

Research on tracking method of moving object under the dynamic background

Jie Wang*, Xin He, Jiahui Yu, and Jianyu Wang

Nanjing University of Science and Technology, School of Automation, Nanjing, China

Abstract. In order to solve the problem that the traditional moving target tracking algorithm is vulnerable to interference and difficult to accurately track the target for a long time, we first introduce the common target tracking algorithm under the dynamic background, and then we determine the KCF algorithm which has better tracking effect and real-time performance. Then, aiming at the problem of target box offset in the longtime tracking, a tracking algorithm combining Kalman filter algorithm and KCF algorithm is proposed. Finally, the actual effect proves to be effective.

Keywords. Moving target tracking algorithm, Kalman filter algorithm, KCF algorithm.

1 Introduction

Moving target tracking is to use the motion information extracted from target detection to find the position matching with the target features in the continuous frame, and finally obtain the complete motion trajectory of the moving target. The main problems of target tracking include appearance deformation, illumination change, fast motion blur, and so on.

2 Common tracking algorithms

2.1 Meanshift algorithm

Meanshift algorithm is easily affected by the surrounding environment, such as light and occlusion, which leads to the tracking loss. In addition, when the moving target and background cannot be segmenting accurately, the background will be regarded as the target.

2.2 Camshift algorithm

Camshift is an improved algorithm based on Meanshift. The basic idea is to use Meanshift algorithm to find the optimal iteration result for each frame of image in the video, and then take the tracking result of each frame as the initial value of the tracking iteration in the next frame of image, and automatically adjust the window size during the whole tracking process.

* Corresponding author: 2833716006@qq.com

Compared with Meanshift algorithm, Camshift algorithm has certain superiority in the direction of moving target tracking, but after tracking for certain time, with large rotational deformation occurs, target information changes, regional matching value becomes higher, and then the target frame changes, which leads to tracking failure.

2.3 TLD algorithm

TLD algorithm is a single target tracking method for a long time. It complements and restricts the detection method and tracking method to determine the tracking result together. At the same time, learning module is added to constantly update the sample information, so as to make the tracking effect more stable and reliable. TLD algorithm can be divided into three modules, detection module, tracking module, learning module.

2.3.1 Detection module

Scan the input images through a scanning window, and then determine whether there is a target in each image. A cascade classifier is used to determine whether there is a target in the input image block, which is divided into three parts: variance classifier, ensemble classifier and nearest neighbour classifier. The variance classifier eliminates all image blocks whose grey value variance is less than 50% of the image blocks in the initial tracking box. The grey value variance can be expressed as:

$$E(p^2) - E^2(p) \tag{1}$$

The ensemble classifier is composed of independent basic classifiers. Each classifier performs a group pixel comparison on images to generate a binary code. Each classifier has a tree structure, and each tree corresponds to a posterior probability distribution of binary code, which can be expressed as:

$$P_i(y|x) = \frac{P}{P+N} \tag{2}$$

i represents i th basic classifier, P and N respectively represents positive and negative samples corresponding to eigenvalues x in the training and updating process. After passing the basic classifier, n posterior probability values are generated, and then the integrated classifier identifies the image with an average posterior probability greater than 0.5 as the target. The nearest neighbour classifier calculates the relative similarity of the new sample, S^r . If it is greater than the threshold value, it will be considered as a positive sample.

2.3.2 Tracking module

X then calculates the position and scale of the target box in frame $T + 1$ according to the coordinate changes of these points. Wherein, Fb error is defined as: starting from the initial position $x(t)$ at time t to track the position $x(t + p)$ generating time $t + p$, and then reverse tracking from position $x(t + p)$ to generate the predicted position $x(t)$ at time t , and denote the Euclidean distance between the initial position $x(t)$ and the predicted position $x(t)$ as the error of the tracker at time t .

2.3.3 Integrated modules

Tracking module and detection module synchronous running independently, each output forecast target box, can lead to the result is not the only, so comprehensive device first clustering, the results of the detector and then according to the result of detector according to class and tracker tracking results of confidence, for both weighted as the final output, so as to achieve the most tracking effect.

2.3.4 Learning module

Its task is to initialize the detector in the first frame, and then update the detector with *PN* learning algorithm in the running process.

