

Aircraft Tracking Based on an Antidrift Multifilter Tracker in Satellite Video Data

Ran Pang , Fang Gao , Peng Zhang, Xiangkun Li, and Yuwei Zhai

Abstract—Using remote sensing video to monitor aircraft dynamics is significant for military applications, airport management, and aircraft rescue. The aircraft has a fixed size and obvious characteristics, so it is suitable for correlation filtering. Correlation filtering algorithms can extract features from input data to predict motion trajectories, and the calculation speed of correlation filtering is fast. Hence, such algorithms are advantageous for tracking targets in remote sensing images. In this article, an antidrift multifilter tracker based on a correlation filter and the Kalman filter is proposed for this purpose. This article proposes a temporal consistency-constrained background-aware correlation filter algorithm based on temporal regularization that resists the model drift caused by clouds by using motion information to correct it. Experimental results show that our proposed method shows improved antidrift performance compared with other advanced tracking methods in cases of cloud occlusion and stable performance in other complex conditions. We believe that our model will be helpful for researchers who are interested in object tracking in satellite video, especially for processing satellite video data with cloud occlusion.

Index Terms—Cloudy conditions, model drift, object tracking, satellite videos.

I. INTRODUCTION

WITH the continuous development of video satellite technology in recent years, many video satellites (constellations) have been successfully launched worldwide. Video satellites can continuously observe dynamic changes on the Earth's surface, enabling long-term dynamic real-time monitoring of targets through remote sensing technology. At present, the Jilin-1 satellite constellation has 31 satellites in orbit, and 12 satellites have video imaging capabilities, as follows. The first-generation color video satellites include the Jilin-1 SP-01, SP-02, and LQ satellites. The Jilin-1 SP-03 satellite is a second-generation color video satellite. The Jilin-1 SP-04–SP-08 satellites are

Manuscript received 30 January 2023; revised 28 March 2023; accepted 19 April 2023. Date of publication 27 April 2023; date of current version 17 May 2023. This work was supported in part by the Scientific and Technological Plan of Changchun under Grant 21ZGG14, in part by the Key Scientific and Technological Research, and in part by the Key Scientific and Technological Research and Development Projects of Jilin 20220508034RC. (Corresponding author: Fang Gao.)

Ran Pang, Peng Zhang, Xiangkun Li, and Yuwei Zhai are with the Chang Guang Satellite Technology Company, Ltd., Changchun 130000, China (e-mail: pangran17@mails.jlu.edu.cn; zp920306@163.com; lixiangkun@charmingglobe.com; zhaiyuwei506@163.com).

Fang Gao is with the College of Computer Science and Technology, Jilin University, Changchun 130012, China. He is now with the Chang Guang Satellite Technology Company, Ltd., Changchun 130000, China (e-mail: gaofang@163.com).

Digital Object Identifier 10.1109/JSTARS.2023.3270884

third-generation dual-mode push-broom and gaze imaging video satellites. The fourth generation of small-batch-production video satellites includes the Jilin-1 GF-03C01–GF-03C03 satellites. These satellites can provide color videos at ten frames per second (fps) for up to 180 s with a spatial resolution of approximately 1 m. These remote sensing videos provide a basis for developing more diverse and convenient applications.

The development of high-resolution remote sensing video satellites has extensively promoted and enriched modern monitoring technologies and methods. The suitability of satellite data for change detection and monitoring applications depends on the data characteristics. Here, we provide a few examples of satellite video applications: oil and gas exploration [1], disaster monitoring [2], marine monitoring [3], monitoring for ecosystem changes and disturbances [4], traffic monitoring [5], change detection [6], and recognizing and monitoring military objects [7], [8]. Object tracking is a core step of such remote sensing data applications. To date, studies on tools for satellite video tracking, such as the video background extractor algorithm [9], have focused on the detection and tracking of moving targets. These algorithms use pretrained object detection modules to find targets in each frame and track them. Nevertheless, it is difficult to enable such a model to distinguish among objects within a class and acquire moving targets precisely.

Some methods based on deep learning have also been pursued in the existing research [10], [11], but after testing, it has been found that the data processing time of these methods is far from satisfying the needs of practical applications. Some methods based on correlation filtering for remote sensing target tracking have been presented in the existing research, which can serve as a reference for our work. However, the existing methods are often oriented toward a single application scenario. It is not always easy to maintain stable conditions in practical applications. The existing methods [7], [12], [13], [14] focus on simple video scenes and have difficulty dealing with complex conditions in target tracking, such as smoke, clouds, and light spots caused by changes in illumination. Algorithms of this kind also have difficulty when the target is moving slowly. Moreover, to date, research on the detection and tracking of moving targets has mainly focused on ground vehicles [13]. At present, research on other vehicles, such as aircraft and ships, is insufficient. Aircraft are a primary means of transportation and military use today. Therefore, this article focuses on developing an algorithm that can track aircraft quickly and accurately based on correlation filters.

The main problems encountered when tracking objects in remote sensing video are as follows:

- 1) Targets can be obscured by clouds.
- 2) Fast movement (large displacement) makes background change leading to tracking difficult.
- 3) Rotation-induced deformation can change the target appearance.
- 4) Because of the large data size, tracking can be time-consuming.

Due to widespread problems, there are many data of general quality that have not been used thus far. In particular, data in which targets are obscured by clouds are often not well utilized because of model drift and other problems.

The main contributions of this article are as follows:

- 1) By considering the potential motion relationships of a moving target during a certain period of time, we introduce a temporal consistency constraint into the BACF algorithm. Extensive experiments show that this method can effectively mitigate model drift. We quickly solve this model by ADMM in the frequency domain.
- 2) In this article, we use the Kalman filter (KF) to estimate the current location of the target from its visual information and then predict its future position by using the observation sequence. By analyzing and comparing the average peak-to-correlation energy (APCE) in each frame, we can estimate the degree to which we believe occlusion occurs. For cases of occlusion, a corrected fusion strategy based on the weighting of multiple trackers is proposed.

II. METHODOLOGY

A. Satellite Video Data and Preprocessing

The video data selected for this study include multiple videos taken by the No. 3 satellite of Jilin-1 provided by ChangGuang Satellite Co., Ltd., and the original satellite videos were acquired by the No. 3 Jilin-1 satellite. There are nine normal moving targets, three targets with complex background changes, four rotating targets, and four targets obscured by clouds. The original satellite videos have lengths of over 30 s with a frame rate of 10 fps and a spatial resolution of 0.92 m. The single-frame image size of the true-color (RGB) video is 12000*5000, and each video contains more than 300 frames. To facilitate our experiment, we do not use the full-time series videos. In addition, for convenience in labeling, we clip some of the data. Based on actual measurements and statistics, we believe that the size of the aircraft in the remote sensing video data is relatively stable. Size changes under a remote sensing lens are caused by rotation and occlusion; however, an aircraft has unique features, and its outer frame is close to square. Therefore, the changes in target size caused by rotation can be ignored. Therefore, we label the targets with a fixed rectangle. Moreover, we think that cloud cover generally affects objects only for discernible targets, and partially obscured targets are not our research targets.

We collected diversity data under four different conditions: normal flight, complex background change, target rotation, and cloud obscuration. Through the verification of different target tracking in complex situations, it is fully proven that our

model can adapt to aircraft tracking in complex situations. In the dataset, there are four series of videos in different cloudy conditions. These four series of videos help us verify that our model has antidrift ability. In Table I, we show the appearance characteristics of the tracking target. In Fig. 1, we show the motion trajectories of 20 sequential targets. The four sequences with occlusion are important research objects. We show the frames in cloudy sequences in Fig. 2.

B. Moving Object Tracking in Satellite Videos

Existing methods for object tracking in remote sensing videos include foreground detection methods, correlation filter methods, and deep learning methods. A common approach is to use time information (as in the background subtraction method, the optical flow method, and the interframe subtraction method) to highlight the areas exhibiting changes in consecutive frames and to start tracking without considering such information for the existing targets. With this approach, under conditions of noise, cloud, and light interference, the moving targets cannot be reliably detected in each frame. Additionally, deep learning methods are rarely selected for remote video applications, mainly because their speed has difficulty meeting the requirements of real applications. Another reason why the deep learning method is difficult to apply is that the existing video data are insufficient and the labeling cost is expensive, which makes it difficult to meet the training needs. Therefore, we instead choose the correlation filtering approach to solve the problem of aircraft tracking.

