Confidence map based KCF object tracking algorithm

1st Baoguo Wei
School of Electronic Information
Northwestern Polytechnical University
Xi'an, China
Wbg@nwpu.edu.cn

2nd Yufei Wang
School of Electronic Information
Northwestern Polytechnical University
Xi'an, China
18435169138@mail.nwpu.edu.cn

3rd Xingjian He
School of Electronic Information
Northwestern Polytechnical University
Xi'an, China
2014302129@mail.nwpu.edu.cn

***Abstract*—Tracking with kernelized correlation filters (KCF) is an excellent object tracking algorithm, which is widely concerned. However, it still has many limitations, such as its tracking performance decreases in complex scenes, and the model is easily contaminated because it is updated every frame. In KCF, each candidate patch in tracked region corresponds to a confidence ratio reflecting the probability that it contains the target, and the tracking patch corresponding with the maximum confidence ratio is the output. We combine all available confidence ratios to obtain a confidence map and infer the tracking scene by analyzing the confidence map. For complex scenes, we dynamically improve KCF to enhance its tracking performance. In addition, we propose an innovative model update mechanism to reduce the computational complexity and model contamination. The experimental results show that compared with the conventional KCF algorithm, the proposed approach improves success rate and precision by 7% and 8% respectively.**

***Index Terms*—object tracking, confidence map, kernelized correlation filter, model update mechanism**

I. INTRODUCTION

Object tracking is a classic problem in computer vision, which has a wide range of applications. [1, 2]. However, it is still difficult due to many challenging aspects, such as occlusion, fast motion, similar target and deformation.

In recent years, many object tracking algorithms based on discriminative learning method are proposed and have made a lot of process [3]. In 2015, Henriques proposed KCF (Tracking with Kernelized Correlation Filters) algorithm [4]. It obtains periodic training samples by cyclically shifting the image and simplifies the calculation process using DFT (Discrete Fourier Transform). "Kernel trick" is simultaneously employed to improve its tracking performance. However, KCF uses the same tracking mechanism in different scenes, it can track accurately in simple scenes, but its tracking performance decreases in complex scenes. When faced tracking scenes with similar target, background clutter or motion blur, it is difficult for KCF to use a single HOG (Histogram of Oriented Gradients) feature to effectively describe the target. When faced tracking scenes with occlusion or fast motion, the limited tracked region in KCF also impairs tracking performance. In addition, KCF model is updated every frame, it is computationally intensive and easily causes model contaminant.

Based on these observations, we propose an improved KCF algorithm based on confidence map. The novelty of our work is that:

- A confidence map is proposed. The confidence map reflects all available confidence ratios. According to the peak response types of the confidence map, we can deduce the type of the tracking scene and the challenging aspects that may be included.
- An improved KCF algorithm is proposed for complex tracking scenes. When faced tracking scenes with similar target, background clutter or motion blur, we combine the color features and HOG features to effectively describe the target. When faced tracking scenes with occlusion or fast motion, we expand the tracked region to improve the tracking performance.
- A model update mechanism is proposed based on the extent of target deformation. The proposed deformation index determines whether the model needs to be updated. The algorithm model is no longer updated every frame, which reduces the amount of calculation and model contamination.

The experimental results show that compared with the conventional KCF algorithm, the proposed approach improves success rate and precision by 7% and 8% respectively.

The rest of this paper is organized in five sections. The second part describes the related works. The third part presents the enhanced algorithm. The fourth part reports the results of the experiment. Finally, the fifth part makes a summary and introduces the future work.

II. RELATED WORKS

In this section, we will discuss discriminative object tracking algorithms and confidence ratio.

One of the biggest breakthroughs in recent object tracking research was the adoption of discriminative learning methods. Discriminative object tracking algorithms regard tracking as an online learning problem. It employs the target area as positive samples and the background area as negative samples to learn a classifier. The classifier can discriminate between the target and surrounding environment to achieve object tracking.

Science and Technology Research Program of Xi'an (2017086CG/RC049), China.

