

Received 18 September 2023, accepted 18 October 2023, date of publication 1 November 2023, date of current version 8 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3329063

RESEARCH ARTICLE

Infrared Small Target Tracking Based on OTrack Model

JIANHUA SHAN¹, YU YANG¹, HENG LIU², AND TAO LIU¹

¹School of Mechanical Engineering, Anhui University of Technology, Maanshan 243032, China

²School of Computer Science and Technology, Anhui University of Technology, Maanshan 243032, China

Corresponding author: Tao Liu (lt_ahut@126.com)


This work was supported in part by the Natural Science Research Project of the Anhui Educational Committee under Grant 2022AH050306, and in part by the Equipment Advanced Research Project (Sharing Technology) under Grant 80912020104.

ABSTRACT Infrared small target tracking plays an important role in military reconnaissance, early warning, video surveillance, and civil applications. For tracking small infrared targets in this paper, a one-stream deep learning model is utilized. In order to integrate the processes of feature extraction and feature fusion, the model uses transformer as the framework's major component and creates a bidirectional information flow between the template and the search picture pairs in the feature extraction stage. Use the head of the model to get the target position. Finally, post-processing of the target area, including tracking success, saves the coordinate information of the target frame; tracking failure, near, in the middle, and far from the target box, searches for the real target. It helps to solve the situation where the target moves fast and encounters a complex background to achieve better tracking results. It is tested on an infrared small target data set, and the results show that the method in this paper reaches 80.50% average tracking accuracy. The image sequences in the data set include sky, sea, and buildings. Tracking video and original images are shown at <https://github.com/AHUT507LAB/Infrared-dim-small-target-tracking>.

INDEX TERMS Infrared small target, small target tracking, post-processing, deep learning.

I. INTRODUCTION

As an important target tracking technology, infrared small target tracking has always been a difficult and hot spot in the field of target tracking and has been widely used in military and civilian fields. In military reconnaissance, early warning, video surveillance, and other aspects, the rapid movement of drones is a major problem that needs attention. In the civilian field, it is more difficult to track the UAV in the complex background. Due to the weak brightness, small size, and lack of obvious shape, texture, and color information of infrared small targets, it is inevitable that a variety of complex backgrounds will appear in the tracking, and some occasions have high requirements for real-time tracking, making the tracking of infrared small targets a huge challenge [1]. A large number of scholars at home and abroad have done a lot of research on infrared small targets and proposed a variety of algorithms.

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

Huang et al. [2] proposed an efficient small target location algorithm initialized using a strong detector created from the shape analysis of foreground spots and a particle filter-based tracker that can handle the fuzziness of template matching. The improved template matching algorithm (TMT), proposed by Liu et al. [3], [4], performs well by calculating correlation coefficients in high-dimensional feature spaces. However, when the background clutter is strong, it is difficult to track the small target accurately for a long time. The improved particle filter (PF) algorithm [5], [6] uses non-negative matrix decomposition to integrate the weight of the current and previous particle distributions, reduces the accuracy error caused by the degradation divergence of the traditional PF algorithm, and achieves a good tracking effect. However, the improved particle filter algorithm only uses the target position and velocity state for feature filtering to maintain tracking and does not consider shape feature information, so it is not suitable for small target tracking. Wei et al. [7], [8] proposed a tracking algorithm based on mean shift and particle filter (MS-PF) to describe the target by using the

statistical characteristics of the image. Under the theoretical framework of particle filtering, sample weights were updated in the iterative process of mean shift. To a certain extent, it overcomes the deficiency of target feature information, but it cannot distinguish weak targets from the edge of the cloud background. Zhang et al. [9] package correlation filtering into probability theory, using intensive sampling to track a surface target. Li et al. [10] proposed a tracking algorithm based on particle filters to solve the uncertainty problem in small target tracking. A differentiated overcomplete dictionary is used to widen the difference between target and background particles. In addition, particle discrimination with sparse representation can improve the accuracy of target motion estimation by increasing the weight of target particles. However, this algorithm does not focus on obtaining an adequate description of the target characteristics. The MS algorithm [11], [12] uses a kernel-weighted gray histogram to represent the appearance of the infrared target and uses the mean shift algorithm to determine the most likely location of the target in the next frame. He et al. [13] proposed a tracking algorithm based on low-rank representation and weighted correlation filtering. In the tracking process, multi-feature weight functions were integrated into correlation filters. However, as the image size increases, the tracking process takes more time to obtain tracking results.

Convolutional neural networks are gradually taking the place of the original infrared small target tracking techniques. More deep learning techniques are used in the present tracking techniques. Although the deep learning approach has great tracking accuracy, the running pace will be slightly slower.

