Vehicle-Mounted Infrared Pedestrian Tracking Based on Scale Adaptive Kernel Correlation Filter

Yuanbin Wang, Yujie Li and Qian Han

***Abstract*—Pedestrian tracking for vehicle-mounted infrared images is essential in the vehicle-assisted driving system. In general, dealing with the change of the target scale in vehicle-mounted pedestrian tracking is tough. To solve this issue, this study proposes a pedestrian tracking algorithm based on Scale Adaptive Kernel Correlation Filter (SAKCF). The median filtering is applied to suppress the background information and improve the ratio of signal-to-noise. Histogram of Intensity (HOI) feature and Histogram of Oriented Gradient (HOG) feature of the image are extracted and input into SVM. In order to make the tracking adaptive to the scale change of the target, the method of SAKCF is applied. The algorithm is divided into two stages: position detection and scale detection. During position detection, a cyclic matrix around the target position is constructed for dense sampling, with the HOG feature extracted to train the position filter by the output response of Gaussian distribution. During scale detection, multi-scale cyclic shift samples are obtained in the central area of the position detection result and zoomed to the fixed scale of the target by bilinear interpolation, with the value of target regression defined by Gaussian distribution. Experiments indicate that the proposed method can acquire stable tracking on pedestrians for vehicle-mounted infrared images and meet real-time requirements with solid robustness.**

***Index Terms*—pedestrian tracking, vehicle-mounted infrared images, pedestrian detection, scale adaptive kernel correlation filter.**

I. Introduction

IN computer vision, pedestrian detection and tracking is a research hotspot, widely applied in automotive driver assistance system, intelligent video surveillance, and other fields. Infrared image can penetrate smoke and fog and is not subject to intense light and flash. It can achieve all-weather remote observation. With the development of infrared technology, vehicle-mounted infrared pedestrian detection and tracking has received more and more attention [1].

According to the working principle, the target tracking algorithm is divided into generative and discriminant models. For the former, there are mainly particle filter tracking, Kalman filter tracking, and mean-shift tracking. For the latter, there are Kernel Correlation Filter Tracking (KCF), Tracking Learning Detection (TLD), and Circulant Structure of Tracking-by-detection with Kernels (CSK). The discriminant model is the mainstream direction in the field of tracking currently. In 2010, Bolme introduced the Minimum Output Sum of Squared Error (MOSSE) filtering algorithm, which constructed an adaptive correlation filter to simulate the appearance of the target, achieved target detection and tracking by calculating the minimum square error between the expected correlation output and the actual correlation output [2]. In 2012, Henriques proposed CSK based on MOSSE, which solved insufficient training samples in target tracking and significantly improved the tracker's performance [3]. In 2015, Henriques proposed the KCF algorithm based on the CSK tracker for the reason of further improving robustness of the algorithm [4]. It expanded the feature in the CSK tracker from a single-channel gray feature to a multi-channel HOG feature, which significantly improved the tracking accuracy. Danelljan adopted Color Name in the framework of correlation filtering and improved the tracking speed by singular value decomposition [5]. Qi improved correlation filtering combined with CNN(Convolutional Neural Networks) and showed superior performance in the aspect of accuracy and speed [6]. Peng Liu proposed a multi-layer background adaptive correlation filtering observation model based on KCF. He used Gaussian windows to suppress the background at each layer of the pyramid. The multi-level context pyramid was employed in learning and tracking to fully use the relation between the target and the background at different levels and improved the tracking performance [7]. Wang Lu extracted HOG features, employed SVM and AdaBoost to detect pedestrians, and combined Kalman filter with a mean shift, which had adaptive windows for pedestrian tracking [8]. Gao Meifeng judged the target size according to the multi-scale sampling results of the target area so that the tracking frame can change with the target size adaptively [9]. In view of the scale change, Zhang Lei proposed an adaptive scale target tracking algorithm, which promoted the development of the correlation filtering algorithm [10]. Hu Qingxin fused the HOG feature, integral channels, and self-similar brightness features by using the parallel weighted feature fusion method. Then, he trained the SVM classifier with these fused features. Finally, he performed pedestrian detection by the trained SVM [11]. Zhang Shaoming proposed a new approach based on modified particle filtering, which overcame the color deficiency and enhanced the robustness [12]. Zhang Xu fused the artificial features and shallow convolution features into the pedestrian target features in the infrared image, which will achieve stable pedestrian tracking and further improve the tracking performance [13].

In the process of vehicle driving, the background is complex and changeable [14]. Pedestrian size variation,

Manuscript received June 10, 2021; revised December 26, 2021. This work was supported by the National Natural Science Foundation of China and the National Key Research and Development Program of Shaanxi Province, China(2019KW-046).

Yuanbin Wang is an associate professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 13379232752@163.com).

Yujie Li is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1678931975@qq.com).

Qian Han is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 947674382@qq.com).

high motion speed, and many other factors all affect tracking results. This paper proposed the SAKCF algorithm to implement stable pedestrian tracking for vehicle-mounted infrared video. The framework of this article is shown in Fig. 1.

