks, namely DTB70, UAVDT, VisDrone2018 [81], UAV123 [1], and UAV123@10fps. Each of the three categories of trackers exhibits unique characteristics and is suitable for specific applications. Firstly, among DCF-based methods, while their overall precision (Prec.) and success rates (Succ.) are slightly lower than those of other approaches, their lower computational complexity makes them ideal for resource-constrained real-time applications. The proposed LDECF stands out in this category, achieving the highest precision of 0.744 and the second-highest success rate of 0.448, delivering the best performance among DCF-based methods. Additionally, LDECF operates at 44.9 FPS on a single CPU, balancing high accuracy with efficiency, which makes it an excellent choice for real-time tracking. Other DCF-based methods, such as UDT and CCF, show slightly lower Prec. Compared to LDECF, although they offer a minor FPS advantage, they are less suitable for complex scenarios. In the CNN-based method, F-SiamFC++ demonstrates a solid balance, with a precision of 0.794 and a success rate of 0.555, achieving top-tier accuracy and efficiency with 255.4 FPS on GPU and 51.6 FPS on CPU. Finally, ViT-based methods maintain excellent performance and speed, except for STARK-ST101. TATrack-ViT achieves a precision of 0.805 and a success rate of 0.586, with frame rates of 203.3 FPS on GPU and 47.7 FPS on CPU, catering to high-precision requirements in complex scenarios. BDTrack-ViT also delivers strong performance, with a precision of 0.789 and a success rate of 0.573, achieving 235.2 FPS on GPU and 56.9 FPS on CPU, balancing efficiency with accuracy. These methods are ideal for environments with abundant computational resources, meeting the demands of high precision and real-time performance.

E. Ablation Study

In this section, we conduct an in-depth analysis of the impact of each component in the proposed method (LDECF): dynamic-sensitivity (DE), adaptive second-order difference spatial regularization (ADSR), and NLM denoising (NLMD). We set up four variants on the DTB70 benchmark for evaluation: Baseline+DE, Baseline+ADSR, Baseline+DE+ADSR, and Baseline+DE+NLMD. The Baseline refers to the BACF method.

Table III and Fig. 10 present AUC scores for each module, allowing for several key conclusions to be drawn: 1) the proposed LDECF tracker shows marked improvements across the board, particularly in deformation (DEF), aspect ratio variation (ARV), and similar objects around (SOA). 2) The enhancements, including DE, ADSR, and NLMD, consistently



(a) Sequence name: bird1_3

(b) Sequence name: person3_s

(c) Sequence name: wakeboard4

(d) Sequence name: truck2

Fig. 9. The real-time CLE and overlap curves in four challenging sequences of UAV123@10fps dataset.

improve upon the baseline or Baseline+DE across all metrics. 3) Among the evaluated modules, the DE module plays a crucial role in enhancing tracking performance relative to the others, while the ADSR module complements it to further boost results. Additionally, the NLMD module effectively improves performance in the LR attribute. As shown in Table III, the DE module outperforms in most attributes, although its performance is lower than that of the ADSR module in certain attributes, such as BC, MB and OCC. This pattern is similarly reflected in Fig. 10.

F. Parameter Analysis

To elucidate the impact of key parameters on performance, we conduct a sensitivity analysis of γ, $\lambda_1$and λ_2 on the DTB70 benchmark. Firstly, as shown in Fig. 11a, the DP score exhibit an overall upward trend as γ rises from 0.07 to 0.14, peaking at $\gamma = 0.14$, and then gradually declining. The AUC scores on both benchmarks follow a similar pattern, but fluctuate at a lower level. Secondly, Fig. 11b depicts the fluctuation of λ_1 as it increases from 0.7 to 1.7 in increments of 0.1. Both DP and AUC scores peak at $\lambda_1 = 1.2$. Except for an abrupt change at $\lambda_1 = 1.1$, the variation in the two metrics remains relatively stable. Lastly, Fig. 11c shows the relationship between the two scores and the parameter λ_2. Specifically, as λ_2 increases from 0.4×10^{-3} to 1.4×10^{-3}, the DP scores show a noticeable upward trend, reaching a peak before gradually declining as the parameter increases further. Simultaneously, the AUC score demonstrates a similar trend but also with smaller fluctuations. In brief, when γ and λ_2 are

