IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 20, 2020, accepted April 16, 2020, date of publication April 27, 2020, date of current version May 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990410

Target Tracking Method Based on Adaptive Structured Sparse Representation With Attention

JIE WANG, SHIBIN XUAN, HAO ZHANG, AND XUYANG QIN
School of Artificial Intelligence, Guangxi University for Nationalities, Nanning 530006, China

Corresponding author: Shibin Xuan (xuanshibin@gxun.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61866003.

ABSTRACT Considering the problems of motion blur, partial occlusion and fast motion in target tracking, a target tracking method based on adaptive structured sparse representation with attention is proposed. Under the framework of particle filtering, the performance of high-quality templates is enhanced through an attention mechanism. Structure sparseness is used to build candidate target sets and sparse models between candidate samples and local patches of target templates. Combined with the sparse residual method, reconstruction error is reduced. After optimally solving the model, the particle with the highest similarity is selected as the prediction target. The most appropriate scale is selected according to the multiscale factor method. Experiments show that the proposed algorithm has a strong performance when dealing with motion blur, fast motion, partial occlusion.

INDEX TERMS Attention mechanism, sparse representation, structure sparse, target tracking.

I. INTRODUCTION

Target tracking automatically locates a target in subsequent frames according to the state of a known target in the initial image frame. Target tracking, as one of the research hotspots in the field of computer vision, plays an important role in the fields of intelligent transportation, medical, military, and intelligent surveillance. According to the methods established by the target observation model, the target tracking methods [1]–[5] can be divided into two categories: discriminative methods and generative methods. The discriminative method establishes an observation model for the foreground and background of the initial frame image and determines the background and target information in subsequent video frames to achieve target tracking. Discriminative methods mainly include correlation filtering methods [6], [7], [26]–[29] and deep learning methods [8], [9], [30]–[37]. The generative method represents the target through the learned appearance model and selects the candidate patch with the smallest reconstruction error as the target area of the next frame. Generative methods mainly include sparse representations [10], [11], [44], mean shift [12], [13], and particle filtering [14]. Although these methods have been proven to achieve good results, they still face challenges from occlusion, deformation, scale change, fast motion, motion blur, lighting change, and background change.

The sparse representation of the image is based on the over-complete dictionary theory proposed by Mallat and Zhang [15] in 1993. Since sparse representation was applied in the field of target tracking by Mei and Ling [16] in 2009, the method of sparse representation has been proven to apply to target tracking. References [17]–[22], [43]–[48] used local, global, or joint sparse models to classify trackers. In [17], [22], the template T represents each target candidate area xi by means of sparse linear combination and uses a dynamic update method to describe the appearance model of the target. Although this type of method has achieved good results, when it encounters occlusion situations, the tracking efficiency decreases sharply because of the global sparse model. In [11], the target candidate region was divided into k patches of the same size x_i^k, which are represented by sparse templates. Reference [16] represented local patches in candidate regions as linear combinations of dictionaries by solving the l_1 minimization problem. Most of these methods are based on static local sparseness. Once similar objects appear in the scene or the target is occluded, the tracking target is easily lost. Zhang *et al.* [24] proposed a tracking algorithm based on a structural sparse appearance model and a particle filtering framework to represent particles and the corresponding local patches jointly.

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

In this paper, there are two main contributions. (1) We propose a novel sparse representation model by combining the structured sparse representation and sparse residuals and add an attention mechanism into the model. The model can significantly improve the performance and reliability of the algorithm. (2) The kernel density characteristics method is used to deal with the problem of target scale change during target tracking. The experiment proves that the strategy is effective.

II. RELATED WORKS

For the convenience of subsequent descriptions, a brief review of the work related to this article is given in this section.

A. STRUCTURAL SPARSE REPRESENTATION TARGET TRACKING MODEL

Mei and Ling [16] proposed robust tracking based on the l_1 norm minimization. The tracking problem was solved as a sparse approximation problem in the particle filtering framework. Each candidate target is represented by the sparseness in the target template set and trivial template set. In addition, l_1 regularized least squares is often used to solve the sparse problem of candidate targets, and then selecting the tracking target with the smallest reconstruction error as the location of the target in the next frame. On this basis, Xu *et al.* [23] proposed a tracking method based on a structured sparse appearance model, which divided the image into n patches of equal size, each local patch represented a fixed part of the target object, and all local patches represented the overall structure of the target.

However, this sparse model has the following disadvantages: (i) Though the l_1 norm can make the coefficients sufficiently sparse, under the interference of complex backgrounds and lighting changes, the sparse assumption is often not true, and the l_1 norm requires higher computational complexity. (ii) Its inability to explore the correlation between different particles is another deficiency, which will have a considerable impact on the robustness of the model.

B. ROBUST TARGET TRACKING BASED ON SPARSE REPRESENTATION

Zhang *et al.* [24] proposed a novel object tracking method of structural sparse representation, which not only makes full use of the inherent relationship between candidate targets and their local patches and learns their joint sparse representation but also preserves the spatial layout in local patches within each candidate target structure, and improves tracking performance by using the internal relationship between particles.

However, when this algorithm encounters the problem of target deformation, the gray features are extremely sensitive to these scenes, the correlation between column vectors will be affected, and it is doubtful whether the coefficients are still sparse. Moreover, this algorithm also lacks effective strategies for dealing with fast movements. Because it cannot adaptively adjust the window size, it learns from the noise as a target when the target changes in scale.

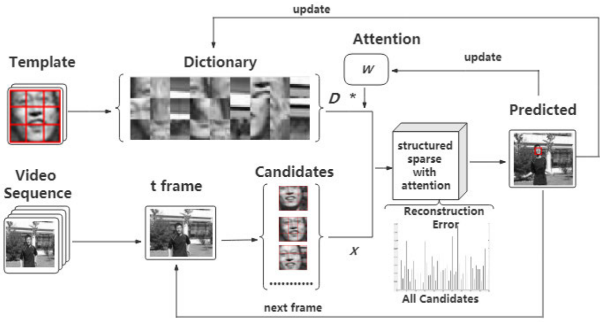


FIGURE 1. **Algorithm main structure.**

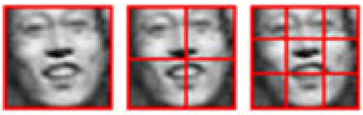


FIGURE 2. ***n* × *n* sized subpatches.**

III. OUR APPROACH

The sparse structured target tracking method utilizes the relationship between the local patches of candidate targets and retains the spatial layout between the local patches of each candidate target to improve the robustness of the algorithm. In the prototype-based tracking algorithm [25], the base vectors are orthogonal, so the coefficients Z corresponding to the orthogonal base vectors are dense. The above model can be solved by iterative optimization.

