Pedestrian Multi-target Tracking Based on SAMF-TLD

Bao Liu, Yanrui Yang, Yuge Zhao

Abstract- **The paper presents an algorithm for multi-target pedestrian tracking based on a tracking-learning detection (TLD) algorithm and a scale-adaptive correlation filter (SAMF). The SAMF algorithm incorporates a scale pool method during tracking, which enhances the robustness of target tracking. However, it lacks an occlusion judgment mechanism, limiting its performance in scenarios involving occlusions. When the target is occluded, the fixed template update method is used to update the filter template, leading to template pollution and algorithm drift. The SAMF algorithm with occlusion judgment proposed in this paper not only addresses this issue but also evaluates the target tracking state by the relationship between the average peak correlation energy and the peak sidelobe ratio of the current frame and the historical mean values. Additionally, the SAMF-TLD algorithm presented in this paper integrates a short-term tracker with the TLD method, enhancing the algorithm's real-time performance and addressing the tracking failure issues of the TLD algorithm under large-scale variations and changes in illumination. Two datasets are used for tracking validation: MOT-16 and 2DMOT-15. Common evaluation measures for multi-target tracking were used for comparison and analysis. The results of the experiment demonstrate that this method outperforms the TLD algorithm in terms of accuracy and real-time and outperforms the SAMF algorithm in the context of anti-occlusion.**

Index Terms—**multi-target tracking, SAMF-TLD, multi-scale kernel correlation filtering, multi-peak confirmation redetection**

I. INTRODUCTION

Pedestrian multi-target tracking [1] refers to the ability to simultaneously identify multiple targets of interest in a video clip and provide accurate and comprehensive motion trajectories for these targets over time. Pedestrians, as typical non-rigid targets, experience significant posture changes during movement and are highly susceptible to occlusion, rendering them challenging to track compared to vehicles and other rigid targets. Consequently, many existing multi-target tracking algorithms concentrate on pedestrians, drawing upon a substantial research foundation in this area. In general, it is difficult to find a general principle for categorizing pedestrian multi-target tracking algorithms. According to the order of trajectory generation, it can be divided into Detection-Based-Tracking (DBT) [2] pedestrian tracking methods and Detection-Free-Tracking (DFT) [3] pedestrian tracking methods. The DBT class algorithm performs target detection on images from different video frames with an excellent detector for each frame of the video sequence and crops the pedestrian image targets according to the bracketing frame to get all the targets in the image. Data association is conducted on the targets, transforming it into a target association problem between the previous and subsequent frames. The DFT class algorithm is characterized by the absence of a detector, which can be used to balance the quality of the input data.

To address challenges such as occlusion, scale variation, missed detections, false detections, and ID switching in pedestrian multi-target tracking, researchers worldwide have proposed a wide range of algorithms. Among the two classes of algorithms mentioned above, the performance of DBT class algorithms depends [4], to a certain extent, on the target detection model employed and requires offline training beforehand [5-9]. The type of target tracked by DBT class algorithms is solely determined by the results of the detection algorithm and cannot be predicted. Conversely, the DFT method is model-free, eliminating the need for specific target detector training and enabling tracking of any target type. This paper presents the requirement for pedestrian multi-target tracking algorithms to achieve long-term tracking of arbitrary targets. So, the method used in this paper is the DFT class algorithm. In the DFT class of algorithms, Kalal proposed the TLD tracking algorithm [10], which is a long-time tracking algorithm that combines three modules of tracking, detection, and learning with excellent performance. Martin proposed the Discriminative Scale Space Tracker (DSST) algorithm, which achieves high tracking accuracy and robustness by constructing a scale space pyramid [11]. Li Y et al. proposed the SAMF algorithm [12], which combines Histogram of Oriented Gradient (HOG) and Color Names (CN) features with a panning filter to detect targets on a multi-scale scaled image block, selecting the panning position with the largest response and the corresponding scale [13]. These algorithms have good robustness and real-time performance, but they struggle to address issues related to illumination variation, target scale changes [14], and poor tracking accuracy when the target is occluded [15].

Manuscript received August 16, 2024; revised February 08, 2025.

This work is supported in part by grant for Beilin District Science and Technology Plan Project (GX2231), the Key Research and Development Program of Shaanxi (2021GY-131), and Yulin Science and Technology Plan Project (CXY-2020-037).

Bao Liu is an associate professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (corresponding author, +86-18149067968, e-mail: xiaobei0077@163.com).

Yanrui Yang is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1592485528@qq.com).

Yuge Zhao is an engineer of Xi'an FinDreams Battery Co., Ltd. Xi'an 710119, China (e-mail: 2686034026@qq.com).

In response to the above issues, this article proposes an occlusion-resistant SAMF-TLD algorithm, with the main contributions summarized as follows.

(1) To address the issue of tracking loss caused by prolonged target occlusion, we propose a SAMF-based target tracking method that incorporates an occlusion judgment mechanism. The method evaluates the target tracking state by examining the relationship between the average peak correlation energy of the current frame and the peak parallax ratio and the historical frame average. It achieves target re-detection using the multi-peak confirmation re-detection method.

(2) To address the decline in tracking accuracy when a target significantly modifies the TLD, this paper proposes the SAMF-TLD algorithm. The method is inspired by TLD and integrates the anti-obscuration SAMF short-term tracking algorithm. It combines the color properties of CN and HOG and incorporates an adaptive updating strategy for filter templates to achieve robust target tracking.

(3) In the paper, the performance of the proposed method is evaluated using 2DMOT-15 and MOT-16 datasets using standard pedestrian multi-target tracking metrics. The results indicate that the SAMF-TLD algorithm can maintain stable and correct tracking when the target undergoes a large-scale variation and generates occlusion, and the tracking accuracy and precision are higher than those of TLD, SORT, and SAMF algorithms.

The structure of this paper is as follows. Chapter II introduces the TLD and SAMF algorithms. Chapter III proposes the SAMF-TLD algorithm and provides a detailed description. Chapter IV carries out simulation experiments and quantitative experiments on the proposed algorithm. Chapter V concludes this paper.

II. Related Works

A. SAMF Target Tracking Algorithm

SAMF target tracking belongs to the class of discriminative algorithms and is known for its high tracking accuracy. SAMF based on the KCF algorithm [16], HOG [17] features are fused with CN features [18-19] to take full advantage of the video frame color information, and the scale pooling method is introduced to improve target tracking robustness.

The SAMF algorithm comprises three parts: filter training, target localization and scale estimation, and model updating. At frame 1 of the video input, the SAMF algorithm approaches the training and computation of the filter by solving a ridge regression problem, denoted as:

$$\min_{f}\sum_{i}\| f^{T}x_{i}-y_{i}\|^{2}+\lambda\| f\|^{2} \tag{1}$$

where x_i is the i-th training sample in the round robin sampling; y_i corresponds to its regression label; f is the column vector, which denotes the weight coefficients; and λ is the regularization factor. After expanding to the kernel space, f can be expressed as:

$$f=\sum_{i}\alpha_{i}\varphi(x_{i}) \tag{2}$$

where $\varphi(x_i)$ is the function that maps x_i to a higher dimensional space. α_i is the weight coefficient, whose vectorization is denoted as α. The closed-form solution for α is:

$$\alpha=(K+\lambda E)^{-1}y \tag{3}$$

where K is the Gaussian kernel autocorrelation matrix of all training samples; E is the identity matrix; Y is the column vector, where each element represents a regression label y_i. The calculation can be performed using the Discrete Fourier Transform (DFT) and the transformation of the properties of the cyclic matrix into the frequency domain:

$$\hat{\alpha}=\frac{\hat{y}}{\hat{k}^{xx}+\lambda} \tag{4}$$

where $\hat{k}^{xx}$ is the element of row 1 of matrix K; superscript " ^" is the fast Fourier transform operation of the vector. During the tracking process, the SAMF algorithm generates the response output for each image block in the frequency domain by applying the appropriate filtering operation after extracting seven image blocks of various scales that are centered on the target point of the preceding frame. and then converts to the time domain and seeks its maximum response value $F_{\max}$ using the discrete Fourier inverse transform, the formulas are as follows:

$$F_{\max}=\arg\max[F^{-1}(\hat{k}^{xZ_s}\odot\hat{\alpha})] \tag{5}$$

where F^{-1} denotes the discrete Fourier inverse transform; $\hat{k}^{xZ_s}$ represents the first element of the Gaussian kernel inter-correlation matrix between the training samples and the test samples; Z_s is the test sample of the j-th scale block; $F_{\max}$ the corresponding coordinate value predicts the target center of the current frame. Finally, the SAMF uses linear weighting to update the model with the target samples x_t and coefficient vectors α_i of the current frame t, as shown in the following equation:

$$\begin{cases}\hat{x}^{t}=(1-\eta)\hat{x}^{t-1}+\eta\hat{x}^{t}\\ \hat{\alpha}^{t}=(1-\eta)\hat{\alpha}^{t-1}+\eta\dfrac{\hat{y}}{\hat{k}^{Z_sZ_s}+\lambda}\end{cases} \tag{6}$$

where η is the learning rate, and η is taken to be 0.02 in the experiment, $\hat{x}^{t-1}$ and $\hat{\alpha}^{t-1}$ denote the target sample and the coefficient vector of the previous frame, respectively; and $\hat{k}^{Z_sZ_s}$ denotes the element of the first row of the Gaussian kernel autocorrelation matrix of the test sample Z_s.

