IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 16, 2020, accepted June 26, 2020, date of publication July 6, 2020, date of current version July 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3007261

Robust Visual Tracking With Occlusion Judgment and Re-Detection

SHAOMING LI[1,2], JUN CHU[1,2], GUOCHONG ZHONG[1,2], LU LENG[1,2,3], (Member, IEEE), AND JUN MIAO[2,4]

[1]School of Software, Nanchang Hangkong University, Nanchang 330063, China
[2]Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition, Nanchang Hangkong University, Nanchang 330063, China
[3]School of Electrical and Electronics Engineering, College of Engineering, Yonsei University, Seoul 120749, South Korea
[4]School of Aeronautical Manufacturing Engineering, Nanchang Hangkong University, Nanchang 330063, China

Corresponding author: Jun Chu (chuj@nchu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61663031, Grant 61866028, and Grant 61661036, in part by the Key Program Project of Research and Development (Jiangxi Provincial Department of Science and Technology) under Grant 20171ACE50024 and Grant 20192BBE50073, and in part by the Foundation of China Scholarship Council under Grant CSC201908360075.

ABSTRACT A detection algorithm is often used to remedy tracking failures in a typical single-target visual tracking algorithm. In practice, when a target is occluded for a long time, neither the tracking module nor the detection module can accurately predict the position of the target. To accurately locate the target, we first introduce the $\ell_1\ell_2$ loss function to reduce the sensitivity of correlation filter-based method to local occlusion. To solve the instability of algorithms based on the single feature in complex scene, we use the histogram of oriented gradient (HOG) features and color names (CN) features to train a filter respectively, and the fusion weights are calculated according to the difference between the response value of each filter and the expected response value. At the same time, we adaptively update the model online by calculating the sensitivity of different filters. We follow the re-detection idea in long-term tracking, the peak to sidelobe ratio (PSR) is used to judge the serious occlusion, and we use support vector machine (SVM) for re-detection after severe occlusion or target out-of-view. In this paper, 34 sets of sequences are selected to evaluate the proposed algorithm. The sufficient experimental results demonstrate that our algorithm has strong anti-occlusion ability and robustness performance. We compare our proposal with several state-of-the-art algorithms under all the sequences of OTB100, and our algorithm yields highly competitive performance for tracking.

INDEX TERMS Correlation filter, re-detection, support vector machine, visual tracking.

I. INTRODUCTION

Visual tracking is one of the most basic problems in computer vision. In brief, given the target information of the first frame, the visual tracking task evaluates the target location in the subsequent frames. It is used in wide-range scenarios, such as vehicle navigation, human-computer interaction, automatic surveillance, to name just a few.

Although visual tracking has been studied for a few decades and great progress has been made, it is still a tough task to design a robust and efficient tracker due to difficulties for both foreground and background variations. Currently, the mainstream method of visual tracking is the discriminative correlation filter (DCF) method combined with deep learning [1]–[5]. But the tracking algorithms based on the traditional correlation filter (CF) have achieved top-ranked performance and drawn increasing attentions because of their superior computation and fair robustness to photometric and geometric variations [6]–[8]. The CF-based trackers convert the correlation operations in the spatial domain to the element-wise multiplications in the frequency domain, which substantially improve the complexity and the tracking speed [9]–[12].

Some evidences show that occlusion and deformation are still the two most difficult problems in visual tracking [13], [14]. When a slight and simple partial occlusion happened to a target during the visual tracking, the target appearance changes slightly, which induces a small error. In such scenes, robust tracking can be performed by some spatially regularized methods or reliable patch

The associate editor coordinating the review of this manuscript and approving it for publication was Abdullah Iliyasu.

model [9], [10], [15]. Nevertheless, a severe occlusion or target out-of-view occurs, the error caused by occlusion and disocclusion tends to accumulate, which will be very large. When the target is out of sight, the appearance of the target is completely invisible, which results in the inability to obtain the appearance. The existing tracking methods cannot achieve a good performance for intricacy local occlusion challenge [15]–[17]. Some long-term tracking algorithms introduced re-detection strategies to solve the problem of the target out-of-view [18]–[20]. These methods use a global search or re-detection strategy to retrieve the target. Therefore, they also work well when a target is heavily occluded.

In this work, we follow the idea in long-term visual tracking and learn a support vector machine (SVM) detector based on [21] for re-detection. Considering the sensitivity of the conventional CF-based trackers to local occlusion, we introduce the $\ell_1\ell_2$ loss function. We combine the histogram of oriented gradient (HOG) feature and color names (CN) feature with the CF method to train two different models, and weight them based on the responses for a robust tracker. Meanwhile, we develop an occlusion judgment strategy and an adaptive online model update strategy for robust visual tracking, that will be carefully discussed in Section III.

The contributions of this paper can be summarized in the following three aspects:

1. Unlike other CF-based methods, we introduce $\ell_1\ell_2$ loss function to reduce the sensitivity to local occlusion, which is a great challenge for object tracking.
2. A new visual tracking framework combined with re-detection is proposed to perform robust tracking, including a novel adaptive online model update strategy based on feature fusion. It iteratively conducts adaptive learning on a variety of features and can adapt to a variety of challenging scenarios.
3. We carefully select 34 sets of video frame sequences from the OTB dataset containing 11 interference factors, including the illumination variation (IV), scale variation (SV), occlusion (OCC), etc., to prove the advantage of our proposal. The sufficient results show that the proposed algorithm has a satisfactory anti-occlusion ability and robustness.

II. RELATED WORK

In this work, we conduct our proposal based on the CF method combining the HOG features and the CN features. In this section, we revisit the related works, including: A) correlation tracking, B) tracking-by-detection, and C) judgment occlusion based on the peak to sidelobe ratio (PSR) [22].

A. CORRELATION TRACKING

Correlation filter originated from the field of signal processing has been widely used in the field of target detection and recognition. In visual tracking, given the datasets, a correlation tracker trains a filter to recognize the target in the subsequent frames. Bolme *et al.* [17] first introduced the correlation filter into visual tracking. Henriques *et al.* [23], [24] further improved the performance of the CF-based tracking by using approximate dense sampling, ridge regression, and kernel trick. Subsequently, Danelljan *et al.* [25] and Dai *et al.* [10] respectively learned a specific regularization term to punish the filter with respect to a large background response and an adaptive spatial regularization to punish the spatial constraints. Danelljan *et al.* [9] improved tracking performance to a new level by learning continuous convolution filters for visual tracking. To further deal with the scale variation problem, three CF-based trackers, namely, the SAMF [26], DSST [27] and RAJSSC [28], have achieved good effects concerning the accuracy and real-time performance. With the recent development of increasingly more CF-based trackers [29]–[32], they have proved their good abilities with respect to their robustness. However, for severe occlusion and out-of-view challenges, these algorithms do not achieve the ideal performance.