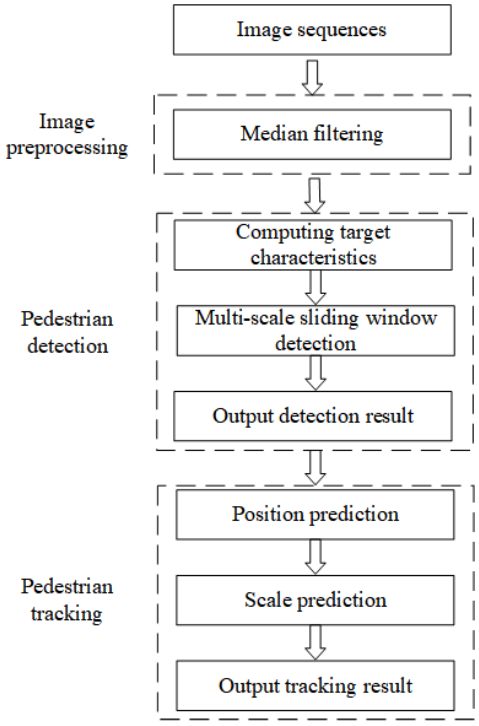


Fig. 1. Framework of the proposed algorithm

II. Filtering on Vehicle-Mounted Infrared Image

Due to the weather change, illumination, camera shaking, and other factors, the infrared image quality is not ideal. In order to enhance the target and suppress the background, Gaussian filtering, mean filtering, and median filtering were used for comparison.

The Structural Similarity Index (SSIM) and Signal to Noise Ratio (PSNR) are applied as indicators to rate the efficacy of the filtering algorithms. The calculation is shown in formulas(1),(2), and(3).

$$PSNR = 10\log_{10}(\frac{255^2}{MSE}) \tag{1}$$

$$MSE = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(G(i,j)-P(i,j))^2}{m \times n} \tag{2}$$

Where n stands for the images height, and m stands for the images width. $P(i,j)$ and $G(i,j)$ are the pixel value of the filtered image and original image at the point of (i,j) separately.

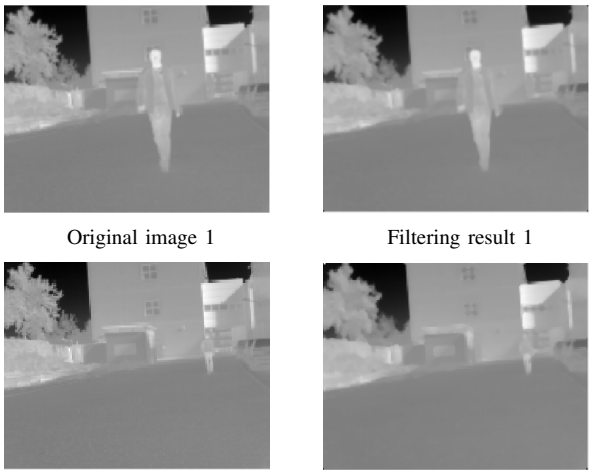
$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_x\sigma_y + C_2)}{({\mu_x}^2 + {\mu_y}^2 + C_1)({\sigma_x}^2 + {\sigma_y}^2 + C_2)} \tag{3}$$

Where μ_x and μ_y are the average gray levels. σ_x and σ_y represent the gray standard deviations.

The performance of the filtering methods is described as shown in Table I. Median filtering performs well on both indicators, and the calculating is simple. So this paper uses median filtering for denoising. The results of median filtering are shown in Fig. 2.

TABLE I
Comparison of PSNR and SSIM of Different Algorithms

Algorithm	PSNR	SSIM
Gaussian filtering	22.3940	0.8740
Mean filtering	28.0070	0.8760
Median filtering	28.9210	0.8760



Original image 1 | Filtering result 1

Original image 2 | Filtering result 2

Fig. 2. Images after median filtering

III. Pedestrian Detection

For the purpose of increasing detection accuracy, SVM is selected for vehicle-mounted infrared pedestrian detection. It is separated into two parts. The first part is to get the HOG and HOI features of the image. The second part is to use a multi-scale sliding window and SVM to detect the pedestrian and then fuse the results of each window.

A. Feature Extraction

1) HOG feature: HOG is the abbreviation of Histogram of Oriented Gradient. HOG feature represents the shape of local objects by describing the intensity and direction of local edge gradient, which is obtained by normalization, gradient calculation, local gradient direction histogram, and in-block normalization. In particular, the histogram in the gradient direction of the local area is a form of local features. As a kind of gradient map, HOG feature pictures are displayed in Fig. 3 and Fig. 4.

Fig. 3 indicates that the gradient map can keep the target's contour and remove irrelevant information well. In Fig. 4, based on fine directional sampling and standardization of local block areas, HOG features perform well in depicting pedestrian features, not sensitive to target deformation.

2) HOI feature: In order to represent the distinction between the background and the pedestrian, the Histogram of Intensity(HOI) is used to encode the luminance information and enhance the robustness of the block normalization. The HOI feature map is shown in Fig. 5.

As shown in Fig. 5, the HOI feature can fully describe the brightness information of the pedestrian with a simple calculation. However, it is not easy to distinguish the background with the same height. Therefore, detection may fail if it is used alone.

