Multi-Regularized Correlation Filter for UAV Tracking and Self-Localization

Junjie Ye, *Graduate Student Member, IEEE*, Changhong Fu, *Member, IEEE*, Fuling Lin, *Graduate Student Member, IEEE*, Fangqiang Ding, *Student Member, IEEE*, Shan An, *Member, IEEE*, and Geng Lu, *Member, IEEE*

***Abstract*—As a sort of model-free tracking approach, discriminative correlation filter (DCF)-based trackers have shown prominent performance in unmanned aerial vehicle (UAV) tracking. Nevertheless, typical DCFs acquire all samples oriented to filter training merely from the current frame by cyclic shift operation in the spatial domain but ignore the consistency between samples across the timeline. The lack of temporal cues restricts the performance of DCFs under object appearance variations arising from object/UAV motion, scale variations, and viewpoint changes. Besides, many existing methods commonly neglect the channel discrepancy in object position estimation and generally treat all channels equally, thus limiting the further promotion of the tracking discriminability. To these concerns, this work proposes a novel tracking approach based on a multi-regularized correlation filter (MRCF), i.e., MRCF tracker. By regularizing the deviation of responses and the reliability of channels, the tracker enables smooth response variations and adaptive channel weight distributions simultaneously, leading to favorable adaption to object appearance variations and enhancement of discriminability. Exhaustive experiments on five authoritative UAV-specific benchmarks validate the competitiveness and efficiency of MRCF against top-ranked trackers. Furthermore, we apply our proposed tracker to monocular UAV self-localization under air–ground robot coordination. Evaluations indicate the practicability of the presented method in UAV localization applications.**

***Index Terms*—Model-free object tracking, multi-regularized correlation filter (MRCF), unmanned aerial vehicle (UAV), vision-based UAV self-localization.**

Manuscript received November 29, 2020; revised April 24, 2021 and May 20, 2021; accepted May 24, 2021. Date of publication June 16, 2021; date of current version February 1, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61806148 and in part by the Natural Science Foundation of Shanghai under Grant 20ZR1460100. *(Corresponding author: Changhong Fu.)*

Junjie Ye, Changhong Fu, Fuling Lin, and Fangqiang Ding are with the School of Mechanical Engineering, Tongji University, Shanghai 201804, China (e-mail: ye.jun.jie@tongji.edu.cn; changhongfu@tongji.edu.cn; linfuling5@tongji.edu.cn; 1753436@tongji.edu.cn).

Shan An is with the School of Computer Science and Engineering, Beihang University, Beijing 100191, China (e-mail: 1906073@buaa.edu.cn).

Geng Lu is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: lug@tsinghua.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIE.2021.3088366.

Digital Object Identifier 10.1109/TIE.2021.3088366

I. Introduction

VISUAL object tracking has received considerable attention with broad aerial robotic applications, e.g., military patrol [1], air surveillance [2], target following [3], and visual navigation [4]. Despite its wide applications, unmanned aerial vehicle (UAV) tracking assumes no prior knowledge and suffers from many visual uncertainties during inference, such as object/UAV motion, and scale variations (SV).

Most commonly, the top of model-free visual tracking approaches can be categorized as discriminative correlation filter (DCF)-based trackers [5]–[7] and deep-learning-based trackers [8]–[10]. Drawing support from the deep hierarchical semantic features with the usage of high-end GPUs, deep-learning-based approaches [8]–[10] have presented promising accuracy in the field of visual tracking. However, the implementation of this type of method relies on expensive GPUs, which is impractical to UAV platforms that are generally equipped with a common CPU. In addition, the stringent real-time requirements of UAV missions also exacerbate the difficulties. To these concerns, there is an urgent need for robust CPU-based tracking approaches suitable for UAV platforms.

Besides the deep-learning-based approaches, the DCF-based trackers are active and prominent in the field of UAV tracking [7], [11]–[14]. DCF-based trackers can generate plenty of cyclic shift training samples and learn filters in the frequency domain extremely efficiently. As a pioneer of the DCF framework, MOSSE [15] shows competitive tracking performance with a speed of hundreds of frames per second (FPS) on the CPU. Later, Henriques *et al.* [16] incorporate the kernel trick into the DCF and achieve better performance. Effective solutions to boundary effect are extensively developed in [5], [17], and [18]. Various features are introduced to represent the object better, e.g., histogram of oriented gradient (HOG) in [19] and color names (CN) in [20]. Nevertheless, these trackers ordinarily train filters merely utilizing the training patch cropped from the current frame, which results in the lack of temporal cues. Hence, they are troubled by challenging scenes such as similar object around, SV, and object/platform motion.

Regarding this, some studies attempt to draw temporal information into the DCF [7], [11], [21], [22]. Li *et al.* [21] propose to restrict the difference of consecutive filters. ASRCF [22] considers the temporal consistency of the spatial regularizer to cope with sudden appearance changes. In [7], the detection

response in the previous frame is incorporated into the learning phase as the temporal cue for the purpose of aberrance repression. The temporal regularization in these methods enforces the components (filters in [21], spatial regularizers in [22], and responses in [7]) in the current frame to be similar to those in the previous frame by minimizing the first-order difference. These approaches generally enhance the robustness to some extent. However, in situations where object appearance changes dramatically, e.g., motion blur and fast motion, the aforementioned methods directly constrain the components instead of tolerating reasonable changes to adapt to appearance variations, leading to tracking failure.

Drawing from our observation, the extent of object appearance within the time interval of image capture is trivial. Namely, the object appearance model is supposed to change at a nearly same rate among consecutive frames. Furthermore, on account of that responses reflect the change of object appearance implicitly in the DCF framework, the deviation of the responses in continuous time ought to change smoothly. To this end, this work proposes to regularize the deviations of responses by means of constraining the change rate, i.e., the second-order difference. As a result, the tracker can effectively adapt to appearance variations.

In addition, many DCFs treat all channels equally, while some channels with redundant information fail to benefit to locate the object, restricting the further enhancement of the tracking robustness. Previous research works [23]–[27] have established that different feature channels contribute to estimating the object position differently. CSR-DCF [23] incorporates channel reliability via a criterion based on the quality of responses. Xu *et al.* [24] present a group feature selection algorithm to highlight features with enhanced discriminability. Lu *et al.* [25] attempt to learn a weight vector for each feature channel. By means of involving channel-wise information, these studies indeed boost the tracking performance.

Differently, this work presents original channel-reliability-aware regularization and optimizes it jointly with the filter, which further promotes the process efficiency and enhances the adaptiveness compared to the aforementioned methods. Thanks to this channel reliability regularization, channels with redundant information are suppressed, and those with discriminable features are emphasized automatically.

In summary, the proposed response-deviation- and channel-reliability-aware regularizations are used in conjunction with the filter regularization to promoting the tracking performance. On this basis, a tracker learning multi-regularized correlation filter (MRCF) is constructed. For instance, Fig. 1 compares MRCF and its baseline [5] on a representative challenging sequence with similar object around and UAV motion. An overall comparison of MRCF against the state-of-the-art (SOTA) tracking approaches is presented in Fig. 2. Moreover, as a fundamental subtask of many UAV applications, current popular vision-based UAV localization approaches generally focus on 2-D–3-D matches utilizing local features [28], [29]. This work provides an original solution for UAV self-localization based on visual tracking. The main contribution of this work is fourfold.

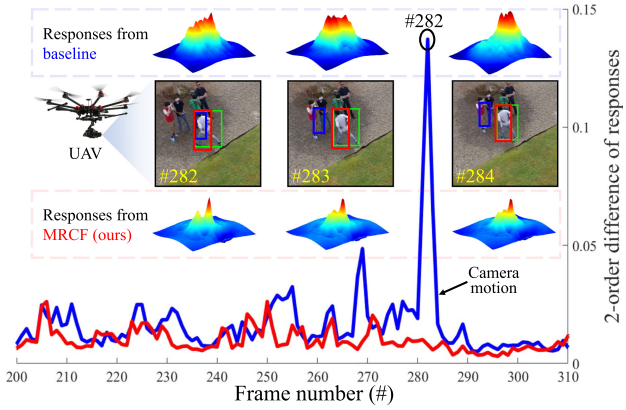

Fig. 1. Qualitative evaluation of the proposed MRCF (in *red*) and its baseline BACF [5] (in *blue*) on $person4\_1$ with similar object and camera motion. The *green* boxes denote ground truth. The second-order difference curves of the responses during frame #200 to #310 are drawn at the bottom, which can be regarded as the change rate of the responses. At the 282th frame, a sudden camera motion occurs. The responses in the BACF fluctuate violently and result in tracking failure. Constraining the second-order difference of responses, the MRCF smooths the change of responses and maintains reliable tracking.