B. TRACKING-BY-DETECTION

To alleviate the stability-plasticity dilemma with respect to online model update in visual tracking, Kalal *et al.* [33] proposed the mechanism by combining the tracking and detection, which helps perform long-term tracking. Ma *et al.* [18] decomposed the task of tracking into translation and scale estimation of object, and they also trained an online random fern classifier to re-detect objects in case of tracking failure. Hua *et al.* [34] trained an additional redetector for a significant geometric change of the object. As [35] stated, the observation model and feature extractor place important roles in visual tracking. Therefore, researchers have improved the performance of the feature extractor by selecting features [36], [37] and fusing the features [38]. Supancic and Ramanan [39] used self-placed learning to select reliable frames to extract additional training data as it progresses, which is more effective than a strong motion model. Overall, the tracking-by-detection mechanism is helpful for the long-term occlusion as well as the significant appearance change. In our proposal, we train an SVM detector based [21] to deal with long-term occlusion as well as the significant appearance changes.

C. JUDGMENT OCCLUSION BASED ON THE PSR

We use the PSR which is mainly proposed to evaluate the compressed radar signal after compression to assist in detecting occlusion [22], [40]. The PSR is calculated as follows:

$$PSR = \frac{max - \mu}{\sigma} \tag{1}$$

where *max* represents the maximum value of the main lobe peak, μ represents the mean value of the Gaussian response map and σ represents the standard deviation of the Gaussian response value. When a target is occluded by the background, the Gaussian response value of the target will have multiple peaks that are exceeded by the other peaks, which will result in tracking failure. The difference between the response map of two consecutive frames is very small, which implies that the temporal information of the video frames can help the tracker to more accurately locate the target. The temporal information between video frame sequences can be represented by the sensitivity value of the *PSR*.

$$S = \sum_{i}^{n} (PSR_i - P_n)^2 \tag{2}$$

$$P_n = \frac{1}{n} \sum_{i}^{n} PSR_i \tag{3}$$

where P_n means the average PSR of the response maps of n consecutive frames. Therefore, serious occlusion and partial occlusion are separated by the PSRs, and the sensitivity of the information of the temporal context is used to address the visual tracking problems except for serious occlusion. In this paper, we combine the judgment occlusion and the re-detection strategy for severe occlusion.

III. THE TRACKING METHOD USING PSR-BASED OCCLUSION JUDGMENT AND SVM RE-DETECTION

In this section, we will detail our method in four aspects: A) $\ell_1\ell_2$ loss function, B) occlusion judgment strategy, C) re-detection strategy, and D) implementation details.

A. $\ell_1\ell_2$ LOSS FUNCTION

Given the feature map, a correlation tracker aims at learning the filter weights to regress the Gaussian label. A classical model based on a correlation filter solves ridge regression problems as follows:

$$min_w \sum_i (f(x_{x_i}) - y_i)^2 + \lambda ||w||^2 \tag{4}$$

where $f(x_i)$ is the regression function obtained by training the mapping function $\varphi(x_i)$ of the feature space, $y_i \in R^{n\times n}$ is the desired Gaussian-shaped response, λ is the regularization term coefficient, and w is the filter template, $||\cdot||$ is the ℓ_2-norm. The goal is to find a function $f(x) = w^T \otimes x$, where $\otimes$ is the correlation operation, that minimizes the error between the regressions to target y_i and $f(x)$.

When the target appearance significantly changes, the error in some feature dimensions may be very large, that's the reason resulting in the instability of the mean square error. Therefore, to improve the performance of the CF-based method which is sensitive to local occlusion and allow large errors to occur when the appearance significantly changes in filter learning process, we replace the loss function in conventional CF-based method using $\ell_1\ell_2$-loss function with an appropriate sparsity [41]. Therefore, equation (4) is converted into the following formula:

$$min_w \sum_i (f(x_i) + e_i - y_i)^2 + \lambda ||w||_2^2 + \tau \sum_i \ell(e_i) \tag{5}$$

where the λ and τ are the weight parameters. Equation (5) can be split into two subproblems, both of which have globally optimal solutions. Therefore, the problem in (5) can be solved by alternately optimizing the two subproblems until the objective function values converge. Where e_i is the difference between the regression values and the expected response value. Its calculation formula is as follows:

$$e_i = \xi(\frac{\tau}{4+2\tau}, \frac{2}{2+\tau} F^{-1}(\hat{y} - \hat{\alpha} \odot \dot{k}_1)) \tag{6}$$

When we alternately optimize the two subproblems, we need to use the dual space to minimize $||f(X) + e - y||_2^2 + \lambda||\,|w|\,|_2^2$ according to w. We denote the dual conjugate of w as α, such that $w = \sum_i \alpha_i \varphi(x_i)$. The problem with respect to α has a closed-form solution denoted as $\hat{\alpha} = \frac{\hat{y}-\hat{e}}{\hat{k}_1+\lambda}$. $F^{-1}(\cdot)$ denotes the inverse Fourier transform, $\hat{y}$ is the expected response in the Fourier domain, $\odot$ is the Hadamard product, $\widehat{k_1}$ denotes the first row^2 of the kernel matrix K and ξ is a shrinkage operator, defined as:

$$\xi(\epsilon, x) = sign(x)max(0, |x| - \epsilon) \tag{7}$$

B. OCCLUSION JUDGMENT STRATEGY

In this paper, the PSR of the fusion response maps is used to judge the occlusion or out-of-view condition of each frame. Then, we process the frame according to the judgment result. When the target is severely occluded, the target appearance is mostly contaminated. At this moment, the PSR of the response map obtained by the correlation operation using the target appearance model is always low.

Over time, the target in Fig.1 (c) has been seriously occluded by the environment, as reflected by the PSR = 5.957 in the 109th frame. At this point, the peak around the target is prominent, even exceeding the target peak, which causes the tracker's prediction to finally contain a target position error, and thus tracking failure occurs. Therefore, the PSR can correctly reflect the state of the target and is also an effective indicator for judging whether the target is occluded.

