Correlation Filter Tracking Based on Multi-Sub-Block Joint Estimation Post-Print

Authors: Cao Jie, Xie Bojiang, Li Wei, Wang Jinhua

Date: 2018-05-20T00:00:00+00:00

Abstract

To address the issue of tracking accuracy degradation in correlation filter algorithms under occlusion, a kernelized correlation filter tracking method based on multi-patch joint estimation is proposed. First, the target is adaptively partitioned according to the geometric features of the tracking bounding box in the initial frame, and each patch is independently tracked using the KCF method to obtain a joint confidence map. Subsequently, the position and scale of the target in the previous frame are employed as prior information for search region sampling, while the weight density of the confidence map within the sample bounding box is utilized as observations; the particle filter algorithm is then applied to achieve optimal estimation of candidate targets. Finally, patches with low confidence are back-projected onto the previous frame image for occlusion detection, thereby preventing erroneous template updates. Both qualitative and quantitative experimental results demonstrate that the proposed method improves tracking accuracy by approximately 10% compared to the original KCF algorithm, exhibits favorable occlusion resistance, and possesses a certain capability for estimating target scale variations.

Full Text

Preamble

Title: Correlation Filter Object Tracking Based on Joint Multi-Block Estimation

Authors: Cao Jiea,b, Xie Bojianga, Li Weib, Wang Jinhuab

Affiliations: aCollege of Computer & Communication; bCollege of Electrical & Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China

Abstract: To address the degradation of tracking accuracy in correlation filter algorithms under occlusion, this paper proposes a kernelized correlation filter tracking method based on joint multi-block estimation. First, the target is adaptively partitioned according to the geometric features of the initial frame bounding box, and each sub-block is independently tracked using the KCF method to obtain a joint confidence map. Then, the position and scale of the target in the previous frame are used as prior information to sample the search region, while the weight density of the confidence map within the sample window serves as the observation value. The particle filter algorithm is employed to achieve optimal estimation of the candidate target. Finally, sub-blocks with low confidence are back-projected onto the previous frame for occlusion detection to prevent erroneous template updates. Qualitative and quantitative experimental results demonstrate that compared with the original KCF algorithm, the proposed method improves tracking accuracy by approximately 10%, exhibits robust occlusion resistance, and possesses certain capability for scale variation estimation.

Keywords: part-based object tracking; correlation filter; particle filter; occlusion detection; scale variation

0 Introduction

Visual object tracking is a hot research topic in computer vision and pattern recognition, playing a vital role in behavior recognition, intelligent surveillance, intelligent transportation, automotive navigation, and human-computer interaction [1]. In recent years, discriminative tracking algorithms based on online learning theory have attracted significant attention from researchers due to their higher tracking accuracy and speed compared with traditional generative methods such as MeanShift [2] and particle filter [3].

Discriminative methods treat object tracking as a typical classification problem—distinguishing the target from the background in an image—and leverage both target and background information during target modeling, thus achieving higher robustness [4]. Correlation filter is a classic discriminative tracking approach. Bolme et al. [5] first introduced the concept of correlation filters to object tracking using least squares training of filter templates, greatly improving tracking real-time performance. The STC method [6] incorporates spatio-temporal context information into correlation filter algorithms in a probabilistic manner and achieves effective scale estimation using a single correlation response confidence map. The CSK method [7] employs kernelized ridge regression to train filter templates, enhancing template effectiveness under nonlinear conditions, and cleverly simplifies the template training process by constructing circulant matrices. The KCF method [8] extends single-channel correlation filters to multi-channel using Gaussian kernel functions on the basis of CSK, enabling more accurate target modeling with complex features and thereby improving tracking accuracy.

