Single Object Tracking in Satellite Videos: A Correlation Filter-Based Dual-Flow Tracker

Yuzeng Chen, *Student Member, IEEE*, Yuqi Tang, *Member, IEEE*, Zhiyong Yin, Te Han, *Student Member, IEEE*, Bin Zou, and Huihui Feng, *Member, IEEE*

***Abstract*—Satellite video (SV) can acquire rich spatiotemporal information on the earth. Single object tracking (SOT) in SVs enables the continuous acquisition of the position and range of a specific object, expanding the field of remote-sensing applications. In SVs, objects are small with limited features and vulnerable to tracking drift. In this article, a correlation filter based dual-flow (DF) tracker is proposed to explore how the hybridization of spatial–spectral feature fusion and motion model can boost tracking. To represent small objects, the DF adaptively fuses complementary features using a state-aware indicator in feature flow. In motion flow, the indicator perceives the confidence of the feature flow. A dual-mode prediction model is then constructed to simulate the object's motion pattern and cooperate linear and nonlinear motion patterns to implement SOT in SVs. The ablation experiments demonstrate that the DF contributes to tracking. Experimental comparisons on 14 real SVs captured by the Jilin-1 satellite constellation show that DF achieves optimal performance with an area under the curve of 0.912 in the precision plot, 0.700 in the success plot, and a speed of 155.2 frames per second. This work would encourage the development of remote-sensing ground surveillance.**

***Index Terms*—Correlation filter (CF), motion model, satellite video (SV), state-aware indicator (SAI), single object tracking (SOT).**

I. Introduction

Satellite video (SV) has become a valuable surface observation data, which provides a wealth of static and dynamic information on specific areas [1]. In 2013, the SkySat-1 video satellite was launched by Skybox Imaging, marking a milestone in the expansion of remote-sensing observation means from imagery to video. SkySat-1 can capture panchromatic video with a ground sample distance (GSD) of 1.1 m and a frame rate of 30 frames per second (FPS). Subsequently, the International Space Station (ISS) released an ultrahigh definition video acquired over Vancouver in 2015. The video was captured by high-resolution camera, Iris, installed on ISS. Iris can shoot RGB SVs with a GSD of 1.0 m at 3 FPS. From 2015 to now, the Jilin-1 satellite constellation of China Changchun Satellite Technology Co., Ltd. was launched one by one. Currently, Jilin-1 is capable of shooting 30 FPS RGB SVs with a GSD of 0.92 m and the coverage of each frame reaching 11.0 km × 4.6 km. The emergence of this advanced data drives the development of the remote-sensing field in the visual community. Single object tracking (SOT) in SVs, served as one of the most fundamental tasks, has prosperous application prospects in dynamic traffic surveillance and analysis [2], ocean monitoring [3], environmental monitoring [4], stereo mapping [5], and super-resolution [6]. SOT in SVs determines the position and range of an object in subsequent frames when its initial state is available only in the first frame. In contrast with SOT in natural videos, it encounters several difficulties, such as follows.

1) Limited features: SVs usually contain three bands (red/green/blue), so the spectral features of the object are limited. Moreover, due to the low resolution, the object occupies few pixels and has few spatial features such as structure, which can lead to difficulties in the accurate identification and positioning.
2) Abnormal states: SVs are filmed by satellite platforms with high-speed moving, small objects accompanied by nonstationary background are susceptible to abnormal states (e.g., occlusion, rotation, background clutter, overtaking, and motion blur), which may cause tracking drift.

To overcome these issues, researchers have conducted research works on SOT in SVs, which can be classified into detection-based [7]–[10] and discriminative methods.

Detection-based methods usually use interframe motion information to detect and track the moving object. Discriminative methods include deep learning based [11]–[14] and correlation filter (CF) based [1], [2], [15]–[19]. Deep learning based algorithms extract the convolutional features of the object to determine its position, and this can increase the computational burden and slow down the tracking speed. CF-based algorithms start by training a filter with a predefined response on all training samples. The correlation operation is converted to element multiplication by fast Fourier transform (FFT) followed by Inverse FFT, resulting in a reduction in storage and computation of several orders of magnitude [20]. It then uses the pretrained filter to locate the object. Furthermore, the filter is updated in subsequent frames. The different methods for SOT in SVs will be elaborated upon in related work (see Section II-B). CF, one of

Manuscript received 28 March 2022; revised 29 May 2022 and 13 June 2022; accepted 19 June 2022. Date of publication 22 June 2022; date of current version 24 August 2022. This work was supported by the National Natural Science Foundation of China under Grant 41971313. *(Corresponding author: Yuqi Tang.)*

Yuzeng Chen, Zhiyong Yin, and Te Han are with the School of Geosciences and Info-Physics, Central South University, Changsha 410012, China (e-mail: yzchen1006@163.com; ryan_yinzy@163.com; tehanrs@163.com).

Yuqi Tang, Bin Zou, and Huihui Feng are with the School of Geosciences and Info-Physics, Central South University, Changsha 410012, China, and also with the Key Laboratory of Metallogenic Prediction of Nonferrous Metals and Geological Environment Monitoring, Central South University, Ministry of Education, Changsha 410083, China (e-mail: yqtang@csu.edu.cn; 210010@csu.edu.cn; hhfeng@csu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3185328

the best discriminative methods, has been successfully applied to SVs [1], [2], [15]–[19]. It uses cyclic shift to construct training samples and converts the correlation operation into element multiplication by FFT, thereby improving accuracy and speed. Despite achieving competitive performance, single hand-crafted feature, such as histogram of oriented gradients (HOG) [21], may be limited in the representations of objects in SVs. However, the local spectrum inside an object region facilitates tracking [22]. Meanwhile, CFs [20], [22]–[24] update the template without evaluating tracking confidence and cause the template contaminated. The tracking drift is an inherent drawback of CFs, resulting in the sample drifting away from the object. Several methods [25]–[27] have been used to overcome tracking drift, but at the cost of high time consumption. These methods ignore a motion model that may be a simple and efficient means. To address problems of limited feature representation and tracking drift, we propose a CF-based dual-flow (DF) tracker. The proposed approach has the following contributions.

1) A CF-based DF tracker that cooperates spatial–spectral features and adaptive motion model is proposed for SOT in SVs. Complementary features representing texture and spectrum of objects are fused to enhance the representation in feature flow. In motion flow, a dual-mode prediction model is constructed synthesizing the linear and nonlinear motion patterns to prevent tracking drift.
2) A state-aware indicator (SAI) is defined to perceive the confidence of tracking. It achieves the adaptive selection of feature weights in feature flow while sensing the abnormal states in motion flow.
3) Ablation experiments are conducted to verify the necessity and performance of the above works for tracking in SVs. Extensive comparisons with 13 representative trackers are used to prove the superiority of the proposed method.

The rest of this article is organized as follows. Related work on SOT is presented in Section II. Section III presents the general tracking framework of Staple [23]. The proposed approach is detailed in Section IV. Section V describes the experiments conducted on SVs. Finally, Section VI concludes this article.

II. Related Work

A. Single Object Tracking

SOT is an open and fascinating field with a wide range of applications such as in surveillance [28], self-driving [29], sports competitions [30], and atmospheric motion [31]. However, many factors constrain the effects of SOT, such as occlusion and deformation, requiring a more robust and accurate tracker. Currently, SOT can be divided into generative and discriminative methods. Generative methods construct a model to represent the object and find a region that is most similar in the search region. Typical methods include mean shift [32], particle filters [33], and sparse representation [34]. How to find efficient features to represent the object is a challenge that has a significant impact on the tracking accuracy and speed. Moreover, generative methods only consider the characteristics of the object itself, which makes it easy for the sample to drift away from the object. Discriminative methods have become a mainstream research issue. Both objects and background regions are used to train the classifier, which makes such trackers more discriminative. Discriminative methods include two frameworks: deep learning based and CF based. CNN-SVM [35], one of the earliest deep learning based algorithms, combines a convolutional neural network (CNN) with a support vector machine (SVM) [36] for SOT. MDNet [37] uses large amounts of data to pretrain the CNNs offline and then fine-tune it online to adapt to changes in objects during SOT. These methods have difficulty in running real time due to their in-depth structure and online fine-tuning. To solve these problems, Bertinetto et al. [38] proposed SiamFC, which uses fully convolutional Siamese network architecture trained end-to-end for SOT. The CNNs are trained offline to solve the similarity learning process and avoid fine-tuning online. In this way, the SiamFC has a good balance of accuracy and speed, earning it the attention of many researchers. Subsequently, many trackers have been proposed such as SiamRPN++ [39] and SiamMask [40]. Although these methods [39]–[44] have performed good performance in natural videos, it remains unknown whether they would work well in SVs. CF-based trackers have emerged as a highlight since the MOSSE [45] was first proposed. CSK [46] introduces a circulant matrix and kernel trick based on MOSSE. KCF [20] then extends the CSK to use multichannel features and introduces multiple kernel functions. However, the scale variation of the object was an unresolved issue until the release of DSST [24] and SAMF [47], which adopt a scale filter to address scale change. To obtain better performance, convolutional features are also used for CFs, but the speed is inferior, such as in C-COT [48] and ECO [49]. In general, tracking results achieved by a single feature are not satisfactory. Thus, the Staple [23] combines the HOG and GCS features for tracking. The GFS-DCF [50] fuses convolutional features, HOG and CN. These trackers fuse multiple features for SOT and get improvement in performance. However, tracking drift is a hassle, and current solutions [25]–[27] mostly come at the expense of running speed.