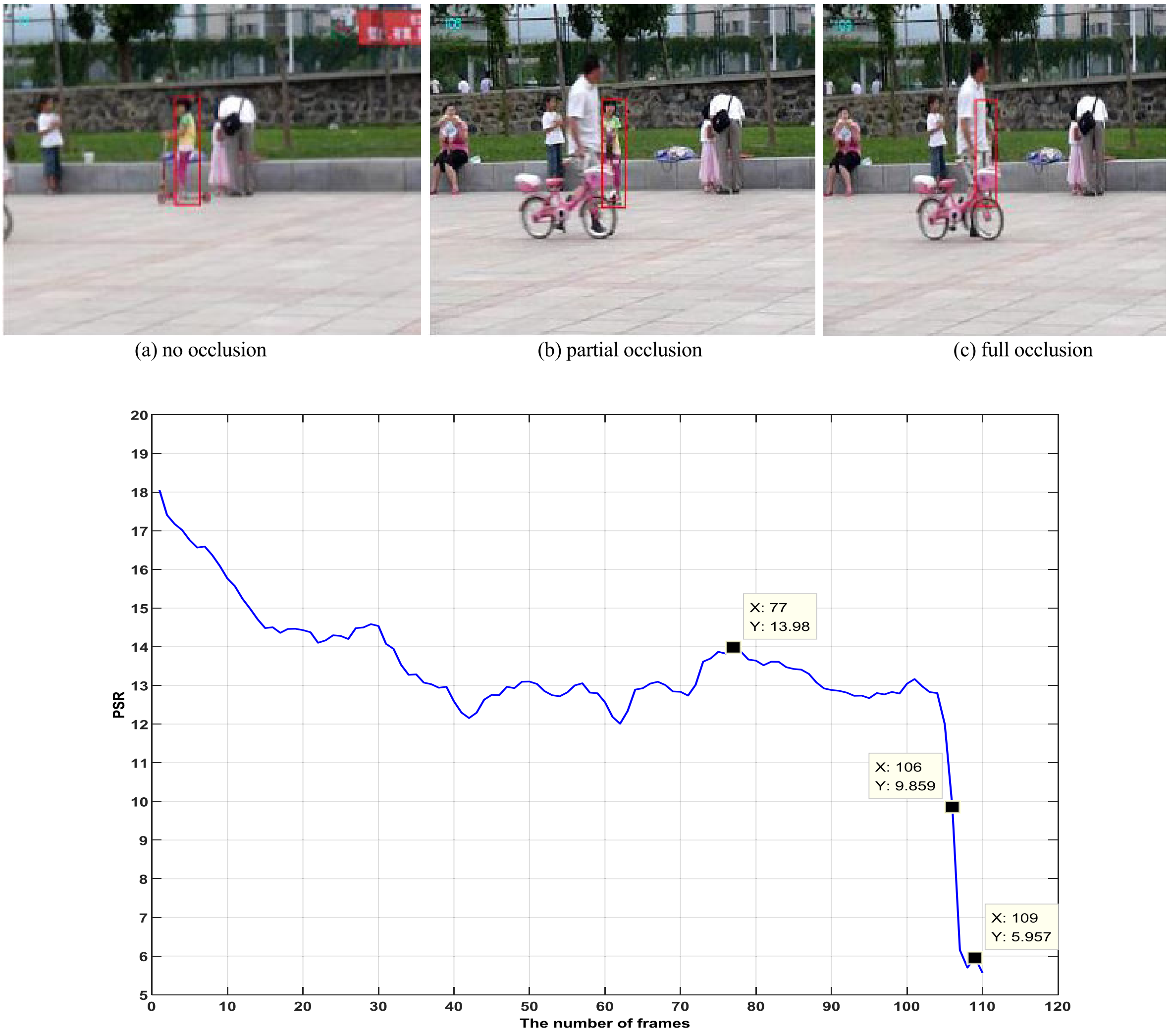
Due to the large variation range of the PSR, in order to find a better occlusion judgment criterion, this paper normalizes the PSR of the Gaussian expected response of the first frame of the video as a reference value, and the formula is as follows:

$$flag = \begin{cases} 1, & \dfrac{PSR_i}{PSR} < \eta \\ 0, & otherwise \end{cases} \tag{8}$$

where η is the similarity coefficient of the current frame's PSR and the reference PSR. When $flag = 0$, it is considered that the target being tracked is not occluded or is partially occluded; and when $flag = 1$, the target is severely occluded. At this moment, it is necessary to judge whether the target appears again and whether the model needs to be updated by the re-detection method.

C. REDECTETION STRATEGY

When the target is severely occluded or even fully occluded, the appearance information of the target is difficult to obtain. In this situation, it is not very reliable to use the tracker to



(a) no occlusion

(b) partial occlusion

(c) full occlusion

(d) The PSR of the Girl2 Sequence

FIGURE 1. In (a), the target is not occluded, and the PSR is always in a range of larger values, such as in the 77th frame when the PSR = 13.98. In (b) the target is partially occluded by the background, and the PSR = 9.859 in frame 106. Here, the peak of the response map begins to weaken, and there is a multi-peak situation, but the target peak has not been exceeded. Over time, in (c), the target has been seriously occluded by the background information, and the PSR = 5.957 in the 109th frame. In (d), the curve denotes the variation of the PSR through the girl2 sequence.

predict the position of the target. When the target appears again, the re-detection strategy can help tracker to detect the position where the target appears, that is, the object detection algorithm can determine whether the target is occluded, and can also obtain the candidate target position. The SVM is essentially a classifier that distinguishes between a target and the environment. Therefore, this paper uses an SVM classifier to judge if the target has reappeared in view and determine the candidate position of the target, which is very helpful for the subsequent tracking of the target occlusion.

D. IMPLEMENTATION DETAILS

The algorithm is mainly improved using LCT algorithm [18] and the Struck algorithm [21]. Here, the (x_0, y_0) is the target initial position; $target_{sz}$ denotes the target scale; X_t is the target status, which includes the position $(\hat{x}_t, \hat{y}_t)$ and scale $\hat{s}_\mathrm{t}$ at time t; CF_i is the appearance model; CF$_\mathrm{s}$ is the scale model; $\otimes$ is the correlation operator; $*$ is the multiplication operation; and HSvm denotes the SVM classifier. The whole algorithm consists of three parts: the prediction of the position of correlation filter model, the scale correlation filter model, and the SVM classifier.

The correlation filter model for the prediction of the target position uses different features for training and updating. The HOG features and the CN features of the target are first extracted and denoted as feats, which are used to train two correlation filters, then we get two response maps using different filters $CF_i(i = 1, 2)$. And we fuse them as follows:

$$r_t = \sum\nolimits_{i=1}^{n} m_i r_i \tag{9}$$

$$r_i = feats_i \otimes CF_\mathrm{i} \tag{10}$$

where r_t denotes the final fusion response map; r_i is the response maps obtained by the correlation operations of the two features and corresponding filters; $feat_i$ denotes i^{th} feature and CF_i is the corresponding model; m_i is the weight of r_i, calculated as follows:

$$m_i = 1 - \frac{d_i}{\sum_{j=1}^{2} d_j} \tag{11}$$

$$d_i = ||r - r_i \oplus \Delta|| \tag{12}$$

To get m_i, we calculate the d_i in (12), which is the difference between the expected response value and real response value of the i^{th} filter; $\oplus$ represents the shift operator; Δ denotes that translating the peak of r_i to the center of response map. We use the same method to get the weights calculated by the sensitivity method of the PSR are used to adaptively update the template. To obtain a more robust model, we use adaptive learning rate γ for per frame, such as follows:

$$\gamma_t = \gamma_{t-1} m_i \tag{13}$$

In this way, we can get a robust model. Subsequently, we get the accurate position prediction in the sequences. At the same time, the scale correlation filter is similar to the position correlation model, but the training of the model uses only the HOG features of the target. The number of scale pyramid layers is 33 layers, which is used to predict the scale of the target. The SVM detector is specifically designed to address the problem when the target is severely occluded. We minimize the convex objective to learn re-detector:

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{N} \xi_i \tag{14}$$

$$s.t.\ \forall i, \forall \mathrm{t} \neq t_i : < w, \delta\Phi_i(t) > \geq \delta(t_i, t) - \xi_i \tag{15}$$

$$\forall i, \quad \xi_i \geq 0 \tag{16}$$

where Φ is the kernel map; $\delta\Phi_{\mathrm{i}}(\mathrm{t}) = \Phi(\mathrm{feats_i}, \mathrm{t_i}) - \Phi(\mathrm{feats_i}, \mathrm{t})$; $(feats_i, t_i)$ specifies the correct transformation of the object; w is the weight vector learned by the SVM; C is the coefficient of ξ_i, following the Struck [21], we set $\mathrm{C} = 100$; ξ denotes the slack variable in SVM. As a matter of fact, the HSvm learns a map $f : F \times T \rightarrow R$, F is the feature space, T is the transformation space. In our HSvm re-detector, the $feats_i$ are features sampled from the reliable location for re-detection. The procedure of re-detection is as follows: First, the occlusion is judged by the PSR of the response map of the target in the current frame. Then, we get the candidate position of the target through the SVM detector. Finally, the response value of the optimal position detected by the SVM detector is compared with the response value of the position predicted by the correlation filter, which then determines whether the target is out of occlusion, if it is, we will locate the position. Therefore, the model obtained through iterative online incremental training has better robustness.

Algorithm 1 The Proposed Algorithm

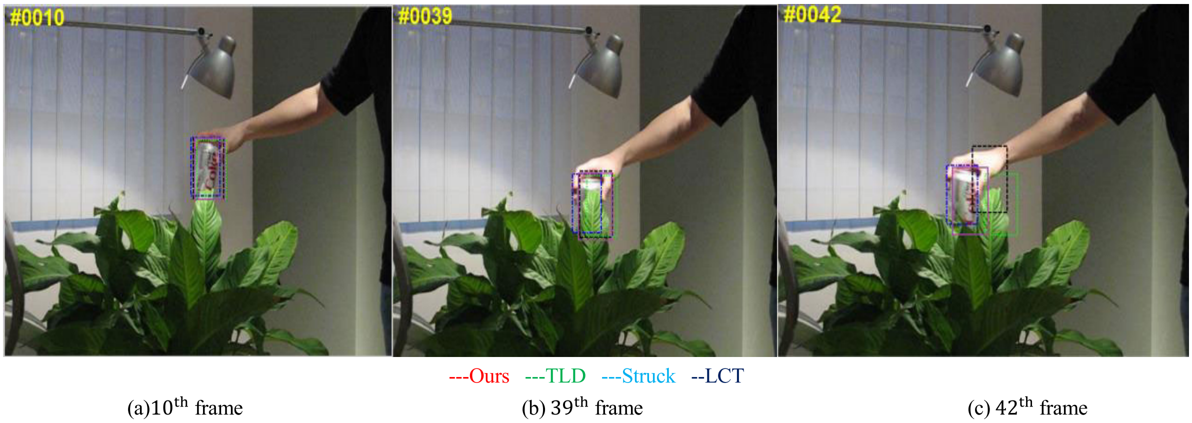
```
Input:$(x_0, y_0)$, target$_{sz}$.
Output: $X_t=(\widehat{x_t}, \widehat{y_t}, \widehat{s_t})$, $CF_i$, $CF_s$, $HSvm$.
Repeat:
  Extract the feats (HOG, CN) from $(x_{t-1}, y_{t-1})$ in
   the last frame.
  Yield response map:
    $r_i = feat_i \otimes CF_i, i = 1, 2$;
  Calculate the sensitivity S of the PSR by (2), the final
   position $(x_t, y_t)$ and final response value $r_t$ by (9);
  Obtain the target size:
    $\hat{s} = feats_{x_t,y_t} \otimes CF_s, s_t = target_{sz} * \hat{s}_t$;
  if flag then:
    $\mathrm{r_i}, (\mathrm{x_i}, \mathrm{y_i}) = \mathrm{HSvm(feats)}$;
    $\mathrm{r_{max}}, (x_{\max}, y_{max}) = \max(\mathrm{r_i}, (\mathrm{x_i}, \mathrm{y_i}))$;
    if $\mathrm{r_{max}} > \mathrm{r_t}$ then:
      Update $\mathrm{CF_i}$;
    else:
      $(\mathrm{x_t}, \mathrm{y_t}) = (\mathrm{x_{t-1}}, \mathrm{y_{t-1}})$;
    end
  end
  Update CFs and HSvm;
Until end of video frame sequence;
```

IV. EXPERIMENT RESULTS AND ANALYSIS

To prove the robustness and real-time performance of our proposed algorithm in the case of severe occlusion, we select six sets of video frame sequences with severe occlusion in the experimental verification of our algorithm. The results are compared and analyzed with those of two long-term tracking algorithms (TLD: tracking learning detection algorithms, and LCT: long-term correction tracking) and the Struck (structured output tracking with kernels) algorithm. The six test video frame sequences are coke, tiger2, girl, basketball, lemming, and liquor, which are 640 × 480, 640 × 480, 128 × 96, 576 × 432, 640 × 480, and 640 × 480 pixels, respectively. The number of frames is 4,958. The proposed algorithm is implemented using MATLAB on a win10 x64 computer with a 3.60 GHz i7 processor and 8GB of memory.

