Neural Networks 161 (2023) 39–54

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Enhanced robust spatial feature selection and correlation filter learning for UAV tracking

Jiajun Wen [a,b,c], Honglin Chu [a], Zhihui Lai [a,b,c,e,*], Tianyang Xu [d], Linlin Shen [a,b,c]

[a] College of Computer Science & Software Engineering, Shenzhen University, Shenzhen 518060, China
[b] Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China
[c] Guangdong Laboratory of Artificial-Intelligence and Cyber-Economics (SZ), Shenzhen University 518060, China
[d] School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi, 214122, China
[e] Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen 518129, China

ARTICLE INFO

Article history:
Received 9 March 2022
Received in revised form 26 December 2022
Accepted 4 January 2023
Available online 24 January 2023

Keywords:
UAV
Spatial feature selection
Correlation filter
Object tracking

ABSTRACT

Spatial boundary effect can significantly reduce the performance of a learned discriminative correlation filter (DCF) model. A commonly used method to relieve this effect is to extract appearance features from a wider region of a target. However, this way would introduce unexpected features from background pixels and noises, which will lead to a decrease of the filter's discrimination power. To address this shortcoming, this paper proposes an innovative method called enhanced robust spatial feature selection and correlation filter Learning (EFSCF), which performs jointly sparse feature learning to handle boundary effects effectively while suppressing the influence of background pixels and noises. Unlike the ℓ_2-norm-based tracking approaches that are prone to non-Gaussian noises, the proposed method imposes the $\ell_{2,1}$-norm on the loss term to enhance the robustness against the training outliers. To enhance the discrimination further, a jointly sparse feature selection scheme based on the $\ell_{2,1}$-norm is designed to regularize the filter in rows and columns simultaneously. To the best of the authors' knowledge, this has been the first work exploring the structural sparsity in rows and columns of a learned filter simultaneously. The proposed model can be efficiently solved by an alternating direction multiplier method. The proposed EFSCF is verified by experiments on four challenging unmanned aerial vehicle datasets under severe noise and appearance changes, and the results show that the proposed method can achieve better tracking performance than the state-of-the-art trackers.

1. Introduction

In recent years, visual tracking based on an unmanned aerial vehicle (UAV) platform has received increasing attention due to its practicability in a wide range of real-world applications, including pedestrian tracking (Smedt, Hulens, & Goedeme, 2015), traffic patrolling (Karaduman & Eren, 2019), aerial assisted refueling (Sun, Yin, Wang, & Xu, 2019), aerial cinematography and monitoring (Bonatti, Ho, Wang, Choudhury, & Scherer, 2019; Liu, Li, He, et al., 2020; Liu, Yuan, Fan, et al., 2022). However, the high flexibility of UAVs leads to rapid viewpoint changes and fast motion, which cause drastic appearance changes of a target, posing great challenges to the tracking process.

Even though UAV tracking has been designed for specific tracking purposes under flight status, it shares similar difficulties with the generic object tracking task (Huang, Fu, Li, Lin, & Lu, 2019; Huang, Zhao, & Huang, 2019), such as occlusion, out-of-plane rotation, background clutter, and viewpoint changes as demonstrated by Wu, Lim, and Yang (2013) and Wu, Lim, and Yang (2015). Recent studies have made significant achievements in the field of generic object tracking (Li, Ma, Wu, He, & Yang, 2019; Liang, Lu, He, & Zheng, 2019), but the tracking performance of these trackers degrades in UAV scenarios (Du et al., 2018, 2019; Mueller, Smith, & Ghanem, 2016). In principle, UAV tracking is challenging in three aspects. First, compared with conventional tracking scenarios, UAV tracking scenarios have a wider viewing angle, which increases the complexity of tracking algorithms. Second, the rapid movement of a UAV camera causes fast changes in viewpoint, which increases the difficulty of realizing robust feature extraction. Third, compared with generic object tracking, objects in aerial videos can be easier interfered with a dynamic background, resulting in a less discriminant appearance for tracking. Although a number of trackers have been designed for UAV tracking (Huang, & Fu et al., 2019; Huang, & Zhao et al., 2019; Li et al., 2020a, 2020b), improving the accuracy and robustness of a tracking model is still an unsolved problem. Therefore, it is

* Corresponding author at: College of Computer Science & Software Engineering, Shenzhen University, Shenzhen 518060, China.
E-mail address: lai_zhi_hui@163.com (Z. Lai).

https://doi.org/10.1016/j.neunet.2023.01.003
0893-6080/

necessary to explore an effective UAV tracking model to address the above-mentioned problems.

The existing research (Liu, Wang, Han, Fan, & Luo, 2017; Xu, Feng, Wu, & Kittler, 2019a, 2019b) has pointed out that sparsity is very effective for tracking tasks. Xu et al. (2019a, 2019b) introduced the ℓ_1-norm into the discriminative correlation filter (DCF) framework to achieve adaptive sparse feature selection, which could alleviate boundary effects. However, this method does not take the structural features into account. The reliability of a learned filter could be sharpened when there are dramatic appearance changes caused by occlusion, fast motion, or background clutters in challenging UAV tasks. To alleviate these problems, this study proposes designing an effective feature selection method for robust object tracking under a UAV shooting environment because feature selection is a key step to improving the tracking performance. In the study of Nie, Huang, Cai, and Ding (2010), an $\ell_{2,1}$-norm-based feature selection and classification framework was proposed to enhance the classification performance. In subsequent studies, this framework was further improved and applied to the fields of action recognition (Wen, Lai, Zhan, & Cui, 2016), domain adaptation learning (Tao, Zhou, & Zhu, 2018), and graph matching (Yu et al., 2021). Recent findings have indicated that the $\ell_{2,1}$-norm is robust and effective in feature selection. Moreover, related studies (Kong, Lai, Wang, & Liu, 2016; Yi, Lai, He, Cheung, & Liu, 2017) have shown that the $\ell_{2,1}$-norm can capture more structural sparsity compared with ℓ_1 and ℓ_2-norm types. The $\ell_{2,1}$-norm has been proven to be very effective in robust feature extraction and feature fusion (Zhang, Liu, Zhen, & Jing, 2020). Inspired by the recent studies and their achievements, this paper proposes an enhanced robust spatial feature selection and correlation filter learning (EFSCF), where an $\ell_{2,1}$-norm-based structural feature selection mode is performed on a filter in the row and column directions. Specifically, to enhance model robustness to outliers, the $\ell_{2,1}$-norm is imposed on the loss term of the tracking model to guarantee accurate candidate identification. In addition to select more discriminant features to enhance the robustness of filter against dramatic appearance changes, this study designs a strong and effective structural feature selection scheme by using the row- and column-based $\ell_{2,1}$-norm to regularize the filter and extract crucial features in both directions simultaneously.

The idea of the proposed method is illustrated in Fig. 1. As shown in Fig. 1, in the training phase, spatial regularization is performed on the filter in rows and columns using the $\ell_{2,1}$-norm for jointly sparse feature selection to enhance the structural sparsity of the filter and extract more discriminative features with high resistance to noise. To maintain high robustness of the proposed model, the temporal term is regularized to ensure the smoothness of learned filters in successive frames under the $\ell_{2,1}$-norm-based jointly sparse feature learning framework. The robustness of the proposed model against noise and appearance changes is validated in the detection phase. As shown in Fig. 1, on the 170th frame of the video S1101 in the UAVDT dataset, the proposed EFSCF tracker can accurately track the target even under rapid viewpoint changes and background clutter circumstances, while both the STRCF and the LADCF lose the target. In recent years, several discriminative correlation filtering tracking methods for UAV scenes have been developed. The ARCF (Huang, & Fu et al., 2019; Huang, & Zhao et al., 2019) suppressed abnormities by enforcing restrictions on the rate of alteration in response maps. The AutoTrack (Li et al., 2020a, 2020b) adaptively learned the filter based on the response maps. In addition, the MSCF (Zheng, Fu, Ye, et al., 2021) was developed to handle tracking mutations by employing the adaptivity of an adaptive hybrid label. Moreover, the ReCF (Lin, Fu, He, et al., 2021) used an inferred response regularization to assist a filter in suppressing turbulent response fluctuations. Unlike the aforementioned studies that indirectly learn more robust filters through response maps, this study directly constrains a filter in different directions through the $\ell_{2,1}$-norm. The row- and column-based $\ell_{2,1}$-norm constraints can learn structural information from complex backgrounds, which is beneficial to improving the UAV tracking accuracy. The proposed method enables features of a learned filter to be sparsely distributed in the spatial domain. These features can be regarded as energy of the target. To this end, the proposed method can accurately focus on the target to alleviate the boundary effects. The proposed method is compared with the state-of-the-art methods, and comprehensive evaluations are conducted on four public UAV tracking datasets, namely, the UAV123 (Mueller et al., 2016), the UAV20L (Mueller et al., 2016), the UAVDT (Du et al., 2018), VisDrone2019 (Du et al., 2019). In addition, generic tracking datasets OTB2013 (Wu et al., 2013) and OTB2015 (Wu et al., 2015) datasets are also used to verify the effectiveness of the proposed method.

The main contributions of our work can be summarized as follows:

- The $\ell_{2,1}$-norm is imposed on the loss term, which makes the model robust to outliers and improves model accuracy and robustness.
- An adaptive feature selection method in the spatial domain is proposed. Particularly, the row- and column-based $\ell_{2,1}$-norm types are combined for jointly sparse feature selection to capture crucial structural features of the target. In this way, the model can learn more discriminant features, which improves its robustness against background interference and noises in a wider search area.
- A jointly sparse feature selection-based tracking framework is constructed to implement the proposed EFSCF tracker, whose effectiveness is validated by extensive experiments on four UAV datasets. The experimental results show the proposed tracker can run at approximately 18 frames per second (FPS) using only hand-crafted features on the CPU and can outperform several state-of-the-art trackers.

The rest of this paper is organized as follows. Section 2 reviews the related theories and technologies Section 3 introduces the proposed model and provides details of the optimization framework. Section 4 presents the results of qualitative and quantitative evaluations and an ablation study. Finally, Section 5 summarizes this work.

2. Related work

The existing tracking methods can be roughly divided into two categories, generative methods and discriminative methods. The generative methods commonly use a model constructed based on similarity measurement methods, such as sparse representation-based tracking method (Bao, Wu, Ling, & Ji, 2012; He, Yi, Cheung, You, & Tang, 2017), PCA-based tracking method (Ross, Lim, Lin, & Yang, 2008), and dictionary learning-based tracking method (Wang, Wang, & Yeung, 2013). This type of model learns the target appearance or motion patterns to obtain the crucial features for data association, and finds the best candidate for the next position. However, generative methods focus only on the target itself while ignoring the background information. Unlike generative methods, discriminative methods consider the tracking task a classification or regression problem (Hare, Saffari, & Torr, 2011; Kalal, Mikolajczyk, & Matas, 2011); particularly, positive and negative samples are extracted from the foreground and background, respectively, to train a classifier to identify the target region from the coming frame. Generally, discriminative methods have better performance than generative methods. Recent studies on discriminative methods have mainly focused on DCF learning due

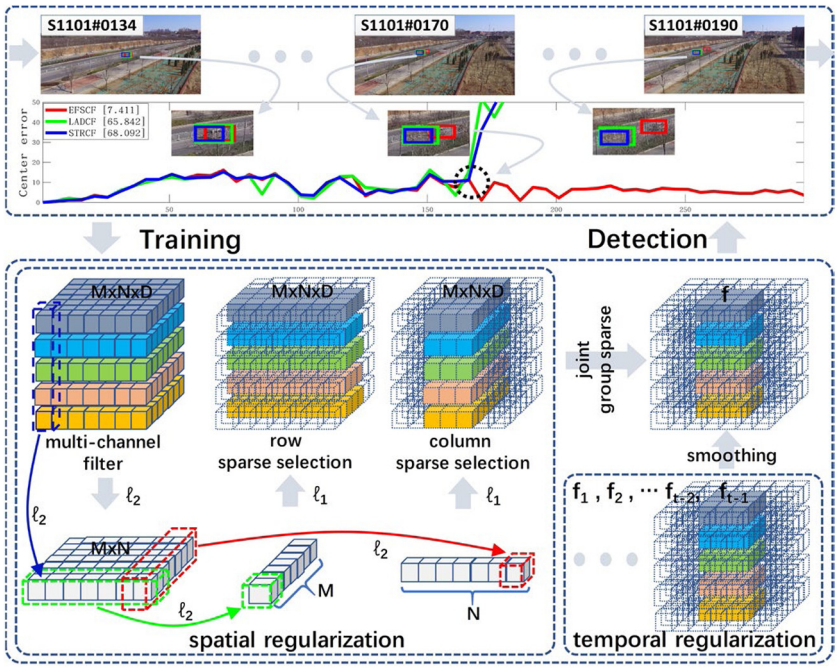


Fig. 1. Illustration of the operational principle of the proposed EFSCF method. In the training phase, spatial regularization is performed on the filter in rows and columns using the $\ell_{2,1}$-norm for realizing jointly sparse feature selection. The temporal term is regularized to ensure the smoothness and robustness of the learned filters in successive frames. In the detection phase, the robustness of the proposed model in resisting noise and appearance changes is validated. The presented case shows that the EFSCF tracker can accurately track the target even under rapid viewpoint changes and background clutter circumstances, while both the STRCF and the LADCF lose the target.