Considering these problems, a large number of scholars have conducted research. Tadej and Leon [14] proposed a multi-scale convolutional neural network, considering that the noise of infrared tracking systems and the lack of robustness features make it difficult to detect space objects within a relatively long observation distance. Deng et al. [15] proposed a thermal infrared target tracking algorithm based on a convolutional neural network. Considering the lack of spatial information, only using the features of the fully connected layer or the features of a single convolutional layer is not suitable for infrared target tracking. Therefore, the algorithm proposes multi-layer convolutional features for thermal infrared tracking (MCFTS) based on correlation filters. The proposed algorithm provides a new idea for infrared moving small target tracking based on neural networks; that is, if the tracking problem is regarded as binary classification, more accurate tracking results can be obtained through the integration of multiple weak classifiers. Fan et al. [16] proposed a convolutional neural network enhancement method that can enhance small targets and suppress background clutter at the same time, aiming at the common problems of the current infrared imaging system caused by the long shooting distance. However, it lacks a large number of data sets for training.

The above shows the feasibility of the deep learning model, but there may be problems in its practical application. Compared with other methods, the method in this paper has a very good effect. This method sets up a free information flow between the template and the search area (i.e., the original image pair) so as to extract the target-oriented features and avoid the loss of discriminant information. Get the prediction through your head. The core idea of this paper is to post-process the selected area of the box after getting the predicted result, so as to achieve better reasoning results and better tracking effects.

The following are the main contributions to this paper:

- (1) The added part of post-processing greatly improves the tracking effect of fast-moving objects in complex environments.
- (2) The method in this paper can achieve real-time under the GPU with less time spent on post-processing.
- (3) This method adopts the transformer framework for tracking.

II. RELATED WORK

In this section, we mainly introduce the small target tracking method and the post-processing module added in this paper.

Correlation filtering algorithm: The basic idea is to design a filtering template and use the template to do correlation operations with the target candidate region. The maximum output response position is the target position of the current frame. MOSSE is the first work on correlation filtering tracking proposed by Bolme et al. [17], which uses multiple samples of targets as training samples to generate better filters. The objective function of MOSSE is to minimize the sum of squares error, and m samples are used to find the least squares solution. Henriques et al. [18] put forward the CSK method, a kernel tracking method based on cyclic matrices, to mathematically solve the problem of dense sampling. The Fourier transform is used to quickly realize the tracking process, but the disadvantage is that it will bring boundary effects. Henriques et al. [19] proposed the KCF method. Compared with the traditional correlation filter algorithm, the KCF algorithm has higher tracking accuracy and faster tracking speed. In the field of target tracking, the KCF algorithm has been widely used and proven to be an efficient and reliable target tracking algorithm. However, this method does not work well when the target is blocked. Lukei et al. [20] proposed the CSRDCF method and introduced a multi-scale tracking strategy to further improve the robustness and accuracy of tracking by tracking targets at different scales. The disadvantage is high computational complexity and sensitivity to target deformation.

Deep learning method: Bertinetto et al. [21] proposed the siamFC method, which used two convolutional neural networks with shared weights to extract the features of template images and search images, respectively, and then determined the location of the target by calculating their similarity. This method has good tracking results when the background