For learning from greyscale images, Bolme et al. [15] proposed the minimum output sum of squared error (MOSSE) filter, in which the minimum output and correlation frequency are applied for tracking. This method requires only simple calculations and can track objects quickly, but it cannot guarantee accurate tracking when the appearance of a moving target changes. Later, Henriques et al. [16] proposed training a correlation filter in kernel space and exploiting the circulant structure of the training patches. In 2014, Henriques et al. [17] proposed the method of kernelized correlation filters (KCF) by adjusting the channel features to multichannel features and introduced a color name (CN) feature for tracking. The CN feature improves the identification ability of a tracker. However, the adaptability of the tracker to rotation and fast motion still requires improvement. Subsequently, Danelljan et al. [18] and [19] proposed a discriminative scale-space tracker (DSST) using a feature pyramid to solve the multiscale changes problem; later, they also presented an improved DSST algorithm. With the rapid development of deep learning, the continuous convolution operator tracker (C-COT) algorithm [20] has emerged as a combination of correlation filtering and a convolutional neural network (CNN), in which spatial location information is simply represented by the features of a shallow CNN. This algorithm won the 2016 visual object tracking (VOT) competition. Similar to C-COT, the discriminative correlation filter with channel and spatial reliability (CSR-DCF) algorithm [21] also applies CNN features in combination with a correlation filter. The use of CNN features improves the robustness of the algorithm. Tang and Feng [22] proposed multiple kernelized

TABLE I
STATE OF DIFFERENT MODELS ON OUR DATASET





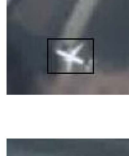
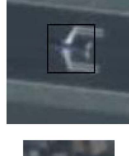
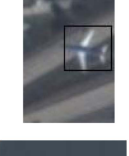



object	image	object_size	image_size	frame_numbers	video_id
cloudy_less_01		30*30	3814*1348	235	1
cloudy_little_01		37*35	12000*5000	216	2
cloudy_little_02		30*32	12000*5000	326	3
cloudy_more_01		43*49	3308*1124	98	4
complex_background_01		36*27	4206*1012	55	5
complex_background_02		37*38	3308*1124	327	4
complex_background_03		37*35	4094*1460	300	6
regular_01		45*39	3308*1124	327	4
regular_02		44*42	5046*1854	326	7
regular_03		55*40	5046*1854	326	7

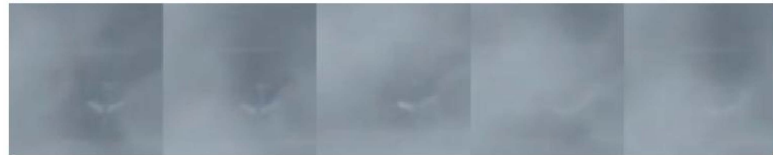
TABLE I
(CONTINUED)

regular_04		80*74	5046*1854	326	7
regular_05		85*73	5046*1854	326	7
regular_06		41*38	5046*1854	326	7
rotation_01		49*43	5046*1854	326	7
rotation_02		43*38	5046*1854	326	7
rotation_03		46*51	5046*1854	326	7
rotation_04		75*72	5046*1854	326	7
rotation_05		42*39	3308*1124	327	7
rotation_06		49*38	5046*1854	326	7
rotation_07		61*59	3308*1124	327	4



Fig. 1. Trajectories of the 20 sequences.

cloudy_less_01



cloudy_little_01



cloudy_little_02



Fig. 2. Frame of the cloudy sequences.

correlation filters (MKCFs) in 2015. MKCF can achieve stronger discrimination than KCF through the introduction of multikernel learning (MKL) into KCF. In 2018, the MKL-based tracker MKCFup [23] was proposed by reconstructing the correlation filter objective function. This improvement significantly reduced the detrimental mutual interference among different particles. In the learning process of the method of spatially regularized discriminative correlation filters (SRDCF) [24], a spatial adjustment component was introduced to punish the correlation filter coefficients in accordance with their spatial positions. After that, Li et al. [25] proposed a tracker based on spatial-temporal regularized correlation filters (STRCF), combining temporal and spatial regularization constraints, which showed better performance than SRDCF in terms of both accuracy and speed.

The efficient convolution operator (ECO) [26] was introduced as a novel formulation for the training and application of a continuous convolution filter. An implicit interpolation model is used to model the learning process in a continuous spatial domain. However, the above-mentioned tracking methods based on correlation filters are sensitive to boundary effects due to boundary samples that are not truly negative samples in real scenes, which affects their tracking performance. In contrast, the BACF [27] is a learning/updating filter that can effectively extract negative samples from the background in real time rather than focusing solely on moving foreground patches. Before this article, some articles studied tracking methods for occlusion. The visibility of the target will be different even if the cloud is completely obscured by the thin environment. In CFME,

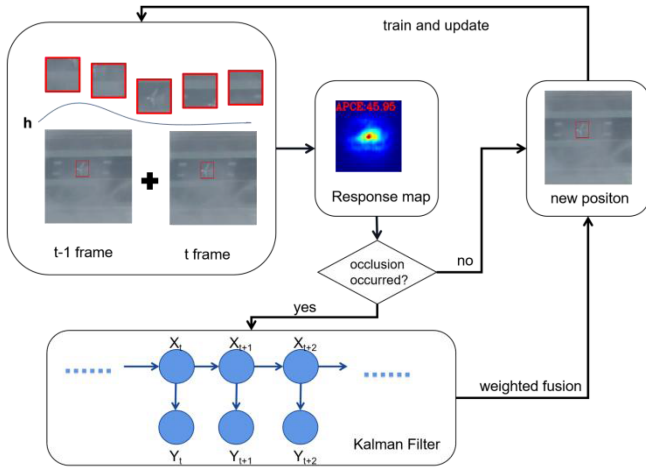


Fig. 3. Pipeline of the proposed ADMFT. First, we use the previous filter to calculate the response map. Then, if the confidence conditions are not met, the model is updated based on the current frame and the previous model. Otherwise, the position will be ensured by the KF result and the correlation filter result. The parameters of the KF are calculated depending on the frame number of occlusion occurrence.

proposed by Xuan et al. [12], occlusion is processed through an update strategy. Recently, some convolutional regression network and motion features [28], [29], [30], [31] are integrated for final target location prediction. Shangtang Intelligent Video Team has performed a series of work on the twin network, including the Siamese region proposal network (SiamRPN) [32], which implements the first high-performance twin network tracking algorithm after introducing detection into tracking. Cavity convolution is introduced into the Siamese box adaptive network (SiamBAN) [33]. Experiments show that cavity convolution can increase the receptive field and improve tracking performance. The anchor-free reference in SiamBAN removes the predefined anchor, which reduces the overall parameters of the model and further improves the speed. Siamese fully convolutional classification and regression (SiamCAR) [34] has an additional centrality branch to better determine the location of the target center point. Through the anchor-free strategy, the regression output of the network is transformed into the distance between the feature map point on the search patch and the four sides of the selected ground-truth box. Such methods are not suitable for all complex environments. Target features will change due to cloud cover. Considering this change when expressing features can better track in a cloudy environment.

For object tracking in a remote sensing video, the tracking updates will constantly drift when occlusion occurs. Under such conditions, the principle of an antidrift multifilter tracker (ADMFT) is to learn a relatively stable model over a certain period of time. However, this regularization strategy imposes unequal penalties on the filter coefficients, causing the filter to learn the appearance features of the deformed target. The ADMFT algorithm uses the BACF to process complex background changes. The BACF can deal with rotation by truly negative samples in real scenes. It is difficult to estimate position solely on the basis of appearance features. However, in general, the motion state of

an aircraft should always be stable. But predicting only by the motion state could not deal with complex motion trajectories. Therefore, the predicted result for the motion state can be used to correct the predicted position.

C. Development of an Antidrift BACF via the Introduction of Temporal Regularization

First, we briefly revisit the BACF formula. The correlation filter learns the optimal $E(h)$ by optimizing the following formula:

$$E(h) = \frac{1}{2} \sum_{t=1}^T \|v_t - hP^T u [\Delta\tau_t]\|_2^2 + \frac{\lambda}{2} \|h\|_2^2 \quad (1)$$

where P is a $D \times T$ binary matrix, with T being the number of pixels. u denotes a training image sample, v denotes the corresponding output centered on the peak of the target, and W represents the correlation filter. $u \in \mathbb{R}^T$, $v \in \mathbb{R}^T$ and $h \in \mathbb{R}^D$. $u[\Delta\tau_i]$ denotes the circular shift operator of U . Operator T denotes a conjugate transpose. λ is a regularization. With the application of the circle shift operator, the number of samples will increase. To improve the speed, we express the above-mentioned formula in the frequency domain as follows:

$$E(h, \hat{g}) = \frac{1}{2T} \left\| \hat{v} - \langle \hat{U}, \hat{g} \rangle \right\|_2^2 + \frac{\lambda}{2} \|h\|_2^2$$

$$s.t. \hat{g} = \sqrt{T}(FP^T \otimes I)h. \quad (2)$$

Here, $\hat{\cdot}$ denotes the discrete Fourier transform, and \otimes denotes the Kronecker product. \hat{g} is an auxiliary variable. F denotes the orthonormal matrix of complex basis vectors for mapping to the Fourier domain for any T -dimensional vectorized signal.

Deformation, occlusion, or a complex background of the target will impact the tracking performance. For example, if occlusion occurs, the BACF tracker will lose the target. Even if the occlusion disappears in subsequent video frames, the tracker cannot relocate the target. In previous studies, termination of the model update process was often used to overcome occlusion. We believe that the main difference between cloud occlusion and the other types of occlusion is that clouds have certain transparency. Many methods of resisting model drift are based on ceasing to update the model when occlusion occurs. We believe that although the apparent features of the target change due to occlusion in the case of cloud occlusion, these changes should not be neglected. For a moving target, the target has a potential motion relationship between consecutive frames. Considering the motion relationship of a moving target within a certain period, we introduce an L2 regularization term constraint and propose minimizing the following objective function to train the improved BACF algorithm:

$$E(h) = \frac{1}{2} \sum_{t=1}^T \|v_t - h^T u [\Delta\tau_t]\|_2^2$$

$$+ \frac{\lambda}{2} \|h\|_2^2 + \frac{\eta}{2} \sum_{t=1}^T \|v - h^T u_{t-1}\|_2^2$$

$$s.t. g = (P^T \otimes I)h \quad (3)$$

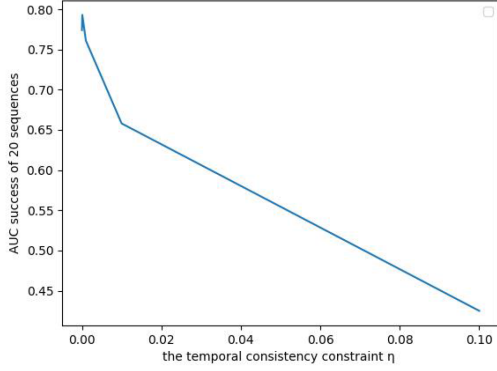
Fig. 4. Impacts of the temporal consistency constraint η on 20 sequences.