The SAMF algorithm is a rapid short-term tracking method that swiftly predicts the target's center and position, while simultaneously performing real-time online updates to the detection model. However, as tracking time extends, the model increasingly incorporates background information, resulting in an accumulation of errors that leads to tracking errors.

In addition, the SAMF algorithm in the tracking process, the search area, and the target box size are fixed; consequently, significant variations in target scale can reduce tracking performance. The algorithm lacks sufficient robustness to address challenging issues in multi-target tracking, such as occlusion, deformation, rotation, and other interferences.

B. TLD Tracking Algorithm

The core idea of the TLD tracking algorithm [8] is that tracking, learning, and detection interact to track a moving target jointly. The algorithm integrates traditional tracking and detection approaches. The algorithm incorporates the P-N mechanism in the learning module [20], which effectively addresses tracking drift failures caused by target occlusion, ensuring improved robustness and accuracy. The detector uses the scanning window grid algorithm to initialize target image features and processes each image frame to correct tracker errors during tracking[21]. The tracker [22] utilizes the median optical flow method to continuously track the target and updates its motion trajectory in real time.

(1) Detector

The detection module serves to locate the target's position, and its results can reinitialize the tracker after a tracking module failure. Upon target loss, the module performs a global search of the new frame and promptly recaptures and resumes tracking when the target reappears. [23-26]. The framework of this module is shown in Fig.1.

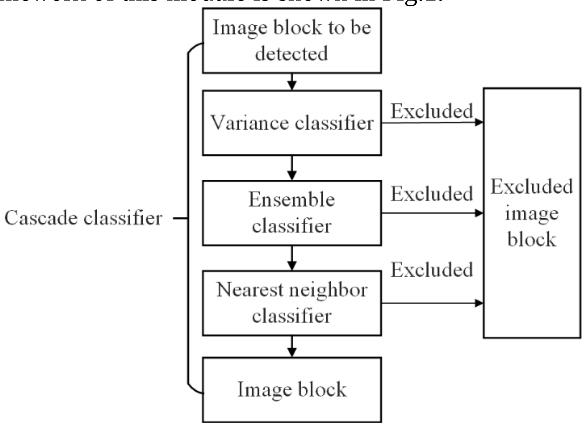

Fig. 1 Detection module framework diagram

(2) Tracker

The tracking module in the TLD uses the Median Flow Tracker (MFT) [27] method based on the pyramid LK optical flow method [28-29]. The method assumes that the target's brightness remains constant across two adjacent frames and that the target trajectory between these frames is both limited and visible. The target's position is then determined by selecting feature points from the previous frame, estimating their displacements between the two frames, and identifying their positions in the current frame.

(3) Learning Modules

The learning module utilizes a semi-supervised approach known as P-N learning theory. The framework of the learning module is illustrated in Fig. 2.

The P-N learning theory optimizes the classifier by leveraging both labeled and unlabeled datasets. Firstly, the module trains the base classifier using a small set of labeled samples. Secondly, it classifies the unlabeled samples by assigning positive and negative labels based on the trained classifier. The N-expert in the P-N learning module selects the most plausible result by comparing the detection module's output with the positive samples generated by the P-expert and then output this result as the final target position.

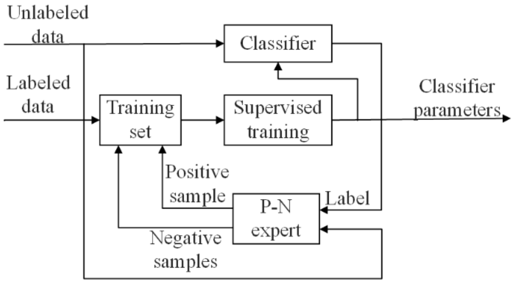

Fig.2 Learning module framework diagram

The P-N learning theory optimizes the classifier by leveraging both labeled and unlabeled datasets. Firstly, the module trains the base classifier using a small set of labeled samples. Secondly, it classifies the unlabeled samples by assigning positive and negative labels based on the trained classifier. The N-expert in the P-N learning module selects the most plausible result by comparing the detection module's output with the positive samples generated by the P-expert and then outputs this result as the final target position.

TLD multi-target tracking algorithm is a long-term tracking algorithm [30]. The re-detection function of the detection module ensures accurate target capture when the target is occluded or moves out of view and restarts the tracking module, which has the advantage of long-term tracking. The disadvantages of this method are also obvious. Firstly, the median optical flow method must satisfy the gray invariance, scale invariance and spatial consistency. Tracking tends to fail when illumination changes or motion deformation is large. Secondly, the operation efficiency is low, as the TLD algorithm processes a large number of image blocks, making it unable to meet the real-time requirements for target tracking. This limitation can be addressed by replacing it with a high-efficiency short-term tracker.

III. SAMF-TLD Pedestrian Multi-target Tracking Algorithm

The tracking and detection modules must operate concurrently and independently, correcting each other because the TLD algorithm is predicated on the optical flow method's lack of reliability. The TLD method's unreliable tracking results can be solved by using the SAMF algorithm, which provides excellent short time tracking performance. As a result, a SAMF-TLD algorithm is presented in this work. The algorithm is a long-time tracking algorithm capable of effectively handling occlusion, scale changes, illumination variations, and fast motion of the target. This method combines the advantages of the SAMF algorithm with adaptive template updating and the TLD long-term tracking algorithm, enhancing both the robustness and accuracy of the tracker.

A. Anti-obscuration SAMF Algorithm

The SAMF algorithm lacks sufficient robustness in handling occlusion interference. In this paper, we address this issue by introducing an occlusion determination method that combines Average Peak-to-Correlation Energy (APCE) and Peak-to-Sidelobe Ratio (PSR). This method employs different strategies based on the target's occlusion state. Th-

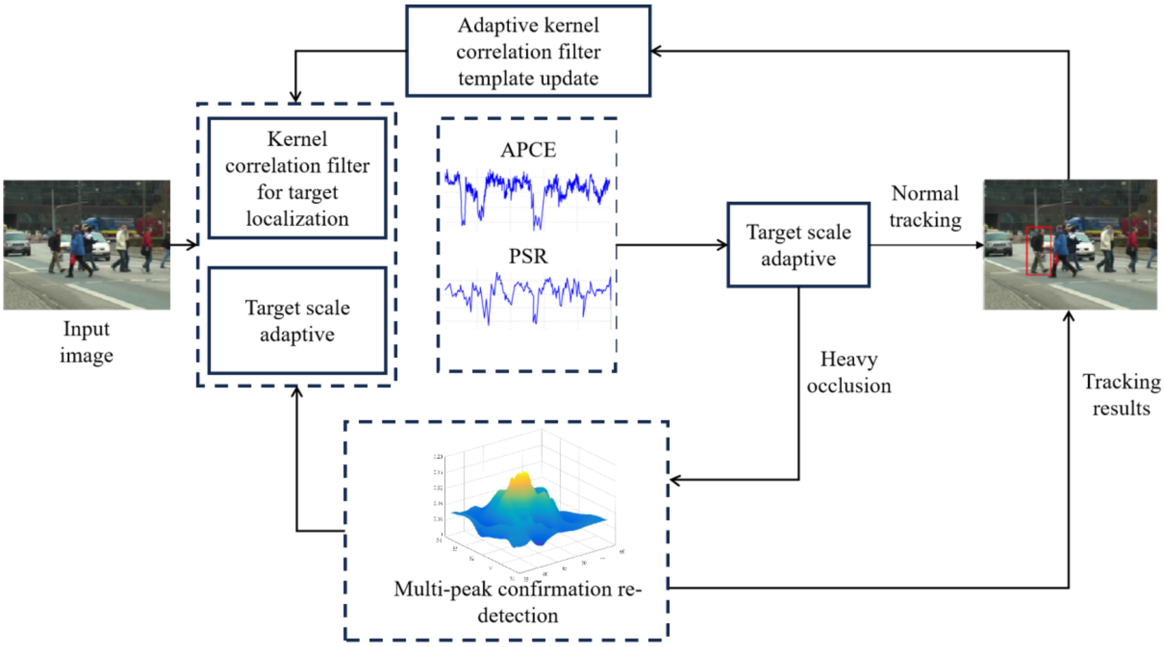

Fig. 3 Overall framework of SAMF algorithm for anti-obscuration

e algorithm adaptively adjusts the learning rate based on the target's APCE and PSR results when it is not occluded or only slightly occluded, preventing occlusion information from being introduced into the filter template. In cases of severe target occlusion, the algorithm employs a multi-peak confirmation re-detection module to enhance performance further by detecting the target. The algorithm framework in Fig. 3.