Based on the advantages and disadvantages of the above two methods, this paper combines structural sparse representation and prototype-based sparse representation and adds an attention mechanism to optimize the objective function to improve the robustness of the algorithm. Finally, the multi-scale factor method is used to solve the problem of target scale change. The algorithm main structure is shown in Fig. 1.

A. STRUCTURAL SPARSE REPRESENTATION MODEL COMBINING AN ATTENTION MECHANISM

In this paper, the target template is selected according to the method in [24]. The target object image in the specified frame is divided into $n \times n$ sized subpatches (Fig.2). All subpatches are vectorized and combined to the target template D. The model in [24] can be described as (1). Z is obtained by solving (1) with the help of the Lagrange multiplier method.

$$\begin{aligned} &\min_{Z} \|X - DZ\|_F^2 + \lambda_1 \|P\|_{2,1} + \lambda_2 \left\|Q^T\right\|_{2,1} \\ &s.t. Z = P + Q, \quad Z = [Z^1, Z^2, \ldots, Z^K] \end{aligned} \tag{1}$$

The sparse residuals used in [25] effectively reduce the reconstruction errors in the tracking process. On this basis, to improve the performance of high-quality templates on targets, an attention mechanism [41], [42] is added to (2), and formula (3) is obtained.

$$X = DZ + e \tag{2}$$

Z is a sparse representation coefficient, and e is a reconstruction error.

$$X = WDZ + e \tag{3}$$

where W represents the weight of the template. For the output y at a certain moment, W represents its attention on each part of the input x, that is, the weight of the contribution of each part of the input x to the output at a certain moment. The template D^k with stronger performance ability for the target in the current frame is given a higher weight W_{high}, which improves the performance of the target template in subsequent frames and enhances the robustness of the algorithm. Finally, considering the sparseness constraint problem of global images and local image patches, this paper considers using a combination of structural sparseness methods to enhance the sparseness of coefficient Z and make full use of the inherent relationship between candidate targets. Thus, combining (1) and (3), we propose a new target tracking model, as shown in (4).

$$\begin{aligned} &\min_{Z,P,Q,W,e} \frac{1}{2} \sum_{k=1}^{K} (\left\| X^k - W^k D^k Z^k - e^k \right\|_F^2 + \lambda_1 \left\| P \right\|_{p,q} \\ &\quad + \lambda_2 \left\| Q^T \right\|_{p,q} + \lambda_3 \left\| W \right\|_2 + \lambda_4 \left\| e \right\|_1) \\ &s.t. Z = P + Q \end{aligned} \tag{4}$$

X^k consists of the k-th patch of all n candidate targets, D^k represents the target template of the k-th patch, Z is the local observation representation of the k-th patch of the target template, W^k represents the weight coefficient of D^k, and e^k represents the k-th reconstruction error of the patch. $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ represents a nonnegative parameter of the regular term. The lp, q hybrid constraints are defined as:

$\|Z\|_{p,q} = ((\sum_i \sum_j |Z_{ij}|^p)^{\frac{q}{p}})^{\frac{1}{q}}$, Z_{ij} denotes the element in the i-th row and the j-th column, the $l_{2,1}$ mixed norm is used for the row group of P so that the relevant local color patches have similar representations; the group lasso penalty is used on the column group of Q to identify outliers at the same time. We divide the solution of (4) into two steps. The first step uses the APG (accelerate proximal gradient) algorithm to solve P, Q.

Set:

$$t(P, Q) = \sum_{k=1}^{K} \left\| X^k - W^k D^k Z^k - e^k \right\|_F^2 \tag{5}$$

$$g(P, Q) = \lambda_1 \left\| P \right\|_{2,1} + \lambda_2 \left\| Q^T \right\|_{2,1} \tag{6}$$

Now we apply the method of composite gradient mapping to (4), and we obtain the following function:

$$\begin{aligned} &\Phi(P, Q; R, S) \\ &\quad = t(R, S) + \langle \nabla Rt(R, S), P - R \rangle + \langle \nabla St(R, S), Q - S \rangle \\ &\quad\quad + \frac{\alpha}{2} \left\| P - R \right\|_F^2 + \frac{\alpha}{2} \left\| Q - S \right\|_F^2 + g(P, Q) \end{aligned} \tag{7}$$

In the m-th APG iteration:

$$\begin{aligned} R^{m+1} &= P^m + \varepsilon_m (\frac{1 - \varepsilon_{m-1}}{\varepsilon_{m-1}})(P^m - P^{m-1}) \\ S^{m+1} &= Q^m + \varepsilon_m (\frac{1 - \varepsilon_{m-1}}{\varepsilon_{m-1}})(Q^m - Q^{m-1}) \end{aligned} \tag{8}$$

(R^{m+1}, S^{m+1}) is linearly represented by W and (R^{m-1}, S^{m-1}), $\varepsilon_m = \frac{2}{\text{m+3}}$

The solution of the m-th iteration is obtained by solving equation (9)

$$(R^m, S^m) = \min_{P,Q} \Phi(P, Q; P^m, Q^m) \tag{9}$$

The solution of equation (9) can be divided into two parts: P and Q.

$$P^m = \min_P \frac{1}{2} \left\| P - V^m \right\|_F^2 + \frac{\lambda_1}{\alpha} \left\| P \right\|_{2,1} \tag{10}$$

$$Q^m = \min_Q \frac{1}{2} \left\| Q - U^m \right\|_F^2 + \frac{\lambda_2}{\alpha} \left\| Q^T \right\|_{2,1} \tag{11}$$

$$\begin{aligned} V^m &= R^m - \frac{1}{\alpha} \nabla Rt(R^m, S^m) \\ U^m &= S^m - \frac{1}{\alpha} \nabla St(R^m, S^m) \end{aligned} \tag{12}$$

After finding P, Q, Z, in the second step, we fix P, Q, Z to solve W and e. W can be obtained by using the ridge regression constraint term, so it can be derived directly.

$$W = ((DZ)^T DZ + \lambda_3 I)^{-1} (DZ)^T (x - e) \tag{13}$$

Then, after fixing W, P, Q, Z, e can be acquired by minimizing $f(e) = \|X - WDZ - e\|_F^2 + \|e\|_1$, which is essentially a convex optimization problem, and can be solved by the contraction operator [25], and the global minimum can be solved by the contraction operator[25], and e can be obtained from (14)

$$e = \beta\tau(X - WDZ) \tag{14}$$

$\beta\tau$ is the contraction operator and defined as

$$\beta\tau(x) = \text{sgn}(x) \cdot (|x| - \tau).$$

B. HANDLING OF SCALE CHANGES

The scale change in the target is always a key issue in target tracking. The existing methods for dealing with the scale change mainly use the scale pyramid method (SAMF) [39] and multiscale factor (DSST) [40], and obtain the best performance with templates scale. In this paper, the predicted target is multiplied by the scale factor of different sizes to extract the corresponding kernel density characteristics. The optimal value obtained by solving formula (17) and template matching is the optimal scale of the predicted target.