TABLE II
COMPARISON OF THE PERFORMANCES OF DIFFERENT MODELS ON OUR DATASET. THE BEST VALUES IS HIGHLIGHTED IN BOLD

Tracker	AUC	FPS
ADMFT (ours)	0.793	83.84
CSRDCF	0.776	21.65
BACF	0.768	97.35
CN	0.762	110.68
SiamRPN	0.745	14.32
SiamCAR	0.722	36.17
SiamBAN	0.755	31.09
CSK	0.708	136.07
ECO	0.750	18.71
MKCFup	0.732	13.66
KCF	0.667	187.86
STRCF	0.609	95.32

where η is a regularization parameter ($\lambda, \eta \geq 0$) and $\mu > 0$ is the corresponding penalty factor, which is used to adjust the function of the target in the previous frame for model training in the current frame. The last term in the above-mentioned formula is a global temporal consistency constraint.

To improve computational efficiency, a correlation filter is usually converted into the frequency domain by means of the Fourier transform. In this way, the proposed filter can be represented in the frequency domain as follows:

$$\begin{aligned}
 E(w, \hat{g}) &= \frac{1}{2} \|\hat{v} - \langle \hat{u}_t, \hat{g} \rangle\|_2^2 + \frac{\lambda}{2} \|h\|_2^2 \\
 &\quad + \frac{\eta}{2} \|\hat{v} - w^\top \hat{u}_{t-1}\|_2^2 \\
 s.t. \hat{g} &= \sqrt{T}(FP^\top \otimes I)h.
 \end{aligned} \quad (4)$$

To solve the above-mentioned formula, we rewrite it using the augmented Lagrange method [35]

$$\begin{aligned}
 \mathfrak{L}(w, \hat{g}, \hat{\zeta}) &= \frac{1}{2} \|\hat{v} - \langle \hat{u}_t, \hat{g} \rangle\|_2^2 + \frac{\lambda}{2} \|h\|_2^2 \\
 &\quad + \eta \|h^\top \hat{u}_{t-1} - \hat{v}\|_2^2
 \end{aligned}$$

$$\begin{aligned}
 &+ 2\hat{\zeta} \left(\hat{g} - \sqrt{L} (FP^\top \otimes I) h \right) \\
 &+ \mu \left\| \hat{g} - \sqrt{L} (FP^\top \otimes I) h \right\|_2^2
 \end{aligned} \quad (5)$$

where ζ denotes a complex Lagrangian multiplier. This equation can be solved iteratively using the ADMM technique, and each of the subproblems, \hat{g} and h , has a closed-form solution.

Subproblem h is solved as follows:

$$\begin{aligned}
 h &= \arg \min_h \mathfrak{L}(h, \hat{g}, \hat{\zeta}) \\
 &= \arg \min_h \left\{ \frac{\lambda}{2} \|h\|_2^2 \right. \\
 &\quad + \hat{\zeta}^\top \left(\hat{g} - \sqrt{T} (FP^\top \otimes I) h \right) \\
 &\quad \left. + \frac{\mu}{2} \left\| \hat{g} - \sqrt{T} (FP^\top \otimes I) h \right\|_2^2 \right\} \\
 &= \left(\left(\mu + \frac{\lambda}{\sqrt{T}} \right) I + \eta \hat{s}_u \right)^{-1} (\mu g + \zeta + \eta \hat{s}_v)
 \end{aligned} \quad (6)$$

where g and ζ are defined as $g = \frac{1}{\sqrt{T}}(PF^\top \otimes I)\hat{g}$ and $\zeta = \frac{1}{\sqrt{T}}(PF^\top \otimes I)\hat{\zeta}$, respectively, $\hat{s}_u = \hat{u}_t^\top \hat{u}$, and $\hat{s}_v = \hat{u}_t^\top \hat{v}$.

Subproblem \hat{g} is solved as follows:

$$\begin{aligned}
 \hat{g} &= \arg \min_{\hat{g}} \mathfrak{L}(h, \hat{g}, \hat{\zeta}) \\
 &= \arg \min_{\hat{g}} \left\| \langle \hat{u}_t, \hat{g} \rangle - \hat{v} \right\|_2^2 \\
 &\quad + \lambda \|h\|_2^2 + \eta \|h^\top \hat{u}_{t-1} - \hat{v}\|_2^2 \\
 &\quad + 2\hat{\zeta} \left(\hat{g} - \sqrt{T} (FP^\top \otimes I) h \right) \\
 &\quad + \mu \left\| \hat{g} - \sqrt{T} (FP^\top \otimes I) h \right\|_2^2.
 \end{aligned} \quad (7)$$

We express problem \hat{g} as an independent problem and directly obtain the solution to (7)

$$\begin{aligned}
 \hat{g}(t)^* &= \frac{1}{\mu} (T\hat{v}_t \hat{u}_t - \hat{\zeta}_t + \mu \hat{h}_t) \\
 &\quad - \frac{\hat{u}_t}{\mu b} (T\hat{v}_t \hat{s}_u(t) - \hat{s}_\zeta(t) + \mu \hat{s}_h(t))
 \end{aligned} \quad (8)$$

where $\hat{s}_u(t) = \hat{u}_t^\top \hat{u}$, $\hat{s}_\zeta(t) = \hat{u}_t^\top \hat{\zeta}$, $\hat{s}_h(t) = \hat{u}_t^\top \hat{h}$ and $b = \hat{s}_u(t) + T\mu$.

Subproblem $\hat{\zeta}$ is solved as follows:

$$\hat{\zeta} = \hat{\zeta} + \mu(\hat{g} - \hat{h}) \quad (9)$$

where $\hat{h} = \sqrt{T}(PF^\top \otimes I)h$ and μ is a penalty factor. We update μ by using the iterative ADMM. $\mu = \min(\mu_{\max}, \beta\mu)$, where μ_{\max} denotes the maximum value of μ and β is a scale factor.

We choose the histograms of oriented gradients (HOG) feature and the CN feature to extract the feature map, where the HOG feature is a gradient feature and the CN feature is a color feature. Accordingly, these two features can complement each other to help better satisfy the tracking objective.

TABLE III

CLE RESULTS FOR 9 TRACKERS ON 20 SEQUENCES. A TRACKER WITH A SMALLER CLE (IN PIXELS) EXHIBITS BETTER PERFORMANCE IN THE TRACKING PROCESS. THE BEST AND SECOND-BEST VALUES ARE HIGHLIGHTED IN BOLD AND UNDERLINED, RESPECTIVELY

Motion state	Target	ADMFT	CSRDCF	CN	BACF	ECO	MKCFup	CSK	KCF	STRCF	Siam RPN	Siam CAR	Siam Ban
cloud occlusion	cloudy_less_01(a)	3.34	5.03	9.06	10.87	38.07	8.04	115.60	8.52	35.67	6.72	4.66	4.56
	cloudy_little_01(b)	3.04	1.86	2.24	2.86	2.79	2.94	66.84	2.43	5.48	137.28	136.10	137.64
	cloudy_little_02(c)	3.19	3.38	3.54	3.00	3.76	3.03	3.87	262.88	3.51	3.20	2.66	4.70
	cloudy_more_01(d)	1.23	0.91	0.58	1.60	0.94	1.08	0.96	1.94	2.30	4.30	2.02	4.30
complex background	complex_back_ground_01(e)	2.99	2.68	2.98	3.25	3.09	3.37	2.92	3.05	4.13	6.77	4.37	5.56
	complex_back_ground_2(f)	4.29	5.07	4.16	4.88	4.55	4.15	4.36	4.80	4.75	6.28	6.26	5.01
	complex_back_ground_03(g)	6.35	5.78	6.44	5.61	5.10	6.16	8.58	5.69	1197.57	636.32	88.89	5.05
regular	regular_01(i)	4.84	4.61	5.40	4.58	4.21	5.02	6.17	6.85	2.24	3.00	2.94	<u>2.63</u>
	regular_02(i)	3.74	<u>4.03</u>	4.40	4.89	5.35	5.82	4.66	20.65	8.76	2.20	2.49	2.46
	regular_03(i)	2.18	3.04	2.04	3.06	2.11	2.28	2.14	3.81	9.82	2.17	4.64	<u>2.10</u>
	regular_02(i)	3.58	4.43	3.70	4.48	3.32	3.73	3.81	23.20	3.80	5.92	5.81	3.43
	regular_02(i)	2.29	2.04	2.85	<u>2.06</u>	2.84	2.04	3.56	4.60	2.84	5.85	6.08	8.82
	regular_02(i)	6.93	2.96	4.69	7.17	3.32	4.44	4.45	11.79	73.69	3.03	2.07	<u>2.52</u>
	regular_02(i)	2.01	3.88	3.34	3.85	<u>2.20</u>	2.86	3.28	4.70	8.12	3.23	3.45	4.10
rotation	rotation_01(j)	2.10	<u>2.16</u>	3.38	2.23	<u>2.16</u>	2.17	3.91	4.66	8.18	4.30	4.38	4.49
	rotation_02(k)	<u>3.58</u>	4.43	3.70	4.48	3.32	3.73	3.81	23.20	3.80	5.92	5.81	3.43
	rotation_01(j)	10.31	11.04	12.42	10.59	10.32	12.29	12.34	23.39	10.89	5.09	6.39	6.82
	rotation_01(j)	5.03	7.29	6.96	6.95	23.54	18.07	15.89	33.95	55.50	2.48	2.69	2.44
	rotation_01(j)	5.91	6.02	6.08	5.79	7.10	<u>5.51</u>	5.64	11.13	4.06	4.29	6.37	5.23
	rotation_01(j)	10.10	<u>8.75</u>	9.52	8.67	22.77	72.22	11.80	12.62	12.09	3.45	2.52	3.12
	rotation_01(j)	6.93	2.96	4.69	7.17	3.32	4.44	4.45	11.79	73.69	3.03	2.07	<u>2.52</u>
mean_CLE		4.363	4.9145	5.02	7.498	8.4485	14.218	23.0065	73.099	42.48	42.48	29.85	10.90

A tracker with a smaller CLE (in pixels) exhibits better performance in the tracking process. The best and second-best values are highlighted in bold and underlined, respectively.