(1) Masking Judgment Mechanism

The output response map's degree of oscillation is primarily represented by APCE [31]; the greater the degree of peak oscillation, the lower the APCE value, and the worse target tracking accuracy. PSR reflects the robustness of the target tracker through the degree of sharpness of the main peaks if the main peaks of the target tracker are relatively sharp [32]. A high PSR value indicates good accuracy in the current target tracking. In the case where the target is occluded or otherwise interfered with, both APCE and PSR are significantly reduced. These two evaluation indexes evaluate the tracking effect of the target from different perspectives, and their combination can effectively improve the judgment accuracy of the occlusion evaluation criterion when the target is in a complex occlusion situation.

Based on APCE changes across different motion states, the threshold value A is set, when $W_{APCE} \geq A$, the target is considered to be in the first occlusion stage, indicating no occlusion or slight occlusion. Otherwise, the target is in severe occlusion and turns to full occlusion and stops the normal template updating and initiates the rechecking module to re-localize the target. In order to make the threshold A adaptively adjustable in the complex tracking environment, it is set as the product of the threshold adjustment coefficient λ and the APCE of the history frame, which is defined as follows:

$$A = \lambda \cdot \frac{\sum_{i=1}^{t-1} W_{APCE_i}}{t-1} \tag{7}$$

where the adjustment factor λ is taken as 0.6 according to the empirical value, A denotes the judgment threshold, and W_{APCE_i} denotes the APCE value of the i th frame.

The judgment threshold B is set based on variations in PSR values under different target occlusion conditions. If $PRS_t > B$, it indicates that the peak value on the target response map has a higher signal-to-noise ratio relative to the side-lobe energy, and the target tracking algorithm has a better anti-jamming performance and tracks the target normally. If $PRS_t < B$, this indicates that the target tracking algorithm is affected by interferences, such as occlusion, which leads to a decrease in the peak value on the target response map and reduced tracking accuracy. Stop the current filter template update and re-evaluate the target's current position.

To make the PSR adaptive across different sequences, it is set as the product of the adjustment coefficient δ and the PSR of the historical frame, defined as follows:

$$B = \delta \cdot \frac{\sum_{i=1}^{t-1} PSR_i}{t-1} \tag{8}$$

where PRS_i represents the PSR value of frame i and the adjustment factor δ takes the empirical value of 0.7.

In summary, to ensure the robustness of the target in various tracking environments, this paper fuses two kinds of occlusion judgment quasi-indicators, APCE and PSR, and the fused indicator is the occlusion judgment mechanism this paper, defined as follows:

$$\theta = \begin{cases} \theta_t, APCE_t \geq A \cap PSR_t \geq B \\ 0, others \end{cases} \tag{9}$$

where θ represents the filter template learning rate. If $\theta = \theta_t$, it indicates that the target is in the first stage state, and the template update can be performed with learning rate θ_t; when $\theta = 0$, W_{APCE_i} and PRS_i do not satisfy the judgment mechanism threshold condition, it indicates that the tracking results of the current frame have low credibility, stopping the template update and retaining the valid target information to start the re-detection algorithm.

(2) Template Updates

The learning rate curve is constructed based on the magnitude relationship between the current frame APCE value W_{APCE_t} and the threshold value A, and the filter template learning rate θ_t is adapted according to the target tracking output response map, as defined by the following equation.

$$\theta_t = \frac{1}{1+e^{-\varepsilon(\frac{W_{APCE_t}-A}{A})}}\theta_0 \tag{10}$$

where θ_0 denotes the initial frame learning rate, which is set to $\theta_0 = 0.02$, and ε denotes the learning rate adjustment parameter.

The algorithm applies an adaptive learning rate to update the filter template when tracking confidence is high. The template update formula is as follows.

$$\begin{cases} \hat{x}^t = (1-\theta_t)\hat{x}^{t-1} + \theta_t \hat{x} \\ \hat{\alpha}^t = (1-\theta_t)\hat{\alpha}^{t-1} + \theta_t \hat{\alpha} \end{cases} \tag{11}$$

where $\hat{x}^{t-1}$ and $\hat{\alpha}^{t-1}$ denote the target samples and coefficient vectors of the previous frame, respectively. Through adaptive template updating, the algorithm ensures the effective tracking of the target under normal motion and slight occlusion.

(3) Multi-peak Confirmation Weight Detection Module

The output response plot exhibits a single, significant peak when the target is not occluded. The highest peak position is selected as the target's position for tracking and prediction.[33]. If the target is severely occluded, multiple peaks appear on the response map, with no prominent main peak. This indicates the existence of multiple positions with high similarity to the target template, necessitating multi-peak re-detection to determine the target's position more precisely.

Multi-peak re-detection for target location determination can. On one hand, it addresses the issues of missed detections and false detections that may arise from relying on a single main peak for detection. On the other hand, different main peaks may represent different target features, reducing the algorithm's dependency on any single feature.

For the initially generated output response map, the positions of multiple peaks can be identified using the following formula:

$$D(s) = N \cdot P(x,y,z) \tag{12}$$

where $D(s)$ indicates that the non-zero element corresponds to the peak value in the output response map. N indicates the local maximal value position, so that the local maximal value position is 1 and the rest of the positions are set to 0, and $P(x,y,z)$ indicates the local maximal value three-dimensional coordinates. The redetection module assesses multiple peaks in the response map. If the following equation is met, a secondary peak position detection is required.

$$P_i \geq \xi \cdot P_{\max} \tag{13}$$

where P_i indicates the size of the i th ($i \leq 3$) peak in the output response map, ξ is the re-detection threshold coefficient, which is taken as 0.5 according to the empirical value. To enable real-time tracking in the SAMF algorithm, the re-detection module is only the secondary detection of up to three response maximal peaks and redetection of the target location to be satisfied by the following equation:

$$P(t) = \max(D_i') \tag{14}$$

where D_i' represents the maximum value of the secondary detection output response centered at $D(i)$, and $P(t)$ represents the target's current position after re-detection.

As shown in Fig. 4, the output response map produces multiple peaks when the target becomes occluded, and the target's location cannot be determined from the response map.

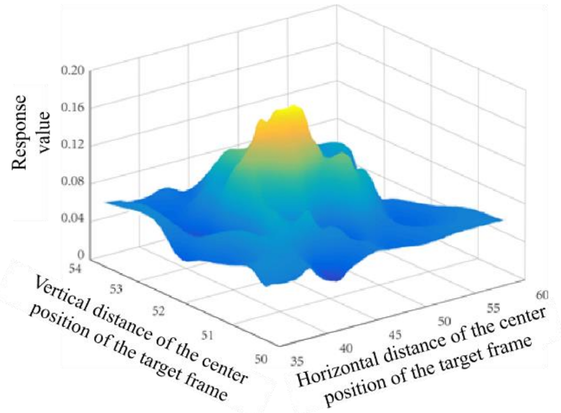

Fig. 4 Schematic diagram of multi-peak confirmation redetection module

The multi-peak confirmation detection module performs secondary re-detection to accurately determine the position of the target after detecting multiple possible target positions. In Fig. 4, three local maxima satisfy the redetection threshold condition, with values of 0.161, 0.149, and 0.137, respectively. To refine the target's location, the location corresponding to these three local maxima can be centered on the target's location, followed by a secondary detection using the SAMF filtering algorithm. In the second detection, a new response map is obtained, and the location of the maximum response is selected as the location of the re-detected target. This approach enhances the target tracking accuracy and robustness under occlusion interference and avoids erroneous target tracking due to a single feature.

B. SAMF-TLD Tracking Algorithm

An effective tracking algorithm should ensure forward and backward continuity, with the target movement's forward and backward tracking trajectories align consistently. The TLD algorithm needs to iteratively calculate from the top of the image pyramid to the bottom in processing video. In actual tracking, variations in optical flow can cause inconsistencies between the forward and backward tracking trajectories. Additionally, the TLD algorithm lacks a color-based feature map and relies solely on grayscale information, making it prone to confusing the target with the background. The SAMF algorithm fully utilizes color information between the video frames to improve the robustness of the target tracking.