B. SOT in SVs

Some methods are developed for SOT in SVs that include detection-based and discriminative. For detection-based methods, Du et al. [9] propose a multiframe optical flow tracker that combines the motion feature (optical flow), integral image, and multiframe difference for SOT. However, the performance of detection-based methods is far from satisfactory on SVs due to the demands on detection accuracy. Discriminative methods are divided into deep learning based and CF based. For the former, the Siamese network is used for SOT in [12], [13], and [51], but the parameters and structures of the networks may need to be adjusted for different SVs. In addition, the deep subnetwork obtains low-resolution representations, which may not be suitable for tracking small objects in SVs [12]. For the latter, faster and more robust CFs are used. Du et al. [2] combine the KCF [20] and frame difference, and a fusion strategy is embedded in the tracking framework for SOT in SVs. Shao et al. [16], [17] incorporate motion feature (optical flow) into the KCF framework achieving superior results. In [19],

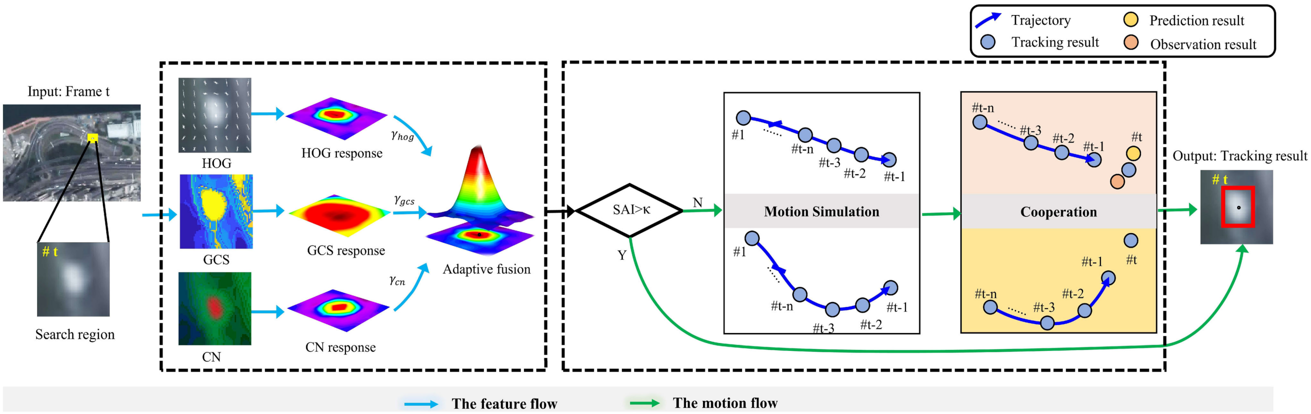

Fig. 1. Overall framework of the proposed CF-based DF tracker. It consists of feature and motion flows. In feature flow, complementary features of HOG, GCS, and CN are fused by adaptive selection of weights γ_{hog}, γ_{gcs}, and γ_{cn}. In motion flow, a SAI is proposed to sense the abnormal states of the object. If the SAI value of the fusion response map is greater than threshold κ, the object's position will be determined by the fusion results. Otherwise, a dual-mode prediction model is constructed synthesizing the linear and nonlinear motion patterns to prevent tracking drift and obtain the object's position when encountering abnormal states.

the KCF framework is also used to track rotating object. Xuan et al. [18] propose a CF embedded with a motion estimation algorithm. However, the tracker [18] is based on the assumption that the motion pattern of the object is linear. The object may be lost when being subjected to a curved path. In addition, using the HOG feature alone in [18] does not guarantee the robustness of the tracker. Thus, some trackers [2], [11], [17] fuse multiple features, but a spectral feature is ignored even though it is as faint as a spatial feature.

III. Tracking Framework

The proposed CF-based DF tracker is modeled on the translation structure of the Staple [23]. The overall motivation of the Staple is elaborated in the following. The desired window p_t locates the object's position in image x_t of frame t and is obtained from set S_t to maximize the score

$$p_t = \text{argmax}_{p\in S_t}\ \ f\left(T\left(x_t, p\right);\ \theta_{t-1}\right) \tag{1}$$

where T denotes the image transformation and θ denotes the model parameter to be solved. Based on parameters θ, function $f(T(x, p); \theta)$ assigns a score to window p on image x. θ should minimize loss $L(\theta; X_t)$.

The loss is determined by the previous images x_i $(i = 1, 2, 3, \ldots, t)$ and the object's positions p_i $(i = 1, 2, 3, \ldots, t)$, and X_t can be written as $X_t = \{(x_i,\ \ p_i)\}_{i=1}^{t}$. The solution of θ is

$$\theta_t = \text{argmin}_{\theta\in\vartheta}\ \{L\left(\theta; X_t\right) +\ \lambda R\left(\theta\right)\} \tag{2}$$

where ϑ denotes the space of parameters and $R(\theta)$ denotes the regularization term with relative weight λ to limit the complexity of the model.

The final score function is a linear fusion of the HOG and GCS scores

$$f\ \ (x) = \gamma_{\text{hog}} f_{\text{hog}}\ \ (x) + \gamma_{\text{gcs}} f_{\text{gcs}}\left(x\right) \tag{3}$$

where γ_{hog} and γ_{gcs} are the weights of the HOG and GCS feature scores, respectively. The overall parameters are $\theta = (\beta, \delta, \gamma_{\text{hog}}, \gamma_{\text{gcs}})$, in which β and δ can be obtained via training and detection parts [23]. The fusion result $f(x)$ is calculated by (3), and the new position of the object is estimated by finding the maximum of $f(x)$. Finally, the parameters $\hat{\beta}$ and δ need to be updated to adapt to changes in the object.

IV. Proposed Approach

In this section, we first introduce the overview of DF tracker. We then cooperate complementary features for tracking. In addition, the SAI and adaptive fusion mechanism of feature flow are described. Finally, a dual-mode prediction model of motion analysis flow is detailed.

A. *Overview of Proposed DF Tracker*

Fig. 1 shows the overall framework of the proposed DF tracker, including feature flow and motion flow. In feature flow, complementary features including the HOG, GCS, and CN of objects are exploited to represent the object. An adaptive fusion mechanism based on a SAI is then used to obtain the fusion results of feature flow. For further refinement, the results are transferred to motion flow. If the SAI value of the fusion response map is great, the object's position will be determined by the fusion results. Otherwise, a dual-mode prediction model is activated to predict the position, which analyzes the previous motion pattern to simulate the motion model.

B. *Complementary Features for Tracking*

Spatial–spectral features, commonly used by CFs [2], [15], [17]–[19], are extracted and fused to represent objects in SVs.

1) *HOG:* It can capture the spatial texture and contours, and has inherent illumination invariance, which makes the HOG suitable for SOT in SVs. However, it is sensitive to deformation because it relies on the spatial layout of the object. It cannot achieve robust tracking for interested objects.

2) *Global Color Statistics:* The GCS feature is a global spectral probability model trained from the foreground and background regions in the first frame. It is inherently invariant to permutation. However, the response map of the GCS feature is flat-peaked, which means it can serve as an auxiliary for SOT.

3) *Color Names:* It is an 11-dimensional spectral label excavated from the spectral features of the target. It can compensate for the information limitation of HOG and GCS with a detail spectrum.

In the training part, the CN is an H-channel image $\mathcal{F}_x : \mathcal{D} \rightarrow \mathbb{R}^H$ obtained from image x and defined as finite grid $\mathcal{D} \subset \mathbb{Z}^2$. The per-image loss is

$$\mathcal{L}\ (x, p, \alpha) = \| \sum_{n=1}^{H} \alpha^n \star \mathcal{F}^n_{T(x,p)} - y \|^2 \tag{4}$$

where α^n is channel n of multichannel image α and $\star$ is circular correlation. The label function y is a desirable Gaussian function that decays from 1 for the center of the object to 0 for the shifted samples of the edge.

