Object Tracking in UAV Videos by Multifeature Correlation Filters With Saliency Proposals

Yan Zhang and Yuhui Zheng, *Member, IEEE*

***Abstract*—The purpose of object tracking is to locate a given target in image sequence, such as people and vehicles. In recent years, with the development of unmanned aerial vehicle (UAV) technology, object tracking in UAV videos has engaged many scholars. It has been widely used in traffic control, water quality inspection, wildlife census, and other fields. However, low resolution, scale change, occlusion, and other challenges have been restricting the development of the tracker. To solve the aforementioned problems, we put forward multifeature correlation filters with saliency proposals. First, we use histogram of oriented gradient features, gray (I) features, and color names features to heighten the representation information of the target, so that our algorithm can accurately locate small targets. Then, we introduce saliency proposals to reposition the occluded target. Finally, we use dynamic update weights instead of the fixed update weights to mitigate the adverse effects caused by template degradation. Experiments demonstrate that our tracker has achieved satisfactory tracking accuracy and AUC scores have reached 0.462, 0.417, and 0.425 on UAV123@10FPS, UAV20 L, and UAVDT datasets, respectively.**

***Index Terms*—Correlation filter, object tracking, saliency proposals, unmanned aerial vehicle (UAV) videos.**

I. Introduction

OBJECT tracking has always been a hot topic in the field of computer vision [45], [46]. It is mainly used to continuously predict target position according to the information in the initial frame. In view of its strong practical application value, it has been diffusely put into use in national defense, industrial manufacturing, and other fields. So far, many tracking algorithms have been proposed and achieved excellent performance. However, most of these algorithms are only applicable to videos taken by cameras. When they are used in UAV videos, their performance tends to decline significantly.

The above reason is mainly due to some unique challenges in the UAV scenario. Let us see Fig. 1(a) for an instance, the two cars have the same apparent characteristics, i.e., their contour and color are similar, which is easy to make the algorithm track the wrong target and ultimately lead to tracking failure. Fig. 1(b) also makes clear that the target has very low resolution, which often only takes up dozens of pixels. In this situation, most algorithms are difficult to extract enough feature information. Besides, occlusion is also one of the challenges. From Fig. 1(c), we can see that the car is occluded by the brand and will eventually disappear from view. When the target reappears, how to relocate the target is a problem that all algorithms have to consider.

(a) (b) (c)

Fig. 1. Some challenges in UAV videos. (a) Similar background. (b) Low resolution. (c) Occlusion.

Up to now, a mass of algorithms have been put forward. The most popular algorithms are based on deep learning [18], [22], [41] and correlation filter [1], [2], [4], [5], [6], [10], [12], [13], [20], [24], [29], [48]. The algorithms based on deep learning relies on the robust feature extraction capability of a neural network to obtain the efficient representation of the target. Dsiam [18] adaptively fuses the deep features of different layers, and further improves the model performance by suppressing background information. Hu et al. [22] fused the appearance information and motion information, and the two features complement each other to predict the target more accurately. STN-Track [41] introduces the transformer [32] to stengthen the global interaction capability of the algorithm. But the above algorithms rely on a general processing unit (GPU) to speed up the calculation, which is difficult to deploy on UAVs. Therefore, the algorithms based on correlation filter seems to be more suitable for UAV tracking.

Correlation operation is used to describe the similarity between two objects. The similarity is proportional to the response value. In view of this, minimum output sum of squared error (MOSSE) [2] uses a correlation filter for object tracking for the first time. Depending on Fourier transform, its speed can

Manuscript received 11 March 2023; revised 26 April 2023; accepted 1 June 2023. Date of publication 5 June 2023; date of current version 26 June 2023. This work was supported in part by Natural Science Foundation of Jiangsu Province under Grant BK20211539, in part by the National Natural Science Foundation of China under Grant U20B2065 and Grant U22B2056, in part by the Qing Lan Project, and in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province Under Grant KYCX23_1372. *(Corresponding author: Yuhui Zheng.)*

Yan Zhang is with the School of Computer, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: 20211220042@nuist.edu.cn).

Yuhui Zheng is with the School of Computer, Nanjing University of Information Science and Technology, Nanjing 210044, China, and also with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: zheng_yuhui@nuist.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3283094

reach 600FPS. On the basic of MOSSE, kernelized correlation filter (KCF) [20] introduces kernel trick, which greatly improves the discrimination ability of the classifier. Considering that the target scale is constantly changing, it is difficult to fit the target effectively with a fixed scale. Adaptive correlation filter with long-term and short-term memory (ILCT) [29] introduces a correlation filter responsible for target confidence and a correlation filter responsible for adjusting scale. To improve efficiency, spatial correlation filter (SCF) [48] reconstructs the support vector machine (SVM) model by taking advantage of the properties of the circular matrix, and combines the alternating optimization process with the discrete Fourier transform to get the optimal solution in real time.

However, boundary effects caused by cyclic sampling inevitably limits the performance of the algorithm. Spatial-temporal adaptive feature weighted correlation filter (FWDCF) [12] assigns different weights to pixels by constructing adaptive feature weights, which not only suppresses background information, but also enhances target information. Inspired by passive attack learning, spatial-temporal regularized correlation filter (STRCF) [24] applies temporary regularization to a correlation filter. A dual color clustering and spatio-temporal regularized correlation regressions-based complementary tracker (CSCT) [13] introduces a two-color clustering histogram model on the basic of STRCF. For the purpose of improving the separability, some algorithms [1], [6], [10] combine the correlation filter and deep feature to heighten the robustness of the tracker. A correlation filter-based dual-flow tracker (DFTrack) [5] uses additional motion features to enable the tracker to track the object more accurately. Besides, Chen et al. [4] used a storage unit to store the historical information of the target, which greatly alleviated the negative impact caused by the change of the target. In view of various challenges in the UAV scenario, we propose the multifeature correlation filters with saliency proposals. Specifically, in order to describe the target in the UAV video more accurately, we use histogram of oriented gradient (HOG), color names (CN), and gray (I) features, and use the peak-to-side lobe ratio (PSR) to adaptively fuse these features. In addition, due to the frequent occurrence of occlusion, the saliency proposals strategy is introduced to enable the algorithm to continue tracking after the object reappears. Finally, we use adaptive weights instead of fixed weights to prevent template degradation. Our main contributions can be summarized as follows.