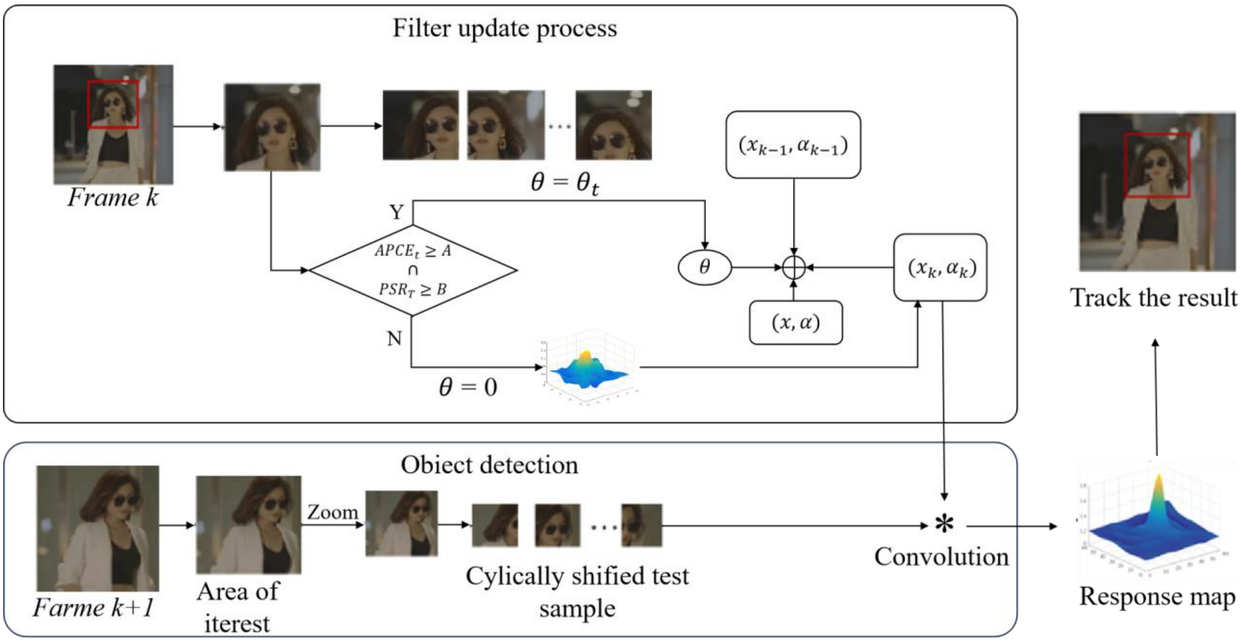

Fig. 5 Schematic diagram of SAMF-TLD algorithm tracking module

The SAMF algorithm includes three steps: initial frame-to-filter training and follow-up tracking. The follow-up tracking includes target detection, filter template updating, and multi-peak confirmation re-detection.

1) Filter training. An image block of size $M \times N$ is selected in frame $M \times N$ to train the filter. The base sample image block is circularly shifted in rows and columns to yield 1 training sample; each sample is set to be $x_i, i = 0,1,\cdots M \times N - 1$ and the set is denoted as X, and each sample is labeled with a two-dimensional Gaussian function, which is denoted as $y_i, i = 0,1,\cdots M \times N - 1$, and the set is denoted as 5.

The response value of $f(z) = \omega^T \cdot z$ is the basis for the SAMF algorithm's classification of the samples. To determine the ideal parameter ω, the classifier is trained as follows:

$$\omega = \arg\min_{w} \left|\varepsilon(x_{m,n}) \odot w - y(m,n)\right|^2 + \lambda \left\|w\right\|^2 \tag{15}$$

where ω represents the classifier's ideal parameters, ε is the kernel mapping space, λ is the regularization parameter, $x_{m,n}$ denotes the data blocks derived from all cyclic shifts, and $y_{m,n}$ denotes the Gaussian function labeling performed on the classified samples.

To find the optimal solution, solve the above equation in the frequency domain using the discrete Fourier transform:

$$\omega = \sum^{m,n} \alpha(m,n)\varepsilon(x_{m,n}) \tag{16}$$

The factor α is expressed as:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}^{xx} + \lambda} \tag{17}$$

where y is the sample label, superscript " $\hat{}$ " denotes the frequency domain transform, and $\hat{k}^{xx}$ denotes the auto-correlated Gaussian kernel output of the training g sample.

2) Target detection. The TLD algorithm uses multiple scales in the detection module to ensure that the output target state closely aligns with the actual motion state of pedestrians. In this paper, the algorithm adds a scale prediction step into the tracking result. It incorporates the scale scaling function from the SAMF algorithm, further enhancing the tracker's scale adaptability. After inputting the next frame image into the detection module, the region of interest in the current frame is selected based on the location and size of the result from the last frame, and the region size is adjusted to seven different scales $M \times N, M, N \in \{0.985, 1.015\}$ by scale scaling method. The resized picture is saved as sample set Z, and the sample filtering response is computed, that is:

$$\hat{f}(z) = \left(\hat{\mathrm{K}}_{x_0,z}\right)^* \otimes \hat{\partial} \tag{18}$$

where ∂ is the corresponding set of coefficients for every sample, $\left(\hat{\mathrm{K}}_{x_0,z}\right)^* \otimes \hat{\partial}$ is the inner product of the kernel function and the set of corresponding coefficients in the high-dimensional space, and $\hat{\mathrm{K}}_{x_0,z}$ is the kernel function for frequency domain operation. The horizontal and vertical coordinates of the maximum response point in the filtered response value obtained in the above equation are the displacement of the target in the current frame corresponding to the previous frame.

3) Filter update. The filter template (x, α) needs to be modified to the target's motion. In Fig. 5, the updating procedure is displayed.

In the case where the target is occluded, the tracker may need to reconstruct the target model to adapt to the target's new location and morphology. However, the frame-by-frame update method in the SAMF algorithm relies solely on the current frame's information, making it challenging to recover the occluded target model. Consequently, the frame-by-frame update method may fail to restore the effectiveness of the tracker when the target is occluded. This section determines the stage of the target by the occlusion judgment criterion, which then provides the current template update learning rate as the criterion for adaptive filter template updating, which yields the current template update learning rate θ as the filter template adaptive update criterion.

Tracking results for the current frame (k frame) is input to the APCE fusion PSR occlusion judgment mechanism, as shown in Fig. 5, which yields a learning rate of θ. If $\theta=\theta_t$, it means that the target is in the state of slight occlusion, and the filter templates are updated; on the contrary. If $\theta=0$, the multi-peak confirmatory detection module is activated, identifying the current maximum peak as the target position. The tracked image block of the current frame is first obtained by the algorithm as the training sample using cyclic shift. Then, templates learned from the current frame are obtained through the Fourier transform and used to update the filter template. Finally, the filter template is adjusted and trained according to the predetermined learning rate. Filter templates after updating (x_k, α_k) is obtained, and the sample template and coefficient template update process are given in the following equation:

$$\begin{cases} x_k = (1-\theta)x_{k-1} + \theta x \\ \alpha_k = (1-\theta)\alpha_{k-1} + \theta\alpha \end{cases} \tag{19}$$

where (x, α) denotes the current filter template parameter.

4) Redetection module. In the SAMF tracking phase, when the target is severely impeded, the learning rate is $\theta=0$. Currently, multiple peaks appear in the target response map. The current filter template update strategy should be paused, and the redetection module should be activated to re-confirm the peaks of the target response map. The redetected target location is obtained as follows.

$$l_t = \max(F_i') \tag{20}$$

where l_t denotes the target position and F_i' denotes the maximum value of the output response obtained after the secondary detection. The algorithm tracking flowchart in Fig. 6.

This subsection proposes a SAMF-TLD long-term multi-target pedestrian tracking algorithm, which is based on the TLD target tracking algorithm with the ability to resist large deformation and occlusion. Subsection four analyzes the performance of the proposed algorithm and verifies its validity from both qualitative and quantitative perspectives.

IV. Experimentations and Analysis

The experimental hardware environment is an AMD Ryzen 7 5800U with Radeon Graphics 1.90 GHz computer with 16.00 GB of RAM, using Windows 10 64-bit system, Python 3.7 and MATLAB R2018a are the platforms for developing algorithms. The experiment's threshold is set at 0.7, and all other parameters are maintained in line with the SAMF method. Since the traditional TLD method framework is limited to tracking single pedestrian motion targets, this study introduces SAMF-TLD, which uses a multi-threaded programming technique to efficiently track multiple pedestrian targets.