to its effectiveness and extensibility. A DCF is a basic prototype of the CF-based tracking methods, and it provides a crucial feature map for a tracking task (Li et al., 2020a, 2020b). In addition, a DCF used cyclic shifting to construct training samples efficiently and reduce the computation cost by converting the process of redundant complex matrix multiplication (spatial domain) into a series of element-wise operations (frequency domain). Solid foundation work about DCFs can be found in the studies of Bolme, Beveridge, Draper, and Lui (2010) and Henriques, Caseiro, Martins, and Batista (2012), which was the motivation for developing numerous extensive DCF methods based on multi-channel learning (Danelljan, Häger, Khan, & Felsberg, 2014; Danelljan, Khan, Felsberg, & Weijer, 2014; Henriques, Caseiro, Martin, & Batista, 2015), multi-kernel learning (Tang & Feng, 2015), multi-view learning (Li, Liu, He, et al., 2016), multi-feature fusion (Xu, Feng, Wu, & Kittler, 2020; Zhu, Wu, Xu, Feng, & Kittler, 2021), multi-frame learning (Sui, Wang, & Zhang, 2020), scale adaptation (Danelljan, & Häger et al., 2014; Danelljan, & Khan et al., 2014; Li & Zhu, 2015), spatial–temporal context (Elayaperumal & Joo, 2021; Zhang, Li, Song, Liu, & Lian, 2018), spatial regularization (Dai, Wang, Lu, Sun, & Li, 2019; Danelljan, Hager, Khan, & Felsberg, 2015; Feng, Han, Guo, Zhu, & Wang, 2019; Li, Tian, Zuo, Zhang, & Yang, 2018; Xu et al., 2019a, 2019b), channel regularization (Liang, Liu, Yan, Zhang, & Wang, 2021; Xu et al., 2019a, 2019b), and continuous convolution operators (Danelljan, Bhat, Shahbaz Khan, & Felsberg, 2017; Danelljan, Hager, Khan, & Felsberg, 2017; Danelljan, Robinson, Khan, & Felsberg, 2016). The above-mentioned methods have shown advanced performances on the public tracking benchmarks (Kristan et al., 2015; Wu et al., 2013, 2015).

The tracking methods under the DCF framework face two main challenges: scale adaptation and boundary effect. To reduce the influence of scale variation on the tracking performance, Li and Zhu (2015) proposed a multi-scale traversal method using the pyramid strategy to obtain multi-scale response maps and then determined the optimal scale based on the maximum point among the response maps. The DSST (Danelljan, & Häger et al., 2014; Danelljan, & Khan et al., 2014) considers the tracking task two independent problems, namely scale detection, and position detection and accordingly designs a parallel training architecture for scale and position filter learning during the tracking process. To optimize the tracking framework, dimensionality reduction was performed to obtain discriminant features in a low-dimensional space to improve the feasibility of real-time tracking (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017). This speed-up strategy has been applied to different tracking methods (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017; Li et al., 2018; Xu et al., 2019a, 2019b). The boundary effect is caused by the cyclic-shift operation that generates unreasonable samples on the boundary, and this effect can significantly degrade the discriminant power of a tracking model. Recently, fixed-weight matrix (Danelljan et al., 2015; Li et al., 2018), adaptive-weight matrix (Dai et al., 2019; Feng et al., 2019), fixed-crop matrix (Galoogahi, Fagg, & Lucey, 2017; Galoogahi, Sim, & Lucey, 2015), and adaptive sparse feature selection (Xu et al., 2019a, 2019b) have been introduced into a DCF framework to mitigate the above-mentioned problem. The approaches based on a fixed weight-matrix (Danelljan et al., 2015; Li et al., 2018) can effectively suppress the boundary region of a filter but cannot adapt well to the severe appearance changes of a target. To increase model adaptation ability to appearance changes, recent studies based on an adaptive weight (Dai et al., 2019; Feng et al., 2019) have proposed a flexible learning-based scheme, which can dynamically adjust the penalty weight according to the variation in the target thus further enhancing the robustness of a tracking model. Unlike previous methods, the cropping matrix method (Galoogahi et al., 2017, 2015) can solve the boundary effect from the perspective of feature selection by directly selecting the target features from the background using a fixed binary mask, and thus ensuring that only genuine samples are used for training. However, the performance of this tacker largely depends on mask accuracy. The methods based on the sparse feature selection (Xu

et al., 2019a, 2019b) can address the problem of the high noise level by introducing sparse constraints to select discriminant features adaptively; namely, sparse constraints are more effective than features derived from the target cropped from the entire search area.

In addition to the above-mentioned tracking methods, deep learning (Liu et al., 2020, 2022) has played an important role in tracking applications. The deep trackers that have connection with the proposed framework will be reviewed here. Recent studies have shown that correlation filtering learning is able to enhance the representation ability of the deep tracking framework. The popular Siamese network explores cross-correlation of the inputs to explore the similarity feature of the target (Bertinetto, Valmadre, Henriques, Vedaldi, & Torr, 2016). To facilitate visual tracking in an unsupervised way, correlation filter was combined with Siamese network for effective forward and backward predictions (Wang et al., 2019). Yuan, Chang, Huang, et al. (2021) designed a self-supervised tracking framework for feature learning based on correlation filter and Siamese network. In this study, to fully validate the effectiveness of the proposed method, Section 4 will compare the proposed method with the popular deep trackers.

In the following, theories related to the proposed model are briefly introduced. Without loss of generality, the expressions of variables presented in this paper are unified as follows: scalars are denoted by italic letters, matrices are denoted by lowercase bold formal letters, while vectors are not bolded. For instance, for a three-dimensional matrix $\mathbf{f}$, its $i$throw, jth column and dth channel are denoted by $\mathbf{f}_{i:}$, $\mathbf{f}_{:j}$, and $\mathbf{f}^d$, respectively.

2.1. Spatial regularization

Danelljan et al. (2015) proposed the spatially regularized DCF (SRDCF) for the first time to handle the boundary effect. A fixed spatial penalty matrix was used to assign low weights to the pixels that were far away from the filter center. However, a fixed penalty matrix could impede the tracking performance due to its inability and inadaptability to significant variations in the target. To capture more effective features, time-domain smoothing constraints were introduced to learn the time-domain changes of the spatial weight matrix, which increases the adaptation of the tracking model (Dai et al., 2019). By performing salient analysis on the spatial regularization term, the spatial weight matrix could be learnt dynamically for effective object tracking (Feng et al., 2019).

2.2. Temporal regularization

In order to avoid rapid degradation of the model over time, SRDCF (Danelljan et al., 2015) makes use of multiple historical samples to learn the current filter. However, this strategy destroys the structure of the circular matrix to a certain extent and adds high computation burden to solving the model. Unlike the SRDCF that employs joint feature learning using multiple frames, Li et al. (2018) added temporal regularization to the DCF framework to develop spatial–temporal regularized correlation filter (STRCF), which guarantees the smoothness of the filter in time domain and improves robustness of the model against occlusion and deformation. The objective function of STRCF is given by

$$\arg\min_{\mathbf{f}} \frac{1}{2}\left\|\sum_{d=1}^{D}\mathbf{f}^d * \mathbf{x}^d - \mathbf{y}\right\|_F^2 + \frac{1}{2}\sum_{d=1}^{D}\left\|\mathbf{w}\odot\mathbf{f}^d\right\|_F^2 + \frac{\mu}{2}\left\|\mathbf{f}-\mathbf{f}'\right\|_F^2, \tag{1}$$

where $*$ represents convolution operator, $\odot$ denotes the Hadamard product, and $\mathbf{x}^d \in \mathbb{R}^{M\times N}$, $\mathbf{f}^d \in \mathbb{R}^{M\times N}$ represent the dth channel of a sample $\mathbf{x}$ and a filter $\mathbf{f}$, respectively; $\mathbf{y} \in \mathbb{R}^{M\times N}$ represents the ideal Gaussian response map; the second term denotes spatial regularization, where $\mathbf{w}$ represents a penalty matrix with a size of $M \times N$; the third term indicates temporal regularization, where $\mathbf{f}'$ represents a filter learned in the previous frame, and μ is a penalty scalar. Temporal regularization maintains the continuity of a filter in the time domain, which is a reasonable approximation for joint multi-frame learning. Lastly, Eq. (1) can be solved efficiently by the ADMM (Boyd, Parikh, Chu, Peleato, & Eckstein, 2010).

2.3. Spatial feature selection

To select features of the search area, a fixed cropping matrix was introduced to obtain target region from the background to generate negative samples for filter training (Galoogahi et al., 2017). Following this technical roadmap, Xu et al. (2019a, 2019b) proposed learning adaptive discriminant correlation filter (LADCF) by introducing the ℓ_1-norm to select spatial features of a filter adaptively, thus overcoming the inadaptability of the learned filter. This model achieved promising results in the visual object tracking competition VOT2018 (Kristan et al., 2019). The objective function of the LADCF can be expressed as follows:

$$\arg\min_{\mathbf{f}} \sum_{d=1}^{D}\left\|\mathbf{f}^d * \mathbf{x}^d - \mathbf{y}\right\|_F^2 + \lambda_1 \sum_{i=1}^{M}\sum_{j=1}^{N}\sqrt{\sum_{d=1}^{D}(\mathbf{f}_{ij}^d)^2} + \lambda_2 \sum_{d=1}^{D}\left\|\mathbf{f}^d - (\mathbf{f}')^d\right\|_F^2, \tag{2}$$

where the second term represents the spatial feature selection term, and λ_1 and λ_2 denote spatial and temporal penalty factors, respectively.

By performing the ℓ_1-norm on a filter in the spatial domain, most of the extracted features in the filter boundary region approach zero, indicating that the noises and redundancy are significantly reduced. On the contrary, the greater element values on the filter, the more discriminative features can be retained to enhance the robustness of the tracker. Since the LADCF0 (Xu et al., 2019a, 2019b) restricts only the overall sparsity, the selected features do not fully consider the structure of a target, which can easily result in an unstable feature learning process when the target changes significantly. Aiming at solving this problem, this study proposes a structural sparsity regularization method to select more discriminant features for robust object tracking, whose feasibility is verified on several challenging UAV sequences.

2.4. Relation to the related studies

Recent studies on subspace-based correlation filter methods have made great progress in the field of object tracking (Zhu et al. 2021; Dong, Yang, & Pei, 2016; Ji & Wang, 2018; Sui, Wang, & Zhang, 2017; Sui, Zhang, & Wang, 2016; Xu et al., 2019a, 2019b, 2020). Zhu et al. (2021) proposed a collaborative representation strategy to learn discriminative filters between successive frames. Xu et al. (2020) introduced an ℓ_1-norm constraint to learn a DCF in a low-rank space. To overcome the problem of channel redundancy and noise in feature representation, Xu et al. (2019a, 2019b) designed a channel learning framework by imposing an ℓ_1-norm constraint on channel coefficients for a sparse solution. To relieve the noise influence, the $\ell_{2,1}$ -norm was introduced to the error term to construct a robust tracker (Sui et al., 2016). In the study of Sui et al. (2017), an elastic net constraint was imposed on a filter to eliminate the distractive features adaptively. To improve the tracking performance further, the ℓ_1-norm was

introduced to the correlation filter framework to obtain a sparse correlation filter (Dong et al., 2016).

The above-mentioned methods, as well as the proposed method, achieve certain improvements under a correlation filter framework. However, it should be noted that the innovation of this study differs from those of the previous work. Namely, even though structural features are very important for DCF tracking, fewer studies focused on structural feature learning in DCFs. To make full use of structural features and address the spatial boundary effect, this study designs a robust filter learning method based on structural regular constraints. To sharpen the focus of filters on relevant features, this work introduces the $\ell_{2,1}$ regularization term into the basic DCF framework to achieve the row and column feature selection. In addition, unlike the proposed method, the previously proposed methods cannot jointly learn crucial features in row and column directions simultaneously. The proposed method is verified by comparative experiments on four UAV datasets and an OTB dataset to validate that it can improve the discriminative power and robustness of a filter.

3. Proposed EFSCF method

This section introduces the proposed EFSCF model, which performs jointly sparse feature selection using row- and column-based $\ell_{2,1}$-norm regularizations in a UAV shooting environment.

3.1. Objective function

In the work of Bolme et al. (2010), the grayscale feature with a single channel representation was used for the first time in the DCF training process, however the learned filter under single channel lacked strong discriminant power. An intuitive and effective way to enhance the robustness of the filter is to extend single channel representation to multi-channel representation, Henriques et al. (2015) and Danelljan, and Häger et al. (2014), Danelljan, and Khan et al. (2014) introduced multi-channel HOG (Dalal et al., 2010) and CN (Weijer, Schmid, Verbeek, & Larlus, 2009) features respectively to train the filter, which enhances the robustness of the filter further by learning comprehensive and multi-channel representation. Therefore, the proposed model of this work will be built under single channel representation first, and then we extend it to a multi-channel version for comprehensive feature learning.

