PAPER

Real-Time Sparse Visual Tracking Using Circulant Reverse Lasso Model

Chenggang GUO[†], Student Member, Dongyi CHEN^{†a)}, and Zhiqi HUANG[†], Nonmembers

SUMMARY Sparse representation has been successfully applied to visual tracking. Recent progresses in sparse tracking are mainly made within the particle filter framework. However, most sparse trackers need to extract complex feature representations for each particle in the limited sample space, leading to expensive computation cost and yielding inferior tracking performance. To deal with the above issues, we propose a novel sparse tracking method based on the circulant reverse lasso model. Benefiting from the properties of circulant matrices, densely sampled target candidates are implicitly generated by cyclically shifting the base feature descriptors, and then embedded into a reverse sparse reconstruction model as a dictionary to encode a robust appearance template. The alternating direction method of multipliers is employed for solving the reverse sparse model and the optimization process can be efficiently solved in the frequency domain, which enables the proposed tracker to run in real-time. The calculated sparse coefficient map represents the similarity scores between the template and circular shifted samples. Thus the target location can be directly predicted according to the coordinates of the peak coefficient. A scale-aware template updating strategy is combined with the correlation filter template learning to take into account both appearance deformations and scale variations. Both quantitative and qualitative evaluations on two challenging tracking benchmarks demonstrate that the proposed algorithm performs favorably against several state-of-the-art sparse representation based tracking methods.

key words: visual tracking, sparse representation, sparse tracker, reverse lasso model, correlation filter template

1. Introduction

Visual tracking aims to online estimate the trajectory of the object of interest in a video sequence, where the object is identified by a bounding box in the first frame. Since the lack of prior information about the object of interest, the online nature of tracking, and the time-varying noises like object deformations and appearance variations, it still remains a very challenging problem in the field of real-time vision analysis.

In recent years, sparse representation theory has been introduced into the field of visual tracking [1]–[20]. In the context of sparse tracking, most trackers maintain an online updated appearance dictionary and represent target candidates as a sparse linear combination of appearance templates drawing from the dictionary. The target candidate that best matches with the appearance dictionary is chosen as the

tracking result of current frame. Nevertheless, due to complex and dynamic factors (such as 11 attributes listed by Wu et al. [21]) in real world tracking scenarios, appearance template that is crucial to a sparse tracker becomes unreliable as time goes on. To deal with appearance degradation during tracking, a number of methods have been proposed, such as holistic and local joint representations [6], [12]–[14], temporal, spatial or structural information [7], [10], [16] and multi-level grouping [5]. All these sparse methods try to alleviate the observation model degradation problem to some extent. In contrary to the trend of designing different robust observation models, recent sparse trackers mostly choose particle filter framework [4]–[7], [10]–[14], [16] as the motion model. Sparse trackers utilize particles sampled from a Gaussian distribution to describe the motion states of an object over time. Each particle represents a possible target candidate sample. Learning the representation of each particle is treated as a repeat sparse coding problem.

However, complex feature representations and expert priors in tracking mechanism designing hinder the further improvement of the performance of sparse trackers. The drawbacks of previous sparse trackers are three-fold: First, existing sparse trackers remain computationally expensive and cannot achieve real-time tracking, especially for trackers formulated under the particle filter framework. As mentioned above, sparse trackers rely heavily on robust appearance descriptors and thus inevitably suffer from high computation cost. It would be made even worse if sparse trackers relatively densely sample particles based on previous states since appearance features for each particle need to be recalculated. Second, dictionary online learning and updating often rely on carefully designed updating mechanisms to account for changes in appearance. For example, several sparse trackers [5], [6], [19] divide the template dictionary into a positive set and a negative set that each of them is handled separately. But accumulated errors from partial occlusion or other factors during tracking may gradually reduce the quality of the dictionary. Third, the final state prediction is made by all sampled particles, but the sample space is quite sparse compared to the full state space, resulting to inaccurate positioning.

