PAPER • OPEN ACCESS

A scale transform and deformation target tracking algorithm based on correlation filtering

View the article online for updates and enhancements.

A scale transform and deformation target tracking algorithm based on correlation filtering

Cong Hu[1,2], Wei Xia[1,2], Peng Chen[1,2], Shun Huang[1,2], Hui Guo[1,2], Xing Jin[1,2*]

[1]School of Automation, China University of Geosciences, Wuhan 430074, China

[2]Hubeikey laboratory of Advanced Control and Intelligent Automation for Complex System, Wuhan, 430074, China

jinxing@cug.edu.cn

Abstract. In order to improve the robustness of the correlation filtering (CF) tracking algorithm and overcome the problem that the traditional correlation filtering method can not deal with the target's scale and deformation change, a feature-adaptive scale adaptive correlation filter tracking algorithm is proposed. Firstly, the target is characterized; then the output filter is calculated by using the correlation filter; finally, the image blocks of different scales are intercepted from the target position of the current frame, and the optimal estimation of the target scale is obtained by the adaptive scale pool model. The experiment selects multiple video sequences for testing and compares the proposed algorithm with other target tracking methods. The experimental results show that the average performance is better than the comparison method.

1. Introduction

Target tracking is very important in the field of computer vision. In recent years, it has developed rapidly. With the influx of high-quality algorithms in various fields, traditional target tracking algorithms such as meanshift tracking, particle filter tracking and sparse matrix are gradually hidden, and related filter tracking has gradually become mainstream.

Bolme first proposed the tracking algorithm based on correlation filter MOSSE(Minimum Output Sum of Squared Error filter)[1]. He used the simplest gray feature and the minimum input square error. The speed is almost 615FPS, and the running precision is also in the middle of the same period. Subsequently, Henriques JF successively proposed KCF(High-speed tracking with kernelized correlation filters)[2], which integrated the ridge regression, kernel function, approximate dense sampling of cyclic shift and multi-channel HOG features into the correlation filter. Although the speed of the algorithm decreased, the precision has made a breakthrough. However, the KCF algorithm does not perform well for target tracking of scale transformations and deformation. When the target scale is reduced, the KCF algorithm collects a lot of useless background information, so that the tracking object becomes the background inside the detection frame. When the target scale becomes larger, the KCF algorithm collects a small portion of the information of the target, so that the tracking object becomes a part of its target. And when the target is deformed, the single feature of KCF algorithm cannot be well tracked.

Aiming at the above problems, based on the KCF algorithm, this paper proposes a feature-adaptive scale adaptive correlation filter tracking method. Firstly, the original grayscale features are changed to the fusion features based on HOG(Histograms of oriented gradients)[3] and

HSV(Hue,Saturation,Value)[4], and the tracking accuracy of the deformation target is improved. Then the scale pool method is used to adaptively scale. The algorithm framework refers to KCF and integrates on it. The characteristics and training of the adaptive scale pool are improved, and the tracking accuracy of the easy-to-deform target is improved on the basis of the guaranteed speed.