Key Science and Technology Program of Guizhou Province (2017GZ60903), China

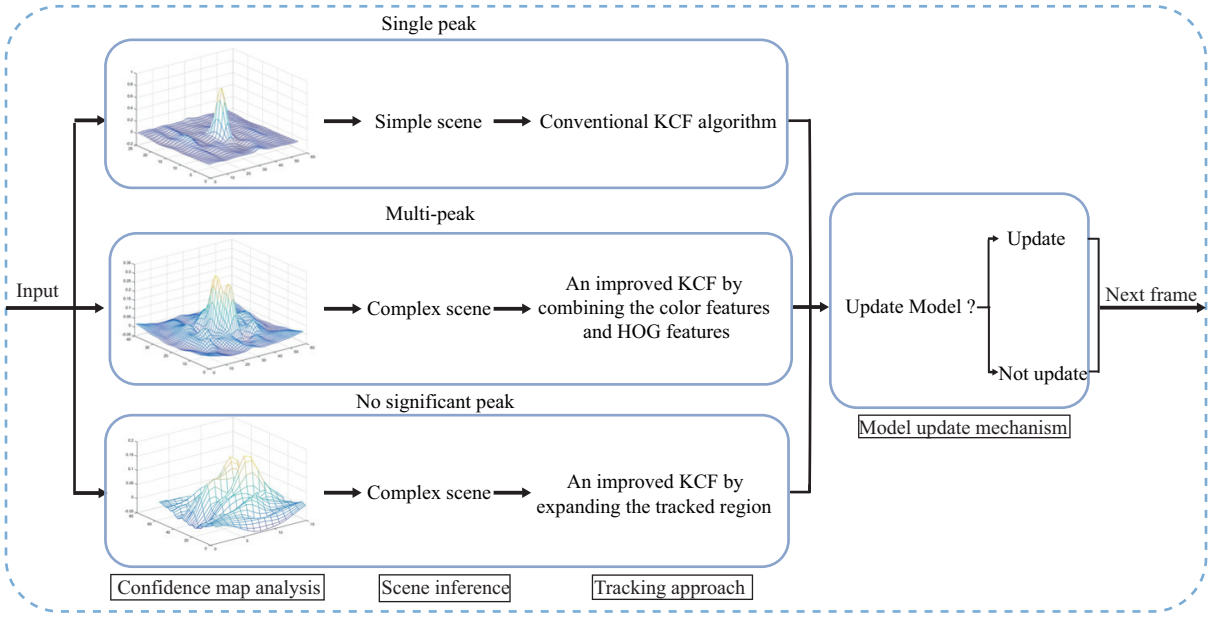


Fig. 1. The flow chart of confidence map based KCF algorithm

Typical examples of classifiers include SVM (Support Vector Machines) [5], random forest classifiers [6] and boosting variants [7].

The classifier after learning is tested on many candidate patches of the tracked region to find the most likely target location [8]. For a specific classifier, every candidate patch will obtain a classification probability reflecting the possibility that it contains the target. In this paper, we use the confidence ratio to represent the classification probability. KCF generally considers the patch corresponding to the maximum confidence ratio as target location. We combine all available confidence ratios to obtain a confidence map, it expresses the probability that any candidate patch in the tracked region contains the target.

III. THE PROPOSED APPROACH

The proposed approach is shown in Fig. 1. We first obtain the confidence map of the input image. By analyzing the local maximum of the confidence map, we class it into three peak response types: single peak confidence map, multi-peak confidence map and the confidence map with no significant peak. For each type, we infer the complexity of the tracking scene and the challenging aspects that may be included. Then, different tracking approaches are employed to improve the tracking performance of KCF. Finally, we use the proposed model update mechanism to determine whether the model needs to be updated. In this section, we introduce the proposed approach from three parts: KCF algorithm, tracking based on confidence map and model update mechanism.