In this paper, we propose an improved real-time sparse tracker based on circulant reverse lasso model that aims to address the above drawbacks. Within the circulant reverse lasso model, appearance templates are encoded by circulant structured target candidates, which is different from the general sparse model. By regarding the set of circular

Manuscript received July 11, 2018.

Manuscript revised September 4, 2018.

Manuscript publicized October 9, 2018.

[†]The authors are with the School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China.

a) E-mail: dychen@uestc.edu.cn

DOI: 10.1587/transinf.2018EDP7248

Fig. 1 Flowchart of the proposed tracking algorithm based on the circulant reverse lasso model. The circulant target candidate set is adopted as a dictionary to encode a single template. The calculated sparse coefficient map represents the similarity scores between the template and the samples. The peak in the coefficient map gives the desired target location.

shifted samples as dictionary atoms, redundant information naturally exists in this dictionary since only one atom best matches the updated target template. The peak sparse coefficient corresponds to the best matched candidate-template pair and thus directly reflects the best location prediction. The circulant structure is adopted here to take place of the particle filter framework. It serves as the core of recently start-of-the-art discriminative correlation filter trackers. Benefiting from the property of circulant structure, complex feature descriptors can be reused for dense sampling such that computation cost would not grow linearly with the number of particles. As for the template learning and updating, we take advantage of the correlation filter to learn a robust correlation template for the reverse lasso model. Templates are handled in an additive way while take into account the current target size. Only appearance features that represent the tracked target are preserved. We maintain just one template so that the reverse lasso model is solved for only once. The flowchart of the proposed tracking algorithm is illustrated in Fig. 1.

The contributions of this work are also three-fold: First, circulant structure is introduced into the reverse sparse representation model to take place of particle filter framework. Target candidates can be implicitly generated and embedded into the sparse representations with circulant structure. Second, correlation filter template is exploited to represent a robust appearance template against partial occlusion and appearance degradation in the context of sparse representation theory. Third, a real-time sparse tracker with promising tracking performance is proposed to further boost sparse representation based visual tracker.

2. Related Work

Visual tracking has been studied extensively in recent years [3], [22], [23], and the tracking performance has wit-

nessed a continual promotion especially after the release of well-labeled visual tracking benchmarks [21], [24], [25]. In this section, two kinds of main stream trackers that closely related to the proposed method are briefly overviewed.

A. sparse representation based trackers:

Sparse representation theory is first introduced into the field of visual tracking by Mei et al. [1]. According to the detailed discussion from Zhang et al. [7], previous sparse representation based trackers can be categorized based on global, local, and joint sparse appearance models. Sun et al. [26] exploited a fragment-based appearance model with consideration of both temporal continuity and discontinuity information. Multi-task sparse learning problem [2], [12], [14] are discussed to learn the sparse representations of all candidates jointly. In [12], particle representations are jointly learned via the use of joint sparsity. The same sparse dictionary templates are used to represent all particles draw from each frame. Furthermore, the cues from multi-view including various types of visual features are considered in a multi-task formulation [14]. Fan et al. [2] proposed to jointly learn multiple dictionaries instead of a single dictionary for multi-task sparse representation problem, which may enforce robustness in coefficient estimation. In the work of [7], the proposed structural sparse appearance model incorporates the properties of the above three approaches and emphasis on the spatial layout structure among local patches inside each target candidate region. Similar ideas of exploiting spatial context information are also integrated into a unified sparse framework in [16].

To improve the efficiency of sparse trackers, Mei et al. [27] proposed to calculate the minimum error bound which serves as a guide for particle resampling. Bao et al. [9] proposed a fast l_1 tracker based on the accelerated proximal gradient approach with guaranteed quadratic convergence. Zhang et al. [13] cast the tracking problem as a low-rank matrix learning problem that can be efficiently solved as a sequence of closed form update operations. Circulant structure is firstly introduced into the sparse representation based trackers by Zhang et al. [4]. In the proposed circulant sparse tracker (CST) [4], the target templates are circulant and the representation of each particle is viewed as a sparse encoding problem. Our work is different from CST in that the additive target template is encoded by circulant target candidates. Therefore, the proposed sparse tracker only need to perform one sparse encoding procedure, which further reduces the computation cost.