The reference target model is represented by the density estimation feature $\hat{q}$ in the feature space [26], as shown in (15). The target candidate is defined at position y and is characterized by the density estimation feature $\hat{p}(y)$, as shown in formula (16):

$$\hat{q} = \{\hat{q}u\}u = 1\ldots m, \quad \sum_{u=1}^{m} \hat{q}u = 1 \tag{15}$$

$$\hat{p}(y) = \{\hat{p}u(y)\}u = 1\ldots m, \quad \sum_{u=1}^{m} \hat{p}u = 1 \tag{16}$$

The k-th block feature dictionary D^k is obtained by collecting the k-th block target model density estimation feature $\hat{q}$. According to the collected density estimation feature $\hat{p}(y)$ of the i-th target candidate at y, the k-th block of the i-th candidate test sample x_i^k is formed. Then, we add feature dictionary D^k and test sample x_i^k to (17) to obtain the similarity value between the target candidate and the target model, and the scale with the largest similarity value is selected as the prediction target scale.

$$p(yt|st) = \frac{1}{\alpha}\exp(-\beta(\sum_{k=1}^{K}\left\|x_i^k - D^k z_i^k\right\|^2)) \tag{17}$$

yt represents the observed value, and st denotes system status; α, β are constant coefficients.

The main process of the proposed adaptive structured sparse representation is as follows:

Algorithm 1 Algorithm Process

```
1. Initial tracking target position $pos(1)$, obtain template dictionary $D$. After multiple experiments, the experimental results are best when the parameters are set as follows: attention mechanism parameter $W$ is set to value 1, the window size padding is 1.5 times the target size, the reconstruction error $e$ is initialized to 0, and the constants' values are $\lambda 1 = 0.001$, $\lambda 2 = 0.001$, $\lambda 3 = 5$, $\lambda 4 = 1$
For I = 2: imgnum(video frames)
2. Good point set sampling is used to obtain candidate target sets $X$
Solving (4) is performed in two steps:
    a. First step: fix $W, e$, solve $Z$ using the APG algorithm
      While t < T(T: the maximum number of iterations, t: number of iterations)
        $m$-th iteration:
          $R^{m+1} = P^m + \varepsilon_m(\frac{1-\varepsilon_{m-1}}{\varepsilon_{m-1}})(P^m - P^{m-1})$
          $S^{m+1} = Q^m + \varepsilon_m(\frac{1-\varepsilon_{m-1}}{\varepsilon_{m-1}})(Q^m - Q^{m-1})$
        To solve $Z$, first solve (11) in two parts $P$, $Q$
          $P^m = \min_P \frac{1}{2}\left\|P - V^m\right\|_F^2 + \frac{\lambda_1}{\alpha}\left\|P\right\|_{2,1}$
          $Q^m = \min_Q \frac{1}{2}\left\|Q - U^m\right\|_F^2 + \frac{\lambda_2}{\alpha}\left\|Q^T\right\|_{2,1}$
        ($U, V$ obtained from (12)). $Z = P + Q$
    b. Second step: fix $Z, P, Q$, solve $W, e$ in sequence according to (13) and (14)
3. According to (17), the candidate with the highest similarity is selected as the tracking target to obtain the target central position loc
4. Based on the loc, the candidates obtained by scaling the target selection box with different proportions are added to (17) to obtain the final predicted target size of the current frame. In addition, we save the target position and size in pos(i)
5. Update template, learning rate $\eta = 0.7$
6. Return to step 2, save pos(i)
7. Output pos
```

IV. EXPERIMENT

In this section, we show the performance of the proposed algorithm on mainstream video datasets and compare it with other algorithms, as well as some technical details in the implementation process.

A. EXPERIMENTAL SETUP

In the experiment, the video image was converted into a grayscale image. The image was initially divided into 2 × 2 local patches of the same size. The template was selected from the video frame image. The template size was consistent with the candidate target local patch size. The search radius of the candidate target is 1.5 times, and the number of candidate targets is 200. After tracking the target in each frame, update the template by comparing the error between the predicted target and the template patch; replace the patch with the largest error in the template with the target patch in the current frame, and use the learning rate η to update the remaining templates. The experimental environment is Matlab2018a, the host frequency is 3.60GHZ, and the memory is 8GB.

B. EXPERIMENTAL EVALUATION INDICATORS

There are three kinds of evaluation indexes in this experiment: average overlap rate, center position accuracy, and accuracy rate. Compare the predicted target frame obtained from the experimental real frame $R_{boundary}$ and $P_{boundary}$ of a given frame. Assume their center positions are $R_{centeral}$ and $P_{centeral}$, the center position error is $E_{centeral} = \|R_{centeral} - P_{centeral}\|_2$, and the average overlap ratio is defined as:

$$AO = \frac{area(R_{boundary} \cap P_{boundary})}{area(R_{boundary} \cup P_{boundary})} \tag{18}$$

The accuracy rate is based on whether the distance $Dist$ between the real coordinate $R_{centeral}$ and the predicted coordinate $P_{centeral}$ is less than 20 to determine the accuracy of the prediction.

$$Dist = \begin{cases} true, & E_{centeral} \le 20 \\ false, & E_{centeral} > 20 \end{cases} \tag{19}$$

TABLE 1. Comparison of the average center location error of different algorithms.

