



Infrared Small Target Recognition with Improved Particle Filtering Based on Feature Fusion

Qian Feng^(✉), Dongjing Cao, Shulong Bao, and Lu Liu

Beijing Institute of Space Mechanics and Electricity, Beijing 100094, China
fengqian_cast@foxmail.com

Abstract. Aiming at the problem of tracking weak targets in different scenarios, an improved particle tracking method is proposed. This paper firstly uses background prediction and extracts the gray and motion features of weak targets to suppress the background of the image, then uses the krill herd algorithm to update the particle weights, and finally uses three different types of scenes to verify the tracking effect of the algorithm. It is experimentally demonstrated that compared with the traditional algorithm, this algorithm enhances the tracking ability of small targets in different infrared scenes.

Keywords: Infrared small target · Feature fusion · Krill herd optimization · Particle filter

1 Introduction

Target recognition is one of the important research topics in computer vision, image processing and machine learning. In recent years, with the rapid development of infrared imaging technology in the field of remote sensing, infrared dim and small moving target detection has also become a research hotspot in image processing technology [1–5].

It is not easy to obtain shapes, textures and other features of the dim and small target. As a result, methods like moment invariant based on the geometric feature of the target, template matching based on the texture feature, mean shift based on the grayscale, probability and density of the target and other traditional extended target tracking methods are hard to guarantee the detection and tracking performance of weak and small targets [6]. Only the information of the target on the sequence image can be used to track and identify while detecting through the dynamic characteristics of the inter-frame correlation. Particle filtering has developed into a critical research direction of small target tracking algorithms for the better performance in the non-linear and non-Gaussian scenes [7, 8]. However, traditional particle filtering is prone to tracking missing under a single feature of the target under strong noise interference. Therefore, extracting multiple features of the target and integrating the particle filter helps to improve the stability of detection. For the detection of dim and small targets in the complex background of infrared sequence images, this paper first uses the multi-frame difference method to suppress the background and uses the grayscale and dynamic

feature of small target to delete the false points. At the same time, considering the large amount of calculation and the problem of sample depletion in resampling of particle filter, the strategy of krill herd optimization algorithm to update particle weights is added to track the dim and small targets in infrared sequence images.

2 Algorithm Framework

In this paper, the algorithm is divided into four modules: preprocessing, gray feature extraction, motion feature extraction and improved particle filter algorithm optimized by krill herd. The framework of the entire algorithm is as follows (Fig. 1):

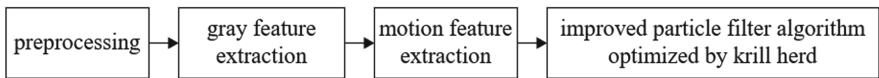


Fig. 1. Algorithm framework

The detail of the proposed algorithm is given in Sects. 2.1–2.4 below.

2.1 Preprocessing

It is no surprise that background suppression is an essential step in preprocessing to improve the detection accuracy of infrared dim targets and accurately separate the background, noise and target points. Common backgrounds can be divided into stationary backgrounds and non-stationary backgrounds. The pixels in the stationary background have strong spatial correlation. The false points are mainly composed of the noise of the detector, which will cause small changes in the background. They can be suppressed by spatial filtering. Pixels in a non-stationary background have a strong time-domain correlation so that the frame difference can be used before threshold segmentation to highlight the target and suppress the background.

Infrared images containing small targets are consist of targets, noise and background. The specific model is as follows:

$$f(x, y) = f_T(x, y) + f_B(x, y) + n(x, y) \quad (1)$$

Where $f(x, y)$ is the infrared image, $f_T(x, y)$, $f_B(x, y)$, $n(x, y)$ represent the target, background and noise [9].

Because of the long distance, the target usually appears like a dot, and the background is a large area of gentle variation with strong pixel correlation. The infrared radiation intensity of the target is generally higher than that of the surrounding background. Therefore, small targets and noises appear as isolated bright spots in infrared images, and the gray value is much larger than the pixels in their neighborhood.