A. EXPERIMENTAL RESULTS

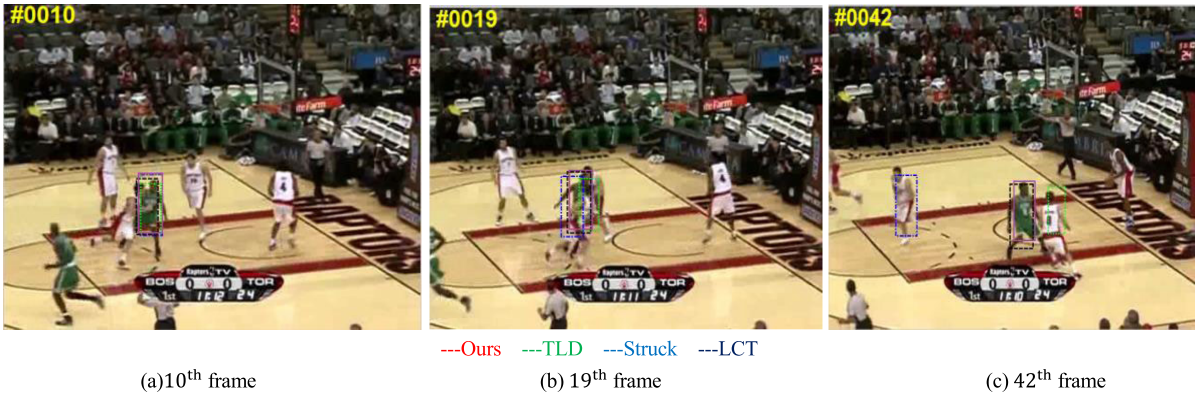
Experiment 1: This experiment provides the results of different algorithms when the moving target is occluded. To verify the tracking performance of the algorithm when the moving target is occluded, the standard test video frame sequence "coke" is used for testing. The results of each algorithm are shown in Fig.2. The TLD, Struck, and LCT algorithms and ours can correctly predict the position of the target in Fig.2 (a) in which the target is not occluded. In Fig.2 (b), the moving target is severely occluded by the leaves. The TLD algorithm experiences a tracking failure, and the detection module scans the entire image via a cascade classifier consisting of a variance classifier, a random fern classifier, and a KNN classifier. Since the target is mostly occluded by



---Ours ---TLD ---Struck --LCT

(a)10th frame (b) 39th frame (c) 42th frame

FIGURE 2. **(a) shows the state of the target when it is not occluded by the environment. The TLD, Struck, LCT and proposed algorithms can correctly predict the position of the target. In (b), the moving target is severely occluded by the leaf. Since the target is mostly occluded by the environment, and the detection time is long, the confidence of the target position is low. In (c), both the TLD and LCT drift, and the Struck algorithm and Ours can locate the target well.**



---Ours ---TLD ---Struck ---LCT

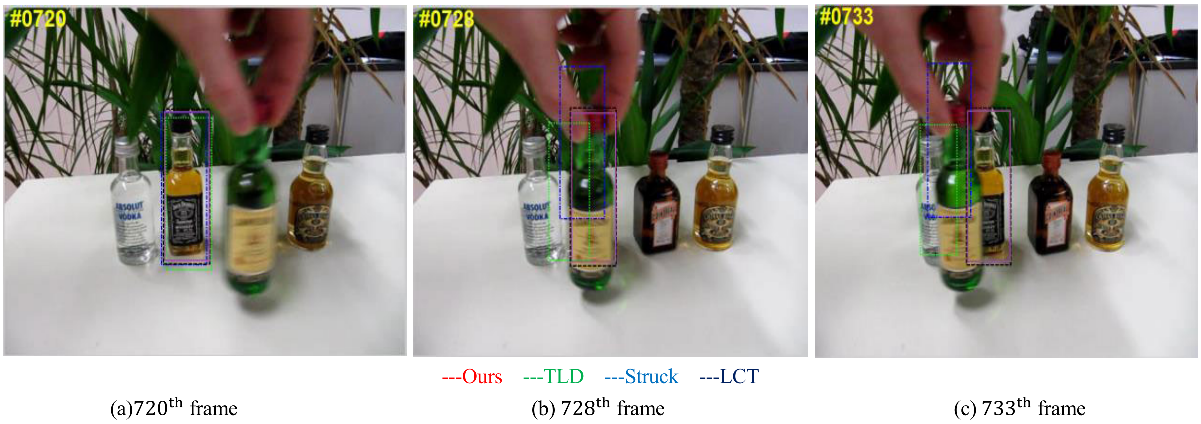
(a)10th frame (b) 19th frame (c) 42th frame

FIGURE 3. **Comparative test of the deformed target is occluded. (a) shows the state of the target when it is not occluded by the environment. In (b), the target is deformed due to the posture change. In (c), the tracking target appears again, and the tracker treats the wrong moving object as the tracking target, resulting in a target tracking drift.**

the environment and the detection time is so long, the confidence of the target position is very low. The Struck algorithm uses the prediction function learned by the online structure output SVM to predict the change of the target position and avoids the intermediate classification phase. The LCT algorithm sets a fixed threshold according to the response value of the target appearance model should determine whether the target is occluded and whether the target appearance model is to be updated. In this section, our algorithm uses the PSR of the current frame of the response map to judge whether the target is severely occluded for re-detection. The model judges whether the target is partially occluded prior to adaptive update according to the peak value of the current frame. In Fig.2 (c), both the TLD and LCT algorithms drift, and the Struck algorithm and ours can locate the target well.

Experiment 2: The target with deformation is occluded. To verify the performance of the proposed algorithm when the moving target is deformed, the video frame sequence "basketball" is used for testing. Fig.3 (a) shows the state of the target when it is not occluded by the environment. The TLD, Struck, LCT, and proposed algorithms can correctly locate the target. When the target is in the 19th frame, the target is deformed due to the pose change. Besides, it is severely occluded by another player's body. The TLD algorithm uses the pyramid optical flow algorithm to track the object to predict the target's motion direction. Due to the interference of the occluded object's motion, when the target appears in Fig.3 (c) again, the tracker treats the wrong moving object as the tracking target, resulting in a target tracking drift. The Struck algorithm uses the prediction function to predict the change of the target position between the current frame and the previous frame. Since the target is severely occluded, when the target reappears in Fig.3 (c), the accumulation of the position change error causes the tracker to no longer adapt to subsequent tracking, eventually resulting in tracking failure. The LCT algorithm utilizes the spatial-temporal context, and the model tracks the target. Therefore, when in Fig.3 (c) 42^{th} frame is reached, both the LCT algorithm and ours can accurately track the target.

Experiment 3: The stationary target is occluded. To verify the performance of the proposed algorithm when the stationary target is occluded, the test video frame sequence "liquor"



(a)720th frame (b) 728th frame (c) 733th frame

FIGURE 4. Contrast test when the stationary target is occluded. In (a), the 720th frame shows that the target is not occluded by the environment. In (b), the target to be tracked is completely occluded by the background information, eventually losing the features associated with the tracker, causing the tracker to fail to find the target. The target is out of occlusion in (c).

is used for the test. The results of each algorithm are shown in Fig.4. (a) show that the target is not occluded by the environment. The TLD, Struck, LCT, and our proposal can correctly track the target. When tracking in the 728th frame in (b), the target to be tracked is completely occluded by the background information, eventually losing the features associated with the tracker, and causing the tracker to fail to find the target. The TLD and the Struck algorithms have not yet determined that the target is occluded, and so the occluded object is mistakenly regarded as the tracking target. Conversely, the LCT and the proposed algorithms successfully determine that the tracking target is occluded. In the 733th frame (the target is out of occlusion), the TLD and the Struck algorithms continue to erroneously track the wrong object, causing severe drift, which eventually leads to a tracking failure. The LCT algorithm and our algorithm can accurately track the target again.

