A Fast Filter Tracker against Serious Occlusion

Libo Zhang, Tiejian Luo, Yihan Sun and Lin Yang
University of Chinese Academy of Sciences
Beijing, China

Abstract—**Many tracking algorithms applied in medical image processing, such as observing the movement of cells, have a great improvement in accuracy and robustness. However, it is difficult to deal with the large area occlusion and complete occlusion. In this paper, we propose a fast scale adaptive tracking algorithm based on correlation filtering. Except tracking the change of the target scale quickly, our method can also deal with the problem of large area occlusion and the complete disappearance of the target. Compared with the outstanding scale adaptive tracking method, the proposed method demonstrates higher performances in terms of the accuracy of tracking the target and the real-time performance.**

I. Introduction

Tracking is playing an increasingly important part in medical image processing for solving some awkward problems, such as observing the movement of cells, variation of protein and flow of material. Besides, tracking algorithm can help researches save much time spent in observing the whole process[1]. Though the medical tracking has been much more advanced than before, some challenges still have to be solved, such as partial or complete occlusion, and the fast motion background clutter. Numerous existing algorithms provide inferior performance when encountered with fast scale variations and large scale occlusion[2]. In this paper, we present a novel medical tracking method using filter tracker for solving the challenging problems.

Recently, correlation filter is used in visual tracking, and has already been applied in many algorithms. Correlation filter is a scheme of tracking-by-detection working by posing the target localization as a classification problem. The decision boundary is acquired through online discriminative classifier depend on image patches from target and background. [3] shows the CSK (circulate structure with kernel) tracker introduces effective accuracy and speed while dealing with the highest speed video frames. [4] propose new model using CSK and achieve better performance, but the previous approaches are limited in particular situations. [5] provides an approach based on finding an adaptive correlation filter minimizing the output sum of squared error(MOSSE). However, both CSK and MOSSE are limited to deal with the situation of fast scale variations and large scale occlusion. Several approaches [6] get the scale variations at low frame-rates which cannot meet the real-time requirement.

In this paper, we use a novel scale adaptive kernelized correlation filter tracker. It achieves competitive performance in accuracy and speed against fast scale variations and large scale occlusion. Our experimental evaluation demonstrates that the proposed scale adaptive and multiple feature integration method achieve a significant performance comparing with the state-of-the-art approach. Our method has enhanced 23.7% and 7.9% on Median OP (Overlap Precision) and Median DP (Distance Precision), and Median CLE (Center Location Error) has reduced to 10.5 pixels. Moreover, our method successfully tracks the targets in almost 70% sequences in the benchmark [7] in total.

II. Fast Scale Adaptive Tracking Method

A. Joint Correlation Filter

Correlation filters with the ability of discrimination have been applied in many directions successfully, including target tracking, target detection, and target correction. MOSSE Algorithm proposes the filter by using the original image pixels as features. This method can track target quickly. However, the accurate degree cannot be guaranteed, especially in face of pose transformation, illumination transformation and occlusion. In this paper, we use HOG features as same as Kalman-Consensus Filtering (KCF) features to train the filter. At the same time, we extract the LBP texture features of target to train the filter. So we can take advantage of the filter that is trained by two kinds of features to achieve target joint tracking.

First of all, we give the objective function as follows:

$$f(z) = w^T z \tag{1}$$

where $f(z)$ represents the output response, w represents the filter, z is the input image. Then we use the given object to train the filter. Training set is given by:

$$T = ((x_1, y_1), (x_2, y_2), ..., (x_i, y_i)) \tag{2}$$

where x_i denotes the data sample, y_i denotes the Gaussian regression label. We use the regularized least squares [**?**] to get the filter w:

$$w = (X^H X + \lambda I)^{-1} X^H y \tag{3}$$

where $X = [\mathrm{x}_1, \mathrm{x}_2, ...\mathrm{x}_n]$, $y = [y_1, y_2, ...y_n]$. The computation of Eq.(3) is complex, the complexity is $o(n^3)$. In order to decrease the computation, we transfer it to Fourier domain for solving through circulant matrix. First of all, we give a target sample X: $\mathrm{x} = [x_1, x_2, ..., x_n]^T, n \times 1$, then we give a transformation matrix P:

$$P = \begin{bmatrix} 0 & 0 & 0 & ... & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & ... & 1 & 0 \end{bmatrix} \tag{4}$$

978-1-5090-1610-5/16/$31.00 ©2016 IEEE

Transform sample X via matrix P then we can get training samples (sample number generated from loop jump is as same as the dimension of the sample characteristic):

$$P\mathrm{x} = [x_n, x_1, ..., x_{n-1}]^T, P^2\mathrm{x} = [x_{n-1}, x_n, ..., x_{n-2}]^T \quad (5)$$