2.   Proposed method

2.1. KCF algorithm principle

The KCF algorithm is a kernel correlation filter tracking method, and it is one of the more efficient algorithms for recent tracking. Most of the tracking algorithms of the KCF algorithm are similar, and the target detection is performed first and then the filter model training (update) is performed. The method is mainly to first train a target initial position model, then detect the next frame target position, determine the target position by calculating the maximum value of the response, and finally train the target new model according to the target new position. Its tracking flow chart is shown in Fig.1.
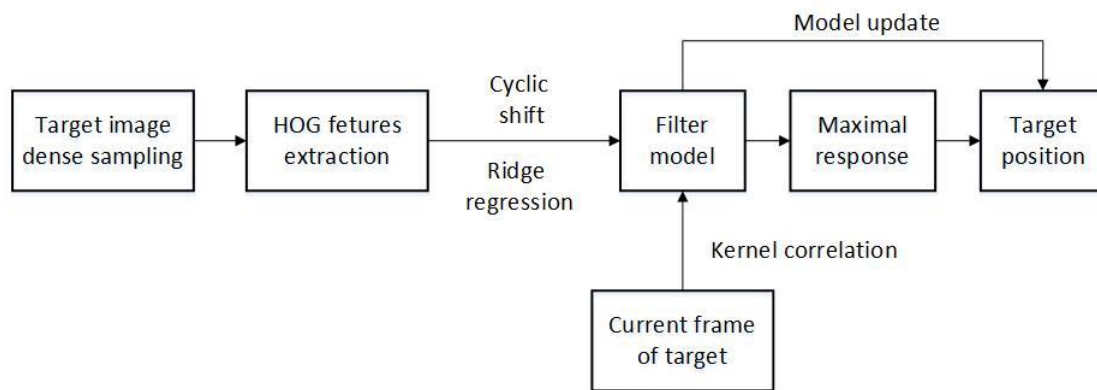

Fig.1 KCF algorithm flow chart

The KCF tracking algorithm firstly collects the training image by cyclically shifting the obtained target center position and the window of the extended range within the previous frame image. If the training image block obtained is x, then the corresponding tag (regression) function after feature extraction is y. Let the linear classifier model be $f(x) = <w, \psi(x)>$. Under the condition of regularized least squares and introducing a kernel function,

$$\omega = \min_{\omega} \sum_{i} | f(x_i) - y_i |^2 + \lambda || \omega ||$$
(1)

Here, the ω is a classifier coefficient, and the $\psi(\bullet)$ denotes the mapping function of the kernel from the original space to the Hilbert feature space, and define the kernel space inner product $k(x, x') = <\psi(x), \psi(x')>; \lambda \geq 0$ as a regularization parameter. Using the properties of the cyclic matrix and the discrete Fourier transform, the optimal solution $\omega = \sum \alpha \psi(x)$ is obtained, where the coefficient α can be expressed as the following formula:

$$\alpha = F^{-1}\{ \frac{F(y)}{F[k(x,x)][k(x,x)] + \lambda} \}$$
(2)

Where F and F^{-1} represent a discrete Fourier transform and an inverse discrete Fourier transform. The core used by KCF is a Gaussian kernel:

$$k^{xx'} = k(x, x') = \exp(-\frac{|x - x'|}{\sigma^2})$$
(3)

Here, σ is the bandwidth parameter in the Gaussian kernel function. For all cyclically shifted blocks, the solution to the $k^{xx'}$ is given by the formula below

$$k^{xx'} = \exp(-\frac{1}{\sigma^2}(\|x\|^2 + \|x'\|^2) - 2F^{-1}(F(x) \odot F*(x'))) \tag{4}$$

Where $\odot$ represents the dot product of the element.

The tracking process of the KCF is based on the result of the training of the above-mentioned frame classifier. The image block z of the same size window is intercepted in the current frame, and then the output response value of the image block in the classifier is calculated, and the position with the largest response value is tracked. The position of the target in the current frame. The calculation of the response value is expressed as

$$\hat{y} = F^{-1}[F(k^{z\hat{x}}) \odot F(\hat{a})] \tag{5}$$

Where $\hat{y}$ is the output response value; $\hat{x}$ and $\hat{a}$ respectively represent the learned appearance template and the target appearance model parameters.

2.2. HOG feature

The HOG feature is a feature descriptor used in the field of computer vision for object detection. It composes and counts the gradient direction histogram of the local region of the image to form the corresponding feature. The implementation process of the HOG feature extraction algorithm for the target image is shown in the figure below.
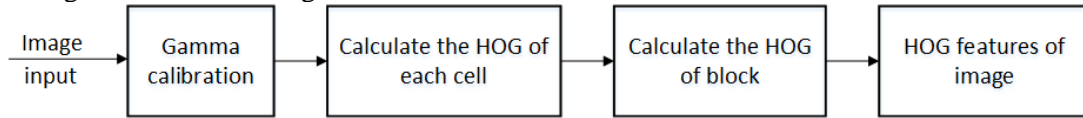

Fig.2 HOG feature extraction

2.3. HSV feature

HSV is a color space created by A. R. Smith in 1978 based on the intuitive nature of color, also known as the Hexcone Model. Since HSV is a relatively straightforward color model, most of the image recognition blocks use the HSV color space. The extraction process of HSV features refers to the method of HOG features.