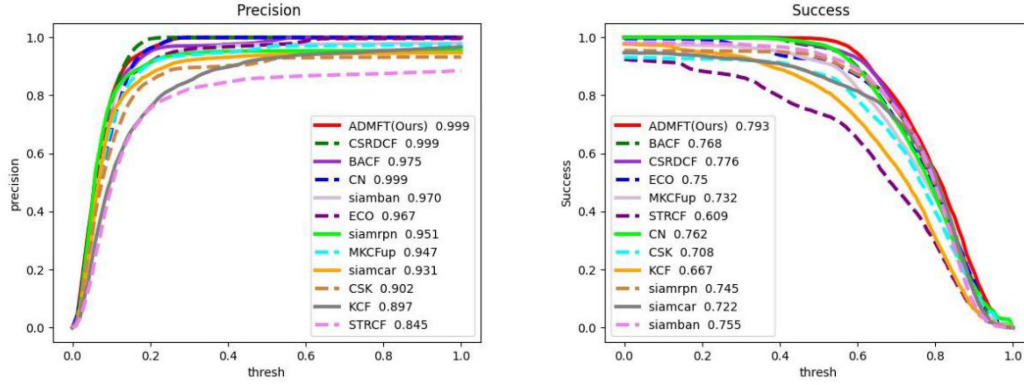


Fig. 5. Overlap success plots over eight challenging attributes. The abscissa shows the overlap threshold, ranging from 0 to 1. Precision and success plots show the performance of our ADMFT compared with other CFTs on 20 sequences.

D. Motion Estimator

In the KF, only the current measured value and the estimated value from the previous sampling period are needed to estimate the state, which does not require much storage space. The number of calculations in each step is small, and the calculation steps are clear, making this filter very suitable for computer processing. The KF can help estimate the positions and velocities of moving targets. However, the parameters of the KF are difficult to determine. To this end, we use a frame-based parameter selection strategy. Specifically, we use the expectation maximization (EM) algorithm to estimate the parameters [36] when the frame number is greater than a certain threshold. When the frame number is greater than a certain threshold, the dynamics and observation model can be written as follows:

$$x_{t+1} = Ax_t + w_t \quad (10)$$

$$y_t = Cx_t + r_t \quad (11)$$

where x_t and x_{t-1} are the state vectors of the system at times t and $t-1$, respectively. In this article, we choose the state vector $x_t = [xs_t, ys_t, xv_t, yv_t]^T$, where xs_t and ys_t are the horizontal and vertical positions of the target, respectively, at time t and xv_t and yv_t are the horizontal and vertical velocities of the target at time t . w_t and r_t are Gaussian-form noise matrices, with the distributions of the covariance matrices being Q_t and R_t . Since the time between any two consecutive frames is short, it can be assumed that moving targets such as vehicles move with uniform linear motion. When occlusion occurs, we use the previous motion state to estimate the motion state under occlusion. Assume that x_t and y_t are given for $0 \leq t \leq T_{occ}$ (the time of occlusion occurrence); then, the likelihood of A , C , Q , and R can be written as follows:

$$\begin{aligned} L(A, C, Q, R | \mathbf{x}, \mathbf{y}) &= p(\mathbf{x}, \mathbf{y} | A, C, Q, R) \\ &= \prod_{t=0}^{T_{occ}} p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{y}_t | \mathbf{x}_t). \end{aligned} \quad (12)$$

This equation can be expanded as follows:

$$\begin{aligned} l(A, C, Q, R | \mathbf{x}, \mathbf{y}) &= \frac{T_{occ}}{2} \log |Q^{-1}| + \frac{T_{occ}+1}{2} \log |R^{-1}| + \beta \\ &\quad - \frac{1}{2} \text{Tr} \left(Q^{-1} \left(\sum_{t=0}^{T_{occ}-1} \mathbf{x}_{t+1} \mathbf{x}_t^T \right. \right. \\ &\quad \left. \left. - \mathbf{x}_{t+1} \mathbf{x}_t^T A^T - A \mathbf{x}_t \mathbf{x}_{t+1}^T + A \mathbf{x}_t \mathbf{x}_t^T A^T \right) \right) \\ &\quad - \frac{1}{2} \text{Tr} \left(R^{-1} \left(\sum_{t=0}^{T_{occ}} \mathbf{y}_t \mathbf{y}_t^T - \mathbf{y}_t \mathbf{x}_t^T C^T \right. \right. \\ &\quad \left. \left. - C \mathbf{x}_t \mathbf{y}_t^T + C \mathbf{x}_t \mathbf{x}_t^T C^T \right) \right) \end{aligned} \quad (13)$$

where Tr is the trace of a matrix and β is a constant. By maximizing $l(A, C, Q, R | \mathbf{x}, \mathbf{y})$ for A , C , Q , and R in turn, we can obtain

$$A = \left(\sum_{t=0}^{T_{occ}-1} \mathbf{x}_{t+1} \mathbf{x}_t^T \right) \left(\sum_{t=0}^{T_{occ}-1} \mathbf{x}_t \mathbf{x}_t^T \right)^{-1} \quad (14)$$

$$C = \left(\sum_{t=0}^{T_{occ}} \mathbf{y}_t \mathbf{x}_t^T \right) \left(\sum_{t=0}^{T_{occ}} \mathbf{x}_t \mathbf{x}_t^T \right)^{-1} \quad (15)$$

$$\begin{aligned} Q &= \frac{1}{T_{occ}} \left(\sum_{t=0}^{T_{occ}-1} \mathbf{x}_{t+1} \mathbf{x}_{t+1}^T - \mathbf{x}_{t+1} \mathbf{x}_t^T A^T \right. \\ &\quad \left. - A \mathbf{x}_t \mathbf{x}_{t+1}^T + A \mathbf{x}_t \mathbf{x}_t^T A^T \right) \end{aligned} \quad (16)$$

$$R = \frac{1}{T_{occ}+1} \left(\sum_{t=0}^{T_{occ}} \mathbf{y}_t \mathbf{y}_t^T - \mathbf{y}_t \mathbf{x}_t^T C^T - C \mathbf{x}_t \mathbf{y}_t^T + C \mathbf{x}_t \mathbf{x}_t^T C^T \right). \quad (17)$$

We represent the motion state estimates as follows:

$$\tilde{\mathbf{x}}_{t+1|t} = A \tilde{\mathbf{x}}_{t|t} \quad (18)$$

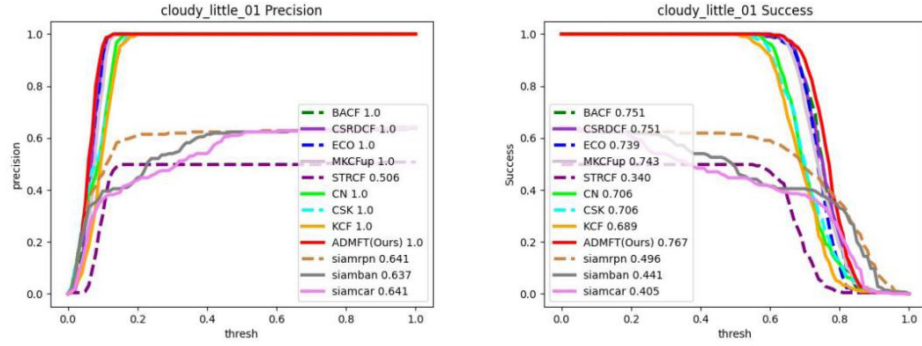


Fig. 6. Precision and success plots show the performance of our ADMFT compared with other CFTs on cloudly_little_01(b).