Thus we get a circulant matrix via sample X:

$$\mathrm{X} = C(\mathrm{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & ... & x_n \\ x_n & x_1 & x_2 & ... & x_{n-1} \\ x_{n-1} & x_n & x_1 & ... & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \\ x_2 & x_3 & x_4 & ... & x_1 \end{bmatrix} \quad (6)$$

$C(\mathrm{x})$ denotes the function of construct circulant matrix via sample x. The constructed circulant matrix is used for replacing the X in Eq.(3). For the constructed circulant matrix X, we can represent it as:

$$\mathrm{X} = Fdiag(\hat{\mathrm{x}})F^H \quad (7)$$

Where F denotes a DFT matrix, $\hat{\mathrm{x}}$ is X's DFT. With the format, we transfer a problem in spatial domain to Fourier domain. Take Eq.(7) into formula $w = (X^H X + \lambda I)^{-1} X^H y$, we can get:

$$X^H X = Fdiag(\hat{\mathrm{x}}^* \odot \hat{\mathrm{x}})F^H \quad (8)$$

HOG and LBP features are extracted to train different filters. By using the characteristic of cyclic matrix, we can simplify the computation, and finally get the filter by taking Eq.(8) into Eq.(3):

$$\hat{w} = \frac{\hat{x} \odot \hat{y}}{\hat{\mathrm{x}}^* \odot \hat{\mathrm{x}} + \lambda} \quad (9)$$

$\hat{w}$ is the kernelized correlation filter by training, x is the feature vector of target in current frame, $\hat{\mathrm{x}}^*$ is a conjugation after Fourier transform of x. $\hat{y}$ is the Gauss shaped regression label, λ is a trade-off parameter. The complex rate of Eq.(9) is reduced to $O(n)$.

Joint target tracking uses the two filters calculate the maximum response values and the coordinate location respectively, and assign the different weight according to the response value, then assign weight to initial coordinate position, and get accurate position by weighted summation. The training set is given by Eq.(2).

We use the following correspondence to train the filter:

$$x_i = P^{(i-1)} x \quad (10)$$

The training samples are obtained by cyclic transfer of object obtained from the first frame. The specific can be expressed as:

$$X = C(x) = \begin{bmatrix} x^{(1)} & x^{(2)} & \cdots & x^{(n)} \\ x^{(2)} & x^{(3)} & \cdots & x^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ x^{(n)} & x^{(1)} & \cdots & x^{(n-1)} \end{bmatrix} \rightarrow \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (11)$$

In the next frame, we get the tracking frame image Z, and its position is the upper frame target position, then according to the Eq.(9) we can get regression response value of cyclic transfer sample. Assuming there is a minor target changes, ideally the response value is:

$$y' = \begin{bmatrix} y_n \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix} \quad (12)$$

Z can be obtained by back stepping:

$$Z = C(z) = \begin{bmatrix} x^{(n)} & x^{(1)} & \cdots & x^{(n-1)} \\ x^{(1)} & x^{(2)} & \cdots & x^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ x^{(n-1)} & x^{(n)} & \cdots & x^{(n-2)} \end{bmatrix} \quad (13)$$

This is the image in the tracking frame:

$$z = \begin{bmatrix} x^{(n)} \\ x^{(1)} \\ \vdots \\ x^{n-1} \end{bmatrix} \quad (14)$$

We can use Z to judge the situation of the target movement, Z is obtained due to the movement of target X. In this case, Z is obtained by X moving down a pixel and the target frame is moving along.

B. Scale Adaptive Method

The proposed algorithm not only deals with scale change of tracking target, but also achieves real-time tracking. At the same time, it solves some important problems, such as inaccurate tracking and great calculating quantity.

In this paper, we introduce the idea of scale adaptive, to ensure that the tracking frame is adapt to target scale changes in the tracking process. The adopted mathematical model is:

$$maxF^{-1}\hat{f}\left(z^{S_i}\right) \quad (15)$$

F^{-1} is the inverse Fourier transform, $S_i \in (S_1, S_2 \ldots S_k)$ is the scale transformation factor, z is the target for the current tracking frame. Respectively, the tracking box is multiplied by the different scale transformation factor, and output corresponding value. The maximum response scale change factor is the target scale changes situation, so it can change the current tracking frame scale in real time.

Due to the variation of the target scales, the filter and the Gauss regression label also changes correspondently. We use the bilinear interpolation method to scale and fix target, in order to avoid the failure of filter update and the recalculate of Gauss regression label. Through the selection of a suitable fixed size method, we can avoid inaccurate tracking and too large amount of calculation because the tracking target is too small. Tracking results are obtained on the test with different fix size on the 28 benchmark sequences [7].