In order to improve the model's robustness against outliers e.g., background pixels and noises, we propose to impose the $\ell_{2,1}$-norm on the loss term in pursuit of minimal residuals for Gaussian-shaped template regression. In addition, to extract effective and discriminant features for enhancing the tracking performance, the $\ell_{2,1}$-norm-based jointly sparse regularization is proposed to perform on a filter in row and column directions for robust feature selection. The formulation of our objective function in single channel version is defined as follows:

$$\arg\min_{\mathbf{f}} \|\mathbf{f}*\mathbf{x}-\mathbf{y}\|_{2,1}+\lambda_1\|\mathbf{f}\|_{2,1}+\lambda_2\left\|\mathbf{f}^{\mathrm{T}}\right\|_{2,1}+\lambda_3\left\|\mathbf{f}-\mathbf{f}'\right\|_F^2. \quad (3)$$

where $\mathbf{x}$ represents the input features; $\mathbf{x}$, $\mathbf{y}$, $\mathbf{f}$, and $\mathbf{f}'$ have the same dimension $M\times N$; $\lambda_i>0\,(i=1,2,3)$. In Eq. (3), the first term denotes the $\ell_{2,1}$-norm-based loss term, which is introduced to fit the Gaussian-shaped $\mathbf{y}$ by convolving a training sample $\mathbf{x}$ with a filter $\mathbf{f}$; the second and third terms represent row sparse constraint and column sparse constraint, respectively, where a superscript T stands for a transposed operator; the fourth term is a temporal smoothing term, which aims to ensure the stability of learned filters in successive frames under the $\ell_{2,1}$-norm based jointly sparse feature learning framework; $\mathbf{f}'$ is a filter learned in the previous frame. To enhance the performance of the learned filter, we generalize our objective function from single channel to multi-channel formulation, namely

$$\arg\min_{\mathbf{f}} \sum_{d=1}^{D}\left\|\mathbf{f}^d*\mathbf{x}^d-\mathbf{y}\right\|_{2,1}+\lambda_1\sum_{i=1}^{M}\|\mathbf{f}_{i:}\|_F + \lambda_2\sum_{j}^{N}\left\|\mathbf{f}_{:j}\right\|_F+\lambda_3\sum_{i=1}^{D}\left\|\mathbf{f}^d-(\mathbf{f}')^d\right\|_F^2, \quad (4)$$

where $\mathbf{x}=[\mathbf{x}^1,\mathbf{x}^2,\ldots,\mathbf{x}^D]\in\mathbb{R}^{M\times N\times D}$, $\mathbf{f}=[\mathbf{f}^1,\mathbf{f}^2,\ldots,\mathbf{f}^D]\in\mathbb{R}^{M\times N\times D}$, and $\mathbf{y}\in\mathbb{R}^{M\times N}$.

The second and third terms in Eq. (4) can be respectively expressed by:

$$\sum_{i=1}^{M}\|\mathbf{f}_{i:}\|_F=\sum_{i=1}^{M}\sqrt{\sum_{j=1}^{N}\sum_{d=1}^{D}(\mathbf{f}_{ij}^d)^2}, \quad (5)$$

and

$$\sum_{j}^{N}\left\|\mathbf{f}_{:j}\right\|_F=\sum_{j=1}^{N}\sqrt{\sum_{i=1}^{M}\sum_{d=1}^{D}(\mathbf{f}_{ij}^d)^2}. \quad (6)$$

Eqs. (5) and (6) correspond to the row-based $\ell_{2,1}$-norm and column-based $\ell_{2,1}$-norm respectively. For Eq. (5), we first calculate the ℓ_2-norm of each row across all channels, respectively, and then calculate the ℓ_1-norm with respect to all rows. For Eq. (6), we first calculate the ℓ_2-norm of each column across all channels, respectively, and then calculate the ℓ_1-norm with respect to all columns. In principle, row sparse constraint and column sparse constraint enable structural sparsity, which improves the filter's discriminant power and interpretability of the learned model. However, current studies on correlation filtering learning cannot achieve feature learning along row and column simultaneously.

3.2. Optimization principle

To solve the proposed objective function, a feasible solution based on the ADMM (Boyd et al., 2010) is proposed to obtain an optimal solution of Eq. (4). Assume that $\mathbf{f}=\mathbf{h}$ serves as an auxiliary constraint of the proposed model. Then, the Lagrange function corresponding to Eq. (4) can be expressed as follows:

$$L(\mathbf{f},\mathbf{h},\boldsymbol{\eta},\mu)=\sum_{d=1}^{D}\left\|\mathbf{f}^d*\mathbf{x}^d-\mathbf{y}\right\|_{2,1}+\lambda_1\sum_{i=1}^{M}\|\mathbf{h}_{i:}\|_F+\lambda_2\sum_{j}^{N}\left\|\mathbf{h}_{:j}\right\|_F + \lambda_3\sum_{d=1}^{D}\left\|\mathbf{f}^d-(\mathbf{f}')^d\right\|_F^2+\frac{\mu}{2}\sum_{d=1}^{D}\left\|\mathbf{f}^d-\mathbf{h}^d+\frac{\boldsymbol{\eta}^d}{\mu}\right\|_F^2, \quad (7)$$

where $\boldsymbol{\eta}=[\boldsymbol{\eta}^1,\boldsymbol{\eta}^2\ldots,\boldsymbol{\eta}^D]$ are the Lagrange multipliers, and $\mu>0$ is a scale factor that controls the convergence. Then, employing the ADMM (Boyd et al., 2010), Eq. (7) can be split into three sub-problems for iterative optimization.

$$\mathbf{f}=\arg\min_{\mathbf{f}}\sum_{d=1}^{D}\left\|\mathbf{f}^d*\mathbf{x}^d-\mathbf{y}\right\|_{2,1}+\lambda_3\sum_{d=1}^{D}\left\|\mathbf{f}^d-(\mathbf{f}')^d\right\|_F^2 + \frac{\mu}{2}\sum_{d=1}^{D}\left\|\mathbf{f}^d-\mathbf{h}^d+\frac{\boldsymbol{\eta}^d}{\mu}\right\|_F^2, \quad (8)$$

$$\mathbf{h}=\arg\min_{\mathbf{h}}\lambda_1\sum_{i=1}^{M}\|\mathbf{h}_{i:}\|_F+\lambda_2\sum_{j}^{N}\left\|\mathbf{h}_{:j}\right\|_F \quad +\frac{\mu}{2}\sum_{d=1}^{D}\left\|\mathbf{f}^d-\mathbf{h}^d+\frac{\boldsymbol{\eta}^d}{\mu}\right\|_F^2, \quad (9)$$

$$\boldsymbol{\eta}=\boldsymbol{\eta}+\mu(\mathbf{f}-\mathbf{h}). \quad (10)$$

3.2.1. Sub-problem $\mathbf{f}$

Given $\mathbf{h}$ and $\boldsymbol{\eta}$, the process of solving $\mathbf{f}$ can be converted to the frequency-based expression for optimization by exploiting the properties of circular matrix (Henriques et al., 2012) and Parseval's theorem (Brigham & Morrow, 1967). Therefore Eq. (8) can be rewritten as follows:

$$\hat{\mathbf{f}} = \arg\min_{\hat{\mathbf{f}}} \sum_{d=1}^{D} \left\| (\hat{\mathbf{f}}^*)^d \odot \hat{\mathbf{x}}^d - \hat{\mathbf{y}} \right\|_{2,1} + \lambda_3 \sum_{d=1}^{D} \left\| \hat{\mathbf{f}}^d - (\hat{\mathbf{f}}')^d \right\|_F^2 + \frac{\mu}{2} \sum_{d=1}^{D} \left\| \hat{\mathbf{f}}^d - \hat{\mathbf{h}}^d + \frac{\hat{\boldsymbol{\eta}}^d}{\mu} \right\|_F^2, \tag{11}$$

where $\hat{\mathbf{f}}$ represents the discrete Fourier transform (DFT) of $\mathbf{f}$ and $*$ denotes a conjugate operator. For each channel in $\hat{\mathbf{f}}$, Eq. (11) can be decomposed into the following form:

$$\hat{\mathbf{f}}^d = \arg\min_{\hat{\mathbf{f}}^d} \left\| (\hat{\mathbf{f}}^*)^d \odot \hat{\mathbf{x}}^d - \hat{\mathbf{y}} \right\|_{2,1} + \lambda_3 \left\| \hat{\mathbf{f}}^d - (\hat{\mathbf{f}}')^d \right\|_F^2 + \frac{\mu}{2} \left\| \hat{\mathbf{f}}^d - \hat{\mathbf{h}}^d + \frac{\hat{\boldsymbol{\eta}}^d}{\mu} \right\|_F^2. \tag{12}$$

Further, Eq. (12) is a convex function, and its closed-form solution can be obtained by

$$\hat{\mathbf{f}}^d = \frac{(\hat{\mathbf{k}}^d \odot \hat{\mathbf{x}}^d) \odot (\hat{\mathbf{k}}^d \odot \hat{\mathbf{y}}^*) + \lambda_3 (\hat{\mathbf{f}}')^d + \frac{\mu}{2} \hat{\mathbf{h}}^d - \frac{\hat{\boldsymbol{\eta}}^d}{2}}{(\hat{\mathbf{k}}^d \odot \hat{\mathbf{x}}^d) \odot \left[\hat{\mathbf{k}}^d \odot (\hat{\mathbf{x}}^d)^* \right] + \lambda_3 + \frac{\mu}{2}}. \tag{13}$$

By using the Sherman Morrison formula (Sherman et al., 2015), $\hat{\mathbf{f}}$ can be efficiently calculated, and more optimization details can be found in STRCF (Li et al., 2018). It should be noted that, in Eq. (13), each element of $\hat{\mathbf{k}}^d$ can be expressed as

$$(\hat{\mathbf{k}}^d)_{ij} = \frac{1}{2\sqrt{\left\| \left[(\hat{\mathbf{f}}^*)^d \odot \hat{\mathbf{x}}^d - \hat{\mathbf{y}} \right]_{ij} \right\|_2}}, \tag{14}$$

After solving $\hat{\mathbf{f}}$, it is transformed to the time domain to obtain an update of $\mathbf{f}$.

3.2.2. Sub-problem $\mathbf{h}$

For the second sub-problem, Eq. (9) is non-convex. In order to further simplify the calculation, Eq. (9) can be divided into two independent problems (Xu, & (Feng et al., 2019)) as follows:

$$\mathbf{h} = \arg\min_{\mathbf{h}} \lambda_1 \sum_{i=1}^{M} \|\mathbf{h}_{i:}\|_F + \frac{\mu}{2} \sum_{i=1}^{M} \left\| \mathbf{f}_{i:} - \mathbf{h}_{i:} + \frac{\boldsymbol{\eta}_{i:}}{\mu} \right\|_F^2, \tag{15}$$

and

$$\mathbf{h} = \arg\min_{\mathbf{h}} \lambda_2 \sum_{j=1}^{N} \|\mathbf{h}_{:j}\|_F + \frac{\mu}{2} \sum_{j=1}^{N} \left\| \mathbf{f}_{:j} - \mathbf{h}_{:j} + \frac{\boldsymbol{\eta}_{:j}}{\mu} \right\|_F^2. \tag{16}$$

For Eq. (15), it can be calculated for each row separately to accelerate the optimization process for obtaining

$$\mathbf{h}_{i:} = \arg\min_{\mathbf{h}_{i:}} \lambda_1 \|\mathbf{h}_{i:}\|_F + \frac{\mu}{2} \left\| \mathbf{f}_{i:} - \mathbf{h}_{i:} + \frac{\boldsymbol{\eta}_{i:}}{\mu} \right\|_F^2. \tag{17}$$

Similar to the work of Xu et al. (2019a, 2019b), we take the derivative of Eq. (17) with regard to $\mathbf{h}_{i:}$, the closed-form solution to Eq. (17) can be obtained as

$$\mathbf{h}_{i:} = \max\left(0, 1 - \frac{\lambda_1}{\mu \left\| \mathbf{f}_{i:} + \frac{\boldsymbol{\eta}_{i:}}{\mu} \right\|_F}\right) \left(\mathbf{f}_{i:} + \frac{\boldsymbol{\eta}_{i:}}{\mu}\right), \tag{18}$$

where the soft-thresholding operator realizes row sparse feature selection of a filter.

By analogy, the closed-form solution to Eq. (16) can be obtained by

$$\mathbf{h}_{:j} = \max\left(0, 1 - \frac{\lambda_2}{\mu \left\| \mathbf{f}_{:j} + \frac{\boldsymbol{\eta}_{:j}}{\mu} \right\|_F}\right) \left(\mathbf{f}_{:j} + \frac{\boldsymbol{\eta}_{:j}}{\mu}\right). \tag{19}$$

Eq. (18) indicates an impact of the weight on the filter in row direction. Similarly, Eq. (19) indicates an impact of the weight on the filter in column direction. By considering these advantages, Eqs. (18) and (19) can be integrated into a comprehensive representation as follows:

$$\mathbf{h}_{ij} = \max\left(0, 1 - \frac{\lambda_1}{\mu \left\| \mathbf{f}_{i:} + \frac{\boldsymbol{\eta}_{i:}}{\mu} \right\|_F} - \frac{\lambda_2}{\mu \left\| \mathbf{f}_{:j} + \frac{\boldsymbol{\eta}_{:j}}{\mu} \right\|_F}\right) \left(\mathbf{f}_{ij} + \frac{\boldsymbol{\eta}_{ij}}{\mu}\right), \tag{20}$$

where $\mathbf{h}_{ij}$ corresponds with all elements across D channels at $i$throw and jth column of $\mathbf{h}$ in the spatial domain.