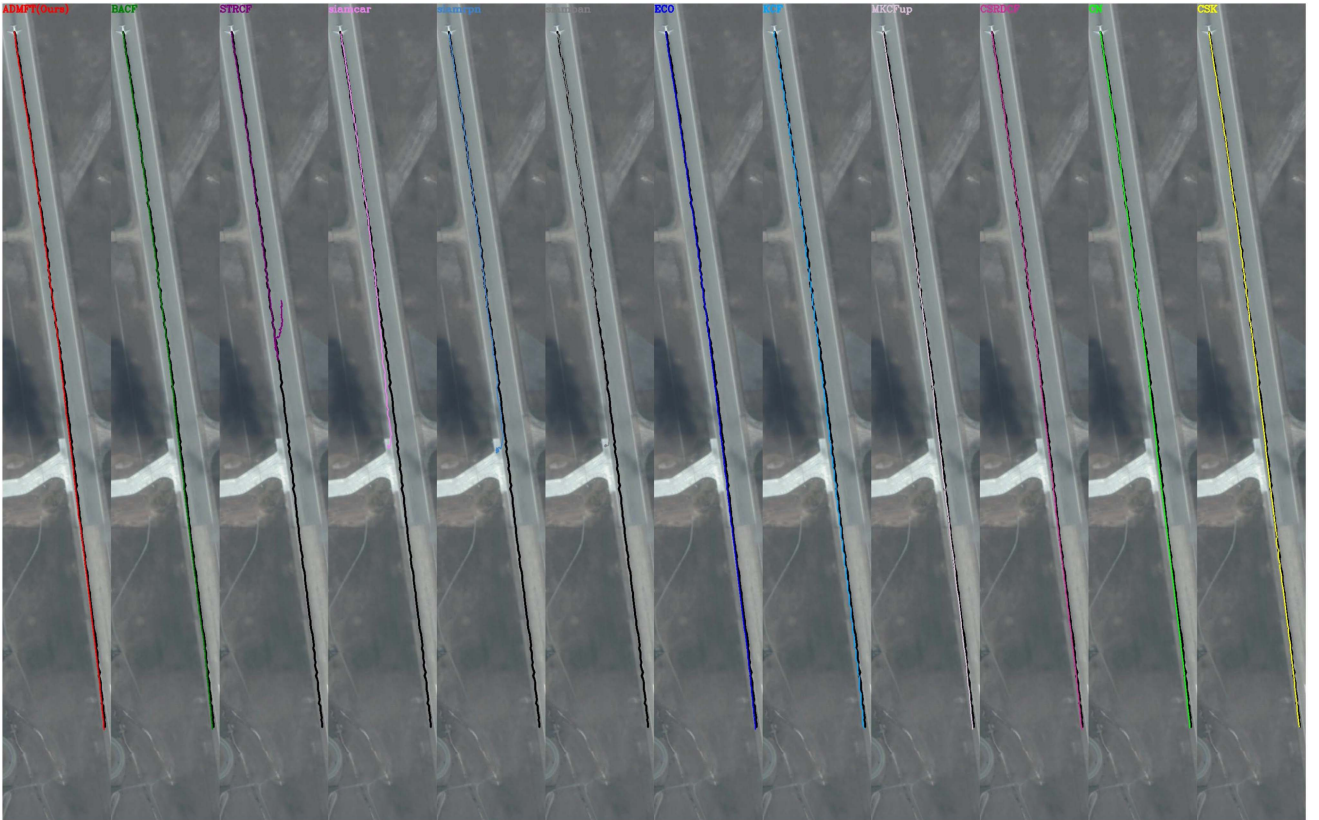


Fig. 7. Tracking performance of our ADMFT compared with other CFTs on cloudly_little_01(b).

$$P_{t+1|t} = AP_{t|t}A^T + Q \quad (19)$$

$$K_{t+1} = P_{t+1}C^T(CP_{t+1}C^T + R)^{-1} \quad (20)$$

$$\tilde{\mathbf{x}}_{t+1|t+1} = \tilde{\mathbf{x}}_{t+1|t} + K_{t+1}(y_{t+1} - C\tilde{\mathbf{x}}_{t+1|t}) \quad (21)$$

$$P_{t+1|t+1} = P_{t+1|t} - K_{t+1}CP_{t+1|t} \quad (22)$$

where $\tilde{\mathbf{x}}_{t+1}$ is the optimal state estimate and K is the KF gain matrix. In the inference stage, the calculation of the KF includes only 10 instances of matrix multiplication, 5 instances of matrix addition, and one calculation of the reciprocal of a 2×2 matrix.

Compared with the computational complexity of the correlation filter, the increase in computational complexity is very small.

The KF offers high accuracy in estimating the target motion state, but the KF is very complex. This filter can converge only when sufficient frames are used to update the filter. To estimate the motion of a moving target before KF convergence, we propose a method of simulating the real motion state by using an assumed motion state. We can assume that the target moves in a uniform, straight line over a short time, even if the target is in a state of turning, stopping due to an emergency, or accelerating. Based on this assumption, the speed of the moving target in the current frame can be estimated from the average

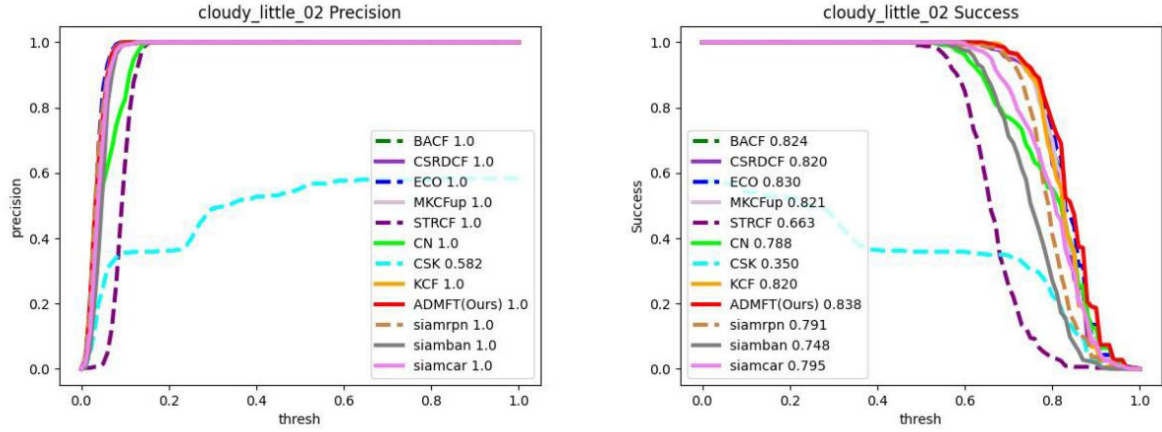


Fig. 8. Performance of our ADMFT compared with other CFTs on cloudy_little_02(c).

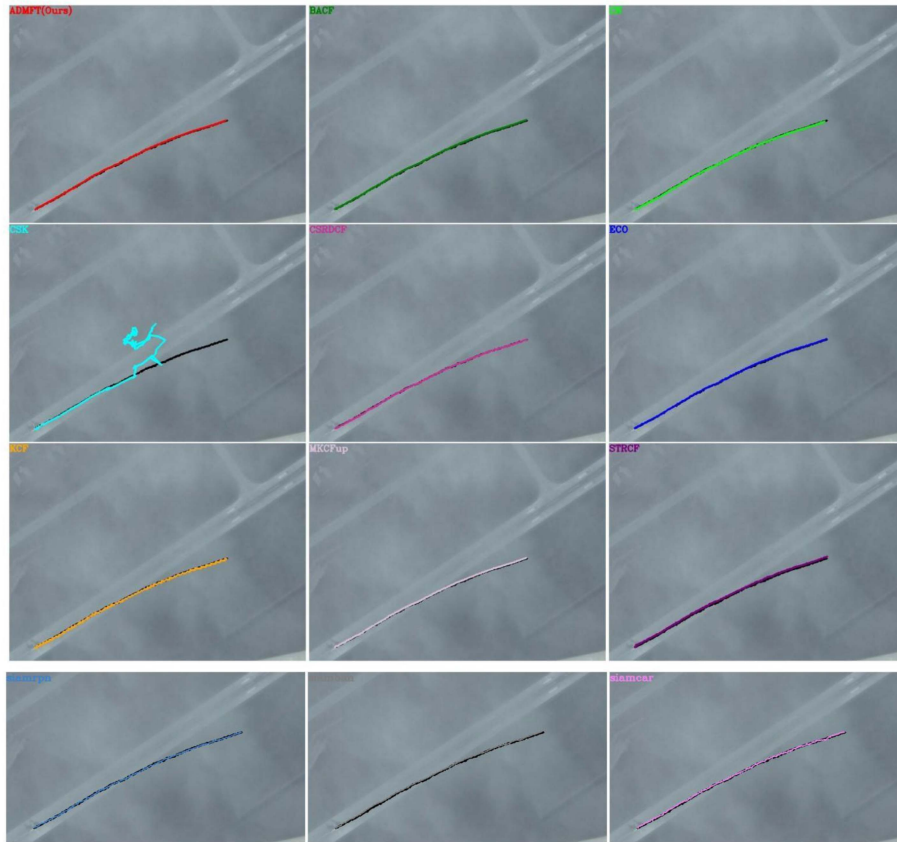


Fig. 9. Tracking performance of our ADMFT compared with other CFTs on cloudy_little_02(c).

displacement with respect to the previous frame. The moving target's position in the current frame can be estimated by using the speed and position of the moving target in the previous frame. Therefore, the values can be estimated as described in the following equations:

$$\Delta x_{t-1} = \frac{1}{n} \sum_{i=1}^n (x_{t-i} - x_{t-i-1}) \quad (23)$$

$$\Delta x_{t-1} = \frac{1}{n} \sum_{i=1}^n (x_{t-i} - x_{t-i-1}) \quad (24)$$

$$P_t = \phi S_{t-1} n \quad (25)$$

where S_{t-1} is the state vector of the target at time $t-1$, $S_{t-1} = (x_{t-1}, y_{t-1}, \Delta x_{t-1}, \Delta y_{t-1})^\top$; $P_t = (x_t, y_t)^\top$ is the position vector of the target at time t ; and ϕ is a transfer matrix,

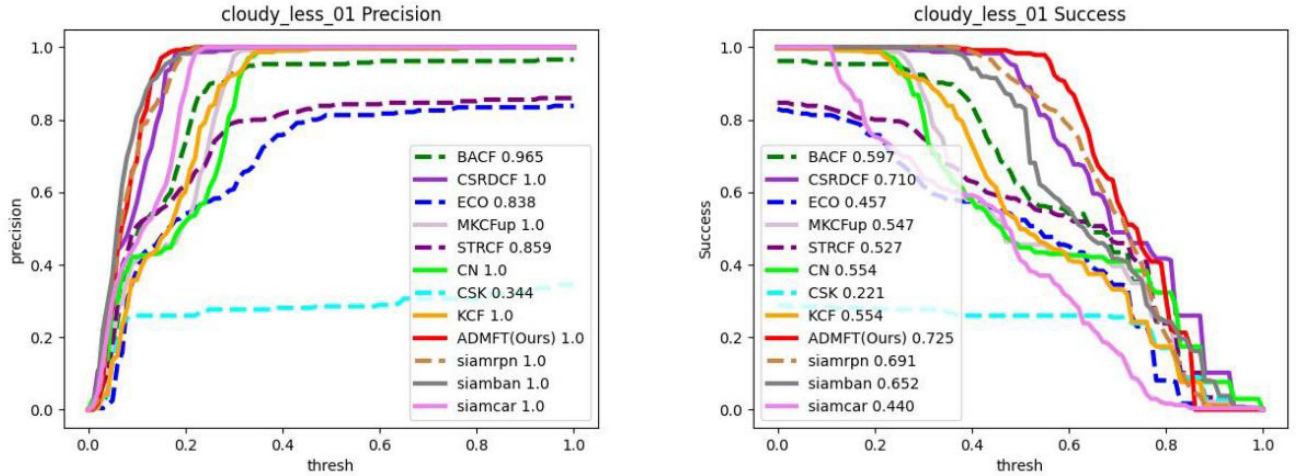


Fig. 10. Precision and success plots showing the performance of our ADMFT compared with other CFTs on cloudy_less_01(a).

which can be written as follows:

$$\phi = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad (26)$$

n is the number of frames used for estimation. If $n > Num_{confident}$, then the parameters can be ensured by the EM algorithm. Using a frame-based parameter selection strategy can help us obtain the most suitable parameters.