Part-based correlation filter tracking methods have become an important means to enhance occlusion resistance, as they can still utilize positioning information from unoccluded parts when the target is partially occluded [9]. Moreover, the high-speed processing capability of correlation filter algorithms makes real-time part-based tracking possible. Reference [10] partitions the target based on its contour, independently tracks sub-blocks using KCF, and determines the overall target position and scale using spatial distribution information among sub-block localization results. Reference [11] performs independent CSK tracking on sub-blocks and uses the peak-to-sidelobe ratio (PSR) of sub-block tracking confidence maps as the criterion for occlusion judgment to update templates. However, existing multi-part-based tracking methods still have limitations. First, they do not consider potential drift in sub-blocks, making it unreasonable to locate the target directly based on the spatial geometric characteristics formed by part-based tracking results. Second, in sub-block template updates, using only the PSR of the confidence map peak as the criterion cannot distinguish between external occlusion and self-deformation.

Therefore, this paper, based on adaptive target partitioning and independent KCF tracking of sub-blocks, employs a particle filter algorithm to sample the multi-block joint confidence map to address the difficulty of target localization and scale estimation in multi-block cue fusion. Particle weights are set according to the weight density formed by the sampling window to achieve optimal estimation of target position and scale. To prevent erroneous template updates due to occlusion, the main component histograms of sub-blocks with low PSR are back-projected onto the target and background regions in the previous frame for occlusion detection.

1 Blocking Strategy and KCF Tracking Principle

1.1 Adaptive Blocking Strategy

Fixed-form image partitioning often cannot flexibly handle the diversity of tracking targets. Since the position and size of the target bounding box in the initial frame are generally manually annotated and can reflect the overall target contour to some extent, we can partition different targets based on the geometric features of the initial frame rectangular bounding box. When the aspect ratio of the initial frame bounding box exceeds a set threshold T_1, it is horizontally divided into two sub-blocks of equal area; when the aspect ratio is below threshold T_2, the bounding box is vertically divided; when the ratio falls between the two thresholds, the bounding box is equally divided into four sub-blocks, as shown in [Figure 1: see original paper].

During tracking of each sub-block, to fully utilize background information while reducing interference from potentially similar targets, this paper restricts the search region of each sub-block to twice the area of the overall tracking bounding box, as illustrated by the dashed box in Figure 1: see original paper.

1.2 KCF Tracking Principle

The KCF tracking method generates a sample set $\{x_1, x_2, ..., x_n\}$ by performing circulant shifts on the image patch x_{t-1} from frame $t-1$, addressing the problem of insufficient training samples in online learning tracking. For different displacement samples of the target, ridge regression is employed to train the optimal filter template w under the criterion of structural risk minimization with mean square loss, minimizing the sum of squared errors in the sample set:

$$\min_w \sum_i ||w^T x_i - y_i||^2 + \lambda ||w||^2 \tag{1}$$

During tracking, the feature space of sample sets is often linearly inseparable. KCF uses a Gaussian kernel function κ to map sample features into a high-dimensional space to make them linearly separable, enhancing the applicability of filter templates under nonlinear conditions. In the kernel space, according to the representation theorem [12], the template w can be represented as a linear combination of all samples:

$$w = \sum_j \alpha_j \varphi(x_j) \tag{2}$$

Substituting this into (1) yields the kernelized ridge regression training process for the filter template:

$$\min_\alpha \sum_i \left(\sum_j \alpha_j \kappa(x_i, x_j) - y_i \right)^2 + \lambda \sum_{i,j} \alpha_i \alpha_j \kappa(x_i, x_j) \tag{3}$$