3.2.3. Sub-problem $\boldsymbol{\eta}$

Given $\mathbf{f}$, $\mathbf{h}$, and μ, $\boldsymbol{\eta}$ can be updated by Eq. (10), where μ is a variable factor that controls the step size of $\boldsymbol{\eta}$, and it is updated after each iteration by

$$\mu_{\text{new}} = \min(\mu_{\max}, \rho\mu), \tag{21}$$

where ρ is a fixed scale factor and $\mu_{\max}$ represents the maximum value of μ.

3.3. Complexity analysis

For the sub-problem $\mathbf{f}$, including the DFT and inverse DFT, the computational complexity of $\mathbf{f}$ is $\mathcal{O}(DMN\log(MN))$, and the complexity for calculating $\hat{\mathbf{k}}$ is $\mathcal{O}(DMN)$; for the second sub-problem $\mathbf{h}$, the computational complexity is $\mathcal{O}(DMN)$; lastly, for the third sub-problem $\boldsymbol{\eta}$, the complexity for is $\mathcal{O}(1)$. Therefore, the overall computational complexity of the proposed method is $\mathcal{O}(QDMN\log(MN))$, where Q represents the maximum number of iterations.

3.4. Tracking framework

The proposed EFSCF tracker includes three steps, namely model training, position and scale detection, and model update, which are explained in the following.

3.4.1. Model training

In the first frame, the filter is initialized in the same way as an STRCF (Li et al., 2018), which adopts a fixed penalty matrix to mitigate the boundary effects. In subsequent frames, the EFSCF model is trained at each frame. Our training process is summarized in Algorithm 1.

3.4.2. Position and scale detection

Similar to the fDSST (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017), hand-crafted features $\mathbf{x}(s)$ with multiple scales a^s are extracted from the search region, where a denotes a scale factor, $s \in \left\{ \left\lfloor \frac{1-N}{2} \right\rfloor, \ldots, 1 \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor \right\}$, and N represents the number of scales. Given $\mathbf{f}$, the multi-channel response maps can be obtained in the frequency domain as shown in Eq. (22).

$$\hat{\mathbf{r}}(s) = \hat{\mathbf{x}}(s) \odot \hat{\mathbf{f}}* \tag{22}$$

Finally, the position and scale can be obtained according to the maximum value in the response maps. For more details, please refer to reference (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017).

3.4.3. Model update

During the tracking process, the appearance of a target keeps changing, showing unpredictable movements and non-rigid transformations. Therefore, if a model depends on only the training results of the initial frame, it will be prone to drift, so the historical filters need to be fused. Most existing CF-based trackers adopt a linear interpolation update strategy by setting a fixed learning rate, which can be expressed as:

$$\mathbf{f}_t = (1-\alpha)\mathbf{f}_{t-1} + \alpha\mathbf{f}, \tag{23}$$

Algorithm 1 EFSCF training process

Input: At t-th frame (t>2), the multi-channel hand-crafted features $\mathbf{x}$ based on current bounding box and the filter $\mathbf{f}^{'}$ based on (t-1)-th frame;
1: **Initialize**: Set $\mathbf{k}$ to identity matrix, μ =1;
2: **for** i =1, 2 **do:**
3: **for** j =1, 2 **do:**
4: Solve sub-problem $\mathbf{f}$ using Eq. (13);
5: Update diagonal matrix $\mathbf{k}$ using Eq. (14);
6: Solve sub-problem $\mathbf{h}$ for row feature selection using Eq. (18);
7: Solve sub-problem $\mathbf{h}$ for column feature selection using Eq. (19);
8: Update $\boldsymbol{\eta}$ using Eq. (10) and update μ using Eq. (21);
Output: The learned filter $\mathbf{f}$ for the current frame.

where α represents a learning rate that controls the proportion of model update. The update strategy with a fixed learning rate cannot adapt to large appearance changes of a target. Therefore, by being motivated by Li et al. (2018) and Xu et al. (2019a, 2019b), this study introduces temporal regularization, integrating historical filters into the current filter training process to prevent model corruption. The last term of the objective function (3) shows that the learned filter approximates the filter learned in the previous frame. In this way, the filter learned at the current frame can be used directly for the detection in the next frame without performing the interpolation update. The corresponding mathematical formula is as follows:

$$\mathbf{f}_t = \mathbf{f}. \tag{24}$$

3.5. Relation to STRCF and LADCF

In this section, the relationship between the proposed method and other closely related methods, including STRCF (Li et al., 2018) and LADCF (Xu et al., 2019a, 2019b), is presented.

3.5.1. Model relationship

The proposed method is based on the DCF framework, and the $\ell_{2,1}$-norm is applied to the loss term and spatial regularization term to enhance the robustness of the model. Meanwhile, similar to the STRCF and LADCF, temporal regularization is introduced to ensure the smoothness of a learned filter during tracking. For spatial regularization, it is necessary to emphasize the difference between the proposed method and LADCF. Considering that each pixel represents a feature in the spatial domain of a filter, to fully explore crucial structural features of a target in different directions (e.g., rows and columns), two $\ell_{2,1}$-norms are imposed on the proposed model to achieve row and column sparsity of a learned filter. As shown in Fig. 2, when the $\ell_{2,1}$-norm is imposed on a filter, row and column sparsity is generated, which differs from the random spatial sparsity by ℓ_1-norm. Actually, the LADCF highlights the overall sparsity, whereas the EFSCF underlines the structural sparsity. The relationship between the two methods and the merits of the proposed method is further discussed in the ablation study in Section 4.4.

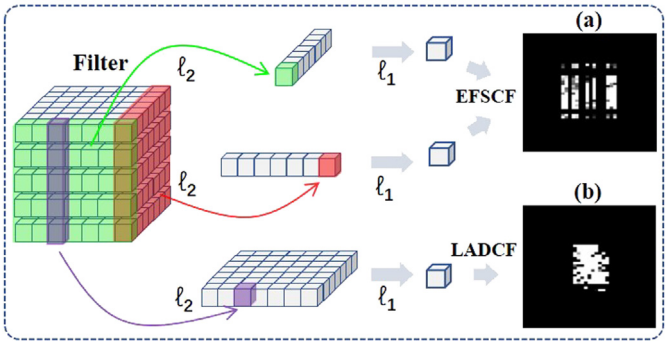


Fig. 2. The regular spatial term comparison results of the EFSCF and LADCF: (a) grayscale visualization of single-channel features of filters in the EFSCF; (b) grayscale visualization of signal-channel features of filters in the LADCF.

3.5.2. Optimization relationship

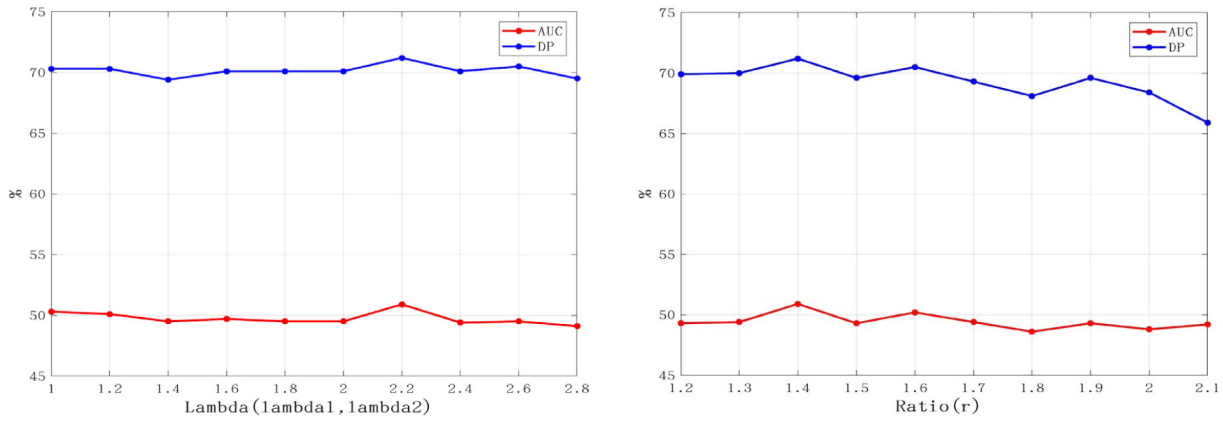
The BACF (Galoogahi et al., 2017) uses the ADMM (Boyd et al., 2010) to optimize the objective function to ensure that optimization complexity is up to $\mathcal{O}(QDMN\log(MN))$. This optimization technique has been adopted in the STRCF, LADCF, and sparse-based correlation filter (SCF) (Ji & Wang, 2018). The optimization method used in this work guarantees that the proposed method can be solved iteratively using the ADMM. To solve the sub-problems efficiently, GFSDCF (Xu, & (Feng et al., 2019)) is employed to perform row and column joint optimization to achieve fast convergence.

4. Experiments

To evaluate the robustness and effectiveness of the proposed tracker, extensive experiments were conducted on four public UAV datasets, and the proposed tracking method was compared with several state-of-the-art methods. In addition, two generic object tracking datasets were used to verify the generalization ability of the proposed model. In order to speed up the experiments and provide a fair tracking environment, non-deep trackers were equipped with only hand-crafted features, namely HOG (Dalal et al., 2010) with 31 channels, and CN (Weijer et al., 2009) with 10 channels. Furthermore, to evaluate the competitiveness of the proposed tracker, the EFSCF was compared with state-of-the-art deep trackers. Specifically, the comparison methods used in the experiment could be divided into the following two main categories:

Trackers with only hand-crafted features: SRDCF (Danelljan et al., 2015), STRCF (Li et al., 2018), LADCF (Xu et al., 2019a, 2019b), ECO (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017), BACF (Galoogahi et al., 2017), ARCF (Huang, & Fu et al., 2019; Huang, & Zhao et al., 2019), CRCDCF (Zhu et al., 2021), MSCF (Zheng et al., 2021), and ReCF (Lin et al., 2021).

Trackers with deep features: STRCF* (Li et al., 2018), LADCF* (Xu et al., 2019a, 2019b), ASRCF* (Dai et al., 2019), CCOT* (Danelljan et al., 2016), ECO* (Danelljan, & Bhat et al., 2017; Danelljan, & Hager et al., 2017), CF2* (Huang, Yang, & Yang, 2015), SiamFC* (Bertinetto et al., 2016), CFNet* (Valmadre, Bertinetto, Henriques,



(a) DP and AUC for different lambda_1 and lambda_2

(b) DP and AUC for different ratio r

Fig. 3. (a) The DP and AUC results of the EFSCF for different λ_1 and λ_2 values on the UAV123 dataset; (b) the DP and AUC results of the EFSCF for a different feature selection ratio r.

Vedaldi, & Torr, 2017), MCPF* (Zhang, Xu, & Yang, 2017), MCCT* (Wang, Zhou, Tian, Hong, Wang, & Li, 2018), UDT* (Wang et al., 2019) and UDT+* (Wang et al., 2019). The suffix * indicates that a tracker uses deep features.

4.1. Implementation details

The key parameters in the experiments were set as follows: spatial and temporal regularization parameters were set to $\lambda_1 = 2.2$, $\lambda_2 = 2.2$ and $\lambda_3 = 15$, respectively; the target size scaling for obtaining the search area was set to $S = 5$; the ratio of the row and column spatial feature selection was calculated by r/S, and $r = 1.4$. In addition, the number of scales of the detector was set to $N = 5$; iterative parameters were set to $\rho = 1.5$, $\mu_{\max} = 0.1$, and $\mu = 1$; the maximum number of iterations was set to $Q = 2$. The EFSCF model was implemented in MATLAB R2017b, and all trackers ran on the PC platform equipped with Intel® Xeon(R) CPU E5-2650 v4 @ 2.20 GHz × 24, 126 GB RAM, and GeForce GTX 1080 GPU. The source code of the proposed EFSCF can be downloaded at https://github.com/HonglinChu/EFSCF.