	TADT	UDT	STRCF	L1APG	STC	BACF	CN	CSK	DSST	SCT4	SRDCF	Staple	Struck	Ours
BlurCar2	3.60	3.44^2	2.96^2	355.70	212.10	81.45	188.68	8361.30	73.94	73.87	96.19	79.90	107.75	1.98^1
Boy	2.08^2	1.54^1	6.86	289.42	1444.95	59.96	91.01	312.02	67.43	92.23	55.70	53.83	79.59	4.77^3
Car4	3.68	2.97	1.91	225.56	10.66	3.44	20.53	19.13	1.72^3	8.60	1.68^2	2.41	149.65	1.15^1
Crossing	1.36^2	1.66	1.41^3	156.95	31.21	41.77	121.54	129.41	43.09	43.42	41.64	42.68	232.58	1.35^1
David2	1.56	1.30^3	1.42	179.09	57.43	1.13^1	1.72	2.33	2.04	2.66	1.46	2.53	198.91	1.18^2
Deer	6.27	4.65^2	4.73^3	139.34	140.43	107.12	233.35	370.10	78.77	97.01	100.31	105.12	76.22	3.19^1
Faceocc1	13.58^2	13.64^3	14.19	217.80	85.44	38.40	29.68	30.52	28.23	34.83	35.95	46.75	57.09	9.64^1
Football	4.46^3	6.11	6.08	218.99	22.28	3.99^2	16.12	16.19	15.76	5.57	5.68	13.00	93.43	3.95^3
Jumping	4.11	3.48^2	3.68^3	165.49	50.70	4.45	60.99	85.97	36.87	4.38	4.47	26.68	80.86	3.36^1
MountainBike	8.46	9.57	10.42	190.60	12.15	9.49	6.70^3	6.51^2	7.76	8.69	8.99	8.94	58.67	6.15^1
Shaking	6.15^2	7.64	6.44^3	187.53	32.33	6.85	14.80	17.16	8.36	10.02	85.44	38.03	13.72	6.07^1
KiteSurf	4.41^3	53.63	66.73	69.26	79.86	13.26	3.82^2	36.47	25.37	67.23	58.92	98.27	86.16	3.63^1
Man	2.09	1.89	1.61^3	152.08	5.62	1.36^2	2.10	1.78	1.57	117.51	1.62	218.35	64.44	1.03^1
Fish	4.34	4.19	3.53^3	172.70	29.35	3.77	35.50	41.19	4.11	5.04	3.42^1	4.05	190.97	3.51^2
Vase	2.43	3.40^3	2.35^2	115.56	507.74	6.32	12.43	19.34	11.11	5.28	4.11	12.36	30.30	2.13^1
Girl	8.74	7.71^2	10.12	31.69	141.78	6.70^1	65.58	77.04	43.58	132.45	54.84	21.96	16.79	8.38^3
Panda	15.37	16.36	14.90^2	74.99	100.35	15.78	16.16	19.73	17.22	63.19	15.10^3	42.21	88.99	14.83^1

TABLE 2. Comparison of the average overlap ratio of different algorithms.

	TADT	UDT	STRCF	L1APG	STC	BACF	CN	CSK	DSST	SCT4	SRDCF	Staple	Struck	Ours
BlurCar2	0.903^3	0.908^2	0.905	0.330	0.607	0.337	0.051	0.021	0.282	0.265	0.333	0.264	0.034	0.910^1
Boy	0.938^1	0.761	0.858^2	0.402	0.400	0.157	0.015	0.003	0.131	0.095	0.146	0.155	0.025	0.829^3
Car4	0.922	0.945^3	0.968^2	0.616	0.761	0.916	0.691	0.700	0.974	0.790	0.972	0.955	0.600	0.976^1
Crossing	0.936^3	0.929	0.937^2	0.040	0.325	0.103	0.010	0.084	0.079	0.084	0.093	0.079	0.527	0.991^1
David2	0.907	0.901	0.907	0.250	0.386	0.943^1	0.917^3	0.855	0.844	0.826	0.902	0.804	0.104	0.928^2
Deer	0.919^3	0.901	0.922^2	0.041	0.053	0.075	0.024	0.015	0.158	0.106	0.095	0.089	0.131	0.925^1
Faceocc1	0.807^2	0.804^3	0.798	0.000	0.338	0.607	0.588	0.577	0.599	0.551	0.597	0.468	0.502	0.931^1
Football	0.805^3	0.800	0.768	0.022	0.540	0.889^2	0.624	0.626	0.640	0.777	0.803	0.664	0.009	0.891^1
Jumping	0.759	0.722	0.868	0.530	0.127	0.820^2	0.053	0.050	0.139	0.772	0.813^3	0.267	0.624	0.828^1
MountainBike	0.785	0.756	0.775	0.025	0.788	0.771	0.896	0.895	0.828^3	0.871^2	0.817	0.782	0.473	0.922^1
Shaking	0.862^2	0.808	0.833^3	0.016	0.419	0.819	0.638	0.609	0.821	0.781	0.051	0.041	0.001	0.911^1
KiteSurf	0.711^3	0.354	0.382	0.000	0.275	0.470	0.762^2	0.260	0.331	0.342	0.374	0.320	0.264	0.864^1
Man	0.838	0.873	0.910	0.470	0.637	0.919^1	0.839	0.874	0.893	0.472	0.903^2	0.000	0.563	0.881^3
Fish	0.893	0.888	0.626	0.246	0.492	0.901	0.385	0.208	0.907^3	0.857	0.900	0.971^1	0.120	0.909^2
Girl	0.922^3	0.928^2	0.953^1	0.000	0.567	0.827	0.596	0.478	0.810	0.834	0.817	0.638	0.010	0.784
Panda	0.630	0.558	0.634^2	0.140	0.000	0.631^3	0.152	0.126	0.137	0.423	0.146	0.030	0.025	0.675^1
Vase	0.716	0.745	0.746	0.030	0.000	0.753	0.810^3	0.793	0.826^2	0.259	0.784	0.404	0.524	0.900^1

In this paper, some challenging videos with low resolution, plane rotation, scale change, deformation, background change, light change, motion blur, and fast motion are selected as experimental videos. STC, UDT, SRDCF, DSST, TADT, Struck, BACF, CN, CSK, L1APG, SCT4 and STRCF are selected as the comparison benchmark algorithm in this paper. The comparison results of 17 videos are shown in Tables 1, 2, and 3. In the same video, the results of the three best-performing algorithms are labeled superscript 1, 2 and 3 respectively.

Table 1 shows the comparison of the average center location error of different algorithms. Lower average center location error indicates that the algorithm's tracking results are better. Table 2 shows the comparison of the average overlap ratio of different algorithms on the video dataset. The higher the average overlap ratio is, the higher the accuracy of the tracking results. Table 3 shows the average tracking accuracy of different algorithms within 20 pixels error. The greater algorithms have better accuracy than other algorithms.

Table 1 shows the average center location error of different algorithms. On most of the videos, our method is the best. In all of the videos, our method's center location error is lower than 10 pixels, only TADT and UDT's results are similar to ours.

Comparing the results of videos Car4, Crossing, and Man, the center location error of our method is lower than 2 pixels. This proves that our method is effective and robust in the test videos.