For efficiency, α is computed in the Fourier domain, which transforms the circular correlation into a Hadamard product. $\hat{\alpha}^n$ is the discrete Fourier transforms of α^n, $*$ is a complex conjunction, and $\odot$ denotes pointwise. According to an approximate formulation in [24], (4) is minimized by choosing

$$\hat{\alpha}^n = 1/\left(\hat{r} + \lambda\right) \cdot \hat{d}^n \tag{5}$$

where

$$\hat{r} = \sum_{n=1}^{H} \left(\hat{\mathcal{F}}^n_{T(x,p)}\right)^* \odot \hat{\mathcal{F}}^n_{T(x,p)} \tag{6}$$

$$\hat{d}^n = \left(\hat{y}\right)^* \odot \hat{\mathcal{F}}^n_{T(x,p)},\ n = 1, \ldots, H. \tag{7}$$

In the detection part, the response score f_{cn} can be obtained from

$$f_{\text{cn}}\ (x; \alpha) = \sum_{u \in \mathcal{D}} \alpha[u]^T\ \mathcal{F}_x\left[u\right]. \tag{8}$$

To adapt to changes in the object, $\hat{\alpha}$ needs to be updated. η_{cn} denotes the learning rate of the CN feature. The parameters $\hat{r}'_t$ and $\hat{d}'_t$ at frame t are separately computed from (6) and (7) in the new position. For $\hat{\alpha}$, parameters $\hat{r}$ and $\hat{d}$ are updated as

$$\hat{r}_t = \left(1 - \eta_{\text{cn}}\right)\ \hat{r}_{t-1} + \eta_{\text{cn}} \hat{r}^{'}_t \tag{9}$$

$$\hat{d}_t = \left(1 - \eta_{\text{cn}}\right)\ \hat{d}_{t-1} + \eta_{\text{cn}} \hat{d}^{'}_t. \tag{10}$$

After obtaining the HOG, GCS, and CN feature scores, avoiding complex functions, we straight use a linear score function

$$f_{fin}\ (x) = \gamma_{\text{hog}} f_{\text{hog}}\ (x) + \gamma_{\text{gcs}} f_{\text{gcs}}\left(x\right) + \gamma_{\text{cn}} f_{\text{cn}}\left(x\right) \tag{11}$$

where f_{fin} is the fusion result and γ_{hog}, γ_{gcs}, and γ_{cn} are the weights of the HOG, GCS, and CN feature scores, respectively. The object's new position is then estimated by finding the maximum in f_{fin}. Thus, the overall model parameters are $\theta\ = \left(\beta,\ \delta,\ \alpha,\ \gamma_{\text{hog}},\ \gamma_{\text{gcs}}, \gamma_{\text{cn}}\right)$, in which β, δ, and α can be obtained from the training and detection part, whereas γ_{hog}, γ_{gcs}, and γ_{cn} can be determined from the adaptive fusion mechanism that will be described in next part. θ will be updated to adapt to changes in the object.

C. SAI and Adaptive Fusion Mechanism

In tracking, the ideal tracking response map tends to a sharp Gaussian distribution, which is vulnerable to interferences (e.g., occlusion, background clutter, motion blur). In order to describe the concentration of distribution, the SAI (12) is proposed to sense the abnormal states of objects and achieve the adaptive selection of feature weights

$$\text{SAI}\ = \frac{wh \sum_{w,h}\left(s_{w,h} - \bar{s}\right)^4}{\left(\sum_{w,h}\left(s_{w,h} - \bar{s}\right)^2\right)^2} - 3 \tag{12}$$

where w and h are the width and height of the response map, respectively, $\bar{s}$ is the average score of feature response map s. If the SAI value of a feature response map is greater than the threshold, the feature is dominant and the result is reliable. Then, we use a mechanism to adaptive fuse the complementary features in Section IV-B, whose weights in (11) are defined by

$$\gamma_{\text{hog}} = \frac{|\text{SAI}_{\text{hog}}|}{|\text{SAI}_{\text{cn}}| + |\text{SAI}_{\text{hog}}|} \tag{13}$$

$$\gamma_{\text{gcs}} = fix_{\text{gcs}} \tag{14}$$

$$\gamma_{\text{cn}} = \frac{|\text{SAI}_{\text{cn}}|}{|\text{SAI}_{\text{cn}}| + |\text{SAI}_{\text{hog}}|} \tag{15}$$

where SAI_{hog} and SAI_{cn} are the SAI values of the HOG and CN feature response maps, respectively, $fix_{\text{gcs}} =\ 0.2$ is derived from extensive experiments.

Based on the adaptive fusion mechanism, the DF can make full use of the dominant feature to track small objects in SVs. The tracking confidence is then assessed for abnormal states based on SAI. If SAI $> \kappa$, the confidence of feature flow results is high and the object's position will be determined at the maximum of the fusion results. Otherwise, the confidence is low and the state of the object is abnormal, in which the position will be obtained base on a dual-mode prediction model.

D. Dual-Mode Position Prediction Model

Objects in SVs are vulnerable to abnormal states such as occlusion, rotation, background clutter, overtaking, and motion blur. Despite the proposed adaptive fusion mechanism can mitigate such impact, abnormal states inevitably degrade the tracking effects and cause tracking drift. Thus, we propose a dual-mode prediction model to obtain the object's position in motion flow. Specifically, after obtaining the object's trajectory on the basis of the historical results, the curvature of the previous trajectory is used to determine the prediction patterns. If the curvature is small, a Kalman filter [52] will be used to obtain the object's position via its linear prediction pattern. Otherwise, the object's trajectory tends to be in a quadratic nonlinear pattern, so its position will be predicted by nonlinear regression.

1) Kalman Filter for Predicting Linear Trajectory: The Kalman filter [52] is a method for estimating the position and velocity of an object from observations with errors. Let

$S_k = [x_k, v_{x,k}, y_k, v_{y,k}]^T$ denote the state vector, where x_k and y_k are the horizontal and vertical positions of the object at frame k, respectively, and $v_{x,k}$ and $v_{y,k}$ are the horizontal and vertical velocities at frame k, respectively. The estimation process can be divided into two parts: time update and state update.

In the time update part, the state equation and error transfer equation of the prediction process can be written as

$$\hat{S}_{\bar{k}} = M\hat{S}_{k-1} + Du_{k-1} \tag{16}$$

$$E_{\bar{k}} = ME_{k-1}M^T + Q_k \tag{17}$$

where $\hat{S}_{\bar{k}}$ is the priori estimate of the state vector at frame k, $\hat{S}_{k-1}$ is the posterior estimate of the state vector at frame $k-1$, D is the control vector, u_{k-1} is Gaussian noise with covariance matrix Q at frame $k-1$, and $E_{\bar{k}}$ is the priori estimate of the error covariance matrix at frame k in the prediction step. The state transition matrix M can be written as

$$M = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{18}$$

The observation equation is

$$Z_k = HS_k + V_k \tag{19}$$

where Z_k is the observation vector at frame k, S_k is the object's actual state at frame k, and V_k denotes Gaussian noise with covariance matrix R. H is a 2 × 4 observation matrix

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{20}$$

In the state update part, the main three equations can be written as follows:

$$K_k = \frac{E_{\bar{k}}H^T}{HE_{\bar{k}}H^T + R_k} \tag{21}$$

$$\hat{S}_k = \hat{S}_{\bar{k}} + K_k\left(Z_k - H\hat{S}_{\bar{k}}\right) \tag{22}$$

$$E_k = (I - K_kH)\ E_{\bar{k}} \tag{23}$$

where K_k denotes the Kalman gain matrix at frame k, $\hat{S}_k$ is the posteriori state estimate corrected by observation vector Z_k at frame k, and I denotes the identity matrix.