1) We integrate HOG, CN, and I features to get the efficient representation of the target, and use PSR for feature fusion, so that the fused features can be more robust to various challenges. The application of multiple features further improves the discriminant ability of the classifier.
2) We introduce the saliency proposals strategy to enable the algorithm to continue tracking after occlusion. In this case, the long-term tracking capability of the algorithm can be effectively improved.
3) We use dynamic weights instead of the original fixed weights to update the model. The adaptive model update strategy can effectively prevent the template from being polluted by noise and make the algorithm more robust.

The rest of this article is organized as follows. Next, related works are introduced in Section II. We present our algorithm in detail in Section III, including features fusion, saliency proposals, and the dynamic weight update strategy. In Section IV, we list the results of our algorithm under multiple benchmarks. We analyze and discuss the results in Section V. Finally, Section VI concludes this article.

II. Related Works

A. Object Tracking by Correlation Filters

Correlation operations are used to describe similarities between two objects. The similarity is proportional to the response value. The main idea of correlation filters is correlation operations. These algorithms use a mass of samples to train correlation filters, and then perform correlation operations between the trained correlation filters and the search image. The location of the target is considered to have the greatest response. In addition, these algorithms use Fourier transforms to improve efficiency by avoiding tedious calculations. MOSSE [2] first applied this idea to object tracking in 2010. Thanks to the Fourier transform, its speed can reach 600FPS. Because MOSSE only uses the I feature and the number of training samples is small, its accuracy is not satisfactory. The real-time tracking speed has attracted a mass of scholars, and correlation filters have developed rapidly since then.

In order to train more robust correlation filters, circulant structure tracking with kernels (CSK) [19] uses cyclic sampling to increase training samples. More samples significantly improve the discriminant ability of the correlation filter to the target. But in the meantime, it also inevitably introduces the boundary effect. Background-aware correlation filter (BACF) [23] uses real training samples to train the correlation filters rather than the false samples produced by shift sampling. Spatially regularized correlation filter (SRDCF) [9] raises the spatial constraint weights to the ridge regression function to restrain background noise. On this basis, STRCF [24] introduces a temporary regularization term on the basic of SRDCF to further alleviate the influence of boundary effects. But SRDCF and STRCF use fixed regularization constraints, which is obviously unreasonable. Saliency-aware dual regularized correlation filter (DRCF) [15] adapts to adjust the constraint weight through the saliency-aware strategy, which effectively alleviates the above problem. Although the regularization constraint effectively alleviates the influence of boundary effect on algorithm performance, it also reduces the solving efficiency of objective function, which ultimately reduces the speed of algorithm. Weighted sample based correlation filter (WSCF) [17] introduces a simple objective function that not only supposes background noise interference, but also causes little extra time consumption.

Besides, Fu et al. [14] proposed multifeature learning as a way to enhance target representation. The use of multiple features fully considers the limitations of each feature, so that various features complement each other, making the algorithm to be applied to more challenging scenarios. Real-time UAV tracking

based on PSR stability (PSRS) [40] also involve the use of multiple features. Consider that the size of the target will change as it moves. How to make the bounding box adapt to the target size is a problem that all algorithms have to consider. Scale adaptive kernel correlation filter (SAMF) [26] and accurate scale estimation for robust visual tracking (DSST) [7] raise the scale estimation strategy. In this case, the algorithm is able to adjust the size of the bounding box in real time to better track the target. Many algorithms [1], [6], [10] now combine correlation filters with deep learning to complement the stengths of each other.

B. Re-Detection Mechanism

Occlusion is easy to occur when the target is moving. As shown in Fig. 1(c), the target is occluded by an obstacle. At this point, the target completely disappears from view, which means that no algorithm can locate the target. The target in the bounding box must be the background. At this time, the update of the model will bring adverse effects. However, a robust algorithm should be able to reposition the target when the occlusion is over. Therefore, many algorithms with a redetection mechanism have been proposed.

Wang et al. [39] combined correlation filters with deep learning. Specifically, correlation filters are used for object tracking, but when tracking quality in the current frame is not good, you only look once version 3 (YOLOv3) is deployed into the algorithm to prevent the tracker from tracking the wrong target. The method in [31] expands the area where the target may appear, and then calculates a pixel-by-pixel color score map, which is used for redetection. Given that the color score of each pixel represents the probability that it is the target, the target location can be determined through the score map. Reliable re-detection for long-term tracking (RDCF) [36] finds the rough location of the target by sparse coding, and then selects several candidate regions by particle filter. Finally, these candidate regions are scored by minimum reconstruction error. Similar to RDCF, Wang et al. [34] used inverse sparse representation directly to reposition the target.

Because of the lower resolution of the target in a UAV video, occlusion occurs more frequently. Therefore, a redetection algorithm is necessary for UAV tracking. Inspired by these algorithms, we propose a redetection algorithm based on saliency proposals. Specifically, we calculate the saliency information of the image patch to determine the rough area, and then perform the correlation operation on these positions with correlation filter. Finally, the exact location is obtained through the response patch.

C. Model Update Strategies

Most algorithms use a fixed learning rate to update the model, which can easily lead to model degradation. If occlusion occurs, inappropriate updates at this time will reduce the performance of the tracker. To deal with the difficulty, many algorithms have been presented.

PSRS [40] introduced PSR stability, and the model is updated only when the PSR stability is higher than a fixed value. In [43], a stability function is used to assess the confidence of the response patch, updating the model only when it is greater than the mean of the historical frames. But the above methods still only use a fixed learning rate to update the model. Xue et al. [42] used a moving average to update the model. In addition, to cope with occlusion, the model is updated by the template of the first frame. Fu et al. [16] updated the classifier with Gaussian process regression. Given some serious challenges in UAV videos, it is necessary to introduce an adaptive update strategy into the tracking algorithm.

III. Proposed Method

In this section, we will introduce the proposed algorithm in detail, including feature fusion, saliency proposals, and the dynamic weight update strategy. The flowchart is presented in Fig. 2. First, we extract HOG, CN, and I features of image patch, and fuse these features through PSR to generate the final response patch. We then determine if redetection is necessary by calculating PSR of the final response patch. That is, if the PSR is less than the predesigned threshold, we consider the current frame to be of poor quality and use saliency proposals to relocate the target. Finally, we use dynamic weights to replace the original fixed weights to prevent template degradation.