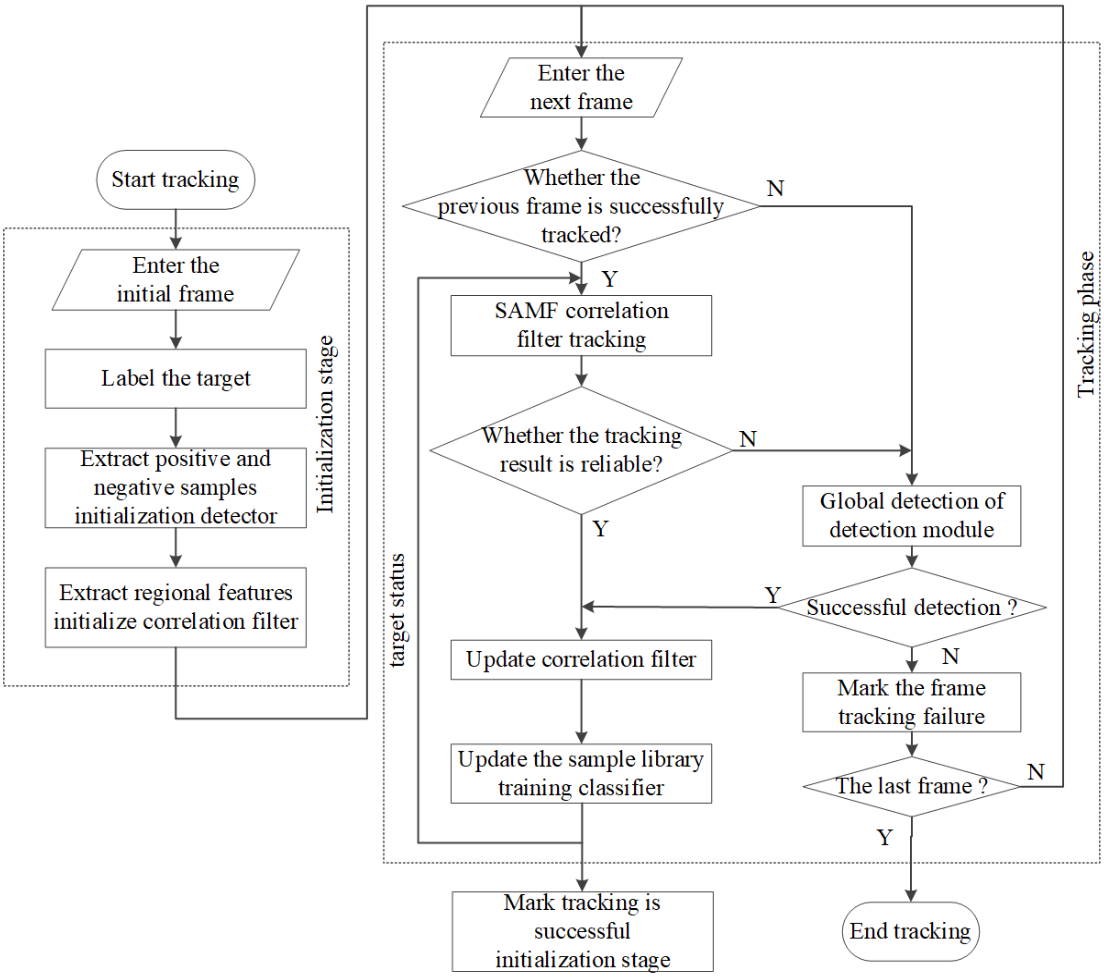

Fig. 6 SAMF-TLD algorithm tracking flowchart

A. Experimental Data

In this paper, four video sequences are selected from the datasets MOT16 test, 2DMOT15 test, and OTB100 dataset to verify the algorithm's effectiveness [34]. Additionally, we also compared the evaluation metrics of this algorithm with those of other algorithms. The information of public datasets in Table I, Table II, and Table III.

B. Evaluation Indicators

To evaluate the performance of our proposed method, this paper selects assessment criteria that are often applied in the multi-target tracking field. Such as MOTA, MOTP, MT, ML, FP, FN, IDS, and Runtimes, for quantitative analysis. The indicator description is shown in Table IV. Among them, the evaluation tracker prefers MOTA and MOTP; the greater the value, the better the tracking impact, and the rest of the indicators are the opposite.

TABLE I
MOT16 TEST DATASET INFORMATION

Data set	Picture resolution	Frame rate	Number of pictures
MOT16-14	1920*1080	25	750
MOT16-12	1920*1080	30	900
MOT16-08	1920*1080	30	625
MOT16-07	640*480	54	16332
MOT16-16	1920*1080	221	11538
MOT16-03	1920*1080	148	104556

TABLE II
2DMOT15 TEST DATASET INFORMATION

Data set	Picture resolution	Frame rate	Number of pictures
Venice-1	1920*1080	30	4563
KITTI-19	1238*374	10	5343
KITTI-16	1224*370	10	1701
ADL-Rundle-3	1920*1080	30	10166
ADL-Rundle-1	1920*1080	30	9306
AVG-TownCentre	1920*1080	2.5	7148
ETH-Crossing	640*480	14	1003
ETH-Linthescher	640*480	14	8930
ETH-Jelmoli	640*480	14	2537
PETS09-S2L2	640*480	7	9641
TUD-Crossing	640*480	25	1102

TABLE III
OTB100 PARTIAL VIDEO SEQUENCE INFORMATION

Data set	Picture resolution	Frame rate	Interference information
Coke	640*480	291	Severe occlusion, illumination change, rapid movement, similar background, and rotation inside and outside the plane
Subway	640*480	175	Severe occlusion, target deformation, similar background

TABLE IV
EVALUATION INDICATORS

Index	Implication
MOTA	Measure the accuracy of the algorithm to track the target
MOTP	Show the degree to which the detection frame and the prediction frame correspond.
MT	More than 80 % of the targets can successfully match the trajectory ratio
ML	More than 20% of track ratios are correctly tracked
FP	Number of misdetected targets
FN	Total number of goals that were missed
IDS	The switching times of the target ID
Runtimes	Operating speed