We test the performance of our approach using DP, CLE and OP. DP is computed as the relative number of frames in the sequence where the centre location error is smaller than a

TABLE I
TRACKING RESULTS OBTAINED FROM TESTING ON DIFFERENT FIXED SIZE.

Method	Median OP	Median DP	Median CLE	Median FPS
70*70	39.2	75.4	13.2	70.2
90*90	50.1	88.3	11.4	60
110*110	65.8	90.2	10.5	45.8
130*130	80.5	96.8	10.1	20.4

TABLE II
TRACKING CONTRAST UNDER DIFFERENT THRESHOLDS ON THE 28 BENCHMARK SEQUENCES.

ε	0.5	0.4	0.3	0.2
occlusion(20%)	0.75	0.82	0.91	0.88
occlusion(50%)	0.71	0.80	0.92	0.84
occlusion(80%)	0.68	0.77	0.89	0.79

certain threshold. CLE is computed as the average Euclidean distance between the ground-truth and the estimated centre location of the target. OP is defined as the percentage of frames where the bounding box overlap surpasses a threshold $t \in [0, 1]$. We report the results at a threshold of 0.5, which correspond to the PASCAL evaluation criteria. We provide results using median DP, CLE and OP. In addition, we calculated the speed of the trackers in median frames per second. The table above indicates that with too small fixed size, although the processing speed is very fast, the tracking accuracy is not very good; with big fixed size, the tracking effect are ensured, but poor in time effectiveness. Therefore, according to our requirements we can set different fixed size. In this paper, we use the fixed size of 110*110 as shown in Table 1.

C. Large Area Occlusion Strategy

During the process of tracking, the target often appear some minor variations, so we need to update the relevant filter by real-time training. Update strategy is given just as follows:

$$\hat{\alpha}_i = (1 - \theta)\, \hat{\alpha}_{i-1} + \theta \hat{\alpha}_{cwi} \tag{16}$$

$\hat{\alpha}_i$ is the updated filter, $\hat{\alpha}_{i-1}$ is the filter of the last frame, $\hat{\alpha}_{cwi}$ is the filter obtained from current tracking objects, θ is the update filter factor.

This filter update strategy is good at dealing with small changes in tracking target, but when the tracking target is occluded with big area, the tracking algorithm will be ineffective. Considering this situation, we propose a new method to update the filter.

We set the threshold value to 0.3, and comparing the maximum response with this threshold every time. If the maximum response value is greater than the threshold value, we do not change θ. If the maximum response value is less than the threshold value, we determine that the target is occluded or disappeared suddenly, at this time we stop the filter update (the filter updated factor is set to 0), and ensure that the scale is no longer updated, then expand our search box.

In this way, our method can track well when the target covered by large area or long time.

I_i represents current frame image. P_{i-1} and S_{i-1} represent last frame target location and scale respectively. w_{i-1} denotes correlation filter (HOG and LBP). S_{fix} is fixed size andε is threshold. P_i and S_i are predicted target location and scale .

According to the location of the previous frame target P_{i-1} and extraction of samples x_i of the current frame I_i at S_{i-1} scale. Different scale change factors are given to the samples x_i, and resize it to fixed size by the bilinear interpolation method then get $x_i^{s_i}$. With different scale samples $x_i^{s_i}$ and correlation filters w_{i-1}, calculate the corresponding value $y_{max}^{s_i}$. The position of the current frame target is a sample with max($y_{max}^{s_i}$), and the corresponding scale factor is the transformation of the target size, and can get the current frame target scale S_i.

According to the previous frame location P_{i-1} and scale S_{i-1}, extract samples x_i from current frame. Computing the correlation response values y_{max} by sample x_i and correlation filter w_{i-1}. According to the relative response value y_{max} and the setting threshold value ε to judge the occlusion, and take different countermeasures according to comparison result.

In the model updating process, we get the current frame target x_{new} using the current frame target x_{new} to train correlation filter w_{new} (HOG and LBP) and using x_{new} and w_{i-1} to get the current frame filter w_i (HOG and LBP).

III. EXPERIMENTS

The algorithm proposed in this paper is realized by C++ programming, and the operating environment is Intel i3 2 core 3.7GHZ 4G RAM. At first, Comparing joint feature representation and HOG feature representation. Secondly, to compare two different scale adaptive methods based on KCF tracking algorithm. Finally, we have compared the proposed algorithm with the current popular algorithm on dataset benchmark.

A. Image Representation using HOG and LBP

In this paper, we combine the HOG and LBP features of the target to represent and track the target. HOG features are good at describing outline features of the reaction target, LBP features are good at reflecting the texture features of the reaction target.