In parallel with the use of conventional sparse representation model, reverse sparse representation has been adopted in the field of visual tracking. Wang et al. [28] proposed a locally weighted distance metric combined with an inverse sparse representation to facilitate robustness against partial noises during object tracking. Compared to [28], the proposed method does not explicitly handle local parts and generates candidate samples through a circular shift operation. Zhou et al. [5] proposed an inverse sparse group lasso model to exploit both the group structure inside candidate samples and local structure between templates and samples.

Zhuang et al. [19] constructed a multitask reverse sparse representation formulation to search subsets of whole candidate set with minimum reconstruction error. Yi et al. [11] proposed a method that combined particle filter and reverse sparse representation to reduce drifting. But these methods formulated a complex reverse sparse representation model that patch candidates are sampled under particle filter framework. In the proposed method, we believe that the circulant structured target candidates are more suitable to be combined with reverse sparse representation model.

B. correlation filter based trackers:

From the competition results on recently released tracking benchmarks [24], [25], discriminative correlation filters (DCF) based tracking methods have achieved very promising performance in terms of accuracy and robustness. Generally, a correlation filter is discriminately learned from a set of circular shifted training samples. The shifted samples are implicitly generated in the frequency domain through the circulant property of correlation. Danelljan et al. proposed a continuous convolution formulation [29] to enable efficient integration of multi-resolution feature maps. Then they further proposed a factorized convolution operator [30] to reduce the former model parameters. The above two methods both inherit the idea of channel and spatial reliability proposed in [31] and [32]. Sun et al. [33] exploited the complementary kernelized ridge regression and convolution neural network for tracking. Fan et al. [34] utilized both local-based and global-based strategies for motion models to preserve the structure of object. Zhang et al. [35] exploited the complementary strengths of correlation filters and particle filters. These methods all achieved top ranks in the VOT 2017 challenge [24]. In this paper, the basic DCF formulation without spatial regularization is adopted to learn the target template for sparse representation. By imposing sparse constraints on the reverse representation model, we find that solving sparse constrained model using correlation filter template is more robust than directly adopting cross correlation in the decision stage.

3. Circulant Reverse Lasso Tracking Model

In this section, we first briefly review the general sparse representation based model that most existing sparse trackers follows. Then the proposed circulant reverse lasso (CRL) tracking model is described in detail. The robust template learning and updating strategy is stated at last.

3.1 General Sparse Tracking Model

In the standard setting of visual tracking, a tracker is provided with a sequence of video frames, and the object to be tracked is annotated in the first frame with a ground truth bounding box. Under the general particle filter framework, N particles on the j^{th} frame are randomly sampled around the current state of the object according to a zero-mean Gaussian distribution. The feature descriptors of the particles are denoted in matrix form as $\mathbf{Y}^{(j)} =$

$[\mathbf{y}_1^{(j)}, \mathbf{y}_2^{(j)}, \dots, \mathbf{y}_N^{(j)}] \in \mathbb{R}^{d \times N}$, where each column is a d dimensional particle description in $\mathbb{R}^d$. The general sparse representation based tracking model seeks to represent each particle $\mathbf{y}_i^{(j)}$ by a sparse combination $\mathbf{c}_i$ of templates that form an appearance dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_M] \in \mathbb{R}^{d \times M}$, where M denotes the number of templates or atoms in the dictionary. The solving process can be formulated as the following l_1 regularized linear regression minimization problem:

$$\tilde{\mathbf{c}}_i = \arg \min_{\mathbf{c}_i} \|\mathbf{y}_i^{(j)} - \mathbf{D}\mathbf{c}_i\|_2^2 + \lambda \|\mathbf{c}_i\|_1 \quad (1)$$

where $\tilde{\mathbf{c}}_i \in \mathbb{R}^M$ denotes the resulting sparse coefficient vector for particle i . By repeating Eq. (1) for each particle, we get the sparse coefficient matrix $\tilde{\mathbf{C}} = [\tilde{\mathbf{c}}_1, \tilde{\mathbf{c}}_2, \dots, \tilde{\mathbf{c}}_N] \in \mathbb{R}^{M \times N}$. Then the reconstruction process can be calculated by $\tilde{\mathbf{Y}}^{(j)} = \mathbf{D}\tilde{\mathbf{C}}$. The particle with the least reconstruction error or the highest similarity with the dictionary is chosen as the tracking result for the j^{th} frame.

3.2 Proposed Sparse Tracking Model

In this paper, we adopt the reverse sparse representation model. Different from the general sparse tracking model, the appearance dictionary here is composed of a circulant structured candidate set $\mathbf{Y}^{(j)} = [\mathbf{y}^{(j)}[\Delta\tau_1], \mathbf{y}^{(j)}[\Delta\tau_2], \dots, \mathbf{y}^{(j)}[\Delta\tau_e]] \in \mathbb{R}^{e \times e}$, where $[\Delta\tau_i]$ means the circular shift operator, and $\mathbf{y}^{(j)}[\Delta\tau_i]$ means a i -step discrete circular shift to the base candidate descriptor $\mathbf{y}^{(j)} \in \mathbb{R}^e$. $\mathbf{y}^{(j)}$ is extracted on a local image patch of the j^{th} frame, which is centered on the $(j-1)^{th}$ predicted coordinates. The appearance template $\mathbf{t}^{(j)} \in \mathbb{R}^e$ is thus sparse decomposed by the target candidate set $\mathbf{Y}^{(j)}$:

$$\tilde{\mathbf{c}} = \arg \min_{\mathbf{c}} \frac{1}{2e} \|\mathbf{t}^{(j)} - \mathbf{Y}^{(j)}\mathbf{c}\|_2^2 + \lambda \|\mathbf{c}\|_1 \quad (2)$$

where $\mathbf{t}^{(j)}$ is the updated template for the j^{th} frame, and λ controls the coefficient sparsity. Solving Eq. (2) directly is infeasible with a high dimension e . To deal with this issue, the virtual samples generated by the circular shift operator is implicitly handled in the frequency domain. More specifically, an auxiliary variable $\mathbf{b}$ is introduced to make an equality constraint $\mathbf{b} - \mathbf{c} = 0$ for Eq. (2). Then the augmented Lagrange multiplier is introduced to incorporate the equality constraint into the objective function. The augmented Lagrangian for equality-constrained Eq. (2) can be formed as:

$$\begin{aligned} (\tilde{\mathbf{c}}, \tilde{\mathbf{b}}, \tilde{\mathbf{u}}) = \arg \min_{\mathbf{c}, \mathbf{b}, \mathbf{u}} & \frac{1}{2e} \|\mathbf{t}^{(j)} - \mathbf{Y}^{(j)}\mathbf{c}\|_2^2 + \lambda \|\mathbf{b}\|_1 \\ & + \mathbf{u}^T(\mathbf{c} - \mathbf{b}) + \frac{\rho}{2} \|\mathbf{c} - \mathbf{b}\|_2^2 \end{aligned} \quad (3)$$

where $\rho > 0$ is the penalty parameter, $\mathbf{u}$ is the Lagrange multiplier and $\mathbf{b}$ is the auxiliary variable. Equation (3) can be decomposed into a series of subproblem updates according to the alternating direction method of multipliers

(ADMM) [36], and each of the subproblems, $\tilde{\mathbf{c}}$ and $\tilde{\mathbf{b}}$, have closed form solutions:

Subproblem c: given $(\mathbf{b}, \mathbf{u})$, $\mathbf{c}$ is updated by solving the optimization problem (4) with the solution (5):

$$\mathbf{c} = \arg \min_{\mathbf{c}} \frac{1}{2e} \|\mathbf{t}^{(j)} - \mathbf{Y}^{(j)} \mathbf{c}\|_2^2 + \mathbf{u}^T \mathbf{c} + \frac{\rho}{2} \|\mathbf{c} - \mathbf{b}\|_2^2 \quad (4)$$

$$= \left(\frac{1}{e} \mathbf{Y}^{(j)T} \mathbf{Y}^{(j)} + \rho \mathbf{I} \right)^{-1} \left(\frac{1}{e} \mathbf{Y}^{(j)T} \mathbf{t}^{(j)} + \rho \mathbf{b} - \mathbf{u} \right) \quad (5)$$

Subproblem b: given $(\mathbf{c}, \mathbf{u})$, $\mathbf{b}$ is updated by solving the optimization problem (6) with the solution (7):

$$\mathbf{b} = \arg \min_{\mathbf{b}} \lambda \|\mathbf{b}\|_1 - \mathbf{u}^T \mathbf{b} + \frac{\rho}{2} \|\mathbf{c} - \mathbf{b}\|_2^2 \quad (6)$$

$$= S_{\lambda/\rho} \left(\mathbf{c} + \frac{\mathbf{u}}{\rho} \right) \quad (7)$$

Update u: given $(\mathbf{c}, \mathbf{b})$, Lagrange multiplier $\mathbf{u}$ is updated with the solution (8):

$$\mathbf{u} = \mathbf{u} + \rho (\mathbf{c} - \mathbf{b}) \quad (8)$$

where $S_k(a) = \text{sign}(a)(|a| - k)_+$ is the coordinate-wise soft thresholding operator. According to the desirable property of circulant matrices, circulant structured candidate set $\mathbf{Y}$ can be diagonalized by the unitary discrete Fourier transform (DFT) matrix $\mathbf{U}$, and the eigenvalues are simply the DFT of the base candidate descriptor $\mathbf{y}$: $\mathbf{Y} = \mathbf{U} \text{diag}(\hat{\mathbf{y}}) \mathbf{U}^H$. Applying this property, we reformulate Eq. (5) in the frequency domain:

$$\mathbf{c} = \mathcal{F}^{-1} \left\{ \frac{\frac{1}{e} \sum_k \hat{\mathbf{y}}_k^{(j)*} \circ \hat{\mathbf{t}}_k^{(j)} + \rho \hat{\mathbf{b}} - \hat{\mathbf{u}}}{\frac{1}{e} \sum_k \hat{\mathbf{y}}_k^{(j)*} \circ \hat{\mathbf{y}}_k^{(j)} + \rho} \right\} \quad (9)$$

where the operators $\circ$ and $\div$ denote elementwise multiplication and elementwise division, respectively. The variable with a hat $\hat{\cdot}$ denotes the DFT of the vector: $\hat{x} = \mathcal{F}\{x\} = Fx$, and the constant matrix F is known as the DFT matrix. The $\mathcal{F}^{-1}$ denotes the inverse DFT. Here the multi-channels case is given in the Eq. (9) with each feature channel of the base candidate in the frequency domain denoted as $\hat{\mathbf{y}}_k$. As such, the computation cost for matrix multiplication in the Eq. (5) can be greatly reduced by performing elementwise multiplication in the frequency domain in the Eq. (9), while the only additional cost is the process of performing 2D DFT.