A. KCF algorithm

The core component of KCF is a discriminative classifier $f(z) = \mathrm{w}^T z$ which discriminates between the target and surrounding environment. For object tracking, the target location is given in the first frame of the video sequence. KCF extracts the HOG feature [9] of the target and obtains training samples by cyclic shift. $\mathrm{x} = [x_1, x_2, \cdots, x_n]^T$ is a HOG feature array, circulating shift it as $P_{\mathrm{x}}^1 = [x_n, x_1, x_2, \cdots, x_{n-1}]^T$. All loop shifted samples $\{P_{\mathrm{x}}^u | u = 0, \cdots, n-1\}$ constitute a circulant matrix $\mathrm{X} = C(\mathrm{x})$. KCF uses the fact that the circulant matrix can be diagonalized by DFT.

$$X = F^H \mathrm{diag}(\hat{\mathrm{x}}) F \tag{1}$$

Where F represents the DFT matrix which computes the DFT of any input vector, as $\mathcal{F}(\mathrm{x}) = \sqrt{n} F\mathrm{x}$, F^H is the Hermitian transpose matrix of F. From now on, a hat $\hat{}$ is used as shorthand for the DFT of a vector.

For Ridge Regression, we can obtain $\mathrm{w} = (\mathrm{X}^T\mathrm{X} + \lambda I)^{-1} X^T y$ by (2), where x_i is a sample and y_i is its regression target.

$$\min_{\mathrm{w}} \sum_i (f(\mathrm{x}_i) - y_i)^2 + \lambda \|\mathrm{w}\| \tag{2}$$

We define the element-wise product as $\odot$. Since X is a circulant matrix, w shown in (3) can be obtained.

$$\hat{\mathrm{w}} = \frac{\hat{\mathrm{x}}^* \odot \hat{y}}{\hat{\mathrm{x}}^* \odot \hat{\mathrm{x}} + \lambda} \tag{3}$$

"Kernel trick" is employed for non-linear regression. The solution of the kernel function can be transformed into a linear combination of samples: $\mathrm{w} = \sum_i \alpha_i \varphi(x_i)$, translates into solving $\alpha = \{\alpha_1, \alpha_2, \cdots, \alpha_n\}^T$. After the diagonalization of the kernel matrix, the Gauss kernel function can be used to obtain as follow:

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

For any two vectors x and x′, their kernel function related operations from the $k^{\mathrm{xx}'}$, the i element of vector $k^{\mathrm{xx}'}$ is

$k^{\mathrm{xx}'} = k(\mathrm{x}', P^{i-1}\mathrm{x})$, $k^{\mathrm{xx}'}$ is two parameters of different relative displacements evaluated in the kernel. The use of Gauss kernels for $k^{\mathrm{xx}'}$ yields the following expression:

$$\hat{k}^{\mathrm{xx}'} = \exp(-\frac{1}{\sigma^2}(\|\mathrm{x}\|^2 + \|\mathrm{x}'\|^2) - 2F^{-1}(\hat{\mathrm{x}} \bigodot \hat{\mathrm{x}}'^*)) \quad (5)$$

For the KCF algorithm using the Gauss kernel function, the HOG feature of the image is extracted first, and the σ in the kernel function of the algorithm is calculated, and the λ in the α is calculated to train the samples. The intensive sampling of the tracked region is used to calculate the correlation between the samples to be measured, and the target location of the frame is obtained.

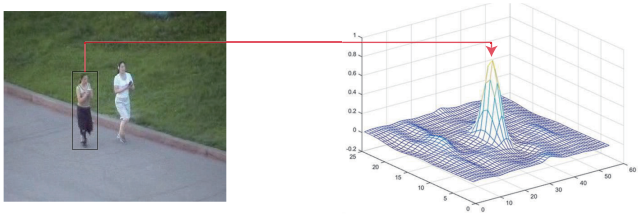
B. Tracking based on confidence map

The confidence map is classed into three different peak response types: single peak confidence map, multi-peak confidence map and the confidence map with no significant peak. We check in order which type the confidence map belongs to and infer its corresponding scene. For different scenes, we employ different tracking approaches.