2.4. Feature fusion

We add HSV features based on the unique characteristics of traditional KCF's feature. This feature fusion uses a simple vector addition. The formula is as follows:

$$k^{xx'} = \exp(-\frac{1}{\sigma^2}(\|x_c\|^2 + \|x_c'\|^2) - 2F^{-1}(\hat{x}_c \odot \hat{x}_c'*)) \tag{6}$$

Where x is a separate feature extracted from the traditional KCF algorithm, while x_c is a mixture of two features, namely the fusion of HOG feature and HSV feature.

2.5. Research on adaptive scale pool

The KCF algorithm uses a single scale, which causes the tracking accuracy to drop when the target is deformed or occluded. The DSST(Discriminative scale space tracking)[5] and SAMF(A scale adaptive kernel correlation filter tracker with feature integration)[6] algorithms have successively proposed the concept of scale filtering and scale pool, which is to find the maximum response of a scale range. We refer to the idea of the two algorithms and propose the concept of adaptive scale pool. The size of the target frame is adjusted by introducing an adaptive scale pool model in combination with HSV feature and HOG feature. Firstly, the scale template of the target is obtained from the tracking result of the previous frame. Secondly, the feature is fused in the adaptive scale pool to get the best response value.

The adaptive scale pool model is as follows: Calculate the corresponding Peak Side Ratio (PSR)[7] to evaluate the current frame tracking results. The PSR calculation method is as shown in Equation. This model uses this update strategy.

$$PSR = \frac{r - \mu}{\sigma} \tag{7}$$

Here, r is the maximum peak response . μ and σ are the mean and standard deviation of the sidelobe region respectively. The larger the PSR value, the tracking result value is more robust. We define the threshold as T, and when PSR > T, the filter model is updated. Otherwise we need calculate $\hat{y}_{\max} = \max\{\hat{y}_{i-1}, \hat{y}_i\}, i = \frac{n+1}{2}$ and get the subscript index. Then update the scale pool search index i.

3. Experimental results and analysis

3.1. Experimental environment and parameter settings

The development platform used in the experiment was MATLAB R2017a, and all experiments were performed on a computer with an Intel i5-3230M CPU (3.6 GHz) and 8 GB of memory. In the experiment, the main frame parameters of the algorithm use the code default parameters provided by the KCF original author, set the scale range and add the adaptive scale pool model. In this paper, the video sequence disclosed in [8] is selected. The video sequence includes various phenomena such as background occlusion, scale change, illumination change, target rotation deformation and motion blur.

In this paper, the most commonly used two indicators are used to evaluate the performance of the tracking algorithm: center position error (CLE), distance accuracy (DP) and image processing speed (FPS).

Among them, the CLE evaluation index is the average Euclidean distance value between the target center (x_T, y_T) and the real center (x_G, y_G), that is $\sqrt{(x_T - x_G)^2 + (y_T - y_G)^2}$; the DP evaluation index refers to the percentage of CLE less than a certain threshold, and the selection threshold is 20 pixels.

3.2. Algorithm performance comparison

In order to verify the performance of our algorithm, this paper used 50 image sequences of OBT-2013, compared its tracking accuracy and speed with the traditional KCF algorithm, and finally evaluated the pros and cons of the two through evaluation criteria.