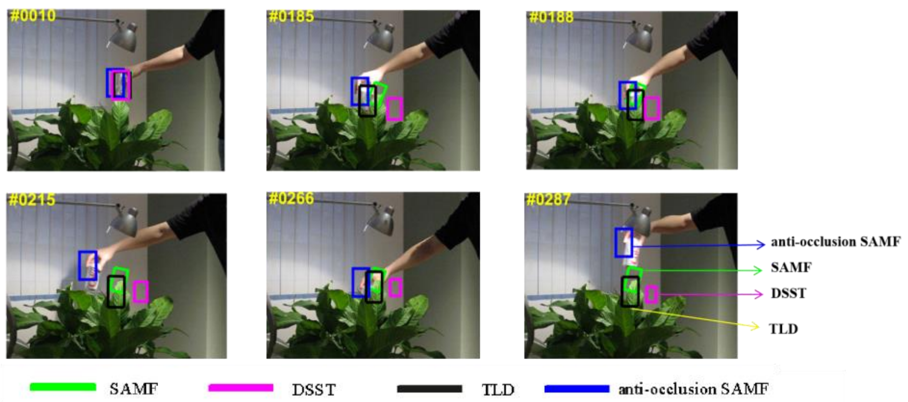

Fig. 7 Example graph of tracking results for Coke video sequences

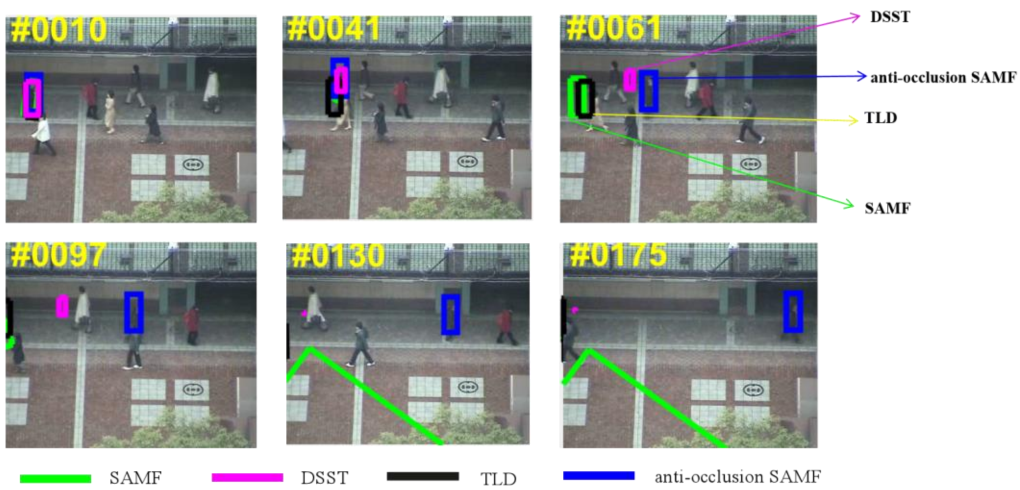

Fig. 8 Subway video sequence tracking result

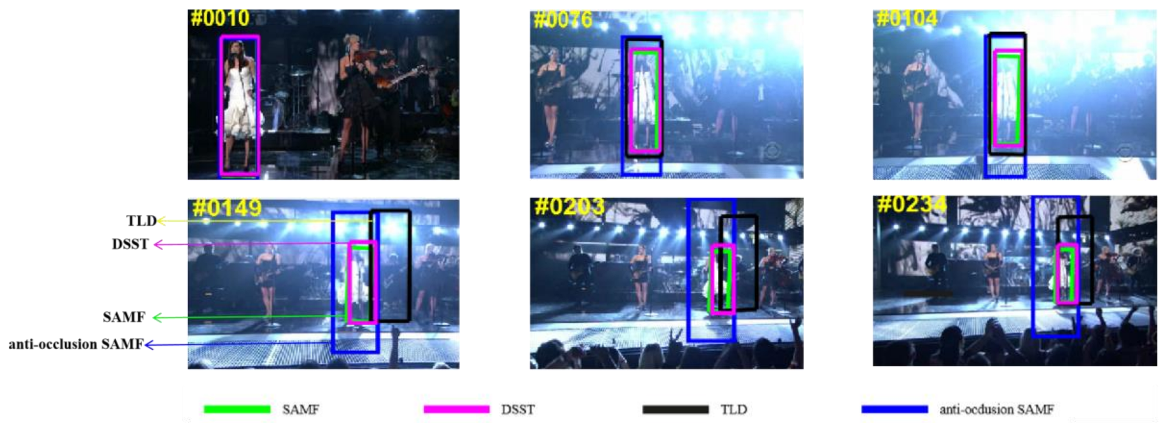

Fig. 9 Singer1 video sequence tracking result

C. Simulation Experiment

(1) Anti-obscuration SAMF Algorithm

In the comparison experiments in this section, SAMF, DSST, and TLD algorithms are used as the comparison algorithms for the anti-occlusion SAMF algorithm. The results of the comparison experiments are shown in Fig. 7, Fig. 8, and Fig 9. The green, purple, black and blue boxes represent the tracking results of the SAMF algorithm, DSST algorithm, TLD algorithm, and anti-occlusion SAMF algorithm, respectively. The initial frame of the three video sequences begins from the tenth frame. The top-left corner displays the automatically labeled frame sequence number, while the tracking results of the first frame are manually annotated.

Fig. 7 shows the tracking results of the four algorithms on the Coke video sequence in the OTB100 dataset, which includes six interference factors, illumination change, fast motion, similar background, in-plane rotation, and out-of-plane rotation.

In the 185th frame, the canister moves behind the green leaf, causing the target to become occluded. At the same time, in the process, it is accompanied by in-plane and out-of-plane rotations. The SAMF algorithm employs a fixed learning rate in the tracking process to update the template. When the target is affected by the occlusion, the correct template update learning rate cannot be determined, causing the algorithm drifting to the original motion trajectory of the target, specifically to the right of its true position. The DSST algorithm faces the issue of failing to track the target accurately when the tracking template is contaminated, resulting in tracking drift. The TLD algorithm has the advantage that the detection module and the tracking module mutually correct the tracking results, effectively preventing tracking drift. Anti-obscuration SAMF algorithm introduces an anti-occlusion judgment strategy. When the target experiences occlusion, a reduced learning rate is used to update the target tracking template, ensuring stable tracking after the occlusion. In the 188th frame, the SAMF algorithm utilizes an incorrect tracking template, leading to tracking failure. In the 215 th frame, the canning target undergoes rapid motion, similar background and in-plane and out-of-plane rotation and leaves the occlusion area at the same time. Currently, the SAMF algorithm and the complete tracking drift continue to track the error along the initial trajectory of the target. Due to the use of the median flow tracking method, the TLD algorithm cannot track the target stably during rapid motion of the target, resulting in tracking drift. The DSST algorithm, similar to the SAMF algorithm, suffers from template contamination, causing the tracking frame to remain at the position where the target is occluded. Anti-obscuration SAMF algorithm still maintains correct tracking after the target leaves the occlusion area. In the 266 th frame, the target is heavily occluded, and the appearance features are all occluded by green leaves. In the 287 th frame, the target moves out of the occlusion area and encounters strong light changes. Anti-obscuration SAMF algorithm is still stable tracking and does not produce drift. However, the remaining three algorithms currently exhibit varying degrees of drift in the tracking box results due to cumulative tracking errors.

Fig. 8 shows the tracking results of the four algorithms on the Subway video sequence in the OTB100 dataset, and the interference factors of this video sequence include severe occlusion, target deformation and similar background.

In the 41st frame, the pedestrian in the black coat is occluded by nearby pedestrians, and the target begins to enter the occlusion area. At the 61st frame, with similar background and occlusion interference, the TLD algorithm, SAMF algorithm and DSST algorithm all have different degrees of drift. The SAMF algorithm does not have an adaptive template update strategy. With the accumulation of errors, the tracking box remains the position where the target is occluded. The SAMF algorithm proposed in this paper does not exhibit tracking drift. In the 97 th frame, the target is occluded twice. The appearance and color information of the target is gradually recovered after it exits the occlusion area in frame 130. The SAMF algorithm and the DSST algorithm over-learn the occlusion information and use a fixed error learning rate, resulting in the target. After a second occlusion, the tracker completely loses its accuracy, resulting in severe tracking drift. The target frame of the DSST algorithm is adjusted to a single point, and the occlusion target information that has been out of the field of vision cannot be searched. The SAMF algorithm tracking frame employs the principle of seven fixed-scale scaling to search for the target position. However, an incorrect tracking template still results in erroneous tracking outcomes. The improved SAMF incorporates an anti-occlusion mechanism, enabling it to promptly identify the occlusion stage of the target when occlusion occurs. To achieve adaptive multi-scale target stable tracking, different template update strategies can be used for different occlusion stages of the target. In the 175 th frame, the target movement is over. Currently, all three comparison algorithms remain near the position where the target undergoes the second occlusion, leading to severe tracking drift. However, the occlusion judgment mechanism introduced in the anti-occlusion SAMF algorithm enhances the robustness of target tracking to a certain extent when the target produces occlusion interference.

Fig. 9 shows the tracking results of the four algorithms on the Singer1 video sequence in the OTB100 dataset, which has four interference factors including severe occlusion, scale change, illumination change, and out-of-plane rotation.

In the 76th frame, the scale of the target changes and the stage illumination gradually becomes stronger. Currently, due to minimal target interference, all four algorithms maintain stable tracking. In frame 149, the target experienced extremely strong illumination until the color feature was almost lost. As the target moved, the stage light gradually dimmed, and the target began to experience slight occlusion interference. At the same time, both the SAMF and DSST algorithms result in the tracking box slightly deviating from the target's center position. In the 203rd frame, the target is out of the occlusion area. Due to error accumulation in the TLD algorithm, tracking errors occur. In contrast, the SAMF algorithm, the DSST algorithm, and the algorithm proposed in this chapter maintain stable tracking. In frame 234, the target rotated out of the plane and became smaller, and the algorithm in this chapter still did not produce tracking drift.

(2) SAMF-TLD Algorithm

In this part, we verify that the SAMF-TLD algorithm produces large-scale variation s in the target by tracking the pedestrian target in the ETH-Linthescher dataset in 2DMOT15, and the pedestrian in the MOT16-03 dataset in MOT16 is used to verify the tracking of the algorithm after the target is occluded. The multi-target tracking results of the test sequence are visualized. The specific results are shown in Fig. 10 and Fig. 11.

The pedestrian data set from the ETH-Linthescher in Fig. 10 is characterized by fixed diagonal shooting of the camera, weak light intensity, pedestrians becoming larger from far and near scales, and weak light changes. This data set is suitable for validating the effectiveness of the method in the presence of large-scale variations and illumination changes. The MOT16-03 pedestrian data set in Fig. 11 has many pedestrian targets. There are several scenarios, including out-of-view, mutual, and self-occlusion, and the volume of data is enormous. It is appropriate for algorithm effect verification in strong, mutual, and long-term occlusion scenarios. The MOT16-08 video sequence was recorded in an open-air shopping mall with a constant flow of people. The background is highly cluttered, presenting significant challenges for target detection and tracking. Furthermore, the target undergoes substantial appearance changes as it moves farther away from the camera, complicating the analysis.