A. Feature Fusion

Considering that the target in UAV videos often has low resolution, it is difficult to describe the target effectively with a single feature. Therefore, we combine HOG, CN, and I features to enhance the target representation information. As we all know, different features can represent information at different levels of the target. For example, HOG and CN represent the contour and color information of the target, respectively. In this case, the fusion of the above features can make the algorithm applicable to more challenging scenarios. In addition, we use PSR to enhance and fuse these features. The definition of PSR is shown below

$$PSR(i) = \frac{g_{\max}(i) - \mu_{sl}(i)}{\sigma_{sl}(i)} \tag{1}$$

where $g_{\max}()$ is the maximum response and $i \in \{HOG,\ CN,\ I\}$. μ_{sl} and σ_{sl} are the mean and variance of the sidelobe, respectively.

We perform correlation operation between the feature patches and the correlation filters to obtain the corresponding response patches $Res(i)$. From Fig. 3, we can see that these response patches are uneven and have multiple peaks, which is unfavorable to the algorithm. In order to make these response patches smoother, we enhance them through PSR

$$R(i) = PSR(i)Res(i) \tag{2}$$

where $R()$ is the enhanced response patch.

Through (1) and (2), we can obtain enhanced response patches $R(HOG), R(CN)$, and $R(I)$. Next, we will fuse these enhanced features. Since we fuse two features and perform the same operation each time, we take $R(HOG)$ and $R(CN)$ as examples for illustration. The fusion process is shown below

$$R(HC) = R(HOG) \odot R(CN) \tag{3}$$

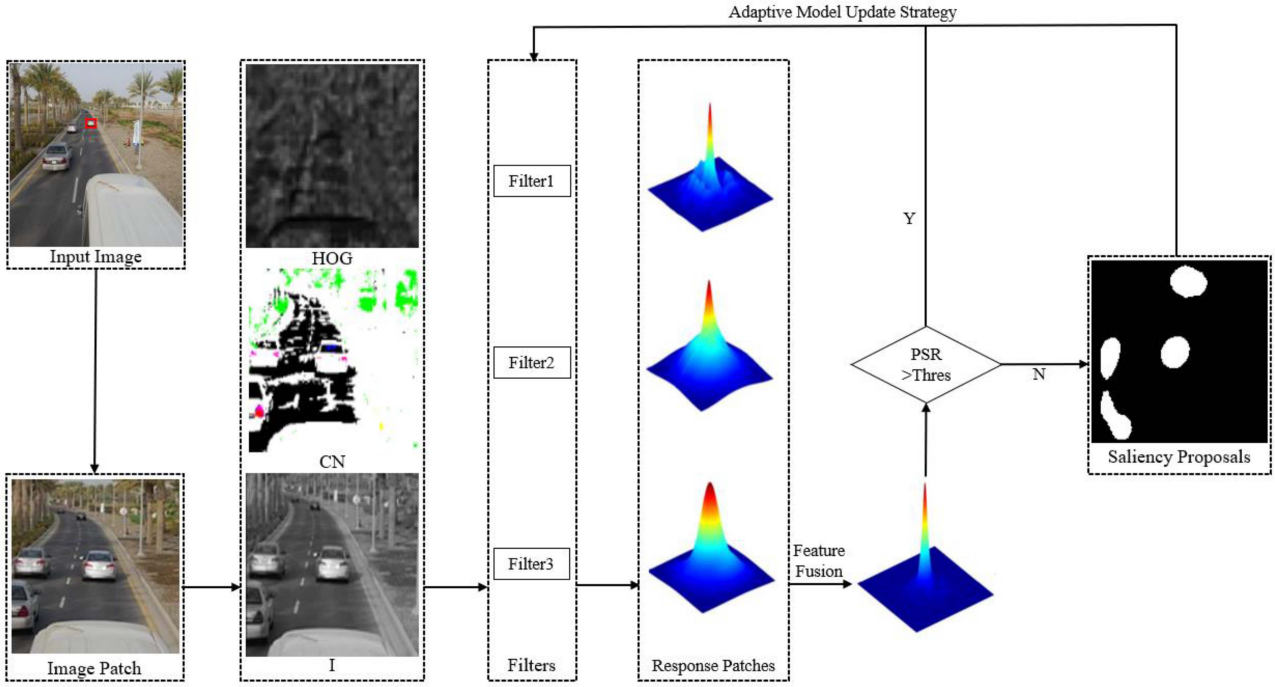

Fig. 2. Flowchart of proposed algorithm.

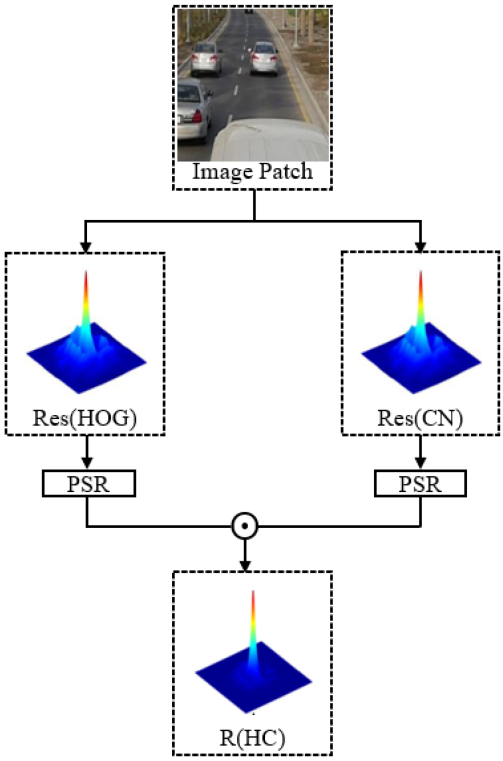

Fig. 3. Features enhancement process.

where $R(HC)$ is the fused response patch and $\odot$ denotes element-wise multiplication. Repeat the above operation, and we can get the response patch $R(HI)$ that fuses $R(HOG)$ and $R(I)$, and $R(CI)$ that fuses $R(CN)$ and $R(I)$. As shown in Fig. 3, compared with the initial response patch, the enhanced response patch is smoother and has an obvious peak.

Finally, in order to get the final response patch, we use PSR to adaptively fuse these response patches

$$R(final) = \sum_j \frac{PSR(j)}{\sum_k PSR(k)} R(j) \tag{4}$$

where $k,\ j \in \{HC,\ HI,\ CI\}$.

