Target Tracking Method Based on Correlation Filter and Particle Filter

Benbin You, Qingwei Gao, Yixiang Lu, Dong Sun

ABSTRACT

Object tracking is one of the most important tasks in computer vision, many researchers have proposed a lot of effective methods to solve the all kinds of the recent problems in object tracking. During the movement of the targets, the change of object scale may produce a lot of difficulties in object tracing problems, and it may lead to be failed in tracking. The paper proposed method is designed both correlation filter and particle filter. In order to achieve the purpose of tracking, the process of correlation filter is to learn an array of filters so that the response value obtained by the convolution of the learned filters is the largest. The particle filter guides the sample particles to the distribution mode of the target state. The samples of the particle strategy can effectively deal with the large scale variations of the target during movement. Experiment on the OTB database shows that the paper proposed method makes good performance.

KEYWORDS

Object Tracking, Correlation Filter, Particle Filter, Scale Variation.

INTRODUCTION

Target tracking is one of the difficult problems in the field of computer vision. It is applied to motion analysis, automatic driving, video monitoring and other fields[1, 2, 14, 13, 15]. In the past few years, a large number of researchers
have proposed many effective methods to solve the challenging problems in the process of target tracking, such as light and shadow change, fast motion, attitude change, etc. The biggest challenge of target tracking is that the target scale changes greatly in the process of moving, based on this challenge, it is difficult to learn robust tracker.

As the main framework of target tracking algorithm, correlation filtering has made good development in speed and robustness. Bolme et al.[4] model the process of target tracking as a learning correlation filter, and learn the filter template by finding the least mean square error. Hare et al.[12] used support vector machine (SVM) with kernel structure output to provide adaptive tracking through online learning. In literature[17], Zhang et al. combined the circular characteristics of the target template to improve the sparse based tracker. Hong et al.[18] proposed a framework of biological inspiration (muster), in which short-term processing and long-term processing cooperate with each other. In literature[16], Ma et al. introduced an online random fern classifier as a re-detection component for long-term tracking. Henriques et al.[3] put forward KCF algorithm. The algorithm uses image block shift, introduces cycle matrix, and improves the operation speed by processing the inverse of the matrix which is not easy to calculate in the loss function. The introduction of image block shift idea solves the problem of less target tracking samples, and improves the robustness of the algorithm to a certain extent. Due to the lack of consideration of the influence of the scale change of the target in the process of motion, the performance of KCF is poor. Danelljan et al.[5] proposed the DSST method, which is based on the image hog feature, and proposed the adaptive scale estimation method to improve the scale change of the target in the process of motion. Particle filter[6-8] is widely used in the field of target tracking. In order to get the best effect, the number of samples must be enough to cover the possible state. However, due to the need to evaluate each particle, dense sampling of particles usually results in a large amount of computation for tracking. Therefore, many researchers have proposed many methods to improve the sampling rate of particle filter in order to solve such a problem. The introduction of importance resampling[6] combines the previous prediction and other auxiliary observation information to give good sampling suggestions. The calculation of likelihood from coarse to fine can effectively focus on more promising particles[9].

The existing tracking algorithm does not solve the problem of scale change, in order to solve this problem, particle filter is used to solve the problem of scale change. Generally speaking, the more the number of sampling particles, the better the robustness, but this will make the calculation time greatly increased. If the sampled particles do not cover the state of the target, the predicted state of the target is not valid. In order to solve this problem, the best way is to gather the sample particles in the target state distribution area. In this paper, correlation filter and particle filter are combined organically. Particle filter can solve the
scale change of the target through the dense sampling technique[9]. Each sample particle is regarded as a sample, and a corresponding response graph is obtained by correlation filtering, in which the maximum value points to the particle. Combined with correlation filtering and particle filtering, experiments in open data sets show that the proposed method is better than the experimental comparison method.

