Adaptive Scale Feature Fusion and Model Update Tracking Algorithm (Postprint)

Authors: Wang Rihong, Yongjun Li, Zhang Lifeng

Date: 2018-10-11T00:00:00+00:00

Abstract

In the kernel correlation filter tracking algorithm, to reduce the interference of clutter such as background-similar objects on the tracker, and to address the model update problem under different tracking result confidence levels, a tracking algorithm with adaptive scale feature fusion and model update is proposed. Extensive feature fusion and scale variation strategies improve the multi-feature scale kernel correlation filter, using multi-peak detection to judge the overall oscillation degree of the response map, and then performing tracking result confidence evaluation on the peaks; in cases of low tracking result confidence such as occlusion and deformation, model update is promptly stopped; during high-confidence model updates, the initial model is introduced for alignment operations to reduce model update errors and suppress model drift. Compared with the kernel correlation filter algorithm, this algorithm has higher accuracy and better stability during target scale variation, occlusion, and deformation. Experimental results on the OTB-50 dataset demonstrate that this algorithm performs better than the kernel correlation filter algorithm in both precision and success rate.

Full Text

Self-adaptive Scale Feature Fusion and Model Update Tracking Algorithm

Wang Rihong, Li Yongjun, Zhang Lifeng
(School of Information & Control Engineering, Qingdao University of Technology, Qingdao, Shandong 266520, China)

Abstract

In kernel correlation filter tracking algorithms, background clutter and similar objects often interfere with the tracker, and model updating under varying tracking result confidence levels presents a significant challenge. To address these issues, this paper proposes a Self-adaptive Scale Feature Fusion and Model Update (SFMU) tracking algorithm. The algorithm improves the multi-feature scale kernel correlation filter through enhanced multi-feature fusion and scale variation strategies. Multi-peak detection is employed to assess the overall oscillation degree of the response map, followed by confidence evaluation of the tracking result based on peak characteristics. Model updating is promptly suspended during low-confidence scenarios such as occlusion and deformation, while high-confidence updates incorporate alignment with the initial model to reduce update errors and suppress model drift. Compared with the kernel correlation filter algorithm, our approach achieves higher accuracy and demonstrates superior stability under target scale variation, occlusion, and deformation. Experimental results on the OTB-50 dataset show that the proposed algorithm outperforms the kernel correlation filter algorithm in both precision and success rate.

Keywords: multi-feature fusion; scale kernel correlation filter; multi-peak detection; high confidence; model update; suppress model drift

---

0 Introduction

Object tracking is a prominent research direction in computer vision with important applications in military operations, smart cities, and other practical domains. Over the past decade, significant advances have been made in object tracking algorithms alongside developments in artificial intelligence. In 2010, Bolme first introduced correlation filtering to visual tracking with the Minimum Output Sum of Squared Error (MOSSE) algorithm [1], which demonstrated remarkable tracking speed and proposed the Peak-to-Sidelobe Ratio (PSR) for evaluating tracking confidence. Building upon this work, Henriques et al. developed the Circulant Structure of tracking-by-detection with Kernels (CSK) [2] and the High-speed tracking with Kernelized Correlation Filters (KCF) [3]. The KCF algorithm leverages the properties of circulant matrices to transform computations into the Fourier domain, employing improved multi-channel Histogram of Oriented Gradients (HOG) features instead of single-channel grayscale features.

Numerous high-performance correlation filter-based algorithms have since emerged. Danelljan et al. proposed an Adaptive Color Attributes for Real-Time Visual Tracking algorithm [4] that utilizes 11 refined color channels from the RGB color space as an alternative to HOG features. They also introduced the Accurate Scale Estimation for Robust Visual Tracking (DSST) algorithm [5], which employs a dedicated filter similar to MOSSE for scale detection, separating scale estimation from position tracking. Possegger et al. presented the DAT algorithm [6], a color-based model-free tracking method that uses Bayesian probability with normalized color histograms from foreground and background regions. Bertinetto et al. combined template and color features in the Staple algorithm [7]. For long-term tracking, Ma et al. developed the Long-term Correlation Tracking (LCT) algorithm [8], which adds a confidence detection filter to the translation and scale filters. In recent years, deep learning has been applied to tracking, with notable algorithms including the MDNet algorithm by Nam et al. [9] and Danelljan et al.' s ECO algorithm [10] that extracts deep features from VGG-Net.