1) Single peak confidence map: if the confidence map has a local maximum m_i which satisfies (6), it is considered as a single peak confidence map.

$$\frac{m_i}{m_j} \geqslant \varepsilon, \forall i \neq j \quad (6)$$

Where m_j is any local maximum of the confidence map other than m_i, ε is the threshold of the peak ratio we set it to 1.5. The single peak confidence map and its corresponding tracking scene are shown in Fig. 2.



(a) Tracking scene (b) Single peak confidence map

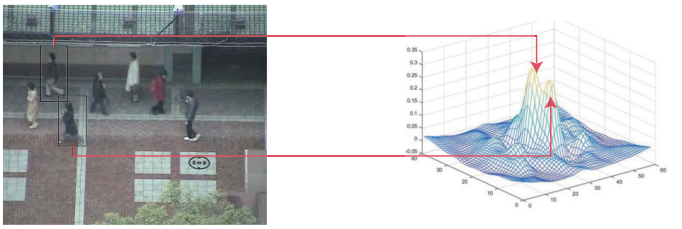
Fig. 2. Single peak confidence map and its corresponding tracking scene

Confidence map analysis: only one candidate patch can be identified as the tracking target, and the probability is high.
Scene inference: simple scene.
Tracking approach: KCF algorithm. The patch corresponding to the maximum of the confidence map is output as the target.

2) Multi-peak confidence map: if the confidence map has multiple local maxima (more than one) greater than the threshold of the peak, we set it to 0.6, and any two of these local maxima satisfy (7), it is considered as multi-peak confidence map.

$$\frac{m_i}{m_j} < \varepsilon, \forall i \neq j \quad (7)$$

Where ε is the same threshold as the previous section. The multi-peak confidence map and its corresponding tracking scene are shown in Fig. 3.



(a) Tracking scene (b) Multi-peak confidence map

Fig. 3. Multi-peak confidence map and its corresponding tracking scene

Confidence map analysis: multiple candidate patches can be identified as the tracking target, and their probabilities are similar.
Scene inference: complex scenes with similar targets, background clutter or motion blur.
Tracking approach: an improve KCF algorithm by combining the color features and HOG features. The color information is widely used in object tracking because it describes the target well [10]. More importantly, it is independent and complementary to the HOG feature adopted by KCF [11]. Therefore, we add the color histogram feature to the original model of KCF to build a more robust object model. It compensates for the inability of KCF to accurately track in the above tracking scenes.

If the t-th frame is a color image, we convert it from the RGB color space to the Lab color space. The information of three channels (L channel, a channel and b channel) is extracted and combined into a color histogram. If the image is a grayscale image, the information of L channel is extracted to form a color histogram. Because the time interval between two adjacent frames is small, their color information is similar. We calculate the color histogram similarity between the candidate patches of the t-th frame and the estimated target of the (t-1)th frame by (8).

$$\rho = \frac{1}{d(\mathrm{c}_{t-1}^{target}, \mathrm{c}_t^i)} \quad (8)$$

Where $\mathrm{c}_{t-1}^{target}$ is the color histogram feature of the estimated target of the (t-1)th frame, c_t^i is the color histogram feature of candidate patches in the t-th frame, $d(\mathrm{c}_{t-1}^{target}, \mathrm{c}_t^i)$ is the Euclidean distance between them. ρ is normalized to obtain the confidence ratio of the candidate patches under the color histogram feature, denoted as $g(z)$. We combine it and the confidence ratio under the HOG feature $f(z)$ and to obtain the final confidence ratio $s(z)$ by (9).

$$s(z) = 0.5f(z) + 0.5g(z) \quad (9)$$

The patch corresponding to the maximum of $s(z)$ is output as the target.

3) The confidence map with no significant peak: if the confidence map is neither single peak nor multi-peak, we consider it as the confidence map with no significant peak. The confidence map with no significant peak and its corresponding tracking scene are shown in Fig. 5.