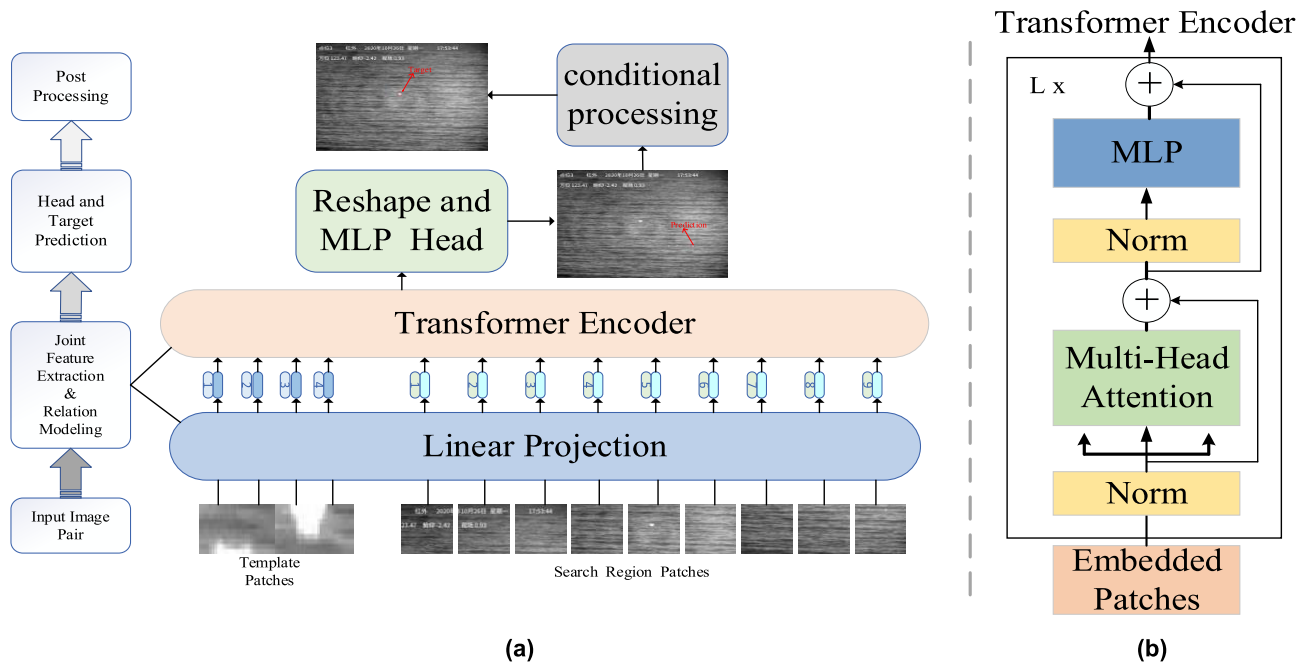


FIGURE 1. (a)The overall framework of the method in this paper Templates and search areas are segmented, linearly projected, and fed into the transformer encoder layer with learnable location coding for joint feature extraction and relationship modeling. After reshaping the head, the target prediction results are obtained, and then the predicted results are post-processed to get the final target frame selection results. The input of the model is a pair of pictures, and the output is the box selection result of the object.(b) The structure of the transformer encoder, which is insert after the multi-head attention operation.for subsequent target classification and regression.

information changes a little in a small range. The disadvantage is that when the deformation is large, there will be a large difference, which will lead to tracking failure. Li et al. [22] proposed the SiamMask method, which uses a neural network to learn the appearance and motion features of the target and track the target in real time in video. However, when the object is partially occluded, the tracking effect will be poor. Wang et al. [23] proposed the SiamRPN method, which utilized the feature extraction capability of deep neural networks to transform the target tracking problem into a similarity measurement problem between the target and the candidate region. However, this method does not do template updates and is time-consuming. Cao et al. [24] proposed the TCTrack method to balance speed and performance by introducing timing information in two dimensions. However, it is very time-consuming.

Ye et al. [25] model, a post-processing module was added in this paper to process the predicted results. Trace success, save the target's coordinate information, and update the template. Tracking failure requires searching and predicting the image. It is helpful to solve the problem that the target moves fast and the background is complicated.

III. METHOD

This section describes the specifics of the proposed methodology and the features of each step. The input image pairs are subjected to feature extraction and relational modeling, then

the target positions are obtained from the head of the model, and finally the target positions are post-processed to generate the final position. An overview of the model is shown in Figure 1.

A. JOINT FEATURE EXTRACTION AND RELATIONSHIP MODELING

This article's approach uses Vision Transformer as the main body of the framework, so we can also use a pre-training model of the Vision Transformer architecture, reducing the time we spend training. Generally, Vision Transformer divides the input image into several patches and then projects each patch as a vector of fixed length into transformer. Subsequent encoder operations are exactly the same as those in the original transformer. The method in this paper inputs a pair of images, template image patch z and search area patch x , and splits the pair of images into multiple patches. Then, the patch after z and x segmentation is projected into the D -dimensional potential space using a trained linear projection layer with parameter E , and the output of the projection is usually called patch embeddings. After that, the learnable location encoding will be added to the patch embeddings to obtain the token embedding, and finally, the two will be spliced together into the transformer encoder layer. In the transformer encoder, after Layer Normalization (LN), the output dimension remains unchanged, and after Multi-Head Attention, inputs are mapped to q, k , and v . If there is only

one head, the dimensions of q, k, and v will not change. If you have n heads, you have n sets of q, k, and v, and finally you concatenate the output of n sets of q, k, and v, and the output dimension remains the same, and then you go through a layer of LN, and the dimension remains the same. Finally, through MLP, the dimension is enlarged and shrunk back, and the dimension remains unchanged. After a block, the dimensions remain the same as the input, so multiple blocks can be stacked. Input the final output of the transformer encoder into the head to get the final trace result. When the method in this paper processes each search, the template image will be input into the model together to obtain dynamic template features, but it has little impact on the speed of reasoning. In this paper, both feature extraction and relationship modeling can be realized. Self-attention in Transform can be described as operation A:

$$\begin{aligned} A &= \text{Soft max}\left(\frac{QK^T}{\sqrt{d_k}}\right) \times V \\ &= \text{Soft max}\left(\frac{[Q_z; Q_x][K_z; K_x]^T}{\sqrt{d_k}}\right) \times [V_z; V_x] \end{aligned} \quad (1)$$

Attention weight calculation can be described as:

$$\begin{aligned} &\text{Soft max}\left(\frac{[Q_z; Q_x][K_z; K_x]^T}{\sqrt{d_k}}\right) \\ &= \text{Soft max}\left(\frac{[Q_z K_z^T, Q_x K_x^T; Q_z K_x^T, Q_x K_z^T]}{\sqrt{d_k}}\right) \\ &\triangleq [w_{zz}, w_{zx}, w_{xz}, w_{xx}] \end{aligned} \quad (2)$$

The self-attention operation A can be further described as:

$$A = [W_{zz}V_z + W_{zx}V_x; W_{xz}V_z + W_{xx}V_x] \quad (3)$$

It contains not only the self-attention of the template image and the search image but also the cross-attention of the two. Self-attention is the feature of extraction, and cross-attention represents the relationship model between them.