Current tracking methods primarily fall into two categories: correlation filter-based and deep learning-based approaches. While deep learning methods achieve higher accuracy, their computational complexity makes real-time performance difficult. Correlation filter methods offer better speed but face limitations: KCF and CN cannot handle scale variations; Staple fuses HOG and color features but with complex feature representation; DSST introduces scale strategies but suffers from instability; and LCT enables long-term tracking but becomes computationally expensive due to multiple filters. This paper proposes the SFMU algorithm, which performs multi-peak detection and PSR confidence evaluation on response maps from multi-feature scale kernel correlation filters to assess tracking confidence, incorporates the initial model during updates to suppress drift, and validates its effectiveness on the OTB-50 dataset [11].

1.1 Kernel Correlation Filter

The kernel correlation filter algorithm employs cyclic shifts for dense sampling to enrich the target sample set, trains a ridge regression classifier, and accelerates computation through diagonalization of circulant matrices using Fast Fourier Transform (FFT) [12]. The trained classifier computes correlation with new input images to generate a confidence response map, with the peak value indicating the predicted target position.

The ridge regression classifier aims to find a function $f(z) = w^T z$ that minimizes the error objective function, solved via least squares:

$$\min_{\mathbf{w}} \sum_{i=1}^{n} (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2$$

Kernel correlation filters introduce kernel functions through a nonlinear mapping function φ, transforming linear mappings into high-dimensional space to obtain the solution in Fourier space:

$$\hat{\alpha} = \frac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}_{xx} + \lambda}$$

where $\mathbf{x}$ represents the target feature vector, is the target filter model, and $\mathbf{z}$ is the image to be detected.

1.2 Multi-feature Scale Kernel Correlation Filter

The target feature vector $\mathbf{x}$ extracted by the ridge regression classifier describes the target and critically determines tracking accuracy. However, relying on a single feature has limitations. This work fuses Color Names (CN) with Histogram of Oriented Gradients (fHOG). The fHOG gradient feature performs well under motion blur and illumination changes but poorly under rapid deformation and fast motion, while color features are insensitive to deformation. Thus, these complementary features enhance robustness. We employ 31-dimensional fHOG, 10-dimensional CN, and 1-dimensional grayscale features to create a 42-dimensional multi-channel feature representation [Figure 1: see original paper].

Since KCF uses a fixed template, handling target scale variation requires constructing multi-scale images [Figure 2: see original paper]. Dense sampling extracts multi-channel features to train the ridge regression classifier, yielding a multi-scale model from which the optimal scale is selected [13].

The multi-scale kernel correlation filter procedure is as follows:

a) Determine the initial image size S_{xy} from the first frame.

b) Set n scale factor vectors S^i_{factor} with step size α_{factor}:

$$S^i_{factor} = S_{xy} \times (1 \pm i\alpha_{factor}), \quad i = 0, 1, \dots, \frac{n-1}{2}$$

where $\alpha_{factor} \in (0, 1)$ is the scale step size, with plus indicating scale increase and minus indicating decrease, and total scale variations equal n.

c) Apply bilinear interpolation [14] to the input image according to scale factors S^i_{factor} to obtain a multi-scale image pool.

d) Use a fixed-size sampling window to extract target patches from each scale, obtaining multi-scale target representations through image scaling and dense sampling.

e) Train multi-scale ridge regression target models:

$$\hat{\alpha}_i = \frac{\hat{\mathbf{y}}_i}{\hat{\mathbf{k}}_{\mathbf{x}i\mathbf{x}_i} + \lambda}$$

where $\hat{\alpha}_i$ and $\hat{y}_i$ are the multi-scale target model and response for scale i, $\mathbf{x}_i$ represents target features, and $\mathbf{z}_i$ is the test image.

f) Compute the optimal scale model:

$$i^* = \arg\max_i \hat{f}_i(\mathbf{z})$$

where $\hat{f}_i(\mathbf{z})$ is the response map from scale i detection.

g) Update the classifier' s target model $\mathbf{x}$ and target features using the strategy described below.

This approach combines multi-channel features with adaptive scale estimation to form a multi-feature scale kernel correlation filter.

2 Improved Model Update Strategy

Model drift represents a major challenge in target tracking, arising from two primary causes: (a) when background clutter or low-confidence situations like occlusion and deformation occur, the tracker incorporates noise into the model during updates; (b) inevitable detection errors accumulate through successive updates, gradually degrading tracking accuracy.

This paper improves upon KCF through two aspects: reducing background clutter interference and evaluating tracking confidence to select high-confidence results, and suppressing model drift during high-confidence updates by considering unavoidable detection errors.

2.1 Multi-peak Detection

Section 1.2 detects the optimal scale model, and KCF determines the target position in the next frame through single-peak detection on the response map. However, noise from similar objects can cause model drift, contaminating the target model and leading to tracking failure. As shown in [Figure 3: see original paper], the green box indicates the tracking result while the red box shows the actual target position. Both locations correspond to large response values, with interference exceeding the true target response. Continuing to track based on the peak position accumulates update errors and gradually loses the target.