E. Tracker Fusion Based on the APCE

To combine the result of motion state prediction with the result of the correlation filter, we propose a combination strategy based on multipeak matching Fig. 3. First, to judge whether the target is occluded, we calculate the APCE, the degree of fluctuation of the response diagram, and the confidence level of the detected target

$$APCE = \frac{|F_{\max} - F_{\min}|^2}{\text{mean} \left(\sum_{w,h} (F_{w,h} - F_{\min})^2 \right)} \quad (27)$$

where F_{\max} , F_{\min} , and $F_{w,h}$ represent the maximum and minimum response values and the response at position (w, h) , respectively.

The current APCE value will be significantly reduced relative to the historical mean when the moving target is blocked, changed, blurred, or lost. Consequently, the current response diagram will oscillate and exhibit a multipeak phenomenon. At this time, confidence in the target center position is considered to be low. Generally, when multipeak oscillation occurs, the response value at the center of the target will also be significantly reduced, that is, the peak F_{\max} will generally be lower than the peak without interference. It can be seen that $F_{y_{\max}}$ reflects the confidence in the target center position from the local part of the response diagram, whereas the APCE reflects its confidence from the overall response diagram. Accordingly, higher confidence can be achieved by combining the two in the current frame t only when the y_{\max} and APCE values are in a certain proportion, represented by α and β . In this article, we set α to 0.5 and β to 0.3. α and β can be adjusted in the

reality; for example, if the movement is complex, the KF should be suppressed, and β should be downwards. If the object has confusing features, α should be downwards. When the historical mean value is exceeded, it is considered that the target center position has high confidence, that is, two conditions need to be met simultaneously.

$$\begin{cases} F_{\max} \geq \alpha \cdot \frac{1}{t-1} \sum_{i=1}^{t-1} F_{i,\max}, & F_{i,\max} \in P_y \\ APCE \leq \beta \cdot \frac{1}{t-1} \sum_{j=1}^{t-1} APCE_j, & APCE_j \in P_E \end{cases} \quad (28)$$

Each time $F_{i,\max}$ and $APCE_j$ are calculated, the values will be saved in the corresponding sets P_y and P_E as a pair of historical values for the next judgment. To reduce the number of calculations in the algorithm, we do not correct the position in each frame; however, when there is multimodal oscillation in the current frame t and the target center position may be judged incorrectly, that is, when the F_{\max} and APCE values do not meet the conditions for high-confidence detection, we introduce motion information to correct the position. At this time, we fuse the motion information with the relevant filter information in a weighted manner, replace the current prediction result with the fused result, and update the filter model with the current result

$$\begin{aligned} position_{real} = & \frac{1}{2(APCE/APCE_1)} position_{CF} \\ & + \left(1 - \frac{1}{2(APCE/APCE_1)} \right) position_{KF}. \end{aligned} \quad (29)$$

III. EXPERIMENT AND ANALYSIS

A. Performance Measures

To evaluate the performance of our proposed algorithm, we use one-pass evaluation (OPE) as the evaluation protocol. This protocol was proposed for the OTB-2013 benchmark [37]. OPE relies on two plots, which are called the accuracy plot and the success plot. The accuracy plot shows the percentage accuracy of the predicted positions relative to the ground-truth values

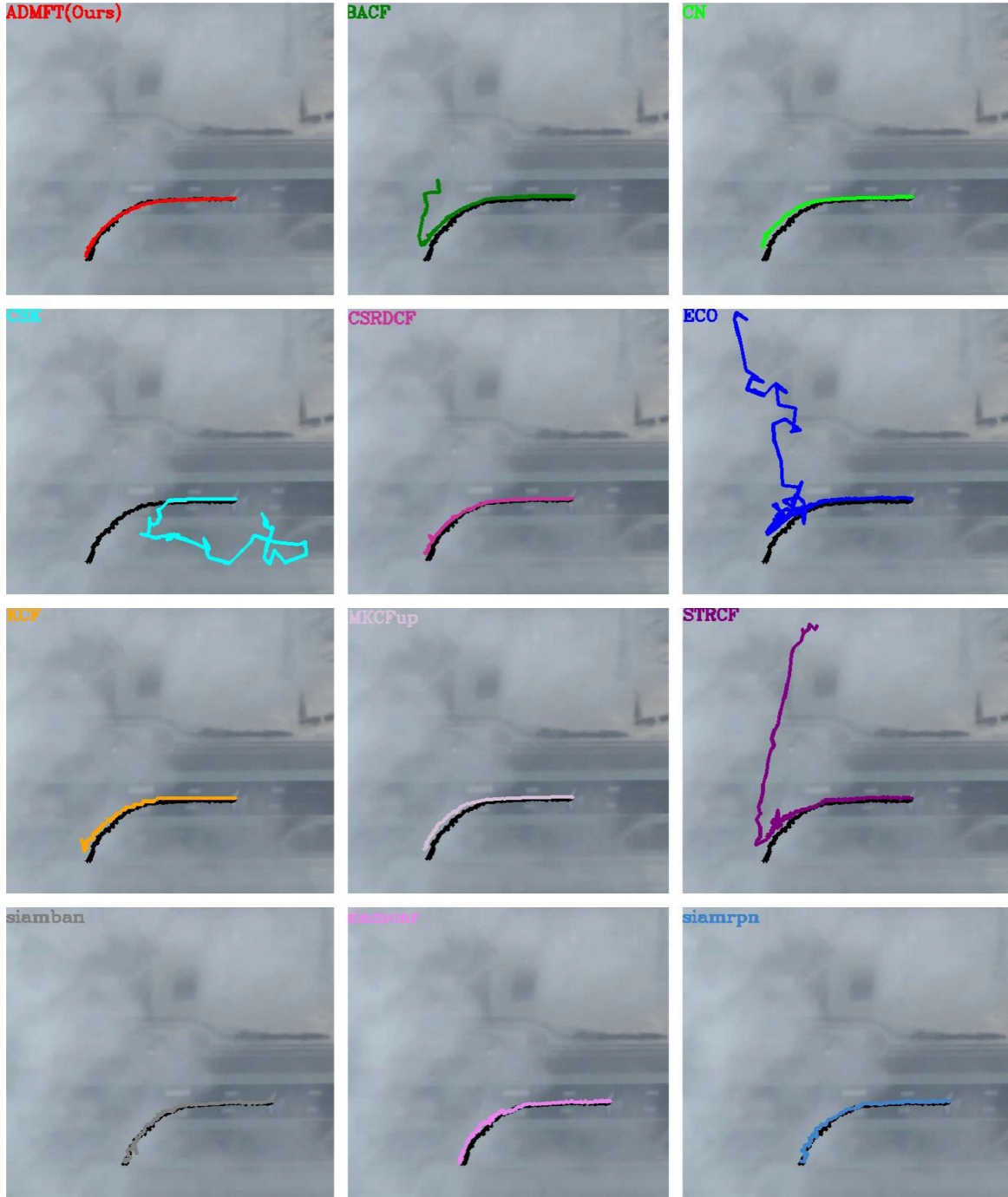


Fig. 11. Tracking performance of our ADMFT compared with other CFTs on cloudy_less_01(a).

at different thresholds. The success plot represents an average overlap measure [38]. Given the result bounding box b_r and the ground-truth bounding box b_g , the success score (success) is calculated as follows:

$$\text{Success}_s = \frac{S\{b_r \cap b_g\}}{S\{b_r \cup b_g\}} \quad (30)$$

where \cap represents the intersection of two regions, \cup represents the union of two regions, and s represents the area of a region.

The AUC is defined as the area under the receiver operating characteristic curve.

To evaluate the performance of our proposed tracker, we also adopt the center location error (CLE), which is the average Euclidean distance between the center location of the estimated target and the ground-truth target center location.

Similar to BACF, we adopt the regularization factor λ is set to 0.01, and η is set to 10^{-4} by experience in Fig. 4. For the ADMM optimization, the number of iterations and the penalty factor μ are set to 2 and 1. The penalty factor at iteration $i+1$ is



Fig. 12. Tracking performance of our ADMFT compared with other CFTs on cloudy_less_01(a). The red boxes represent the ADMFT labels, while the black boxes indicate positive sample labels.