B. HEAD AND LOSS

The structure of the head section is also relatively simple, including three branches, which respectively predict classification score P, offset value O predicted to compensate for downsampling quantization error, and normalized bounding box size S. Each branch is stacked with L convolution layers. The point with the highest predicted score is taken as the target position (x_d, y_d) , and the value of the corresponding position is taken out of O and S to calculate the final target enclosing box.

$$\begin{pmatrix} x, y \\ z, w \end{pmatrix} = \begin{pmatrix} x_d + O(0, x_d, y_d), & y_d + O(1, x_d, y_d) \\ S(0, x_d, y_d), & S(1, x_d, y_d) \end{pmatrix} \quad (4)$$

In terms of loss function, for classification branches, the same weighted focal loss as that in CornerNet was adopted, and the position farther away from the GT center had a

lower weight. For the iou for the regression branch, use the commonly used iou loss as well as the combination of the ℓ_1 loss. Overall loss function:

$$L_{track} = L_{cls} + \lambda_{iou}L_{iou} + \lambda_{L1}L_1 \quad (5)$$

C. POST-PROCESSING OF THE BOX SELECTION AREA

Before obtaining the post-processing result, we will calculate the target area image obtained in the previous frame to obtain the average pixel value of the target area image in the previous frame. Defines the average length and width of the target, starting with the length and width of the artificially selected target box in the first frame. Define the average horizontal and average vertical velocity of the target, starting with 0. After obtaining the target frame of the current frame, calculate the horizontal speed, vertical speed, and speed of the frame before the target and the current frame. These values are used to judge whether the camera is moving. When the camera is moving, some calculation processing is carried out on the image of the target area to calculate the pixel proportion value of the previous frame in the target area, the image pixel change proportion value of the current frame in the target area, and the image pixel proportion value of the current frame in the target area. Record the number of camera movements. The camera is not moving. Set the three values for the camera movement calculation to default values. Then judge whether there is any abnormal situation. (1) the camera does not move and the target frame is too large; (2) when the camera moves, the proportion value of image pixels in the target area of the current frame is too large, and the proportion value of image pixels in the target area of the current frame is too large, and the target speed is too large.

Nothing unusual happened, which means the object didn't blend into the background. Calculate the scale factor K (related to the number of camera moves). Update the target box in two cases. (1) The length and width of the target frame are not greater than the length and width of the previous frame multiplied by the scale factor K, indicating successful tracking. The coordinate parameters of the target frame need to be saved, and the average length and width and average horizontal and vertical speeds need to be updated. (2) The length and width of the target frame should be at least one larger than those of the previous frame multiplied by the scale coefficient K, and the target position can be obtained by using the last saved coordinate parameters and average speed.

When an abnormal situation (1) occurs, the image pixels in the target area of the current frame are recalculated to change the proportion value of the image pixels in the target area of the current frame. Calculate the maximum proportional value of image pixel change in the target area, which is the maximum value of the first two values. Then it is necessary to judge whether the target is integrated into the background. In two cases, the target is considered to be integrated into the

background: (1) When the change ratio of image pixels in the target area of the current frame is too small, the target is integrated into the background; (2) When the maximum proportion of image pixel change in the target area is small and the horizontal speed and vertical speed are small, the target is integrated into the background. The target is blended into the background, and the target frame of the previous frame is used as the target frame of the current frame. In other cases, the target is not integrated into the background, and it needs to be searched near, in the middle, or far away from the target frame of the current frame. When looking for a target in the nearby area, the step size in the x direction of the image coordinate system is one-sixth of the average length, the step size in the y direction of the image coordinate system is one-sixth of the average width, the number of cycles is 3, the elements in the list are $-1, 0, 1$ randomly selected, and the length is 3. When looking for the target in the middle region, the step length in the x direction of the image coordinate system is a quarter of the average length, the step length in the y direction of the image coordinate system is a quarter of the average width, the number of cycles is 4, the elements in the list are $-1, 0, 1$ randomly selected, and the length is 4. When looking for the target in the distant area, the step length in the x direction of the image coordinate system is 1.25 times divided by 9 of the length of the target frame in the previous frame, the step length in the y direction of the image coordinate system is 1.25 times divided by 9 of the width of the target frame in the previous frame, the number of cycles is 12, the elements in the list are $-1, 0, 1$ randomly selected, and the length is 12. The number of loops is reduced by one for each loop; the value in the x direction in the upper left corner of the target box of the current frame is the value in the x direction of the target box of the previous frame plus the step length in the x direction, the average length, and the corresponding element in the current loop value list. In the same way, the value of the y direction in the upper left corner of the target box of the current frame can be obtained. Calculate the image pixel value of each target box during the loop, add these values to a new list, and when the loop ends, take the image pixel value of the largest target box as the tracking result, along with the height and width of the target box and the previous frame of the same. When an exception (2) occurs, the processing procedure is the same as when an exception (1) object does not blend into the background.