3.3 Robust Template Learning and Scale-Aware Updating

It can be found that the subproblem $\mathbf{c}$ is essentially a ridge regression that shares a lot common with correlation filtering methods. In fact, we find that the template learned from a correlation filter is very robust and suitable for the sparse tracking. The correlation filter template can be learned by minimizing the following objective [37]:

$$\tilde{\mathbf{t}} = \arg \min_{\mathbf{t}} \frac{1}{2e} \|\mathbf{r} - \mathbf{Y}^{(j)T} \mathbf{t}\|_2^2 + \frac{\eta}{2} \|\mathbf{t}\|_2^2 \quad (10)$$

where $\mathbf{r} \in \mathbb{R}^e$ is the desired correlation response, $\mathbf{Y}^{(j)}$ is a circulant matrix generated by the feature descriptor extracted

Algorithm 1 CRL-based visual tracking algorithm

Input: frame j ; previous predicted coordinates $(x^{(j-1)}, y^{(j-1)})$; correlation filter template $\mathbf{t}^{(j)}$; parameter λ, ρ, η and ζ ; number of ADMM iterations $iter$.

Output: newly predicted coordinates (x^j, y^j) , newly updated template $\mathbf{t}^{j+1}$.

- 1: Extract the feature descriptor $\mathbf{y}^{(j)}$ of local image patch centered on $(x^{(j-1)}, y^{(j-1)})$;
- 2: **for** $i = 1$ to $iter$ **do**
- 3: Solve Subproblem $\mathbf{c}$ using Eq. (9);
- 4: Solve Subproblem $\mathbf{b}$ using Eq. (7);
- 5: Update Lagrange multiplier $\mathbf{u}$ using Eq. (8);
- 6: **end for**
- 7: Find the peak score (x^j, y^j) in the coefficient vector $\mathbf{c}$;
- 8: Update template $\mathbf{t}^{j+1}$ using Eq. (11) and Eq. (12);
- 9: **return** $(x^j, y^j), \mathbf{t}^{j+1}$.

on the j^{th} frame according to the j^{th} predicted coordinates, and η is a regularization parameter. The solution to Eq. (10) can be also computed efficiently in the Fourier domain:

$$\mathbf{t} = \mathcal{F}^{-1} \left\{ \frac{\frac{1}{e} \hat{\mathbf{y}}^{(j)} \circ \hat{\mathbf{r}}}{\frac{1}{e} \sum_k \hat{\mathbf{y}}_k^{(j)*} \circ \hat{\mathbf{y}}_k^{(j)} + \eta} \right\} \quad (11)$$

It is worth noting that the linear regression objective $\mathbf{r} \leftarrow \mathbf{Y}^{(j)T} \mathbf{t}$ in Eq. (10) also appears in the Eq. (5). Therefore, it exists an inner connection between the correlation template and the proposed reverse lasso tracking model. The final template $\mathbf{t}^{(j+1)}$ for the $(j+1)^{th}$ frame is updated dynamically to handle rapid changes both in target appearance and target size:

$$\mathbf{t}^{(j+1)} = \mathbf{t}^{(j)} + \zeta \Phi \circ \mathbf{t} \quad (12)$$

where ζ denotes the template updating rate. Φ is a binary mask that takes a non-zero value when pixels fall in the rectangle bounding box specified by the j^{th} predicted coordinates $rec(x^{(j)}, y^{(j)}, w^{(j)}, h^{(j)})$. So the appearance template is handled in a cumulative way and only the appearance features correspond to the foreground target are preserved. Algorithm 1 summarizes our CRL tracking method.

4. Experimental Results

In this section, we present experimental results. First, we introduce the experimental setup. Then we provide quantitative, qualitative and attributes comparisons on two public datasets with existing sparse trackers and other state-of-the-art trackers.

4.1 Experimental Setup and Evaluation Metrics

The proposed tracking method is implemented with Matlab, and all experiments are evaluated on a PC with Intel Core i7-8700K CPU @ 3.7GHz, 16GB RAM, and Matlab R2017a.

