Multimedia Tools and Applications (2023) 82:12245–12262
https://doi.org/10.1007/s11042-022-13703-4

The moving target tracking and segmentation method based on space-time fusion

Jie Wang[1] · Shibin Xuan[1,2] · Hao Zhang[1] · Xuyang Qin[1]

Received: 22 December 2020 / Revised: 9 March 2022 / Accepted: 24 August 2022 /
Published online: 21 September 2022

Abstract
At present, the target tracking method based on the correlation operation mainly uses deep learning to extract spatial information from video frames and then performs correlations on this basis. However, it does not extract the motion features of tracking targets on the time axis, and thus tracked targets can be easily lost when occlusion occurs. To this end, a spatiotemporal motion target tracking model incorporating Kalman filtering is proposed with the aim of alleviating the problem of occlusion in the tracking process. In combination with the segmentation model, a suitable model is selected by scores to predict or detect the current state of the target. We use an elliptic fitting strategy to evaluate the bounding boxes online. Experiments demonstrate that our approach performs well and is stable in the face of multiple challenges (such as occlusion) on the VOT2016 and VOT2018 datasets with guaranteed real-time algorithm performance.

Keywords Target tracking · Kalman filtering · Segmentation · Elliptic fitting

1 Introduction

Target tracking has become a popular research topic in the field of computer vision because of its wide application and great potential in areas such as intelligent surveillance, driverless driving, human–computer interaction, and intelligent transportation. Before 2010, target tracking was mostly done using classical algorithms such as particle filtering [10], Kalman filtering [32], mean drift [35], and the optical flow method. In 2010, Bolme et al. [3] applied the correlation filtering method to tracking; later KCF [16], BACF [19], SRDCF [7], DSST [8], CACF [24], and Siamese [2] methods were employed. In 2016, Bertinetto et al. [2] proposed a tracking method that combines a Siamese network in deep learning with correlation filtering

✉ Shibin Xuan
xuanshibin@gxun.edu.cn

[1] School of Artificial Intelligence, Guangxi Minzu university, Nanning 530006, China

[2] Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Nanning 530006, China

and achieved great success. So far, most deep-learning-based target tracking methods [4, 5, 17, 20, 22, 29, 30, 36] have been based on this method. In 2018, Li et al. [21] proposed the SiamRPN method that combines SiameseFC with RPN, abandoning the traditional multiscale detection method. Wang et al. [29] proposed the SiamMask method that combines SiamRPN networks with sharp segmentation networks, ensuring that the SiameseFC and RPN networks can be used in the same way. Tracking accuracy is significantly improved by tracking targets based on segmentation results, while compromising on the accuracy of the SiamRPN method.

Although these methods have achieved excellent results, they focus more on the merits of the features of the target and neglect the construction of motion models for target tracking. Even though good features can easily improve the performance of a tracker, target tracking faces problems such as occlusion, lighting changes, scale changes, deformation, and motion blur. Moreover, there are no methods available for extracting good features in every scenario. In the face of heavy occlusion, it is difficult for detection or segmentation-based methods to extract sufficient features as the target does not appear in the field of view. In contrast, Kalman filtering methods can accurately predict the state of a target by learning from the state of the target in the past frame in the absence of sufficient target features.

The major contributions of this study are as follows. Firstly, we propose the use of Kalman filtering to build motion models in combination with the SiamMask method to address the problem of missing tracked targets in complex environments resulting from the lack of accurate information on segmented target objects. Secondly, an elliptical fitting strategy is used to evaluate the angle and size of the rotating bounding boxes, and an attention mechanism is used to focus the model more on the contribution of the target subject area and to reduce the influence of the background.

The strengths of the proposed system are as follows. Firstly, we refine the tracking results of the tracking model by combining the segmentation model. This makes this method can still maintain high accuracy and robustness in a variety of complex environments. Secondly, by combining Kalman filter method, we propose a spatiotemporal motion model which can effectively alleviate the negative impact of occlusion. Benefit from this, even if we can't extract enough effective target appearance features, we can also track them in a short time. Thirdly, we use the ellipse fitting strategy to refine the final boundary box, which helps us greatly improve the accuracy of the algorithm on the premise of consuming minimal resources.

Section 1 focuses on a brief introduction to our approach. Section 2 introduces related work, and Section 3 describes our approach in detail, including the main structure and core modules of the algorithm. Section 4 compares our approach to other popular algorithms on two datasets, VOT2016 and VOT2018; the strengths and weaknesses of our method are analyzed, and we propose future directions to address the weaknesses. Finally, the full text is reviewed and a reasonable conclusion is provided in Section 5.

2 Related works

In this section, we briefly review the research progress on Siamese networks in the target tracking field in recent years. Bertinetto et al. [2] proposed the SiamFC method, combining the Siamese network with related filtering methods for the first time and successfully applying it to target tracking. However, the SiamFC method has weak adaptability to the environment, cannot be adapted to changes in scale, and its accuracy and precision cannot meet the complex circumstances of tracking requirements. SiamRPN was proposed by Li et al. [21]. This method

focuses on the introduction of RPN networks. By pre-setting multiple anchors, the position and size of the target in the current frame are determined through pre-learned classification branches and position branches. Mask_RCNN [15] adds a branch on the basis of Faster RCNN [27] to segment the target instance while achieving target detection. SiamMask [29] refers to the Mask_RCNN method and adds segmentation branches on the basis of SiamRPN, maps the segmentation image back to the original image, and uses the segmentation object as the final tracking result to achieve real-time target tracking. The SiamMask method greatly improves tracking accuracy. However, because only the influence of positive samples on the tracking results is considered, SiamMask often incorrectly segments backgrounds with high similarity to the target into the target when intraclass interference and severe occlusion occur, resulting in inaccurate tracking results or even loss.

The above methods all focus on the complete network parameters of offline training, and there is almost no online learning strategy. However, the uncertainty of the tracked object and the complex, changeable tracking scene mean that it is difficult for the pre-trained network to fully represent changeable target tracking and the influence of the background on target tracking for each video image. Therefore, a reasonable online learning strategy is necessary.

The main emphasis of the current popular method is to track the network based on offline training. Zhang [36] proposed relying on temporal and spatial context information to model the temporal and spatial information of the tracking target through a Bayesian framework to obtain a correlation between the target and the surrounding features. The Kalman filter is based on the state transition equation and the observation state, and an optimal estimation is obtained by combining these two Gaussian distributions, which is used for linear filtering and prediction problems.