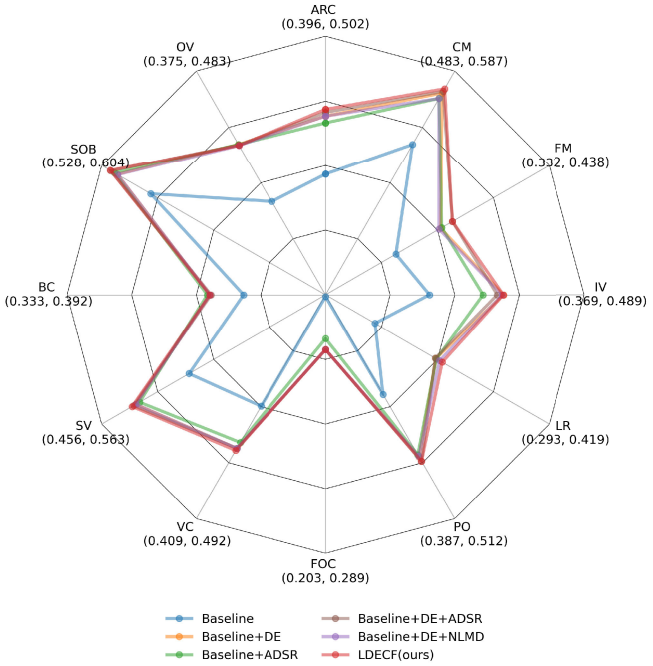
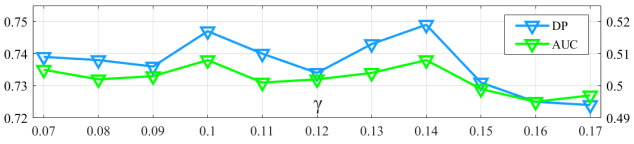
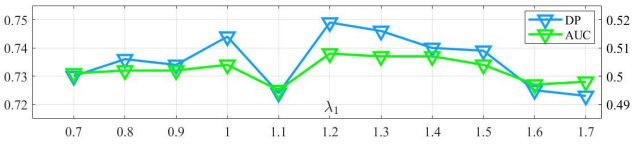


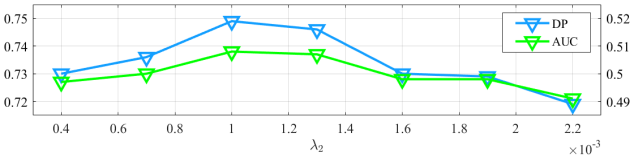
Fig. 10. Ablation study of three modules on baseline (BACF) on the UAV123@10fps benchmark, with the numerical intervals of the attribute axis displayed below each attribute name.



(a) Impact of parameter γ



(b) Impact of parameter λ_1



(c) Impact of parameter λ_2

Fig. 11. Key parameters analysis on DTB70.

set within a certain range, the increases in these key parameters initially have a positive impact on both metrics. However, beyond a certain point, further increases lead to performance degradation.

G. Failure Cases

The proposed LDECF outperforms other advanced DCF tracking algorithms across three UAV datasets. However, due to the complexity of UAV tracking scenarios, LDECF still has certain limitations. Fig. 8 illustrates failure cases of LDECF in three challenging sequences. In the sequence person9, only the target's upper body appears in the initial frame, while only the lower body remains visible by frame 16. When the target fully reappears in the field of view by frame 30, LDECF fails to correctly identify it, instead erroneously tracking other objects with similar motion trajectories. Consequently, incorrect historical information is used in filter learning when the target is out of view, significantly reducing the tracker's discriminative capability. In frame 1032 of the sequence person19_3, the target is occluded to the extent that only the lower legs are visible, and it remains fully obscured until frame 1071. With minimal or absent visual information, the appearance features are compromised by irrelevant information, impacting filter training. In the wakeboard2 sequence, the target experiences rapid out-of-plane rotation between frames 215 and 218, causing substantial changes in aspect ratio and viewing angle. These abrupt transformations result in significant appearance variations that the LDECF tracker fails to handle effectively.

V. Discussion

As discussed, existing DCF-based methods often fail to: 1) account for the stochastic noise generated during the photon counting process; 2) handle the inconsistency of the target and background motion rate and make effective use of the joint historical information of features, filters and response maps. To address these challenges, we propose the LDECF tracker. First of all, we apply the NLM algorithm to denoise the image before feature extraction, enhancing the discriminative power of the target's shape. Second of all, we introduce the concept of dynamic-sensitivity error to learn the switching difference of motion speed in a highly dynamic scene by paying attention to the response level of both forward tracking and historical backtracking. Finally, to better align the spatial regularizer with this strategy, we introduce an adaptive second-order difference spatial regularization term. Notably, this combined strategy allows for the simultaneous optimization of both filter and spatial regularization weights.