Considering the non-stationary background of common infrared remote sensing images and the long distance of imaging, this article focuses on small targets moving at a constant speed under non-stationary backgrounds. This paper adopts the first few

frames as the prediction background, and then segments the images to obtain suspicious targets.

Since the target occupies very few pixels in a single-frame image, the image of the previous k frames can be taken and averaged as the predicted background image. If the original image is $f(x, y, t_n)$ and the predicted background image is $f'(x, y, t_n)$, then the predicted background image is:

$$f'(x, y, t_n) = \frac{1}{k} \sum_{n=1}^k f(x, y, t_n) \quad (2)$$

The results show that k taking 10 can better balance the relationship between the amount of calculation and robustness.

The difference processing is made between the original image $f(x, y, t_n)$ and the predicted background image $f'(x, y, t_n)$ to obtain the sequence difference image $g(x, y, t_n)$ after the background is removed:

$$g(x, y, t_n) = |f(x, y, t_n) - f'(x, y, t_n)| \quad (3)$$

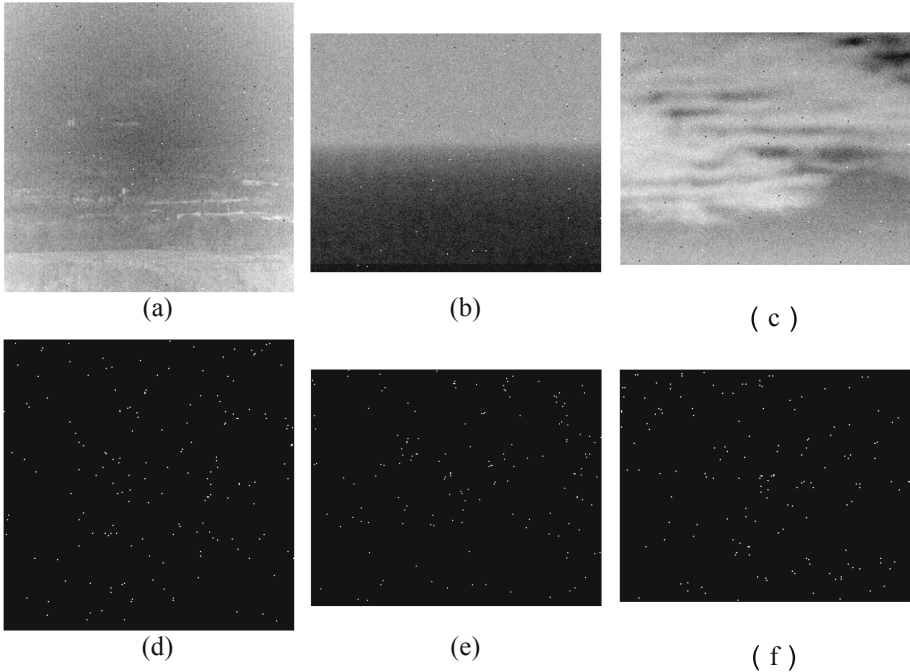


Fig. 2. Preprocessing results.

The results of the preprocessing experiment are as follows:

Among them, (a), (b) and (c) in Fig. 2 are the three kinds of backgrounds in the video sequence, and (d), (e) and (f) are the background suppression results calculated by Eqs. (2) and (3). As can be seen from the figure, the algorithm can suppress the background well and highlight the target point using the gray and motion features of the targets.

2.2 Gray Feature Extraction

After predicting the background image, the obtained image can be subtracted with value T to obtain a binary image $F(x, y, t_n)$, in which the non-zero gray value are the set of suspected target points:

$$F(x, y, t_n) = \begin{cases} 0, & g(x, y, t_n) < T \\ g(x, y, t_n), & g(x, y, t_n) \geq T \end{cases} \quad (4)$$

Where $F(x, y, t_n)$ is the binary map of the suspect target point set, $g(x, y, t_n)$ is the sequence difference image after background removal, and T is the segmentation threshold selected according to experience.