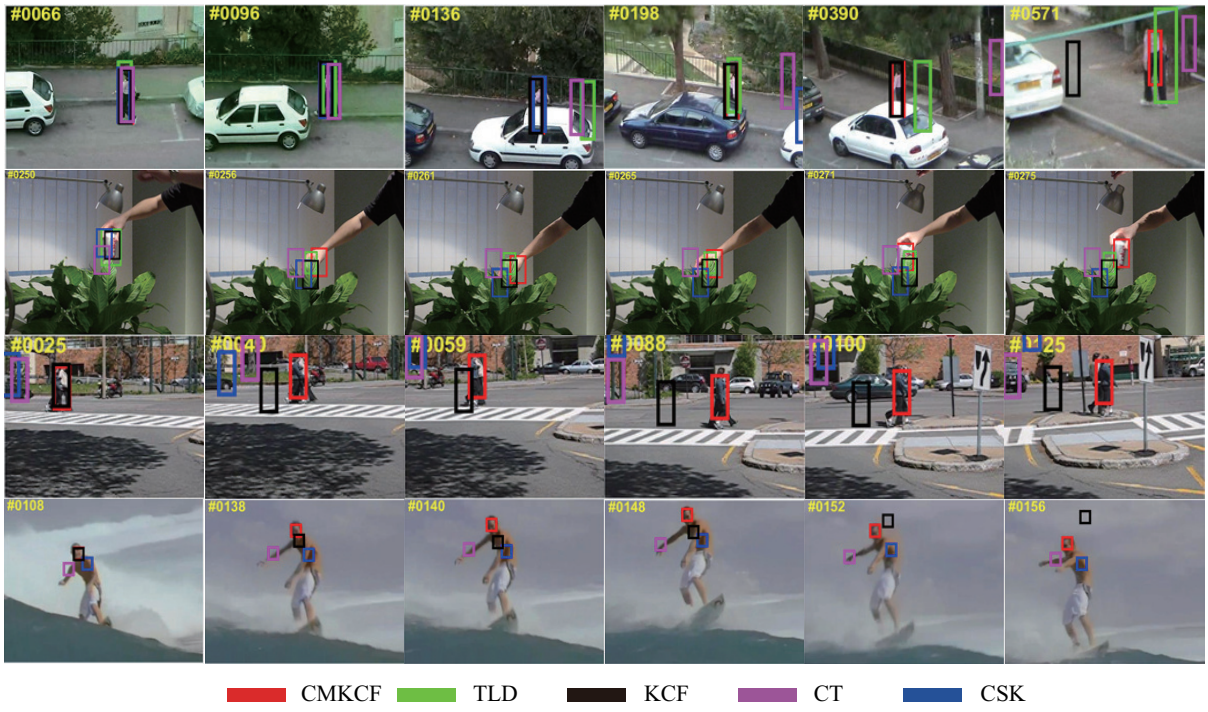
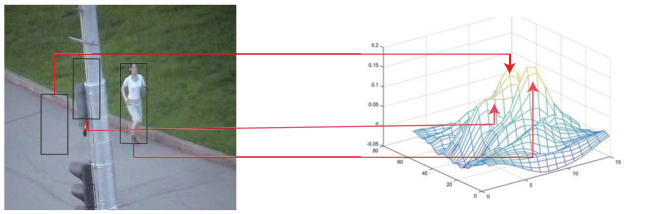


Fig. 4. The tracking results of different algorithms on the same video sequence (from top to bottom are Woman, Coke, Couple and Surfer).



(a) Tracking scene (b) Confidence map with no significant peak

Fig. 5. The confidence map with no significant peak and its corresponding tracking scene

Confidence map analysis: no candidate image patch can be identified as the tracking target.

Scene inference: complex scenes with occlusion or fast motion.

Tracking approach: an improve KCF by expanding the tracked region. Double the tracked region of the conventional KCF. The improved algorithm is more likely to track successfully in the above scenes.