B. Saliency Proposals

The target is easily occluded when moving, which brings severe challenges to the tracker. In this case, most algorithms cannot effectively track the objects that reappear after occlusion. To effectively locate the occluded target after the target reappears, the saliency proposals are introduced. Considering the fact that there are often large differences in targets and backgrounds, the saliency information can help the tracker reposition the target. Specifically, the saliency proposals strategy consists of the following two parts: 1) the saliency information of image patch is extractedand 2) look for the areas that need to be redetected.

We utilize the approach in [21] to obtain the saliency features of the image. First, the image $I(k)$ in kth frame is preprocessed. We get the amplitude feature $A(k)$ and phase feature $P(k)$ of image $I(k)$ by Fourier transform $F()$. The equations are as follows:

$$A(k) = AC\left(F\left(I(k)\right)\right) \tag{5}$$

$$P(k) = PC\left(F\left(I(k)\right)\right). \tag{6}$$

Fig. 4. Redetection based on saliency proposals. (a) Occluded target. (b) Saliency map. (c) Saliency proposals.

The log spectrum representation $L(k)$ can be obtained by

$$L(k) = \log\left(A(k)\right) \tag{7}$$

where $log()$ is log function. $A(k)$ can be approximately expressed by the convolution between H_n and $L(k)$, so the spectral residual $r(k)$ can be obtained by following

$$r(k) = H_n - H_n \star L(k) \tag{8}$$

where $\star$ denotes the convolution operator and H_n is defined as

$$H_n = \frac{1}{n^2}\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}. \tag{9}$$

The saliency features $S(k)$ in the image patch can be acquired by $r(k)$, and the saliency features are obtained by following

$$S(k) = G(k) \star F^{-1}[\exp\left(r(k) + P(k)\right)]^2 \tag{10}$$

where $G(k)$ is a Gaussian filter and F^{-1} is inverse Fourier transform.

As shown in Fig. 4(a), when the occlusion is over, the car comes back into our view. To ensure that our algorithm can continue tracking this car, we cut out an image patch and extract the saliency information of the image patch through (5)–(10). The saliency map is shown in Fig. 4(b). Finally, we use the adaptive thresholding [3] to segment the saliency map to obtain the saliency proposals. In Fig. 4(c), the bright area is the area to be redetected. After obtaining the saliency proposals, we carry on the correlation operation between the filters and the areas to be redetected to obtain the response patches of the redetected areas. We think that in all response patches, the value represents the probability that the corresponding position is the target.

C. Adaptive Model Update Strategy

In object tracking task, the algorithm needs to process continuous image sequences. This also means that the target is constantly changing. Therefore, the robust algorithm should be able to update the model in real time to make the most of the temporal information of the target. Now, lots of algorithms based on correlation filter will update the model at each frame. However, they often use the fixed weight to update the model, which is obviously unreasonable. For example, when occlusion occurs, the target tracked by the algorithm is the background. However, most algorithms will continue to update the template, which will pollute the template and lead to template degradation. For a tracker, the template plays a very important role. The polluted template has adverse effects on the algorithm. To deal with the above-mentioned problem, we use the dynamic weight instead of the original fixed weight.

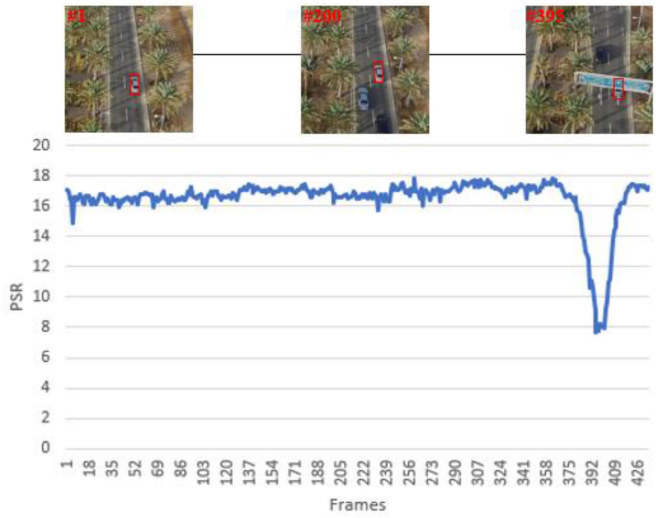

Fig. 5. PSR varies with target conditions.

From [2], we can know that the tracking status of the tracker in the current frame can be described by PSR. When the tracking quality is good, the PSR is often high. Otherwise, the PSR will shake violently. From Fig. 5, it is clear that occlusion occurs and PSR begins to drop sharply in the 358th frame. When the car is completely occluded, the PSR is the lowest. As the target gradually enters the field of vision, the PSR gradually rises. After occlusion, PSR inclines to be stable once more. In light of this, we intend to use PSR to measure the weight of the model. Specifically, if the tracker can accurately track the target, we will give a larger update weight; otherwise, the model has a small update weight to prevent the model from being polluted.

However, in different scenarios, the fluctuation range of PSR is also various. Therefore, we calculate the average value of historical frames to measure the tracking status of the current frame, as defined below

$$\beta = \frac{PSR_t}{mean\left(\sum_{i=1}^{t-1} PSR_i\right)} \tag{11}$$

where PSR_t is the PSR in the tth frame and $mean()$ denotes the mean function. By comparing the PSR of the current frame with the average PSR, we can evaluate the confidence of the tracking result of the current frame. The larger the β, the higher the confidence, and we will assign greater update weight to the current frame. Then, we use β to measure the update weight V. Considering that the target does not change significantly between adjacent frames, in order to prevent drastic changes in update weights, we adopt an exponential form for updating and set an initial learning rate

$$V = \frac{1}{1 + e^{\alpha + \gamma \times \beta}} \tag{12}$$

where α and γ are hyperparameters. Specifically, we use the following equation to update the model:

$$x_t^{\text{model}} = (1 - \eta V)\, x_{t-1}^{\text{model}} + \eta V x_t \tag{13}$$

where η is the initial update weight and x_{t-1}^{model} is the model in $(t-1)$th frame. x_t denotes the vectorized image.