According to the 4 neighborhoods traversal image $F(x, y, t_n)$, we can find all the suspected target points with connected 4 domains. Cause the infrared weak small target spots are usually slightly larger than the noise points, we will arrange the connected domain according to the size and select top k largest connected domains, which will be used as candidate target points, and the others are eliminated as noise. At this point, we have obtained the grayscale features of the suspected target points with position information.

2.3 Motion Feature Extraction

According to the positional correlation of the movement of the small targets between frames, the distance of the candidate target points between frames can be used as the basis for discrimination. In this paper, the shortest Euclidean distance is used as the trajectory of the target points and the initial target position is determined from the candidate target points.

$$D_k^n = \sqrt{|c(x_k^n) - c(x_k^{n-1})|^2 + |c(y_k^n) - c(y_k^{n-1})|^2} \quad (5)$$

Where D_k^n represents the distance of the k_{th} candidate target point between the n_{th} frame and the $n-1_{th}$ frame, and $c(x_k^n)$ represents the centroid x coordinate of the k_{th} candidate target point in the n_{th} frame, $c(x_k^{n-1})$ represents the centroid x coordinate of the $k-1_{th}$ candidate target point in the $n-1_{th}$ frame, $c(y_k^n)$ represents the centroid y coordinate of the k_{th} candidate target point in frame $n-1$, $c(y_k^{n-1})$ represents the x coordinate of the centroid of the k_{th} candidate target point in the $n-1_{th}$ frame. According to the experimental results of this paper, k takes 4.

Infrared dim targets move slowly on the imaging surface, and sometimes the distance between frames is less than one pixel. As a sequence of the random and messy appearance and no correlation between frames of noise, it is helpful for us to use motion distance of candidate target points between two adjacent frames as a criterion to effectively eliminate noise points.

2.4 Improved Particle Filter Algorithm Optimized by Krill Herd

Assuming that the target is moving at a constant speed, the state variable of the target is defined as $S = \{x, y, v_x, v_y\}$, where (x, y) is the position of the center of the target, (v_x, v_y) is the horizontal and vertical speed of the target. We define the state transition matrix F and random noise W_k as:

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, W_k = \begin{bmatrix} S_p * rand \\ S_p * rand \\ S_v * rand \\ S_v * rand \end{bmatrix} \tag{6}$$

Where S_p represents the status constant variable in the horizontal direction, and S_v represents the status constant variable in the vertical direction. According to the experiment, S_p takes 25 and S_v takes 5.

The steps of the improved particle filter algorithm optimized by krill herd are as follows (Fig. 3):

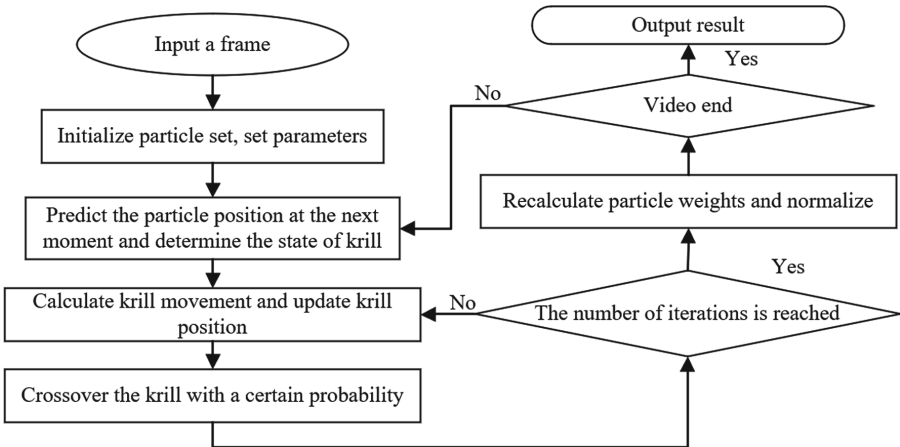


Fig. 3. The steps of the improved particle filter algorithm optimized by krill herd.

Step 1: Initialization

Establish the target motion model:

$$X_k = FX_{k-1} + W_k \quad (7)$$

Where F is the state transition matrix, W_k is the random noise.