C. Model update mechanism

During the tracking process, the target will be deformed, the algorithm model needs to be constantly updated [12]. The KCF model is updated every frame. In this way, the calculation is very heavy and the model is easily contaminated. We propose an innovative model update mechanism to address the problem, whether the model is updated depends on the extent of target deformation.

The first frame of the video sequence contains the complete information of the target. The extent of target deformation can be measured by calculating the difference between the estimated target of the current frame and the given target of the first frame. Since KCF extracts the HOG feature of each frame, the deformation extent d can be obtained by (10).

$$d = d(\mathrm{x}_1, \mathrm{x}_t) \tag{10}$$

Where x_1 is the HOG feature of the given target of the first frame, x_t is the HOG feature of the estimated target of the current frame, $d(\mathrm{x}_1, \mathrm{x}_t)$ is the Euclidean distance between them. If d is greater than the threshold we set it to 1.5, it means the deformation of the target is large enough to update the model. The model update method is the same as KCF method.

IV. EXPERIMENTS

We implement our method with MATLAB on a desktop computer with Inter Core i5, 3.20GHz, 4GB RAM. The video sequences we use come from OTB (Visual tracker benchmark) dataset [13]. The proposed approach denoted by CMKCF is compared with related state-of-the-art algorithms such as KCF [4], TLD [14], CT [15], CSK [16]. Experimental results show that CMKCF has a better tracking performance.

A. Quality evaluation

To evaluate the proposed approach, we compare the experimental result of CMKCF with the results of KCF [4], TLD [14], CT [15], CSK [16]. Fig. 4 shows the tracking results of different algorithms on the same video sequence. The video sequences we use are Woman, Coke, Couple and Surfer.

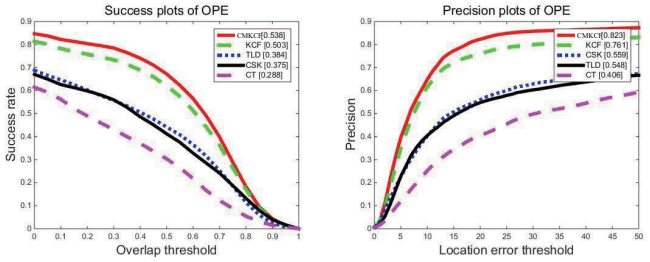


Fig. 6. The experimental results of the proposed algorithm on OTB dataset

For Woman, we track a woman who is walking along the road, it means that the target is constantly deformed. At approximately 136th frame, the target is occluded by a white car, TLD and CT fail to track. At approximately 571th frame, KCF also fails to track, CMKCF still accurately tracks the target. The experimental results show that CMKCF has a better performance compared to other algorithms in this video sequence.

In Coke, a canister in the hand of a man is determined as the tracking target. In this video sequence, occlusion and fast motion occur simultaneously. At approximately 250th frame, CT is slightly offset, TLD and CSK start to drift. At approximately 271th frame, the tracking boxes of TLD, KCF, CT and CSK all drift, only CMKCF accurately tracks the target.

For Couple, the tracking scene with similar targets and background clutter is complex. At approximately 25th frame, only CMKCF and KCF accurately track the target. At approximately 40th frame, KCF using a single HOG feature does not describe the target well, its tracking box seriously deviates from the target. From this frame, only CMKCF can accurately track the target. It means that CMKCF has a stable performance on this video sequence.

In surfer, the main challenging aspect is fast motion. Faced with this challenging aspect, CT and CSK, at approximately 108th frame, fail to track the target. At approximately 138th frame, the target is not in the tracked region of KCF due to fast motion, the tracking box of KCF starts to drift. However, CMKCF expanding the tracked region of KCF accurately tracks the target.

B. Quantitative evaluation

OTB dataset [13] contains 50 video sequences and two evaluation metrics: success rate and precision. Fig. 6 shows the experimental results of the proposed algorithm on OTB. We compare the experimental result of CMKCF with the results of KCF [4], TLD [14], CT [15], CSK [16] in TABLE I, which includes success rate, precision and tracking speed. The results show that the proposed approach and KCF are obviously better than other algorithms. Moreover, compared with the conventional KCF algorithm, the proposed approach has a slowdown in the tracking speed, but it improves success rate and precision by 7% and 8% respectively.