Secondly, judge whether the target is lost. If the target is integrated into the background and the ratio of image pixel change in the target area of the current frame is small, it is considered that the target is lost. If the target does not blend into the background, the target frame is too large, and the target is considered missing. When the object is lost and the camera does not move, the speed of the object and the last saved coordinate parameters are used to obtain the frame selection result of the frame. When the object is lost and the camera moves, the last saved coordinate parameters are used

as the result of the current frame. When the target is not lost and the target is integrated into the background, the previous saved coordinate parameters need to be predicted to get the frame selection result for the current frame.

Finally, the length and width of the target box are judged. When the gap between the length and width of the target box is too large, the average value of the length and width is used to replace the length and width, and the pixel coordinates of the upper left corner of the target box are processed to obtain the final result and update the target box. We use multiple judgment conditions and related data values to process the situation of fast-moving targets and complex backgrounds, search for targets near, in the middle, and far from the target box, and predict the target position of the current frame by using the target box information of the previous frame and the target information of the current frame. Whether the selected area of the box is post-processed or not, the tracking result is shown in Figure 2.

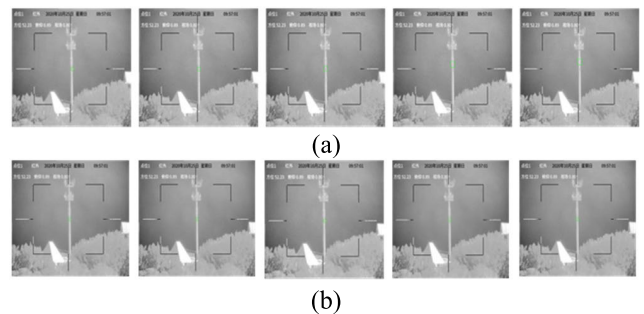


FIGURE 2. Post-processing is carried out on the selected area of the frame. (a) The box selection area is not post-processed; (b) Post-process the box selection area.

IV. EXPERIMENT

A. EXPERIMENTAL PLATFORM AND DATA SET

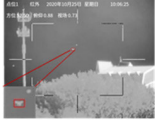




The experiments were carried out on a computer with 16 GB of RAM and a 2.60 GHz Intel i7-6700HQ CPU. The code was implemented in Pycharm on Ubuntu 18.04.6 LTS.

The experiment uses the collected infrared UAV data set Anti_UAV [26], which contains more than 300 videos and is stored as MP4 files, among which there are many videos of fast-moving UAVs and complex backgrounds. The image sequences in the dataset include various backgrounds (sky, sea, building, forest) and two lighting conditions (day and night). The image resolution is 640×512 . The frame rate of the image is 25 frames per second. Table 1 shows images of representative data sets for five typical conditions. These include IN (interference), IB (integration background), LI (low illumination), CB (complex background), and FM (fast motion). Use the red rectangle box to select the target. Target sizes are all about 10×10 .

B. EVALUATION INDEX

For a comprehensive evaluation, tracking accuracy was calculated to describe the positioning capability of the method.

TABLE 1. Five typical cases represent detailed data images.

File name	Image examples	Resolution	frame	Situation
N3		640x512	1025	Interference
N4		640x512	1489	Integration Background
N6		640x512	900	Low Illumination
N11		640x512	2000	Complex Background
N14		640x512	1674	Fast Motion

The spatial tracking performance of the IOU evaluation model was calculated.

1) IOU

The IOU computes the ratio of the intersection (A_{Inter}) and union area (A_{All}). It is defined as follows:

$$IOU_t = \frac{A_{Inter}}{A_{All}} \tag{6}$$

where t represents each frame in the video.

2) ACC

The Acc computes tracking accuracy for all frames. The definition is as follows:

$$Acc = \sum_{t=1}^T \frac{IOU_t + p_t}{T} - 0.2 \times \left(\sum_{t=1}^T \frac{p_t}{T} \right)^{0.3} \tag{7}$$

where p_t equal to 1 when IOU_t equal to 0, otherwise equal to 0. T represents the total number of frames.

3) FPS

The FPS computes the ratio of the F and t. It is defined as follows:

$$FPS = \frac{F}{t} \tag{8}$$

where F represents the total frame number of the infrared small target video, and t represents the total time used by this method to track the infrared weak target video.

C. OBSTA COMPARISON WITH ALOGRITHM

Several experiments were carried out to test the method in Section III. In order to show the effectiveness of the proposed method, we compare it with eight other methods. Including CSK, KCF, CSRDCF, SiamFC, SiamMask, SiamRPN, OTrack, and TCTrack. The first three are based on correlation filtering, and the last five are based on deep learning. This paper did not fine-tune the model but used the original weights given by the original author on GitHub. We use equations (6) and (7) to calculate the accuracy of five typical cases to evaluate the effectiveness of the method discussed. Table 2 shows the validity results and real-time performance of these methods in five typical cases on the data set. The higher the average tracking accuracy of the nine methods in five typical cases, the better the validity. When we use better equipment, FPS gets a big boost.

TABLE 2. Uses different methods to calculate ACC and average FPS in five typical cases. The larger the value of ACC, the better the tracking effect, and the larger the value of average FPS, the better the real-time performance.