Sampling independent and identically distributed particles $\{x_k^i, \frac{1}{N}\}_{i=1}^N$ according to

$$x_k^i \sim q(x_k^i | x_{k-1}^i, z_k) \quad (8)$$

Set iterations I to 1, set krill group size N , maximum induction speed N^{max} , maximum foraging speed V_f , maximum random diffusion speed D^{max} , time interval Δt , maximum number of iterations I_{max} , the inertia weight of induced motion w_n and inertial weight w_f of foraging behavior.

Step 2: Prediction

Calculate the weight of the new particle set $\{x_k^i, w_k^i\}_{i=1}^N$ according to the current observed value z_k :

$$w_k^i = w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{p(x_k^i | x_{k-1}^i, z_k)} \quad (9)$$

Use the state function of Eq. (7) to calculate the state value x_k^i of the particle at the next moment, then use Eq. (9) to calculate the weight of each particle. The particle state value x_k^i is used as the position of krill X_i .

Step 3: Update

$$N_i^{new} = N^{max} \alpha_i + w_n N_i^{old} \quad (10)$$

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target} \quad (11)$$

Where N indicates the induced motion speed, max indicates the maximum speed, α_i indicates the individual induced direction, α_i^{local} indicates the neighboring individual induced direction, α_i^{target} indicates the optimal individual induced direction, $w_n \in [0, 1]$ indicates the inertial weight of the original motion.

$$F_i^{new} = V_f \beta_i + w_f F_i^{old} \quad (12)$$

$$\beta_i = \beta_i^{food} + \beta_i^{best} \quad (13)$$

Where F_i^{new} is the speed vector of the foraging behavior of the i_{th} krill, V_f is the speed of foraging behavior, β_i^{food} is the effect of food on the i_{th} krill, and β_i^{best} is the optimal fitness of the i_{th} krill, $w_f \in [0, 1]$ is the inertia weight of foraging behavior, and F_i^{old} is the speed vector of the last foraging behavior.

$$D_i^{new} = D^{max} \left(1 - \frac{l}{l_{max}} \right) \delta \quad (14)$$

Where D^{max} is the maximum speed of random diffusion, l_{max} is the maximum number of iterations, and δ is a directional vector with each variable obeying $[-1, 1]$ uniform distribution.

$$X_i^l = X_i^{l-1} + \Delta t (N_i^{new} + F_i^{new} + D_i^{new}) \quad (15)$$

Where Δt is the time interval.

Use Eqs. (10) (12) (14) to calculate the induced foraging and diffusion movement of each krill individual, and finally calculate the position change of each krill in each iteration according to Eq. (15).

Step 4: Cross operation

$$C_r = 0.2 \hat{K}_{i,best} \quad (16)$$

Where C_r is the crossover probability [10–12], $\hat{K}_{i,best}$ represents the best fitness of the i_{th} krill.

Calculate the crossover probability according to Eq. (16) and perform crossover operation on individual krill groups.

Step 5: Determine the Termination Conditions

Whether the maximum number of iterations l_{max} is reached, otherwise step 3.

Step 6: Weight Normalization

$$w_k^i = \frac{w_k^i}{\sum_{i=1}^{N_s} w_k^i} \quad (17)$$

$$\hat{x}_k = \sum_{i=1}^N w_k^i x_k^i \quad (18)$$

Recalculate the weight of each particle, update the particle weight according to Eq. (9), normalize the particles according to Eq. (17), and finally output the estimated state value of each particle according to Eq. (18).

Step 7: Iteration

Repeat steps 2 to 6 until the prediction is completed.

3 Simulation Results and Analysis

Three sets of video sequences with different backgrounds were selected for the experiment to confirm the performance of the method in this paper. The first test video came from the detection and tracking data set of small and weak aircrafts with infrared images under the ground/air background of Hui Bingwei et al. [13], the simulation experiment was performed on the hardware platform of AMD Ryzen 5 4600H CPU 3.00 GHz and memory 16 GB, and the development environment is MatlabR2017b (Table 1).