Multi-peak detection [15] first identifies peaks exceeding threshold θ to construct a multi-peak matrix $\mathbf{P}$:

$$\mathbf{P} = \mathcal{B}(\hat{f}(\mathbf{z}) > \theta)$$

where $\hat{f}(\mathbf{z})$ is the optimal model response map and $\mathcal{B}(\cdot)$ produces a binary matrix matching the response map size. A secondary detection within the search region defined by peak locations in $\mathbf{P}$ selects the maximum response as the current frame' s peak F_{cur}.

The preliminary confidence is assessed using the peak value F_{cur} relative to the historical peak set $\mathbf{F} = \{F_i | i = 1, 2, \dots, n\}$:

$$u = \frac{1}{n}\sum_{i=1}^{n} F_i$$

When $F_{cur} > u$, the peak confidence is considered high, distinguishing the target from similar objects and reducing interference.

2.2 Peak Confidence Evaluation

Multi-peak detection assesses overall response map oscillation and determines peak location, providing preliminary confidence evaluation. However, it cannot fully distinguish targets under occlusion or deformation. As shown in [Figure 4: see original paper], rapid occlusion across consecutive frames produces two close peaks with low oscillation amplitude, indicating insufficient peak strength and clutter interference. The tracker mistakes the interference for the target, necessitating immediate model update suspension.

To further evaluate confidence, we introduce the PSR method from the original MOSSE algorithm [16], which measures oscillation in an 11×11 window around the peak:

$$PSR = \frac{g_{max} - \mu_s}{\sigma_s}$$

where g_{max} is the peak value, and μ_s and σ_s are the mean and standard deviation of window s. Higher PSR values indicate a more Gaussian-like distribution, suggesting more accurate tracking. We set threshold $\gamma = 1.8$; when $PSR > \gamma$, the tracking result is deemed high-confidence.

2.3 Suppressing Model Drift

After evaluating peak value and oscillation amplitude, high-confidence tracking results trigger model updates. To address model drift from inevitable update errors, we align updates with the initial target model [16].

The method proceeds as follows:
a) The detector identifies the optimal scale model passing multi-peak detection and confidence evaluation.
b) The peak position P_t serves as the estimated location. A search window is constructed at P_t, and the initial model parameters are used to compute a predicted target position P_t^*.
c) If the Euclidean distance between P_t and P_t^* centers is below threshold ε, update the model; otherwise, suspend updating.

The alignment leverages the fact that the target always retains some similarity to its initial appearance throughout tracking. In correlation filter tracking, successful results produce approximately Gaussian response maps with high peak strength and large oscillation amplitude. Small oscillation indicates low peak strength. The PSR method efficiently determines confidence with minimal computation.

2.4 Algorithm Flow

The SFMU algorithm flow is illustrated in [Figure 5: see original paper]:

Input: Video sequence and initial target position.
Output: Tracking results, precision plots, and success rate plots.

a) Determine target position in the first frame.

b) Extract multi-scale targets using the multi-feature scale kernel correlation filter, extract multi-channel features, and obtain the optimal target scale model.

c) Apply multi-peak detection to assess overall oscillation of the optimal model response map (Equation 7).

d) For multi-peak maps, perform secondary detection and select the maximum response as the current peak. Evaluate preliminary confidence using Equation 8.

e) Assess peak strength using PSR confidence evaluation (Equation 9).

f) Combine peak value (Equation 8) and PSR (Equation 9) for comprehensive confidence assessment. Perform model drift suppression during high-confidence updates.

g) Construct a search window at peak P_t and use the initial model to predict target position P_t^*. Update the model with P_t^* if $\|P_t - P_t^*\| < \varepsilon$; otherwise, skip updating.

h) Peaks failing PSR evaluation do not trigger model updates.

3 Experimental Analysis

Experiments were conducted on MATLAB R2016b under Windows 7 64-bit, with an Intel(R) CoreTM i5-6500 CPU @ 3.20 GHz and 4 GB RAM. The OTB-50 dataset was used for evaluation via precision and success rate metrics. For fair comparison, key parameters align with KCF: scale adaptation uses three scale factors $\{0.99, 1, 1.01\}$ with step size 0.01; multi-peak detection threshold $\theta = \frac{2}{3}F_{max}$; PSR threshold $\gamma = 1.8$; and drift suppression threshold $\varepsilon = 5$ pixels.

Five algorithms were evaluated: CSK, TLD [17], DSST, KCF, and the proposed SFMU.

3.1 Overall Performance Comparison

As shown in [Figure 6: see original paper] (overall precision) and [Figure 7: see original paper] (overall success rate), SFMU achieves 0.748 precision and 0.578 success rate, representing significant improvements of 10.98% and 12.45% over KCF, respectively.