4.2. Datasets and evaluation metrics

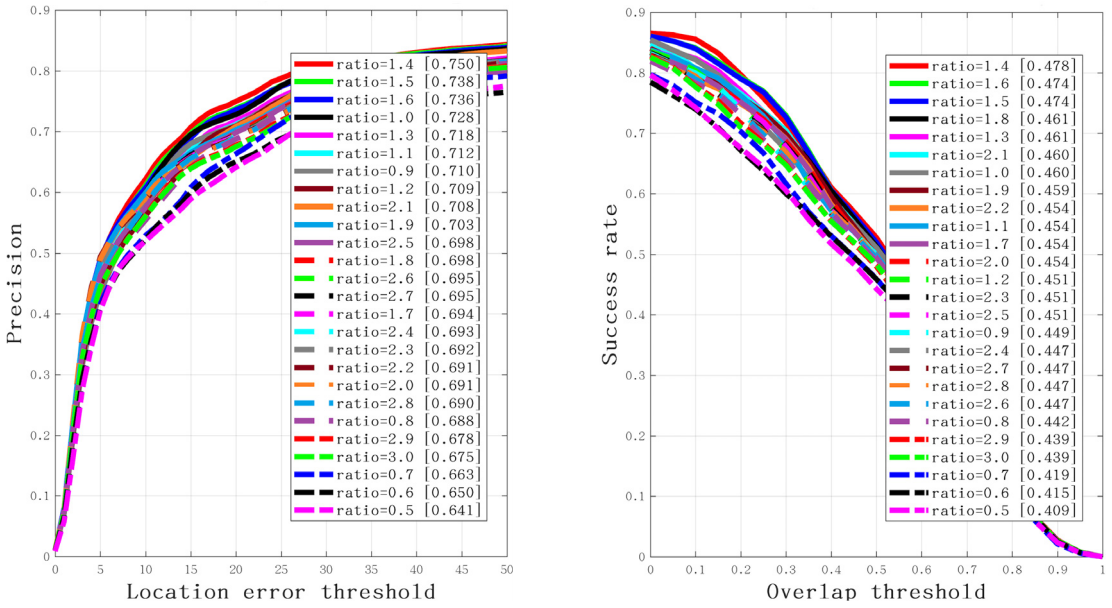
We extensively evaluated our method on the UAV123 (Mueller et al., 2016) and UAV20L (Mueller et al., 2016), UAVDT (Du et al., 2018), VisDrone2019 (Du et al., 2019), OTB2013 (Wu et al., 2013) and OTB2015 (Wu et al., 2015) datasets. The UAV123 dataset includes 123 aerial video sequences with a high resolution and a low altitude, and the total number of frames exceeds 110 K; currently, this is the largest public UAV tracking dataset. The UAV20L is a subset of the UAV123 dataset and contains 20 aerial video sequences for long-term tracking. The UAVDT focuses on tracking of vehicles and it consists of 30 training sequences and 70 test sequences; the test set includes 20 sequences for detection(DET) and multi-object tracking (MOT) tasks, and 50 sequences for single-object tracking (SOT) task. The VisDrone2019 dataset consists of 86 training sequences, 11 validation sequences, and 60 test sequences for SOT tasks. The OTB2013 and OTB2015 datasets contain 51 and 100 generic object tracking sequences, respectively. This accounts for a total of 430 sequences, including 123 sequences from the UAV123 dataset, 20 sequences from the UAV20L dataset, 50 test sequences of the SOT task from the UAVDT dataset, 86 training sequences of the SOT task from the VisDrone2019, and all sequences from the OTB2013 and OTB2015 datasets.

In the experiment, the one-pass evaluation (OPE) strategy of Wu et al. (2013) and Wu et al. (2015) was adopted, and the precision and success plots were used to evaluate the performance of different trackers. Three primary measures, including distance precision (DP), area under the curve (AUC), overlap precision (OP), were used to rank the trackers. The DP denoted the precision score, *i.e.*, the percentage of frames for which the distance between the center of the tracked bounding box and the center of the ground truth bounding box was within a threshold of 20 pixels. AUC denoted the success score, *i.e.*, the area under the curve of success plot. OP denoted the ratio of the number of the frames whose overlap rate between the tracked bounding box and the ground truth bounding box was more than 0.5 to the total number of frames.

4.3. Parameter analysis

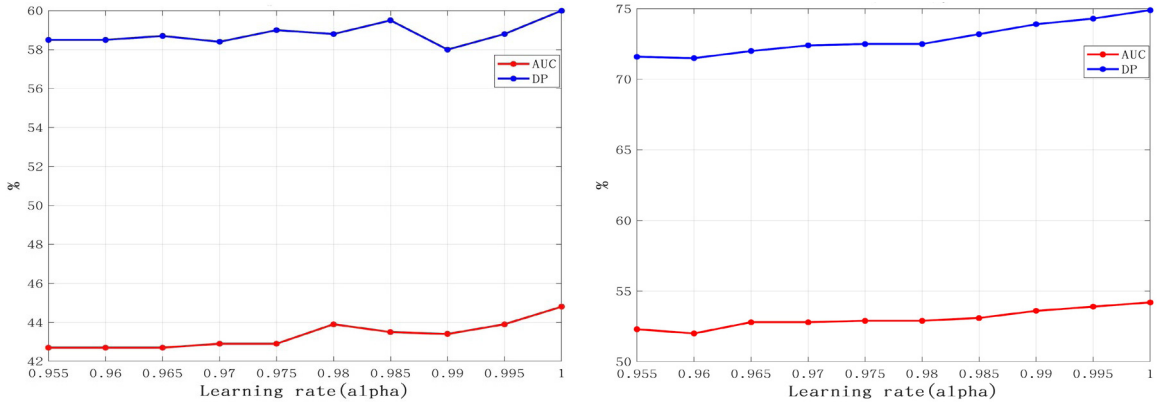
The spatial penalty parameters, feature selection ratio, and learning rate were analyzed. To reduce the complexity of parameter tuning, it was assumed that the sparse row constraint had the same importance as the sparse column constraint in the feature selection process, $\lambda_1 = \lambda_2$. As shown in Fig. 3(a), the DP and AUC results on the UAV123 datasets had the maximum at $\lambda_1 = 2.2$ and $\lambda_2 = 2.2$. In addition, as shown in Fig. 3(b), the impact of different values of ratio r was examined to validate the tracking performance on the UAV123 dataset. It was noted that a larger value of r indicated that more features in the search area were retained, which meant that more background pixels would be included, which would increase the risk of tracking drift. On the contrary, a smaller r indicated that fewer features were retained, which indicated that some of the crucial features for locating and tracking would be lost. In the experiment on the UAV123 dataset, the DP and AUC had the maximum at $r = 1.4$.

The precision and success plots on the UAVDT dataset are presented in Fig. 4. The proposed method's results obtained under different ratio values were sorted according to the AUC and DP values, and as shown in Fig. 4, the proposed EFSCF had the best performance when the ratio was equal to 1.4. The performances of the EFSCF on the UAV20L and VisDrone2019 datasets for different values of learning rate α are presented in Figs. 5(a) and 5(b), respectively. According to the results, when α increased from



(a) Precision plots of OPE-UAVDT (b) Success rate plots of OPE-UAVDT

Fig. 4. The precision and success plots under different ratio values on the UAVDT datasets.



(a) DP and AUC on UAV20L (b) DP and AUC on VisDrone2019

Fig. 5. (a) The DP and AUC results of the EFSCF for different values of learning rates α on the UAV20L dataset; (b) the DP and AUC results of the EFSCF for different values of learning rates α on the VisDrone2019 dataset.

Table 1
The average value of all the four UAV datasets on AUC and DP (pixels<20).

	Loss term	Spatial regularization	Temporal regularization	Average AUC%	Average DP%
STRCF	ℓ_2-norm	ℓ_2-norm	ℓ_2-norm	46.2	65.5
LADCF	ℓ_2-norm	ℓ_1-norm	ℓ_2-norm	46.4	65.8
V-1	$\ell_{2,1}$-norm	ℓ_2-norm	ℓ_2-norm	46.8	66.7
V-2	$\ell_{2,1}$-norm	ℓ_1-norm	ℓ_2-norm	46.6	65.9
V-3	ℓ_2-norm	$row-\ell_{2,1}+column-\ell_{2,1}$	ℓ_2-norm	49.2	69.8
EFSCF	$\ell_{2,1}$-norm	$row-\ell_{2,1}+column-\ell_{2,1}$	ℓ_2-norm	49.6	70.4

(Note: Red and blue fonts represent the top two results, respectively.)

0.955 to 1.0, Eq. (23) degenerated into Eq. (24), and the EFSCF obtained the maximum at $\alpha = 1$ in terms of DP and AUC.

4.4. Ablation study

In this section, the components, including the loss term and spatial regularization are compared and analyzed under different norms, and the impact of jointly sparse feature selection on the proposed method's performance is discussed. As shown in Table 1, three additional algorithms were designed and denoted by Algorithms V-1, V-2, and V-3. Algorithm V-1 was developed based on the STRCF by replacing the norm of loss term in Eq. (1) with the $\ell_{2,1}$-norm; Algorithm V-2 evolved from the LADCF by replacing the loss term in Eq. (2) with the $\ell_{2,1}$-norm; Algorithms

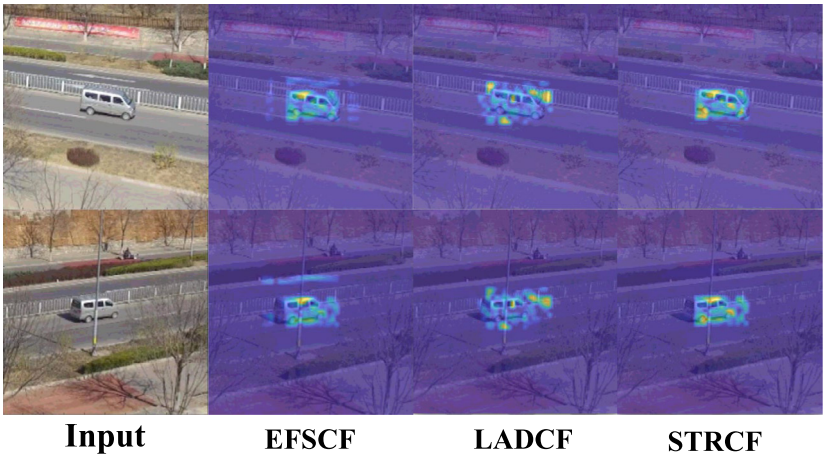


Fig. 6. The visualization of filters using the S1101 in the UAVDT dataset.

Table 2
Comparison results of different feature combinations used in the Algorithm V-3-based tracker on the UAV123, UAV20L, and UAVDT datasets (%).

Dataset		V-3	EFSCF
UAV123	AUC	50.8	**50.9**
	DP	**71.8**	71.2
UAV20L	AUC	**45.3**	45.2
	DP	60.3	**60.4**
UAVDT	AUC	46.0	**47.8**
	DP	72.6	**75.0**

(Note: The boldfaced values denote the highest accuracy among all the comparison methods on a dataset.)

V-3 evolved from the LADCF by replacing the regularization term in Eq. (2) with the row- and column-based $\ell_{2,1}$-norm. Compared with the STRCF and LADCF, Algorithms V-1 and V-2 applied the $\ell_{2,1}$-norm only to the loss term, and the improvement in performance was not significant. In contrast, the AUC and DP of Algorithm V-3 were 3% and 4.3% higher than those of the STRCF, respectively. Similarly, compared with the LADCF, Algorithm V-3 achieved improvements of 2.8% and 4% in the AUC and DP, respectively, demonstrating the effectiveness of the proposed jointly sparse feature selection method and indicating that using the row- and column-based $\ell_{2,1}$-norm was highly effective.

Next, Algorithm V-3 was compared with the EFSCF method on the UAV123, UAV20L, and UAVDT datasets using only the HOG and CN features. The performances of Algorithm V-3 and EFSCF on the UAV123 and UAV20L datasets are presented in Table 2, where it can be seen that their performances were approximately equal. On the UAVDT dataset, which contains highly challenging scenarios, weather, altitude, and viewpoint change, the EFSCF performed better than Algorithm V-3, which further proved that using the $\ell_{2,1}$-norm could improve the model's robustness against changes in target appearance.

The results indicated that the LADCF achieved spatial sparsity by using the ℓ_1-norm. In contrast, the EFSCF achieved structural sparsity by employing the row- and column-based $\ell_{2,1}$-norm. Further, as shown in Fig. 6, the search area contains too much background content, such as trees, road signs, streetlights, and railings, and the STRCF and LADCF were susceptible to background interference in the feature selection process. In contrast, in the EFSCF method, the majority of the energy of a learned filter gathered on the genuine target, even when the background was cluttered and complex. At the same time, the EFSCF could extract discriminant features from the background, thus providing additional clues for tracking. For instance, the features of railings, traffic directions, and roadbeds could be helpful to reduce the jitter of tracking results and improve stability further.

4.5. Quantitative analysis

4.5.1. Comparison of trackers with hand-crafted features

Overall performance: Different algorithms were quantitatively evaluated on four UAV datasets. Table 3 shows the AUC, DP (pixels < 20), and OP (IOU > 0.5) of the EFSCF and nine state-of-the-art trackers with only hand-crafted features. On the UAV123 dataset, the AUC of the EFSCF was 0.4% and 1.5% higher than those of the ECO and LADCF, respectively. The DP of the EFSCF was 1% higher than that of the LADCF. Even though the DP performance of the EFSCF was 1.2% lower than that of the ECO, the EFSCF performed better than the ECO on the AUC dataset and achieved almost the same competitiveness as the ECO on the OP. Moreover, the experimental results demonstrated that the overall performance of the EFSCF on all public datasets used in this study was better than that of the ECO. On the long-term tracking dataset UAV20L, the performance of the ECO degraded severely; the EFSCF performed better than the second-best tracker LADCF by 0.7%, 1.1%, and 0.5% regarding the AUC, DP, and OP metrics, respectively. This result supported the conclusion that the EFSCF was highly robust for long-term tracking. In addition, it is worth mentioning that, on the UAVDT dataset containing challenging scenarios (e.g., weather, altitude, and viewpoint changes), the EFSCF achieved the best score and performed better than the LADCF by 8.3% on DP and 4.5% on AUC. Compared with the second-best tracker, the ARCF, the proposed tracker could achieve 0.5% and 0.6% better performance on the DP and AUC, respectively. In terms of OP, the EFSCF achieved a score of 52.8%, which was only 0.3% lower than that of the tracker CRCDCF. On the VisDrone2019 dataset, the EFSCF performed better than the LADCF regarding the AUC, DP, and OP, but the AUC performance of the EFSCF was 1.2% and 0.2% lower than those of the ECO and ARCF, respectively. However, it should be noted that the EFSCF performed better than the ECO in terms of both DP and OP on the VisDrone2019 dataset. Considering the diversity between datasets, the individual performance of a tracker on a specific dataset cannot reflect its real performance. In view of that, following the work of Galoogahi et al. (2017) and Li et al. (2020b, 2020b), the average values of DP, AUC, OP, and FPS were calculated on the four UAV datasets.