Defining the kernel function dot product $\kappa(x, x') = \langle \varphi(x), \varphi(x') \rangle$ and solving (3) yields the closed-form solution of the template:

$$\alpha = (K + \lambda I)^{-1} y \tag{4}$$

where K is the kernel matrix composed of elements $\kappa(x_i, x_j)$; I is the identity matrix; and y is a vector composed of elements y_i. Directly solving for coefficient α using (4) involves heavy computation. Reference [7] has proven that the matrix K formed by circulant sample sets under kernel mapping remains circulant. By fully exploiting the special property that any circulant matrix can be diagonalized by Fourier transform matrices, as shown in (5), the training process is simplified:

$$K = C(k) = F \cdot \mathrm{diag}(\hat{k}) \cdot F^H \tag{5}$$

where F is the Fourier coefficient matrix; $\hat{k}$ is a vector composed of elements $\hat{k}_i$; $\text{diag}(\hat{k})$ is a diagonal matrix formed from vector $\hat{k}$; and $C(k)$ is the circulant matrix constructed from vector k. Substituting (5) into (4) and utilizing circulant matrix properties yields:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k} + \lambda} \tag{6}$$

When frame t arrives, the base sample z is extracted centered at the target coordinates in frame $t-1$. Similarly, through circulant shifting and fast Fourier transform to the frequency domain, the response value of each sample is calculated to generate the confidence map of the target location in the current frame, with its maximum value indicating the target position. The response process is shown in (8):

$$\hat{f}(z) = \hat{k}_{xz} \odot \hat{\alpha} \tag{8}$$

where the elements of k_{xz} are $\kappa(x_i, z)$.

2 Proposed Method

The overall framework of the proposed method is shown in [Figure 2: see original paper]. First, the target is adaptively partitioned according to the geometric features of the initial frame bounding box. Then, each sub-block is independently tracked using KCF to generate confidence maps and their respective weights. On this basis, the particle filter algorithm is incorporated to optimally estimate the target position and scale from the multi-block joint confidence map. An occlusion detection module is integrated to enable differentiated template updates for sub-blocks.

2.1 Particle Filter Estimation of Multi-Block Joint Confidence Map

When tracking a target in a part-based manner, it is often assumed that the centroid of the geometric pattern formed by sub-block tracking results represents the target center. However, this approach fails to consider that sub-blocks may drift due to occlusion, making accurate target localization impossible, and cannot estimate target scale variation. Inspired by the single correlation response confidence map scale estimation process in the STC algorithm, this paper expands the state estimation variables and employs a particle filter algorithm to estimate the target size and center coordinates from the multi-block joint confidence map.

Independent KCF tracking of target sub-blocks yields confidence maps $C_{j,t}$ for each sub-block. According to reference [13], the reliability of tracking results can be evaluated by the peak-to-sidelobe ratio (PSR) of each sub-block' s confidence map. The normalized weight of each sub-block is calculated as:

$$w_{j,t} = \frac{\mathrm{PSR}_{j,t}}{\sum_{j=1}^{M} \mathrm{PSR}_{j,t}} \tag{9}$$

where $\mathrm{PSR}_{j,t} = \frac{G_{j,t} - \mu_{j,t}}{\sigma_{j,t}}$, with $G_{j,t}$ being the peak value of confidence map j, and $\mu_{j,t}$ and $\sigma_{j,t}$ being the mean and standard deviation of the sidelobe region.

Considering that within the same sub-block confidence map, weights should gradually decrease from the maximum peak to surrounding areas, and that target scale variation may cause overlapping boundaries among multiple sub-blocks leading to layered confidence maps, a cosine window h is introduced during multi-block joint confidence map computation to mitigate edge effects. The window center depends on the maximum peak location in the corresponding sub-block confidence map, and its range is determined by the sub-block target size. This ensures higher weights at the center and weights approaching zero at edges. The multi-block joint confidence map is:

$$C_t = \sum_{j=1}^{M} w_{j,t} \cdot (C_{j,t} \odot h_j) \tag{10}$$

where the sub-block target size in cosine window h_j is determined by the overall scale estimation result from frame $t-1$ and the initial frame blocking strategy.