Table 1. Properties of experimental scenes.

Video sequence	Backgrounds	Frame length	Image size	Target size
1	Ground	256	256 × 256	2 × 2
2	Sea and sky	280	280 × 228	2 × 2
3	Clouds	245	250 × 200	2 × 2

The experiment compares the classical particle filtering algorithm and the improved method in this paper from the tracking time per frame and error under three common backgrounds. Define the tracking error as the Euclidean distance between the target centroid in the tracking algorithm and the calibrated target centroid:

$$error = \sqrt{(x - x^*)^2 + (y - y^*)^2} \quad (19)$$

The number of particles in the algorithm is 300.

Video sequence 1 is the background with ground. (a) and (b) in Fig. 4 are the tracking effects of the basic particle filtering algorithm and the improved particle filtering algorithm on the video, respectively, from the 30th, 120th, 180th, and 220th frames of the video. Figure 4(c) is the comparison of the single frame operation time of the two algorithms, and Fig. 4(d) is the comparison of the tracking errors of the two algorithms.

It can be seen from Fig. 4 (a1) (b1) that the basic particle filtering algorithm is disturbed and the centroid shifted during the first few frames of the video, and the particle filter algorithm optimized by krill herd can track the target more stably and has better robustness. It can be seen from Fig. 4(c) that adding particle optimization does not greatly extend the calculation time. The calculation time of the two algorithms is similar, and the basic particle filtering algorithm is slightly faster. It can be seen from Fig. 4(d) that the error of the optimized particle filter algorithm is smaller than that of the basic particle filter algorithm in the middle of the video.

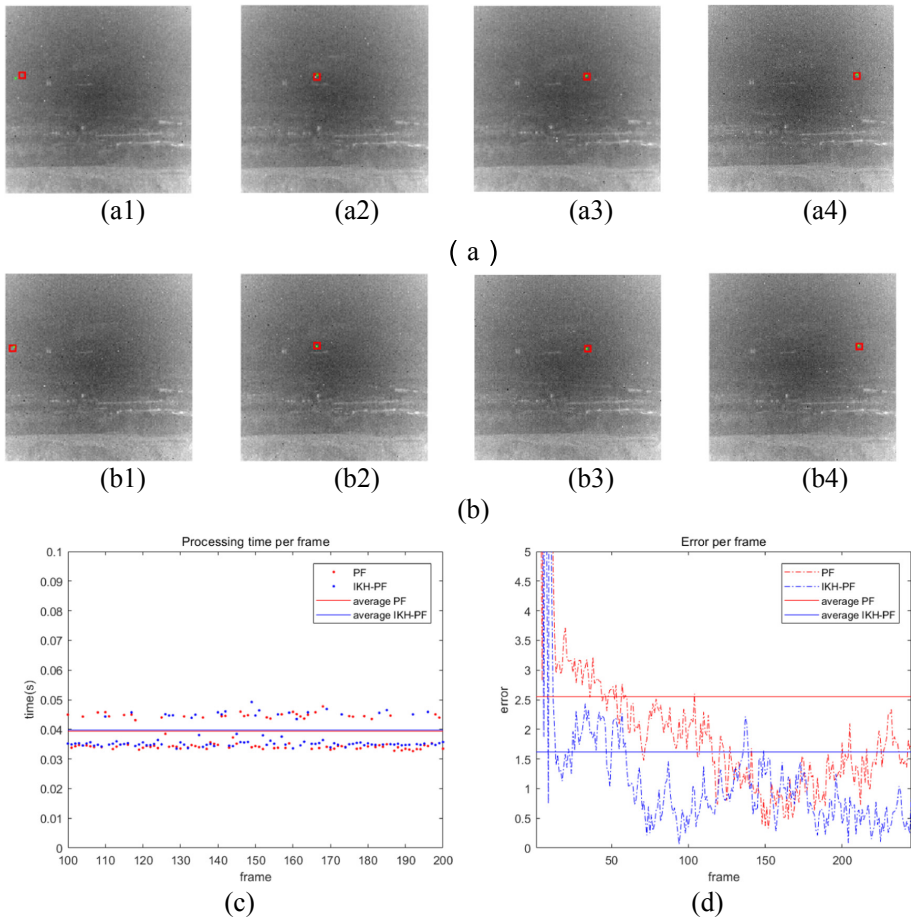


Fig. 4. Tracking results under background with ground in Experiment 1.