To validate our approach and demonstrate the effectiveness of LDECF, we conducted extensive experiments. Section III explores the effectiveness of the denoising method for optimizing feature input and the existence of dramatic fluctuation of response differences in highly dynamic scenarios. Sections IV-C and IV-D compare state-of-the-art trackers using handcrafted features and those using deep learning, with qualitative analyses intuitively demonstrating how LDECF efficiently leverages the joint historical information to exhibit strong robustness in highly dynamic scenarios. Further, the ablation experiments in Section IV-E verify the impact of each component setting.

Indeed, LDECF has the following limitations: 1) The NLM algorithm entails significant computational overhead as the search and neighborhood windows increase in size. 2) When the target moves completely out of the camera's field of view and does not reappear within a short period, the tracker cannot utilize the joint historical information to detect the target's reappearance, as the lack of necessary information to support filter learning.

TABLE IV
LIST OF THE MAIN ACRONYMS USED IN THIS PAPER

Acronym	Meaning
ADMM	an algorithm (Alternating Direction Method of Multipliers) that solves convex optimization problems by breaking them into smaller pieces
ADSR	the proposed adaptive second-order difference spatial regularizer that suppresses abrupt changes from background noise
ARC	aspect ratio change
ARV	aspect ratio variation
BC	background clutter
CF	correlation filter
CM	camera motion
CN	color names, a handcrafted feature designed to provide a representation of colors in an image
CNN-based	a type of capturing the semantic cues from raw images by using the Convolutional Neural Networks
DCF-based	an online visual tracker that employs a Discriminative Correlation Filter for visual object tracking
DE	Dynamic Sensitivity Error, the proposed strategy that captures the inconsistencies in target change rates in highly dynamic environments
DEF	deformation
FCM	fast camera motion
FFT	an algorithm that computes the Discrete Fourier Transform (DFT) of a sequence, or its inverse (IDFT)
FM	fast motion
FOC	full occlusion
HOG	histogram of oriented gradients, a handcrafted feature that provides a gradient-based representation of edges and shapes in an image
IPR	in-plane rotation
IV	illumination variation
LR	low resolution
MB	motion blur
NLMD	a denoising method based on Non-Local Means, which leverages the similarity between image patches
OCC	occlusion
OV	out-of-view
POC	partial occlusion
SOA	similar objects around
SOB	similar objects
SV	scale variation
UAV	unnamed aerial vehicle
VC	viewpoint change
ViT-based	a visual model based on the architecture of a transformer originally designed for text-based tasks

Fortunately, several strategies can be considered to address these limitations. First, employing lightweight models to extract deep features and combining with handcrafted features can enhance the learning of appearance models while maintaining computational efficiency. Second, designing a structure that allows the tracker to temporarily disregard the joint historical information of a lost target provides a practical solution. When the target reappears, this structure enables the tracker to retrieve pre-forgotten information, leveraging the observation that reappearance locations often do not deviate significantly from disappearance locations. Moreover, techniques developed in related fields offer valuable insights into tackling similar challenges. For instance, the barrier Lyapunov function method has been successfully applied to handle state constraints in switched nonlinear systems, as demonstrated in [82]. Similarly, the improved Lyapunov–Krasovskii function presented in [83] offers an effective framework for managing time-delay terms and high-order dynamics, which could inspire future advancements in adaptive tracking under dynamic conditions.

VI. CONCLUSION

In this paper, a dynamic-sensitivity enhanced correlation filter with adaptive second-order difference spatial regularization is proposed to address the inconsistency in dynamic video motion rates for real-time UAV tracking. Our method is driven by the template responses between forward tracking and historical backtracking. This algorithm facilitates filter updates in dynamic environments by striking a balance between maintaining consistency with previous filter templates and adapting flexibly to rapid target changes. Additionally, we introduce the Non-Local Means algorithm before the feature extraction phase of the correlation filter algorithm to denoise the stochastic noise, enhancing the contours of the extracted target features. Extensive experiments demonstrate that our method achieves optimal tracking performance across all three commonly used UAV datasets. In the future, we will explore lightweight ViT variants to ensure computational efficiency for real-time UAV applications. Since this work focuses solely on tracking with general shallow feature target representation, our future work will prioritize: 1) incorporating adaptive constraint-aware techniques and managing inter-frame delays; 2) developing appropriate scale estimation strategies; and 3) efficient CF learning of deep features.