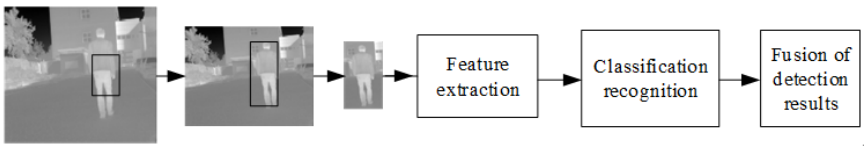


Fig. 6. Flowchart of multi-scale sliding window detection

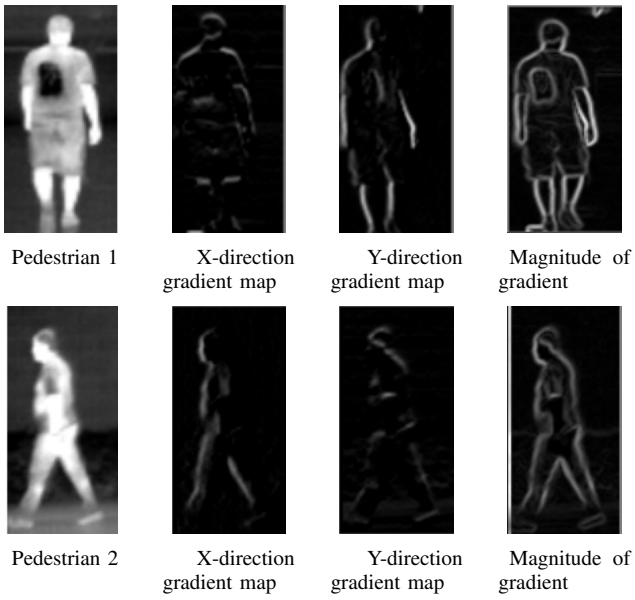


Fig. 3. Gradient map of infrared pedestrian image

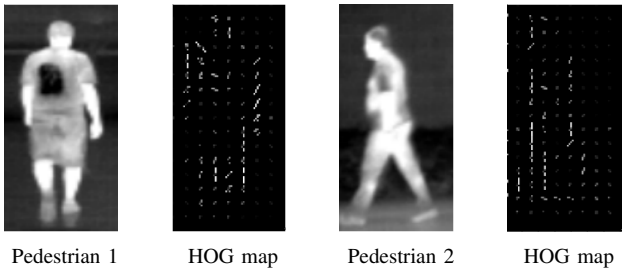


Fig. 4. HOG map of infrared pedestrian image

B. Detection Based on Multi-Scale Sliding Window

A sliding window traversing from left to right and top to bottom is employed for scanning to determine the specific position of the pedestrian. SVM is applied to judge whether it contains the pedestrian in each corresponding window. The steps are as follows. Firstly, a minimum size window is set to scan the whole image. Afterwards, the detection window is scaled, and the scaling ratio is set to 1.2. Then, the image is scanned repeatedly to extract the pedestrian features in each corresponding window. Finally, the detection results of each scale are fused. Fig. 6 shows the flowchart of the detection. For the input image, the size is $W \times H$, the size of the detection window is $W_i \times H_i$, and scale series is expressed as:

$$N = \left[\frac{\log(S_N/S_1)}{\log(\Delta_s)} + 1 \right] \tag{4}$$

Where $S_1 = 1$ represents the smallest scale. S_N is the largest scale, which is caculated as:

$$S_N = \min(\frac{W}{d_w^m}, \frac{H}{d_H^m}) \tag{5}$$

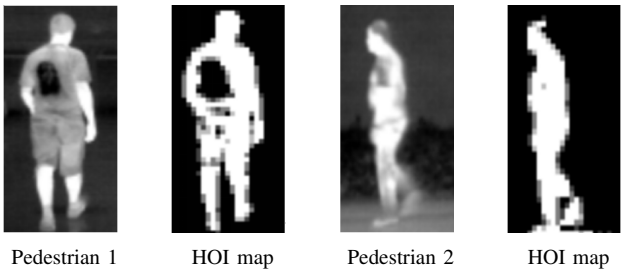


Fig. 5. HOI map of infrared pedestrian image

Where d_W^m and d_H^m are the width and height of the smallest search window separately.

C. Testing results

Vehicle-mounted infrared pedestrian detection mainly includes the training stage and detection stage. In the training stage, negative and positive training samples are input first, then HOG and HOI features are extracted and parallel fused, and the SVM model is constructed and employed to classify the fusion features. In the detection stage, different scale windows are applied to traverse with the same step size. In each corresponding window, the trained SVM model is adopted to detect pedestrians. Finally, the windows containing pedestrians are fused, and the detection results are output. The initial values of C and σ in the SVM model are 10 and 0.01, respectively, and the kernel function is RBF. The detection result is shown in Fig. 7.

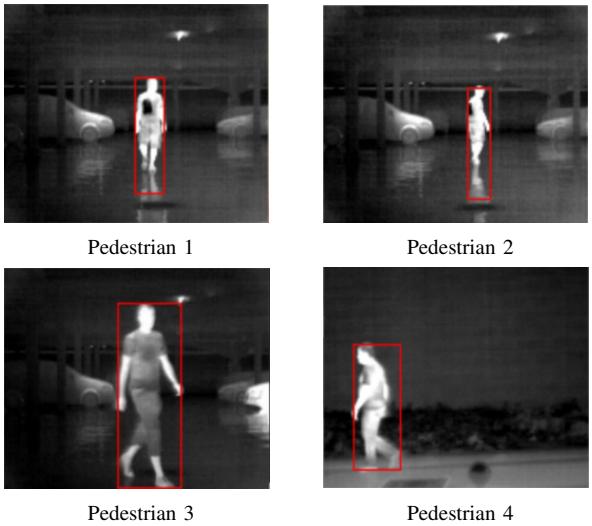


Fig. 7. Detection results of infrared pedestrian images

IV. Pedestrian Tracking

The detection result is utilized as the input to the tracking algorithm after pedestrian detection. Instead of the

KCF algorithm, this study proposes a new algorithm, SAKCF, to establish the relationship of the moving pedestrian in each frame and then achieve automatic pedestrian tracking.

A. The Tracking Algorithm of KCF

The tracking algorithm of KCF is based on correlation filtering to realize target tracking. Fig. 8 depicts the framework of the KCF tracking algorithm.