Video sequence 2 is the sea-sky background. (a) and (b) in Fig. 5 are the tracking results of the basic particle filtering algorithm and the improved particle filtering algorithm on the video, respectively from the 30th, 120th, 180th, and 220th frames of the video. Figure 5(c) is the comparison of the single frame operation time of the two algorithms, and Fig. 5(d) is the comparison of the tracking errors of the two algorithms.

As can be seen from Fig. 5(a1) (b1), the gray level of the background is similar in the video, so the target characteristics are more obvious, and both algorithms have good performance. Figure 5(c) displays that the calculation time of the two algorithms is similar, and the basic particle filtering algorithm is slightly faster. What's more, Fig. 4 (d) demonstrates that the error of the two algorithms are about the same in the front and the end part of the video, and optimized particle filter algorithm is smaller than that of the basic particle filter algorithm in the middle of the video.

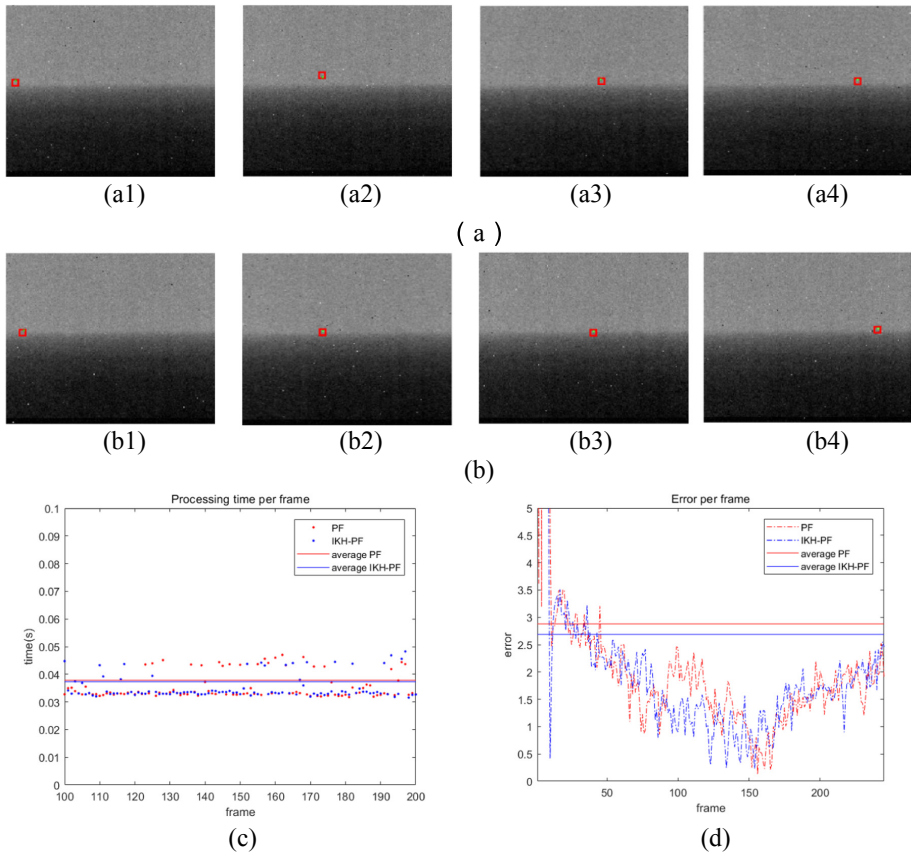


Fig. 5. Tracking results under background with sea and sky in Experiment 2.