We provide a comprehensive list of abbreviations and their corresponding meanings in this section. Table IV lists key abbreviations used throughout this paper as well as the definitions of tracking attributes commonly referenced in UAV tracking evaluations.

REFERENCES

[1] M. Mueller, N. G. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 445–461.

[2] Y. Wiseman, "Real-time monitoring of traffic congestions," in *Proc. IEEE Int. Conf. Electro Inf. Technol. (EIT)*, May 2017, pp. 501–505.

[3] 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. Vehicles*, vol. 8, no. 9, pp. 4155–4168, Sep. 2023.

[4] R. Bonatti, C. Ho, W. Wang, S. Choudhury, and S. Scherer, "Towards a robust aerial cinematography platform: Localizing and tracking moving targets in unstructured environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 229–236.

[5] M. Monajjemi, J. Bruce, S. A. Sadat, J. Wawerla, and R. Vaughan, "UAV, do you see me? Establishing mutual attention between an uninstrumented human and an outdoor UAV in flight," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 3614–3620.

[6] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. ECCV Workshops*, Amsterdam, The Netherlands, Oct. 2016, pp. 850–865.

[7] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[8] Z. Cao, Z. Huang, L. Pan, S. Zhang, Z. Liu, and C. Fu, "TCTrack: Temporal contexts for aerial tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14778–14788.

[9] X. Wang, D. Zeng, Q. Zhao, and S. Li, "Rank-based filter pruning for real-time UAV tracking," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2022, pp. 1–6.

[10] W. Wu, P. Zhong, and S. Li, "Fisher pruning for real-time UAV tracking," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–7.

[11] M. Liu, Y. Wang, Q. Sun, and S. Li, "Global filter pruning with self-attention for real-time UAV tracking," in *Proc. BMVC*, 2022, p. 861.

[12] J. Kugarajeevan, T. Kokul, A. Ramanan, and S. Fernando, "Transformers in single object tracking: An experimental survey," *IEEE Access*, vol. 11, pp. 80297–80326, 2023.

[13] Y.-F. Yu, Y. Zhang, L. Chen, P. Ge, and C. L. P. Chen, "Multi-scale enhanced features correlation filters learning with dual second-order difference for UAV tracking," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 2, pp. 3232–3245, Feb. 2024.

[14] H. Zhang, Y. Li, H. Liu, D. Yuan, and Y. Yang, "Feature block-aware correlation filters for real-time UAV tracking," *IEEE Signal Process. Lett.*, vol. 31, pp. 840–844, 2024.

[15] 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, Apr. 2023.

[16] L. Wang et al., "Auto-perceiving correlation filter for UAV tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5748–5761, Sep. 2022.

[17] 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, Oct. 2021.

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

[19] Y. Zhang, Y.-F. Yu, K.-K. Huang, and Y. Wang, "Channel attentional correlation filters learning with second-order difference for UAV tracking," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

[20] 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.

[21] J. Chen, T. Xu, B. Huang, Y. Wang, and J. Li, "ARTracker: Compute a more accurate and robust correlation filter for UAV tracking," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[22] B. He, F. Wang, X. Wang, H. Li, F. Sun, and H. Zhou, "Temporal context and environment-aware correlation filter for UAV object tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5630915.

[23] 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.

[24] C. Fu, J. Jin, F. Ding, Y. Li, and G. Lu, "Spatial reliability enhanced correlation filter: An efficient approach for real-time UAV tracking," *IEEE Trans. Multimedia*, vol. 26, pp. 4123–4137, 2024.

[25] J. Lin, J. Peng, and J. Chai, "Real-time UAV correlation filter based on response-weighted background residual and spatio-temporal regularization," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

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

[27] A. Buades, B. Coll, and J.-M. Morel, "Non-local means denoising," *Image Process. On Line*, vol. 1, pp. 208–212, Sep. 2011.

[28] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Jun. 2017, pp. 5998–6008.

[29] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision—ECCV 2020*. U.K.: Springer, Aug. 2020, pp. 213–229.

[30] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[31] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2021, pp. 10347–10357.