2) Nonlinear Regression for Predicting Nonlinear Trajectory: The Kalman filter is derived from the linear system, which is prone to tracking failure for nonlinear. Frequently, objects in SVs are moving smoothly along curved roads. We use quadratic nonlinear regression to simulate the trajectories with nonlinear pattern and predict the object's position. Let $(z_i,\ x_i),\ i = 1, 2, 3, \ldots, k$ denote the object's position x_i in the *x*-axis direction from frames z_1 to z_k. The quadratic function of the trajectory can be expressed as

$$x_i = b_0 + b_1 z_i + b_2 z_i^2 \tag{24}$$

where b_0, b_1, and b_2 are obtained by solving

$$\min \sum_{i=1}^{k} \left(b_0 + b_1 z_i + b_2 z_i^2 - x_i\right)^2. \tag{25}$$

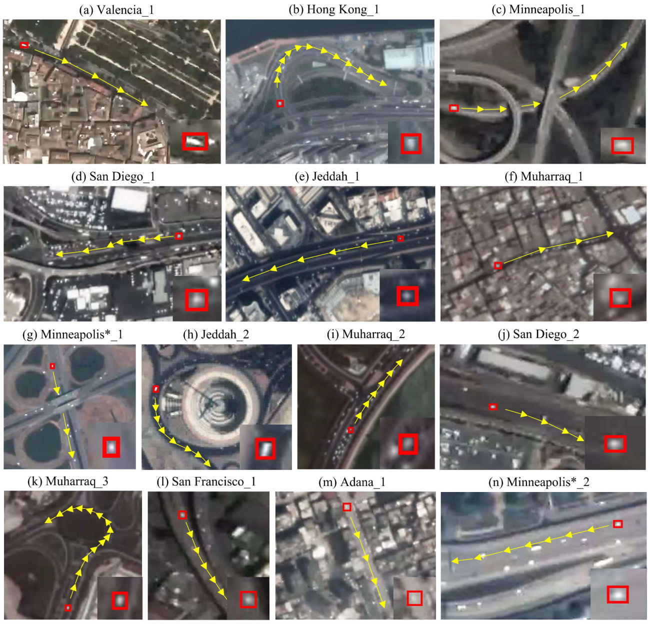

Fig. 2. Overview of SV dataset used in experiments. In each video, a selected vehicle object (marked by a red rectangle) is enlarged and displayed in the lower right corner. The trajectories of the objects are roughly indicated by the yellow arrows. Minneapolis _1 is the first cropped sequence of the Minneapolis SV, whereas Minneapolis∗ and Minneapolis are different SVs acquired from the same area. (a) Valencia_1. (b) Hong Kong_1. (c) Minneapolis_1. (d) San Diego_1. (e) Jeddah_1. (f) Muharraq_1. (g) Minneapolis∗_1. (h) Jeddah_2. (i) Muharraq_2. (j) San Diego_2. (k) Muharraq_3. (l) San Francisco_1. (m) Adana_1. (n) Minneapolis∗_2.

Through simplification, the normal equation of (25) can be written as

$$\begin{bmatrix} k & \sum_{i=1}^{k} z_i & \sum_{i=1}^{k} z_i^2 \\ \sum_{i=1}^{k} z_i & \sum_{i=1}^{k} z_i^2 & \sum_{i=1}^{k} z_i^3 \\ \sum_{i=1}^{k} z_i^2 & \sum_{i=1}^{k} z_i^3 & \sum_{i=1}^{k} z_i^4 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{k} x_i \\ \sum_{i=1}^{k} z_i x_i \\ \sum_{i=1}^{k} z_i^2 x_i \end{bmatrix}. \tag{26}$$

By solving the coefficient matrix $[b_0, b_1, b_2]^T$, the trajectory equation (24) can be obtained to simulate the motion pattern of the object. Similarly, the function in the *y*-axis direction can be solved, and the object's position can also be obtained.

V. Experiments

A. Experimental Setups

1) Dataset: For comprehensive evaluation, nine large-size SVs (Valencia: 4096 × 2160, Hong Kong: 4096 × 3072, Minneapolis: 4096 × 2160, San Diego: 4096 × 2160, Jeddah: 4096 × 3072, Muharraq: 4096 × 2160, San Francisco: 3840 × 2160, Adana: 4096 × 2160 and Minneapolis∗: 4096 × 2160) are cropped into 14 small-size SVs. Fig. 2 presents the experimental dataset, and Table I presents the size of objects and frames. Supported by Chang Guang Satellite Technology Co., Ltd., the SVs have a GSD of 0.92 m, and the videos are 8-b quantization RGB images with spectra ranging from 437 to 723 nm. A total of 14 moving vehicles are labeled by 4374 minimum horizontal bounding boxes. The vehicles have a maximum size of 12 × 14 pixels and a minimum size of 6 × 8 pixels. Each video has

TABLE I
INFORMATION OF THE 14 CROPPED SVS, IN WHICH “VALENCIA_1” DENOTES THE FIRST CROPPED REGION OF THE VALENCIA VIDEO

Videos	Frame Size	Target Size
Valencia_1	511 × 332	14 × 12
Hong Kong_1	437 × 301	10 × 10
Minneapolis_1	248 × 173	10 × 8
San Diego_1	385 × 230	8 × 8
Jeddah_1	435 × 233	8 × 8
Muharraq_1	241 × 169	8 × 6
Minneapolis*_1	330 × 321	8 × 10
Jeddah_2	301 × 202	12 × 14
Muharraq_2	223 × 199	6 × 8
San Diego_2	230 × 154	10 × 8
Muharraq_3	237 × 206	8 × 8
San Francisco_1	123 × 137	8 × 8
Adana_1	235 × 184	10 × 10
Minneapolis*_2	237 × 132	10 × 8

TABLE II
LIST OF ABNORMAL STATES AND CORRESPONDING SV DATASET

Abnormal State	Description	Corresponding Datasets
occlusion	The object is partially or fully occluded by clouds, bridges or overpasses.	Valencia_1, Adana_1, Minneapolis_1, Minneapolis*_1.
rotation	The object rotates in the image plane.	Hong Kong_1, Jeddah_2, Muharraq_3.
background clutter	The background is similar to the object.	Minneapolis*_2, Jeddah_1.
overtaking	The object overtakes similar target in close proximity, or is overtaken.	San Diego_1, Muharraq_1, Muharraq_2.
motion blur	The object is blurred due to the motion of object or camera.	San Francisco_1, San Diego_2.

a dominant abnormal state based on the characteristics of the scenario, and a short description of all states is given in Table II.

2) Evaluation Methodology: The precision plot and success plot are applied to measure the tracking performance [53], [54]. Center location error (CLE) calculates the average Euclidean distance between the center of the ground truth and estimated bounding box. The precision plot shows the percentage of frames for which the CLE is smaller than predefined thresholds T_p. Considering the low resolution of SV accompanied by small size of objects, we use thresholds $T_p \in [1,\ 20]$ to measure the performance in positioning. In the success plot, the overlap is used for evaluation. Given ground truth R_G and estimated bounding box R_T, the overlap can be calculated by

$$\text{overlap} = \frac{|R_G \cap R_T|}{|R_G \cup R_T|} \tag{27}$$

TABLE III
EFFECTS OF SAI THRESHOLDS ON TRACKING EFFECTS

κ	Precision plot	Success plot
0	0.907	0.686
-0.15	0.876	0.653
-0.3	0.875	0.657
-0.45	0.905	0.687
-0.6	**0.912**	**0.700**
-0.75	0.872	0.659
-0.9	0.871	0.651

The first is shown in bold.

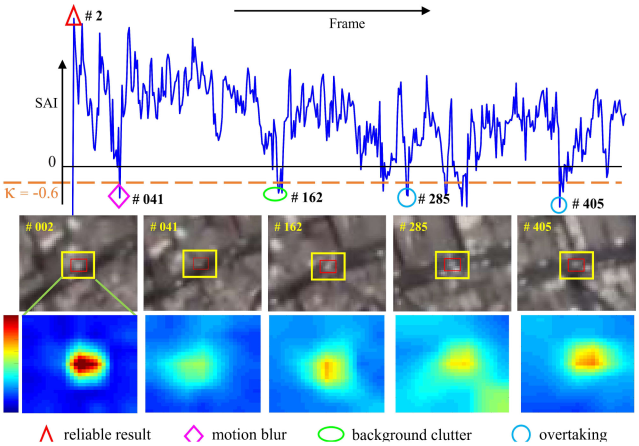

Fig. 3. Monitoring process of abnormal states. From first to third row: SAI versus frames, tracking samples (tracking results in red and search regions in yellow), and response maps. The state is normal in frame #002. In frames #041, #162, #285, and #405, the object is under abnormal states, and the response maps change more or less compared with #002. The abnormal states are perceived by comparing the SAI with threshold κ.

where $\cap$ and $\cup$ denote intersection and union operators, respectively, and $|\cdot|$ is the number of pixels in the region [53], [54]. The success plot shows that the success rate surpasses the threshold range $T_s \in [0,\ 1]$, and measures the tracker's performance in positioning and estimating the size of the object. In this article, all trackers are ranked by the area under the curve (AUC) of the precision plot and success plot. Compared with the precision plot, the success plot is more representative [9]. Thus, we mainly rank trackers based on the AUC of the success plot, and use the FPS to evaluate tracking speed.