Video sequence 3 is a background with clouds. (a) and (b) in Fig. 6 are the tracking results of basic particle filtering algorithm and improved particle filtering algorithm on the video, respectively from the 40th, 80th, 160th, and 200th frames of the video. Figure 6(c) is the comparison of the single frame operation time of the two algorithms, and Fig. 6(d) is the comparison of the tracking errors of the two algorithms.

It can be seen from Fig. 6(a1) (b1) that in backgrounds where the grayscale is close to the target, the elementary particle filtering algorithm suffers some interference and there are cases of loss. The particle filter algorithm optimized by the krill herd can track the target more stable, and the robustness is better. In addition, Fig. 6(c) illustrates that the calculation time of the two algorithms is similar, and the basic particle filter algorithm is slightly faster. Furthermore, the particle filter algorithm with krill herd optimization is highly robust under the background with large interference in Fig. 6(d).

Considering that particle filtering is a probabilistic algorithm, the results of each experiment will be different, so we have run each experiment twelve times and the average of the results is taken. The calculation data of the two algorithms in three backgrounds is shown in the following Tables 2 and 3:

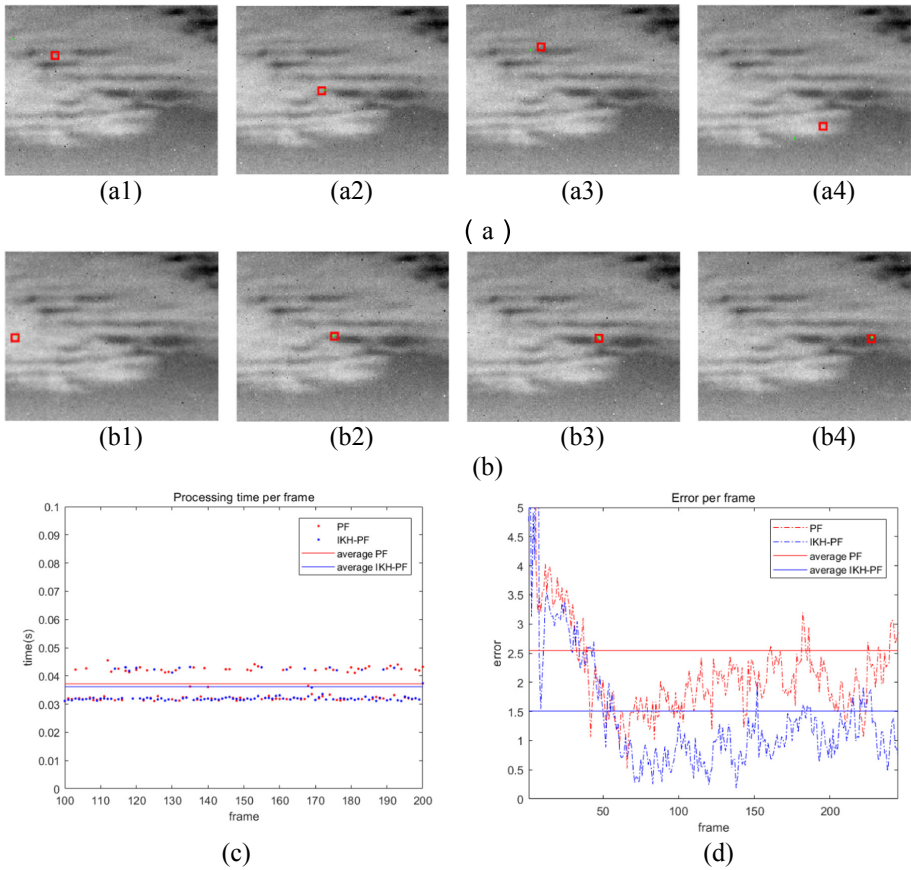


Fig. 6. Tracking results under background with clouds in Experiment 3.