In this study, Kalman filtering is used to construct a motion model and to predict the state of the target when there are obvious deviations and errors in the tracking. It is experimentally demonstrated that this method is superior to the SiamMask algorithm when faced with occlusion problems and in-class interference problems.

3 Our method

In this section, we will describe our approach in detail. We divide the tracking system into the following modules: a prediction module, a segmentation module, and a correction module. The prediction module uses an efficient prediction of the state of the object in the video image that appears heavily occluded or subject to in-class interference. The segmentation module uses a Siamese network with a segment branch to efficiently segment the target object in each frame. The correction module uses an elliptic fitting strategy to correct the final bounding box for the segmentation results in the video image. The main structure of the algorithm is shown in Fig. 1.

3.1 Prediction module

In target tracking, most of the existing methods only focus on how to extract quality features, while ignoring the continuity of target tracking in space–time. SiamMask [29] determines the final state of the target based on the segmentation results, but in several experiments we found that the segmentation branch can easily confuse similar objects in the background with the tracked target when they are disturbed by inner classes. Kalman filtering does not depend on the merits of the extracted features but rather relies on the movement trend of the target in the

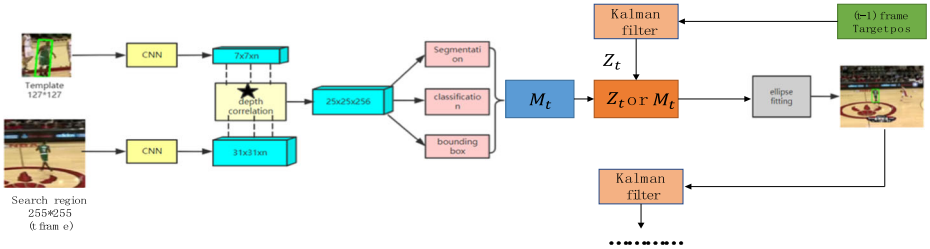


Fig. 1 Algorithm structure diagram

spatiotemporal sequence. Given a set of video observation sequences Y_t, the observation state can be expressed linearly by the state variable Z_t as

$$Y_t = H_t Z_t + n_t, \quad t = 1, 2, \ldots, N, \tag{1}$$

where H_t is the observation matrix, n_t is the observation noise, Z_t represents the state of the target at time t. The transfer of the state of the target can be represented by the linear state transfer equation

$$Z_t = \Phi_{t,t-1} Z_{t-1} + w_t, \quad t = 1, 2, \ldots, N, \tag{2}$$

where $\Phi_{t,t-1}$ is the state transfer matrix, w_t is the error of the state model, and its covariance matrix is Q_t for the error of the state transfer model.

The state update process is a two-step process: state prediction and error matrix prediction, Kalman gain calculation, status update, error matrix update, and status update. The process is as follows: The state prediction equation is

$$\widehat{Z}_t = \Phi_{t,t-1} \widehat{Z}_{t-1}. \tag{3}$$

The covariance prediction equation is

$$P_t^- = \Phi_{t,t-1} P_{t-1}^{\mu} \Phi_{t,t-1}^T + Q_t. \tag{4}$$

The Kalman gain equation is

$$K_k = P_t^- H_t^T \left[H_t P_t^- H_t^T + R_t \right]^{-1}. \tag{5}$$

The state update equation is

$$\widehat{Z}_t = \widehat{Z}_t^- + K_t \left[Y_t - H_t \widehat{Z}_t^- \right]^{-1}. \tag{6}$$

The covariance update equation is

$$P_t = P_t^- - K_t H_t P_t^-. \tag{7}$$

In this study, the center point of the target is modeled as a characteristic point $X = [x, y]^T$ as a uniformly accelerated motion, and a quadratic polynomial motion model is obtained:

$$\left.\begin{array}{c} X_t = X_{t-1} + v_{t-1}\Delta t + \frac{1}{2} a_{t-1} (\Delta t)^2 \\ v_t = v_{t-1} + a_{t-1}\Delta t \\ a_t = a_{t-1} \end{array}\right\}, Z_t = \begin{bmatrix} X_t \\ v_t \\ a_t \end{bmatrix}. \tag{8}$$

The error of the state transition model, $\Phi_{t,t-1}$, and the observation matrix H_t are

$$\Phi_{t,t-1} = \begin{bmatrix} I_2 & I_2\Delta t & \frac{1}{2}I_2(\Delta t)^2 \\ 0_2 & I_2 & I_2\Delta t \\ 0_2 & 0_2 & I_2 \end{bmatrix}, H_t = [\, I_2 \quad 0_2 \quad 0_2 \,], \tag{9}$$

where I_2represents a two-dimensional identity matrix and 0_2 represents a two-dimensional zero matrix. According to Eqs. (1) and (2), the Kalman filter can be used to accurately predict the state of the target. The final state of target S_t in this study is determined by

$$S_t = \begin{cases} M_t, \ dist \leq \sigma \text{ and } & score \geq \eta, \\ Z_t, \ dist > \sigma \quad \text{or} & score < \eta, \end{cases} \tag{10}$$

where M_t is the target state obtained by splitting the branch, Z_t is the target state predicted by Kalman filtering, ***dist*** is the Euclidean distance between the state of the target center in the previous frame S_{t-1} and M_t, and ***score*** is the score of the feature with dimensions $1 \times 1 \times 256$, which represents the similarity between target and candidate samples. The more similar the candidate and the template are, the higher the score will be. We choose a more reasonable target state between M_t and Z_t by using Eqs. (10). If the target is severely blocked or out of view, all ***score*** values are lower than η. However, if there are similar objects in the candidate area, the actual possible state of the target cannot be distinguished by the ***score*** value alone. Here we default to the case in which the target does not move in a large range between two frames. The rationale for this decision is that the target state $M_t$obtained by segmentation may have large deviations, and similar objects in the candidate area may be mistakenly identified as tracking targets. Through many experiments, we found that, when ***score*** $< \eta$, problems such as the target being occluded in a large area or the target leaving the field of view often appear in the video image. In such a case, the original tracker still considers that most of the target should be visible in the field of view. Therefore, we have to choose a background with a higher similarity to the template as the target to continue tracking. We have derived optimal values for the parameters in numerous experiments and have set $\sigma = 100$ and $\eta = 0.9$. At the same time, to ensure the stability of the tracker, we assume that the value of ***dist*** should be within a certain range and that the target's trajectory will not exhibit large-scale fluctuations. This is because in the experiment we found that, except for images with fast-moving objects, the targets in the other images rarely move long distances. The long-distance movement of the tracking result is often caused by the loss of the tracking target. Therefore, to improve the robustness of the algorithm, we have used Eq. (10) as the selection criteria.