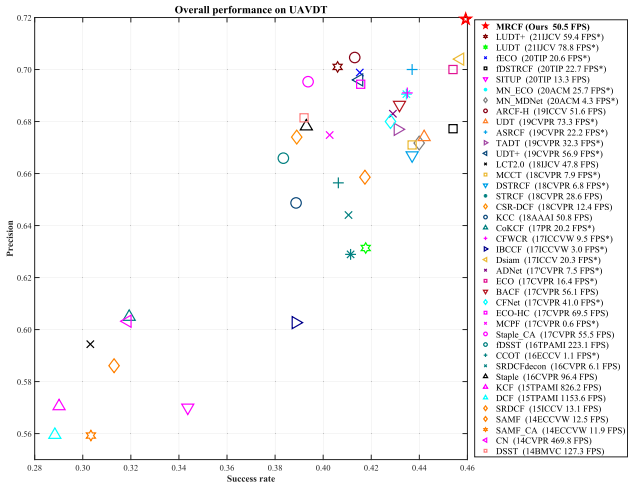

Fig. 2. Overall comparison against other 40 SOTA CPU- and GPU-based tracking approaches on the UAVDT dataset [30]. The legend presents the source of trackers and their tracking speed in FPS. Note that the superscript * denotes GPU speed. The result illustrates the outperformance of our MRCF in precision as well as success rate compared to other trackers while maintaining sufficient real-time tracking speed.

1) A new response-deviation-aware regularization is proposed. Through regularizing the second-order difference of the responses, the variations of responses are smoothed, and the perception of the tracker to object appearance variations is enhanced.
2) A novel channel-reliability-aware regularization is proposed to achieve automatic channel weight distribution. Introduced into the training phase, the weights are optimized jointly with filters in the training phase. As a result, the attention of the tracker is drawn to reliable channels.
3) Comprehensive experiments are conducted on large-scale benchmarks specific for UAV tracking, including UAV123@10fps [31], DTB70 [32], UAVDT [30],

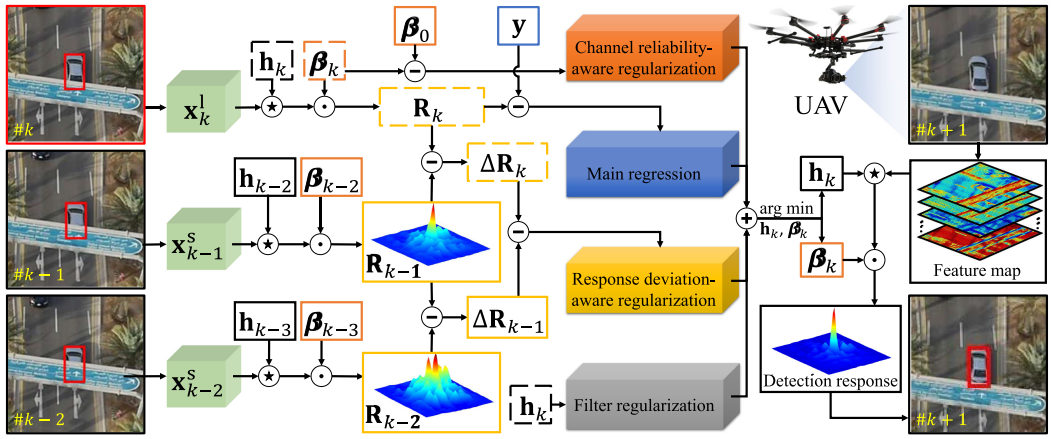

Fig. 3. Overall flowchart of the proposed MRCF. In the training stage of the kth frame, the 2-norm of difference between $\mathbf{y}$ and $\mathbf{R}_k$ is calculated as the main regression (the blue block). The filter regularization, as shown in the gray block, is composed of the 2-norm of the filter $\mathbf{h}_k$ in the current frame. The second-order difference of responses is introduced as the response-deviation-aware regularization, i.e., $\Delta\mathbf{R}_k - \Delta\mathbf{R}_{k-1}$ (the goldenrod block). The channel-reliability-aware regularization, as shown in the orange block, consists of the difference between the channel weight distributions in the current frame and that in the initialization, i.e., $\boldsymbol{\beta}_k - \boldsymbol{\beta}_0$. Once the $(k+1)$th frame arrives, a detection response map is obtained by the optimized $\mathbf{h}_k$ and $\boldsymbol{\beta}_k$, in conjunction with the feature map of the searching patch. The tracked object is then precisely localized according to the detection response. (Sequence courtesy of benchmark UAV123@10fps [31].)

VisDrone2018-test-dev [33], and VisDrone2020-SOT [34]. The results indicate the favorableness of the proposed method compared with SOTA trackers.

4) Based on the proposed tracking approach, an original UAV self-localization system toward air–ground collaboration is constructed; evaluations on practical scenarios validate its practicability and robustness.

II. Multi-Regularized Correlation Filter

In this section, the baseline [5] of the presented approach is previewed and followed by a detailed description of the presented tracker. The diagram in Fig. 3 sketches the workflow of the proposed framework.

A. Preview of the Background-Aware Correlation Filter (BACF)

The overall function of the BACF [5] can be formulated as

$$\mathcal{E}(\mathbf{h}_k) = \frac{1}{2}\sum_{d=1}^{D}\left\|\mathbf{y} - \mathbf{x}_k^d \star (\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d)\right\|_2^2 + \frac{\kappa}{2}\sum_{d=1}^{D}\left\|\mathbf{h}_k^d\right\|_2^2 \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^N$ is the ideal response, and $\mathbf{x}_k^d, \mathbf{h}_k^d \in \mathbb{R}^N$ are the dth channel of the vectorized feature and the filter in the kth frame, respectively. $\star$ denotes the spatial correlation operator and the superscript ^{d} indicates the dth channel among total D channels. $\mathbf{P}$ is a binary matrix proposed to alleviate the boundary effect. Moreover, κ is a preset coefficient. The second term is filter regularization, which dedicates to preventing overfitting.

In the detection process, a response map $\mathbf{R}_{k+1}$ is generated via calculating the spatial correlation between the searching patch $\mathbf{x}_{k+1}^{\mathrm{s}}$ cropped from the newly captured frame and the learned filter $\mathbf{h}_k$, i.e., $\mathbf{R}_{k+1} = \mathbf{x}_{k+1}^{\mathrm{s}} \star \mathbf{h}_k$. The location of the target is updated according to the position of the maximum in $\mathbf{R}_{k+1}$.

Although the BACF gives a favorable solution to the boundary effect, its training phase treats all channels equally, and the filter is learned merely with the information of the current frame, which limits the tracking performance from further promotion.

B. Response-Deviation-Aware Regularization

To incorporate the DCF framework with temporal cues properly, a response-deviation-aware regularization is presented. Constraining the second-order difference of responses, the regularization $\mathcal{E}_1$ is as follows:

$$\mathcal{E}_1 = \frac{\lambda}{2}\sum_{d=1}^{D}\left\|\Delta\mathbf{R}_k^d - \Delta\mathbf{R}_{k-1}^d\right\|_2^2 \quad (2)$$

where λ is a hyperparameter controlling the weight of the response-deviation-aware regularization. $\Delta\mathbf{R}_k^d \in \mathbb{R}^N$ obtained following (3) is the difference between the responses from the kth frame and the $(k-1)$th frame, i.e., $\mathbf{R}_k^d$, $\mathbf{R}_{k-1}^d \in \mathbb{R}^N$, respectively:

$$\Delta\mathbf{R}_k^d = \mathcal{S}(\mathbf{R}_k^d) - \mathcal{S}(\mathbf{R}_{k-1}^d). \quad (3)$$

Remark 1: Note that the peaks of $\mathbf{R}_k^d$ and $\mathbf{R}_{k-1}^d$ are shifted to the center by the shift operator $\mathcal{S}(\cdot)$. $\Delta\mathbf{R}_{k-1}^d$ is obtained in the same way. Fig. 1 presents a comparison between the second-order difference of responses before and after constraint.