TABLE 1. Success rate and accuracy. The first column is the sequence of the video frames and the total number of frames. The following columns are the success rate and accuracy of the different algorithms in different video frame sequences, where red represents the best result, and blue is the SECOND-BEST result. "–" indicates that the algorithm's result in the video frame sequence is too bad, and so the data will be ignored during the statistical analysis to avoid any adverse effects on the overall results.

VIDEO FRAME		TLD	STRUCK	LCT	OURS
COKE (291)	OP	39.9	66.5	65.1	70.9
	DP	56.3	75.4	79.7	89
TRIGER2(356)	OP	27	54.3	61.8	69
	DP	41.1	60.5	68.5	66.4
GIRL (500)	OP	56.6	73.4	66.3	99.8
	DP	80.5	94	89	100
BASKETBALL (725)	OP	--	20.6	75.1	70.3
	DP	--	22.4	90.9	99.2
LEMMING (1336)	OP	52.9	47.8	71.1	80.3
	DP	75.1	54.1	78.7	80.1
LIQUOR (1741)	OP	51.4	40.3	57.4	56.4
	DP	54.6	37.5	69.1	70.4

B. COMPARATIVE ANALYSIS OF EXPERIMENTAL RESULTS

To better illustrate the superiority of the proposed algorithm, we compare it with related algorithms. The proposed algorithm, TLD, Struck, and LCT are tested using 6 different video frame sequences, and then we compare the results of the different algorithms for each sequence of video frames. We do not make any modifications to the parameters of the existing algorithms. The overlap precision (OP) and the distance precision (DP) of the above four different algorithms are shown in Table 1.

We follow the common tracking and judging criteria according to the target. 1) Success rate: First, the overlap rate (score) is calculated according to each frame's predicted area ($\mathrm{ROI_t}$) and manually labeled area ($\mathrm{ROI_{gt}}$).

$$score = \frac{ROI_t \bigcap ROI_{gt}}{ROI_t \bigcup ROI_{gt}} \tag{17}$$

Then, after setting different overlapping thresholds, the success rate under the different overlapping thresholds is statistically calculated. Finally, the final success rate (the mean) is obtained by calculating the area under the curve or area under curve (AUC). 2) Accuracy: First, calculate the weighted average of the distance between the prediction center position and the ground truth position, which is the center location error (CLE). Then, calculate the accuracy according to the different error thresholds. Finally, the position error being no greater than 20 pixels is taken as the final precision of the precision map. Table 1 shows that the overall tracking performances of the TLD and the Struck algorithms are not so good, and the proposed algorithm's tracking performance is the best, followed by LCT. In the 6 video frame sequences, the TLD algorithm performs well only in the girl sequence because the other 5 video frame sequences are also affected by different factors such as deformation, illumination, scale, etc. The TLD algorithm uses the pyramid optical flow method that cannot conduct accurate visual tracking in complex scenes, and so the robustness is not so good. The Struck algorithm uses only Haar to calculate the integral map of the feature, and cannot stably predict the position of the target under the influence of various factors, especially when the target is occluded. Struck still takes the predicted value with the highest probability as the correct location, and the accumulation of wrong tracking information eventually leads to errors. The LCT algorithm judges the occlusion of the target through the response value of the target appearance model, which can avoid the wrong learning of the model to some extent. Therefore, it can stably track the target in a long target tracking time. However, since the LCT uses only the gray feature of the target and its illumination invariant gray feature to train, when the target pose changes or there is similar object interference (such as in the tiger2 and liquor video frame sequences), the tracking performance is not ideal. Our approach uses the idea of LCT to train using multiple features fusion and uses the SVM re-detection strategy to track the target in a complex scene with occlusion. Compared with LCT, the comprehensive performance has a certain improvement, which is mainly due to the selection of the target features and the PSP-based occlusion judgment strategy in the model training process.

TABLE 2. Center location error and FPS. The proposed algorithm has the smallest center position error in the "coke", "tiger2", "girl", "basketball", and "lemming" video frame sequences, and the effect for the "liquor" sequence is worse than that of the LCT algorithm. However, the average center position error is the smallest at 11.91 pixels. In "tiger2", the FPS is the largest at 13 FPS. In "coke" and "basketball", the FPS of the LCT algorithm is the largest, and it is followed by the proposed algorithm. In "girl", "lemming", and "liquor", the FPS of TLD is the largest, and the proposed algorithm of this paper is second. the LCT has the largest average frame rate at 19.84 fps, and our algorithm in this paper is the second at 16.68 fps. Therefore, the comprehensive performance of the proposed algorithm is the best.

VIDEO FRAME		TLD	STRUCK	LCT	OURS
COKE	CLE	25.08	13.7	12.08	10.72
	FPS	17.40	11	20.02	17.6
TRIGER2	CLE	37.10	21.64	16.19	15
	FPS	12.64	8.87	12.84	13
GIRL	CLE	9.79	2.57	5.10	2.15
	FPS	20.06	9.01	19.56	19.60
BASKETBALL	CLE	213.86	118.26	5.06	4.96
	FPS	19.81	7.38	37.68	20.20
LEMMING	CLE	15.99	37.75	15.64	11.60
	FPS	10.60	7.35	8.11	9.09
LIQUOR	CLE	37.58	90.99	26.72	27
	FPS	21.22	6.57	20.66	20.60
MEAN	CLE	56.57	47.49	13.47	11.91
	FPS	16.96	8.36	19.84	16.68

Table 2 compares the CLE and the FPS (Frames Per Second) of four different algorithms. (The real-time performance of the algorithm is calculated using the total time consumption of the tracking algorithm and the total number of frames of the video sequences). The larger the FPS is, the higher the real-time performance of the algorithm. The first column of Table 2 represents different video frame sequences, the other columns represent the CLE and the FPS of the different algorithms, and last row represents the average CLE and average FPS of the different algorithms. The best result for each line is marked in red, and the second-best result is marked in blue. Table 2 provides the qualitative analysis of the TLD, Struck, and LCT algorithms, and ours.