3.2 Segmentation module

We used SiamMask [29] as the segmentation module of this study. SiamMask uses RPN [21] to calculate simple classification scores and bounding boxes, so that the candidate window of a fully convolutional Siamese network encodes the necessary information to generate a pixel-level binary segmentation mask. Two inputs (a template and a search area) go through the same convolutional neural network f_θ, and a deep cross-correlation of the two feature maps is performed to obtain

$$g_\theta(T, SR) = f_\theta(T) * f_\theta(SR). \tag{11}$$

SiamMask uses a simple two-layer neural network h_ϕ with a learned parameter ϕ to predict a binary mask of size $w \times h$. The predicted mask of the nth candidate window $g_\theta^n(T, SR)$ is

$$m_n = h_\phi\left(g_\theta^n(T, SR)\right). \tag{12}$$

From Fig. 1, we can see that there are three branches paralleling the segmentation branch: classification branch, a regression branch, and a segmentation branch. The classification branch is used to distinguish the target from the background. It predicts each sample as a target and a background score, and its loss function is recorded as L_{cls}. The regression branch fine-tunes the candidate area to obtain the predicted position and bounding box size, and the loss function is recorded as L_{reg}. The segmentation branch extracts the feature with the highest score in the feature map and decodes it to generate a segmented binary mask; its loss function is denoted as $L_{\boldsymbol{mask}}$. The total loss function of the SiamMask method is therefore

$$L_{3B} = \lambda_1 L_{\boldsymbol{mask}} + \lambda_2 L_{reg} + \lambda_3 L_{cls}, \tag{13}$$

whereλ_1, λ_2,and λ_3 are the parameters.

3.3 Correction module

After many experiments, we found that the segmentation results often do not perfectly strip the target from the background. The SiamMask method uses the smallest rectangular bounding box of the segmentation mask as the final result in the current frame. Even if the segmentation result contains a small part of the background, it will have a greater impact on the final bounding box. In this study, the ellipse-fitting strategy is used to finely select the rotating bounding box, so that the final result is more biased toward the torso of the target to reduce the accuracy drop caused by the inaccurate segmentation of a small part. An ellipse can be represented by a conical equation with the following constraints:

$$\begin{gathered} F(i,j) = ai^2 + bij + cj^2 + di + ej + f = 0, \\ b^2 - 4ac < 0, \end{gathered} \tag{14}$$

where a, b, c, d, e, and f are the coefficients of the ellipse and i, j is a point on the ellipse. Because the image needs to be rotated around the center of the ellipse, the following transfer matrix is used to calculate the coordinates of the transferred point in the original image:

$$M = \begin{bmatrix} \cos\theta & \sin\theta & (1-\cos\theta) i_{cen} & -\sin\theta j_{cen} \\ -\sin\theta & \cos\theta & \sin\theta i_{cen} & -(1-\cos\theta) j_{cen} \end{bmatrix}, \tag{15}$$

where θ is the rotation angle and (i_{cen}, j_{cen}) is the center point. If $\boldsymbol{Mask}_a$ is the set of all points in the segmentation mask, then$\boldsymbol{Mask}_b$, the point set of the segmentation mask after the transfer, is given by

$$Mask_b = M * \begin{bmatrix} i \\ j \\ 1 \end{bmatrix}, \forall (i,j) \in Mask_a. \tag{16}$$

rec_a is the smallest rectangular bounding box of the ellipse of the target mask after rotation. The smallest rectangle of the segmentation result is $rec_{\boldsymbol{mask}}$. The intersection rec_l of rec_a and $rec_{\boldsymbol{mask}}$ is calculated as the optimized bounding rectangle. The segmented image rec_l is then rotated back to the original position according to the rotation angle θ, and the rotated rec_l^θ is outputted as the final bounding box. Figure 2 shows the main flow of the calculations.

Fig. 2 Ellipse fitting strategy

4 Experiment

In this section, we evaluate the improved methods we propose on the VOT2016 and VOT2018 datasets, and we compare them with a number of popular methods. The experimental results demonstrate that this proposed method has great accuracy and precision. To reflect the fairness of comparison, the SiamMask part employed here uses the same structure and parameters as in Wang et al. [29]. Our experimental setup made use of computer with a Ryzen7 4800 h CPU, a GeForce GTX 1650Ti GPU, and 16 GB of memory, running on a Windows 10 operating system under a Python program.

4.1 Evaluation criteria

The evaluation indicators used in this study were the average overlap ratio, tracking length, failure rate, and robustness. The average overlap ratio is the intersection ratio between the area of the predicted target and the real area. The larger the value of this ration, the greater is the error. The tracking length is the number of frames in which the error from the start of tracking to the center point is lower than the acceptable range of the threshold. The failure rate is specified as follows: When the overlap rate is lower than the threshold, the tracking has failed, and the bounding box is reinitialized. The shorter the track length of each segment, the greater is the failure rate. During the kth algorithm repeated measurement process, the video robustness is calculated using

$$Rs = e^{-aM}, M = \frac{F1}{N}, \tag{17}$$

where M is the average time of failures, $F1$ is the total time of failures, N is the length of the video sequence, and a is a parameter. $F(i,\ k)$ represents the number of times that the recording algorithm fails to track in the video image and reinitialize after five frames.

Table 1 VOT2016 accuracy comparison

	camera motion	empty	illum change	motion change	occlusion	size change	Mean	Weighted mean	Pooled
ECO	0.5534	0.5776	**0.6841**	0.4934	0.4419	0.4936	0.5407	0.5340	0.5421
TADT	**0.5693**	0.5806	0.6282	0.5273	0.4571	0.5023	0.5441	0.5460	0.5504
VITAL	0.5361	0.5756	0.5619	0.5387	**0.5121**	0.5205	0.5408	0.5426	0.5561
ECO_HC	0.5530	0.5776	0.6841	0.4934	0.4419	0.4936	0.5406	0.5340	0.5411
SiamMask	0.5337	0.5689	0.6580	0.5544	0.3878	**0.5944**	0.5495	0.5487	0.5487
SiamRPN	0.5093	0.5408	0.6637	0.5111	0.384	0.5672	0.5167	0.5161	0.5132
SiamCAR	0.4977	0.4926	0.6195	0.4350	0.3072	0.4945	0.4744	0.4735	0.4694
Ours	0.5631	**0.6072**	0.6247	**0.5632**	0.4318	0.5926	**0.5638**	**0.5704**	**0.5743**