	N3	N4	N6	N11	N14	Average	FPS
CSK	1.26	6.07	1.12	12.17	0.45	4.21	27.3
KCF	5.43	11.45	7.68	17.54	2.54	8.93	31.3
CSR-DCF	10.38	13.78	12.47	20.57	2.35	11.91	30.6
SiamFC	65.42	67.56	74.21	73.83	1.26	56.45	24.2
SiamMask	68.86	65.79	72.00	72.87	1.17	56.14	14.6
SiamRPN	75.45	50.01	75.46	68.69	1.32	54.19	21.4
TCTrack	71.52	7.17	64.08	77.25	1.54	44.31	16.7
OTrack	46.17	57.98	61.35	72.80	53.05	58.27	13.2
Ours	86.46	83.33	77.98	91.37	63.36	80.50	12.8

The tracking speed of the method based on correlation filtering is fast. However, in the case of complex backgrounds and fast-moving objects, the effect of the algorithm will be poor. Methods based on deep learning are generally superior to algorithms based on correlation filtering. This is helped by the flexibility of deep learning. However, this approach requires large data sets to train, limiting their performance and The bottleneck of feature representation and sensitivity to regression branches [27] are also major issues. It is important to ensure good feature expression ability while avoiding information loss caused by deep networks. The sensitivity of regression branches affects not only the allocation of positive and negative samples but also the overall optimization of the

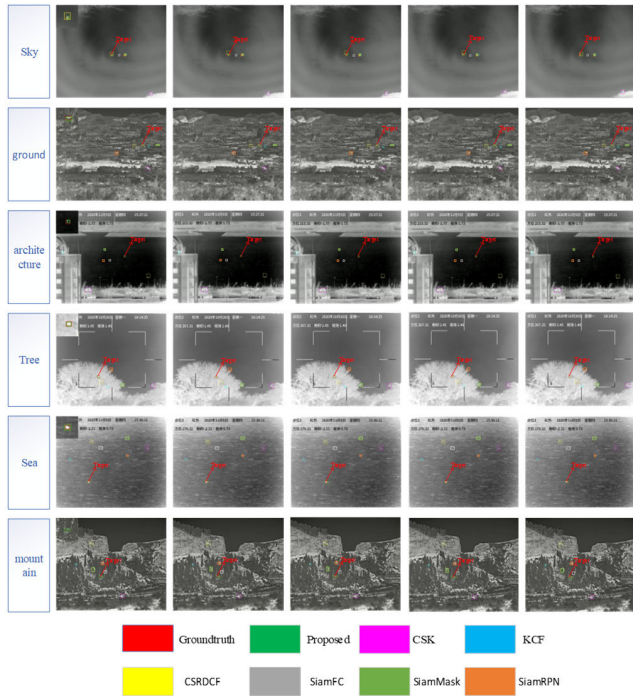


FIGURE 3. On the public data set, the target object has been marked with text, and the image is enlarged in the upper left corner. The pictures are continuous frames, and the target template of the first frame of the video is correct. The method in this paper is compared with the single target tracking algorithm based on correlation filtering. The experimental results show that the algorithm based on correlation filtering has a higher false alarm rate, and the tracking frame of the method in this paper is the most consistent with the real frame and has a better tracking effect. The method in this paper is compared with the single-target tracking algorithm based on deep learning. The experimental results show that the method in this paper does not have tracking errors and realizes continuous tracking.

network. The method in this paper performs optimally under various attributes and has better tracking performance than other methods under various attributes. The tracking speed of OTrack is almost the same, but the accuracy has improved a lot. The method proposed in this paper has better robustness to the fast motion and complex background of infrared small targets.

FIG. 3 shows the comparison of tracking results between the proposed method and the previous six tracking methods under six backgrounds. It can be seen that the proposed method has a higher tracking rate and better robustness against various backgrounds.

FIG. 4 shows the comparison between the proposed method and the newer algorithm TCTrack in five typical cases.

Figure 3 shows: (1) the tracking results between the method in this paper and the method based on correlation filtering. The real-time performance of the correlation filtering method is okay. However, with complex background interference and a fast-moving target, it is difficult for the correlation filtering algorithm to find the real target, and it is easy to produce false boxes, resulting in a high false tracking rate. However, our method achieves excellent tracking results.

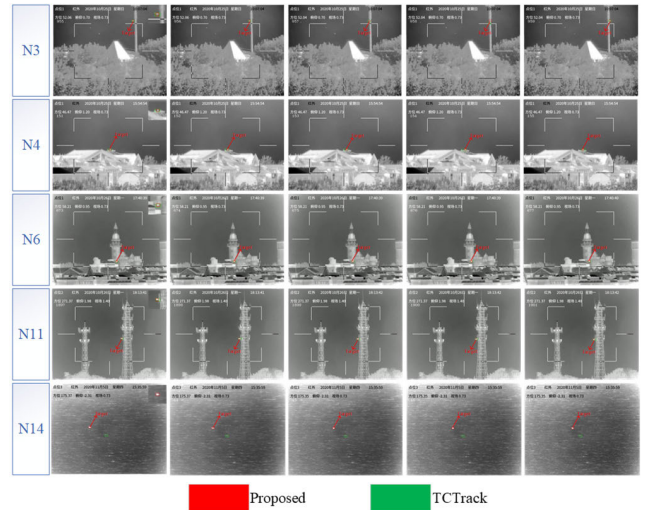


FIGURE 4. On the public data set, the target object has been marked with text, enlarged in the upper right corner of the image; the image is a continuous frame; the frame number is marked in the upper left corner of the image; and the target template of the first frame of the video is correct. The method in this paper and the TCTrack algorithm are used to track in five typical cases. The tracking results show that the tracking box of the proposed method is more consistent with the real target, and the tracking accuracy is higher.