1) Dense sampling on cyclic matrix: KCF tracking algorithm employs cyclic matrix dense sampling to acquire samples and employs a cosine window to suppress marginal noise [15]. The cyclic matrix is defined as (6).

$$C(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \tag{6}$$

The base samples is in the first row of $C(\mathbf{x})$. It is used as the positive samples at training. Furthermore, the other rows are moved to the right by one element from the previous row, and the rightmost element moves to the leftmost. They are used as negative samples. This is called cyclic shift.

Given a vector $\mathbf{x} = [x_1, x_2, \cdots x_n]^T$, the permutation matrix P is introduced as:

$$P = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{7}$$

The cyclic shift of element $\mathbf{x}$ is expressed as $P\mathbf{x}$. $P^u\mathbf{x}$ denotes the cyclic shift of u elements to $\mathbf{x}$. When the element u is negative, it represents the shift in the opposite direction. Thus the cyclic shift of $\mathbf{x}$ can be expressed as:

$$\{P^u\mathbf{x} | u = 0, 1, 2, \cdots, n-1\} \tag{8}$$

Combine these cyclic shift vectors and obtain another definition form of $\mathbf{x}$:

$$X = C(\mathbf{x}) = \begin{bmatrix} (P^0\mathbf{x})^T \\ (P^1\mathbf{x})^T \\ (P^2\mathbf{x})^T \\ \vdots \\ (P^{n-1}\mathbf{x})^T \end{bmatrix} \tag{9}$$

2) Filter training: Suppose there are training samples $(x_1, y_1), (x_2, y_2), ...(x_i, y_i), ...(x_m, y_m)$. The squared error between sample $\mathbf{x}_i$ and regression target y_i is caculated and minimized [16]by the target function $f(\mathbf{z}) = \mathbf{w}^T\mathbf{x}$, that is:

$$\min_{\mathbf{w}} \sum_{i=1}^{m} (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2 \tag{10}$$

Where y_i is the label of the candidate sample, the regularization parameter λ is used to avoid overfitting in calculation process. $\mathbf{w}$ represents the correlation filter and $\|\|$ is the modulus value [17]. It can be written in the form of a matrix:

$$\min_{\mathbf{w}} \sum_{i=1}^{m} (X\mathbf{w} - \mathbf{y})^2 + \lambda \|\mathbf{w}\|^2 \tag{11}$$

Where $X = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \cdots, \mathbf{x}_n]^T$, $\mathbf{y}$ is a column vector, and each element corresponds to a label value y_i. The least square method can be used to obtain $\mathbf{w}$:

$$\mathbf{w} = (X^T X + \lambda I)^{-1} X^T \mathbf{y} \tag{12}$$

When $\mathbf{w}$ is expressed in the form of complex field, it is expressed as:

$$\mathbf{w} = (X^H X + \lambda I)^{-1} X^H \mathbf{y} \tag{13}$$

Where I is the identity matrix, X^{H} is the Hermitian transpose of X. For real numbers, equation (12) is equivalent to (13). It can be seen from equation (13) that the calculation is huge, and the time complexity is $\mathrm{O}(n^3)$. Therefore, the cyclic matrix is introduced to simplify the calculation. According to the cyclic convolution theory, the Discrete Fourier Transform (DFT) may diagonalize any cyclic matrix as seen in formula (14).

$$X = F\, diag(\hat{\mathbf{x}})\, F^H \tag{14}$$

Where $\hat{\mathbf{x}}$ is the Fourier transform of $\mathbf{x}$, $diag()$ means diagonalization, F is a constant matrix, H means conjugate transpose. Substituting (14) into equation (13), we obtain:

$$\mathbf{w} = (F diag(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}}) F^H + \lambda I)^{-1} X^H \mathbf{y} \tag{15}$$

Where $\odot$ refers to the dot product operation between the elements in the matrix, $*$ refers to the complex conjugate, $\hat{\mathbf{w}}$ is simplified as:

$$\hat{\mathbf{w}} = \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \tag{16}$$

Equation (16) is a linear filter, which is obtained by ridge regression training [18]. $\hat{\mathbf{w}}$ is the solution of $\mathbf{w}$ in frequency domain. Only the dot product operation is involved in the solution process, and the time complexity is $\mathrm{O}(n)$. $\mathbf{w}$ can be obtained by inverse Fourier transform to minimize the complexity in calculation. For linear inseparable problems, the concept of high-dimensional kernel function is introduced, and the expression is:

$$\mathbf{w} = \sum_i \alpha_i \phi(\mathbf{x}_i) \tag{17}$$

At this point, the objective function is:

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} = \sum\nolimits_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_j) \tag{18}$$

The kernel matrix $\mathbf{K}$ can be obtained by calculation of $k(\mathbf{x}_i, \mathbf{x}_j)$. It is expressed as:

$$\mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j) \tag{19}$$

It can be testified that k is a cyclic matrix. After the kernel function is added, the coefficient matrix $\hat{\alpha}$ is solved in the dual space, and expressed as:

$$\hat{\alpha} = \frac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda} \tag{20}$$