C. Channel-Reliability-Aware Regularization

For the sake of highlighting reliable channels and suppressing misleading channels at the same time, this work proposes to learn a channel weight distribution $\boldsymbol{\beta}_k$ in the process of model training. Therefore, the channel-reliability-aware regularization $\mathcal{E}_2$ is constructed as

$$\mathcal{E}_2 = \frac{\gamma}{2}\left\|\boldsymbol{\beta}_k - \boldsymbol{\beta}_0\right\|_2^2 \quad (4)$$

where $\boldsymbol{\beta}_k = \mathrm{diag}(\beta_k^1, \beta_k^2, \ldots, \beta_k^D)$ is a diagonal matrix composed of all D channel weights. Moreover, γ is a preset constant and $\boldsymbol{\beta}_0$ is the initial weight distribution. The channel weights

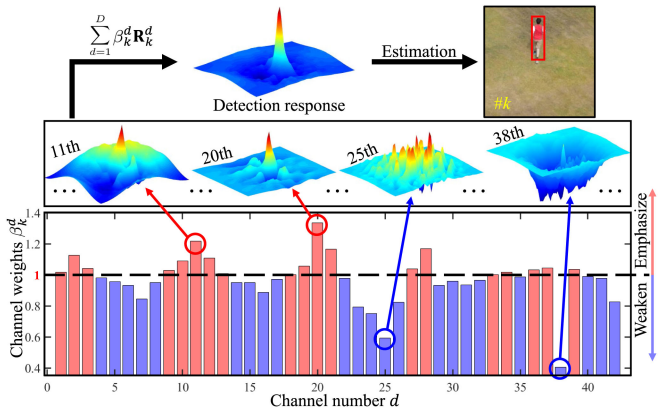

Fig. 4. Visualization of the optimized channel weights $\boldsymbol{\beta}_k$. Since the distribution is initialized to 1, $\beta_k^d = 1$ is set as a threshold to judge whether the channel is reliable. The larger β_k^d is, the closer the channel response to the ideal Gaussian label $\mathbf{y}$ and thus more beneficial to object localization. Therefore, channels with weights greater than 1 are emphasized and *vice versa*. (Sequence courtesy of benchmark UAV123@10fps [31].)

in the current frame are enforced to change smoothly and pay more attention to the robust channels at the same time.

Remark 2: In practice, the weight of each channel is initialized to 1, i.e., $\beta_0^d = 1$. Fig. 4 shows a visualization of the distribution for optimized channel weights on the kth frame.

D. Modeling and Optimization of the MRCF

Incorporating the above two proposed regularizations together with the filter regularization, the objective function of the MRCF is formulated as

$$\mathcal{E}(\mathbf{h_k}, \boldsymbol{\beta}_k) = \frac{1}{2}\sum_{d=1}^{D}\left\|\mathbf{y} - \beta_k^d \mathbf{x}_k^d \star (\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d)\right\|_2^2 + \frac{\kappa}{2}\sum_{d=1}^{D}\left\|\mathbf{h}_k^d\right\|_2^2 + \mathcal{E}_1 + \mathcal{E}_2. \tag{5}$$

By introducing an auxiliary variable $\mathbf{g}_k^d = \mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d$, (5) can be rewritten as

$$\mathcal{E}(\mathbf{h}_k, \mathbf{g}_k, \boldsymbol{\beta}_k) = \frac{1}{2}\sum_{d=1}^{D}\left(\left\|\mathbf{y} - \beta_k^d \mathbf{x}_k^d \star \mathbf{g}_k^d\right\|_2^2 + \kappa\left\|\mathbf{h}_k^d\right\|_2^2 + \gamma\left\|\beta_k^d - \beta_0^d\right\|_2^2 + \lambda\left\|\Delta\mathbf{R}_k^d - \Delta\mathbf{R}_{k-1}^d\right\|_2^2\right). \tag{6}$$

Since the dth channel response of the kth frame can be formulated as $\mathbf{R}_k^d = \beta_k^d \mathbf{x}_k^d \star \mathbf{g}_k^d$, (6) can be transformed into the frequency domain via Parseval's theorem as

$$\mathcal{E}(\mathbf{h}_k, \hat{\mathbf{g}}_k, \boldsymbol{\beta}_k) = \frac{1}{2}\sum_{d=1}^{D}\left(\frac{1}{N}\left\|\hat{\mathbf{y}} - \beta_k^d \hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d\right\|_2^2 + \kappa\left\|\mathbf{h}_k^d\right\|_2^2 + \gamma\left\|\beta_k^d - \beta_0^d\right\|_2^2 + \frac{\lambda}{N}\left\|\beta_k^d \hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d - 2\widehat{\mathcal{S}(\mathbf{R}_{k-1}^d)} + \widehat{\mathcal{S}(\mathbf{R}_{k-2}^d)}\right\|_2^2\right)$$

$$\text{s.t.,} \quad \mathbf{g}_k^d = \sqrt{N}\mathbf{F}\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d \tag{7}$$

where $\mathbf{F} \in \mathbb{C}^{N\times N}$ is the discrete Fourier transform (DFT) matrix and the superscript $\hat{(\cdot)}$ stands for the DFT operator, e.g., $\hat{\delta} = \sqrt{N}\mathbf{F}\delta$.

Introducing $\zeta \in \mathbb{R}^{N\times D}$ (defined as $\zeta = [\zeta^1, \zeta^2, \ldots, \zeta^D]$) as the Lagrange multiplier and μ as the penalty factor, (7) can be reformulated as

$$\mathcal{L}(\mathbf{h}_k, \hat{\mathbf{g}}_k, \boldsymbol{\beta}_k, \hat{\boldsymbol{\zeta}}) = \mathcal{E}(\mathbf{h}_k, \hat{\mathbf{g}}_k, \boldsymbol{\beta}_k) + \frac{\mu}{2}\sum_{d=1}^{D}\left\|\hat{\mathbf{g}}_k^d - \sqrt{N}\mathbf{F}\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d + \frac{\hat{\zeta}_k^d}{\mu}\right\|_2^2. \tag{8}$$

To boost the process efficiency, the ADMM technique [35] is applied in this work to optimize (8) by solving the following subproblems iteratively.

1) Subproblem $\mathbf{h}_k^*$: If $\hat{\mathbf{g}}$, $\boldsymbol{\beta}$, and $\hat{\boldsymbol{\zeta}}$ are fixed. $\mathbf{h}^*$ can be easily obtained as

$$\begin{aligned}\mathbf{h}_k^{d*(i+1)} &= \arg\min_{\mathbf{h}_k}\left\{\frac{1}{2}\sum_{d=1}^{D}\left(\kappa\left\|\mathbf{h}_k^{d(i)}\right\|_2^2 + \mu\left\|\hat{\mathbf{g}}_k^{d(i)} - \sqrt{N}\mathbf{F}\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^{d(i)} + \frac{\hat{\zeta}_k^{d(i)}}{\mu}\right\|_2^2\right)\right\} \\ &= \frac{N(\mu\mathbf{g}_k^d + \zeta_k^d)}{\kappa + \mu N}\end{aligned} \tag{9}$$

where the superscripts $^{(i)}$ and * indicate the ith iteration and the conjugate operation, respectively. For brevity, $^{(i)}$ is omitted from the final expression of (9). $\mathbf{g}_k^d = \frac{1}{\sqrt{N}}\mathbf{P}\mathbf{F}^{\mathsf{T}}\hat{\mathbf{g}}_k^d$ and $\zeta_k^d = \frac{1}{\sqrt{N}}\mathbf{P}\mathbf{F}^{\mathsf{T}}\hat{\zeta}_k^d$ are calculated, respectively, via the inverse discrete Fourier transform (IDFT) operation.