Table 2 VOT2018 accuracy comparison

	camera motion	empty	illum change	motion change	occlusion	size change	Mean	Weighted mean	Pooled
ECO	0.121	0.0272	0.0229	0.0237	0.0263	0.0273	0.0233	0.0221	0.0226
TADT	0.4004	0.3811	0.3843	0.3806	0.3120	0.3944	0.3755	0.3825	0.3809
VITAL	0.4373	0.3899	0.4388	0.4235	0.3076	0.4558	0.4088	0.4110	0.4113
ECOHC	0.0056	0.0092	0.0025	0.0151	0.0088	0.0280	0.011	0.0112	0.0105
SiamMask	0.4900	0.4241	**0.5253**	0.4974	0.2673	**0.5465**	0.4584	0.4582	0.4523
SiamRPN	0.4603	0.4045	0.5140	0.4764	0.2622	0.5144	0.4386	0.4353	0.4312
SiamCAR	0.4308	0.3632	0.4638	0.3647	0.2715	0.4178	0.3853	0.3843	0.3823
Ours	**0.4902**	**0.4552**	0.4862	**0.5392**	**0.3933**	0.5415	**0.4843**	**0.4840**	**0.4812**

4.2 Experimental results

The proposed algorithm was tested and evaluated on the VOT2016 and VOT2018 datasets, and we compared the results with those from ECO [9], ECO_HC and VITAL [28], SiamMask, SiamRPN, and TADT [11], and SiamCAR [13]. The videos were divided into nine categories, and the results closely reflect the performance capabilities of each algorithm in different scenarios. The experimental results demonstrate that the algorithm has good performance when facing various challenges and that its stability is obviously stronger than that of the compared algorithms.

4.3 Analysis of experimental results

From Table 1, we can see that our method has achieved good results in motion variation, camera motion, and scale variation and is first in average video accuracy, which demonstrates that the algorithm is robust. The segmentation results can accurately segment the target in the face of motion state changes, the Kalman filter can more accurately predict the target location and fine-

Table 3 Comparison of the algorithm in VOT2016 coverage, failure rate, and expected average coverage

	Overlap	Failures	EAO
ECO	0.5340	21.3960	0.3226
TADT	0.5460	19.9735	0.3006
VITAL	0.5426	18.3748	0.3227
ECO_HC	0.5340	21.3960	0.3226
SiamMask	0.5487	**0.000**	0.5775
SiamRPN	0.5161	**0.000**	0.5639
SiamCAR	0.4735	**0.000**	0.5431
Ours	**0.5704**	**0.000**	**0.6167**

Table 4 Comparison of the algorithm in VOT2018 coverage, area under the curve, and average coverage

	Overlap	AUC	EAO
ECO	0.2088	0.2076	0.2191
TADT	0.3825	0.3809	0.4317
VITAL	0.4110	0.4113	0.4699
ECO_HC	0.0112	0.0155	0.0105
SiamMask	0.4582	0.4523	0.5344
SiamRPN	0.4353	0.4312	0.5096
SiamCAR	0.3843	0.3812	0.4710
Ours	**0.4840**	**0.4812**	**0.5446**

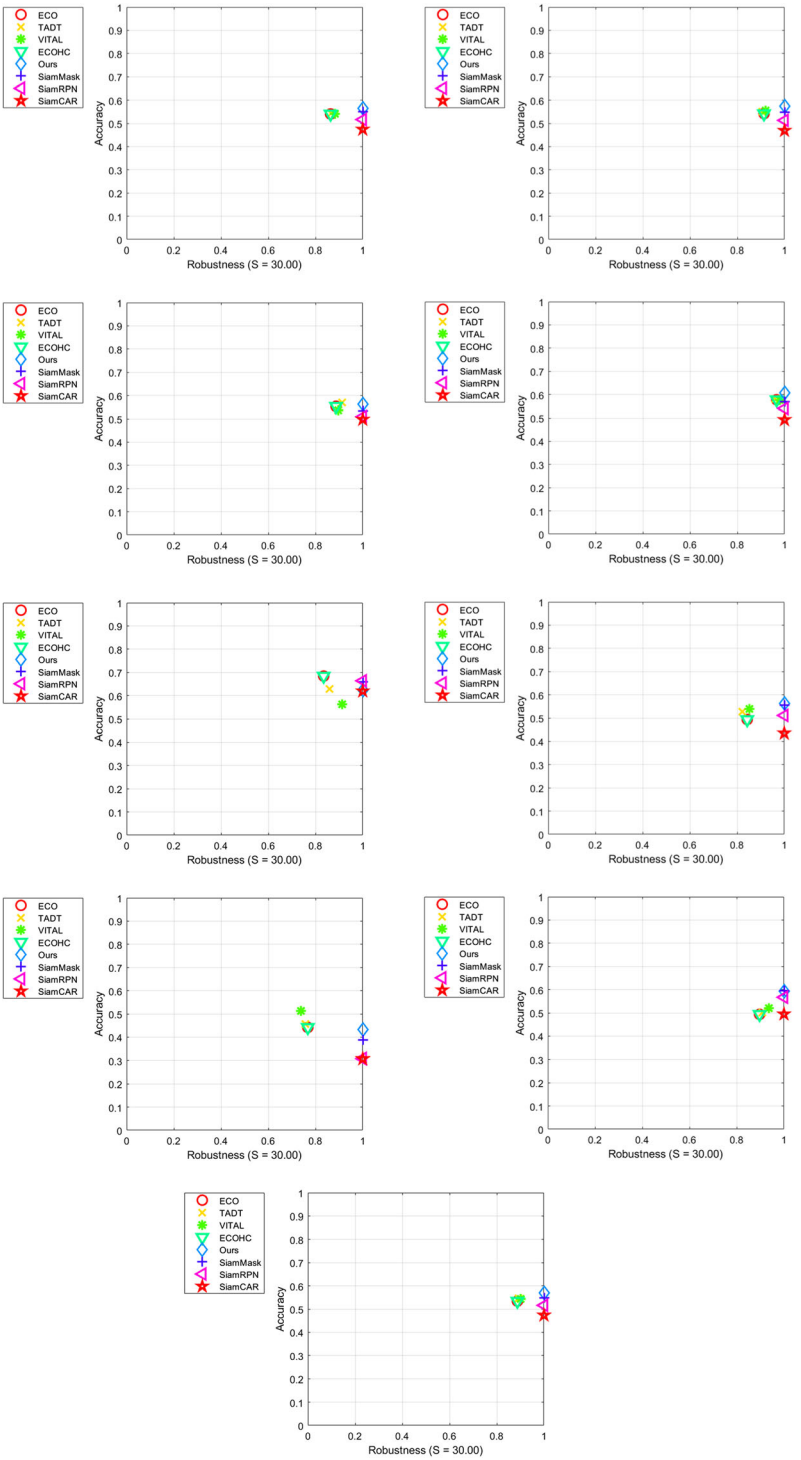


Fig. 3 EAO comparison results (VOT2016)