Where $\hat{\mathbf{k}}^{xx}$ is the kernel autocorrelation of vector $\mathbf{x}$ in frequency domain. Equation (20) is a nonlinear filter trained by kernelized ridge regression. For any two vectors $\mathbf{x}$ and $\mathbf{x}'$, the kernel correlation between them is:

$$k_i^{\mathbf{xx}'} = k(\mathbf{x}', p^{i-1}\mathbf{x}) \tag{21}$$

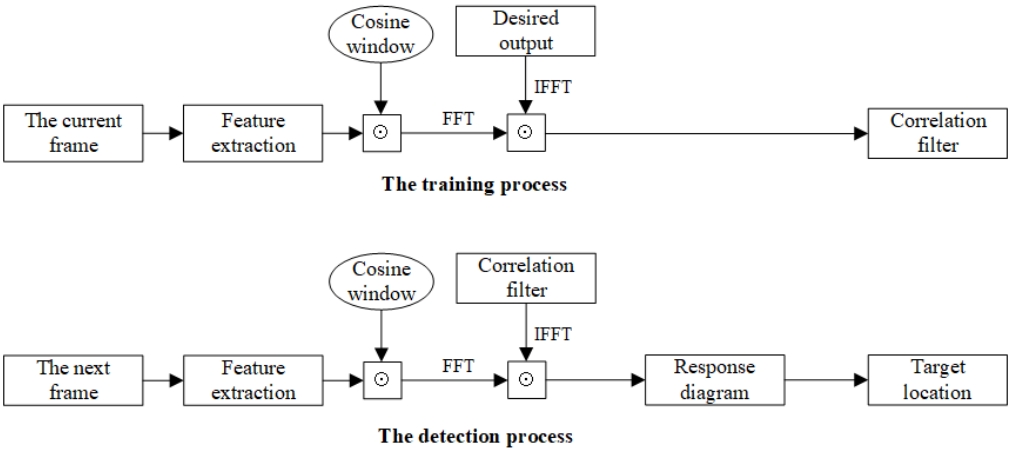


Fig. 8. Framework of KCF tracking algorithm

3) Rapid detection: After correlation filter training, the area with the same size as the previous frame is searched in the present frame and then set as the base sample for the detection sample. So that cyclic offset sampling is performed to construct the detection sample set Z. Then, the trained filter is applied to calculate the detection sample $\mathbf{z}$ to obtain filter response.

$$f(\mathbf{z}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{z}_i, \mathbf{x}_i) \tag{22}$$

$$f(\mathbf{z}) = (K^{\mathbf{z}})^T \alpha \tag{23}$$

Where $f(\mathbf{z})$ is the cyclically shifted candidate samples response value. $\mathbf{K^z}$ is the kernel matrix consisting of candidate samples and training samples. $\mathbf{K^z}$ is diagonalized in Fourier domain and substituted into Equation (23) :

$$\hat{f}(\mathbf{z}) = \hat{\mathbf{k}}^{\mathbf{xz}} \odot \hat{\alpha} \tag{24}$$

Where $\hat{\mathbf{k}}^{\mathbf{xz}}$ is the kernel cross correlation between the training sample $\mathbf{x}$ and the detection sample $\mathbf{z}$ in Fourier domain. The position of $\hat{f}(\mathbf{z})$, which has the largest response value, is the position of the target in the current frame [19].

4) Template update: The update strategy is as follows:

The update strategy of the target model is expressed as equation (25).

$$\mathbf{x}_t = (1-\beta)\mathbf{x}_{t-1} + \beta\mathbf{x}' \tag{25}$$

Where $\mathbf{x}'$ is the currently detected area, $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ denote the target template of the current frame and the previous frame, respectively. β is the learning factor.

The update strategy of the filter template is as formula(26).

$$\alpha_t = (1-\beta)\alpha_{t-1} + \beta\alpha_{x'} \tag{26}$$

Where $\alpha_{\mathbf{x}'}$ is the coefficient of $\mathbf{x}'$, α_t and α_{t-1} are the update coefficient of the classifier in previous frame and current frame separately.

B. SAKCF Tracking Algorithm

The KCF tracking algorithm detects the target position with a fixed size tracking window. Nevertheless, the size of the pedestrian will change with the movement of the vehicle, making it impossible to track the target accurately. In this paper, the scale correlation filter of DSST(Discriminative Scale Space Tracking) is added to the KCF tracking algorithm. The new algorithm, SAKCF, is adopted to establish training samples of different scales around the target location and then construct a scale filter to obtain the scale of the maximum response [20].

1) Principle of scale adaptation: Scale detection is to extract the multi-scale cyclic shift samples in the central region of position detection, and then zoom the multi-scale samples to the fixed scale of the target by bilinear interpolation method. Fig. 9 is the schematic diagram of scale adaptation. Si means the sample of different scales, and Ri represents the corresponding output response.

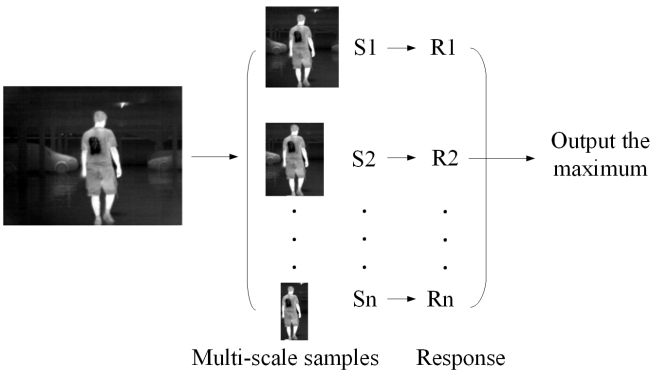


Fig. 9. Schematic diagram of scale adaptation

A 28-dimensional image feature block is used to train filters, composed of a one-dimensional gray feature block and a 27-dimensional HOG feature block. The scale filter can be obtained by training each dimension feature. The cost function of the l_{th} dimension is:

$$\varepsilon = \left\| \sum_{l=1}^{d} h^l \otimes f^l - g \right\|^2 + \lambda \sum_{l=1}^{d} \left\| h^l \right\|^2 \tag{27}$$