2.4 KCF algorithm

KCF algorithm, also known as kernel correlation filtering algorithm, is an improved algorithm on the basis of CSK (Exploring the Circulant Structure of Tracking By-Detection with Kernels). KCF algorithm uses the information of the previous image frame and the current image frame to train a target detector in the process of tracking, and then uses the target detector to detect the next frame of image, carries out correlation calculation, and determines whether it is the background or the moving target, and finally updates the target detector with new results. When the target detector is trained, the area where the moving target is selected as the positive sample, and the area around the target is selected as the negative sample. Set the training sample set as (x_i, y_i) , then the linear regression function is:

$$f(x_i) = \omega^T x_i \quad (3)$$

where, ω is the column vector representing the weight coefficient, which can be solved by the least square method. x_i is the distance between the calculated label $f(x_i)$ of minimizing the sampled data and the real label y_i of the real target position of the next frame, that is, the more the calculated label resembles the real label, the closer the position of the next frame is found to its real position,

$$\min_{\omega} \sum_i (f(x_i) - y_i)^2 + \lambda \|\omega\|^2 \quad (4)$$

where, λ is used to control the structural complexity of the system, and Equation (5) is converted into matrix form, and the following can be obtained:

$$\min_{\omega} \|\mathbf{X}\omega - \mathbf{y}\|^2 + \lambda \|\omega\|^2 \quad (5)$$

where, each row of $\mathbf{X} = [x_1, x_2, \dots, x_n]^T$ represents a vector, \mathbf{y} is a column vector. If the derivative is equal to 0, we can get:

$$\omega = (\mathbf{X}^T \mathbf{X} + \lambda)^{-1} \mathbf{X}^T \mathbf{y} \quad (6)$$

Since it involves calculations in the Fourier domain and involves complex matrices, Equation (5) is transformed into:

$$\omega = (\mathbf{X}^H \mathbf{X} + \lambda)^{-1} \mathbf{X}^H \mathbf{y} \quad (7)$$

Because in the KCF algorithm, the target sample gets the training sample of the trainer through cyclic shift, and in the Fourier space, the cyclic matrix has the diagonalization property, namely:

$$\mathbf{X} = F \text{diag}(\hat{\mathbf{x}}) F^H \tag{8}$$

$$\hat{\mathbf{x}} = F(\mathbf{x}) = \sqrt{n} \mathbf{F} \tag{9}$$

where, \mathbf{x} represents the value of x vector after Fourier transform, \mathbf{F} is the Fourier matrix, and the expression is:

$$\mathbf{F} = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & \omega & \dots & \omega^{n-2} & \omega^{n-1} \\ 1 & \omega^2 & \dots & \omega^{2(n-2)} & \omega^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & \omega^{n-1} & \dots & \omega^{(n-1)(n-2)} & \omega^{(n-1)^2} \end{bmatrix} \tag{10}$$

Substitute Equation (2-8) into the regression equation to simplify and obtain:

$$\omega = F \text{diag} \left(\frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \theta \hat{\mathbf{x}} + \lambda} \right) F^H \mathbf{y} \tag{11}$$

The Fourier variation of both sides can be obtained as follows:

$$\omega = \frac{\hat{\mathbf{x}} \theta \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \theta \hat{\mathbf{x}} + \lambda} \tag{12}$$

In linear problems, the objective function can be substituted into ridge regression equation. However, in reality, most of them are nonlinear problems. At this time, the concept of high-dimensional solution and kernel function can be introduced, and the kernel function can be introduced to simplify parameter training. In high-dimensional space, the nonlinear problem can be transformed into a linear problem.

$$\mathbf{w} = \sum_i \alpha_i \phi(\mathbf{x}_i) \tag{13}$$

where, $\phi(\mathbf{x}_i)$ is the function which is mapped to the high space, then the objective function can be expressed as:

$$f(\mathbf{z}) = \mathbf{w}^T \mathbf{z} = \sum_{i=1}^n \alpha_i \mathbf{k}(\mathbf{z}, \mathbf{x}_i) \tag{14}$$

where, \mathbf{k} represents the kernel function, defined as follows:

$$\mathbf{k}(\mathbf{x}, \mathbf{x}') = \phi^T(\mathbf{x}) \phi(\mathbf{x}') \tag{15}$$

The kernel function is introduced to simplify parameter training:

$$\alpha = (\mathbf{k} + \lambda \mathbf{I})^{-1} \mathbf{y} \tag{16}$$

where, \mathbf{k} represents the kernel matrix, and the above equation can be simplified by using the diagonality of the cyclic matrix:

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

where, $\hat{\mathbf{k}}^{xx}$ is the value of the Fourier change of the elements in the \mathbf{k} matrix.