[32] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[33] L. Yang, S. Gu, C. Shen, X. Zhao, and Q. Hu, "Skeleton neural networks via low-rank guided filter pruning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7197–7211, Dec. 2023.

[34] J. Mao, H. Yang, A. Li, H. Li, and Y. Chen, "TPrune: Efficient transformer pruning for mobile devices," *ACM Trans. Cyber Phys. Syst.*, vol. 5, no. 3, pp. 1–22, 2021.

[35] Y. Li et al., "EfficientFormer: Vision transformers at MobileNet speed," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, Jan. 2022, pp. 12934–12949.

[36] Y. Rao, W. Zhao, B. Liu, J. Lu, J. Zhou, and C. J. Hsieh, "DynamicViT: Efficient vision transformers with dynamic token sparsification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2021, pp. 13937–13949.

[37] H. Yin, A. Vahdat, J. M. Alvarez, A. Mallya, J. Kautz, and P. Molchanov, "A-ViT: Adaptive tokens for efficient vision transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10799–10808.

[38] S. Li, Y. Yang, D. Zeng, and X. Wang, "Adaptive and background-aware vision transformer for real-time UAV tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 13989–14000.

[39] Y. Li, M. Liu, Y. Wu, X. Wang, X. Yang, and S. Li, "Learning adaptive and view-invariant vision transformer for real-time UAV tracking," in *Proc. 41st Int. Conf. Mach. Learn.*, 2024, pp. 1–32.

[40] 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.*, Jun. 2010, pp. 2544–2550.

[41] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 702–715.

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

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

[44] Y. Yu, L. Chen, H. He, J. Liu, W. Zhang, and G. Xu, "Second-order spatial–temporal correlation filters for visual tracking," *Mathematics*, vol. 10, no. 5, p. 684, Feb. 2022.

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

[46] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4665–4674.

[47] F. Zhang, S. Ma, Y. Zhang, and Z. Qiu, "Perceiving temporal environment for correlation filters in real-time UAV tracking," *IEEE Signal Process. Lett.*, vol. 29, pp. 6–10, 2022.

[48] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893.

[49] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1512–1523, Jul. 2009.

[50] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1401–1409.

[51] 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. (ICCV)*, Oct. 2021, pp. 15437–15446.

[52] H. Zhang, G. Liu, Y. Zhang, and Z. Hao, "Robust multi-model visual tracking with distractor-aware template-coupled correlation filters joint learning," *IEEE Trans. Multimedia*, vol. 26, pp. 1813–1828, 2024.

[53] P. Liu, G. Li, W. Zhao, and X. Tang, "A coupling method of learning structured support correlation filters for visual tracking," *Vis. Comput.*, vol. 40, no. 1, pp. 181–199, Jan. 2024.

[54] 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. Autom. (ICRA)*, May 2020, pp. 2365–2371.

[55] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 2756–2759.

[56] J. Wang, Y. Guo, Y. Ying, Y. Liu, and Q. Peng, "Fast non-local algorithm for image denoising," in *Proc. Int. Conf. Image Process.*, 2006, pp. 1429–1432.

[57] S. Li and D. Yeung, "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2017, vol. 31, no. 1, pp. 4140–4146.

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

[59] 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. (CVPR)*, Jun. 2019, pp. 1369–1378.

[60] 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. (CVPR)*, Jun. 2019, pp. 1308–1317.

[61] 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.

[62] 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.

[63] 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, Feb. 2021.

[64] B. Wang, W. Li, B. Zhang, Y. Liu, and J. Du, "Correlation filters for uav online tracking based on complementary appearance model and reversibility reasoning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 5, pp. 3983–3997, May 2024.

[65] 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.

[66] 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, no. 2, pp. 400–418, Feb. 2021.

[67] I. Sosnovik, A. Moskalev, and A. Smeulders, "Scale equivariance improves Siamese tracking," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 2764–2773.

[68] H. Zuo, C. Fu, S. Li, K. Lu, Y. Li, and C. Feng, "Adversarial blur-deblur network for robust uav tracking," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 1101–1108, 2023.

[69] 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. (ICCV)*, Oct. 2021, pp. 10428–10437.

[70] B. Kang, X. Chen, D. Wang, H. Peng, and H. Lu, "Exploring lightweight hierarchical vision transformers for efficient visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 9612–9621.