CORRELATION FILTERING

The reason that correlation filter can be widely used in computer vision is that the calculation method of cyclic matrix is applied to the inverse process of matrix. The introduction of cyclic matrix increases a large number of negative samples at the same time, which enhances the robustness of filter. Target tracking is modeled as follows: Take the image block \( x \) with the size of \( M \times N \), and the cycle matrix \( X \) corresponding to the image block \( x \) is composed of \( x_{m,n} \), where \( (m,n) = \{0,1,...,M-1\} \times \{0,1,...,N-1\} \), extract \( K \) HOG features of the image block, and the corresponding label of each training sample \( x_{m,n} \) is Gaussian function \( y_{m,n} \). Define \( X_k \) as the \( k \)-th column of cyclic matrix \( X \), \( (k = 1,2,...,K) \). Then the optimization model is:

\[
\arg \min_{w_k} \frac{1}{2} \sum_x \left\| X_k w_k - y \right\|_F^2 + \lambda \left\| w_k \right\|_F^2 \tag{1}
\]

Where \( \left\| \cdot \right\|_F \) is \( F \) norm, \( y = [y_{0,0},...,y_{M,N}]^T \), \( \lambda \) is the regularization parameter. Because equation (1) is a convex function, its derivative is obtained and made equal to 0.

\[
w = (X^H X + \lambda I)^{-1} X^H y \tag{2}
\]

Where \( w \) is a set of filters, \( X^H \) is the conjugate transpose of matrix \( X \). The \( X \) of the above formula is a cyclic matrix, in which the first row represents the real sample, and each subsequent row is obtained by shifting an element from the previous row. According to the properties of cyclic matrix, we can know:

\[
X = Fdiag(\hat{x})F^H \tag{3}
\]
\( \hat{x} \) is the form of discrete Fourier transform of \( x \), matrix \( F \) is a coefficient matrix independent of \( x \), that is, DFT matrix. From equation (3), we can get:

\[
X^H X = F \text{diag}(x^*)F^H F \text{diag}(x)F^H
= F \text{diag}(x^* \odot x)F^H
\] (4)

When equation (4) is introduced into equation (2), it can be obtained that:

\[
w = \text{diag}(\frac{x^*}{x^* \odot x + \lambda})y
\] (5)

Equation (5) can be simply written as:

\[
w = \frac{x^* \odot y}{x^* \odot x + \lambda}
\] (6)

\( \odot \) is Hadamard product.

**PARTICLE FILTER**

Particle filter is based on the importance sampling of Bayesian sequence. It uses a set of finite weighted samples to fit the posterior distribution of state variables recursively. \( s_t, y_t \) are defined as the state quantity of the target at time \( t \) and its observation. The posterior density function \( p(s_t|y_{1:t-1}) \) of each time \( t \) can be obtained by two steps recursively, i.e. prediction and update. In the prediction stage, the state quantity of the time is predicted according to the observation value \( y_{1:t-1} = \{y_1, y_2, ..., y_{t-1}\} \) and probability density function \( p(s_t|s_{t-1}) \) of the first \( t - 1 \) time:
\[ p\left(s_t|y_{1:t-1}\right) = \int p\left(s_t|s_{t-1}\right)p\left(s_{t-1}|y_{1:t-1}\right)dS_{t-1} \quad (7) \]

In formula (7), \( p(s_t|y_{1:t-1}) \) is known at time \( t \). \( p(s_t|s_{t-1}) \) is the state prediction density function. When the observation value \( y_t \) at time \( t \) is known, the state variable at time \( t \) can be obtained by the following formula:

\[
p(s_t|y_{1:t}) = \frac{p(y_t|s_t)p(s_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (8)
\]

\( P \) in equation (8) is likelihood function. The posterior probability \( p(s_t|y_{1:t}) \) can be obtained by the approximation of \( n \) particles \( \{s_t^i\}_{i=1}^n \):

\[
p(s_t|y_{1:t}) \approx \sum_{i=1}^n \pi_t^i \delta(s_t - s_t^i) \quad (9)
\]

In equation (9), \( \delta(\cdot) \) is a Dirichlet function, and \( \pi_t^i \) is the weight of the \( i \)-th particle at time \( t \). The weight of each particle can be given as follows:

\[
\pi_t^i \propto \pi_{t-1}^i \frac{p(y_t|s_t^i)p(s_t^i|s_{t-1}^i)}{q(s_t^i|s_{t-1}^i, y_t)} \quad (10)
\]

\( q(\cdot) \) in equation (10) is the importance density function, which makes \( \pi_t^i \) proportional to \( \pi_{t-1}^i \), namely \( \pi_t^i \propto \pi_{t-1}^i p(y_t|s_t^i) \). In order to avoid the degradation of resampling algorithm, according to literature [6], the weight \( \pi_{t-1}^i \) is set to \( \pi_{t-1}^i = 1/n, \forall i \). Then we can get:

\[
\pi_t^i \propto p(y_t|s_t^i) \quad (11)
\]
The above resampling method is to derive particles according to the weight of the previous step, and all new particles will be updated through the next frame likelihood function.