2) Subproblem $\mathbf{g}_k^*$: Given $\mathbf{h}$, $\boldsymbol{\beta}$, and $\hat{\boldsymbol{\zeta}}$, $\hat{\mathbf{g}}^*$ can be acquired by solving

$$\hat{\mathbf{g}}_k^{*(i+1)} = \arg\min_{\hat{\mathbf{g}}_k}\left\{\frac{1}{2}\sum_{d=1}^{D}\left(\frac{1}{N}\left\|\hat{\mathbf{y}} - \beta_k^d\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d\right\|_2^2 + \frac{\lambda}{N}\left\|\beta_k^d\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d - 2\widehat{\mathcal{S}(\mathbf{R}_{k-1}^d)} + \widehat{\mathcal{S}(\mathbf{R}_{k-2}^d)}\right\|_2^2 + \mu\left\|\hat{\mathbf{g}}_k^d - \sqrt{N}\mathbf{F}\mathbf{P}^{\mathsf{T}}\mathbf{h}_k^d + \frac{\hat{\zeta}_k^d}{\mu}\right\|_2^2\right)\right\}. \tag{10}$$

Since $\hat{\mathbf{y}}$ is dependent on $\hat{\mathbf{x}}(n) = [\hat{\mathbf{x}}_k^1(n), \hat{\mathbf{x}}_k^2(n), \ldots, \hat{\mathbf{x}}_k^D(n)]^{\mathsf{T}}$ and $\hat{\mathbf{g}}(n) = [\hat{\mathbf{g}}_k^1(n), \hat{\mathbf{g}}_k^2(n), \ldots, \hat{\mathbf{g}}_k^D(n)]^{\mathsf{T}}$, (10) can be transferred to N independent subexpressions

$$\hat{\mathbf{g}}_k^{*(i+1)}(n) = \arg\min_{\hat{\mathbf{g}}_k(n)}\left\{\frac{1}{2}\left(\frac{1}{N}\left\|\hat{\mathbf{y}}(n) - \hat{\mathbf{x}}_k^{\mathsf{T}}(n)\boldsymbol{\beta_k}\hat{\mathbf{g}}_k(n)\right\|_2^2 + \frac{\lambda}{N}\left\|\hat{\mathbf{x}}_k^{\mathsf{T}}(n)\boldsymbol{\beta_k}\hat{\mathbf{g}}_k(n) - 2\widehat{\mathcal{S}(\mathbf{R}_{k-1}^d)}(n) + \widehat{\mathcal{S}(\mathbf{R}_{k-2}^d)}(n)\right\|_2^2 + \mu\left\|\hat{\mathbf{g}}_k(n) - \sqrt{N}\mathbf{F}\mathbf{P}^{\mathsf{T}}\mathbf{h}_k(n) + \frac{\hat{\zeta}_k(n)}{\mu}\right\|_2^2\right)\right\}. \tag{11}$$

Deriving $\hat{\mathbf{g}}_k(n)$, (11) is reformulated as

$$\begin{aligned}\hat{\mathbf{g}}_k{}^{*(i+1)}(n) &= b\left(\boldsymbol{\beta_k}^{\mathsf{T}}\hat{\mathbf{x}}_k(n)\hat{\mathbf{x}}_k^{\mathsf{T}}(n)\boldsymbol{\beta_k} + \mu bN\mathbf{I}_D\right)^{-1} \\ &\times \left(\boldsymbol{\beta_k}^{\mathsf{T}}\hat{\mathbf{x}}_k(n)\hat{\mathbf{y}}(n) + 2\lambda\boldsymbol{\beta_k}^{\mathsf{T}}\hat{\mathbf{x}}_k(n)\widehat{\mathcal{S}(\mathbf{R}_{k-1})}(n)\right. \\ &\left.-\lambda\boldsymbol{\beta_k}^{\mathsf{T}}\hat{\mathbf{x}}_k(n)\widehat{\mathcal{S}(\mathbf{R}_{k-2})}(n) - N\hat{\boldsymbol{\zeta}}(n) + \mu N\hat{\mathbf{h}}_k(n)\right) \quad (12)\end{aligned}$$

where $b = 1/(1+\lambda)$. Since the matrix $\boldsymbol{\beta}_k$ is symmetric, $\boldsymbol{\beta}_k^{\mathsf{T}} = \boldsymbol{\beta}_k$.

Remark 3: The matrix inversion operation is still involved in (12), which is not computation friendly. Therefore, the Sherman–Morrison formula, i.e., $(\mathbf{A}+\mathbf{uv}^{\mathsf{T}})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{u}(\mathbf{I}_D + \mathbf{v}^{\mathsf{T}}\mathbf{A}^{-1}\mathbf{u})^{-1}\mathbf{v}^{\mathsf{T}}\mathbf{A}^{-1}$, is applied in this work. In this case, $\mathbf{A} = \mu bN\mathbf{I}_D$ and $\mathbf{u} = \mathbf{v} = \boldsymbol{\beta}_k^{\mathsf{T}}\hat{\mathbf{x}}_k(n)$. Consequently, (12) is simplified as

$$\begin{aligned}\hat{\mathbf{g}}_k^{*(i+1)}(n) &= \frac{1}{\mu N}\left(\mathbf{I}_D - \frac{\boldsymbol{\beta_k}\hat{\mathbf{x}}_k(n)\hat{\mathbf{x}}_k^{\mathsf{T}}(n)\boldsymbol{\beta_k}}{\mu bN + \hat{\mathbf{x}}_k^{\mathsf{T}}(n)\boldsymbol{\beta_k}\boldsymbol{\beta_k}\hat{\mathbf{x}}_k(n)}\right) \\ &\times \left(\boldsymbol{\beta_k}\hat{\mathbf{x}}_k(n)\hat{\mathbf{y}}_k(n) + 2\lambda\boldsymbol{\beta_k}\hat{\mathbf{x}}_k(n)\widehat{\mathcal{S}(\mathbf{R}_{k-1})}(n)\right. \\ &\left.-\lambda\boldsymbol{\beta_k}\hat{\mathbf{x}}_k(n)\widehat{\mathcal{S}(\mathbf{R}_{k-2})}(n) - N\hat{\boldsymbol{\zeta}}(n) + \mu N\hat{\mathbf{h}}_k(n)\right). \quad (13)\end{aligned}$$

3) Subproblem $\boldsymbol{\beta}$: Assuming that $\mathbf{h}$, $\hat{\mathbf{g}}$, and $\hat{\boldsymbol{\zeta}}$ are fixed, $\boldsymbol{\beta}$ can be obtained easily by taking the partial derivatives of $\boldsymbol{\beta}$ to zero

$$\begin{aligned}\beta_k^{d(i+1)} &= \arg\min_{\beta_k^d}\left\{\frac{1}{2N}\sum_{d=1}^{D}\left\|\hat{\mathbf{y}} - \beta_k^d\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d\right\|_2^2\right. \\ &+ \frac{\lambda}{2N}\sum_{d=1}^{D}\left\|\beta_k^d\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d - 2\widehat{\mathcal{S}(\mathbf{R}_{k-1}^d)} + \widehat{\mathcal{S}(\mathbf{R}_{k-2}^d)}\right\|_2^2 \\ &\left.+ \frac{\gamma}{2}\sum_{d=1}^{D}\left\|\beta_k^d - \beta_0^d\right\|_2^2\right\} \\ &= \frac{(\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d)^{\mathsf{T}}(\hat{\mathbf{y}} + 2\lambda\widehat{\mathcal{S}(\mathbf{R}_{k-1}^d)} - \lambda\widehat{\mathcal{S}(\mathbf{R}_{k-2}^d)}) + \gamma N\beta_0^d}{(1+\lambda)(\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d)^{\mathsf{T}}(\hat{\mathbf{x}}_k^d \odot \hat{\mathbf{g}}_k^d) + \gamma N.}\end{aligned} \quad (14)$$

4) Lagrangian Multiplier Update: According to [35], in each iteration, the Lagrangian multiplier $\hat{\boldsymbol{\zeta}}_k$ and the penalty factor μ are updated as

$$\hat{\boldsymbol{\zeta}}_k^{(i+1)} = \hat{\boldsymbol{\zeta}}_k^{(i)} + \mu\left(\hat{\mathbf{g}}_k^{*(i+1)} - \hat{\mathbf{h}}_k^{*(i+1)}\right) \quad (15)$$

$$\mu^{(i+1)} = \min(\mu_{\max}, \eth\mu^{(i)}) \quad (16)$$

where $\eth$ is the step length.

E. Model Update

In the MRCF, a linear interpolation method is applied to update the object appearance model $\hat{\mathbf{x}}_k^{\text{model}}$ as

$$\hat{\mathbf{x}}_k^{\text{model}} = (1-\eta)\hat{\mathbf{x}}_{k-1}^{\text{model}} + \eta\hat{\mathbf{x}}_k^{\text{l}} \quad (17)$$

where η stands for the update rate and $\hat{\mathbf{x}}_k^{\text{l}}$ denotes the features extracted from the learning patch in the Fourier domain. To this end, we use $\hat{\mathbf{x}}_k^{\text{model}}$ rather than $\hat{\mathbf{x}}_k$ in solutions to $\mathbf{g}_k^*$ and $\boldsymbol{\beta}$.

F. Tracking Framework

As each newly captured frame arrives, the searching patch centered on the position of the object in the previous frame will be cropped first. Different from our baseline and other standard DCFs, the response from each channel is element-wise multiplied by the optimized weight β_k^d. Therefore, the detection response $\mathbf{R}_{k+1}$ in the MRCF is formulated as

$$\mathbf{R}_{k+1} = \mathcal{F}^{-1}\left(\sum_{d=1}^{D}\beta_k^d \odot \hat{\mathbf{x}}_{k+1}^{d,\text{s}} \odot \hat{\mathbf{g}}_k^d\right) \quad (18)$$

where $\mathcal{F}^{-1}$ denotes the IDFT operator.