IV. Experiments

A. Experimental Setup

In the experiment, our algorithm was implemented with MATLAB 2019a. The PC with an NVIDIA GTX 2080ti GPU and Intel I9-9900X CPU was used to carry out all the experiments. The dimensions of HOG features, CN features, and I features are 31, 11, and 1, respectively. The cell of HOG features is 4×4. α and γ are 9.2 and -10.4, respectively. The initial learning rate η is 0.0245. In each frame, the size of the search area is five times the size of the target. Besides, if the PSR of the fused response patch is less than 9, the saliency proposals will be performed.

B. Datasets and Compared Algorithms

We evaluate our tracker through the UAV123@10FPS [30] dataset, which contains 91 image sequences and 123 groundtruth. Unlike previous datasets, all image sequences in UAV123@10FPS are captured by UAVs, covering massive challenges such as camera motion, occlusion, and fast motion. 20 long image sequences are extracted from UAV123@10FPS, which constitutes the UAV20 L dataset. UAV20 L is mainly used to measure whether the algorithm can track the target for a long time. Besides, we also conduct relevant experiments on the unmanned aerial vehicle detection and tracking dataset (UAVDT) [11] dataset.

We compare our tracker with some classical trackers, such as SAMF, PSRS, DSST, STRCF, SRDCF, discriminative correlation filter tracker with channel and spatial reliability (CSRDCF) [27], discriminative scale space tracker (fDSST) [8], BACF and kernel cross-correlator (KCC) [33]. In addition, we also compare our tracker with some trackers on the basic of deep learning, such as hierarchical convolutional features tracker (HCFT) [28], integrate boundary and center correlation filter (IBCCF) [25], co-trained kernelized correlation filter (CoKCF) [44], multi-task correlation particle filter (MCPF) [47], unsupervised deep tracking (UDT)+ [35], fast efficient convolution operators (fECO) [37], learning unsupervised deep tracking (LUDT) [38], and improved learning unsupervised deep tracking (LUDT)+ [38].

C. Evaluation Metrics

Center location error (CLE) and overlap ratio (OR) are often used to evaluate the tracking results. Specifically, CLE refers to the distance between the predicted position and the groundtruth, which is defined as

$$CLE = \sqrt{(x_{tr} - x_{gt})^2 + (y_{tr} - y_{gt})^2} \tag{14}$$

where (x_{tr}, y_{tr}) is center coordinate of prediction result and (x_{gt}, y_{gt}) is center coordinate of the groundtruth. OR is used to represent the overlapping area between the predicted position and the groundtruth, which can be obtained by following

$$OR = \frac{S_{tr} \cap S_{gt}}{S_{tr} \cup S_{gt}} \tag{15}$$

where S_{tr} and S_{gt} represent the predicted position and the groundtruth, respectively. $\cap$ and $\cup$ refer to intersection and union, respectively.

On the basic of CLE and OR, we also use precision score (PS) and success score (SS) to quantitatively describe the tracking result of the tracker. FPS is also used to describe the speed of the tracker. In addition, we also use precision plots and success plots to compare various algorithms more intuitively. In this article, the legend of the precision plots shows the PS when the threshold is 20. The legend of success plots denotes the SS when the threshold is 0.5. Considering that the success plot is a closed curve, the area under the curve can be calculated, which is area under curve (AUC). We rank each tracker through AUC.

D. Comparison With Other Algorithms

1) Overall Performance Evaluation: We evaluate the overall performance of our tracker on UAV123@10FPS dataset. From Fig. 6(a), (b), and Table I, it is evident that our algorithm has achieved the best tracking result. The PS, SS, and AUC of our algorithm are 0.630, 0.559, and 0.462, respectively. Compared with the baseline, that is BACF, the AUC of our algorithm is improved by 0.049. Besides, STRCF, SRDCF and CSRDCF have all introduced regularization terms to mitigate the impact of boundary effects, so they have achieved good performance. In particular, the AUC of STRCF reaches 0.457, which ranks second. Because our algorithm is improved on the basis of BACF and still uses real image patches instead of samples generated by cyclic sampling to train the correlation filter, our algorithm can also effectively alleviate the boundary effect. Benefiting from the modules introduced in Section III, the AUC of our algorithm is higher than STRCF, which ranks first. For PSRS, it introduces PSR stability to improve the model, but it only uses PSR stability to update the model. In contrast, we not only use PSR to update the model dynamically, but also use PSR to enhance features. So compared with PSRS, the tracking result of our algorithm is greater then it. However, although the use of multiple features improves performance, it also causes lots of time. So the FPS can only achieve 13.

2) Long-Term Tracking Performance: As we all know, the longer the image sequence, the more challenges the target will experience, which also brings greater difficulties to the algorithm. Therefore, the long-term tracking performance is an important indicator to measure the performance of the tracker. In view of this, we carry on experiments on the UAV20 L benchmark, and the results are presented in Fig. 6(c), (d), and Table II. The PS, SS, and AUC of our algorithm are 0.596, 0.497, and 0.417, respectively, which shows that our algorithm has good long-term tracking performance. Due to the introduction of the temporal regularization term and spatial regularization

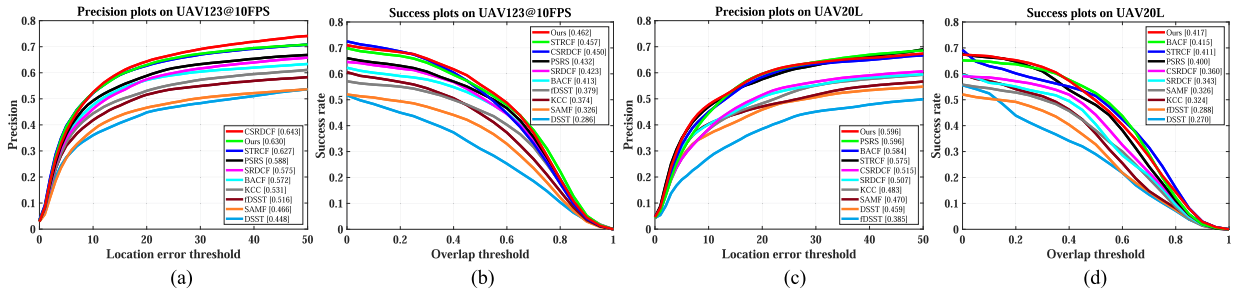

Fig. 6. Precision plots and success plots. The precision and AUC scores of success plots are given in brackets. (a) Precision plots on UAV123@10FPS. (b)Success plots on UAV123@10FPS. (c) Precision plots on UAV20L. (d) Success plots on UAV20L.