3) Implementation Details: The weight λ is set to $1e-3$, and the fixed area is 60^2. Considering that the changes of objects are stable, the learning rates η_{hog}, η_{gcs}, and η_{cn} are set to 0.01, 0.005, and 0.005, respectively. The effects of SAI threshold κ on tracking result are presented in Table III. An optimal result is obtained with $\kappa = -0.6$, and a sample is shown in Fig. 3. The other parameters are set to the same as those in Staple [23], and all trackers are executed on a workstation with a 3.20 GHz Intel(R) Xeon(R) Gold 6134 CPU (32-core) and NVIDIA GeForce RTX 2080 Ti GPU.

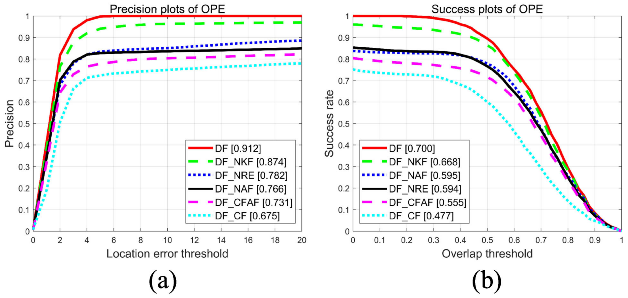

Fig. 4. (a) Precision plot and (b) success plot of the variant trackers on 14 SVs. The values in the legends are the AUC. "OPE" = One-pass evaluation, which initializes a tracker in the first frame and lets it run to the end of the sequence.

TABLE IV
COMPONENTS AND RESULTS OF ABLATION EXPERIMENTS

Trackers	CF	AF	KF	RE	Precision plot	Success plot
DF_CF	√	—	—	—	0.675	0.477
DF_CFAF	√	√	—	—	0.731	0.555
DF_NAF	√	—	√	√	0.766	0.595
DF_NRE	√	√	√	—	0.782	0.594
DF_NKF	√	√	—	√	0.874	0.668
DF	√	√	√	√	**0.912**	**0.700**

Four components are listed to represent the variants. "CF" = Translation structure of staple. "AF" = Adaptive fusion of complementary features. "KF" = Kalman filter in the motion flow. "RE" = Nonlinear regression in the motion flow.

B. Ablation Study

To validate the proposed DF, five variants are conducted, including two of addition experiments (DF_CF and DF_CFAF) and three of removal experiments (DF_NAF, DF_NRE, and DF_NKF). Fig. 4 shows the precision and success plots, and Table IV summarizes the components and experimental results of these trackers. DF_CF is the baseline tracker, indicating that DF has only the translation structure of Staple. DF_CFAF achieves adaptive fusion of CN over DF_CF, and DF_NAF removes the adaptive fusion of CN from DF. DF_NRE and DF_NKF remove the nonlinear regression and the Kalman filter of the dual-mode prediction model from DF, respectively.

1) For Feature Flow: In Table IV, by comparing with the baseline DF_CF and DF_CFAF, it can be seen that the AUC of the precision plot is improved from 0.675 to 0.731 (5.6% improvement) and the success plot is enhanced from 0.477 to 0.555 (7.8% improvement) using the feature flow. While comparing the DF and DF_NAF, we find a 14.6% and 10.5% reduction in the AUC of the precision and success plots after removing the adaptive fusion part from DF. Due to the absence of feature flow, the DF_NAF cannot adaptively fuse the complementary features of the object, making it difficult to represent small objects, which leads to tracking failure. Fig. 5 shows the tracking examples of DF_NAF and DF, where DF can discriminate object from background and avoid tracking drift.

2) For Motion Flow: By comparing the DF and DF_CFAF, it can be seen that the AUC of the precision plot is reduced from 0.912 to 0.731, whereas the success plot is reduced from 0.700 to 0.555 without the motion flow. This is due to the inability to perceive the abnormal states of the object and predict its position. Therefore, DF_CFAF encounters tracking drift. Comparing the DF_CF, DF_NAF yields a gain of 9.1% in the precision plot and 11.8% in the success plot. Furthermore, to evaluate the effects of the motion flow, DF_NRE and DF_NKF are added for validation. As presented in Table IV, the AUC of DF is superior to those of DF_NRE and DF_NKF, and the DF preforms optimal performance. This is because the dual-mode prediction model cooperates the linear and nonlinear motion patterns, allowing it to handle abnormal motions such as lane changes and turns. As shown in Fig. 6(a), the vehicle moves on a straight road when another one with similar features passes by quickly. The DF_NRE locates the vehicle, whereas DF_NKF encounters failure. This is because the Kalman filter predicts the linear trajectory more precisely than the nonlinear. In Fig. 6(b), a vehicle encounters complete occlusion by bridges while traveling at high speed on a curved highway. In this case, DF_NKF locates the vehicle, whereas DF_NRE loses it when subjected to the occlusion by a bridge. This is attributed to the property of nonlinear regression in DF_NKF. Overall, the proposed DF can determine the prediction mode based on the motion patterns, so it achieves superior results.

C. Comparison With State-of-the-Art Methods

We compared the proposed method with 13 trackers, namely, KCF [20], SAMF [47], Staple[23], C-COT [48], fDSST [55], ECO [49], SiamRPN [42], SiamRPN++ [39], ASRCF [56], GFS-DCF [50], CFME [18], SiamFC++ [57], and TransT [58]. These methods include CF based and deep learning based. The CF-based CFME is an open-source design for SOT in SVs. Few trackers are tailored for SVs. The codes are not public and some key variables are omitted. Moreover, these methods were tested on unpublished datasets and different benchmarks. Therefore, we selected CFME for comparison. Table V summarizes the characteristics of trackers and experimental results, sorted by AUC of the success plot. Fig. 7 presents the average precision and success plots. With AUC of 0.912 and 0.700 in the precision and success plots, the proposed method performs remarkable performance, whereas KCF achieves the worst. CFME produces competitive performance due to the fact that the motion average and Kalman filter are embedded in KCF to mitigate tracking drift, ranking the first in the compared trackers. The proposed DF tracker boosts CFME by 10.2% and 9.8% in the precision and success plots, respectively. Compared with ECO, the champion of VOT2017, the proposed method provides a gain of 19.9% in the precision plot and 17.3% in the success plot due to the exploitation of potential spatial–spectral features. Compared with ASRCF and GFS-DCF, the proposed approach reaches 23.1% and 20.6% boost in the success plot due to the consideration of motion model. This suggests that the motion information contained in adjacent frames facilitates tracking in SVs. In contrast with SiamRPN++, the proposed method achieves a solid improvement in accuracy. Compared with Staple, DF increases the precision and success plots by 24%+, and compared with SiamFC++ and TransT trackers, the proposed method exceeds

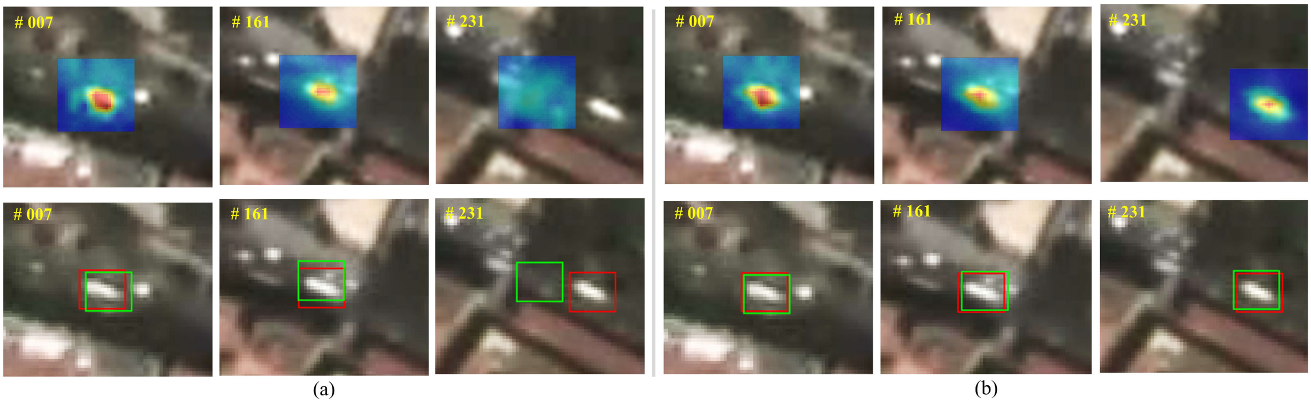

Fig. 5. Visualization of the (a) DF_NAF and (b) DF in Valencia_1. The number in the upper left corner of image indicates the frame. The first row shows the response maps. High response scores are in red and low scores are in blue. The second row presents the tracking results (in green rectangles) and ground truth (in red rectangles).