[71] Y. Wu et al., "Learning motion blur robust vision transformers with dynamic early exit for real-time UAV tracking," 2024, *arXiv:2407.05383*.

[72] S. Li, X. Yang, X. Wang, D. Zeng, H. Ye, and Q. Zhao, "Learning target-aware vision transformers for real-time UAV tracking," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4705718.

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

[74] 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.

[75] 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.

[76] 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. (CVPR)*, Jun. 2020, pp. 11920–11929.

[77] 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.

[78] 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. Autom.*, May/Jun. 2021, pp. 503–509.

[79] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1396–1404.

[80] D. Zeng, M. Zou, X. Wang, and S. Li, "Towards discriminative representations with contrastive instances for real-time UAV tracking," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2023, pp. 1349–1354.

[81] L. Wen et al., "VisDrone-SOT2018: The vision meets drone single-object tracking challenge results," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, Jan. 2019, pp. 469–495.

[82] H. Wang, W. Liu, and M. Tong, "Adaptive fuzzy fast finite-time output-feedback tracking control for switched nonlinear systems with full-state constraints," *IEEE Trans. Fuzzy Syst.*, vol. 32, no. 3, pp. 958–968, Mar. 2024.

[83] K. Xu, H. Wang, and P. X. Liu, "Adaptive fixed-time control for high-order stochastic nonlinear time-delay systems: An improved Lyapunov–Krasovskii function," *IEEE Trans. Cybern.*, vol. 54, no. 2, pp. 776–786, Feb. 2024.

Yu-Feng Yu (Member, IEEE) received the Ph.D. degree in statistics from Sun Yat-sen University, Guangzhou, China, in 2017. 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. He is currently an Associate Professor with the Department of Statistics, Guangzhou University, Guangzhou. His research interests include image processing, statistical optimization, pattern recognition, machine learning, and computer vision.

Zhongsen Chen received the B.Sc. degree in information and computing science from Shaoguan University in 2022. He is currently pursuing the M.Sc. degree in applied statistics with Guangzhou University. His research interests include visual tracking and machine learning.

Yang Zhang received the B.Sc. degree in applied mathematics and the M.Sc. degree in statistics from Guangzhou University in 2020 and 2024, respectively. He is currently pursuing the Ph.D. degree in statistics with the University of Macau. His research interests include visual tracking and machine learning.

Chuanbin Zhang received the B.S. degree in automation from Southwest Jiaotong University, Chengdu, China, in 2013, the M.S. degree in computer science and technology from Sun Yat-sen University, Guangzhou, China, in 2016, and the Ph.D. degree in computer science from the University of Macau, Macau, SAR, China, in 2024. He is currently a Lecturer with the School of Computer Science and Software, Zhaoqing University, Zhaoqing, China. His research interests include artificial intelligence, machine learning, fuzzy clustering, Bayesian methods, image processing, and their applications.

Weiping Ding (Senior Member, IEEE) received the Ph.D. degree in computer science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2013. From 2014 to 2015, he was a Post-Doctoral Researcher with the Brain Research Center, National Chiao Tung University, Hsinchu, Taiwan. In 2016, he was a Visiting Scholar with the National University of Singapore, Singapore. From 2017 to 2018, he was a Visiting Professor with the University of Technology Sydney, Australia. He has published over 360 articles, including over 170 IEEE TRANSACTIONS. His 20 authored/co-authored papers have been selected as ESI Highly Cited Papers. He has co-authored five books. He holds more 50 approved invention patents, including three U.S. patents and one Australian patent. His main research interests include deep neural networks, granular data mining, and multimodal machine learning. He serves as an Associate Editor/Area Editor/Editorial Board Member for more than ten international prestigious journals, such as IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON FUZZY SYSTEMS, IEEE/CAA JOURNAL OF AUTOMATICA SINICA, IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON INTELLIGENT VEHICLES, IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE, *Information Fusion*, *Information Sciences*, *Neurocomputing*, *Applied Soft Computing*, *Engineering Applications of Artificial Intelligence*, and *Swarm and Evolutionary Computation*. He was/is the Leading Guest Editor of Special Issues in several prestigious journals, including IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Information Fusion*, and *Information Sciences*. He is the Co-Editor-in-Chief of three international journals, such as *Journal of Artificial Intelligence and Systems*, *Artificial Intelligence Advances*, and *Sustainable Machine Intelligence Journal*. He ranked within the top 2% ranking of scientists in the World by Stanford University (2020–2024).