Fig. 10 Tracking results for larger scales and deformations (a) Frame 1, (b) Frame 19, (c) Frame 39, (d) Frame 48, (e) Frame 51, (f) Frame 55

Fig. 11 Tracking results with dense occlusion and self-occlusion (a) Frame 1, (b) Frame 52, (c) Frame 85, (d) Frame 123, (e) Frame 148, (f) Frame 208

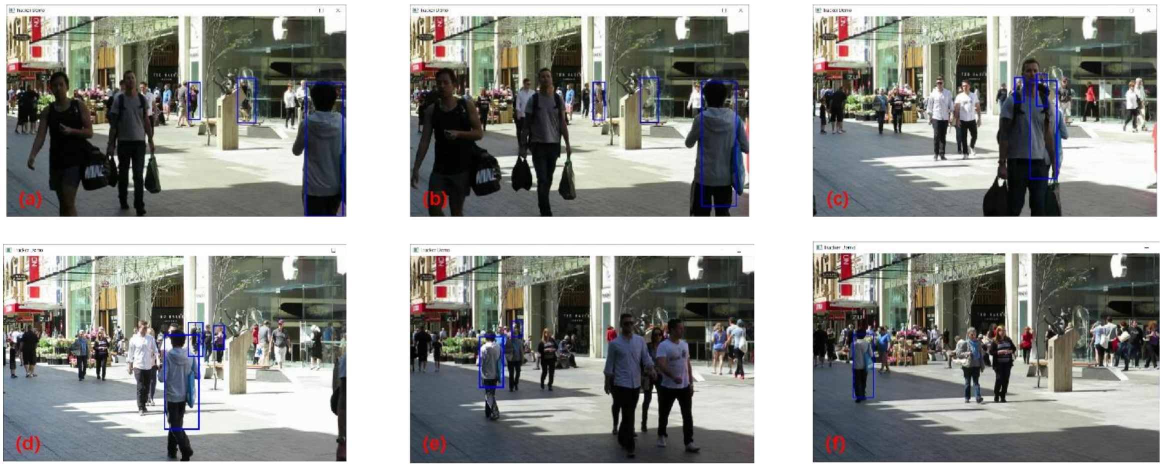

Fig. 12 Tracking results under large scale changes and occlusion interference (a) Frame 1, (b) Frame 15, (c) Frame 57, (d) Frame 90, (e) Frame 205, (f) Frame 435

Fig. 10(a) to (f) present the results of large-scale variations and illumination changes of the target in a static camera view. The figure shows three cases in the video sequence in which people coming from far and approaching, moving, and being occluded and leaving view. The algorithm can accurately recognize and track the target, whether it undergoes large-scale variations, slight occlusions, or appears small in size. There is no tracking inaccuracy or trajectory jumping, and the target's complete motion trajectory is preserved.

As shown in Fig. 11, the two moving targets on the right become obscured, experiencing mutual occlusion at frame 52. The left target remains partially masked until the two targets separate at frame 148. Despite this, tracking continues to operate as intended, with no tracking errors or loss. The woman on the left has entirely blended into the throng and lost her tracking features in frame 208, making the tracking frame disappear. The other two targets are still being tracked correctly and are not affected. The SAMF-TLD method sustains accurate tracking even under severe pedestrian occlusion and mutual target obstruction, enabling the continuation of the tracking task.

Fig. 12 (a) to (f) illustrate the pedestrian tracking results across three stages: the unoccluded state, the occluded state, and the point where the target moves out of view. In the first frame, the two targets farther from the camera are partially occluded. By the 15th frame, both targets are nearly completely occluded. In the 57th frame, only limited target information is retained for tracking. In frame 205, the woman wearing a purple jacket moves out of view, resulting in the target being classified as lost, and the tracking process is terminated. At this point, although the target farthest from the camera appears very small, tracking remains stable. At frame 435, the target farthest from the camera moves out of view. Nevertheless, the algorithm maintains stable tracking of the remaining targets without any tracking drift.

D. Analysis

This section presents simulation results that validate the effectiveness of the proposed algorithm in tracking the target under four interference scenarios: illumination change, large-scale and illumination change, a large number of targets, and large-scale changes with occlusion interference. We compare our algorithm in this chapter with TLD algorithm, SORT algorithm and SAMF algorithm [35] in terms of indicators and analyze the result data.

Firstly, the occlusion-resistant SAMF-TLD algorithm is compared with the TLD algorithm, the SORT algorithm, and the SAMF algorithm, respectively, on the KITTI-16 pedestrian video sequences, and the comparison results are shown in Table V.

As shown in the table, the algorithms all have high accuracy and can track the target better. The SAMF-TLD method has the highest accuracy and precision. The accuracy of the algorithm is improved by 19.9 % based on the TLD algorithm, and the accuracy is marginally higher than that of the TLD algorithm. Since the SORT method uses fast kernel operations to derive the response results, it is more real-time. Nevertheless, the chart shows that the tracking performance of the SORT and SAMF algorithms is worse, indicating that the multi-target tracking's accuracy and precision are impacted by changes in the search scale.

ETH-Linthescher video frames are captured with a fixed diagonal camera, and the scale of pedestrians changes as they approach the lens, accompanied by slight illumination changes. Table VI presents the performance comparison results of each algorithm under the dual interference of scale changes and illumination variations in the video sequence.

Excessive changes in target scale significantly impact the algorithm's tracking accuracy and can easily result in tracking drift during actual tracking. Table VI shows the comparison of the index results of each algorithm when the target has scale changes and weak illumination changes. In the scene with scale change, the accuracy of the algorithm proposed in this chapter is 22.4% higher than that of the TLD algorithm, 18.1% higher than that of the SAMF algorithm, and 26.3% higher than that of the SORT algorithm.

The MOT16-03 video sequence contains the largest number of frames, the most severe occlusion, and the highest pedestrian density in the MOT16 dataset. It is shot in a busy street at night, where the pedestrian targets have serious occlusion problems. It is suitable to test the performance of the algorithm in the case of many pedestrian targets. The tracking comparison of each algorithm is shown in Table VII.

In the MOT16-03 video frame, all algorithms run extremely slowly due to the large number of pedestrians. The detector processes numerous irrelevant windows while detecting targets, consuming memory and significantly reducing the algorithms' running speed. Due to the strong similarity and too close distance between each target, the MOTA and MOTP index values of the algorithm in the quantitative analysis results are low, but the algorithm in this chapter adaptively updates the tracker template to ensure that the tracking accuracy and precision are still higher than other algorithms.

MOT16-08 video sequence captured on a mall street during daytime. It also has disturbances such as scale variation and repeated occlusion of the target, Table VIII gives the performance comparison results of each algorithm under large scale variation and occlusion disturbances.

Compared with other tracking algorithms, the algorithm proposed in this paper can detect the target better according to the original detection module. In the occlusion interference scene, because the SAMF-TLD algorithm introduces occlusion judgment, the accuracy is higher than that of the comparison algorithm. It is 17.5% better than TLD algorithm, 10.3% better than SAMF algorithm, and 11.6% better than SORT algorithm. Also, FN data results are better than other algorithms. However, the real-time performance of the algorithm is worse than the SORT because the detection module needs to traverse the image block.

The algorithm proposed in this paper uses integrated classifiers within a cascade structure to redetect targets when they are occluded or mutually occlude each other, thereby reducing the frequency of target identity switches during tracking. This section analyzes the algorithm qualitatively and quantitatively in MOT-15 Test and MOT16 Test datasets respectively. The qualitative analysis results show the occlusion-resistant SAMF-TLD algorithm can continuously maintain stable tracking despite illumination variations of the target, longer periods of occlusion, and changes in the scale of the target, and does not produce tracking drift as other algorithms do. The quantitative analysis results indicate the superiority of the proposed algorithm over the TLD algorithm, the SAMF algorithm, and the SORT algorithm in terms of tracking accuracy, particularly in scenarios involving illumination variations, large scale variations, and occlusion of the target. The algorithm proposed in this chapter improves target tracking and ensures the robustness of multi-target tracking in complex scenarios, including scale changes, occlusion, and illumination variations.