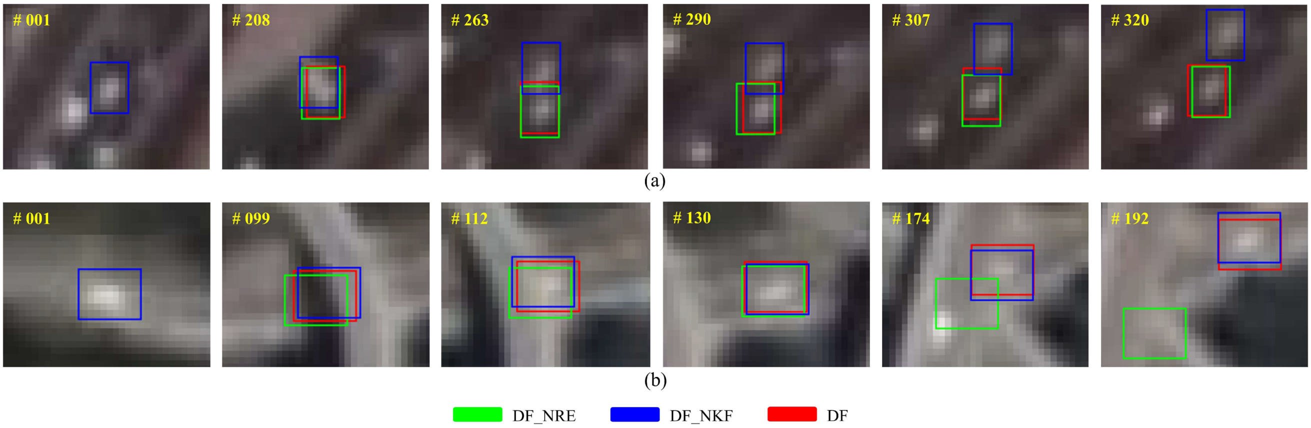

Fig. 6. Visualization of tracking results of DF_NRE, DF_NKF, and DF in SVs: (a) Muharraq_2 and (b) Minneapolis_1. The current frame of the video is displayed in the upper left corner of each image, best viewed in color.

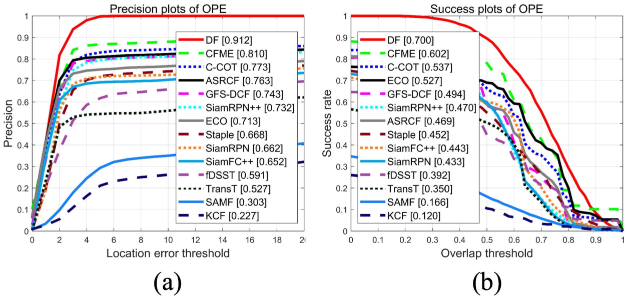

Fig. 7. (a) Average precision plot and (b) success plot of all trackers over 14 real SVs.

them by 26% and 38.5%, 25.7% and 35%, in the precision and success plots, respectively. Overall, experimental results verify that the proposed DF tracker well tracks the objects. It is attributed to both the adaptive fusion mechanism incorporated in feature flow and dual-mode prediction model embedded in motion flow. Moreover, DF is capable of running at over 155 FPS on the CPU. Compared with trackers operating on the CPU or GPU, DF can achieve real-time speed in tracking objects of SVs. Experimental results demonstrate the state-of-the-art effects and superior speed of the proposed tracker.

Fig. 8 shows the precision and success plots of per-state to evaluate the strengths and weaknesses of trackers. For clarity, Fig. 9 shows the radar plots for top seven trackers. We find that, for the precision plots, the DF ranks highest in three (occlusion, overtaking, and motion blur) out of five states and achieves the first in overall AUC. For the success plots, DF ranks among top two trackers in four out of five states. The reason why DF achieves inferior results under rotation datasets is that slight background jitter would affect the position of the object, weakening the performance of the dual-mode prediction model. The proposed method achieves the fourth place under the background clutter data. This is because the object is relatively similar to the background, which limits the extraction of prominent features. It can be seen DF achieves significant improvement under the occlusion state. This is attributed to the SAI and dual-mode prediction algorithm. The SAI perceives the occlusion and nonocclusion states, and the signal is then transmitted to the dual-mode prediction algorithm. It synthesizes the linear and nonlinear motion patterns to handle occlusion

TABLE V
DETAILS OF TRACKERS AND EXPERIMENTAL RESULTS ON 14 SVs

Trackers	Framework	Features	MS	MTD	Precision Plot	Success Plot	FPS
KCF (TPAMI 2015)	KCF	HOG	—	—	0.227	0.120	**373.2^{C}**
SAMF (ECCV 2014)	KCF	HOG + CN	√	—	0.303	0.166	62.3^{C}
TransT (CVPR 2021)	Transformer	ConvFeat	√	—	0.527	0.350	11.5^{G}
fDSST (TPAMI 2017)	CSK	HOG	√	—	0.591	0.392	**181.2^{C}**
SiamRPN (CVPR 2018)	SiameseFC	ConvFeat	√	—	0.662	0.433	61.5^{G}
SiamFC++ (AAAI 2020)	SiameseFC	ConvFeat	√	—	0.652	0.443	78.7^{G}
Staple (CVPR 2016)	DCF + II	HOG + GCS	√	—	0.668	0.452	72.3^{C}
ASRCF (CVPR 2019)	KCF	ConvFeat + HOG	√	—	0.763	0.469	21.1^{G}
SiamRPN++ (CVPR 2019)	SiameseFC	ConvFeat	√	—	0.732	0.470	28.4^{G}
GFS-DCF (ICCV 2019)	DCF	ConvFeat + HOG + CN	√	—	0.743	0.494	5.8^{G}
ECO (CVPR 2017)	CCF	ConvFeat	√	—	0.713	0.527	1.2^{G}
C-COT (ECCV 2016)	CCF	ConvFeat	√	—	**0.773**	**0.537**	0.3^{G}
CFME (TGRS 2020)	KCF	HOG	—	√	**0.810**	**0.602**	130.3^{C}
DF (Ours)	DCF + II	HOG + GCS + CN	—	√	**0.912**	**0.700**	**155.2^{C}**

The first, second best, and third best are shown in color. "MS" = Mechanisms for scale. "MTD" = Mechanisms for tracking drift. (For framework, KCF = Kernelized correlation filter, Transformer = Transformer-like, CSK = Circulant structure of tracking-by-detection with kernels, SiameseFC = Fully convolutional Siamese network, DCF = Discriminative correlation filter, II = Integral image, and CCF = Continuous convolution filter. For features, HOG = Histogram of oriented gradients, CN = Color names, ConvFeat = Convolutional features, and GCS = Global color statistics. For FPS, G means that the ConvFeat extraction depends on the GPU and C means that the feature extraction depends on the CPU.)