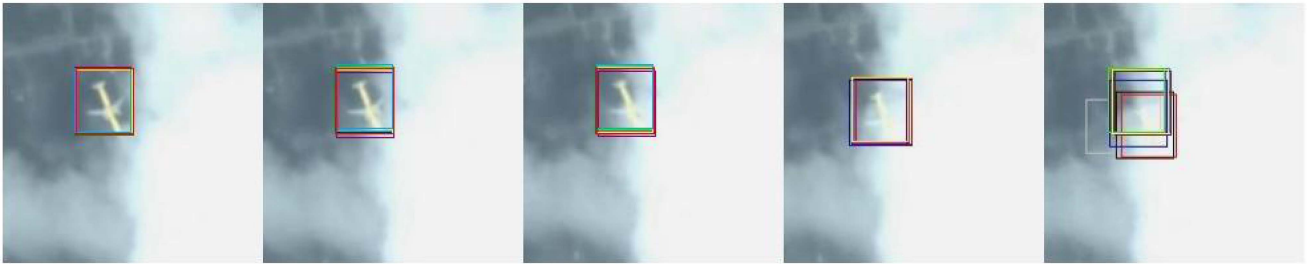


Fig. 13. Tracking performance of our ADMFT compared with other CFTs on cloudy_more_01(d). The red boxes represent the ADMFT labels, while the black boxes indicate positive sample labels.

updated by $\mu_{i+1} = \min(\mu_{max}, \beta\mu_i)$, where $\mu = 1$, $\beta = 0.1$, and $\mu_{max} = 10^3$.

For cloudy data, drift often occurs in the last frame; therefore, when estimating, we focus on its trajectory and drift degree.

B. Quantitative Evaluation

In our experiment, to ensure fair comparisons, a list of the most advanced trackers of the same type, i.e., the top-performing trackers that function similarly to the proposed ADMFT, was compiled as the set of trackers considered for comparison. For this purpose, the efficient convolution operator with handcrafted features (ECO) was selected from among trackers based on handcrafted features as a representative tracking model with good performance. The circulant structure of the tracking-by-detection with kernels (CSK) algorithm also achieves good performance by introducing the kernel technique and ridge regression into MOSSE. The CN approach is a good method to obtain color features. It achieves good performance on images with obvious color contrast. The CSR-DCF algorithm combines spatial reliability and channel reliability methods in image segmentation to more accurately select the effective target tracking area. MKCFup significantly reduces the detrimental mutual interference among different particles. The STRCF method constrains the effective scope of the filter template to overcome the boundary effect. The BACF is the basic method on which our improved algorithm is based. SiamRPN combines the twin network in tracking and the regional recommendation network in detection: the twin network can adapt to the tracking target so that the algorithm can use the information of the tracked

target to complete the initialization of the detector. The regional recommendation network allows the algorithm to predict the target location more accurately. SiamBAN adopts the anchor-free strategy, which does not preset the size of the anchor box so that the box has more powerful degrees of freedom. SiamCAR has an additional centrality branch to better determine the location of the target center point. We compare our improved method with the above-mentioned advanced methods.

Our model ensures high accuracy and a good antidrift ability while maintaining high operation efficiency. The purpose of our experiments is to verify that under a variety of different target states, it achieves an AUC that is higher than those of other trackers. It can adapt to complex conditions. The performance and programming language specifications are given in the following table.

Moreover, the frame rate of our method reaches 83.84 fps, which is only 0.13 times slower than that of the BACF before improvement. Our experimental environment is as follows: the algorithms are executed on a Windows 10 system with an Intel(R) Core(TM) i7-9700 CPU and 16 GB of RAM. The performance is given in Table I.

Table III shows the CLE results (in pixels) achieved by our proposed tracker and the other approaches on 20 sequences. Our model achieves satisfactory performance among the compared methods, with a small CLE. The results show that the ADMFT is robust on video sequences with fast motion, occlusion, and deformation of the tracking targets.

The ADMFT uses a correlation filter for position estimation. When occlusion occurs, its influence will be corrected based on the motion state. Its average CLE is 4.363 pixels, which is



Fig. 14. Tracking performance of our ADMFT on videos occluded by white blocks with transparency 0.8. According to the results, the performance of the ADMFT can remain stable even under a high degree of occlusion.

greatly superior to the results for the other correlation-filter-based trackers (CFTs). These results show that the proposed combination of the KF and a correlation filter is quite effective for position estimation.

Furthermore, a frame-by-frame comparison of the CLEs on the 20 sequences is shown in Fig. 5. The vertical axis represents the CLE, while the horizontal axis represents the frame number in the image sequence. The proposed ADMFT produces favorable results for the 20 targets. Compared with the other eight methods, the ADMFT handles cloud occlusion well. In the complex_background_01–03 videos, the targets suffer from background clustering, and the ADMFT achieves better performance. On videos showing aircraft moving at regular and slow speeds, almost all trackers perform well. In videos with aircraft rotation (rotation_01–04), the target appearance changes significantly with deformation and illumination variation, and most trackers fail to track the target at the beginning of the image sequences. However, our method succeeds in estimating the position of the target.

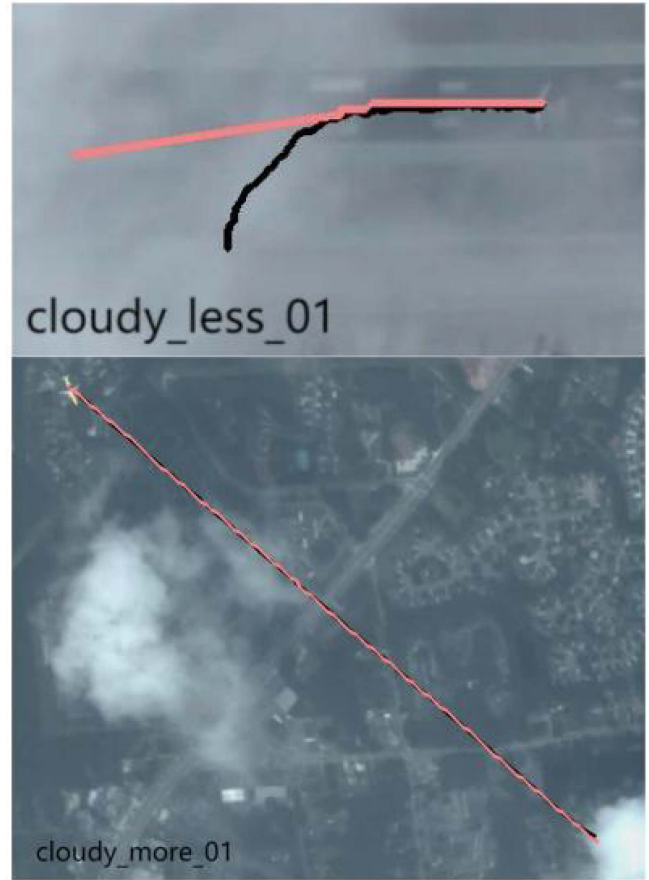


Fig. 15. Trajectories of CFME for the sequence cloudy_more_01(d) and cloudy_less_01(a).

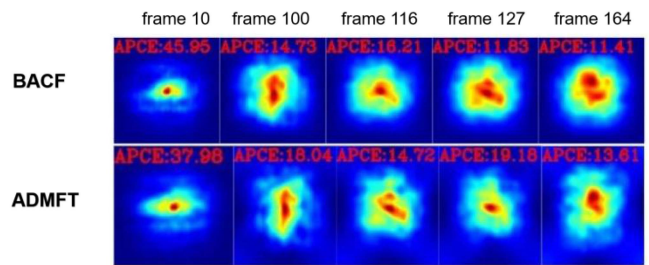


Fig. 16. Pictorial representation of ADMFT and BACF updates for the sequence cloudy_less_01(a) (frames #10, #100, #116, #127, and #164).



Fig. 17. Comparison of the influence of the ADMFT motion estimator on cloudy_less_01(a).

It can be seen from the above-mentioned findings that our algorithm can track targets more accurately under normal conditions than the BACF method before improvement. Moreover, the ADMFT has an improved tracking ability in cases of occlusion, especially in terms of the balance between resisting occlusion and performing well in general situations. Consequently, our method is far superior to the existing methods. Occlusion is widely encountered in remote sensing images. Because general methods do not have a high ability to deal with the problem of cloud occlusion, even though targets in cloud-occluded image data can be observed by the human eye, these data have often been directly abandoned in previous practice. Under the previously available methods, such cloud-obscured data have not been well utilized, and the quantity of data discarded for this reason is considerable. Therefore, the ability to effectively use these data is expected to be of great significance in research and practical applications.

C. Antidrift Ability Evaluation

From a comparison of the CLEs for *cloudy_little_01* and *cloud_little_02* in Fig. 1(b) and (c), it can be seen that the ADMFT achieves constant, stable tracking in the case of thin clouds. To more intuitively see whether the target suffers from a position offset under cloud occlusion, we comprehensively consider the accuracy and success rate results for the *cloudy_little_01*(b) video and observe the corresponding trajectories in Figs. 6 and 7. It is found that in the case of thin cloud occlusion, although the original method is only slightly disturbed, our model can still more effectively resist the interference and achieve better performance.

We also comprehensively consider the accuracy and success rate results for the *cloudy_little_02* video and observe the corresponding trajectories in Figs. 8 and 9. Again, it is found that in the case of thin cloud occlusion, the original method is only slightly disturbed, but our model still performs better. However, the Siam-trackers model drifts in this sequence, and the Siam-tracker cannot deal with sudden changes in two adjacent frames.