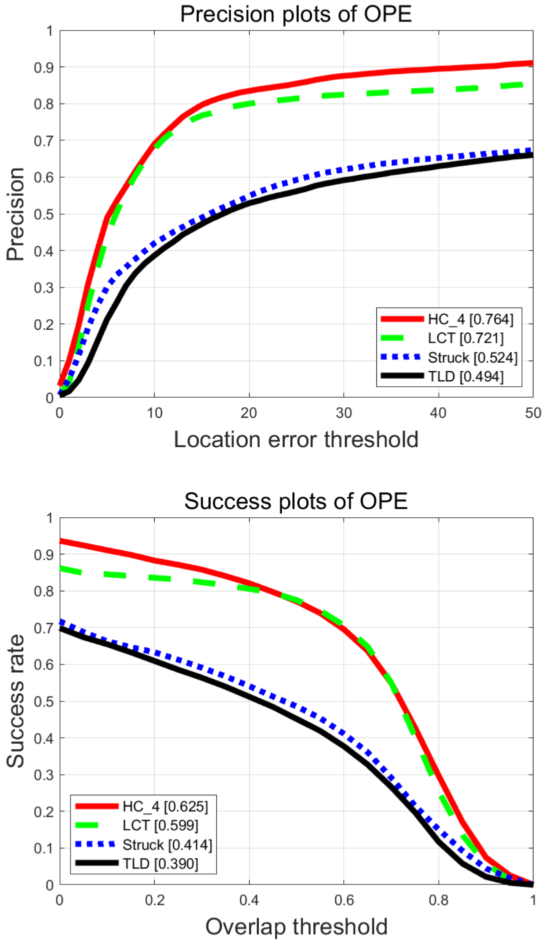


FIGURE 5. Success and accuracy maps for TLD, Struck, LCT, and Ours (HC_4) on 34 selected sequences.

To better illustrate the tracking performance of the proposed algorithm in various complex scenarios, this paper quantitatively analyzes the selected 34 video frame sequences. The experimental results are shown in Fig.5. In Fig.5, we can see that compared with other tracking algorithms, the success rate and accuracy of the proposed algorithm are the highest at 0.625 and 0.764, respectively; therefore, the robustness of the proposed algorithm is the best.

C. STATE-OF-THE-ART COMPARISON ON OTB100

To be more persuasive, we compare our algorithm with other state-of-the-art methods related to this paper on all sequences in OTB100. It has 100 video test sequences, which contain 11 challenges, including illumination variance, scale variation, occlusion, deformation, motion blur, fast motion, in-plane rotation, out-of-plane rotation, out-of-view, background clutter and low resolution. We follow the two criteria, the success rate and the precision, which are described in detail in Section IV.B. To evaluate the performance, we compare our algorithm with 7 state-of-the-art tracking methods relevant to our proposal, including LMCF [42], SRDCF [25], Staple [7], STAPLE_CA [43], Struck [21], TLD [33], LCT [18]. In general, the precision and success rate of our

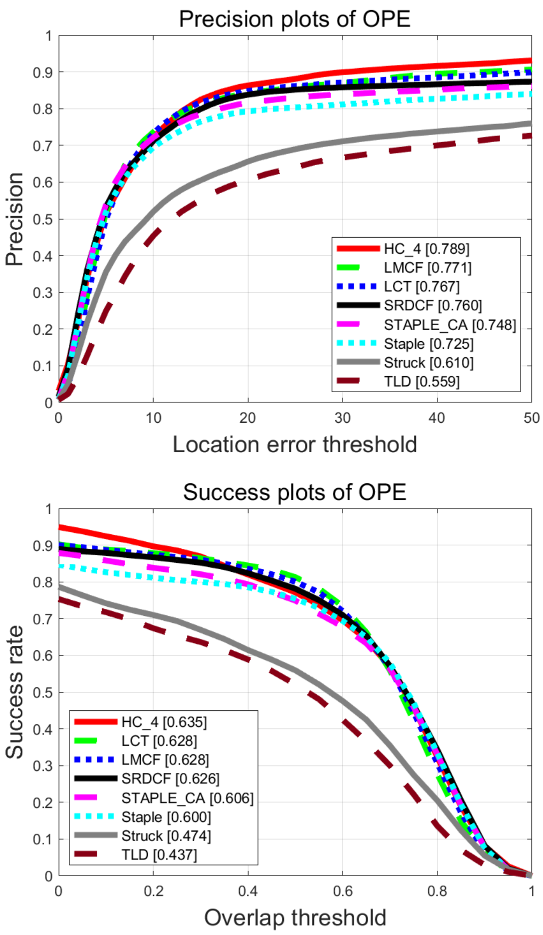


FIGURE 6. **Comparison of the performance of Ours (HC_4) with other algorithms under all sequences of OTB100. The top is the precision plot and the bottom is the success rate plot.**

algorithm are 1.8% and 0.7% higher than those of the second best algorithms in terms of the two indices. In the comparison with Struck, our algorithm achieves 17% and 16.1% improvements in precision and success rate, respectively. Compared with LCT, we can see that our algorithm obtained 2.2% and 0.7% improvements in precision and success rate, respectively. As shown in Fig.6, our method is superior to the compared algorithms in both precision and success rate under all sequences of OTB100, which demonstrates the highly competitive performance of our proposal.

V. CONCLUSION

In this paper, we propose an occlusion judgment tracker based on the CF framework. To solve the instability of the algorithms based on the single feature in complex scenes, we extract the HOG and CN features to train the model and track the target. Therefore, our algorithm has the advantage that the traditional method does not have before and after occlusion. Furthermore, we introduce the $\ell_1\ell_2$ loss function to reduce the sensitivity of CF-based methods to local occlusion. Next, we propose an adaptive online model update strategy based on the sensitivity value S of the PSR to get a robust appearance model. As a complement of the target out-of-view, the PSR is used to determine whether the target is severely occluded, and then detect the disocclusion by the SVM. According to the experimental results, compared with the existing related algorithms, our algorithm has certain advantages and robust performance. However, with respect to deep learning-based tracking algorithms, the accuracy of our algorithm needs to be improved. This is mainly because the traditional features are not comprehensive enough in the training and learning of the model.

REFERENCES

[1] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Learning discriminative model prediction for tracking," in *Proc. ICCV*, Oct. 2019, pp. 6182–6191.

[2] Q. Wang, J. Gao, J. Xing, M. Zhang, and W. Hu, "DCFNet: Discriminant correlation filters network for visual tracking," 2017, *arXiv:1704.04057*. [Online]. Available: http://arxiv.org/abs/1704.04057

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Jun. 2016, pp. 770–778.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[5] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ATOM: Accurate tracking by overlap maximization," in *Proc. CVPR*, Jun. 2019, pp. 4660–4669.

[6] M. Danelljan, F. S. Khan, M. Felsberg, and J. V. D. Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. CVPR*, Jun. 2014, pp. 1090–1097.

[7] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. CVPR*, Jun. 2016, pp. 1401–1409.

[8] J. Chu, X. Tu, L. Leng, and J. Miao, "Double-channel object tracking with position deviation suppression," *IEEE Access*, vol. 8, pp. 856–866, 2020.

[9] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. ECCV*, Oct. 2016, pp. 472–488.

[10] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. CVPR*, Jun. 2019, pp. 4665–4674.