Remark 4: The complex spatial correlation operation is transformed into the frequency domain to boost computational efficiency.

Afterward, the tracker will estimate the position of the object as the location of the maximum in the response. The update of the object's size is following [36]. Centered on the updated location, a training patch is extracted from the image, and the appearance model is updated by (17). In the end, the correlation filter is trained following Section II-D.

G. UAV Self-Localization System

The framework of the localization system is based on the open-source software in [37], which relies on a camera equipped with an infrared-pass filter to recognize infrared LEDs. In contrast, given the prior positions of four markers, the MRCF is applied to simultaneously track them in images captured by a common camera in our case. Then, the tracking results in the 2-D images are transferred to 3-D positions in the world coordinate. The coarse pose of the UAV is estimated by a perspective-n-point algorithm. Ultimately, the optimized location of the UAV is obtained after iteratively refining the reprojection error.

III. Experiment

In this section, substantial experiments are conducted among the proposed MRCF and other SOTA methods on five widely applied UAV object tracking benchmarks, i.e., UAV123@10fps [31], DTB70 [32], UAVDT [30], VisDrone2018-test-dev [33], and VisDrone2020-SOT (including the training, validation, and test-dev subsets) [34], with totally 410 sequences and over 223K frames. Afterward, some onboard tracking tests are performed for applicability evaluation. Moreover, a UAV localization system based on our tracking approach is designed; evaluations on practical captured sequences are also presented.

A. Implementation Details

This work uses MATLAB R2019a on a PC with an i7-8700K CPU and an Nvidia GeForce RTX 2080 GPU to conduct the experiments. As for the representation of objects, the MRCF uses a combination of CN [20], HOG [19], and gray-scale features. The number of ADMM iterations is set to be 3, and other parameters are set to $\mu = 1$, $\mu_{\max} = 100$, $\eth = 10$, and $\beta_0^d = 1$. The learning rate η of the appearance model is set to 0.0199. The regularization coefficients κ, λ, and γ are set as 0.01, 0.004, and 10, respectively.

Remark 5: All parameters are kept fixed in the whole experiment. The tracking code and some tracking videos are presented at[1,2], respectively. Since MATLAB and ROS[3] can collaborate efficiently, in typical real-world UAV tracking-related applications, the tracking results of the MRCF can be further processed and transferred to the executive mechanism through ROS for UAV control.

B. Evaluation of UAV Tracking Performance

1) Metrics: The experiments follow one-pass evaluation as in [31], which involves two metrics, i.e., precision and success rate. The precision is characterized by the center location error (CLE) of the estimated rectangle box and the ground truth. The percentage of the frames whose CLE is below the given threshold is presented as the precision plot (PP), in which the precision at 20 pixels is used to rank trackers. Besides, the success rate is measured as the intersection over union (IoU) between ideal boxes and predicted ones. The success plot (SP) shows the proportion of the frames, whose IoU is greater than a preset maximum threshold. In practice, the area under the curve (AUC) on the SP is used to rank the success rate of trackers.

2) Comparison With CPU-Based Trackers: Nineteen SOTA CPU-based trackers, i.e., DCF [16], KCF [16], SRDCF [38], DSST [36], CN [20], LCT2.0 [39], SAMF [40], SRDCFdecon [41], Staple [42], BACF [5], fDSST [43], CSR-DCF [23], ECO-HC [44], Staple_CA [45], SAMF_CA [45], STRCF [21], ARCF-H [7], KCC [46], and SITUP [47], are used for evaluation.

Remark 6: All involved trackers in this work are evaluated on the same computer using the officially released source code from their authors without any tuning.

Overall performance comparison: Fig. 5 exhibits that the MRCF achieves competitive performance on all five benchmarks. Specifically, on UAV123@10fps, the MRCF achieves the best precision (0.666) and also leads in the AUC (0.485), exceeding the second-best precision (0.643) and the AUC (0.468) by 3.6%, respectively. On the DTB70 benchmark, the MRCF also ranks first in precision (0.666) as well as AUC (0.466). On the UAVDT benchmark, the MRCF gains the highest precision (0.719) and AUC (0.459). On VisDrone2018-test-dev, the MRCF consistently ranks first in terms of both the precision (0.812) and the success rate (0.600). On the VisDrone2020-SOT benchmark, the precision of the MRCF (0.774) is slightly lower than that of ECO-HC (0.778), while the MRCF realizes the best success rate (0.569). In addition to the competitive tracking performance, the MRCF maintains a real-time speed of 41.6 FPS on average. These cheerful results demonstrate that the proposed approach is adequate for real-time UAV object tracking. Some qualitative analyses among the top six trackers in evaluation are exhibited in Fig. 6.

Long-term tracking (LTT) evaluation: As one of the most common scenes in practical UAV tracking, LTT contains numerous challenges such as large occlusion (LO), out of view, and fast motion; robust performance in LTT scenarios illustrates the practicability of trackers. To provide a comprehensive evaluation, the UAV20L [31] benchmark, in conjunction with sequences in the aforementioned five benchmarks, whose lengths longer than 1700 frames, are merged as an LTT benchmark, dubbed UAV-LTT. The UAV-LTT benchmark consists of 40 sequences with an average length of 2572 frames, a maximum length of 5527 frames, and a minimum length of 1717 frames. Evaluation results on UAV-LTT are presented in Fig. 5(f). The MRCF ranks first in terms of both the precision (0.664) and the AUC (0.452), demonstrating its satisfying LTT performance. This favorable results in LTT can be attributed to the response regularizer and the channel regularizer. The aberrance in the LTT process is effectively suppressed by the response-deviation-aware regularization, while the channel-reliability-aware regularization dedicates to adaptively emphasize reliable channels.

TABLE I
ATTRIBUTE-ORIENTED EVALUATION ON UAVDT

Trackers	BC	CM	IV	LO	LTT	OB	OM	SV	SO
DCF	0.236	0.261	0.308	0.232	0.289	0.293	0.243	0.249	0.252
DSST	0.333	0.360	0.412	0.329	0.531	0.397	0.332	0.354	0.383
KCF	0.235	0.267	0.312	0.229	0.312	0.298	0.244	0.255	0.251
CN	0.269	0.288	0.349	0.255	0.333	0.319	0.267	0.265	0.312
LCT2.0	0.246	0.285	0.329	0.233	0.353	0.314	0.257	0.270	0.270
SAMF	0.269	0.283	0.315	0.258	0.342	0.300	0.257	0.265	0.290
SRDCF	0.351	**0.383**	**0.442**	0.328	0.531	0.406	0.362	0.397	0.412
SRDCFdecon	0.339	0.374	0.430	0.321	0.515	0.397	0.351	0.389	0.411
Staple	0.323	0.351	0.428	0.324	0.541	0.405	0.346	0.363	0.382
fDSST	0.315	0.363	0.395	0.332	0.518	0.369	0.312	0.337	0.380
SAMF_CA	0.269	0.275	0.302	0.258	0.393	0.271	0.229	0.254	0.296
Staple_CA	0.326	0.349	0.429	0.325	0.539	0.405	0.344	0.366	0.379
KCC	0.332	0.351	0.431	0.304	0.480	0.398	0.322	0.341	0.390
CSR-DCF	0.345	0.347	0.398	**0.352**	0.487	0.373	0.342	0.366	0.355
MCCT-H	0.343	0.367	0.415	0.348	0.565	0.390	0.343	0.384	0.389
STRCF	0.340	0.365	0.421	0.319	0.525	**0.407**	0.341	0.388	**0.421**
ARCF-H	0.352	0.372	0.416	0.339	0.569	0.398	0.347	0.386	0.402
SITUP	0.277	0.313	0.377	0.257	0.487	0.355	0.273	0.313	0.317
ECO-HC	**0.375**	0.381	0.435	**0.363**	**0.603**	0.391	**0.367**	**0.400**	0.372
BACF	**0.364**	**0.384**	**0.461**	0.335	**0.582**	**0.444**	**0.371**	**0.405**	**0.425**
MRCF (ours)	**0.399**	**0.435**	**0.463**	**0.388**	**0.606**	**0.446**	**0.400**	**0.438**	**0.469**
Δ (%)	6.57%	13.13%	0.45%	6.74%	0.56%	0.47%	7.70%	8.13%	10.30%

The top three trackers in terms of each attribute are highlighted with *red*, *green*, and *blue* colors. Δ represents the percentages of the MRCF exceeding the second-best performances.