In the case of medium cloud cover, some methods fail badly, and the trajectory offset is serious. All other methods show performance fluctuations to varying degrees, whereas our method is stable. The CN approach uses color features. The CSR-DCF algorithm combines CNN features to better express the target characteristics. MKL achieves a stronger distinguishing ability than KCF in MKCFup. The results of these trackers show slight deviations. When the target is initially occluded, these methods can obtain features that are not occluded to continue tracking the target. However, when the target is occluded and then exposed, the previously occluded part is often exposed first, and because the features of this part have not been learned for some time, they cannot well represent the target. Improving the feature extraction capabilities is effective for transient cloud occlusion but cannot overcome the influence of the occlusion caused by dense clouds. However, feature extraction over a certain period of time can solve this problem. In the *cloudy_less_01*(a) video, the STRCF results drift because of rotation. The STRCF method does not

intensively extract negative examples from the background in real time rather than focusing on moving foreground patches. Introducing the time consistency constraint into the BACF algorithm endows it with good tracking ability that can adapt to complex scenes. We comprehensively consider the precision and success rate results for *cloudy_less_01*(a) and observe the corresponding trajectories in Figs. 10–12.

As seen in the above-mentioned figures, in cases of occlusion, our method effectively limits the model drift compared with the original BACF algorithm. Through incorporating motion features, the prediction of the target position is improved.

In the case of thick cloud occlusion, the model update process gradually causes the model to prioritize cloud features over real target features; however, our model is less polluted when the target enters a cloud. In this case, the target becomes seriously occluded within five frames. It can be seen that other methods suffer from model drift, resulting in a gradually increasing target offset. Due to improper updates, the CN, CSR-DCF, and MKCFup methods also exhibit model drift. In contrast, the correction process based on the motion state and time regularization helps our method predict the target's position more accurately. When the clouds are thick and the aircraft body is severely blocked, our method can still closely track the target in Fig. 13, while other methods suffer from offset.

We used white blocks to simulate clouds of different transparencies and performed tests using the simulated data to verify the performance of the ADMFT. We selected several targets with good tracking performance for each tracker as the research targets and used white blocks with transparency values of 0.2, 0.5, and 0.8 to occlude half of each target to test the performance of the trackers. In this verification test, our method can still achieve stable tracking under occlusion with a transparency value of 0.8, as shown in Fig. 14.

In CFME, a method using short-term motion state prediction to replace the prediction value in the case of model drift is proposed. After testing in four cloud occlusion videos, the CFME method is an efficient tracking method that performs well in the case of a simple motion state. However, after verification in the video, the CFME method cannot track the target normally in the case of occlusion with a complex motion state in Fig. 15.

D. Ablation Experiment

The temporal regularization strategy can improve the antidrift ability of a tracker. The features extracted over a period of time are more stable than those extracted from only one frame. Here, we compare the peak distributions before and after the introduction of temporal regularization. It can be seen in the peak diagram that the peak fluctuation with our method is slight in Fig. 16, resulting in a better anti-interference effect.

The motion estimator also helps to correct the tracking results. To test its contribution, we used only the antidrift BACF with temporal regularization on *cloudy_less_01*(a). The tracking result is shown in the left part of Fig. 17, and the tracker fusion result is shown in the right part of the figure.

As seen in the above-mentioned comparisons of the influence of the motion estimator and the temporal regularization strategy, these two model components both improve the tracker's ability. The motion estimator incorporates motion information, which is always stable in aircraft tracking. The temporal regularization strategy can help the tracker learn stable features over a certain period of time.

The features of the target will change upon entering a cloud. During this process, the target states will serve as our essential reference values for updating the model. Model drift often occurs before and after the target enters the cloud. The greater the degree of occlusion is the more pronounced the model drift. Some tracker models will drift during the update process. Our method restricts the model update by means of L2 regularization and uses the motion state to correct the trajectory. Thus, tracking failure caused by drift is successfully avoided.

IV. DISCUSSION

The ADMFT was written in Python and implemented on a PC with a 3.00 GHz CPU and 16 GB of memory. Our experiments show that the developed ADMFT can process video data at more than 83 fps. In summary, the remote sensing image features remain unchanged, and the main model drift is caused by cloud occlusion. For this type of drift, we successfully solve the drift problem by learning features over a time series and correcting the target model. Moreover, in the general process of aircraft flight, the proposed method can effectively obtain the features of the target aircraft to address the aircraft tracking problem in complex scenes. The ADMFT can be widely used as an aircraft tracker for satellite video. However, the ADMFT only works for aircraft, so we did not consider the scaling problem or target labeling with rotation rectangles. Moreover, the tracking efficiency can be further improved by implementing the algorithm in a parallel processing system.

V. CONCLUSION

This article describes an effective method for tracking moving aircraft in satellite videos. For video tracking of moving aircraft, we design a BACF based on temporal regularization to address the model drift caused by cloud occlusion and use the ADMM to speed up the solution process. In addition, the KF helps improve detection accuracy. The APCE is used to judge whether the trajectory needs to be corrected. The motion information can correct the trajectories of objects in simple environment and temporal regularization can help resist the influence of occlusion. We tested our method on satellite videos by tracking 20 moving aircraft of different sizes and in different dynamic states. The proposed ADMFT algorithm achieved better tracking accuracy than the most advanced existing algorithms. Specifically, its tracking effect for targets under cloud cover is better than that of other advanced methods. The ADMFT exhibits good robustness and can handle various remote sensing video environments for airplanes. In future work, we will further test the tracking capabilities for other remote sensing targets, such as ships and ground vehicles.

REFERENCES

- [1] C. Hu et al., "MODIS detects oil spills in Lake Maracaibo, Venezuela," *Eos Trans. Amer. Geophysical Union*, vol. 84, no. 33, pp. 313–319, 2003.
- [2] J. Yang et al., "Research on natural disaster emergency monitoring system," *Int. J. Sensor Netw.*, vol. 32, no. 4, pp. 218–229, 2020.
- [3] R. Danovaro et al., "Implementing and innovating marine monitoring approaches for assessing marine environmental status," *Front. Mar. Sci.*, vol. 3, 2016, Art. no. 213.
- [4] A. K. Skidmore et al., "Priority list of biodiversity metrics to observe from space," *Nature Ecol. Evol.*, vol. 5, no. 7, pp. 896–906, 2021.
- [5] G. Kopsiaftis and K. Karantzos, "Vehicle detection and traffic density monitoring from very high resolution satellite video data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2015, pp. 1881–1884.
- [6] Y. K. Seong, Y.-H. Choi, J.-A. Park, and T. S. Choi, "Scene change detection in the hard disk drive embedded digital satellite receiver for video indexing," in *Proc. Dig. Tech. Papers Int. Conf. Consum. Electron.*, 2002, pp. 210–211.
- [7] F. Shi et al., "A method to detect and track moving airplanes from a satellite video," *Remote Sens.*, vol. 12, no. 15, 2020, Art. no. 2390.
- [8] D. Yuan et al., "Learning adaptive spatial-temporal context-aware correlation filters for UAV tracking," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 18, no. 3, pp. 1–18, 2022.
- [9] O. Barnich and M. Van Droogenbroeck, "ViBe: A powerful random technique to estimate the background in video sequences," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 945–948.
- [10] Y. Cui, B. Hou, Q. Wu, B. Ren, S. Wang, and L. Jiao, "Remote sensing object tracking with deep reinforcement learning under occlusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2021.
- [11] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 3645–3649.
- [12] S. Xuan, S. Li, M. Han, X. Wan, and G.-S. Xia, "Object tracking in satellite videos by improved correlation filters with motion estimations," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1074–1086, Feb. 2020.
- [13] X. Zhang et al., "Visual object tracking by correlation filters and online learning," *ISPRS J. Photogrammetry Remote Sens.*, vol. 140, pp. 77–89, 2018.
- [14] F. Shi et al., "Detecting and tracking moving airplanes from space based on normalized frame difference labeling and improved similarity measures," *Remote Sens.*, vol. 12, no. 21, 2020, Art. no. 3589.
- [15] 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.
- [16] J. F. Henriques et al., "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 702–715.
- [17] 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.
- [18] 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.
- [19] M. Danelljan et al., "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–5.
- [20] M. Danelljan et al., "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 472–488.
- [21] A. Lukežić et al., "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6309–6318.
- [22] M. Tang and J. Feng, "Multi-kernel correlation filter for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3038–3046.
- [23] M. Tang, B. Yu, F. Zhang, and J. Wang, "High-speed tracking with multi-kernel correlation filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4874–4883.
- [24] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4310–4318.
- [25] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4904–4913.
- [26] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "Eco: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6931–6939.

- [27] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1144–1152.
- [28] 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.
- [29] D. Yuan et al., "Robust thermal infrared tracking via an adaptively multi-feature fusion model," *Neural Comput. Appl.*, vol. 35, pp. 3423–3434, 2023.
- [30] X. Shu et al., "Adaptive weight part-based convolutional network for person re-identification," *Multimedia Tools Appl.*, vol. 79, pp. 23617–23632, 2020.
- [31] D. Yuan, X. Chang, P.-Y. Huang, Q. Liu, and Z. He, "Self-supervised deep correlation tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 976–985, 2021.
- [32] L. Bertinetto et al., "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 850–865.
- [33] Z. Chen, B. Zhong, G. Li, S. Zhang, and R. Ji, "Siamese box adaptive network for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6667–6676.
- [34] D. Guo, J. Wang, Y. Cui, Z. Wang, and S. Chen, "SiamCAR: Siamese fully convolutional classification and regression for visual tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6268–6276.
- [35] S. Boyd, N. Parikh, and E. Chu, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. & Trends Mach. Learn.*, USA, pp. 1–122, Mar. 2014.
- [36] B. Schwartz, S. Gannot, and E. A. P. Habets, "Online speech dereverberation using Kalman filter and EM algorithm," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 2, pp. 394–406, Feb. 2015.
- [37] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.
- [38] L. Čehovin, M. Kristan, and A. Leonardis, "Is my new tracker really better than yours?," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2014, pp. 540–547.