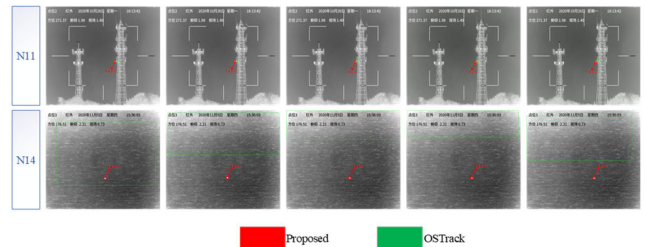


FIGURE 5. Test the effects of n11 and n14 videos using the Ostrack method and the text method, the target object has been marked with text. The experiment shows that the OTrack method has tracking failures in both kinds of videos, and some target boxes are too large to achieve continuous tracking. The method in this paper successfully tracks the small targets in two videos, and the tracking effect is good, and continuous tracking is realized.

(2) the tracking results between the method in this paper and the deep learning-based approach. On the whole, good results were produced. In public data sets, tracking based on deep learning is effective when the background is not particularly complex, such as the sky background. However, when the background becomes more complex and the target is blocked, the tracking effect will become worse. Different from the methods based on deep learning, the method proposed in this paper presents a post-processing module that is conducive to improving the tracking rate of infrared small targets and helping to solve the problems of fast-moving infrared small targets with complex backgrounds. Tracing results show that the method in this paper is useful and can be used in practice.

Figure 4 shows the tracking results of the proposed method and the newer algorithm TCTrack in five typical cases.

The tracking accuracy of the proposed method is better, and the tracked rectangle box is more consistent with the real target.

We compared the tracking performance of this article's approach to the OTrack method using videos of complex backgrounds and fast-moving targets in the collected data set. The trace effect is shown in Figure 5.

Figure 5 shows the trace effect of this article's method and the OTrack method. In general, the method presented in this paper has a good tracking effect on complex backgrounds and fast-moving targets. Therefore, the improvements in this paper based on the OTrack method are effective.

V. CONCLUSION

This paper is based on OTrack modeling to track small infrared targets. Joint feature extraction and relational modeling are performed on the input image pairs. Then, the head of the model is used to obtain the position information of the target. Finally, the target position is post-processed to get the final result. The addition of a post-processing component helps solve the problem of complex backgrounds and fast-moving objects. Experiments show that the method in this paper is superior to other methods. The method can not only track small infrared targets in complex backgrounds but also fast-moving infrared targets.

Finally, the method proposed in this paper achieves effective tracking and is expected to be applied to remote monitoring and early warning of anti-UAV. However, the method used in this paper also has some shortcomings. When the object is obscured, for example, when it is hidden behind a tree or building, a lost pursuit may occur. So in the future, we need to find solutions to the deficiencies.

ACKNOWLEDGMENT

The authors would like to thank the editors and anonymous reviewers for the time and effort spent in handling this study and constructive comments provided to improve the presentation and quality.