Compared with the simple HOG features, the method proposed in this paper improves median DP and OP by 13.1% and 12.0% respectively. At the same time, our proposed method can reduce the median CLE from 16.5 to 8.2 pixels. Our experimental results show that the combined method greatly improves the accuracy and robustness of the tracking. Therefore, in the scale adaptive method, we also use two kinds of features to track the target.

TABLE III
COMPREHENSIVE EVALUATION ON JOINT AND HOG REPRESENTATION TRACKING.

Method	Median OP	Median DP	Median CLE	Median FPS
HOG and LBP	80.2	90.5	8.2	60.2
HOG	71.6	80.0	16.5	102.4

TABLE IV
TRACKING RESULTS AIMED AT SCALE ADAPTION.

Method	Median OP	Median DP	Median CLE	Median FPS
KCF	40.2	78.4	14.8	102.4
KCF with scale adaptive	42.1	82.3	15.2	15.8
Our method	52.0	88.8	10.5	45.8

B. Robust Scale Estimation

Table 4 shows the tracking results of our scale adaptive method. Compared with other methods without scale adaptive or are of scale adaptive without fixed size setting. The former method has an obvious advantage of the speed but not very ideal at tracking accuracy.

The speed of the latter is much lower than the former method, but the accuracy has a slight improvement and has the ability to estimate the variation of the target scale. Too big tracking target leads to extra calculation burden and the low speed. The Selected target is too small to attain nice tracking performance. Therefore, our proposed method taking a fixed size strategy to track the target has an obvious improvement both in the processing speed and the accuracy. From above data, our method has improved the Median OP and Median DP. Our method also has enhanced the second method by 23.7% and 7.9%. Moreover, Median CLE has reduced to 10.5 pixels.

From the data, we can easily get the conclusion that the scale adaptive method that we proposed not only can track object accurately but also can accurate estimate the target scale transformation. What's more, it gains an obvious advantage of the speed that far exceeding the requirements of real-time.

C. Comparison with State-of-the-Art

In order to better illustrate the superiority of the proposed algorithm, we have compared our proposed method with the accuracy and robustness of the popular algorithm. The whole experiment process is carried out on dataset benchmark. The tracking results are shown in the table 5.

The accuracy represents how well the bounding box predicted by the overlaps with ground truth. The robustness keeps a record of how many times the tracker loses the target, which means the overlap becomes zero. In this case the accurate estimation of scale is crucial to model update which affects both the accuracy and robustness. From the table above, we can easily find that our algorithm achieves the best results on both accuracy and robustness. Compared with two other algorithms, our algorithm can be well adapted to the variation of the target scale, and has good robustness to occlusion, so the tracking results are better than that of other algorithms.

TABLE V
TRACKING RESULTS COMPARISON OF DIFFERENT ALGORITHMS.

	Accuracy			Robustness		
Dataset	KCF	STC	Ours	KCF	STC	Ours
boy	0.752	0.766	0.821	2	1	0
car4	0.572	0.521	0.75	2	2	1
carScale	0.322	0.421	0.731	2	2	1
couple	0.501	0.494	0.652	2	3	2
crossing	0.792	0.651	0.862	0	0	0
david	0.543	0.532	0.721	0	0	0
dog1	0.421	0.552	0.343	2	5	1
doll	0.721	0.621	0.822	1	0	1
jogging	0.887	0.703	0.768	1	1	1
trellis	0.601	0.642	0.746	0	1	0
freeman1	0.618	0.563	0.753	3	6	2
Freeman3	0.422	0.322	0.54	8	10	6
Mean	0.596	0.566	0.709	1.9	2.6	1.3

IV. CONCLUSIONS

In this paper, we present a fast scale adaptive tracking method based on correlation filtering for solving serious occlusion in medical image processing. This method can track the change of the target scale quickly and deal with the problem of large area occlusion, even the complete disappearance of the target. And, the powerful features including HOG and LBP are fused together to further boost the overall performance. Whats more, compared with the best scale adaptive tracking method, our method tracks the target more accurately and meets the need of the real-time performance better.

REFERENCES

[1] L. Zhang, L. Yang, and T. Luo, "Unified saliency detection model using color and texture features." *Plos One*, vol. 11, no. 2, 2016.

[2] L. Zhang, Y. Sun, T. Luo, and M. M. Rahman, "Note: A manifold ranking based saliency detection method for camera," *Review of Scientific Instruments*, vol. 87, no. 9, pp. 309–320, 2016.

[3] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 702–715.

[4] J. Henriques, J. Carreira, R. Caseiro, and J. Batista, "Beyond hard negative mining: Efficient detector learning via block-circulant decomposition," in *proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2760–2767.

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

[6] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Computer vision and pattern recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1822–1829.

[7] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2411–2418.