Parameters: The parameters are empirically determined and fixed for all the test sequences as follows. An image patch is cropped with a margin size of 0.5 times the

target mean size and then resized into 255x255. Then the cropped image patch is fed into the 2-layer AlexNet trained by the CFNet to extract deep features. The η and ρ are set to 10. λ is set to 0.001. The circulant reverse lasso model is optimized for 3 iterations (*iter* in Algorithm 1). The correlation filter template is updated with a learning rate ζ of 0.005.

Trackers: To evaluate our tracking algorithm, we conduct experiments with two popular visual tracking dataset, the OTB 2013 benchmark [21] and the 2015 benchmark [25], against 19 tracking methods, including the 5 sparse trackers reported in [21]: SCM [20], ASLA [10], LSK [18], MTT [12], and L1APG [9], 3 recently proposed sparse trackers: CST [4], RSST [7], ISGL [5], the top 10 tracking methods in [21] besides the sparse trackers: Struck [38], TLD [39], CXT [40], VTD [41], VTS [42], CSK [43], and additional recent trackers for comparison: KCF [44], CFNet-2 [37], SAMF [45], DSST [46], TGPR [47]. Note that CFNet-2 is chosen to serve as a baseline of correlation filter trackers since we reuse the Alexnet 2-layer configuration trained by the CFNet. We only list the numerical results reported in the papers of the 3 recently proposed sparse trackers [4], [5], [7] due to lack of available codes.

Evaluation Metrics: The performance of our CRL tracker is quantitatively evaluated by average precision rates and average success rates under the one pass evaluation (OPE) criterion. The OPE means that trackers are evaluated throughout a test sequence with initialization from the ground truth position in the first frame. The precision plot gives the percentage of frames whose estimated location is within the given threshold Euclidean distance of the ground truth. Meanwhile, the overlap score is calculated by

$S = |b_p \cap b_g| / |b_p \cup b_g|$, where b_p is the predicted bounding box, b_g is the ground truth bounding box, $\cap$ and $\cup$ means the intersection and union of two regions, respectively, and $|\cdot|$ denotes the number of pixels in the region. The success plot gives the percentage of frames where the overlap score is greater than a threshold. The area under curve (AUC) is applied in each success plot to rank trackers. In legends and tables, we report the average distance precision at 20 pixels for each tracker. For clarity, we present the top 10 trackers in each plot.

4.2 Quantitative Results and Qualitative Analysis

1) Overall Performance on OTB 2013 Benchmark

Figure 2 illustrates the comparison of the proposed CRL tracker with the other 19 selected trackers. It is evident that the proposed CRL tracker outperforms existing sparse trackers and other state-of-the-art trackers in terms of success rate and precision. In addition, we list the numerical performance scores for each tracker in Table 1 to

Fig. 2 Overall performance comparison on the OTB 2013 benchmark using precision and success plots of OPE criterion. The legend contains the area-under-the-curve (AUC) score and the average distance precision score at 20 pixels for each tracker. Our tracker (CRL) performs favorably against the state-of-the-art trackers.

Table 1 Tracking performance on the OTB 2013 and OTB 2015 benchmark. The best results are marked in bold-face fonts. The second best results are marked by underlines.

Trackers	OTB 2013		OTB 2015	
	Precision (%)	AUC Score (%)	Precision (%)	AUC Score (%)
Sparse Trackers				
CRL (Ours)	80.0	60.5	<u>78.3</u>	<u>57.6</u>
SCM [20]	64.9	49.9	56.9	44.5
ASLA [10]	53.2	43.4	51.4	40.9
LSK [18]	50.5	39.5	49.5	38.5
MTT [12]	47.5	37.6	44.7	34.6
L1APG [9]	48.5	38.0	46.4	35.7
CST-HOG [4]	77.7	54.8	—	—
RSST-Deep [7]	<u>78.9</u>	<u>59.0</u>	78.9	58.3
ISGL [5]	69.5	50.2	—	—
Other Trackers				
Struck [38]	65.6	47.4	63.8	46.1
TLD [39]	60.8	43.7	59.5	42.6
CXT [40]	57.5	42.6	55.3	41.3
VTD [41]	57.6	41.6	51.1	36.7
VTS [42]	57.5	41.6	50.4	36.2
CSK [43]	54.5	39.8	51.9	38.5
KCF [44]	74.0	51.4	69.6	47.7
CFNet-2 [37]	75.7	57.0	76.0	57.3
SAMF [45]	78.5	57.9	75.1	55.2
DSST [46]	73.7	55.4	69.0	51.8
TGPR [47]	73.5	52.8	65.6	47.3