According to the data in Table 2, the proposed algorithm has the highest average overlap rate on Car4, David2, Faceocc1, Jumping, Mountain Bike and other video sets. On shaking video, the overlap rate of our method reaches 0.911, which is much higher than the 0.862 of the second TADT. On the Crossing video set, the overlap rate of our method reaches 0.991, which is significantly higher than the rate of 0.937 of the second STRCF method. Before modeling, our algorithm segments the template and tracking target into local blocks, which enhances the local information of the target and mitigates the impact of target changes on the tracking result. The introduction of an attention mechanism further improves the accuracy and robustness of the proposed algorithm.

TABLE 3. Comparison of the average tracking accuracy of different algorithms (within 20 pixels error).

	TADT	UDT	STRCF	L1APG	STC	BACF	CN	CSK	DSST	SCT4	SRDCF	Staple	Struck	Ours
BlurCar2	1.00 ^{1}	1.00 ^{1}	1.00^1	0.55	0.53	0.05	0.01	0.01	0.07	0.06	0.05	0.06	0.62	1.00 ^{1}
Boy	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	0.00	0.44	0.19	0.02	0.00	0.20	0.14	0.20	0.24	0.25	1.00 ^{1}
Car4	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	0.95	0.97	1.00 ^{1}	0.35	0.36	1.00 ^{1}	0.97	1.00 ^{1}	1.00 ^{1}	0.91	1.00 ^{1}
Crossing	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	0.30	0.55	0.29	0.01	0.15	0.24	0.25	0.30	0.28	0.00	1.00 ^{1}
David2	1.00 ^{1}	1.00 ^{1}	1.00^1	0.00	0.34	1.00 ^{1}	1.00	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	0.20	1.00 ^{1}
Deer	0.94	1.00 ^{1}	0.99^3	0.03	0.00	0.03	0.01	0.01	0.07	0.04	0.03	0.03	0.11	1.00 ^{1}
Faceocc1	0.92 ^{2}	0.87^3	0.85	0.00	0.24	0.29	0.26	0.23	0.22	0.19	0.24	0.07	0.00	0.98^1
Football	1.00 ^{1}	0.99	0.98	0.01	0.80	1.00	0.80	0.80	0.80	1.00 ^{1}	1.00 ^{1}	0.80	0.02	1.00^1
Jumping	1.00 ^{1}	1.00^1	1.00 ^{1}	0.00	0.00	0.98	0.05	0.05	0.05	1.00 ^{1}	1.00^1	0.31	0.02	1.00 ^{1}
MountainBike	1.00 ^{1}	0.99	0.98	0.00	0.82	1.00 ^{1}	1.00^1	1.00^1	1.00^1	1.00 ^{1}	1.00 ^{1}	1.00	0.04	1.00 ^{1}
Shaking	0.98 ^{3}	0.95	0.98^3	0.20	0.19	0.98^3	0.71	0.56	1.00 ^{1}	0.94	0.01	0.02	0.48	0.99^2
KiteSurf	0.90^3	0.48	0.46	0.34	0.40	0.62	1.00 ^{1}	0.35	0.42	0.00	0.46	0.00	0.00	0.99^2
Man	1.00 ^{1}	1.00 ^{1}	1.00 ^{1}	0.97	1.00 ^{1}	1.00^1	1.00^1	1.00 ^{1}	1.00^1	0.00	1.00 ^{1}	0.00	0.82	1.00^1
Fish	1.00 ^{1}	0.99	1.00 ^{1}	0.89	0.42	1.00 ^{1}	0.40	0.04	1.00^1	1.00^1	1.00 ^{1}	1.00 ^{1}	0.68	1.00 ^{1}
Girl	1.00^1	1.00 ^{1}	1.00^1	0.00	0.87	0.94	0.86	0.55	0.93	1.00 ^{1}	0.99	0.86	0.23	0.99
Panda	1.00 ^{1}	1.00 ^{1}	0.94	0.00	0.00	0.99	0.25	0.16	0.22	0.40	0.38	0.00	0.82	1.00 ^{1}
Vase	0.63	0.72	0.74	0.35	0.00	0.77	0.81 ^{2}	0.79	0.85 ^{1}	0.42	0.81^2	0.02	0.66	0.75

As seen in Table 3, the proposed algorithm has the highest center accuracy on the BlurCar2, Boy, and Faceocc1 video sets, and the tracking accuracy is the highest.

Within the range of 20 pixels, the tracking accuracy of the proposed algorithm on most of the videos, such as Crossing, Deer and Jumping, reached 1; even on Faceocc1 video, our accuracy reached 0.98 and ranked first. The accuracy of most of the other algorithms, such as UDT, is less than 0.9. When the above video shows fast motion, motion blur and partial occlusion, the proposed algorithm can still track the target accurately. This is mainly due to the sparse model used in the proposed algorithm, which makes full use of the local and global information of the target, greatly reduces the error caused by the change in target appearance, and improves the robustness of the algorithm.

As can be seen from the tables above, In Boy, Car4, Football, Jumping, Deer and other videos, our method achieves better and more robust performance compared with that of the other methods. Our method also worked well for some of the fast-moving videos, such as Boy, and the partial occlusion videos. The structured sparse representation method not only considers the spatial layout structure of the image blocks inside each target candidate region but also considers the internal relations between the target candidate regions and between the local blocks. On this basis, this paper proposes to add an attention mechanism to strengthen the online learning of the target in the template, and continuously weaken the influence of the background on the tracking results. The attention mechanism can help us to obtain more discriminant information from sparse coding coefficients. With the continuous updating of the template and attention parameters, the reconstruction error of the moving target is continuously reduced, so the proposed method has a stronger performance on moving targets with motion blur, such as Deer, and partial occlusion. The feature of relatively uniform sampling of good point sets helps to collect more evenly distributed samples during the sampling process. It can quickly determine the approximate location of the target, decrease the algorithm's running time, and improve the efficiency of the algorithm. In the process of video processing such as the Car4 video, a reasonable scaling strategy is helpful for reasonably predicting the change in the target scale based on the size of the previous frame and the information of the current frame. The kernel density feature is not easily affected by the change in illumination and scale, which helps the system to obtain higher reliability.

From Figs. 3, 4, and 5, it can be seen that the proposed algorithm has excellent performance in terms of overlap and center accuracy, and it is different from other comparison algorithms. The proposed algorithm benefits from the attention mechanism and therefore has stronger robustness. In the case of motion blur, the proposed algorithm can accurately track the target. At the same time, because the good point set sampling is used in this paper, the sampling point with a larger sampling range is more evenly distributed so it can better handle some challenges such as fast movement. When dealing with partial occlusion and partial deformation, structured sparse representation considers the commonness between particles and the spatial structure of local blocks, so it has strong robustness when dealing with partial occlusion.