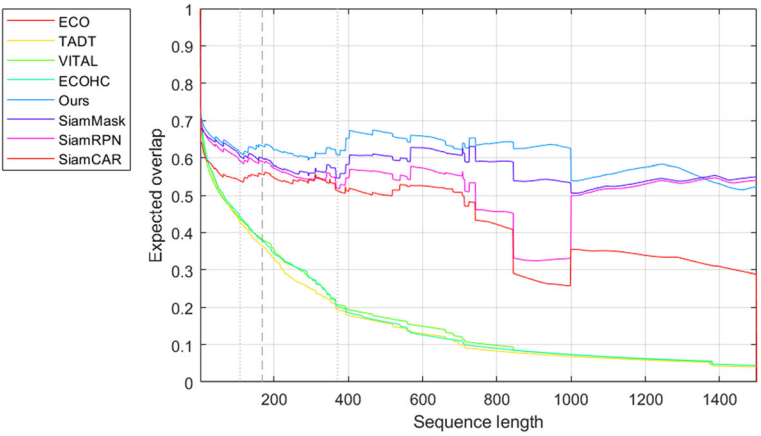
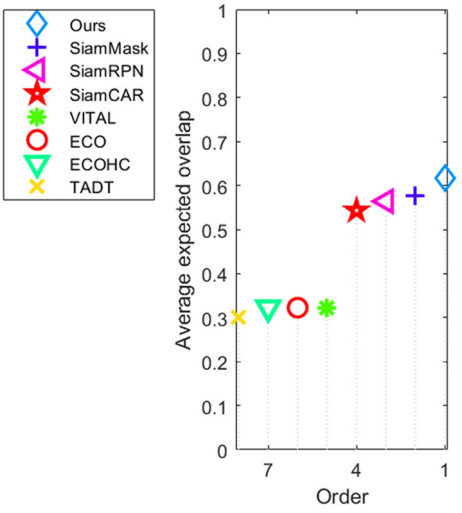


Fig. 4 Expected overlap curve (VOT2016)

tune the target state results obtained from segmentation, and the scale variation response achieves excellent results. The elliptic-fitting strategy can fine-tune the segmentation results to achieve good accuracy. The method performs well in the face of the blocking problem on the VOT2018 dataset listed in Table 2, being superior to the other comparison algorithms, and performs worse than ECO, TADT, and VITAL on the VOT2016 dataset. These performance differences can be explained as follows: ECO uses more comprehensive features (CNN + HOG+CN) to cope with the blocking problem of the single feature target tracking algorithm. VITAL uses the generated confrontation network that randomly generates numerous membranes and retains the most robust membranes among the target features to increase the positive sample data. TADT uses pixel-level losses to guide channel selection, and the VITAL and the accuracy results of TADT are significantly higher than our algorithm; however, Compared to SiamMask our algorithm still achieves an improvement of 0.05 accuracy. From Tables 3 and 4, we can see that the strategy proposed here has achieved first place in terms of accepted average overlap (EAO), overlap, and failure metrics, with a strong overall performance, outperforming

Fig. 5 Expected overlap score (VOT2016)



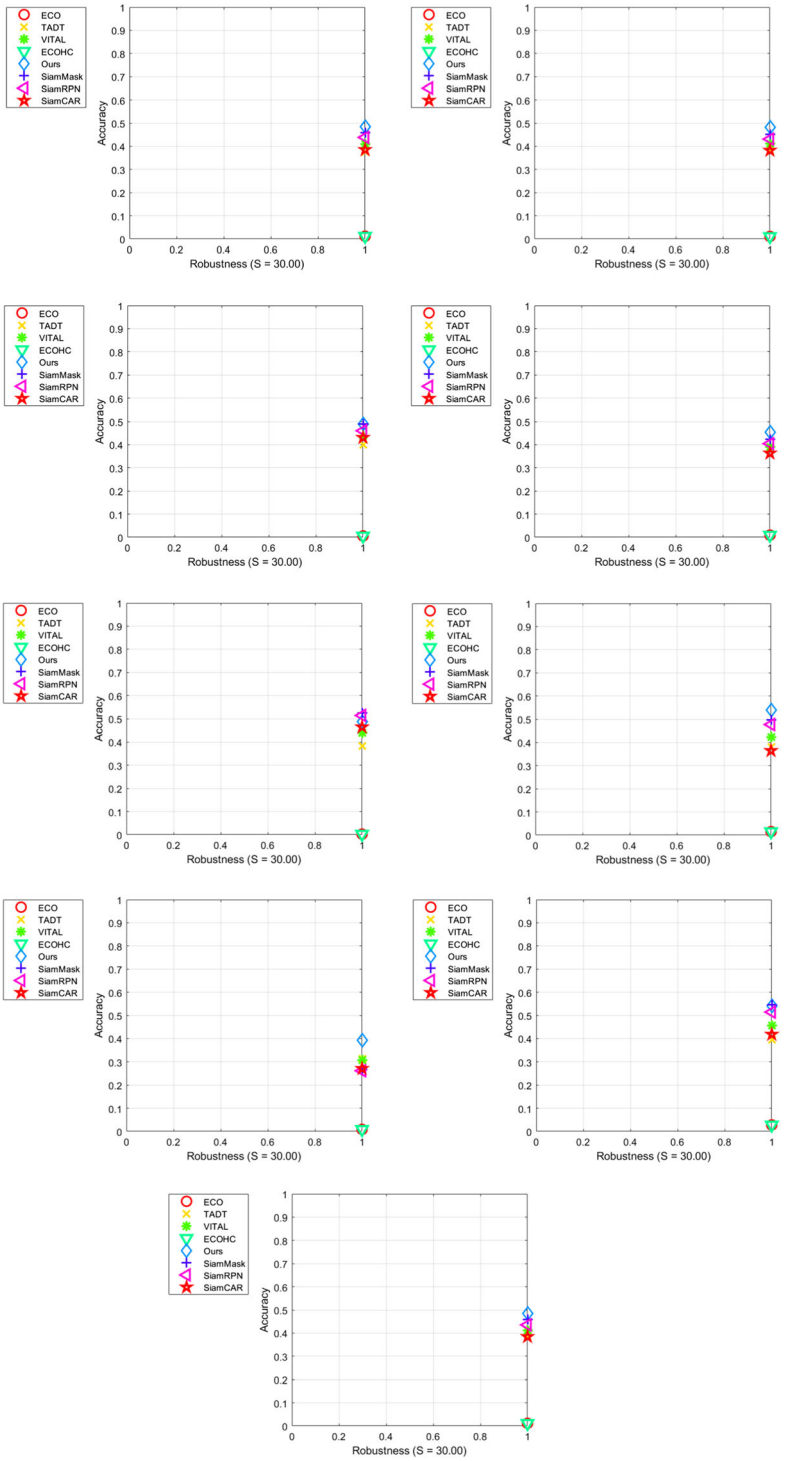


Fig. 6 EAO comparison results (VOT2018)