The particle filter tracking method first establishes a state transition equation incorporating target motion affine information to predict and constrain particle states, then uses a likelihood function constructed from the latest observations to set particle weights, moving them from low-likelihood to high-likelihood regions. In the state-space model, the following state transition equation and likelihood function are constructed:

$$x_t = Ax_{t-1} + v_t \tag{11}$$

$$p(y_t|x_t) = \frac{1}{\sqrt{2\pi}\delta} \exp\left(-\frac{\rho^2}{2\delta^2}\right) \tag{12}$$

where $x_t = [x, y, w, h]^T$ represents the hidden state at time t, corresponding to the center coordinates and scale of the candidate target bounding box; A is the transition matrix composed of affine transformations; and v_t is the Gaussian noise covariance matrix corresponding to the state equation.

The computation of ρ is shown in (13). Let $M(x_t)$ denote the weight enclosed by the tracking box corresponding to state x_t in the multi-block joint confidence map, and $S(x_t)$ be the area of the tracking box. Then the weight density of the confidence map within the tracking box is:

$$\rho = \frac{M(x_t)}{S(x_t)} \tag{13}$$

According to Monte Carlo principles and the law of large numbers, the posterior probability of the true target state can be approximated by the normalized weights of N sampled particles. The weight update process using the suboptimal proposal distribution $q(x_t|x_{t-1}, y_t)$ is:

$$\omega_t^{(i)} \propto \omega_{t-1}^{(i)} \frac{p(y_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{t-1}^{(i)}, y_t)} \tag{14}$$

The optimal estimate of the target state at time t according to the maximum a posteriori criterion is:

$$\hat{x}_t = \sum_{i=1}^{N} \omega_t^{(i)} x_t^{(i)} \tag{15}$$

The particle filter estimation process for the multi-block joint confidence map is as follows:

1. **Particle Initialization**: Set particle number N and generate random sampling particle set $\{x_1^{(i)}\}_{i=1}^N$ based on prior target state information, with corresponding weights $\omega_1^{(i)} = 1/N$.
2. **Multi-Block Confidence Map Fusion**: Generate the joint confidence map C_t using (10) based on independent tracking confidence maps $C_{j,t}$ and their weights $w_{j,t}$, combined with window function h_j.
3. **State Prediction**: Perform importance sampling on the one-step state transition equation (11) to obtain predicted particle states $x_t^{(i)}$ at time t.
4. **Particle Weight Update**: Calculate the confidence map weight density $\rho_t^{(i)}$ enclosed by each particle' s predicted state using (13), and obtain the corresponding particle weights $\omega_t^{(i)}$ at time t using (12) and (14).
5. **Target State Estimation**: Normalize $\omega_t^{(i)}$ and estimate the true target state using (15) to obtain the optimal estimate of candidate target position and scale.
6. **Resampling**: Randomly select N particles from the particle state set $\{x_t^{(i)}\}_{i=1}^N$ according to $\omega_t^{(i)}$ to generate the resampled particle set $\{\tilde{x}_t^{(i)}\}_{i=1}^N$.

2.2 Occlusion-Based Sub-Block Template Update

Both occlusion and self-deformation can cause degradation in sub-block tracking confidence. However, when target appearance changes due to external occlusion, the occluding object often originates from the background region, whereas self-deformation more likely originates from the target region. Therefore, when sub-block tracking confidence is low, this paper back-projects the sub-block onto the search region in the previous frame to obtain the probability of pixels belonging to target or background regions, thereby detecting whether the target is occluded.

If sub-block q has confidence below threshold at frame t, its normalized histogram is:

$$H_q^t(u) = \frac{1}{n}\sum_{i=1}^{n} \delta[b(C_q^t(x_i, y_i)) - u] \tag{17}$$

where n is the total number of pixels in sub-block q; $C_q^t(x_i, y_i)$ represents the pixel value at location (x_i, y_i) in sub-block q at frame t (hue component in HSV space for color targets, grayscale value for gray targets); and δ is the Kronecker delta function.