As seen in Fig. 4, when the threshold of this article is approximately 0.1, the overlap ratio of the proposed algorithm is close to 98%, while TADT, UDT, and STRCF can only reach approximately 90%∼95%. When the threshold is 1, our method shows a greater advantage than other algorithms. According to Fig. 5, in the face of multiple challenges, the proposed algorithm makes full use of effective scaling strategies to ensure tracking accuracy. Because of the effective adjustment of target size, the center error is greatly reduced. It can be seen in Fig. 5 that when the threshold is low, the gap between the proposed algorithm and other comparison algorithms is small; when the threshold is larger than 15, the average center accuracy of this paper is beyond 0.9 which is higher than the 0.8 accuracy of the rest of the comparison algorithms.

Table 1, Table 2, Table 3, Fig. 4, and Fig. 5 show that the proposed algorithm, which combines the structured sparse

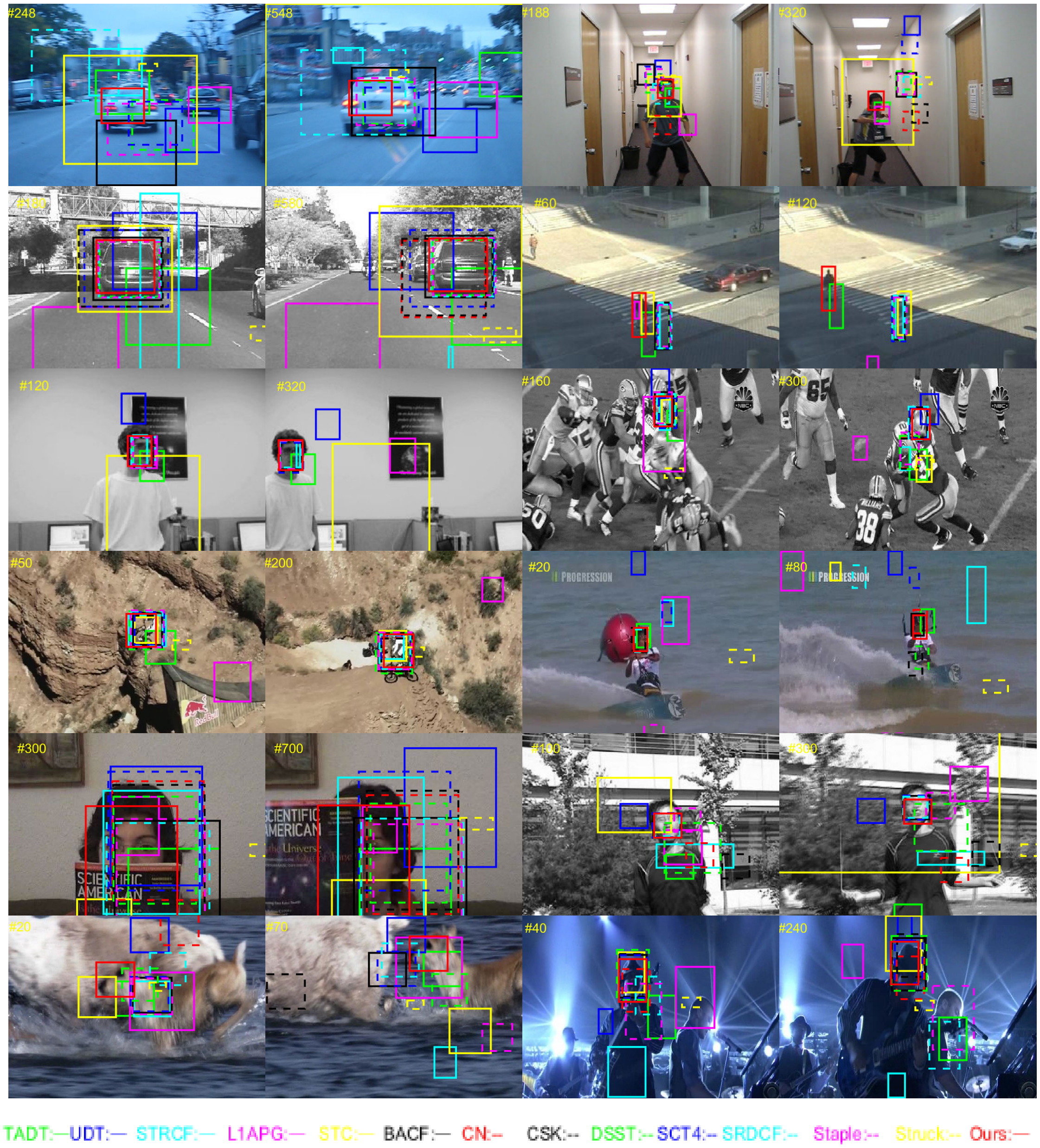

#248
#548
#188
#320
#60
#120
#120
#320
#160
#300
#50
#200
#20
#80
#300
#700
#20
#70
#40
#240
TADT:— UDT:— STRCF:— L1APG:— STC:— BACF:— CN:-- CSK:-- DSST:-- SCT4:-- SRDCF:-- Staple:-- Struck:-- Ours:—

FIGURE 3. Comparison of the actual effects of 13 methods on video sets (the solid red line is ours).

representation and the tracking method based on the sparse prototype, shows excellent processing capability compared to that of the other algorithms when dealing with partial occlusion, fast movement, light change and motion blur, and has better performance in terms of accuracy, center error and overlap. The attention mechanism improves the robustness of the algorithm. Fig. 3 specifically shows the comparison of the actual effect of this paper and other algorithms on multiple video sets.

We can see from Fig. 4 that our method shows an advantage from the beginning compared with the other methods, but as the threshold increases, this advantage decreases. This indicates that our strategy is effective for dealing with size, but there is room for improvement, and the larger the threshold is, the smaller the gap between methods. The slowly changing smooth curves in FIG. 4 and FIG. 5 also prove that our method is robust, which effectively demonstrates that adding an attention mechanism to the sparse structure is a correct choice. A reasonable weighting mechanism makes the attention mechanism more robust, which makes the template perform better and have more weight in previous frames. Some trackers use an intensive sampling method to override the state of the target object, but this can cause some other problems. First, it is hard for them to sample all possible particle filters which may include object states. However, the more uniform sampling method of good point sets greatly reduces the possibility of incomplete collection of target samples. Second, comparing with some methods that only use

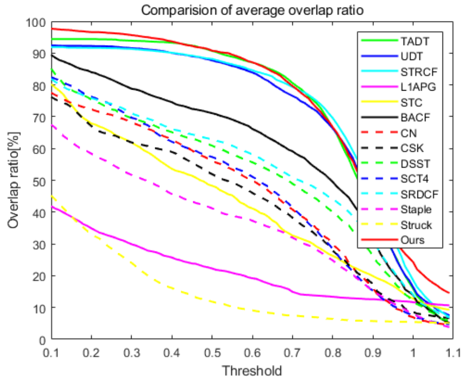


FIGURE 4. Comparison of the overlap ratio on OTB100.