Table 2. Processing time of algorithm

Video sequence	Algorithm	Average processing time(s)	Total processing time(s)	Average error
1	PF	0.0374	9.8401	2.5438
	IKH-PF	0.0398	10.1765	1.5007
2	PF	0.0374	9.0376	3.8052
	IKH-PF	0.0378	9.4603	2.6754
3	PF	0.0362	8.7630	2.5502
	IKH-PF	0.0377	9.6131	1.6163

Table 3. Rate of algorithm

Video sequence	Algorithm	Missing detection rate (%)	False alarm rate (%)	Accuracy rate (%)
1	PF	2.33	3.67	89.33
	IKH-PF	1.67	1.33	92.67
2	PF	4.00	3.33	93.33
	IKH-PF	2.67	2.00	95.00
3	PF	2.67	4.67	87.67
	IKH-PF	1.33	2.00	90.33

4 Conclusions

The above experiments demonstrate that the preprocessing algorithm using grayscale and motion features proposed in this paper can effectively suppress background in complex environments and highlight the small dim target. Then the algorithm uses the particle filter algorithm based on krill herd optimization to track the target. As we can see, although part of the operation speed is sacrificed, the robustness and accuracy of tracking are effectively improved. As a result, the detection of infrared dim targets can be accurately achieved.

References

1. Deng, H., Sun, X., Liu, M., et al.: Entropy-based window selection for detecting dim and small infrared targets. *Pattern Recogn.* **61**, 66–77 (2017)
2. Song, M., Wang, S., Lü, T., et al.: A method for infrared dim small target detection in complex scenes of sky and ground. *Infrared Technol.* **40**(10), 996–1001 (2018). (in Chinese)
3. Gao, C., Meng, D., Yang, Y., et al.: Infrared patch-image model for small target detection in a single image. *IEEE Trans. Image Process.* **22**(12), 4996–5009 (2013)
4. Gao, C., Zhang, T., Li, Q.: Small infrared target detection using sparse ring representation. *IEEE Aerosp. Electron. Syst. Mag.* **27**(3), 21–30 (2012)
5. Shi, Y., Wei, Y., Yao, H., et al.: High-boost-based multiscale local contrast measure for infrared small target detection. *IEEE Geosci. Remote Sens. Lett.* **15**(1), 33–37 (2018)
6. Zhang, S., Huang, X., Wang, M.: Algorithm of infrared background suppression and small target detection. *J. Image Graph.* **21**(8), 1039–1047 (2016). (in Chinese)
7. Han, H., Ding, Y.S., Hao, K.R., et al.: Particle filter for state estimation of jump Markov nonlinear system with application to multi-targets tracking. *Int. J. Syst. Sci.* **44**(7), 1333–1343 (2013)
8. Han, H., Ding, Y.S., Hao, K.R., et al.: An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking. *Comput. Math Appl.* **62**(7), 2685–2695 (2011)
9. Hao, Chen, Caiwen, Ma., Yuecheng, Chen, et al.: Multi dim feature detection and tracking algorithm based on multi feature fusion in complex background. *Acta Photonica Sinica* **38**(9), 2444–2448 (2009). (in Chinese)

10. Zhang, Z., Jiang, Q., Xu, L.: Airborne single observer passive location based on krill herd immune particle filter. *Fire Control Command Control* **40**(4), 92 (2015). (in Chinese). <https://doi.org/10.3969/j.issn.1002-0640.2015.04.023>
11. Guo, W., Gao, Y., Liu, P.: An improved krill herd algorithm with adaptive inertia weight. *J. Taiyuan Univ. Technol.* **47**(5), 651 (2016). (in Chinese). <https://doi.org/10.16355/j.cnki.issn1007-9432tyut,2016.05.017>
12. Liu, Z., Lu, H., Ren, J.: Co-evolutionary gravitational krill herd algorithm. *Adv. Eng. Sci.* **50**(6), 217 (2018). <https://doi.org/10.15961/j.jsuese.201800012>
13. Hui, B.W., Song, Z.Y., Fan, H.Q., et al.: Weak and small aircraft target detection and tracking data set in infrared sequence images. [DB/OL]. *Sci. Data Bank* (2019). <https://doi.org/10.11922/csdata.2019.0074.zh>. Accessed 28 Oct 2019