In practice, since sub-block histograms have many components, directly using the full histogram for back-projection is computationally expensive and susceptible to background component interference. Considering that occluding objects and targets occupy large proportions in the search region with slowly changing pixel values, we select components with large histogram values and concentrated spatial distribution for back-projection. We choose the 5 largest components from histogram H_q^t to form $\hat{H}_q^t$ and calculate their spatial dispersion σ_u:

$$\sigma_u = \frac{1}{m_u}\sum_{(x,y)\in u} ||(x, y) - \mu_u||^2 \tag{18}$$

where m_u is the total number of pixels belonging to component u, and μ_u is the coordinate of the target location. Based on σ_u, we select 3 components with smaller spatial dispersion (more concentrated distribution) from $\hat{H}_q^t$ to form $\tilde{H}_q^t$, and back-project them onto the search region in the previous frame to obtain the pixel probability distribution map $I_{t-1}(x, y)$:

$$I_{t-1}(x, y) = \sum_{u\in \tilde{H}_q^t} \delta[b(I_{t-1}(x, y)) - u] \tag{19}$$

According to the tracking box position in frame $t-1$, we determine the target and search regions in the back-projection map and count the number of foreground pixels in the target region A_o and the entire search region A_s. The probability of pixels in sub-block q being back-projected to the target region is:

$$P_q = \frac{A_o}{A_s} \tag{20}$$

If P_q is less than the occlusion 判定阈值 th_P, it indicates that target pixels in the sub-block have a higher probability of originating from the background region in the previous frame, meaning background occlusion occurs in this sub-block, and its template update should be stopped. Otherwise, the template is updated normally. The differentiated template update process for sub-blocks is:

$$\hat{H}_q^t = \begin{cases} \hat{H}_q^{t-1} \cdot (1-\eta) + \hat{H}_q^t \cdot \eta & \text{if } P_q > th_P \\ \hat{H}_q^{t-1} & \text{if } P_q \leq th_P \end{cases} \tag{21}$$

3 Experiments and Analysis

3.1 Experimental Parameters and Evaluation Metrics

The hardware platform for experiments is configured as: 4.2 GHz quad-core Core i7, Windows 7 + MATLAB R2014a. The blocking thresholds T_1 and T_2 are empirical values obtained by traversing the video database based on the aspect ratio of the initial frame target, set to 1.6 and 0.4 respectively in experiments. The kernelized correlation filter parameters are identical to KCF. In the target state estimation process, the particle number is 50, and the state noise follows a Gaussian distribution: $x_t \sim N(0, \text{diag}(0.15, 0.15, 0.01, 0.01))$. The occlusion 判定阈值 th_P is 0.8.

To fully evaluate the effectiveness of the proposed method, we adopt the standard evaluation system established in reference [15] and compare our method with several popular tracking algorithms using center location error (CLE), distance precision (DP), and overlap precision (OP) as metrics. CLE is the Euclidean distance between the algorithm-labeled target center and the ground truth. DP is the ratio of frames where CLE is below a certain threshold (20 pixels) to the total video sequence length, reflecting the center positioning accuracy during tracking. OP is the ratio of frames where the overlap region (OR) score exceeds a set threshold (0.5) to the total sequence length, reflecting how well the tracking box fits the target size. The OR score is calculated as:

$$\text{score} = \frac{\text{area}(B_t \cap B_g)}{\text{area}(B_t \cup B_g)} \tag{22}$$

where B_t is the tracking box at frame t, and B_g is the corresponding ground truth box.

3.2 Qualitative Analysis

The qualitative analysis experiment selected 2,960 frames from the Benchmark [15] test set, containing common challenging scenarios such as occlusion and scale variation. The proposed method was compared with classic correlation filter algorithms including KCF and CSK on the same video sequences.