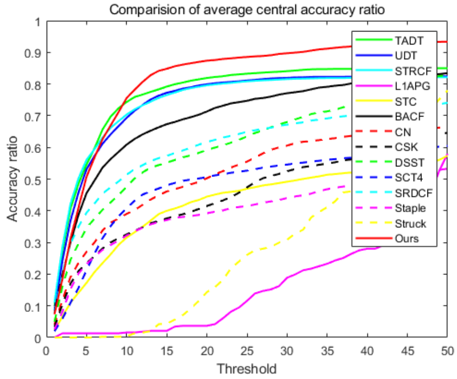


FIGURE 5. Comparison of accuracy ratio on OTB100.

simple template updates in tracing, our method can reduce the possibility of replacing or updating valid target templates due to the added attention mechanism and online update strategy. Third, simple features such as gray features are disturbed by external information, while feature extraction methods based on kernel density are less susceptible to other information.

The experiments show that the proposed algorithm achieves good results. It achieves excellent results in dealing with fast motion of targets, changes in lighting, and motion blur. It has certain effects when dealing with partial occlusion and deformation problems. It is found that the algorithm still lacks effective coping strategies when facing the problem of targets that are out of view, completely occluded, and when the shape of the target changes drastically.

V. CONCLUSION

In this paper, we proposed a new moving target tracking method by combining structural sparse representation and prototype-based sparse tracking and introducing an attention mechanism. The proposed method can effectively improve the accuracy rate of tracking and the overlap rate of tracking, and the adopted scale change strategy can ensure that the algorithm can perform well in the target scale change without reducing the tracking accuracy, thus greatly enhancing the robustness of the algorithm. In the algorithm solving process, we used the APG algorithm to solve the target model step by step, and then solved the optimal scale of the predicted target through the similarity between the template and the core density characteristics of the predicted target at different scales. The algorithm realized robust tracking of the target and updated the target template according to the tracking results. Experimental results show that this method achieves the goal of stable target tracking. In future work, we will extend our idea and methodology to other multimedia applications such as segmentation [49], detection [50], recommenders [51] and dehazing [52].

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Comput. Surv.*, vol. 38, no. 4, p. 13, Dec. 2006.

[2] K. Cannons, “A review of visual tracking,” Dept. Comput. Sci. Eng., York Univ., Toronto, ON, Canada, Tech. Rep. CSE-2008-07, Jul. 2008.

[3] S. Salti, A. Cavallaro, and L. Di Stefano, “Adaptive appearance modeling for video tracking: Survey and evaluation,” *IEEE Trans. Image Process.*, vol. 21, no. 10, pp. 4334–4348, Oct. 2012.

[4] K. Kaur, R. Khurana, and A. K. S. Kushwaha, “Deep survey on visual object tracking in surveillance environment,” in *Proc. Int. Conf. Res. Intell. Comput. Eng. (RICE)*, San Salvador, El Salvador, Aug. 2018, pp. 1–6.

[5] X. Lan, S. Zhang, P. C. Yuen, and R. Chellappa, “Learning common and feature-specific patterns: A novel multiple-sparse-representation-based tracker,” *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 2022–2037, Apr. 2018.

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

[7] 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.

[8] N. Wang and D.-Y. Yeung, *Learning a Deep Compact Image Representation for Visual Tracking*. Red Hook, NY, USA: Curran Associates, 2013, pp. 809–817.

[9] L. Bertinetto, “Fully-convolutional Siamese networks for object tracking,” in *Proc. Comput. Vis.-ECCV Workshops (ECCV)*, in Lecture Notes in Computer Science, vol. 9914. Cham, Switzerland: Springer, 2016, pp. 850–865.

[10] G. Li, Z.-Y. Liu, H.-B. Li, and P. Ren, “Target tracking based on biological-like vision identity via improved sparse representation and particle filtering,” *Cognit. Comput.*, vol. 8, no. 5, pp. 910–923, Oct. 2016.

[11] B. Liu, J. Huang, C. Kulikowski, and L. Yang, “Robust visual tracking using local sparse appearance model and K-Selection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2968–2981, Dec. 2013.

[12] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Hilton Head Island, SC, USA, Jun. 2000, pp. 142–149.

[13] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[14] K. Nummiaro, E. Koller-Meier, and L. Van Gool, “An adaptive colorbased particle filter,” *Image Vis. Comput.*, vol. 21, no. 1, pp. 99–110, Jan. 2003.

[15] S. G. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[16] X. Mei and H. Ling, “Robust visual tracking using l_1 minimization,” in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Kyoto, Japan, 2009, pp. 1436–1443.

[17] H. Li, C. Shen, and Q. Shi, “Real-time visual tracking using compressive sensing,” in *Proc. CVPR*, Providence, RI, USA, Jun. 2011, pp. 1305–1312.

[18] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, “Minimum error bounded efficient tracker with occlusion detection,” in *Proc. CVPR*, Colorado Springs, CO, USA, Jun. 2011, pp. 1257–1264.

[19] X. Mei and H. Ling, “Robust visual tracking and vehicle classification via sparse representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2259–2272, Nov. 2011.

[20] T. Zhang, S. Liu, C. Xu, S. Yan, B. Ghanem, N. Ahuja, and M.-H. Yang, “Structural sparse tracking,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 150–158.

[21] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, “Robust visual tracking via multi-task sparse learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2042–2049.

[22] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Low-rank sparse learning for robust visual tracking," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2012, pp. 470–484.

[23] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 1822–1829.

[24] T. Zhang, C. Xu, and M.-H. Yang, "Robust structural sparse tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 473–486, Feb. 2019.

[25] D. Wang, H. Lu, and M.-H. Yang, "Online object tracking with sparse prototypes," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 314–325, Jan. 2013.

[26] Y. Wang, X. Luo, L. Ding, J. Wu, and S. Fu, "Robust visual tracking via a hybrid correlation filter," *Multimedia Tools Appl.*, vol. 78, no. 22, pp. 31633–31648, Nov. 2019.