3.2 Performance Under Scale Variation, Occlusion, and Deformation

[Figure 8: see original paper] and [Figure 9: see original paper] show performance under scale variation; [Figure 10: see original paper] and [Figure 11: see original paper] under occlusion; and [Figure 12: see original paper] and [Figure 13: see original paper] under deformation. In all cases, SFMU achieves the best precision and success rates, confirming its robustness.

3.3 Qualitative Analysis

Qualitative results on eight challenging video sequences from OTB-50 are compared in [Figure 14: see original paper] (a–h: Walking2, Gym, Car4, Tiger1, Jogging1, CarScale, Dog1, David). Sequence characteristics are detailed in .

Scale variation: In Walking2 frames 189, 194, 221, and 470, SFMU's bounding box adapts consistently to target scale changes, demonstrating effective scale adaptability.

Background clutter and occlusion: In Tiger1 frames 70, 174, 323, and 334, SFMU shows less susceptibility to background interference and can halt model updates during occlusion, maintaining high tracking accuracy.

Deformation and rotation: In Gym frames 659, 669, 679, and 689, despite dramatic appearance changes over tens of frames, SFMU continues tracking while other algorithms fail.

Illumination variation, low resolution, motion blur, and fast motion: Across all sequences, SFMU demonstrates the most stable performance, achieving robust tracking throughout.

4 Conclusion

This paper improves the multi-feature scale kernel correlation filter by introducing multi-peak detection on optimal scale model response maps to assess oscillation degree and PSR confidence evaluation, enhancing robustness to scale variation, occlusion, and deformation. Model drift is suppressed by aligning updates with the initial model. Experimental results on OTB-50 demonstrate excellent tracking performance and potential for further research.

References

[1] Cheng Yue, Li Jianzeng. Adaptive feature fusion correlation filter tracking algorithm combined with learning rate adjustment [J/OL]. Application Research of Computers, 2019, 36(8). http://www.arocmag.com/article/02-2019-08-060.html

[2] Xu Yulong, Wang Jiabao, et al. Scale-adaptive tracking based on kernelized correlation filter [J]. Application Research of Computers, 2016, 33(11): 3513-3516, 3520.

[3] Henriques J F, Caserio R, Martins P, et al. High-speed tracking with kernelized correlation filters [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2015, 37(3): 583-596.

[4] Danelljan M, Khan F S, Felsberg M, et al. Adaptive color attributes for real-time visual tracking [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1090-1097.

[5] Danelljan M, Hager G, Khan F S, et al. Accurate scale estimation for robust visual tracking [J]. British Machine Vision Conference, 2014: 65.1-65.11.

[6] Possegger H, Mauthner T, Bischof H. In defense of color-based model-free tracking [C]// Computer Vision & Pattern Recognition. 2015: 2113-2120.

[7] Bertinetto L, Valmadre J, Golodetz S, et al. Staple: complementary learners for real-time tracking [C]// Computer Vision & Pattern Recognition. 2016.

[8] Ma Chao, Yang Xiaokang, Zhang Chongyang, et al. Long-term correlation tracking [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2015.

[9] Han B, Nam H, et al. Learning multi-domain convolutional neural networks for visual tracking [C]// Computer Vision & Pattern Recognition. 2016.

[10] Danelljan M, Bhat G, Khan F S, Felsberg M, et al. ECO: efficient convolution operators for tracking [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2017: 6931-6939.

[11] Wu Y, Lim J, Yang M H. Online object tracking: a benchmark [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2411-2418.

[12] Wang Yanchuan, Huang Hai, Li Shaomei, et al. Adaptive target tracking algorithm based on fusion of two models [J]. Application Research of Computers, 2017, 34(12): 3828-3833.

[13] Qian Tanghui, Luo Zhiqing, Li Guojia, et al. Scale adaptive improvement of kernel correlation filter tracking algorithm [J]. Journal of Computer Applications, 2017, 37(03): 811-816.

[14] Du Jingwen, Huang Shan. Scale adaptive tracking algorithm of HLBP texture feature [J]. Computer Engineering and Design, 2018, 39(07): 1889-1894,

1929.

[15] Wang Mengmeng, Liu Yong, Huang Zeyi, et al. Large margin object tracking with circulant feature maps [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2017: 4800-4808.

[16] Liu Yang. Research on kernelized correlation filter tracking algorithm and its improvement [D]. Beijing: Beijing Institute of Technology, 2016.

[17] Tong Yuan, Fei Shumin, Shen Jie. Fast object tracking method based on TLD algorithm [J]. Application Research of Computers, 2018, 35(1): 317-320.