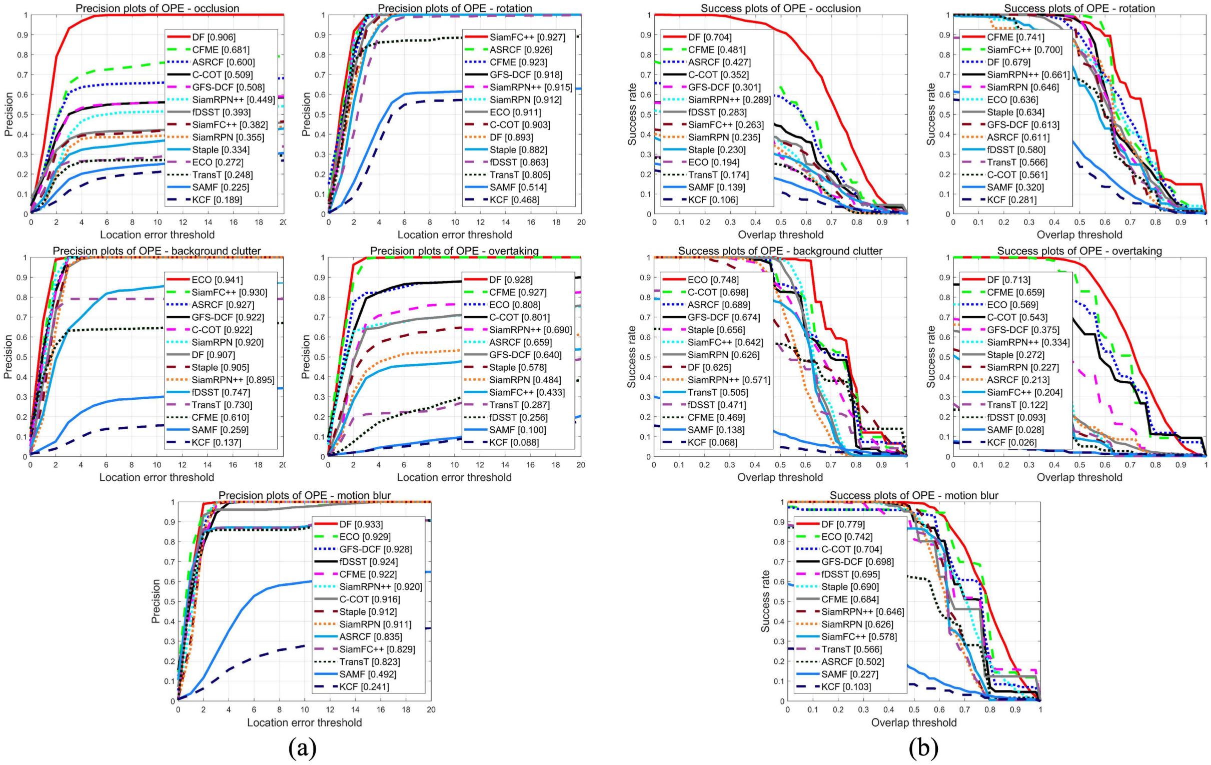

Fig. 8. (a) Precision plots and (b) success plots of comparison experiments with 13 trackers under five abnormal states: occlusion, rotation, background clutter, overtaking, and motion blur.

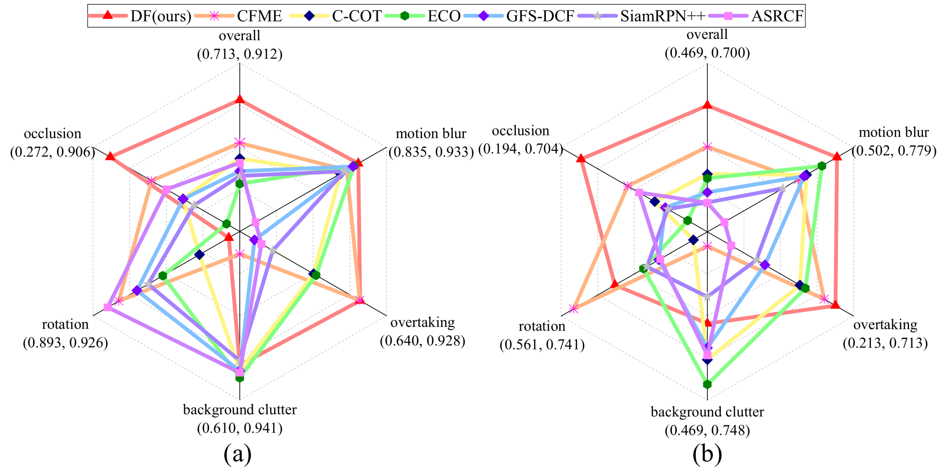

Fig. 9. AUC of (a) precision plots and (b) success plots under abnormal states. The top seven trackers are shown. The values in parentheses indicate the range of AUC in terms of overall and per state.

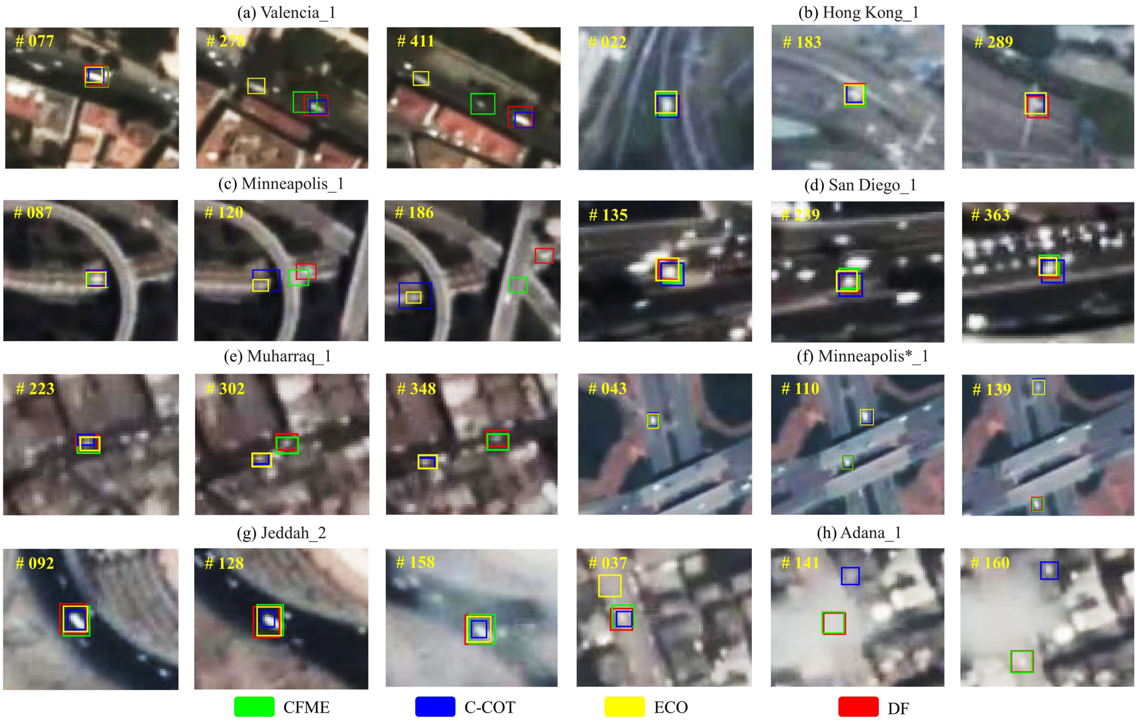

Fig. 10. Qualitative examples for top four trackers. (a) Valencia_1. (b) Hong Kong_1. (c) Minneapolis_1. (d) San Diego_1. (e) Muharraq_1. (f) Minneapolis∗_1. (g) Jeddah_2. (h) Adana_1.

of objects, yielding significant performance. Overall, DF is capable of coping with the abnormal states of the objects through the hybridization of the spatial–spectral feature fusion and the motion model.

In visual comparison, tracking examples of the top four trackers are shown in Fig. 10. In Fig. 10(c), a vehicle is occluded twice when moving along a curved highway. DF is capable of sensing the abnormal state and predicting the object's position, whereas C-COT and ECO all lose the object. Although the CFME can predict the object's position, it loses it due to limited consideration of the nonlinear motion pattern of the object. As an overtaking case in Fig. 10(e), a vehicle, similar to the buildings and vehicles parking on the sides of the road, travels along a narrow street. Only the CFME and DF capture the object in all frames, whereas the DF tracks more accurately. In other cases shown in Fig. 10, the proposed DF could track objects with higher accuracy.

VI. Conclusion

SOT has great potential in remote-sensing surveillance. In this article, we explore the SV SOT from the perspective of spatial–spectral feature fusion and motion model and propose a CF-based DF tracker to address problems of limited feature representation and tracking drift. In feature flow, an adaptive mechanism is employed to fuse complementary features. The results are then refined in motion flow. A dual-mode prediction model is constructed to simulate the motion patterns for searching the object's position, allowing the tracker robust to abnormal states. Extensive experiments on 14 SVs prove the outstanding performance in tracking objects of SVs. Future work should focus on solving the rotation of objects.

Acknowledgment

The authors wish to thank the editors and reviewers for their valuable comments and useful suggestions.

References

[1] Y. M. Wang, T. Y. Wang, G. Zhang, Q. Cheng, and J. Q. Wu, "Small target tracking in satellite videos using background compensation," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7010–7021, Oct. 2020.

[2] B. Du, Y. Sun, S. Cai, C. Wu, and Q. Du, "Object tracking in satellite videos by fusing the kernel correlation filter and the three-frame-difference algorithm," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 2, pp. 168–172, Feb. 2018.

[3] R. Abileah and S. Vignudelli, "Video from earth orbiting satellites (for oceanographers)," in *Proc. Oceans MTS/IEEE Washington*, 2015, pp. 1–7.