[27] S. Zhang, W. Lu, W. Xing, and L. Zhang, "Learning scale-adaptive tight correlation filter for object tracking," *IEEE Trans. Cybern.*, vol. 50, no. 1, pp. 270–283, Jan. 2020.

[28] Z. Song, J. Sun, and B. Duan, "Collaborative correlation filter tracking with online re-detection," in *Proc. IEEE 3rd Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, Chengdu, China, Mar. 2019, pp. 1303–1313.

[29] H. Nishimura, Y. Nagai, K. Tasaka, and H. Yanagihara, "Object tracking by branched correlation filters and particle filter," in *Proc. 4th IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nanjing, China, Nov. 2017, pp. 79–84.

[30] Y. Liu, P. Wang, and H. Wang, "Target tracking algorithm based on deep learning and multi-video monitoring," in *Proc. 5th Int. Conf. Syst. Informat. (ICSAI)*, Nanjing, China, Nov. 2018, pp. 440–444.

[31] 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.*, Salt Lake City, UT, USA, Jun. 2018, pp. 8971–8980.

[32] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Y. Choi, "Action-driven visual object tracking with deep reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2239–2252, Jun. 2018.

[33] L. Wang, T. Liu, B. Wang, J. Lin, X. Yang, and G. Wang, "Learning hierarchical features for visual object tracking with recursive neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, Sep. 2019, pp. 3088–3092.

[34] X. Cheng, Y. Gu, B. Chen, Y. Zhang, and J. Shi, "Weighted multiple instance-based deep correlation filter for video tracking processing," *IEEE Access*, vol. 7, pp. 161220–161230, 2019.

[35] Y. Zhang, Y. Huang, and L. Wang, "Multi-task deep learning for fast online multiple object tracking," in *Proc. 4th IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nanjing, China, Nov. 2017, pp. 138–143.

[36] Y. Qi, Y. Wang, and Y. Liu, "Object tracking based on deep CNN feature and color feature," in *Proc. 14th IEEE Int. Conf. Signal Process. (ICSP)*, Beijing, China, Aug. 2018, pp. 469–473.

[37] J. Xiang, G. Zhang, and J. Hou, "Online multi-object tracking based on feature representation and Bayesian filtering within a deep learning architecture," *IEEE Access*, vol. 7, pp. 27923–27935, 2019.

[38] X. Zhou, L. Xie, P. Zhang, and Y. Zhang, "An ensemble of deep neural networks for object tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 843–847.

[39] 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.

[40] 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. 1–5.

[41] Y. Zeng, X. Guo, H. Wang, M. Geng, and T. Lu, "Efficient dual attention module for real-time visual tracking," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Sydney, NSW, Australia, Dec. 2019, pp. 1–4.

[42] S. Peng, S.-I. Kamata, and T. P. Breckon, "A ranking based attention approach for visual tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, Sep. 2019, pp. 3073–3077.

[43] Z. Liu, J. Wang, G. Liu, and L. Zhang, "Discriminative low-rank preserving projection for dimensionality reduction," *Appl. Soft Comput.*, vol. 85, Dec. 2019, Art. no. 105768, doi: 10.1016/j.asoc.2019.105768.

[44] Z. Liu, Z. Lai, W. Ou, K. Zhang, and R. Zheng, "Structured optimal graph based sparse feature extraction for semi-supervised learning," *Signal Process.*, vol. 170, May 2020, Art. no. 107456, doi: 10.1016/j.sigpro.2020.107456.

[45] W. Ou, D. Yuan, D. Li, B. Liu, D. Xia, and W. Zeng, "Patch-based visual tracking with online representative sample selection," *J. Electron. Imag.*, vol. 26, no. 3, May 2017, Art. no. 033006.

[46] W. Ou, D. Yuan, Q. Liu, and Y. Cao, "Object tracking based on online representative sample selection via non-negative least square," *Multimedia Tools Appl.*, vol. 77, no. 9, pp. 10569–10587, May 2018, doi: 10.1007/s11042-017-4672-3.

[47] X. Ma, Q. Liu, W. Ou, and Q. Zhou, "Visual object tracking via coefficients constrained exclusive group LASSO," *Mach. Vis. Appl.*, vol. 29, no. 5, pp. 749–763, Jul. 2018, doi: 10.1007/s00138-018-0930-2.

[48] Q. Liu, X. Ma, W. Ou, and Q. Zhou, "Visual object tracking with online sample selection via lasso regularization," *Signal, Image Video Process.*, vol. 11, no. 5, pp. 881–888, Jul. 2017, doi: 10.1007/s11760-016-1035-x.

[49] H. Yu, F. He, and Y. Pan, "A scalable region-based level set method using adaptive bilateral filter for noisy image segmentation," *Multimedia Tools Appl.*, vol. 79, nos. 9–10, pp. 5743–5765, Mar. 2020, doi: 10.1007/s11042-019-08493-1.

[50] Q. Quan, F. He, and H. Li, "A multi-phase blending method with incremental intensity for training detection networks," *Visual Comput.*, Jun. 2020. [Online]. Available: https://link.springer.com/content/pdf/10.1007/s00371-020-01796-7.pdf, doi: 10.1007/s00371-020-01796-7.

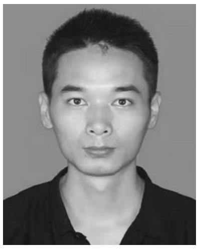
[51] Y. Pan, F. He, and H. Yu, "A novel enhanced collaborative autoencoder with knowledge distillation for top-N recommender systems," *Neurocomputing*, vol. 332, pp. 137–148, Mar. 2019, doi: 10.1016/j.neucom.2018.12.025.

[52] J. Zhang, F. He, and Y. Chen, "A new haze removal approach for sky/river alike scenes based on external and internal clues," *Multimedia Tools Appl.*, vol. 79, nos. 3–4, pp. 2085–2107, Jan. 2020, doi: 10.1007/s11042-019-08399-y.

JIE WANG is currently pursuing the master's degree with the Guangxi University for Nationalities, Nanning, China. His main research interests include image processing and pattern recognition.

SHIBIN XUAN received the Ph.D. degree in computer science and technology from Sichuan University, Chengdu, China, in 2011. He is currently a Full Professor and a Master's Supervisor with the School of Information Science and Engineering, Guangxi University for Nationalities, Nanning, China. His main research interests include image processing and pattern recognition.

HAO ZHANG is currently pursuing the master's degree with the Guangxi University for Nationalities, Nanning, China. His main research interests include image processing and deep learning.



XUYANG QIN is currently pursuing the master's degree with the Guangxi University for Nationalities, Nanning, China. His main research interests include image processing and pattern recognition.