[11] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognit.*, vol. 76, no. 1, pp. 323–338, Apr. 2018.

[12] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. CVPR*, Jul. 2017, pp. 6638–6646.

[13] M. Kristan *et al.*, "The seventh visual object tracking VOT2019 challenge results," in *Proc. ICCV*, 2019, pp. 2206–2241.

[14] H. Fan, F. Yang, P. Chu, L. Yuan, and H. Ling, "TracKlinic: Diagnosis of challenge factors in visual tracking," 2019, *arXiv:1911.07959*. [Online]. Available: http://arxiv.org/abs/1911.07959

[15] Y. Li, J. Zhu, and S. C. H. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *Proc. CVPR*, Jun. 2015, pp. 353–361.

[16] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. ICCV*, Oct. 2017, pp. 1135–1143.

[17] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. CVPR*, Jun. 2010, pp. 2544–2550.

[18] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proc. CVPR*, Jun. 2015, pp. 5388–5396.

[19] B. Yan, H. Zhao, D. Wang, H. Lu, and X. Yang, "'Skimming-perusal' tracking: A framework for real-time and robust long-term tracking," in *Proc. ICCV*, Oct. 2019, pp. 2385–2393.

[20] T. Zhu, J. Chu, and J. Miao, "A long-term tracking model based on tracking failure detection strategy and weighted random forest," in *Proc. ICMITE*, 2016, pp. 179–191.

[21] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. S. Torr, "Struck: Structured output tracking with kernels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2096–2109, Oct. 2016.

[22] X. Lu and H. Sun, "Parameter assessment for SAR image quality evaluation system," in *Proc. APSAR*, 2007, pp. 58–60.
[23] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proc. ECCV*, Oct. 2012, pp. 702–715.
[24] 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.
[25] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. ICCV*, Dec. 2015, pp. 4310–4318.
[26] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. ECCV*, Sep. 2014, pp. 254–265.
[27] 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.
[28] M. Zhang, J. Xing, J. Gao, X. Shi, Q. Wang, and W. Hu, "Joint scale-spatial correlation tracking with adaptive rotation estimation," in *Proc. ICCV*, Dec. 2015, pp. 595–603.
[29] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. CVPR*, Jun. 2018, pp. 4904–4913.
[30] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. CVPR*, Jun. 2018, pp. 4844–4853.
[31] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Joint group feature selection and discriminative filter learning for robust visual object tracking," in *Proc. ICCV*, Oct. 2019, pp. 7950–7960.
[32] U. Kart, A. Lukezic, M. Kristan, J.-K. Kamarainen, and J. Matas, "Object tracking by reconstruction with view-specific discriminative correlation filters," in *Proc. CVPR*, Jun. 2019, pp. 1339–1348.
[33] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.
[34] Y. Hua, K. Alahari, and C. Schmid, "Occlusion and motion reasoning for long-term tracking," in *Proc. ECCV*, Sep. 2014, pp. 172–187.
[35] N. Wang, J. Shi, D.-Y. Yeung, and J. Jia, "Understanding and diagnosing visual tracking systems," in *Proc. ICCV*, Dec. 2015, pp. 3101–3109.
[36] L. Leng, J. Zhang, M. K. Khan, X. Chen, and K. Alghathbar, "Dynamic weighted discrimination power analysis: A novel approach for face and palmprint recognition in DCT domain," *Int. J. Phys. Sci.*, vol. 5, no. 17, pp. 2543–2554, Dec. 2010.
[37] L. Leng, J. Zhang, J. Xu, M. K. Khan, and K. Alghathbar, "Dynamic weighted discrimination power analysis in DCT domain for face and palmprint recognition," in *Proc. ICTC*, Nov. 2010, pp. 467–471.
[38] L. Leng, M. Li, C. Kim, and X. Bi, "Dual-source discrimination power analysis for multi-instance contactless palmprint recognition," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 333–354, Jan. 2017.
[39] J. S. Supancic, III, and D. Ramanan, "Self-paced learning for long-term tracking," in *Proc. CVPR*, Jun. 2013, pp. 2379–2386.
[40] Y. Yuan, J. Chu, L. Leng, J. Miao, and B. G. Kim, "A scale adaptive object tracking algorithm with occlusion detection," *EURASIP J. Image Video Process.*, vol. 7, pp. 1–15, Dec. 2020.
[41] Y. Sui, Z. Zhang, G. Wang, Y. Tang, and L. Zhang, "Real-time visual tracking: Promoting the robustness of correlation filter learning," in *Proc. ECCV*, Oct. 2016, pp. 662–678.
[42] M. Wang, Y. Liu, and Z. Huang, "Large margin object tracking with circulant feature maps," in *Proc. CVPR*, Jul. 2017, pp. 4024–4029.
[43] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. CVPR*, Jul. 2017, pp. 1396–1404.

SHAOMING LI received the B.S. degree from Sichuan Agricultural University, Ya'an, China, in 2018. He is currently pursuing the M.S. degree with Nanchang Hangkong University, Nanchang, China. His research interests include computer vision, image processing, and machine learning.

JUN CHU received the Ph.D. degree from Northwestern Polytechnic University, Xi'an, China, in 2005.

She was a Postdoctoral Researcher at the Exploration Center of Lunar and Deep Space, National Astronomical Observatory, Chinese Academy of Sciences, from 2005 to 2008. She was a Visiting Scholar at the University of California, Merced, USA. She is currently the Director of the Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition and a Full Professor with the School of Software, Nanchang Hangkong University. Her research interests include computer vision and pattern recognition. She is a member of the Computer Vision Special Committee, China Computer Federation.

GUOCHONG ZHONG received the M.S. degree from Nanchang Hangkong University, Nanchang, China, in 2018. His research interests include computer vision and pattern recognition.

LU LENG (Member, IEEE) received the Ph.D. degree from Southwest Jiaotong University, Chengdu, China, in 2012.

He performed Postdoctoral Research at Yonsei University, Seoul, South Korea, and the Nanjing University of Aeronautics and Astronautics, Nanjing, China. He was a Visiting Scholar at West Virginia University, USA. He is currently an Associate Professor with Nanchang Hangkong University and also a Visiting Scholar with Yonsei University. He has published more than 70 international journal articles and conference papers. He has been granted several scholarships and funding projects for his academic research. His research interests include image processing, biometric template protection, and biometric recognition.

Dr. Leng is a member of the Association for Computing Machinery (ACM), the China Society of Image and Graphics (CSIG), and the China Computer Federation (CCF). He is a Reviewer of several international journals and conferences.

JUN MIAO received the Ph.D. degree from Nanchang University, Nanchang, China, in 2015.

He is currently a Researcher of the Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition and an Associate Professor with the School of Aeronautical Manufacturing Engineering, Nanchang Hangkong University. His research interests include computer vision, 3D reconstruction, and pattern recognition.