Where λ is the regularization parameter, h^l is the l-dimension scale correlation filter, and f^l is the target feature extracted in the l-dimension. g^l is the correlation filter response expected from the l-dimension. In order to obtain the approximate results, a scale filter that updates

the denominator and numerator separately is selected as:

$$H^l = \frac{\bar{G}F^l}{\sum_{k=1}^{d} \bar{F}^k F^k + \lambda} = \frac{A_t^l}{B_t} \tag{28}$$

For the goal of adjusting to the change of target scale, a series of samples with various scales are selected, and the selecting principle of sample size is:

$$a^n P \times a^n R, n \in \left\{ \left[-\frac{S-1}{2} \right], \cdots, \left[\frac{S-1}{2} \right] \right\} \tag{29}$$

Where P and R are the width and height of the target window of the previous frame, respectively. The scale factor α is 1.02, and the total levels of scale filter S is 33. $\lfloor \rfloor$means rounding down. Concatenating 33 scale samples into 33-layer scale pyramid, the map of response value is obtained based on the calculation formula(30), and the maximum response is the size of the current frame target. The formula is:

$$y_s = F^{-1} \left\{ \frac{\sum_{l=1}^{d} \bar{A}^l Z^l}{B + \lambda} \right\} \tag{30}$$

Where A_{t-1}^l and B_t are the numerator and denominator of the update filter in the previous frame, respectively.

The corresponding model update strategy is:

$$A_t^l = (1-\eta) A_{t-1}^l + \eta \bar{G}_t F_t^l \tag{31}$$

$$B_t = (1-\eta) B_{t-1} + \eta \sum_{k=1}^{d} \bar{F}_t^k F_t^k \tag{32}$$

Where η is the learning rate.

2) The tracking steps of SAKCF: SAKCF, which takes the KCF tracking algorithm as the tracking frame and introduces the target scale estimation after acquiring the target position, changes with the size of the target adaptively. The flowchart of SAKCF is shown in Fig. 10.

Detect the pedestrian in the sequence, and obtain the image block $\mathbf{x}_1$ at the target position. Then train the classifier to achieve the initial position filter coefficient α and the scale filter parameter A.

According to equation (24), calculate the position-related filter response in the i-th frame. We can determine the target position of the current frame according to the maximum value of the response.

Take the currently predicted target position as the center, and extract multi-scale samples according to equation (29).

Calculate the response of the scale kernel correlation filter, and the greatest value is the required scale.

According to the predicted scale and the predicted target position, calculate the position of the tracking frame .

Update the filter model parameters α, A, B and target model $\mathbf{x}_i$.

C. Experimental Results and Analysis

1) Qualitative analysis: Continuous images are selected from the LSI far-infrared pedestrian data set to assess the performance of the proposed algorithm. The hardware environment is Inter (R) Core(TM) i5-8250U@1.6GHz, the software is Python 3.6, and the memory is 8G computer. The simulation results are as follows.

In the first situation, the size of the pedestrian decreases gradually. As shown in Fig. 11 and Fig. 12, the algorithm of SAKCF can track the target stably, and KCF also captures the target. Since the tracking box of the initial frame is too large and remains unchanged, the pedestrian always moves within the given area of the image.

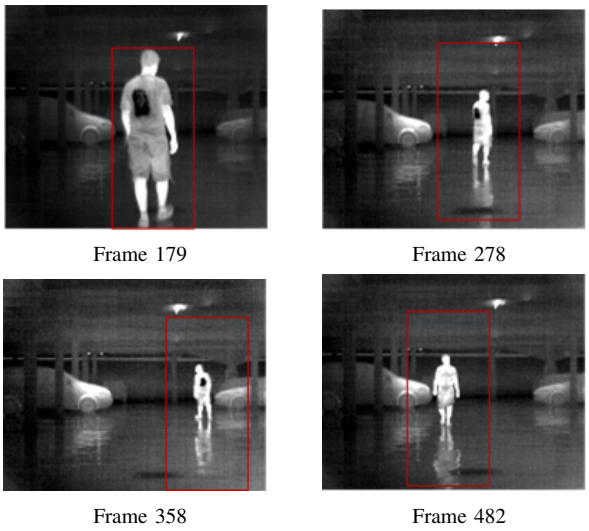


Fig. 11. Tracking result of pedestrian with decreasing scale by KCF

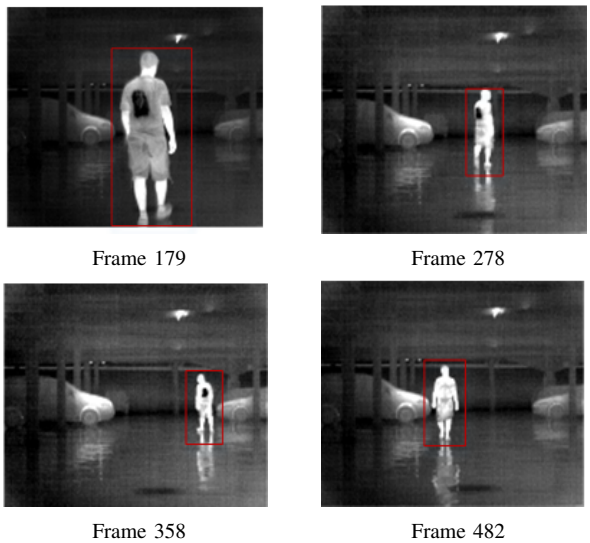


Fig. 12. Tracking result of pedestrian with decreasing scale by SAKCF

In the second situation, the size of the pedestrian increases gradually. As shown in Fig. 13 and Fig. 14, the KCF tracking algorithm leads to a blurry appearance when the target moves rapidly. Besides, the tracking area only contains local information, and the target is finally lost. Instead, the SAKCF algorithm can adjust the size of the tracking frame in time and can keep tracking continuously.