We obtained Fig.3 and Fig.4 through testing. Table 1 showed the tracking accuracy of each algorithm in (a), (c) and (d) in Fig.4. Fig.3 showed the tracking effect of the main four algorithms in some image sequences.(a) and (b) in Fig.4 showed the iterative tracking accuracy and success rate of each algorithm for all sequences. In (c) and (d) of Fig.4, the correlation curves of the respective algorithms and the correlation curves of the deformed image sequences are shown. It can be found that the algorithm proposed by our algorithm is significantly improved compared with the KCF algorithm in the case of target scale transformation, and slightly worse than the SAMF algorithm; And the algorithm was optimal in the case of faster target deformation.

Table 1. Plots of precision

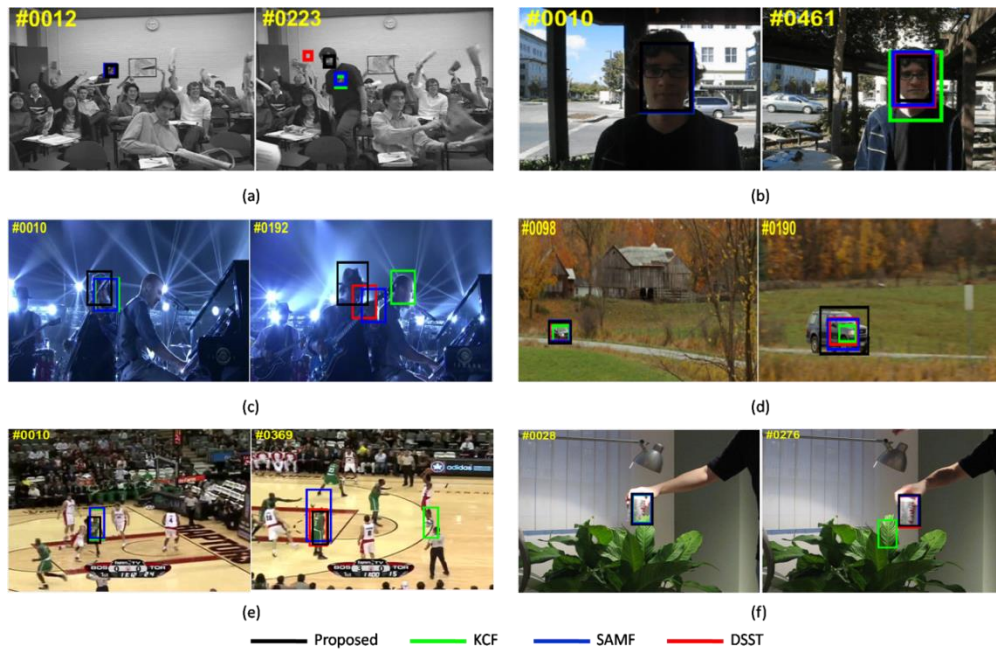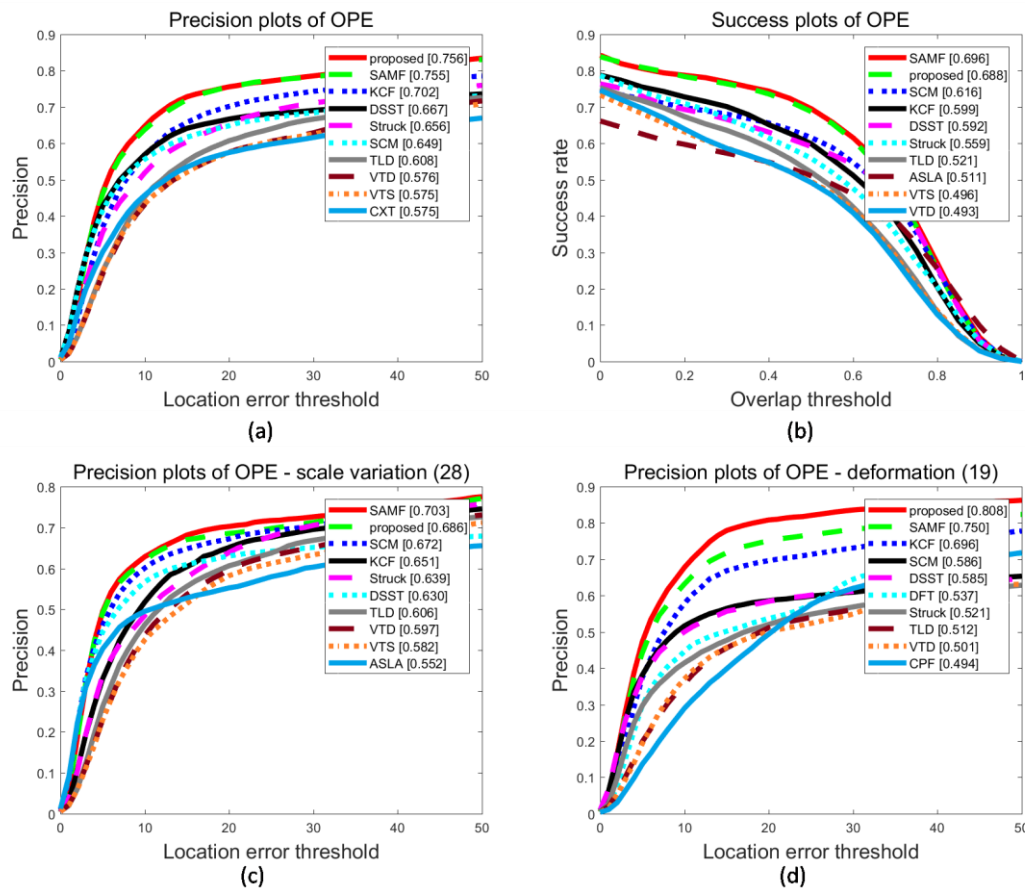| The Plot	Proposed	SAMF	DSST	KCF	Struck
Average	**0.756**	0.755	0.667	0.702	0.656
Scale variation	0.686	**0.703**	0.630	0.651	0.639
deformation	**0.808**	0.750	0.585	0.696	0.521