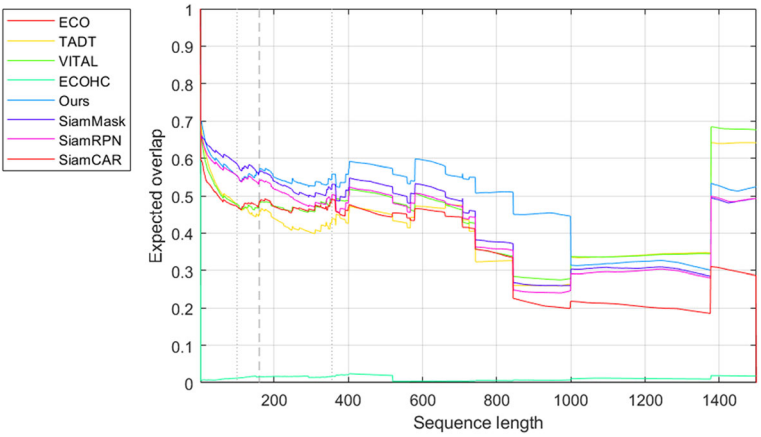


Fig. 7 Expected overlap curve (VOT2018)

SiamMask by almost 0.04, which indicates that the improvement of the algorithm is effective. Figures 3 and 6 list the A-R ranks of EAO metrics of various algorithms for nine types of videos in VOT2016 and VOT2018. It can be seen that the robustness and accuracy of the algorithm are higher than those other algorithms in most of the challenges.

From the expected overlap curves in Figs. 4 and 6, it can be seen that the algorithm will not be like ECO and other methods for which the coverage decreases significantly as the number of video frames increases, because the method adopts deep learning to extract features, and the depth features are more robust. The segmentation result is not easily affected by the previous target motion state, and the selection strategy adopted in Eq. (10) is also reasonable, even in the face of long video frames. For the video challenges, robustness is still guaranteed. From the expected overlap scores in Figs. 5 and 8, we see that the algorithm is much stronger than other algorithms in terms of the average expected overlap scores, indicating that our algorithm has high accuracy compared to other comparison algorithms. This strength also can be attributed to the segmentation branch we used and the elliptic-fitting strategy used to optimize the bounding box of the segmentation results.

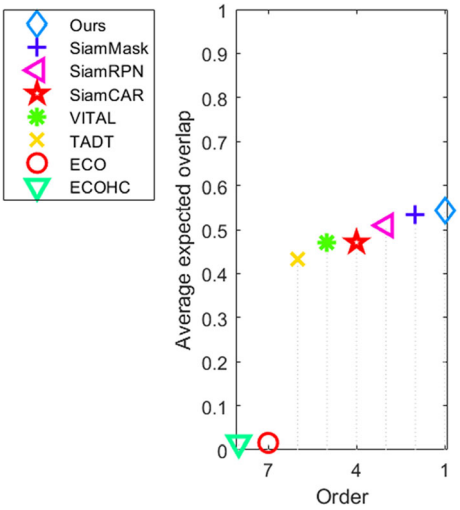


Fig. 8 Expected overlap score (VOT2018)

Figure 9 shows the real-time performance of this algorithm and other algorithms in multiple video frames, in which the red rotating rectangle shows the performance of this algorithm. It can be seen that the algorithm performs well in multiple video frames with large time intervals, which demonstrates the excellent stability of the algorithm Figs. 6, 7, 8 and 9.