2) Quantitative analysis: For the purpose of evaluating the performance of the suggested method, this paper adopted Distance Precision (DP), success rate, and frame rate (Frames Per Seconds, FPS) for quantitative comparative analysis.

The Center Location Error (CLE) describes the distance

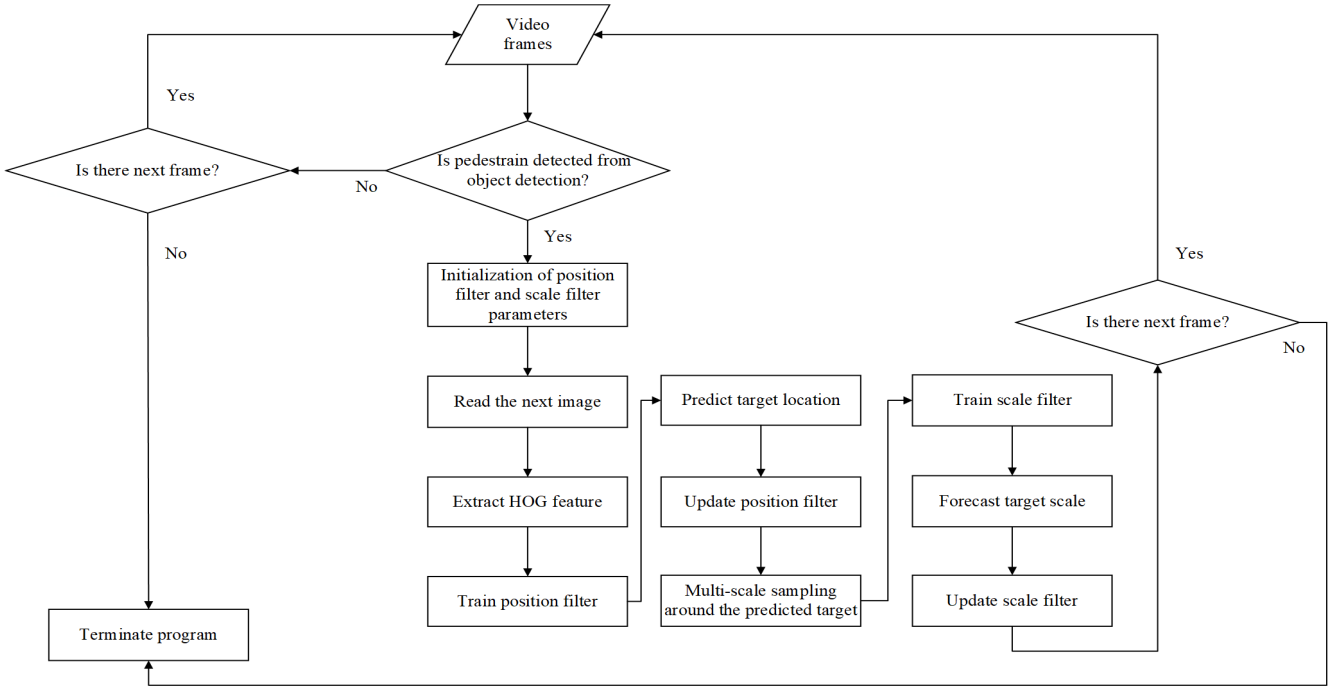


Fig. 10. Flowchart of SAKCF algorithm

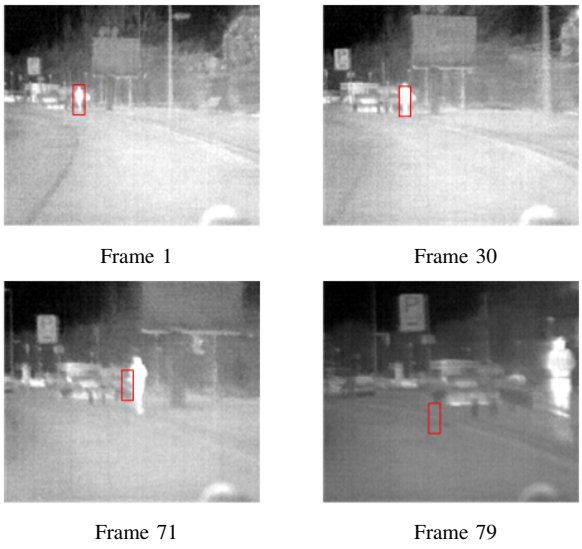


Fig. 13. Tracking result of pedestrian with increasing scale by KCF

Fig. 14. Tracking result of pedestrian with increasing scale by SAKCF

accuracy, expressed by equation (33).

$$CLE = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{33}$$

Where (x_1, y_1) represents the center obtained by tracking and (x_2, y_2) is the actual center. The smaller the CLE, the better the tracking effect. In this paper, 20 pixels are selected as the deviation threshold.

The success rate is expressed by Overlap Success(OS), as in equation (34). When the overlap rate of each frame is higher than a given threshold, it means that the frame is successful.

$$OS = \frac{area(S_G \cap S_T)}{area(S_G \cup S_T)} \tag{34}$$

Where S_G and S_T represent the actual target area and tracking target area, respectively. $\cup$ is the union set of the two areas, $\cap$ denotes the intersection of the two areas. The range of OS is in [0,1], and the threshold value is 0.5.