TABLE I
RESULTS ON UAV123@10FPS DATASET

	Ours	SAMF	STRCF	DSST	SRDCF	KCC	CSRDCF	PSRS	fDSST	BACF
PS	0.630	0.466	0.627	0.448	0.575	0.531	0.643	0.588	0.516	0.572
SS	0.559	0.397	0.544	0.311	0.511	0.447	0.536	0.526	0.459	0.506
AUC	0.462	0.326	0.457	0.286	0.423	0.374	0.450	0.432	0.379	0.413
FPS	13	12	26	98	13	40	11	20	153	52

TABLE II
RESULTS ON UAV20 L DATASET

	Ours	SAMF	STRCF	DSST	SRDCF	KCC	CSRDCF	PSRS	fDSST	BACF
PS	0.596	0.470	0.575	0.459	0.507	0.483	0.515	0.596	0.385	0.584
SS	0.497	0.381	0.515	0.290	0.405	0.369	0.442	0.480	0.327	0.525
AUC	0.417	0.326	0.411	0.270	0.343	0.324	0.360	0.400	0.288	0.415
FPS	17	13	22	68	9	35	10	16	98	40

term, the PS, SS, and AUC of STRCF are 0.575, 0.515, and 0.411, respectively, which ranks second, only inferior to our algorithm. From Table II, we can see that the AUC of most algorithms is less than 0.4, which means that their long-term tracking performance is poor. We think that occlusion is more likely to occur in long-term tracking tasks, and the saliency proposals strategy can effectively relocate the occluded target to mitigate the adverse impact of occlusion. In addition, because the algorithm based on correlation filter needs to update the model every frame, inappropriate updates will accumulate frame by frame. This phenomenon is more obvious in long-term tracking tasks. The adaptive model update strategy can effectively avoid inappropriate updating of model to solve the above problem.

3) Comparison With Trackers Based on Deep Learning: We compare our tracker with the trackers on the basic of deep learning on the UAVDT dataset. As shown in Table III, the AUC of our algorithm is significantly higher than other trackers. However, because we fuse HOG features, CN features, and I features of the image patch, which inevitably cause additional time consumption. In this case, the speed of our algorithm is not satisfactory. However, it is worth noting that these algorithms based on deep learning rely on GPU to speed up the calculation. However, our algorithm does not depend on GPU. In addition, one of the advantages of a UAV is that it is lightweight, which makes it difficult to carry additional hardware equipment. Therefore, our tracker is more suitable for UAV tracking.

TABLE III
RESULTS ON UAVDT DATASET

Tracker	AUC	FPS	GPU
HCFT	0.355	19	✓
IBCCF	0.389	3	✓
CoKCF	0.319	20	✓
MCPF	0.403	0.6	✓
UDT+	0.415	56	✓
fECO	0.415	20	✓
LUDT	0.418	78	✓
LUDT+	0.406	59	✓
Ours	0.425	16	✗

4) Attribute-Based Evaluation: We test the performance of our algorithm in various attributes, and the results are presented in Fig. 7. We introduce scale pools into our algorithm to select the best target scale by calculating the maximum response. Although the size of the target often changes, our algorithm still achieves excellent performance. Similar object and background clutter will have a negative impact on the tracker, making the predicted position gradually deviate from the groundtruth. However, the use of multiple features can make the algorithm locate the target more accurately, which makes the predicted position have

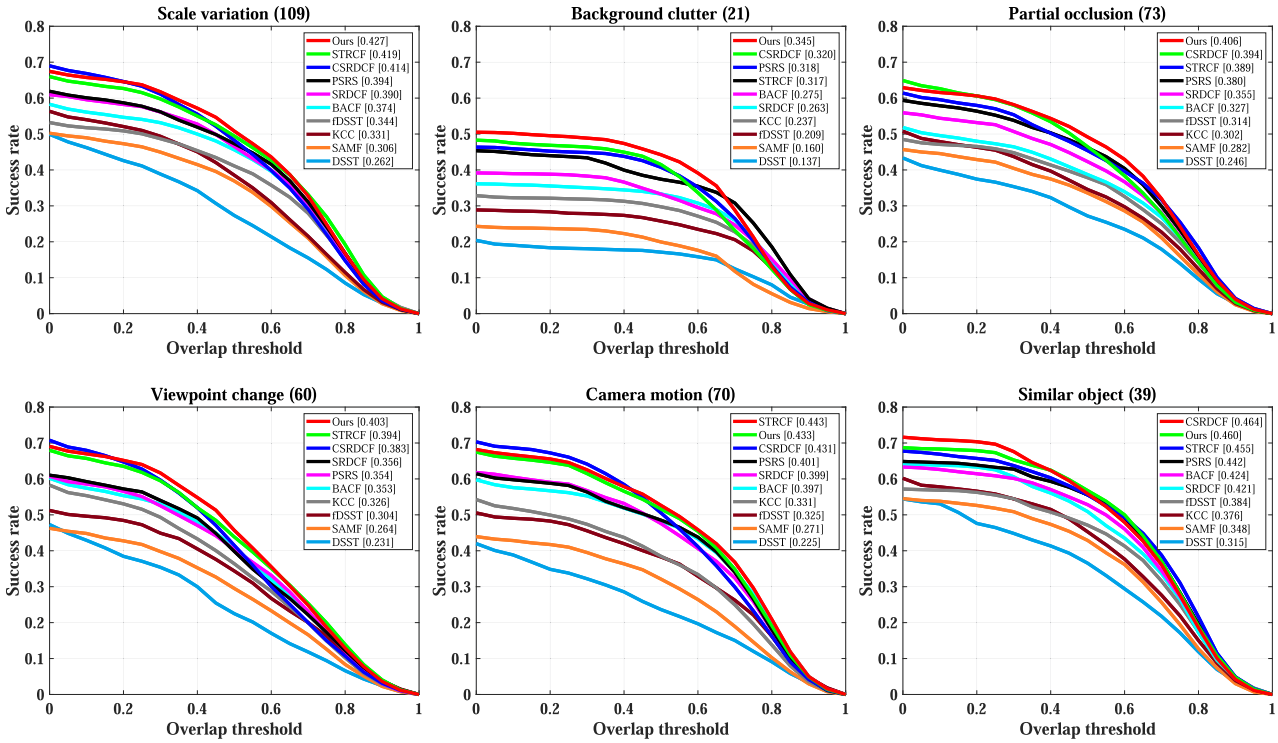

Fig. 7. Precision plots and success plots. The precision and AUC scores of success plots are given in brackets.

higher confidence. In this case, the impact of similar background and background clutter will be greatly mitigated. Besides, the saliency proposals strategy enables our algorithm to track the occluded target again. From Fig. 7, it is evident that our algorithm outperforms other algorithms when occlusion occurs. Viewpoint change and camera motion are unique challenges in a UAV video, which is also the biggest difference from a traditional video. It is clear that our algorithm has achieved satisfactory performance in both scenarios.