Attribute-oriented evaluation: To analyze the performance of trackers under different visual ambiguity in detail, UAVDT annotates all sequences with nine attributes, i.e., LO, object blur (OB), background clutter (BC), illumination variations (IV), SV, camera motion (CM), small object (SO), LTT, and object motion (OM). In this subsection, an analysis of these attributes is conducted. As presented in Table I, the MRCF gains the best AUC on all attributes. In particular, the MRCF shows superior performance on UAV-tracking-specific attributes, exceeding the second-best tracker by 13.13%, 10.30%, 8.13%, and 7.70%, on CM, SO, SV, and OM, respectively. This comprehensive performance strongly verifies the versatility of our approach.

Remark 7: On challenging attributes that generally lead to sudden variations of object appearance, e.g., BC, CM, LO, and OM, the MRCF yields a satisfying performance and surpasses the second-best result with a large margin, which attributes to that the response-deviation-aware regularization effectively smooths the variation of responses and raises the adaptiveness of the tracker in sudden appearance variation conditions.

[1][Online]. Available: https://github.com/vision4robotics/MRCF-Tracker
[2][Online]. Available: https://youtu.be/XzkreAPynE4
[3][Online]. Available: https://www.ros.org/

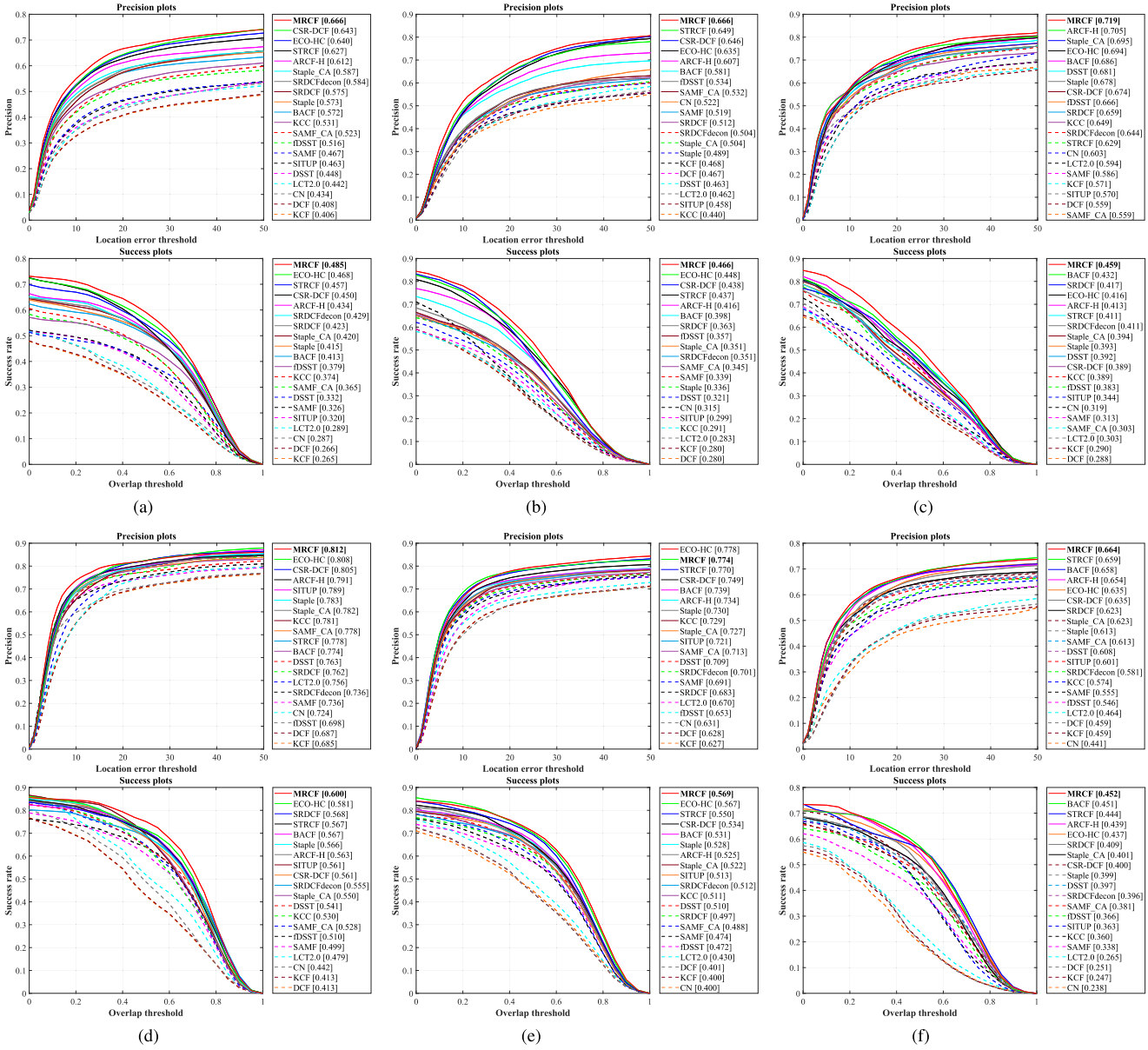

Fig. 5. PPs and SPs of the MRCF and other compared CPU-based trackers on (a) UAV123@10fps, (b) DTB70, (c) UAVDT, (d) VisDrone2018-test-dev, (e) VisDrone2020-SOT, and (f) UAV-LTT benchmarks. The proposed MRCF tracker exhibits competitive performance among all SOTA approaches.

3) Comparison With GPU-Based Trackers

This work further compares the MRCF with other 21 SOTA GPU-based trackers, including CFWCR [48], CCOT [49], ADNet [50], CoKCF [51], ECO [44], IBCCF [52], MCPF [53], MCCT [54], DSTRCF [21], ASRCF [22], TADT [55], UDT [56], UDT+ [56], CFNet [57], DSiam [58], LUDT [59], LUDT+[59], MN_ECO [60], MN_MDNet [60], fECO [61], and fDSTRCF [61], on the UAVDT benchmark to present a more comprehensive evaluation. The results are displayed in Table II. The GPU-based trackers generally utilize deep features for appearance representation and rely on high-end GPUs to cover the high computation load. With only handcrafted features, the MRCF tracker is not inferior to SOTA GPU-based trackers. The results show that the MRCF achieves the best precision (0.719) and success rate (0.459). These competitive performances own to our proposed two schemes, which enable the tracker to

TABLE II
PERFORMANCE COMPARISON WITH GPU-BASED TRACKERS ON UAVDT

Tracker	Venue	Prec.	Succ.	FPS	Tracker	Venue	Prec.	Succ.	FPS
MRCF	**Ours**	**0.719**	**0.459**	50.5	DSTRCF	18'CVPR	0.667	0.437	6.8*
LUDT+	21'IJCV	**0.701**	0.406	**59.4***	MCCT	18'CVPR	0.671	0.437	7.9*
LUDT	21'IJCV	0.631	0.418	**78.8***	IBCCF	17'ICCVW	0.603	0.389	3.0*
fECO	20'TIP	0.699	0.415	20.6*	CoKCF	17'PR	0.605	0.319	20.2*
fDSTRCF	20'TIP	0.677	**0.454**	22.7*	ECO	17'CVPR	0.700	0.454	16.4*
MN_ECO	20'ACM	0.691	0.435	25.7*	CFWCR	17'ICCVW	0.691	0.435	9.5*
MN_MDNet	20'ACM	0.672	0.440	4.3*	MCPF	17'CVPR	0.675	0.403	0.6*
UDT+	19'CVPR	0.696	0.415	56.9*	CFNet	17'CVPR	0.680	0.428	41.0*
UDT	19'CVPR	0.674	0.442	**73.3***	Dsiam	17'ICCV	**0.704**	**0.457**	20.3*
ASRCF	19'CVPR	0.700	0.437	22.2*	ADNet	17'CVPR	0.683	0.429	7.5*
TADT	19'CVPR	0.677	0.431	32.3*	CCOT	16'ECCV	0.656	0.406	1.1*

The superscript * means GPU speed.