TABLE V
COMPARISON OF TRACKING PERFORMANCE OF ALGORITHMS UNDER VARYING TARGET ILLUMINATION

Method	MOTA	MOTP	MT	ML	FP	FN	IDS	Runtime(s)
TLD	50.4%	71.3%	15.1%	45.8%	2496	9315	879	0.9
SAMF	46.7%	31.8%	17.2%	39.4%	3867	21340	891	1.6
SORT	31.6%	23.8%	24.3%	59.6%	3028	20378	1413	42.0
SAMF-TLD	**61.8%**	**79.5%**	**31.9%**	**24.7%**	**2103**	**7567**	**453**	1.3

TABLE VI
COMPARISON OF TRACKING PERFORMANCE OF ALGORITHMS UNDER LARGE TARGET SCALE AND ILLUMINATION VARIATION

Method	MOTA	MOTP	MT	ML	FP	FN	IDS	Runtime(s)
TLD	41.3%	66.4%	16.6%	51.2%	8823	64325	873	0.6
SAMF	45.6%	33.5%	18.6%	34.2%	5631	15622	642	1.4
SORT	37.4%	54.3%	23.1%	49.8%	8432	38433	1123	38.4
SAMF-TLD	**63.7%**	**71.4%**	**40.4%**	**28.6%**	7845	88125	**553**	1.1

TABLE VII
COMPARISON OF TRACKING PERFORMANCE OF ALGORITHMS UNDER A LARGE NUMBER OF TARGETS

Method	MOTA	MOTP	MT	ML	FP	FN	IDS	Runtime(s)
TLD	38.7%	54.6%	28.3%	50.2%	9213	88421	3321	0.2
SAMF	29.1%	48.4%	6.7%	61.4%	8027	67919	2283	0.9
SORT	23.6%	55.2%	25.5%	70.4%	11923	99275	8892	33.2
SAMF-TLD	**59.4%**	**63.8%**	**29.1%**	**44.5%**	**6638**	79487	**1537**	0.4

TABLE VIII
COMPARISON OF TRACKING PERFORMANCE AT LARGER SCALES AND IN THE PRESENCE OF OCCLUSION INTERFERENCE

Method	MOTA	MOTP	MT	ML	FP	FN	IDS	Runtime(s)
TLD	53.9%	76.4%	18.4%	45.8%	6642	23490	557	1.0
SAMF	61.1%	79.0%	45.8%	54.2%	5668	11790	433	2.1
SORT	59.8%	79.6%	25.5%	22.7%	8698	43730	1558	54.0
SAMF-TLD	**71.4%**	**80.1%**	**51.8%**	26.6%	**4452**	**10279**	**317**	1.1

V. CONCLUSION

This paper makes contributions to the problems of scale variation and occlusion during multi-target tracking of pedestrians. Although existing methods improve tracking accuracy for pedestrians, they are unable to address the challenges of target scale variation and occlusion. In view of this, the SAMF-TLD algorithm combined with the anti-obscuration SAMF algorithm is proposed, and multi-threaded programming is used to track multiple targets. The performance improvement of the algorithm is implemented. The scenarios of qualitatively analyzing this paper's algorithm in the 2DMOT-15 and MOT16 datasets and quantitatively comparing it with other algorithms. In contrast, the algorithm in this paper improves the precision and accuracy of target tracking. It ensures robust multi-target tracking in complex scenarios such as target scale variation and occlusion. The future research will focus on enhancing the detection module to improve the algorithm's real-time performance.

REFERENCES

[1] G. Wang, M. Song, and J. N. Hwang. "Recent Advances in Embedding Methods for Multi-Object Tracking: A Survey," *arXiv-preprint https://arxiv.org/abs/2205.1076*, 2022.

[2] Z. Shi, C. Sun, Q. Cao, Z. Wang, and Q. Fan. "Residual attention-based tracking-by-detection network with attention driven data augmentation," *Journal of Visual Communication and Image Representation*, vol. 80, pp. 103312, 2021.

[3] C. Feichtenhofer, A. Pinz, and A. Zisserman. "Detect to track and track to detect," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3038-3046, 2017.

[4] H. Liu, C. Li, J. An, G. Wei, and J. Ren. "Multiple object tracking based on kernelized correlation filter," *Laser & Optoelectronics Progress*, vol. 56, no. 12, pp. 1-8, 2019.

[5] W. Yu. "Object Tracking via Background-Aware Correlation Filter with Elliptical Search Area," *Engineering Letters*, vol. 31, no. 4, pp. 1747-1758, 2023.

[6] J. Dai, Y. Li, K. He, and J. Sun. "R-fcn: Object detection via region based fully convolutional networks," *Advances in Neural Information Processing Systems*. vol. 29, pp. 1-9, 2016.

[7] Q. Hu, H. Wu, J. Wu, J. Shen, H. Hu, Y. Chen, L. Wang, and H. Zhang. "Spatio-temporal Self-learning Object Tracking Model Based on Anti-occlusion Mechanism," *Engineering Letters*, vol. 31, no. 3, pp.1141-1150, 2023.

[8] N. Wojke, A. Bewley, and D. Paulus. "Simple online and realtime tracking with a deep association metric," *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3645-3649, 2017.

[9] Z. Yang, P. Huang, D. He, Z. Cai, and Z. Yin. "SiamMMF: multi-modal multi-level fusion object tracking based on Siamese networks," *Machine Vision and Applications*. vol. 34. no. 7, 2023.

[10] P. Bergmann, T. Meinhardt, and L. Leal-Taixe. "Tracking without bells and whistles," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 941-951, 2019.

[11] Z. Kalal, K. Mikolajczyk, and J. Matas. "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp.1409-1422, 2011.

[12] 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 2014.

[13] J. Wen, H. Chu, Z. Lai, T. Xu, and L. Shen. “Enhanced robust spatial feature selection and correlation filter learning for UAV tracking,” *Neural Networks*, vol. 161, pp. 39-54, 2023.
[14] Y. Liu, N. Zhang, H. Liu, J. Tian, and T. Tian. “Downward-Looking Ship Target Tracking Based on Rotated DSST Algorithm,” *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*, pp. 6418-6421, 2023.
[15] E. Gundogdu, and A. A. Alatan. “Good features to correlate for visual tracking,” *IEEE Trans Image Process*, vol. 27, pp. 2526–2540, 2018.
[16] M. Danelljan, G. Häger, F. Khan, and M. Felsberg. “Accurate scale estimation for robust visual tracking,” *British Machine Vision Conference*, pp. 1-5, 2014.
[17] N. Dalal, and B. Triggs. “Histograms of oriented gradients for human detection,” *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 886-893, 2005.
[18] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer. “Adaptive color attributes for real-time visual tracking,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1090-1097, 2014.
[19] M. Danelljan, G. Häger, F. Khan, and M. Felsberg. “Accurate scale estimation for robust visual tracking,” *British Machine Vision Conference*, pp. 1-12, 2014.
[20] Z. Kalal, J. Matas, and K. Mikolajczyk. “Pn learning: Bootstrapping binary classifiers by structural constraints,” *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 49-56, 2010.
[21] Z. Kalal, J. Matas, and K. Mikolajczyk. “Online learning of robust object detectors during unstable tracking,” *2009 IEEE 12th International Conference on Computer Vision Workshops*, pp. 1417-1424, 2009.
[22] Z. Kalal, K. Mikolajczyk, and J. Matas. “Face-tld: Tracking-learning-detection applied to faces,” *2010 IEEE International Conference on Image Processing*, pp. 3789-3792, 2010.
[23] S. N. Sharma, A. Khachane, and D. Motwani. “Multi-object tracking using TLD framework,” *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 1766-1769, 2016.
[24] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. “Fast keypoint recognition using random ferns,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*. vol. 32, no. 3, pp. 448-461, 2009.
[25] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof. “On-line random forests,” *2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV)*, pp. 1393-1400, 2009.
[26] Lindenbaum. M, Markovitch. S, and Rusakov. D. “Selective sampling for nearest neighbor classifiers,” *Machine learning*, vol 54, pp.125-152, 2004.
[27] T. Xu, C. Huang, Q. He, G. Guan, and Y. Zhang. “An improved TLD target tracking algorithm,” *2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 2051-2055, 2016.
[28] J. S. Pérez, N. M. López, and A. S. de la Nuez. “Robust optical flow estimation,” *Image Processing On Line*, vol. 3, pp. 252-270, 2013.
[29] T. Senst, V. Eiselein, and T. Sikora. “Robust local optical flow for feature tracking,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 9, pp. 1377-1387, 2012.
[30] L. Xue, Z. Wang, and Y. Chen. “Multi-target tracking algorithm based on TLD under dynamic background,” *International Journal of Hybrid Information Technology*, vol. 8, no. 7, pp. 267-276, 2015.
[31] M. Wang, Y. Liu, and Z. Huang. “Large margin object tracking with circulant feature maps,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4021-4029, 2017.
[32] H. Mostafavi, A. Sloutsky, and A. Jeung. “Detection and localization of radiotherapy targets by template matching,” *2012 Annual Inter-national Conference of the IEEE Engineering in Medicine and Bi-ology Society*, pp. 6023-6027, 2012.
[33] A. Lukezic, T. Vojir, L. ˇCehovin Zajc, J. Matas, and M. Kristan. “Discriminative correlation filter with channel and spatial reliability,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6309-6318, 2017.
[34] The Multiple Object Tracking Benchmark. https://motchallenge.net/.
[35] Y. Li, and J. Zhu. “A scale adaptive kernel correlation filter tracker with feature integration,” *Computer Vision-ECCV*, pp. 254-265, 2015.