REFERENCES

- [1] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7380–7399, Nov. 2022.
- [2] Y. Huang, J. Llach, and C. Zhang, "A method of small object detection and tracking based on particle filters," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.
- [3] R. Liu, Y. Lu, C. Gong, and Y. Liu, "Infrared point target detection with improved template matching," *Infr. Phys. Technol.*, vol. 55, no. 4, pp. 380–387, Jul. 2012.
- [4] E. Lee, E. Gu, H.-J. Yoo, and K.-H. Park, "Moving dim-target tracking algorithm using template matching," in *Proc. Int. Conf. Technol. Adv. Electr., Electron. Comput. Eng. (TAECE)*, May 2013, pp. 294–297.
- [5] F. Wang, E. Liu, J. Yang, S. Yu, and Y. Zhou, "Target tracking in infrared imagery using a novel particle filter," *Chin. Opt. Lett.*, vol. 7, no. 7, pp. 576–579, 2009.
- [6] C. Le, L. Zhenghua, and W. Sentang, "Tracking of infrared radiation dim target based on mean-shift and particle filter," in *Proc. IEEE Chin. Guid., Navigat. Control Conf. (CGNCC)*, Aug. 2014, pp. 671–675.
- [7] K. Wei, Y. Q. Zhao, and Q. Pan, "IR target tracking based on mean shift and particle filter," *J. Optoelectron. Laser*, vol. 19, no. 2, pp. 213–217, 2008.
- [8] X. P. Yin, L. L. Xia, M. C. He, and W. Cheng, "An improved particle filtering algorithm based on mean shift algorithm for infrared target tracking," *Adv. Mater. Res.*, vols. 1079–1080, pp. 650–653, Dec. 2014.
- [9] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Proc. ECCV*, 2014, pp. 127–141.
- [10] Z. Li, H. Fu, J. Li, F. Ge, W. Shao, and G. Jin, "Dim moving target tracking algorithm based on particle sparse representation," *High Power Laser Part. Beams*, vol. 28, no. 2, 2016, Art. no. 021001.
- [11] Z. Wang, Q. Hou, and L. Hao, "Improved infrared target-tracking algorithm based on mean shift," *Appl. Opt.*, vol. 51, no. 21, p. 5051, Jul. 2012.
- [12] I. Leichter, "Mean shift trackers with cross-bin metrics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 695–706, Apr. 2012.
- [13] Y.-J. He, M. Li, J. Zhang, and J.-P. Yao, "Infrared target tracking via weighted correlation filter," *Infr. Phys. Technol.*, vol. 73, pp. 103–114, Nov. 2015.
- [14] T. Petrič and L. Žlajpah, "Smooth continuous transition between tasks on a kinematic control level: Obstacle avoidance as a control problem," *Robot. Auton. Syst.*, vol. 61, no. 9, pp. 948–959, 2013.
- [15] Q. Deng, H. Lu, H. Tao, M. Hu, and F. Zhao, "Multi-scale convolutional neural networks for space infrared point objects discrimination," *IEEE Access*, vol. 7, pp. 28113–28123, 2019.
- [16] Z. Fan, D. Bi, L. Xiong, S. Ma, L. He, and W. Ding, "Dim infrared image enhancement based on convolutional neural network," *Neurocomputing*, vol. 272, pp. 396–404, Jan. 2018.
- [17] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550, doi: 10.1109/CVPR.2010.5539960.
- [18] J. F. Henriques, R. Caseiro, P. Martins, J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Computer Vision—ECCV 2012 (Lecture Notes in Computer Science)*, vol. 7575, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Germany: Springer, 2012, doi: 10.1007/978-3-642-33765-9_50.
- [19] 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, doi: 10.1109/TPAMI.2014.2345390.
- [20] A. Lukei, T. Vojir, L. C. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6309–6318.
- [21] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "Fully-convolutional Siamese networks for object tracking," 2016, *arXiv:1606.09549*.
- [22] 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.*, Jun. 2018, pp. 8971–8980.
- [23] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1328–1338.
- [24] Z. Cao, Z. Huang, L. Pan, S. Zhang, Z. Liu, and C. Fu, "TCTrack: Temporal contexts for aerial tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14798–14808.
- [25] B. T. Ye, H. Chang, B. P. Ma, and S. G. Shan, "Joint feature learning and relation modeling for tracking: A one-stream framework," in *Computer Vision—ECCV 2022 (Lecture Notes in Computer Science)*, vol. 13682, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds. Cham, Switzerland: Springer, 2022, pp. 341–357.
- [26] N. Jiang, K. Wang, X. Peng, X. Yu, Q. Wang, J. Xing, G. Li, G. Guo, Q. Ye, J. Jiao, J. Zhao, and Z. Han, "Anti-UAV: A large-scale benchmark for vision-based UAV tracking," *IEEE Trans. Multimedia*, vol. 25, pp. 486–500, 2023.
- [27] X. Yuan, G. Cheng, G. Li, W. Dai, W. X. Yin, Y. C. Feng, X. W. Yao, Z. L. Huang, X. Sun, and J. W. Han, "Progress in small object detection for remote sensing images," *J. Image Graph.*, vol. 28, no. 6, pp. 1662–1684, 2023.



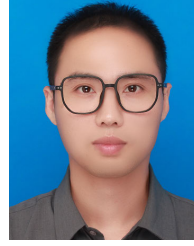
JIANHUA SHAN received the doctorate degree from the University of Science and Technology of China, in 2007. He is currently a Professor with the School of Mechanical Engineering, Anhui University of Technology, Anhui, China. He has published more than 20 papers, the monograph “Python Implementation of Convolutional Neural Networks,” and authorized five invention patents. His research interests include machine vision, robotics, and artificial intelligence.



HENG LIU was a Visiting Scholar with Northumbria University and the University of East Anglia, from 2016 to 2017. He is currently a Professor with the School of Computer Science and Technology, Anhui University of Technology, Anhui, China. He has published more than 90 SCUEI papers as the first author or corresponding author and more than ten authorized invention patents. His research interests include visual perception and deep learning.



YU YANG received the B.S. degree from the Anhui Wenda College of Information Engineering, Anhui, China, in 2017. He is currently pursuing the M.S. degree with the Anhui University of Technology, Anhui. His research interests include artificial intelligence, robotics, and computer vision.



TAO LIU received the master’s degree in mechanical and electronics from the Anhui University of Technology, in 2012. He is currently pursuing the Ph.D. degree in mechatronic engineering and automation from Shanghai University. He is also a Lecturer with the School of Mechanical Engineering, Anhui University of Technology. His current research interests include robotics and artificial intelligence.

...