In order to compare the proposed approach and the conventional KCF in more detail, we show their success rate and precision under specific challenging aspects in TABLE II. The results show that the proposed approach is better than conventional KCF in almost all challenging aspects.

TABLE I
THE COMPARISON OF EXPERIMENTAL RESULTS BETWEEN CMKCF, KCF, TLD, CT AND CSK

Trackers	Success rate	Precision	Mean FPS[a]
CMKCF	0.538	0.823	98
KCF[4]	0.503	0.761	172
TLD[14]	0.384	0.548	28
CT[15]	0.375	0.559	64
CSK[16]	0.288	0.406	320

[a]FPS represents Frames Per Second.

TABLE II
THE COMPARISON OF EXPERIMENTAL RESULTS UNDER SPECIFIC CHALLENGING ASPECTS BETWEEN CMKCF AND KCF

Challenging aspects	Success rate		Precision	
	CMKCF	KCF[4]	CMKCF	KCF[4]
Illumination Variation	0.527	0.418	0.793	0.721
Scale Variation	0.505	0.455	0.789	0.706
Occlusion	0.610	0.591	0.808	0.789
Deformation	0.570	0.529	0.837	0.771
Motion Blur	0.613	0.566	0.828	0.748
Fast motion	0.565	0.504	0.783	0.671
In-Plane Rotation	0.530	0.503	0.778	0.738
Out-of-Plane Rotation	0.566	0.78	0.831	0.763
Background Clutter	0.649	0.566	0.920	0.797
Overall	0.538	0.503	0.823	0.761

V. CONCLUSION

After an in-depth analysis of KCF, we propose an improved KCF tracking algorithm based on confidence map. The tracking scene is inferred from the different peak response types of the confidence map. For complex scenes, we dynamically improve the KCF algorithm to enhance its tracking performance. In addition, we propose an innovative model update mechanism to reduce the amount of calculation and model contamination. Through a number of evaluation and comparative experiments, it proves that the proposed approach is effective and improves the success rate and precision of KCF. In future work, we will make more accurate inferences about tracking scenes by studying the confidence map or other methods. For certain scenes, we will propose a specific tracking approach to accurately track.

REFERENCES

[1] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[2] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognition*, vol. 76, pp. 323–338, 2018.

[3] C. Sun, H. Lu, and M.-H. Yang, "Learning spatial-aware regressions for visual tracking," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 8962–8970.

[4] 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, 2015.

[5] W. Zuo, X. Wu, L. Lin, L. Zhang, and M.-H. Yang, "Learning support correlation filters for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[6] W. Wang, C. Wang, S. Liu, T. Zhang, and X. Cao, "Robust target tracking by online random forests and superpixels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 7, pp. 1609–1622, 2018.

[7] W. Jiang, Y. Wang, and D. Wang, "Robust visual tracking using a contextual boosting approach," *Journal of Electronic Imaging*, vol. 27, no. 2, p. 023012, 2018.

[8] S.-H. Bae and K.-J. Yoon, "Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 595–610, 2018.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[10] H. Possegger, T. Mauthner, and H. Bischof, "In defense of color-based model-free tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2113–2120.

[11] A. Lukezic, T. Vojir, L. C. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability." in *CVPR*, vol. 6, 2017, p. 8.

[12] P. L. Mazzeo, P. Spagnolo, M. Leo, P. Carcagnì, M. Del Coco, and C. Distante, "Dense descriptor for visual tracking and robust update model strategy," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–11, 2017.

[13] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[14] Z. Kalal, K. Mikolajczyk, J. Matas *et al.*, "Tracking-learning-detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, p. 1409, 2012.

[15] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *European conference on computer vision*. Springer, 2012, pp. 864–877.

[16] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European conference on computer vision*. Springer, 2012, pp. 702–715.