5) Visualization Results: In order to qualitatively evaluate our algorithm, we show some visualization results, as shown in Fig. 8. In the first image sequence, its length exceeds 2000 frames, which is mainly used to measure whether the algorithm can track the target for a long time. Obviously, in the 2100th frame, our algorithm can still track the cyclist very well. However, other algorithms obviously start to appear tracking drift, i.e., the bounding boxes are obviously larger than the size of the target. In the second image sequence, the target of tracking is a ship, which is gradually away from us. In this image sequence, the biggest challenge is scale variation. At the beginning, the size of the target is large. But later, the size gradually decreased, and finally became only a point. It is apparent that our algorithm can effectively fit the scale variation of the target. But both CSRDCF and SAMF have failed to track the ship. In the third image sequence, the tracked target has very low resolution, which only takes up dozens of pixels. For most algorithms, it is difficult to extract enough information. Thanks to the joint action of HOG, CN, and I features, our tracker can still track the car with low resolution. However, all of other trackers end in disaster. It can be seen that our tracker is obviously superior to other trackers in this challenging scenario. Let us look at the fourth image sequence. The target is a surfer. In the process of surfing, deformation often occurs, which easily makes the algorithm introduce too much background noise. Even so, our tracker still achieves satisfactory performance. In short, our tracker can be used in multiple scenarios, such as scale variation, low resolution, and so on.

TABLE IV
RESULTS OF ABLATION STUDIES

	Baseline	+P1	+P1+P2	+P1+P2+P3
PS	0.572	0.617	0.622	0.630
SS	0.506	0.542	0.553	0.559
AUC	0.413	0.448	0.456	0.462

E. Ablation Studies

To verify the function of feature fusion (P1), the saliency proposals strategy (P2), and the dynamic update weight strategy (P3), we carry on ablation studies on the UAV123@10FPS dataset and the results are shown in Table IV. Because our algorithm is improved on the basis of BACF, we use BACF as the baseline for comparison. The combination of HOG, CN, and I features effectively enhances the representation information of the target and further improves the performance of the algorithm. We can see that benefiting from the use of multiple features, the PS, SS, and AUC of the algorithm increase by 0.045, 0.036, and 0.035, respectively. It can be said that feature fusion has

Fig. 8. Visualization results.

greatly improved the performance of our algorithm. Besides, as described in Section III-B, the saliency proposals strategy can effectively relocate the occluded target, which greatly improves the performance of our algorithm in occlusion scenes. So the PS, SS, and AUC increases by 0.005, 0.011, and 0.008, respectively. Finally, when all modules are used, i.e., the algorithm we proposed, the PS, SS, and AUC of our algorithm reach 0.630, 0.559, and 0.462, respectively. Compared with the baseline, that is BACF, the AUC increases by 0.049. In general, feature fusion enhances the representation information of the target, the saliency proposals strategy is conducive to tracking the occluded target, and the dynamic update weight strategy can effectively prevent template degradation. From Table IV, it is evident that the above modules are beneficial to the improvement of the performance.

V. Discussion

In this article, in order to accurately track targets in UAV videos, we propose the multifeature correlation filters with saliency proposals. As is well known, compared to videos captured by cameras, targets in UAV videos often have lower resolution, which means that algorithms are difficult to extract sufficient representation information. To address the above issue, we use HOG, CN, and I features to enhance the discriminative ability of the algorithm. The existing algorithms [7], [26], [40] fully demonstrate that the use of multiple features is beneficial for improving tracking accuracy. However, our algorithm utilizes PSR to enhance various features before fusing them. From Table I, we can see that our algorithm has achieved the best performance. Besides, Fig. 3 shows that the enhanced response patch is smoother, considering that algorithms based on correlation filters determine the position of targets through maximum response. Therefore, a smooth response patch is more advantageous for the algorithm to accurately predict the position of the target. However, although the use of multiple features improves the accuracy of the algorithm, it inevitably increases computational complexity, which makes it difficult for our algorithm to track targets in real-time. The FPS of our algorithm is about 15. The future work will focus on achieving a balance between tracking accuracy and tracking speed. Specifically, the representation ability of manual features is too weak, such as HOG features that can only describe the contour features of the target, and CN features that can only describe the color features of the target. Considering that deep neural networks have stronger representation capabilities, we plan to replace manual features with deep features. The shallow deep features can describe the texture, color, and other features of the target, which means we do not need to use very complex network structures. In this case, feature extraction will not consume a significant amount of time.

In UAV videos, occlusion is also an issue that cannot be ignored. From Fig. 1(c), we can see that after the target is occluded, the algorithm cannot continue tracking the target. In order to enable our algorithm to reposition the target, we introduce saliency proposals. Due to the overly complex network structure of existing detection algorithms based on deep learning, they often rely on GPU that is difficult to be carried on UAVs for accelerating computation. Therefore, we use the saliency features of an image patch to determine the areas that

need to be redetected. Fig. 7 shows that our algorithm has achieved excellent performance in occlusion attribute. Besides, saliency proposals enable our algorithm to reposition the target after occlusion ends, which also improves the long-term tracking performance.

Another reason why our algorithm has robust long-term tracking performance is that the adaptive model update strategy effectively prevents template degradation. As is well known, when the target is occluded, updates at this time often contaminate the template, which will seriously affect the performance of the algorithm. We use the PSR of each frame to dynamically adjust the update weights. When the target is occluded, the model hardly updates, which effectively prevents template contamination. When the target reappears, our algorithm still exhibits good robustness.