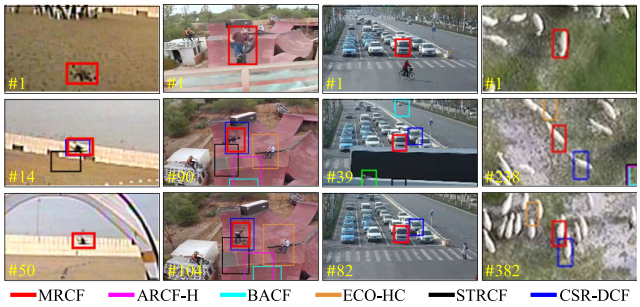

Fig. 6. Qualitative analyses among the top six trackers. From left to right, the sequences are $uav3$ from UAV123@10fps, $BMX3$ from DTB70, $S0601$ from UAVDT, and $uav0000093\_01817\_s$ from VisDrone2018-test-dev.

TABLE III
ABLATION STUDY OF THE PROPOSED MRCF TRACKER

Settings	BACF	BACF + RD	BACF + CR	**MRCF**
Avg. prec.	0.653	0.704	0.708	**0.716**
Avg. AUC	0.452	0.494	0.492	**0.503**

perceive the appearance changes of the object and focus on reliable channels adaptively.

Remark 8: The speed of these GPU-based trackers is achieved on a high-end GPU. While using only a single CPU, the proposed MRCF tracker still runs at a speed of over 50 FPS with satisfying tracking performance on UAVDT. This GPU independence of the MRCF can greatly reduce the power consumption of the tracker and extend the operation endurance of UAVs. For these reasons, we argue that the presented approach is sufficient for UAV object tracking applications.

4) Ablation Study

For effectiveness verification of the proposed schemes, the performance comparison of the MRCF with various modules enabled is conducted on four benchmarks, Table III shows the average precision and AUC. With the activation of the **r**esponse-**d**eviation-aware (RD) and the **c**hannel-**r**eliability-aware (CR) regularizations in the BACF, the performance of the tracker improves smoothly. Incorporating both RD and CR, the precision and the AUC of the MRCF tracker are further improved to 0.716 and 0.503, exceeding the baseline by 9.6% and 11.3%. It can be clearly seen that the two modules proposed in this article remarkably boost the tracking performance.

5) Onboard Tests

In apart from favorable tracking performance, the onboard applicability is also a crucial metric to evaluate a UAV tracking approach. Therefore, the MRCF is further implemented on a UAV platform with typical onboard PC, i.e., Intel NUC8i7HVK with a single Intel Core i7-8809G CPU and a 32-GB RAM. The distance between the target and the UAV is about 50 m in our tests. Tracking performance of three tests is presented in Fig. 7, the tracking speed on specific tasks is reported in the legends. The MRCF realizes real-time onboard processing speed

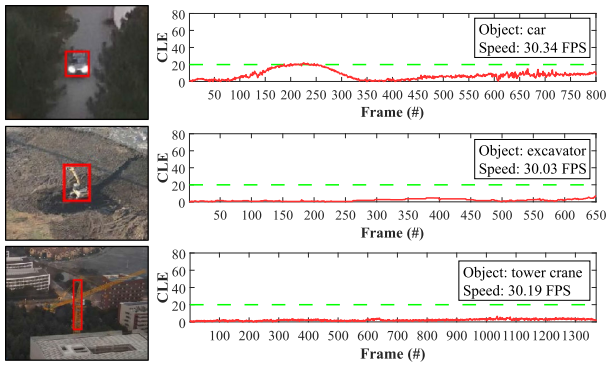

Fig. 7. Onboard tracking performance of three tests. The tracked object is marked with red boxes.

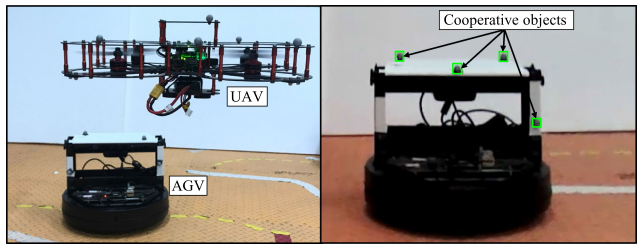

Fig. 8. Experimental setup (left) and image from UAV perspective (right). The cooperative objects, i.e., the markers are tracked by the MRCF simultaneously. The tracking results in images are then transferred to the world coordinate for UAV self-localization.

while maintaining precise tracking, verifying its practicability on UAV-tracking-related applications.

C. PRACTICAL TESTS OF THE UAV SELF-LOCALIZATION SYSTEM

To testify the practicability of the proposed UAV self-localization system, four tests of UAV-automated guided vehicle (AGV) collaboration covering 2915 frames in total captured in a laboratory environment are conducted. The images are captured by an Intel RealSence (R200) camera with a resolution of 1280 × 720 in 10 FPS. The experimental setup is displayed in Fig. 8. A motion capture system developed by Quanser, namely, Autonomous Vehicles Research Studio,[4] is adopted to capture the ground-truth localization of the UAV. The location of the UAV recorded by the motion capture system is used as the ground-truth. The trajectory estimated by our MRCF-based localization system and its error is reported in Fig. 9. It can be seen that the error is within an acceptable range. One conclusion can be drawn that our visual tracking-based UAV localization system can achieve favorable performance.

Remark 9: The proposed localization system can work in various scenes since the target of our tracking approach can be arbitrary objects.

[4][Online]. Available: https://www.quanser.com/products/autonomous-vehicles-research-studio/

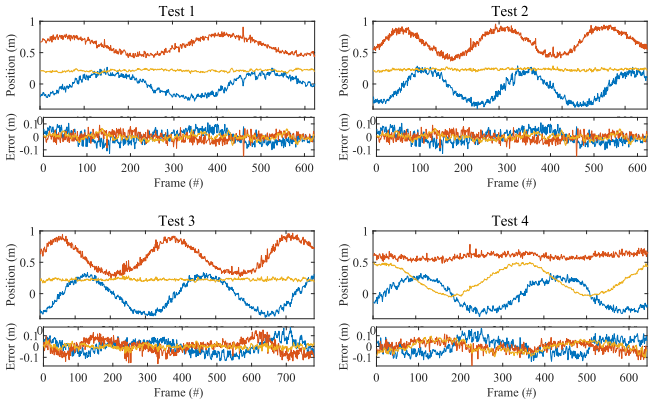

Fig. 9. Trajectory estimation and error of our UAV localization system on four tests. **Red**, **blue**, and **orange** curves denote x, y, and z positions of the UAV, respectively.

IV. Conclusion

This article proposed a multi-regularized correlation filter, i.e., MRCF tracker. It can smooth the deviation of responses adaptively and optimize channel weight distributions jointly. Comprehensive experiments on multiple UAV tracking benchmarks show that the MRCF performs prominently against other SOTA trackers, meeting the real-time requirement of UAV applications and, thus, can be well applied to the UAV object tracking tasks. Moreover, an original solution to UAV self-localization is provided based on our MRCF. Evaluations verify its solid performance and feasibility. We strongly believe that this work can significantly contribute to the communities of both UAV tracking and self-localization. ■

References

[1] F. Tu, S. S. Ge, Y. Tang, and C. C. Hang, "Robust visual tracking via collaborative motion and appearance model," *IEEE Trans. Ind. Electron.*, vol. 13, no. 5, pp. 2251–2259, Oct. 2017.

[2] Q. Wu, H. Wang, Y. Liu, L. Zhang, and X. Gao, "SAT: Single-shot adversarial tracker," *IEEE Trans. Ind. Electron.*, vol. 67, no. 11, pp. 9882–9892, Nov. 2020.

[3] M. Zhang, X. Liu, D. Xu, Z. Cao, and J. Yu, "Vision-based target-following guider for mobile robot," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9360–9371, Dec. 2019.

[4] M. Guan, C. Wen, M. Shan, C. Ng, and Y. Zou, "Real-time event-triggered object tracking in the presence of model drift and occlusion," *IEEE Trans. Ind. Electron.*, vol. 66, no. 3, pp. 2054–2065, Mar. 2019.

[5] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1135–1143. [Online]. Available: https://ieeexplore.ieee.org/document/8237391

[6] X. Cheng, Y. Zhang, L. Zhou, and Y. Zheng, "Visual tracking via auto-encoder pair correlation filter," *IEEE Trans. Ind. Electron.*, vol. 67, no. 4, pp. 3288–3297, Apr. 2020.

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

[8] T. Yang, P. Xu, R. Hu, H. Chai, and A. B. Chan, "ROAM: Recurrently optimizing tracking model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6718–6727. [Online]. Available: https://ieeexplore.ieee.org/document/9156508

[9] P. Voigtlaender, J. Luiten, P. H. S. Torr, and B. Leibe, "Siam R-CNN: Visual tracking by re-detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6578–6588. [Online]. Available: https://ieeexplore.ieee.org/document/9156711

[10] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4660–4669. [Online]. Available: https://ieeexplore.ieee.org/document/8953466

[11] 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.*, to be published, doi: 10.1109/TGRS.2020.3030265.