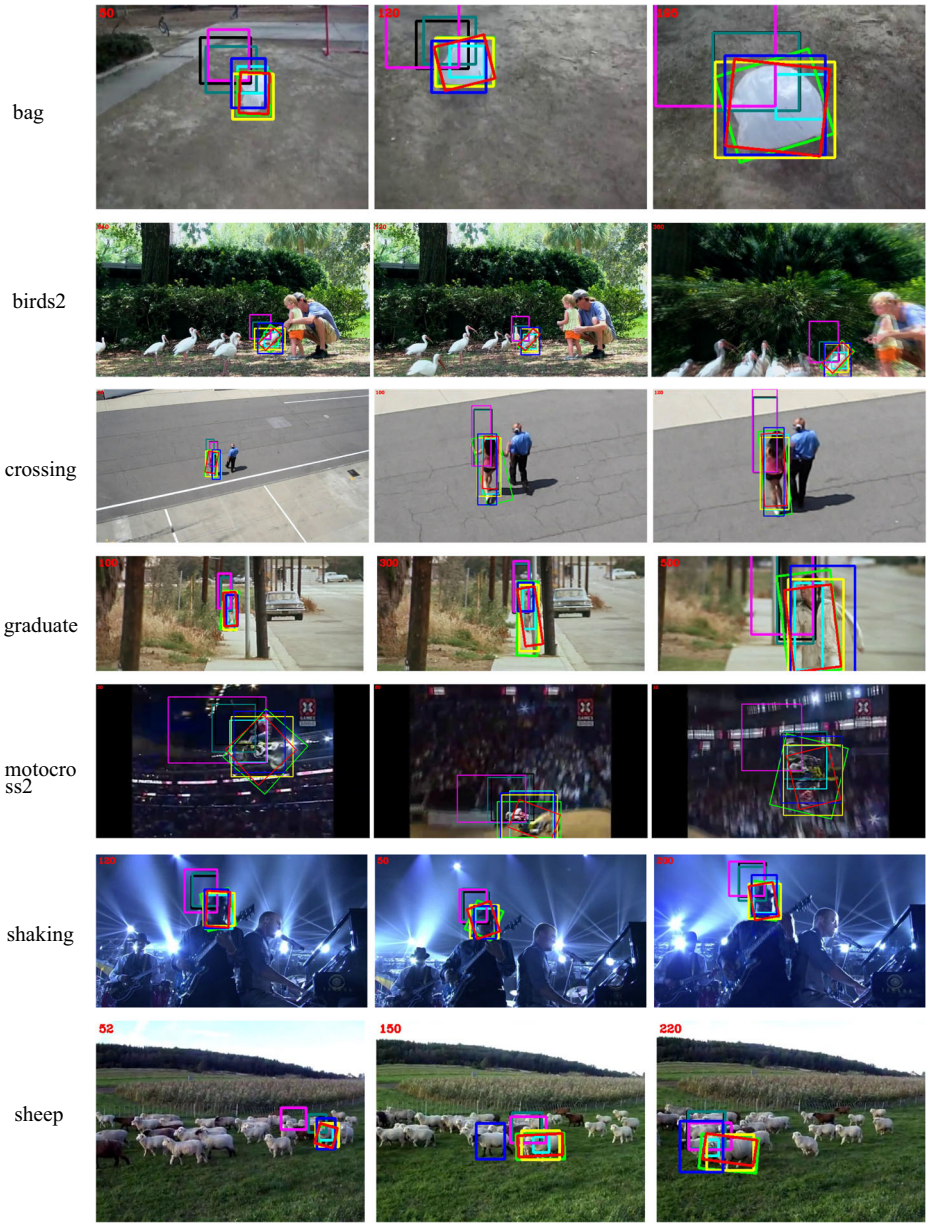


Fig. 9 Effect display diagram

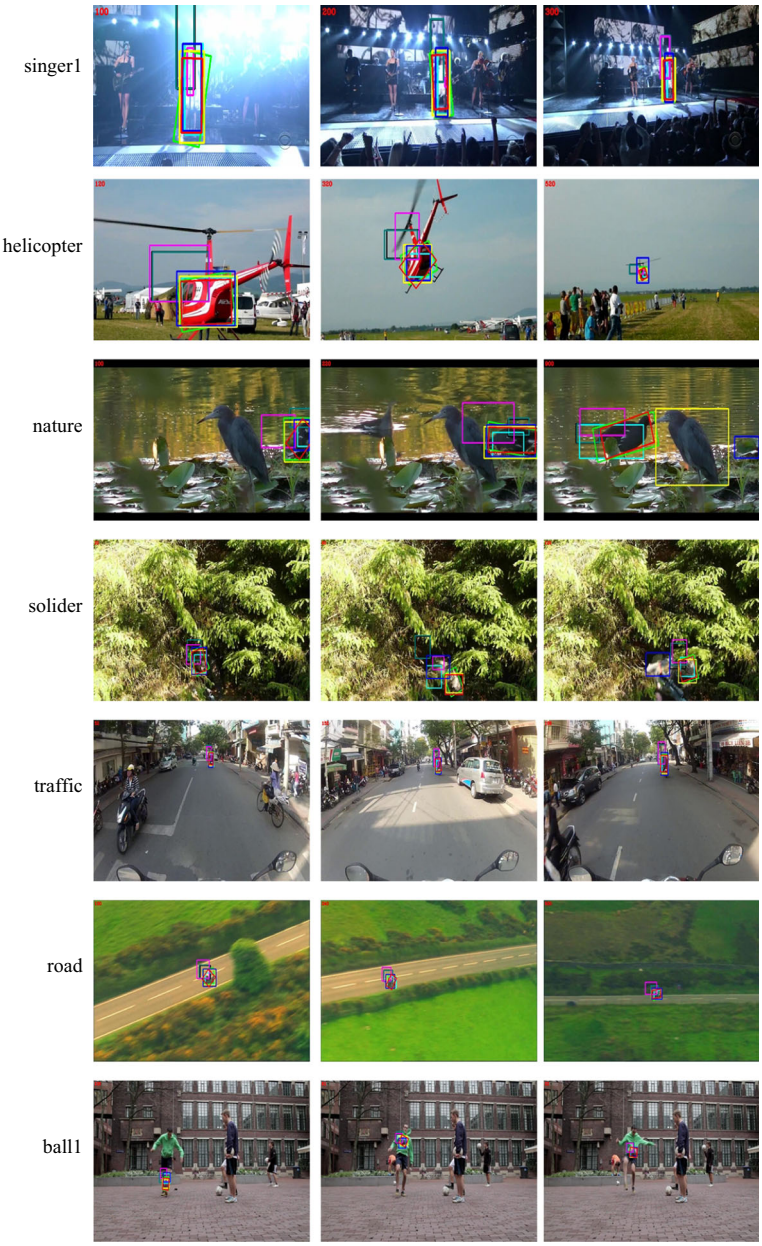


Fig. 9 (continued)

4.4 Future outlook

Although the experimental part of the algorithm has exhibited powerful advantages, remaining robust in the face of various challenges, the performance of the algorithm is still unsatisfactory under changing illumination in the video, as indicated in Table 1. The Kalman filter is not accurate enough

in predicting the target state, which is why it does not work well in the analysis of the results. The performance contribution of the tracker is not high enough. In the future, we can consider making full use of the time–space context information to predict the current state of the target more accurately by comparing the information between successive frames and construct a better tracker by combining detection or segmentation methods.

5 Conclusion

The performance of a tracker is commonly degraded when it is faced with a heavily occluded target because effective target features cannot be extracted. In view of this, a spatiotemporal fusion approach to motion target tracking and segmentation is proposed in this study. Based on Siamese networks and segmentation structures, the method utilizes a spatiotemporal motion target tracking model combined with Kalman filtering to mitigate the occlusion problem during tracking by extracting motion features of the tracked target on the time axis and building a motion model of the motion target on the time series. Because current target tracking methods neglect the importance of employing an online strategy, we propose to use Kalman filtering to construct a motion model of the target and to reasonably predict the motion of the target when the target is missing or heavily occluded in a short period of time. We use a segmentation network to segment the target from the background to achieve accurate tracking and elliptic-fitting strategy to correct the error caused by imprecise segmentation results and to improve the tracker's accuracy. The experiments demonstrate that this method is feasible and achieves excellent results when compared with other algorithms. However, there remain problems of insufficient segmentation accuracy and insufficient prediction accuracy. From [1, 6, 13, 23, 25, 31, 34], we can foresee that the algorithm combining segmentation and tracking will become more pervasively used in the future and that the target tracking method combining deep learning and traditional methods [12, 14, 18, 26, 33] has a bright future.

Acknowledgments This research is partially supported by National Natural Science Foundation of China(61866003).

Declarations

Conflict of interest The authors declare they have no conflict of Interests.