**DETECTION AND TRACKING**

According to the filter \( w \) learned from the above correlation filter and the known target image block \( x \), for the search box with the size of \( M \times N \), the response value of the \( i \)-th particle can be given by the following formula:

\[
r = F^{-1}(F(W) \odot F((y'_i, \pi)))
\]  

(12)

In formula (12), \( y'_i \) is the observation value of the \( i \)-th particle at the time, \( F \) and \( F^{-1} \) are the Fourier change and its inverse transformation respectively, \( \odot \) is the Hadamard product, and formula (12) is to guide particle \( i \) according to the maximum response value \( r \) in the search area. The tracking phase can be divided into four steps. The first step is to use the probability model \( p(s_t|s_{t-1}) \) to generate particles, and then resample them. The second step is to apply the filter \( w \) obtained by learning to each particle to move it to a stable position. The third step is to calculate the response value when the filter is \( w \), which is used to determine the weight of each particle. Finally, in order to update the filter \( w \) for target tracking, a new tracking strategy similar to that in reference [3] is adopted, which only applies to the new sample \( x_k \) of the current frame to follow the new model:

\[
F\left(\overline{x}_{k}^{t}\right) = (1 - \eta)F\left(\overline{x}_{k}^{t-1}\right) + \eta F\left(\overline{x}_{k}^{t}\right)
\]  

(13)

\[
F\left(\omega_{k}^{t}\right) = (1 - \eta)F\left(\omega_{k}^{t-1}\right) + \eta F\left(\omega_{k}^{t}\right)
\]  

(14)

The \( \mu \) in formula (13) and formula (14) is the learning rate.

The detailed description of this algorithm is shown in algorithm 1:
Algorithm 1: iterative process of correlation filter and particle filter

Input: the image block with the size of $M \times N$, the corresponding label is $y$, and the number of particle samples is $n$.

Output: filter template $w$.

(1) Correlation filtering:

1. Obtain the circulant matrix of the image block with the size of $M \times N$, and calculate the filter template $w$ by formula (1) according to the corresponding label $y$.

2. The calculation of filter template $w$ needs to solve the inverse matrix and transform both sides, so it is easy to derive equation (6) and calculate according to equation (6).

(2) Particle filter:

1. $s_t$, $y_t$ is the state quantity of the target at time $t$, and its observable measurement. The trajectory of the target is estimated by (7), (8) and (9).

2. Simplified formula (10) according to reference [6].

(3) Detection and tracking:

1. According to the learned filter template $w$, the corresponding response of the $i$-th particle is calculated by equation (12).

2. Equations (13) and (14) give the iterative method for updating the new sample $x_k$.

EXPERIMENTAL ANALYSIS AND DISCUSSION

The method proposed in this paper and the experimental comparison method are all running in Intel (R) core (TM) 2 Duo CPU T7700, 8GB running memory, matlab2016. The value of parameter $\lambda$ is 0.0001, the value of $\mu$ is 0.01, and the number of particles is set to 100. The proposed method is tested in two open datasets, OTB-2013 [10] and OTB-2015 [11], which are composed of 50 and 100 video sequences respectively. In this paper, distance precision and overlap precision are used to measure the performance of the method. Distance precision measures the accuracy of a certain distance between the center position and the real ground position. Overlap precision measures the overlap rate between the tracking value and the real ground value boundary box.

In Figure 1, the target tracking method proposed in this paper performs well in two indicators of target tracking performance. Another area under the curve
(AUC) can also be used to measure target tracking performance, which increases with the increase of tracking effect. DSST ranked second got 0.5734 points due to the use of scale information. However, the proposed target tracking strategy based on correlation filter and particle filter gets 0.636 points, because the particle filter has a better effect on correcting the scale change of the target in motion.

![Figure 1. Distance accuracy chart and overlapping accuracy chart.](image)

From table I, it can be seen that CSK is the fastest, because in algorithm CSK, it only considers the brightness information of the image, and does not solve the scale problem. From the distance accuracy and overlapping accuracy of the evaluation index, it can be seen that the values of the two are obviously smaller. Compared with KCF, fps value is smaller than KCF, and the reason of running speed lower than KCF is that the main time consumed is in particle sampling. According to the distribution of particle sampling, the scale change information of the target in motion is calculated. In terms of distance accuracy, the two are equivalent, because the KCF does not consider the change of the target's scale change in the process of motion, resulting in the low overlapping
accuracy of the KCF. The method proposed in this paper aims at the scale problem in the target's motion, and the overlapping accuracy is significantly higher than that in the KCF algorithm. The improvement of the overlapping accuracy in Table I shows that it can really cope with the change of the target's scale change.