[12] Y. Wang, L. Ding, and R. Laganiere, "Real-time UAV tracking based on PSR stability," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019, pp. 144–152.

[13] B. Li, C. Fu, F. Ding, J. Ye, and F. Lin, "ADTrack: Target-aware dual filter learning for real-time anti-dark UAV tracking," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 1–8.

[14] 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.*, 2021, pp. 1–8.

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

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

[17] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3074–3082.

[18] C. Fu, Z. Huang, Y. Li, R. Duan, and P. Lu, "Boundary effect-aware visual tracking for UAV with online enhanced background learning and multi-frame consensus verification," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4415–4422.

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

[20] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1090–1097.

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

[22] 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.*, 2019, pp. 4670–4679. [Online]. Available: https://ieeexplore.ieee.org/document/8953651

[23] A. Lukežič, T. Vojír, L. Čehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6309–6318. [Online]. Available: https://ieeexplore.ieee.org/document/8099998

[24] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Joint group feature selection and discriminative filter learning for robust visual object tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7950–7960. [Online]. Available: https://ieeexplore.ieee.org/document/9010061

[25] X. Lu, C. Ma, B. Ni, and X. Yang, "Adaptive region proposal with channel regularization for robust object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 4, pp. 1268–1282, Apr. 2021.

[26] Q. Wang, Z. Teng, J. Xing, J. Gao, W. Hu, and S. Maybank, "Learning attentions: Residual attentional siamese network for high performance online visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4854–4863.

[27] D. Li, G. Wen, Y. Kuai, and F. Porikli, "End-to-end feature integration for correlation filter tracking with channel attention," *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1815–1819, Dec. 2018.

[28] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient & effective prioritized matching for large-scale image-based localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1744–1756, Sep. 2017.

[29] J. Zhang, R. Liu, K. Yin, Z. Wang, M. Gui, and S. Chen, "Intelligent collaborative localization among air-ground robots for industrial environment perception," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9673–9681, Dec. 2019.

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

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

[32] S. Li and D.-Y. Yeung, "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 4140–4146.
[33] L. Wen *et al.*, "VisDrone-SOT2018: The vision meets drone single-object tracking challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2019, pp. 469–495.
[34] H. Fan *et al.*, "VisDrone-SOT2020: The vision meets drone single object tracking challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2020, pp. 728–749.
[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Found. *Trends Mach. Learn.*, vol. 3, pp. 1–122, 2010.
[36] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–11.
[37] M. Faessler, E. Mueggler, K. Schwabe, and D. Scaramuzza, "A monocular pose estimation system based on infrared LEDs," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 907–913.
[38] 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.*, 2015, pp. 4310–4318.
[39] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Adaptive correlation filters with long-term and short-term memory for object tracking," *Int. J. Comput. Vis.*, vol. 126, no. 8, pp. 771–796, 2018.
[40] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2014, pp. 254–265.
[41] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1430–1438.
[42] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1401–1409.
[43] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Aug. 2017.
[44] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6638–6646. [Online]. Available: https://ieeexplore.ieee.org/document/8100216
[45] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1396–1404. [Online]. Available: https://ieeexplore.ieee.org/document/8099635
[46] C. Wang, L. Zhang, L. Xie, and J. Yuan, "Kernel cross-correlator," in *Proc. AAAI Conf.* Artif. Intell., 2018, pp. 4179–4186.
[47] H. Ma, S. T. Acton, and Z. Lin, "SITUP: Scale invariant tracking using average peak-to-correlation energy," *IEEE Trans. Image Process.*, vol. 29, pp. 3546–3557, 2020.
[48] Z. He, Y. Fan, J. Zhuang, Y. Dong, and H. Bai, "Correlation filters with weighted convolution responses," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 1992–2000.
[49] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 472–488.
[50] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Y. Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2711–2720. [Online]. Available: https://ieeexplore.ieee.org/document/8099631
[51] L. Zhang and P. N. Suganthan, "Robust visual tracking via co-trained kernelized correlation filters," *Pattern Recognit.*, vol. 69, pp. 82–93, 2017.
[52] F. Li, Y. Yao, P. Li, D. Zhang, W. Zuo, and M.-H. Yang, "Integrating boundary and center correlation filters for visual tracking with aspect ratio variation," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 2001–2009.
[53] T. Zhang, C. Xu, and M.-H. Yang, "Multi-task correlation particle filter for robust object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4335–4343. [Online]. Available: https://ieeexplore.ieee.org/document/8099995
[54] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4844–4853.
[55] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1369–1378.
[56] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised deep tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1308–1317.
[57] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2805–2813. [Online]. Available: https://ieeexplore.ieee.org/document/8100014
[58] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1763–1771. [Online]. Available: https://ieeexplore.ieee.org/document/8237458
[59] N. Wang, W. Zhou, Y. Song, C. Ma, W. Liu, and H. Li, "Unsupervised deep representation learning for real-time tracking," *Int. J. Comput. Vis.*, vol. 129, pp. 400–418, 2021.
[60] J. Zhao, K. Dai, D. Wang, H. Lu, and X. Yang, "Online filtering training samples for robust visual tracking," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 1488–1496.
[61] 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.

Junjie Ye (Graduate Student Member, IEEE) was born in Yibin, China. He received the B.E. degree in mechanical engineering in 2020 from Tongji University, Shanghai, China, where he is currently working toward the M.Sc. degree in mechanical engineering.

His research interests include visual object tracking, deep learning, and robotics.

Changhong Fu (Member, IEEE) received the Ph.D. degree in robotics and automation from the Computer Vision and Aerial Robotics Lab, Technical University of Madrid, Madrid, Spain, in 2015.

During his Ph.D., he held two research positions with Arizona State University, Tempe, AZ, USA, and Nanyang Technological University (NTU), Singapore. He then worked with the NTU as Postdoctoral Research Fellow. He is currently an Associate Professor with the School of Mechanical Engineering, Tongji University, Shanghai, China. He is leading more than five projects related to the vision for unmanned systems. He has authored or coauthored more than 70 journal and conference papers (in publications including IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, *IEEE Geoscience and Remote Sensing Magazine*, IEEE/ASME TRANSACTIONS ON MECHATRONICS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, the IEEE/CVF Conference on Computer Vision and Pattern Recognition, the IEEE/CVF International Conference on Computer Vision, the IEEE International Conference on Robotics and Automation, and the IEEE/RSJ International Conference on Intelligent Robots and Systems) related to the intelligent vision and control for unmanned aerial vehicle. His research interests include intelligent vision and control for unmanned systems in complex environment.

Fuling Lin (Graduate Student Member, IEEE) received the B.Eng. degree in mechanical engineering from Tongji University, Shanghai, China, in 2019, where he is currently working toward the M.Sc. degree in mechanical engineering.

His research interests include robotics, visual object tracking, and computer vision.

Fangqiang Ding (Student Member, IEEE) is working toward the B.Eng. degree in mechanical engineering with Tongji University, Shanghai, China.

His research interests include unmanned aerial vehicles and computer vision.

Shan An (Member, IEEE) received the B.E. degree in automation engineering from Tianjin University, Tianjin, China, in 2007, and the M.E. degree in control science from Shandong University, Jinan, China, in 2010. He is currently working toward the Ph.D. degree in computer vision with the School of Computer Science and Engineering, Beihang University, Beijing, China.

His research interests include deep learning, visual simultaneous localization and mapping, and robotics.

Mr. An was a Program Committee Member for ACM Multimedia 2019/2020 and ACM Multimedia Asia 2019. He is a Reviewer for IEEE TRANSACTION ON NEURAL NETWORK AND LEARNING SYSTEMS, IEEE TRANSACTION ON MULTIMEDIA, *Pattern Recognition*, and *Pattern Recognition Letters* and International Joint Conference on Artificial Intelligence 2021.

Geng Lu (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in control science and engineering from the Department of Automation, Tsinghua University, Beijing, China, in 1999, 2002, and 2004, respectively.

From 2004 to 2006, he was a Postdoctoral Scholar with the Department of Electrical Engineering, Tsinghua University. Since 2006, he has been with the Department of Automation, Tsinghua University, where he is currently an Associate Professor. His research interests include robust control, nonlinear control, signal processing, and aerial robots.