Table 3
The comparison results of the EFSCF and nine state-of-the-art trackers based on the hand-crafted features in terms of AUC, DP (pixels < 20) and OP (IOU > 0.5) on UAV123, UAV20l, UAVDT and VisDrone2019 datasets.

		ReCF TITS21	MSCF ICRA21	CRCDCF TCSVT20	ARCF ICCV19	LADCF TIP19	STRCF CVPR18	BACF CVPR17	ECO CVPR17	SRDCF CVPR15	EFSCF OURS
UAV123	AUC%	47.6	48.1	47.7	47.0	49.4	47.8	45.8	50.5	45.9	50.9
	DP%	68.7	69.2	68.7	67.4	70.2	67.8	65.6	72.4	66.5	71.2
	OP%	57.7	56.9	58.5	56.9	57.5	56.0	55.1	60.0	54.9	59.8
UAV20L	AUC%	39.3	38.6	41.2	39.6	44.5	41.6	39.7	41.9	34.3	45.2
	DP%	54.4	56.6	57.6	55.9	59.3	56.0	55.4	54.7	50.7	60.4
	OP%	48.7	45.8	53.1	47.4	55.5	51.5	50.7	49.9	43.8	56.2
UAVDT	AUC%	47.1	46.8	47.8	47.0	43.1	41.8	43.9	42.8	42.8	47.6
	DP%	71.5	73.4	73.7	74.0	66.2	63.8	70.4	71.1	69.0	74.5
	OP%	52.3	51.7	53.1	51.8	47.5	45.8	47.5	44.7	45.5	52.8
VisDrone2019	AUC%	54.4	54.2	50.8	54.6	48.4	53.7	51.0	55.6	47.9	54.4
	DP%	72.5	73.0	72.0	73.5	67.4	74.2	69.9	74.7	65.3	74.9
	OP%	68.3	68.1	64.1	67.8	60.8	68.3	64.1	68.5	59.4	69.0

(Note: Red, blue and green fonts represent the top three results, respectively)

Table 4
The average values of the EFSCF and nine state-of-the-art trackers based on the hand-crafted features in terms of AUC, DP (pixels < 20), OP(IOU > 0.5) and FPS on the UAV123, UAV20L, UAVDT, and VisDrone2019 datasets.

		ReCF TITS21	MSCF ICRA21	CRCDCF TCSVT20	ARCF ICCV19	LADCF TIP19	STRCF CVPR18	BACF CVPR17	ECO CVPR17	SRDCF CVPR15	EFSCF OURS
Average Value	AUC%	47.1	46.9	46.9	47.1	46.4	46.2	45.1	47.7	43.3	49.6
	DP%	66.8	68.1	68.0	67.7	65.8	65.5	65.3	68.2	64.7	70.4
	OP%	56.8	55.6	57.2	56.0	55.3	55.4	54.4	55.8	50.9	59.5
	FPS	49	30	11	20	20	19	27	38	9	18

(Note: Red, blue and green fonts represent the top three results, respectively.)

Table 5
The comparison results of the EFSCF and nine state-of-the-art trackers based on the hand-crafted features in terms of AUC and DP (pixels < 20) on the OTB2013 and OTB2015 datasets.

		ReCF TITS21	MSCF ICRA21	CRCDCF TCSVT20	ARCF ICCV19	LADCF TIP19	STRCF CVPR18	BACF CVPR17	ECO CVPR17	SRDCF CVPR15	EFSCF OURS
OTB2013	AUC%	62.1	64.6	67.5	64.2	67.5	68.7	65.7	66.7	61.8	69.3
	DP%	81.7	85.8	88.0	85.0	86.4	89.2	86.1	88.9	82.3	89.2
OTB2015	AUC%	60.6	63.0	65.8	61.7	66.4	65.7	62.1	64.4	59.1	67.5
	DP%	79.4	83.6	86.3	81.8	86.4	86.5	82.4	85.8	77.6	87.5

(Note: Red, blue and green fonts represent the top three results, respectively.)

As shown in Table 4, the proposed tracker ranked first in terms of average AUC, DP, and OP. The proposed EFSCF surpassed the LADCF by 3.2%, 4.6%, and 4.2% in terms of the AUC, DP, and OP, respectively. In addition, EFSCF could run at approximately 18 fps, which was very close to the speed of the LADCF of 20 fps. Thus, the EFSCF achieved a more robust performance on the four UAV datasets compared to the other trackers.

Next, experiments were conducted on the OTB2013 and OTB2015 datasets to verify the generalization ability of the proposed EFSCF. As shown in Table 5, the proposed tracker ranked first, having the best AUC and DP scores among all trackers. In terms of the DP and AUC, the proposed tracker performed better than the LADCF by 2.8% and 1.8% on the OTB2013 dataset and by 1.1% and 1.1% on the OTB2015 dataset, respectively.

Attribute-based performance: The proposed EFSCF was compared with nine state-of-the-art hand-crafted trackers in terms of different attributes, including camera motion defined in the UAV123, viewpoint change and fast motion defined in the UAV20L, background clutter and scale variation defined in the UAVDT, and occlusion defined in the VisDrone2019. The precision and success plots of the six attributes are presented in Fig. 7. In the proposed method, the combination of row and column sparseness could learn more discriminant features to enhance the model's robustness to appearance changes caused by rapid motion, viewpoint changes, background clutter, scale changes, and occlusion. Compared with the LADCF, the EFSCF achieved improvements of 1.9% (DP) and 1.8% (AUC) in camera motion, 1.7% (DP) and 1.1% (AUC) in viewpoint change, and 3.4% (DP) and 2.3% (AUC) in fast motion. The superiority of the EFSCF over the LADCF was 7.8% (DP) and 5.3% (AUC) in background clutter attribute, 13.6% (DP) and 6.6% (AUC) in scale variation attribute, and 11.2% (DP) and 10.4% (AUC) in occlusion attribute. Consequently, the proposed EFSCF achieved the best performance in the attributes of viewpoint change, fast motion, background clutter, scale variation, and occlusion among all methods.

4.5.2. Comparison with trackers using deep features

The proposed method was compared with 12 state-of-the-art trackers using deep features to perform a comprehensive evaluation. As shown in Fig. 8, on the UAV20L, the proposed EFSCF using only hand-crafted features achieved the DP score of 60.4%, which was only 0.1% worse than that of the best tracker MCCT*, but outperformed the other six deep trackers (MCCT*, UDT*, CCOT*, MCPF*, UDT+*, CF2*) in terms of AUC. On the UAVDT dataset, the proposed tracker performed better in terms of DP and AUC than the six deep trackers (i.e., ASRCF*, ECO*, LADCF*, SiamFC*, STRCF*, and CFNet*). Table 6 shows the running speed results of the proposed algorithm and 12 state-of-the-art deep feature-based trackers. It should be noted that 12 deep feature-based trackers needed to run on the GPU device due to a large number of computing resources required, which increased the

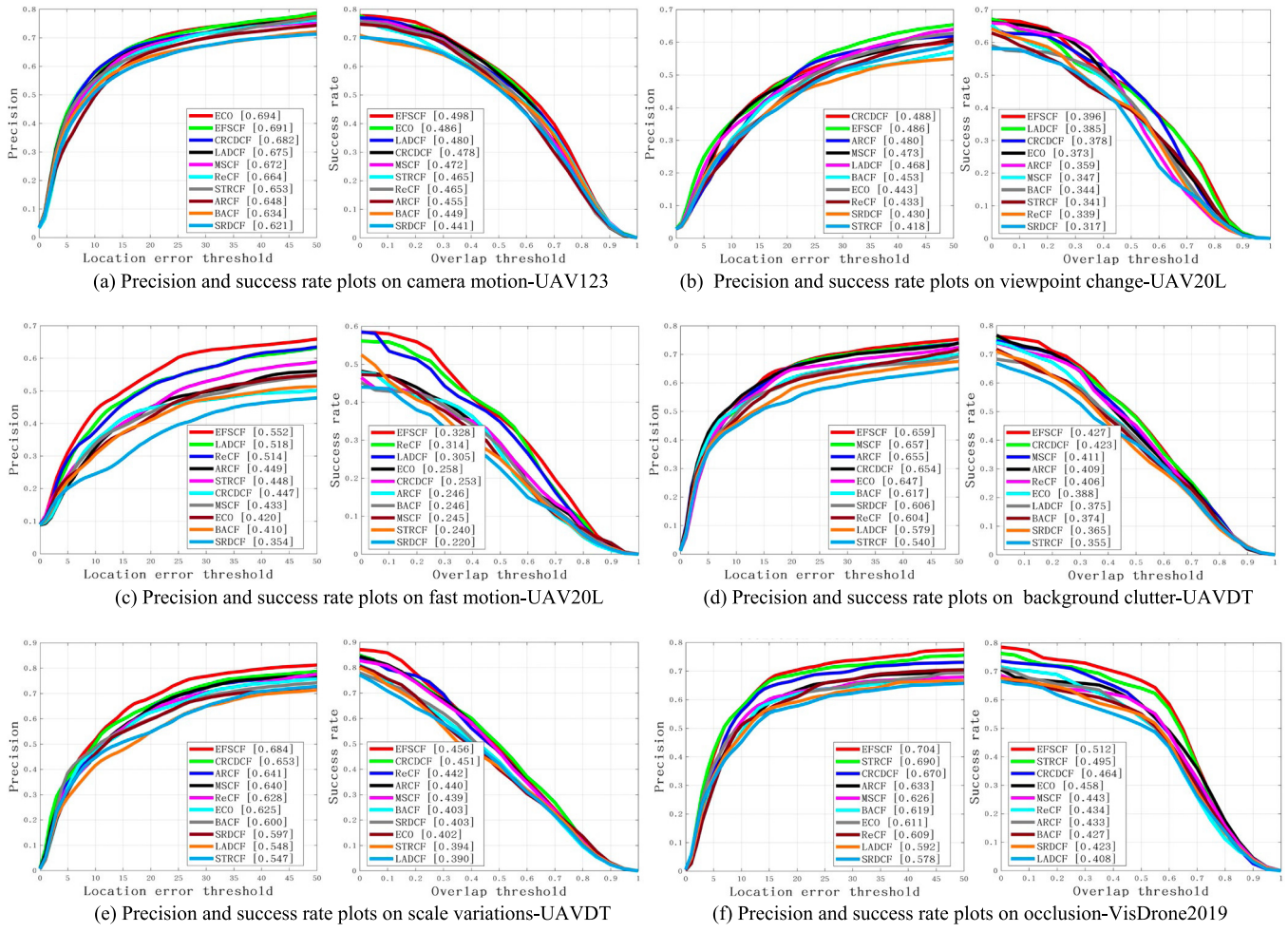


Fig. 7. The precision and success plots of six attributes, including camera motion, viewpoint change, fast motion, background clutter, scale variations, and occlusion on the UAV123, UAV20L, UAVDT, and VisDrone2019 datasets. In the legend, all trackers are ranked according to DP (pixels < 20) and AUC scores.