Occlusion Scenario Analysis: Videos (a)-(d) in [Figure 5: see original paper] all exhibit different degrees of target occlusion. Under normal conditions, all three algorithms can effectively track the original target. When the target undergoes large-area occlusion, CSK, which uses only simple grayscale features for modeling, is easily disturbed by occluding objects with similar grayscale, leading to significant tracking box drift, as shown in frame 432 of the *girl* video. The KCF algorithm demonstrates certain robustness to short-term occlusion through its dense discriminative strategy using multi-channel local feature samples. However, due to its non-differentiated template update strategy, it is prone to erroneously incorporating occluding objects into the target template, causing tracking drift. In contrast, our algorithm can accurately locate the target using information from unoccluded sub-blocks even under large-area occlusion, as demonstrated in the *freeman4* and *girl* videos. When the target is completely occluded, all three methods lack predictive capability for target motion information, causing tracking boxes to lock onto the occluding object with the highest confidence peak. CSK and KCF lack occlusion detection mechanisms, making it difficult to recapture the target when it reappears. Our algorithm, employing occlusion detection for template updates, can effectively suppress target drift and achieve re-capture, as shown in the *football* and *jogging2* videos.

Scale Variation Scenario Analysis: Videos (e) and (f) in [Figure 5: see original paper] show obvious target scale variation. In addition to our method and KCF, we include STC and DSST [16]—correlation filter algorithms with scale estimation capability—for comparison. The results show that KCF, which does not consider target scale variation, maintains a constant tracking box size, easily introducing excessive background information or failing to capture the complete target features, leading to tracking failure, as seen in the *walking2* and *carscale* sequences. STC estimates only the target response confidence map and can sense scale variation well when the confidence map is reliable, but fails to effectively estimate scale when occlusion makes the confidence map unreliable. DSST constructs a multi-scale sample set of target appearance using feature pyramids and employs a scale correlation filter to select the optimal scale sample. However, its scale estimation process is easily disturbed by occluding objects. Our method performs joint estimation on multi-block confidence maps and their respective weights, which can alleviate occlusion effects to some extent, as shown in frames 203-283 of the *walking2* video.

Figure 6: see original paper-(b) show the per-frame center error curves for occlusion videos. Combined with [Figure 5: see original paper], when targets experience different degrees of occlusion (corresponding to frames 50, 270, 80,

and 432 of each video sequence), CSK and KCF tracking errors increase significantly, while our method maintains better tracking accuracy. However, due to the random sampling nature of the particle filter method for target localization, our approach exhibits more small fluctuations in tracking error compared to the other two methods. Figure 6: see original paper-(f) show the overlap rate curves for scale variation scenarios. Although our scale estimation method is slightly inferior to the advanced DSST, it can still maintain high overlap rates when partial occlusion occurs (frame 283 in *walking2*).

3.3 Quantitative Analysis

presents quantitative comparison of our method with several classic algorithms (KCF, CSK, STC, DSST, Struck [17]) on 12 video sequences from Benchmark containing occlusion or scale variation, using center error as the metric.

The results show that our method achieves smaller tracking errors on most video sequences and maintains good tracking accuracy in scenes with occlusion and scale variation such as *Jogging*, *Girl2*, and *Walk2*. However, our method does not achieve optimal performance on all videos (e.g., *ClifBar*). Analysis reveals that in such videos, the target state changes frequently and significantly. With increasing video frames, the particle filter algorithm may suffer from particle degradation during extensive iterations, leading to ineffective target state estimation. In terms of real-time performance, our method incorporates particle filter confidence map estimation and occlusion detection modules on top of part-based tracking. While this improves tracking accuracy in occlusion scenarios, it also increases computational complexity, resulting in an overall average speed of 38 frames per second.

Figure 7: see original paper-(b) show the overall distance precision and overlap precision score curves for all algorithms. It can be observed that our method improves distance precision by approximately 10% and 3% compared with KCF and DSST respectively, while overlap precision is slightly lower than DSST.