In general, the use of multiple features enhances the representation information of the target, saliency proposals help our algorithm reposition occluded target, and the adaptive model update strategy effectively prevents template degradation. The ablation experiments verify the effectiveness of the above strategies.

VI. Conclusion

In this article, we propose the multifeature correlation filters with saliency proposals. Specifically, we use HOG, CN, and I features to better describe the target, and use PSR to enhance these features. The enhanced feature patch is smoother, which is conducive to the algorithm to accurately predict the location of the target. In addition, considering that occlusion often occurs in a UAV video, we introduce the saliency proposals strategy to help our algorithm reposition the occluded target. In this case, the long-term tracking performance of the algorithm will be greatly improved. Finally, we use dynamic update weight to replace the original fixed weight to adapt to the complex scene in UAV videos. Experiments on several datasets show that our algorithm can achieve satisfactory accuracy. However, our algorithm is difficult to track the target in real time. In the future work, we intend to use deep features to replace manual features. On the one hand, deep features have stronger representation ability than manual features. On the other hand, the use of shallow features will not cause huge time consumption. Based on this, the tracking accuracy and speed are expected to reach a balance.

References

[1] G. Bhat, J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg, "Unveiling the power of deep tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 483–498.

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

[3] D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *J. Graph. Tools*, vol. 12, no. 2, pp. 13–21, 2007.

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

[5] Y. Chen, Y. Tang, Z. Yin, T. Han, B. Zou, and H. Feng, "Single object tracking in satellite videos: A correlation filter-based dual-flow tracker," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6687–6698, 2022.

[6] 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.

[7] 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.

[8] 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.

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

[10] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 472–488.

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

[12] F. Du, P. Liu, W. Zhao, and X. Tang, "Spatial–temporal adaptive feature weighted correlation filter for visual tracking," *Signal Process.: Image Commun.*, vol. 67, pp. 58–70, 2018.

[13] J. Fan, H. Song, K. Zhang, Q. Liu, and W. Lian, "Complementary tracking via dual color clustering and spatio-temporal regularized correlation learning," *IEEE Access*, vol. 6, pp. 56526–56538, 2018.

[14] C. Fu, F. Lin, Y. Li, and G. Chen, "Correlation filter-based visual tracking for UAV with online multi-feature learning," *Remote Sens.*, vol. 11, no. 5, 2019, Art. no. 549.

[15] 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.

[16] C. Fu, Y. Zhang, R. Duan, and Z. Xie, "Robust scalable part-based visual tracking for UAV with background-aware correlation filter," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2018, pp. 2245–2252.

[17] R. Han, W. Feng, and S. Wang, "Fast learning of spatially regularized and content aware correlation filter for visual tracking," *IEEE Trans. Image Process.*, vol. 29, pp. 7128–7140, 2020.

[18] A. He, C. Luo, X. Tian, and W. Zeng, "A. twofold siamese network for real-time object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4834–4843.

[19] Joao 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.*2012, pp. 702–715.

[20] 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.

[21] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[22] Z. Hu, D. Yang, K. Zhang, and Z. Chen, "Object tracking in satellite videos based on convolutional regression network with appearance and motion features," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 783–793, 2020.

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] A. Lukezic, T. Vojir, L. C. 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.

[28] 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.

[29] 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, pp. 771–796, 2018.

[30] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 445–461.
[31] F. Tang and Q. Ling, "Contour-aware long-term tracking with reliable re-detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4739–4754, Dec. 2020.
[32] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, vol. 30, pp. 6000–6010.
[33] C. Wang, L. Zhang, L. Xie, and J. Yuan, "Kernel cross-correlator," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, vol. 32, pp. 4179–4186.
[34] H. Wang, W. Ma, S. Zhang, G. Chen, H. Ge, and Y. Du, "Robust visual object tracking with multiple features and reliable re-detection scheme," *IEEE Access*, vol. 8, pp. 98810–98826, 2020.
[35] 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.
[36] N. Wang, W. Zhou, and H. Li, "Reliable re-detection for long-term tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 730–743, Mar. 2019.
[37] 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.
[38] 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.
[39] X. Wang, K. Zhang, S. Li, Y. Hu, and J. Yan, "An optimal long-term aerial infrared object tracking algorithm with re-detection," *IEEE Access*, vol. 7, pp. 114320–114333, 2019.
[40] 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.
[41] X. Xu et al., "STN-track: Multiobject tracking of unmanned aerial vehicles by swin transformer neck and new data association method," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 8734–8743, 2022.
[42] X. Xue, Y. Li, H. Dong, and Q. Shen, "Robust correlation tracking for UAV videos via feature fusion and saliency proposals," *Remote Sens.*, vol. 10, no. 10, 2018, Art. no. 1644.
[43] X. Xue, Y. Li, and Q. Shen, "Unmanned aerial vehicle object tracking by correlation filter with adaptive appearance model," *Sensors*, vol. 18, no. 9, 2018, Art. no. 2751.
[44] L. Zhang and P. N. Suganthan, "Robust visual tracking via co-trained kernelized correlation filters," *Pattern Recognit.*, vol. 69, pp. 82–93, 2017.
[45] L. Zhang, L. Song, B. Du, and Y. Zhang, "Nonlocal low-rank tensor completion for visual data," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 673–685, Feb. 2021.
[46] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 270–294, Jun. 2022.
[47] 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.
[48] W. Zuo, X. Wu, L. Lin, L. Zhang, and M.-H. Yang, "Learning support correlation filters for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 5, pp. 1158–1172, May 2019.

Yan Zhang was born in 1998. He received the B.E. degree in computer science and technology, in 2021, from the Nanjing University of Information Science and Technology, Nanjing, China, where he is currently working toward the M.E. degree in computer science and technology.

His current research interests include multimedia processing and object tracking.

Yuhui Zheng (Member, IEEE) was born in Shanxi, China, in 1982. He received the B.S. degree in chemistry and the Ph.D. degree in computer science from the Nanjing University of Science and Technology, Nanjing, China, in 2004 and 2009, respectively.

From 2014 to 2015, he was a Visiting Professor with the Digital Media Laboratory, School of Electronic and Electrical Engineering, Sungkyunkwan University, Seoul, Korea. He is currently a Full Professor with the School of Computer, Nanjing University of Information Science and Technology. His current research areas include image and video analysis, scene understanding, visual tracking, and pattern recognition.