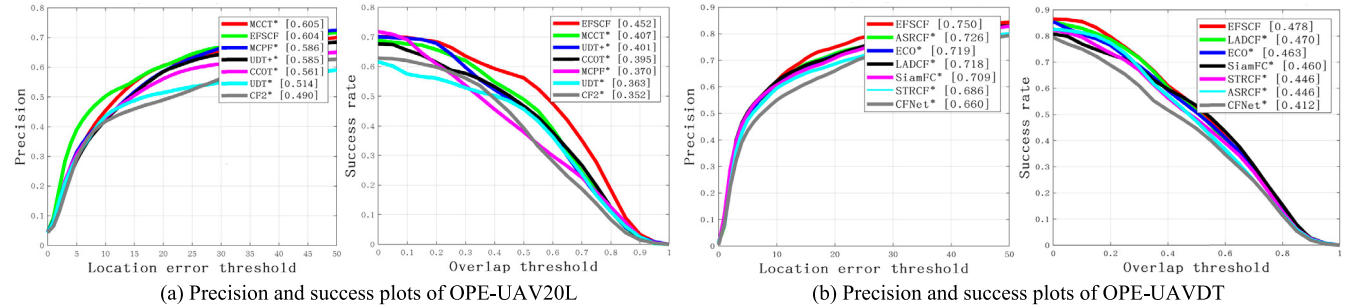
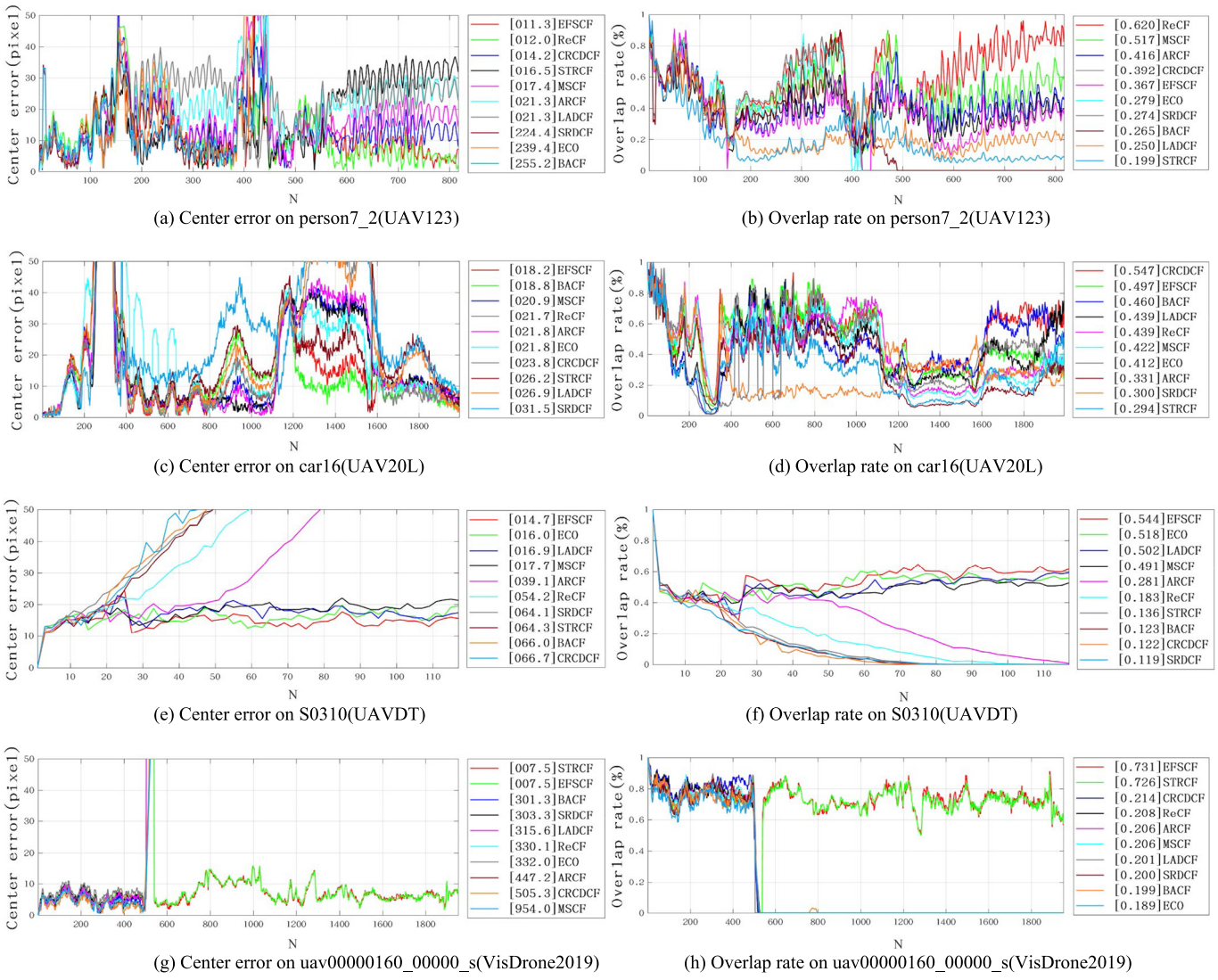


Fig. 8. The precision and success plots of the EFSCF and the state-of-the-art trackers based on the deep features on the UAV20L (a) and UAVDT (b) datasets. In the legend, all trackers are ranked according to DP (pixels < 20) and AUC scores; Suffix * represents the deep version of the corresponding tracker.

Table 6
The speed comparison between the EFSCF and the state-of-the-art trackers using deep features.

	ASRCF*	LADCF*	STRCF*	ECO*	CFNet*	SiamFC*	UDT+*	UDT*	MCCT*	MCPF*	CCOT*	CF2*	EFSCF
FPS	16	7	4	9	70	86	51	30	8	0.5	0.8	14	18
Device	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU	CPU



(a) Center error on person7_2(UAV123)
(b) Overlap rate on person7_2(UAV123)
(c) Center error on car16(UAV20L)
(d) Overlap rate on car16(UAV20L)
(e) Center error on S0310(UAVDT)
(f) Overlap rate on S0310(UAVDT)
(g) Center error on uav00000160_00000_s(VisDrone2019)
(h) Overlap rate on uav00000160_00000_s(VisDrone2019)

Fig. 9. The center error and overlap rate plots of the EFSCF and nine state-of-the-art trackers using the hand-crafted features on the four challenging sequences. The person7_2 with the SV, FM, VC, and CM attributes, car16 with the SV, FM, and OCC attributes, S0310 with the BC, SV, and OCC attributes and uav00000160_00000_s with the OCC and SV attributes were from the UAV123, UAV20L, UAVDT, and VisDrone2019 datasets, respectively. It should be noted that the SV, FM, VC, CM, OCC, and BC denoted the scale variation, fast motion, viewpoint change, camera motion, occlusion, and background clutter, respectively. In the legend, all trackers are ranked according to the average center error and average overlap rate. *N* is the number of frames.

hardware requirements of UAV applications. In contrast, the proposed algorithm could run at approximately 18 FPS on the CPU device.

4.5.3. Comparison with trackers using different feature combinations

Next, the proposed tracker was compared with the trackers using a combination of different features. As shown in Table 7, different feature combinations were used to evaluate the performance of the tracker based on Algorithm V-3. It is worth noting that only the row- and column-based $\ell_{2,1}$-loss was used to constrain the regularization term of the Algorithm V-3-based tracker. The HOG features can well describe the appearance and shape of a local area; CN features are robust to the rotation while maintaining high discrimination; CNN features are more discriminative due to their semantics. In the experiment, the CNN feature maps of the 'res4x' layer in the ResNet50 model were used. Table 7 shows that the Algorithm V-3-based tracker with the HOG features could easier obtain better tracking accuracy compared to the Algorithm V-3-based tracker with CN features. Thus, by combining the HOG and CN features, the tracking performance could be further improved. When the three types of features, the HOG, CN, and deep features, were combined, the Algorithm V-3-based tracker achieved the best results in terms of success rate and precision.

4.6. Qualitative analysis

In Fig. 9, the qualitative comparison results of the EFSCF and nine state-of-the-art trackers based on hand-crafted features on the challenging sequences are presented. In each frame, the center error denoted the pixel deviation between the center position of a tracked bounding box and the center of the ground-truth bounding box, and the overlap rate denoted the intersection-over-union of the tracked bounding box and the

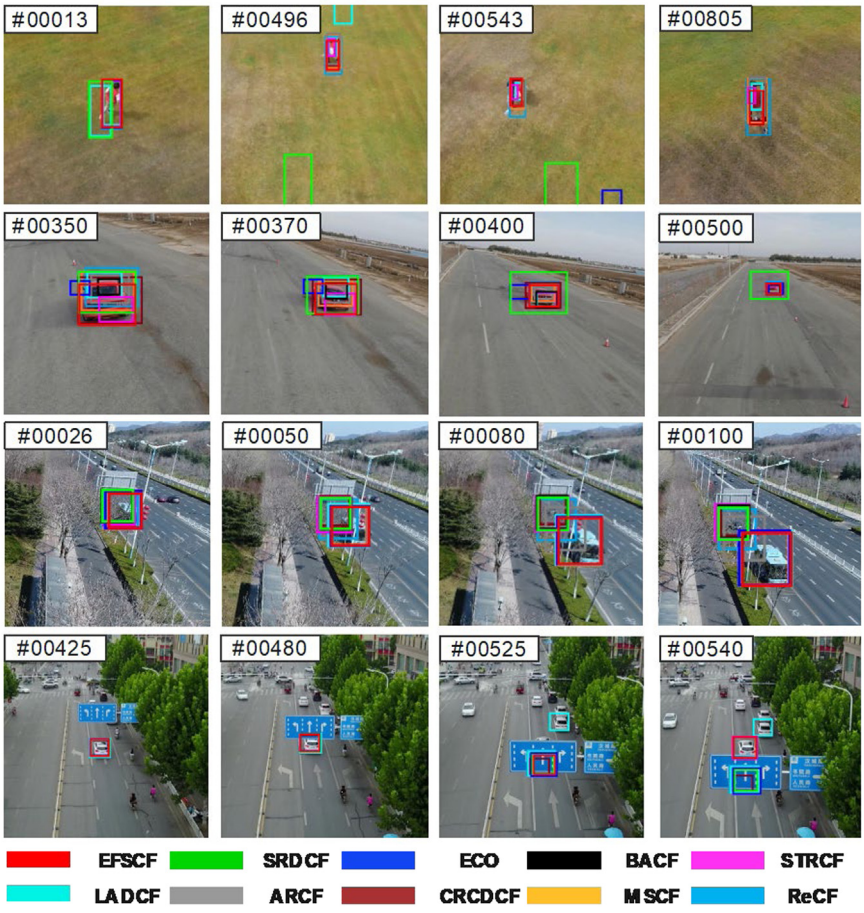


Fig. 10. The tracking results of person7_2, car16, S0310, and uav00000160_00000_s sequences (from top to down).

Table 7
Comparison of different feature combinations used in the Algorithm V-3-based tracker on the UAV123 dataset.

Features	Success rate	Precision
CN	0.388	0.654
HOG	0.499	0.697
HOG+CN	0.508	0.718
CNN+HOG+CN	**0.541**	**0.772**

(Note: The boldfaced values denote the highest accuracy on an index among all the comparison methods.)

ground-truth bounding box. For each sequence, the average center error was calculated by $\frac{1}{N}\sum_{i=1}^{N}\left(C_p^i - C_{GT}^i\right)$, where C_p^i and C_{GT}^i denoted the center positions of the tracked bounding box and ground-truth bounding box, respectively, and N denoted the total number of frames. The average overlap rate was calculated by $\frac{1}{N}\sum_{i=1}^{N}\left(\frac{B_p^i \cap B_{GT}^i}{B_p^i \cup B_{GT}^i}\right)$, where B_p^i and B_{GT}^i denoted the tracked bounding box and ground-truth bounding box, respectively. The targets of person7_2, car16, S0310, and uav00000160_00000_s all underwent severe appearance variations, and the proposed method ranked as the top one in terms of average center error. In terms of the average overlap rate, the proposed EFSCF still performed better than the LADCF and STRCF methods. As shown in Fig. 10, both person7_2 and S0310 had cluttered background and viewpoint changes, but the proposed EFSCF could accurately track the target. In the car16 sequence, the car moved fast on the road with scale changes, but the proposed tracker could track the car with high precision. In the uav00000160_00000_s sequence, the car was fully occluded at the 525th frame so that all trackers could not locate the target position. In addition, when the car reappeared at the 540th frame, the proposed tracker resumed tracking. Thus, benefiting from the structural sparsity feature selection method, the EFSCF contributed to overcoming the appearance variations, showing the superiority of the proposed method over the LADCF method.

4.7. Limitations

The efficiency is one of the limitations of the proposed method. As shown in Table 4, the FPS of the proposed method was lower than those of other trackers, such as MSCF and ReCF. From the perspective of theoretical analysis, the computation complexities of EFSCF, MSCF, and ReCF are of the same level. In fact, the MSCF and ReCF do not solve their models in every frame. On the contrary, the proposed algorithm requires model training and updating at every frame, resulting in a lower FPS compared with the MSFC and ReCF. The results also indicated that the FPS of the proposed method was almost the same as those of most of the trackers (e.g., ARCF, LACF, and STRCF). This limitation could be addressed in several ways, including program optimization, parallel computing technique, or upgradation of the hardware.

Regarding scale adaptation, the proposed method used the same strategy as the fDSST (Danelljan, & Hager et al., 2017, Danelljan, & Bhat et al., 2017). In this work, the proposed method mainly focuses on learning effective features. However, it does not perform well in scale adaptation. The index of overlap rate can well reflect the scale adaptation ability of a tracker. As shown by the overlap rate of person7_2 in Fig. 9, although the proposed

method had high center localization, it did not adapt well to the scale variation of a person. Future work could explore efficient model updating and scale adaptation strategies to enhance the overall performance of the proposed model.

5. Conclusion

This paper proposes an effective tracking method to explore the structural sparsity of learning discriminant features extracted from the foreground and background while handling the boundary effect and unexpected noise. The proposed method performs row and column sparse feature selection simultaneously to realize robust object tracking for the first time. The proposed method is verified by comprehensive evaluation experiments on four public UAV tracking datasets, and the results show that the overall performance of the proposed method outperforms those of nine state-of-the-art tracking methods based on handcrafted features. In addition, experiments on the UAV20L and UAVDT datasets, which use only HOG and CN descriptors, demonstrate the superiority of the proposed tracker compared with 12 other deep learning-based trackers. In addition, the proposed tracker can be efficiently solved by the ADMM-based iterative optimization method, which ensures that it can run at approximately 18 fps, meeting the basic requirement for video analysis on UAV platforms. Extensive experiments on the OTB2013 and OTB2015 datasets verify the potential generalization of the proposed tracker for generic tracking tasks.