[4] C. J. Legleiter and P. J. Kinzel, "Surface flow velocities from space: Particle image velocimetry of satellite video of a large, sediment-laden river," *Front. Water*, vol. 3, May 2021, Art. no. 652213.

[5] W. Li et al., "Research on multiview stereo mapping based on satellite video images," *IEEE Access*, vol. 9, pp. 44069–44083, 2021.

[6] H. Liu, Y. Gu, T. Wang, and S. Li, "Satellite video super-resolution based on adaptively spatiotemporal neighbors and nonlocal similarity regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8372–8383, Dec. 2020.

[7] S. A. Ahmadi, A. Ghorbanian, and A. Mohammadzadeh, "Moving vehicle detection, tracking and traffic parameter estimation from a satellite video: A perspective on a smarter city," *Int. J. Remote Sens.*, vol. 40, no. 22, pp. 8379–8394, Nov. 2019.

[8] F. Shi, F. Qiu, X. Li, Y. W. Tang, R. F. Zhong, and C. K. Yang, "A method to detect and track moving airplanes from a satellite video," *Remote Sens.*, vol. 12, no. 15, Aug. 2020, Art. no. 2390.

[9] B. Du, S. Cai, and C. Wu, "Object tracking in satellite videos based on a multiframe optical flow tracker," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 3043–3055, Aug. 2019.

[10] W. Ao, Y. W. Fu, X. Y. Hou, and F. Xu, "Needles in a haystack: Tracking city-scale moving vehicles from continuously moving satellite," *IEEE Trans. Image Process.*, 2020, vol. 29, pp. 1944–1957, doi: 10.1109/tip.2019.2944097.

[11] 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, Feb. 2020.

[12] J. Shao, B. Du, C. Wu, M. Gong, and T. Liu, "HRSiam: High-resolution Siamese network, towards space-borne satellite video tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 3056–3068, Feb. 2021.

[13] K. Zhu et al., "Single object tracking in satellite videos: Deep Siamese network incorporating an interframe difference centroid inertia motion model," *Remote Sens.*, vol. 13, no. 7, Apr. 2021, Art. no. 1298.

[14] J. Feng et al., "Cross-frame keypoint-based and spatial motion information-guided networks for moving vehicle detection and tracking in satellite videos," *ISPRS J. Photogramm. Remote Sens.*, vol. 177, pp. 116–130, Aug. 2021.

[15] Y. Guo, D. Yang, and Z. Chen, "Object tracking on satellite videos: A correlation filter-based tracking method with trajectory correction by Kalman filter," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 9, pp. 3538–3551, Sep. 2019.

[16] J. Shao, B. Du, C. Wu, and L. F. Zhang, "Tracking objects from satellite videos: A velocity feature based correlation filter," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7860–7871, Oct. 2019.

[17] J. Shao, B. Du, C. Wu, and L. Zhang, "Can we track targets from space? A hybrid kernel correlation filter tracker for satellite video," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8719–8731, Nov. 2019.

[18] S. Xuan, S. Li, M. Han, X. Wan, and G. 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.

[19] S. Y. Xuan et al., "Rotation adaptive correlation filter for moving object tracking in satellite videos," *Neurocomputing*, vol. 438, pp. 94–106, May 2021.

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

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

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

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

[24] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 65.61–65.11.

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

[26] 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.

[27] H. K. Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4630–4638.

[28] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey," *Neurocomputing*, vol. 300, pp. 17–33, Jul. 2018.

[29] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *Int. J. Robot. Res.*, vol. 39, no. 8, pp. 895–935, Jul. 2020.

[30] M. Manafifard, H. Ebadi, and H. A. Moghaddam, "A survey on player tracking in soccer videos," *Comput. Vis. Image Understanding*, vol. 159, pp. 19–46, Jun. 2017.

[31] H. A. Fuenzalida, R. Sanchez, and R. D. Garreaud, "A climatology of cutoff lows in the Southern Hemisphere," *J. Geophys. Res., Atmos.*, vol. 110, no. D18, Sep. 2005, Art. no. D18101.

[32] R. T. Collins, "Mean-shift blob-tracking through scale space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2003, pp. 234–240.

[33] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[34] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. C. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.

[35] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, vol. 37, pp. 597–606.

[36] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[37] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4293–4302.

[38] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, vol. 9914, pp. 850–865.

[39] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4277–4286.

[40] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1328–1338.

[41] G. Bhat, J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg, “Unveiling the power of deep tracking,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, vol. 11206, pp. 493–509.

[42] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, “High performance visual tracking with Siamese region proposal network,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8971–8980.

[43] S. Pu, Y. B. Song, C. Ma, H. G. Zhang, and M. H. Yang, “Deep attentive tracking via reciprocative learning,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, vol. 31, pp. 1935–1945.

[44] Y. Song et al., “VITAL: Visual tracking via adversarial learning,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8990–8999.

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

[46] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, “Exploiting the circulant structure of tracking-by-detection with kernels,” in *Proc. Eur. Conf. Comput. Vis.*, 2012, vol. 7575, pp. 702–715.

[47] Y. Li and J. Zhu, “A scale adaptive kernel correlation filter tracker with feature integration,” in *Proc. Eur. Conf. Comput. Vis.*, 2015, pp. 254–265.

[48] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, “Beyond correlation filters: Learning continuous convolution operators for visual tracking,” in *Proc. Eur. Conf. Comput. Vis.*, 2016, vol. 9909, pp. 472–488.

[49] 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.

[50] T. Xu, Z. Feng, X. Wu, and J. Kittler, “Joint group feature selection and discriminative filter learning for robust visual object tracking,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7949–7959.

[51] L. Ruan, Y. Guo, D. Yang, and Z. Chen, “Deep Siamese network with motion fitting for object tracking in satellite videos,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, Mar. 2022, Art. no. 6508005.

[52] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, 1960.

[53] 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.

[54] Y. Wu, J. Lim, and M. H. Yang, “Object tracking benchmark,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[55] M. Danelljan, G. Hager, 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.

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

[57] Y. D. Xu, Z. Y. Wang, Z. X. Li, Y. Yuan, and G. Yu, “SiamFC++ : Towards robust and accurate visual tracking with target estimation guidelines,” in *Proc. 34th AAAI Conf. Artif. Intell./32nd Innov. Appl. Artif. Intell. Conf./10th AAAI Symp. Educ. Adv. Artif. Intell.*, 2020, vol. 34, pp. 12549–12556.

[58] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu, “Transformer tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8126–8135.

Yuzeng Chen (Student Member, IEEE) received the B.S. degree in geographic information science from Southwest University of Science and Technology, Mianyang, China, in 2020. He is currently working toward the M.S. degree in surveying engineering with the School of Geosciences and Info-Physics, Central South University, Changsha, China.

His research interests include visual tracking and machine learning.

Yuqi Tang (Member, IEEE) received the Ph.D. degree in photogrammetric and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2013.

Since 2017, she has been an Associate Professor with the School of Geosciences and Info-Physics, Central South University, Changsha, China. Her research interests include object identification/tracking, land-cover/-use classification and change detection in multisource remote sensing image, and natural resource monitoring.

Zhiyong Yin received the M.S. degree in surveying and mapping, in 2018, from the School of Geosciences and Info-Physics, Central South University, Changsha, China, where he is currently working toward the Ph.D. degree in surveying engineering.

His research interests include small object detection and image/video processing.

Te Han (Student Member, IEEE) received the B.S. degree in surveying engineering from the School of Geosciences and Info-Physics, Central South University, Changsha, China, in 2017, where he is currently working toward the Ph.D. degree in surveying engineering.

His current research focuses on land-cover/-use change detection with heterogeneous remote sensing images.

Bin Zou received the Ph.D. degree in cartography and geographic information engineering from Central South University, Changsha, China, in 2009.

He is currently a Professor with the Department of Geomatics and Remote Sensing, School of Geosciences and Info-Physics, Central South University. His research interests include remote sensing monitoring and geographic modeling with special focus on spatial-temporal changes of natural resources and environment pollution.

Huihui Feng (Member, IEEE) received the Ph.D. degree in cartography and geographic information system from Beijing Normal University, Beijing, China, in 2012.

From 2012 to 2013, he was a Research Associate with the Institute of Urban Environment, Chinese Academy of Sciences, Xiamen, China. From 2013 to 2016, he was a Research Associate with the Nanjing Institute of Geography and Limnology, Chinese Academy of Sciences, Nanjing, China. Since 2016, he has been an Associate Professor with the School of Geosciences and Info-Physics, Central South University, Changsha, China. His research interests include land-cover/-use change simulation, remote sensing of resources and environment, and ecological and environmental effects of land-cover/-use change.