References

1. Ahrnbom M, Nilsson MG, Ardö H (2021) Real-time and online segmentation multi-target tracking with track revival re-identification. In: VISIGRAPP, pp 777–784

2. Bertinetto L, Valmadre J, Henriques JF, et al. (2016) Fully-convolutional siamese networks for object tracking. European Conference on Computer Vision. Springer, Cham, 850–865
3. Bolme DS, Beveridge JR, Draper BA, et al. (2010) Visual object tracking using adaptive correlation filters. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE
4. Cheng S, Zhong B, Li G, et al. (2021) Learning to filter: Siamese relation network for robust tracking. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4421–4431
5. Choi J, Jin Chang H, Jeong J, et al. (2016) Visual tracking using attention-modulated disintegration and integration. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4321–4330
6. Choudhuri A, Chowdhary G, Schwing AG (2021) Assignment-Space-based Multi-Object Tracking and Segmentation. Proceedings of the IEEE/CVF International Conference on Computer Vision, 13598–13607
7. Danelljan M, Hager G, Shahbaz Khan F, et al. (2015) Learning spatially regularized correlation filters for visual tracking. Proceedings of the IEEE International Conference on Computer Vision, 4310–4318
8. Danelljan M, Häger G, Khan FS et al (2016) Discriminative scale space tracking. IEEE Trans Pattern Anal Mach Intell 39(8):1561–1575
9. Danelljan M, Bhat G, Shahbaz Khan F, et al. (2017) Eco: Efficient convolution operators for tracking. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 6638–6646
10. Gu L, Liu J, Wang C, Cao M (2013) Particle filter tracking based on fragment multi-cue integration. Int J Appl Math Stats: 31–40
11. Guo D, Wang J, Cui Y, et al. (2020) SiamCAR: Siamese Fully Convolutional Classification and Regression for Visual Tracking. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6269–6277
12. Han K, Peng J, Yang Q, Tian W (2021) An end-to-end dehazing Siamese region proposal network for high robustness object tracking. IEEE Access 9:91983–91994
13. Han W, Lekamalage CKL, Huang GB (2022) Efficient joint model learning, segmentation and model updating for visual tracking. Neural Netw 147:175–185
14. Han X, Qin Q, Wang Y, et al. (2022) CS-Siam: Siamese-Type Network Tracking Method with Added Cluster Segmentation. International Conference on Advanced Data Mining and Applications. Springer: Cham, 251–262
15. He K, Gkioxari G, Dollár P, et al. (2017) Mask r-cnn. Proceedings of the IEEE International Conference on Computer Vision, 2961–2969
16. Henriques JF, Caseiro R, Martins P et al (2014) High-speed tracking with kernelized correlation filters. IEEE Trans Pattern Anal Mach Intell 37(3):583–596
17. Huang B, Chen J, Xu T, Wang Y, Jiang S, Wang Y, Wang L, Li J (2021) SiamSTA: Spatio-Temporal Attention based Siamese Tracker for Tracking UAVs , Computer Vision Workshops (ICCVW) 2021 IEEE/CVF International Conference on, pp. 1204–1212
18. Jiang S, Xu B, Zhao J, Shen F (2021) Faster and simpler siamese network for single object tracking. https://doi.org/10.48550/arXiv.2105.03049
19. Kiani Galoogahi H, Fagg A, Lucey S (2017) Learning background-aware correlation filters for visual tracking. Proceedings of the IEEE International Conference on Computer Vision, 1135–1143
20. Kiran M, Nguyen-Meidine LT, Sahay R, Cruz RMOE, Blais-Morin LA, Granger E (2022) Generative target update for adaptive siamese tracking. https://doi.org/10.48550/arXiv.2202.09938
21. Li B, Yan J, Wu W, et al. (2018) High performance visual tracking with siamese region proposal network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8971–8980
22. Li B, Wu W, Wang Q, et al. (2019) Siamrpn++: Evolution of siamese visual tracking with very deep networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4282–4291
23. Lukezic A, Matas J, Kristan M (2020) D3S-A Discriminative Single Shot Segmentation Tracker. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7133–7142
24. Mueller M, Smith N, Ghanem B (2017) Context-aware correlation filter tracking. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1396–1404
25. Noor S, Waqas M, Saleem MI et al (2021) Automatic object tracking and segmentation using unsupervised SiamMask. IEEE Access 9:106550–106559
26. Oleksiienko I, Iosifidis A (2022) 3D object detection and tracking. Deep Learning for Robot Perception and Cognition. Academic Press, 313–340
27. Ren S, He K, Girshick R et al (2016) Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149
28. Song Y, Ma C, Wu X, et al. (2018) Vital: Visual tracking via adversarial learning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8990–8999
29. Wang Q, Zhang L, Bertinetto L, et al. (2019) Fast online object tracking and segmentation: A unifying approach. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1328–1338
30. Wang N, Song Y, Ma C, et al. (2019) Unsupervised deep tracking. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1308–1317

31. Wang J, He Y, Wang X, Yu X, Chen X (2019) Prediction-tracking-segmentation. https://doi.org/10.48550/arXiv.1904.03280
32. Xu J, Xun J et al (2012) Data fusion for target tracking in wireless sensor networks using quantized innovations and Kalman filtering. SCIENCE CHINA Inf Sci 55(03):530–544
33. Yang D (2022) Research on multi-target tracking technology based on machine vision. Appl Nanosci:1–11. https://doi.org/10.1007/s13204-021-02293-6
34. Yao R, Lin G, Xia S, Zhao J, Zhou Y (2020) Video object segmentation and tracking: A survey. ACM Transactions on Intelligent Systems and Technology (TIST):1–47. https://doi.org/10.1145/3391743
35. Yin H, Chai Y, Yang SX, Yang X (2011) Fast-moving target tracking based on mean shift and frame-difference methods. J Syst Eng Electron 22(04):587–592
36. Zhang J, Jin X, Sun J et al (2020) Spatial and semantic convolutional features for robust visual object tracking. Multimed Tools Appl 79(21):15095–15115

Jie Wang is currently pursuing his master's degree at the Guangxi Minzu university, Nanning, China. His main research interests include image processing and pattern recognition.

Shibin Xuan received his PhD in computer science and technology from Sichuan University, Chengdu, China, in 2011. He is currently a full professor and master's supervisor in the School of Information Science and Engineering, Guangxi Minzu university, Nanning, China. His main research interests include image processing and pattern recognition.

Hao Zhang is currently pursuing his master's degree at the Guangxi Minzu university, Nanning, China. His main research interests include image processing and deep learning.

Xuyang Qin is currently pursuing his master's degree at the Guangxi Minzu university, Nanning, China. His main research interests include image processing and pattern recognition.