Future work could focus on speed acceleration by using techniques such as algorithm optimization and parallel computing and tracker robustness enhancement. In addition, the promotion of object-tracking applications in a UAV shooting environment could be considered.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61976145 and Grant 62272319, in part by the Guangdong Basic and Applied Basic Research Foundation 2021A1515011318, and in part by the Shenzhen Municipal Science and Technology Innovation Council under Grant JCYJ20190808113411274 and Grant JCYJ20210324094413037.

References

Bao, C., Wu, Y., Ling, H., & Ji, H. (2012). Real time robust L1 tracker using accelerated proximal gradient approach. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 1830–1837).

Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. S. (2016). Fully-convolutional Siamese networks for object tracking. In *European conference on computer vision* (pp. 850–865).

Bolme, D. S., Beveridge, J. R., Draper, B. A., & Lui, Y. M. (2010). Visual object tracking using adaptive correlation filters. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 2544–2550).

Bonatti, R., Ho, C., Wang, W., Choudhury, S., & Scherer, S. (2019). Towards a robust aerial cinematography platform: Localizing and tracking moving targets in unstructured environments. In *IROS* (pp. 229–236).

Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2010). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, *3*(1), 1–122.

Brigham, E. O., & Morrow, R. E. (1967). The fast Fourier transform. *IEEE Spectrum*, *4*(12), 63–70.

Dai, K., Wang, D., Lu, H., Sun, C., & Li, J. (2019). Visual tracking via adaptive spatially-regularized correlation filters. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4670–4679).

Dalal, N., et al. (2010). Histograms of oriented gradients for human detection to cite this version : HAL Id : inria-00548512 Histograms of Oriented Gradients for Human Detection. In *IEEE computer society conference on computer vision and pattern recognition* (pp. 886–893).

Danelljan, M., Bhat, G., Shahbaz Khan, F., & Felsberg, M. (2017). ECO: Efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6638–6646).

Danelljan, M., Häger, G., Khan, F. S., & Felsberg, M. (2014). Accurate scale estimation for robust visual tracking. In *Proceedings of the british machine vision conference*.

Danelljan, M., Hager, G., Khan, F. S., & Felsberg, M. (2015). Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 4310–4318).

Danelljan, M., Hager, G., Khan, F. S., & Felsberg, M. (2017). Discriminative scale space tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(8), 1561–1575.

Danelljan, M., Khan, F. S., Felsberg, M., & Weijer, J. V. D. (2014). Adaptive color attributes for real-time visual tracking. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 1090–1097).

Danelljan, M., Robinson, A., Khan, F. S., & Felsberg, M. (2016). Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *Eur. conf. comput. vis.* (pp. 472–488).

Dong, Y., Yang, M., & Pei, M. (2016). Visual tracking with sparse correlation filters. In *IEEE international conference on image processing* (pp. 439-443).

Du, D., et al. (2018). The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the european conference on computer vision* (pp. 370–386).

Du, D., et al. (2019). VisDrone-SOT2019 : The vision meets drone single object tracking challenge results. In *Proceedings of the IEEE international conference on computer vision workshops*.

Elayaperumal, D., & Joo, Y. H. (2021). Aberrance suppressed spatio-temporal correlation filters for visual object tracking. *Pattern Recognition*, *115*, Article 107922.

Feng, W., Han, R., Guo, Q., Zhu, J., & Wang, S. (2019). Dynamic saliency-aware regularization for correlation filter-based object tracking. *IEEE Transactions on Image Processing*, *28*(7), 3232–3245.

Galoogahi, H. K., Fagg, A., & Lucey, S. (2017). Learning background-aware correlation filters for visual tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 1135–1143).

Galoogahi, H. K., Sim, T., & Lucey, S. (2015). Correlation filters with limited boundaries. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 4630–4638).

Hare, S., Saffari, A., & Torr, P. H. S. (2011). Struck : Structured output tracking with kernels. In *International conference on computer vision* (pp. 263–270).

He, Z., Yi, S., Cheung, Y. M., You, X., & Tang, Y. Y. (2017). Robust object tracking via key patch sparse representation. *IEEE Transactions on Cybernetics*, *47*(2), 354–364.

Henriques, J. F., Caseiro, R., Martin, P., & Batista, J. (2015). High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(3), 583–596.

Henriques, J. F., Caseiro, R., Martins, P., & Batista, J. (2012). Exploiting the circulant structure of tracking-by-detection with kernels. In *European conference on computer vision* (pp. 702–715).

Huang, Z., Fu, C., Li, Y., Lin, F., & Lu, P. (2019). Learning aberrance repressed correlation filters for real-time UAV tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 2891–2900).

Huang, J., Yang, X., & Yang, M. (2015). Hierarchical convolutional features for visual tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 3074–3082).

Huang, L., Zhao, X., & Huang, K. (2019). Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–17.

Ji, Z., & Wang, W. (2018). Correlation filter tracker based on sparse regularization. *Journal of Visual Communication and Image Representation*, *55*, 354–362.

Kalal, Z., Mikolajczyk, K., & Matas, J. (2011). Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(7), 1409–1422.

Karaduman, M., & Eren, H. (2019). UAV traffic patrolling via road detection and tracking in anonymous aerial. *Journal of Intelligent and Robotic*, *95*(2), 675–690.

Kong, H., Lai, Z., Wang, X., & Liu, F. (2016). Breast cancer discriminant feature analysis for diagnosis via jointly sparse learning. *Neurocomputing*, *177*, 198–205.

Kristan, M., et al. (2015). The visual object tracking VOT2015 challenge results. In *Proceedings of the IEEE international conference on computer vision workshops* (pp. 564–586).

Kristan, M., et al. (2019). The sixth visual object tracking VOT2018 challenge results. In *Proc. eur. conf. comput. vis.*.

Li, Y., Fu, C., Ding, F., et al. (2020a). AutoTrack: Towards high-performance visual tracking for UAV with automatic spatio-temporal regularization. In *Proceedings of the IEEE international conference on computer vision and pattern recognition* (pp. 11923–11932).

Li, X., Liu, Q., Fan, N., Zhou, Z., He, Z., & Jing, X.-Y. (2020b). Dual-regression model for visual tracking. *Neural Networks*, *132*, 364–374.

Li, X., Liu, Q., He, Z., et al. (2016). A multi-view model for visual tracking via correlation filters. *Knowledge-Based Systems*, *113*, 88–99.

Li, X., Ma, C., Wu, B., He, Z., & Yang, M. H. (2019). Target-aware deep tracking. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 1369–1378).

Li, F., Tian, C., Zuo, W., Zhang, L., & Yang, M. H. (2018). Learning spatial–temporal regularized correlation filters for visual tracking. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 4904–4913).

Li, Y., & Zhu, J. (2015). A scale adaptive kernel correlation filter tracker with feature integration. In *Proceedings of the european conference on computer vision* (pp. 254–265).

Liang, Y., Liu, Y., Yan, Y., Zhang, L., & Wang, H. (2021). Robust visual tracking via spatio-temporal adaptive and channel selective correlation filters. *Pattern Recognition*, *112*, Article 107738.

Liang, Y., Lu, X., He, Z., & Zheng, Y. (2019). Multiple object tracking by reliable tracklets. *Signal, Image Video Process*, 823–831.

Lin, F., Fu, C., He, Y., et al. (2021). Recf: Exploiting response reasoning for correlation filters in real-time UAV tracking. *IEEE Transactions on Intelligent Transportation Systems*.

Liu, Q., Li, X., He, Z., et al. (2020). Learning deep multi-level similarity for thermal infrared object tracking. *IEEE Transactions on Multimedia*, *23*, 2114–2126.

Liu, R., Wang, D., Han, Y., Fan, X., & Luo, Z. (2017). Adaptive low-rank subspace learning with online optimization for robust visual tracking. *Neural Networks*, *88*, 90–104.

Liu, Q., Yuan, D., Fan, N., et al. (2022). Learning dual-level deep representation for thermal infrared tracking. *IEEE Transactions on Multimedia*.

Mueller, M., Smith, N., & Ghanem, B. (2016). A benchmark and simulator for UAV tracking. In *European conference on computer vision* (pp. 445–461).

Nie, F., Huang, H., Cai, X., & Ding, C. (2010). Efficient and robust feature selection via joint $\ell 2$;1-norms minimization. In *Advances in neural information processing systems* (pp. 1–9).

Ross, D. A., Lim, J., Lin, R. S., & Yang, M. H. (2008). Incremental learning for robust visual tracking. *International Journal of Computer Vision*, *77*(1–3), 125–141.

Sherman, J., Morrison, W. J., Sherman, J., Morrison, W. J., Sherman, B. Y. J., & Morrison, W. J. (2015). Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, *21*(1), 124–127.

Smedt, F. D., Hulens, D., & Goedeme, T. (2015). On-board real-time tracking of pedestrians on a UAV. In *IEEE computer society conference on computer vision and pattern recognition workshops* (pp. 1–8).

Sui, Y., Wang, G., & Zhang, L. (2017). Correlation filter learning toward peak strength for visual tracking. *IEEE Transactions on Cybernetics*, *48*(4), 1290–1303.

Sui, Y., Wang, G., & Zhang, L. (2020). Joint correlation filtering for visual tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(1), 167–178.

Sui, Y., Zhang, Z., & Wang, G. (2016). Real-time visual tracking: Promoting the robustness of correlation filter learning. In *Proceedings of european conference on computer vision* (pp. 662–678).

Sun, S., Yin, Y., Wang, X., & Xu, D. (2019). Robust visual detection and tracking strategies for autonomous aerial refueling of UAVs. *IEEE Transactions on Instrumentation and Measurement*, *68*(12), 4640–4652.

Tang, M., & Feng, J. (2015). Multi-kernel correlation filter for visual tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 3038–3046).

Tao, J., Zhou, D., & Zhu, B. (2018). Robust latent regression with discriminative regularization by leveraging auxiliary knowledge. *Neural Networks*, *101*, 79–93.

Valmadre, J., Bertinetto, L., Henriques, J., Vedaldi, A., & Torr, P. H. S. (2017). End-to-end representation learning for correlation filter based tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2805–2813).

Wang, N., Song, Y., Ma, C., Zhou, W., Liu, W., & Li, H. (2019). Unsupervised deep tracking. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 1308–1317).

Wang, N., Wang, J., & Yeung, D. Y. (2013). Online robust non-negative dictionary learning for visual tracking. In *Proc. IEEE int. conf. comput. vis.* (pp. 657–664).

Wang, N., Zhou, W., Tian, Q., Hong, R., Wang, M., & Li, H. (2018). Multi-cue correlation filters for robust visual tracking. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition* (pp. 4844–4853).

Weijer, J. V. D., Schmid, C., Verbeek, J., & Larlus, D. (2009). Learning color names for real-world applications. *IEEE Transactions on Image Processing*, *18*(7), 1512–1523.

Wen, J., Lai, Z., Zhan, Y., & Cui, J. (2016). The L2, 1-norm-based unsupervised optimal feature selection with applications to action recognition. *Pattern Recognition*, *60*, 515–530.

Wu, Y., Lim, J., & Yang, M. H. (2013). Online object tracking: A benchmark. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*.

Wu, Y., Lim, J., & Yang, M. H. (2015). Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(9), 1834–1848.

Xu, T., Feng, Z. H., Wu, X. J., & Kittler, J. (2019a). Joint group feature selection and discriminative filter learning for robust visual object tracking. In *Proceedings of the IEEE international conference on computer vision* (pp. 7949–7959).

Xu, T., Feng, Z.-H., Wu, X.-J., & Kittler, J. (2019b). Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. *IEEE Transactions on Image Processing*, *28*(11), 5596–5609.

Xu, T., Feng, Z.-H., Wu, X.-J., & Kittler, J. (2020). Learning low-rank and sparse discriminative correlation filters for coarse-to-fine visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(10), 3727–3739.

Yi, S., Lai, Z., He, Z., Cheung, Y. M., & Liu, Y. (2017). Joint sparse principal component analysis. *Pattern Recognition*, *61*, 524–536.

Yu, Y.-F., Xu, G., Jiang, M., Zhu, H., Dai, D.-Q., & Yan, H. (2021). Joint transformation learning via the L2, 1-norm metric for robust graph matching. *IEEE Transactions on Cybernetics*, *51*(2), 521–533.

Yuan, D., Chang, X., Huang, P. Y., et al. (2021). Self-supervised deep correlation tracking. *IEEE Transactions on Image Processing*, *30*, 976–985.

Zhang, K., Li, X., Song, H., Liu, Q., & Lian, W. (2018). Visual tracking using spatio-temporally nonlocally regularized correlation filter. *Pattern Recognition*, *83*, 185–198.

Zhang, J., Liu, L., Zhen, L., & Jing, L. (2020). A unified robust framework for multi-view feature extraction with L2, 1-norm constraint. *Neural Networks*, *128*, 126–141.

Zhang, T., Xu, C., & Yang, M. H. (2017). Multi-task correlation particle filter for robust object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4335–4343).

Zheng, G., Fu, C., Ye, J., et al. (2021). Mutation sensitive correlation filter for real-time UAV tracking with adaptive hybrid label. In *IEEE international conference on robotics and automation* (pp. 503-509).

Zhu, X.-F., Wu, X.-J., Xu, T., Feng, Z.-H., & Kittler, J. (2021). Complementary discriminative correlation filters based on collaborative representation for visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(2), 557–568.