The frame rate is an indicator related to real-time performance. The comparison is shown in Table II, III, and Table IV in the aspect of distance accuracy, tracking success rate, and frame rate obtained by the two algorithms in situation one and situation two, respectively.

Table II and Table III illustrate the distance accuracy and tracking success rate with the SAKCF tracking algorithm of the first experiment, respectively, improving by 0.5% and 80% than KCF. Meanwhile, the distance accuracy and tracking success rate of the second experiment increases by 25.3% and 66.7%, respectively. Compared with KCF, no matter how the pedestrian target becomes smaller or larger, SAKCF can continue tracking targets and achieve better results. Although Table IV indicates that the calculation

speed of SAKCF is a little bit less than that of KCF, it can still achieve the real-time goal of pedestrian tracking for vehicle-mounted infrared video.

TABLE II
COMPARISON OF DISTANCE ACCURACY

Algorithm	Decreasing scale	Increasing scale
KCF	99.5%	74.7%
SAKCF	100%	100%

TABLE III
COMPARISON OF TRACKING SUCCESS RATE

Algorithm	Decreasing scale	Increasing scale
KCF	19.7%	31.1%
SAKCF	99.7%	97.8%

TABLE IV
COMPARISON OF FRAME RATE (f/s)

Algorithm	Decreasing scale	Increasing scale
KCF	87.64	330.53
SAKCF	30.69	125.26

V. CONCLUSION

In this work, SAKCF is proposed to track pedestrians based on vehicle-mounted infrared images. The edge information and brightness information of the pedestrian are obtained during the detection process through the HOG and HOI features. The result of pedestrian detection using SVM is excellent by traversing the image through sliding windows. During tracking, position detection and scale detection are performed, respectively. Two groups of typical vehicle-mounted infrared pedestrian tracking videos are selected in the experiment. The result shows that the tracking success rate in the two experiments increases by 80% and 66.7%, respectively. Even though the tracking speed decreases to some extent, it still meets the real-time requirements. According to the result of the experiment, we can conclude that the method can be applied to track the pedestrian with solid robustness.

REFERENCES

[1] Z. W. Xu, J. J. Zhuang, Q. Liu, J. K. Zhou, and S. W. Peng, "Benchmarking a large-scale fir dataset for on-road pedestrian detection," *Infrared Physics & Technology*, vol. 96, pp. 199–208, 2019.

[2] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2544–2550.

[3] J. F. Henriques, C. Rui, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part IV*, pp. 702–715.

[4] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[5] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1090–1097.

[6] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, and M. H. Yang, "Hedged deep tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4303–4311.

[7] P. Liu, C. Liu, W. Zhao, and X. L. Tang, "Multi-level context-adaptive correlation tracking," *Pattern Recognition*, vol. 87, pp. 216–225, 2019.

[8] L. Wang, "Research on vehicle infrared night vision pedestrain detection and tracking technology," Ph.D. dissertation, University of Electronic Science and technology, 2017.

[9] M. F. Gao and X. X. Zhang, "Scale adaptive kernel correlation filtering for target tracking," *Laser Optoelectronics Progress*, vol. 55, no. 4, pp. 284–290, 2018.

[10] L. Zhang, Y. J. Wang, and H. H. Sun, "Adaptive scale target tracking using nuclear correlation filter," *Optics and Precision Engineering*, vol. 24, no. 002, pp. 448–459, 2016.

[11] Q. X. Hu and P. Lu, "Pedestrian detection in infrared images based on multi-feature fusion," *Journal of Tongji University*, vol. 036, no. 0z1, pp. 157–160,195, 2016.

[12] S. M. Zhang, J. P. Hu, and Y. Shi, "Pedestrian tracking in infrared video based on improved particle filter," *Journal of Tongji University (Natural Science)*, pp. 1883–1887, 2015.

[13] X. Zhang, "Research on pedestrian tracking method based on infrared image," Ph.D. dissertation, Nanjing University of Aeronautics and Astronautics, 2019.

[14] S. Gite and K. Kotecha, "Evaluating the impact of ann architecture for driver activity anticipation in semi-autonomous vehicles," *Engineering Letters*, vol. 29, no. 3, pp. 873–880, 2021.

[15] K. Y. Cheng, W. X. Shi, B. W. Zhou, and J. X. Wu, "Robust kcf pedestrian tracking method with complex scene," *Journal of Nanjing University of Aeronautics Astronautics*, pp. 625–635, 2019.

[16] Y. Meng and X. Yang, "Overview of target tracking algorithms," *Acta Automatica Sinica*, vol. 045, no. 007, pp. 1244–1260, 2019.

[17] Y. Z. Zhang, R. Zheng, J. N. Bao, and S. D. Zhu, "Pedestrian tracking scale algorithm based on multiple correlation filters," *Journal of Northeastern University(Natural Science)*, pp. 1228–1233,1239, 2019.

[18] A. F. Jamali, A. Mustapha, and S. A. Mostafa, "Prediction of sea level oscillations: Comparison of regression-based approach," *Engineering Letters*, vol. 29, no. 3, pp. 990–995, 2021.

[19] X. Yang, Z. Hao, Y. Lei, C. Yang, and P. X. Liu, "A joint multi-feature and scale-adaptive correlation filter tracker," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2018.

[20] R. Soundrapandiyan and P. V. S. S. R. C. Mouli, "An approach to adaptive pedestrian detection and classification in infrared images based on human visual mechanism and support vector machine," *Arabian Journal for Science and Engineering*, vol. 43, no. 8, pp. 3951–3963, 2018.