### Table I. Experimental Results.

| Algorithm | Distance accuracy | Overlap accuracy | Fps |
|-----------|-------------------|------------------|-----|
| STRUCK    | 0.623             | 0.572            | 45  |
| CSK       | 0.520             | 0.420            | 353 |
| DSST      | 0.715             | 0.634            | 30  |
| KCF       | 0.748             | 0.586            | 218 |
| Methods of this paper | 0.766 | 0.678 | 32 |

AUC in Table II represents the area under the overlapping accuracy curve. As can be seen from table II, three different particle numbers are set in this paper to explain the influence of the increase of particle number on the tracking performance. With the increase of particle number, AUC value increases in otb-2013 and otb-2015. Note that a tracker with 10 particles can get similar results with a tracker with 50 particles. These results show that the proposed method can achieve good tracking performance even with a small number of particles.

### Table II. Effect of Different Particle Numbers on Tracking Performance.

| Number of particles | 10  | 50  | 100 |
|---------------------|-----|-----|-----|
| OTB-2013            | 64.8| 65.2| 66.5|
| OTB-2015            | 61.0| 61.7| 62.3|

### Conclusion

In this paper, a target tracking method based on correlation filter and particle filter is proposed. The sampling particles are guided to the state distribution mode of the target, and the particle sampling strategy can deal with the scale...
change of the target in the process of motion well, so as to obtain strong tracking performance. The method proposed in this paper proves that the algorithm can deal with the scale change better than the comparative experimental method in the three evaluation indexes of distance precision, cross precision and area under the arc, so as to obtain a good tracking effect.

ACKNOWLEDGEMENTS

National Natural Science Foundation of China (NSFC) (61502007); Anhui Provincial Education Department natural Science Key Project(KJ2018A0012).

REFERENCES

1. YILMAZ A, JAVED O, SHAH M. Object tracking: a survey[J]. ACM Computing Surveys, 2006, 38(4): 13.

2. GAO J, ZHANG T, YANG X, et al. Deep relative tracking[J]. IEEE Transactions on Image Processing, 2017, 26(4): 1845-1858.

3. 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, 2014, 37(3): 583-596.

4. BOLME D S, BEVERIDGE J R, DRAPER B A, et al. Visual object tracking using adaptive correlation filters[C]//2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2010: 2544-2550.

5. DANELLJAN M, HAGER G, KHAN F, et al. Accurate scale estimation for robust visual tracking[C]//British Machine Vision Conference, Nottingham, September 1-5, 2014. BMVA Press, 2014.

6. ISDRD M, BLAKE A. Condensation—conditional density propagation for visual tracking[J]. International journal of computer vision, 1998, 29(1): 5-28.

7. ZHANG T, JIA K, XU C, et al. Partial occlusion handling for visual tracking via robust part matching[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1258-1265.

8. ZHANG T, LIU S, XU C, et al. Structural sparse tracking[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 150-158.

9. YANG C, DURAISSWAMI R, DAVIS L. Fast multiple object tracking via a hierarchical particle filter[C]//Tenth IEEE International Conference on Computer Vision. 2005, 1: 212-219.

10. 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.
11. Wu Y, LIM J, YANG M H. Object tracking benchmark[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1834-1848.

12. HARE S, GOLODETZ S, SAFFARI A, et al. Struck: Structured output tracking with kernels[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 38(10): 2096-2109.

13. GAO J, ZHANG T, YANG X, et al. Deep relative tracking[J]. IEEE Transactions on Image Processing, 2017, 26(4): 1845-1858.

14. YILMAZ A. JAVED O, Shah M[J]. Object tracking: A survey, 2006, 38(1): 81-93.

15. ZHOU S K, CHELLAPPA R, MOGHADDAM B. Visual tracking and recognition using appearance-adaptive models in particle filters[J]. IEEE Transactions on Image Processing, 2004, 13(11): 1491-1506.

16. ZHANG T, XU C, YANG M H. Multi-task correlation particle filter for robust object tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4335-4343.

17. Zhang T, Bibi A, Ghanem B. In defense of sparse tracking: Circulant sparse tracker[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 3880-3888.

18. HONG Z, CHEN Z, WANG C, et al. Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 749-758.