Fig.3 Tracking results of different algorithms

Fig.4 Plots of precision

Therefore, we have verified that the feature fusion and adaptive scale pool are integrated into the KCF algorithm, and the tracking accuracy is greatly improved when the target scale changes and deforms.

4.   Conclusion

We improved the KCF algorithm from the aspects of scale adaptation and feature fusion. Firstly, the HOG feature was merged with the HSV feature to extract the more detailed features of the target. Secondly, according to the adaptive scale pool model, the maximum response value was quickly and accurately selected, and the scale was realized. Adapt to the model's update strategy. The algorithm in this paper uses OTB-2013 data set to test and evaluate. By verifying that the algorithm has good scale adaptability, the accuracy of tracking is improved under the premise of ensuring the computing speed. The shortcoming of this paper is that the selection and processing methods of features are not particularly good, and efforts will be made to improve in this area in subsequent research.

References
[1] Bolme D S, Beveridge J R, Draper B A, et al. Visual object tracking using adaptive correlation filters[C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010: 2544-2550.
[2] Henriques J F , Caseiro R , Martins P , et al. High-Speed Tracking with Kernelized Correlation Filters[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3):583-596.
[3] Yan G, Yu M, Yu Y, et al. Real-time vehicle detection using histograms of oriented gradients and AdaBoost classification[J]. Optik, 2016, 127(19): 7941-7951.
[4] Kolkur S, Kalbande D, Shimpi P, et al. Human skin detection using RGB, HSV and YCbCr color models[J]. arXiv preprint arXiv:1708.02694, 2017.
[5] Danelljan M, Häger G, Khan F S, et al. Discriminative scale space tracking[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(8): 1561-1575.
[6] Li Y, Zhu J. A scale adaptive kernel correlation filter tracker with feature integration[C]//European conference on computer vision. Springer, Cham, 2014: 254-265.
[7] Savvides M, Kumar B V K V, Khosla P. Face verification using correlation filters[J]. 3rd IEEE Automatic Identification Advanced Technologies, 2002: 56-61.
[8] Wu Y, Lim J, Yang M H. Online object tracking: A benchmark[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2013: 2411-2418.