The new image frame can then be detected by multiplying the correlation and regression coefficients calculated by the filter. KCF algorithm uses the rotation matrix to collect the samples, train the classifier, increase the training samples by translation, and improve the accuracy. The inverse problem of matrix is transformed into discrete Fourier domain, so that the process of inverse matrix is avoided in the algorithm, and the calculation speed of ridge regression is improved. In the aspect of training, the cyclic matrix property is used to train in the frequency domain. In terms of acceleration of kernel regression, the kernel function can also be transformed into the frequency domain for training and detection, which greatly improves the speed. The tracking accuracy and running speed are significantly improved, so that the algorithm can meet the real-time requirements.

3 Tracking algorithm combining Kalman and KCF algorithm

For KCF target tracking algorithm, in the process of tracking, because the size of the target box is set, once the moving target and the target box drift, if it can't be adjusted in time, it may lead to the continued drift of the target box. Therefore, a tracking algorithm combining Kalman filter algorithm and KCF algorithm is proposed. KCF algorithm is firstly used for target tracking, and then we use the prediction of Kalman filtering algorithm to detect moving target in each frame of the image, and we can determine the movement direction of the moving target, finally we can receive the tracking image and maintain template matching, compare the result of the match and the matching threshold, Then the Kalman filter and moving target template are updated for tracking.

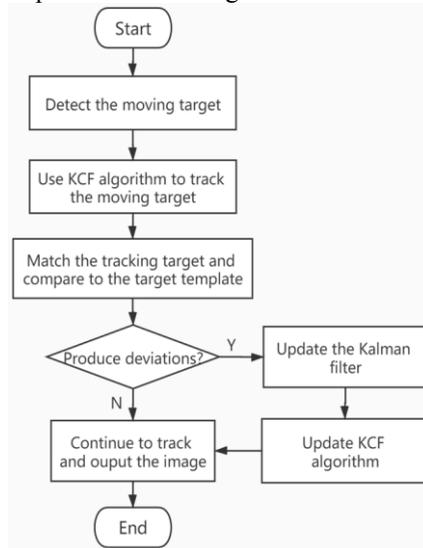


Fig. 1. Flowchart of moving target tracking algorithm.

The basic idea of Kalman filter algorithm is to realize the mutual iteration of prediction, infer the current state through the previous state, and then update according to the deviation.

Let the state and observation equation of the system be:

$$\mathbf{X}_k = \mathbf{A}\mathbf{X}_{k-1} + \mathbf{B}\mathbf{U}_k + \mathbf{W}_k \quad (18)$$

$$\mathbf{Z}_k = \mathbf{H}\mathbf{X}_k + \mathbf{V}_k \quad (19)$$

where, \mathbf{X}_k represents the system state at time k ; \mathbf{A} is the state transfer matrix, representing how to predict the current state from the previous state; \mathbf{B} is the control input matrix,

representing how the control quantity is acted on the current state; \mathbf{W}_k is the system process noise, and the covariance is \mathbf{Q} . \mathbf{Z}_k is the measured value of the system at time; \mathbf{H} is the observation matrix of the system; \mathbf{V}_k is the measurement noise of the system and the covariance is \mathbf{R} .