4 Conclusion

Based on kernelized correlation filter tracking, this paper investigates multi-block joint confidence map estimation and template update. First, a multi-block joint confidence map is generated based on each sub-block' s confidence map and its weight. Then, the particle filter algorithm samples the joint confidence map, and particle weights are set according to the weight density enclosed by sample windows to achieve optimal estimation of target position and scale. For sub-block template updates, back-projection is used to map low-confidence sub-blocks onto target and background regions for occlusion detection, preventing tracking drift caused by erroneous template updates.

Validation on public test sets through qualitative and quantitative analysis demonstrates that the proposed method maintains high tracking accuracy in scenarios with partial occlusion and scale variation. However, during particle filter estimation of the confidence map, particle degradation may occur during extensive iterations, leading to increased target state estimation error (as shown in Figure 6: see original paper). Future work will investigate more effective resampling strategies to address this issue.

References

[1] Ramalakshmi V, Alex M G. Visual object tracking using discriminative correlation filter [C]// Proc of International Conference on Communication and Electronics Systems. 2016: 1-6.

[2] Mehmood R, Nawaz R, Rao N I. Occlusion handling in meanshift tracking using adaptive window normalized cross correlation [C]// Proc of International Bhurban Conference on Applied Sciences and Technology. 2014: 126-129.

[3] 李伟, 曹洁, 李军, 等. 考虑噪声扰动干扰的室内行人跟踪方法 [J]. 四川大学学报: 工程科学版, 2016, 48(6): 172-179.

[4] Smeulders A W, Chu D M, Cucchiara R, et al. Visual tracking: an experimental survey [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2014, 36(7): 1442-68.

[5] Bolme D S, Beveridge J R, Draper B A, et al. Visual object tracking using adaptive correlation filters [C]// Proc of Computer Vision and Pattern Recognition. 2010: 2544-2550.

[6] Zhang K, Zhang L, Liu Q, et al. Fast visual tracking via dense spatio-temporal context learning [C]// Proc of European Conference on Computer Vision. [S. l.]: Springer International Publishing, 2014: 127-141.

[7] Henriques J F, Rui C, Martins P, et al. Exploiting the circulant structure of tracking-by-detection with kernels [C]// Proc of European Conference on Computer Vision. 2012: 702-715.

[8] Henriques J F, Caseiro 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.

[9] Yao R, Xia S, Shen F, et al. Exploiting spatial structure from parts for adaptive kernelized correlation filter tracker [J]. IEEE Signal Processing Letters, 2016, 23(5): 658-662.

[10] 段伟伟, 杨学志, 方帅, 等. 分块核化相关滤波目标跟踪 [J]. 计算机辅助设计与图形学学报, 2016, 28(7): 1160-1168.

[11] 王暐, 王春平, 付强, 等. 基于分块的尺度自适应 CSK 跟踪算法 [J]. 电光与控制, 2017(2): 25-29.

[12] Williams C K I. Learning with kernels: support vector machines, regularization, optimization, and beyond [J]. IEEE Trans on Neural Networks, 2005, 16(3): 781-781.

[13] Lee K, Park S H, Yoon K J. PSR-deterministic search range penalization method on kernelized correlation filter tracker [C]// Proc of International Conference on Ubiquitous Robots and Ambient Intelligence. 2016: 856-860.

[14] 王法胜, 鲁明羽, 赵清杰, 等. 粒子滤波算法 [J]. 计算机学报, 2014, 37(8): 1679-1694.

[15] Wu Y, Lim J, Yang M H. Object tracking benchmark [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2015, 37(9): 1834.

[16] Danelljan M, Häger G, Khan F S, et al. Accurate scale estimation for robust visual tracking [C]// Proc of British Machine Vision Conference. 2014: 65.

[17] Hare S, Saffari A, Torr P H S. Struck: structured output tracking with kernels [C]// Proc of IEEE International Conference on Computer Vision. 2012: 263-270.

Note: Figure translations are in progress. See original paper for figures.