Constant prediction, update, re-prediction and re-update in the whole process of Kalman filtering can generally be divided into seven steps to complete, with seven formulas to achieve.

$$\mathbf{X}_k = \mathbf{A}\mathbf{X}_{k-1} + \mathbf{B}\mathbf{U}_k \tag{20}$$

$$\mathbf{P}_k = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \tag{21}$$

$$\mathbf{Y}_k = \mathbf{Z}_k - \mathbf{H}\mathbf{X}_k \tag{22}$$

$$\mathbf{S}_k = \mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R} \tag{23}$$

$$\mathbf{K}g_k = \mathbf{P}_k\mathbf{H}^T/\mathbf{S}_k \tag{24}$$

$$\hat{\mathbf{X}}_k = \mathbf{X}_k + \mathbf{K}g_k\mathbf{Y}_k \tag{25}$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}g_k\mathbf{H})\mathbf{P}_k \tag{26}$$

The above formula can be divided into three parts. In the first part, Equation (3-2) and Equation (3-3) are used to make predictions. Equation (3-4) $k - 1$ is used to calculate the predicted value of the state at time k , and Equation (3-5) is used to calculate the predicted value of the covariance corresponding to state vector \mathbf{X}_k . In the second part, the gain is calculated using Equation (3-6). In the third part, Equations (3-7) and (3-8) are used to update the system state at time k . Equation (3-7) is used to calculate the optimal value of the system state at time k . Equation (3-7) is used to calculate the covariance corresponding to the optimal result of the system at time k .

4 Experimental verification

4.1 Comparison between TLD and KCF algorithms

Both TLD and KCF algorithms can track moving targets more accurately. In order to improve the accuracy and real-time performance of the system, this paper uses frame rate per second (FPS) and accuracy as evaluation criteria to compare the two algorithms. The data annotation software Labellmg was used to manually mark the centre point, and the threshold value was set as 20 pixels. The FPS and accuracy of the two were shown in the figures below.

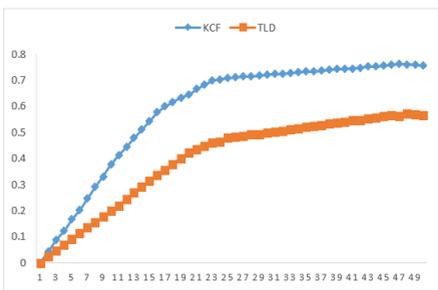


Fig. 2. Algorithm Accuracy Line Graph Figure.

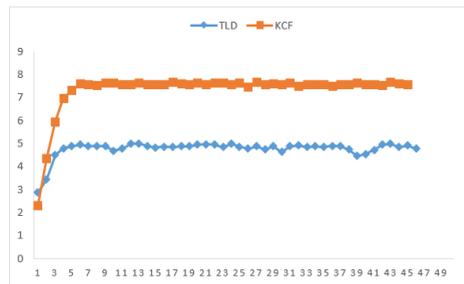


Fig. 3. FPS Line Graph.

4.2 Comparison of KCF algorithm before and improved

It can be seen that the accuracy and FPS of KCF algorithm are higher than that of TLD algorithm, the adoption of KCF algorithm is closer to real-time tracking.

By comparing the tracking renderings of the original KCF algorithm and the improved, the tracking effect is significantly improved, the matching degree between the target box and the moving target is better, and the drifting problem is also significantly relieved. Although the average frame rate has decreased, its accuracy rate has increased significantly. Therefore, the tracking algorithm combining Kalman filter and KCF algorithm has better tracking effect.

Table 1. Average frame rate and accuracy of KCF and improved KCF algorithm.

| | KCF algorithm | the improved KCF algorithm |
|-------------|---------------|----------------------------|
| Average FPS | 7.3471 | 6.9767 |
| Accuracy | 0.732 | 0.883 |

5 Summary

In this paper, we mainly compare and analyse the principle and implementation of the four tracking algorithms under the dynamic background. In consideration of tracking effect and real-time performance, the KCF algorithm is finally determined. Then, aiming at the drifting problem of moving target and target frame, a tracking algorithm combining Kalman filter algorithm and KCF algorithm is proposed. Finally, the experimental results prove that it has been improved greatly in the problem of target frame drift.

References

1. Liu Yang, Review of moving target detection technology [J]. Science, Technology and Economy Guide, 27(17):183(2019).
2. Liu Rong, Jin Guowei Moving Target Detection and Tracking Based on Background Difference and Optical Flow Method [J], Modern Manufacturing Technology and Equipment, (02):8-11(2015).
3. Zheng Jian, Research on video target tracking algorithm based on Meanshift [D], Lanzhou University of Technology, (2018).
4. Tian Ruhui, Improved video tracking algorithm based on Camshift [J], Electronic Production, (15):47-50(2020).