Fig. 3 Overall performance comparison on the OTB 2015 benchmark using precision and success plots of OPE criterion. The legend contains the area-under-the-curve (AUC) score and the average distance precision score at 20 pixels for each tracker. Our tracker (CRL) performs favorably against the state-of-the-art trackers.

further compare with three recent sparse trackers (CST, RSST, ISGL). Our tracker also outperforms its recently proposed counterparts by 2.3%, 1.1%, 10.5%, and 5.7%, 1.5%, 10.3% in terms of precision and success rate, respectively. The RSST tracker using deep features achieves the closest performance to CRL but runs at 1.8 frames per second, which is much slower than CRL.

Note that the proposed CRL tracking method extracts the same deep features as CFNet-2, and the correlation filter template learning of CRL is basically the same as the process of CFNet-2. The differences are that the current target size is reflected in the template updating process of CRL and CRL utilizes a reverse sparse model to predict the similarity score between each circulant candidate and the template. From the tracking results in Table 1, it shows that solving reverse sparse model is more robust than directly cross correlation that achieves performance gains of 4.3% and 3.5% in terms of precision and success rate, respectively, while both two methods train basically the same correlation filter template.

CST is the first method that introduces the property of circulant matrix into sparse representation based tracking model. However, our CRL method achieves a better tracking performance (+2.3%/+5.7%), and faster tracking speed (16.3 $\times$) by using a reverse sparse representation model and better online template handling.

2) Overall Performance on OTB 2015 Benchmark

To evaluate the proposed CRL method, more experiments on a larger benchmark are conducted. Figure 3 shows the OPE success and precision plots of the top 10 performing tracking methods on the OTB 2015 benchmark. Numerical experimental results on the OTB 2015 benchmark for all the 20 methods are also listed in Table 1.

In general, the proposed CRL method performs well against other state-of-the-art tracking methods. Among the sparse representation based baseline methods evaluated on the OTB 2015 benchmark, the proposed CRL methods outperforms against the SCM, ASLA, LSK, MTT, L1APG by at least 21%/13% in terms of the precision as well as AUC scores. It also shows comparable results (−0.6%/−0.7%) against the RSST method with deep features. Furthermore, the proposed CRL method achieves better or similar results than recent circulant structure based tracking methods

including KCF (+8.7%/+9.9%), CFNet-2 (+2.3%/+0.3%), SAMF (+3.2%/+2.4%), DSST (+9.3%/+5.8%). It is worth noting that the above four methods are chosen to compare with the proposed CRL method since no spatial regularizations have been applied in the correlation filter template learning process. And experimental results demonstrate that the proposed CRL methods is more robust by adopting a reverse sparse decomposition with a correlation filter template.

3) Attribute-based Evaluation

Each sequence in the OTB dataset are tagged with 11 attributes which enables the comparison of trackers under different challenging situations separately. We present the precision and success plots of OPE for 8 attributes in Fig. 4. From the figure plots it shows that the proposed CRL method performs well in handling different challenge scenarios, including illumination variation (IV), out of plane rotation (OPR), occlusion (OCC), deformation (DEF), motion blur (MB), out of view (OV) and background clutter (BC). For the factor of scale variation (SV), CRL is slightly better than CFNet-2 only in precision plots (+1.2%). This may suggest CRL to adopt a more effective scale variation handling strategy in the future. In particular, CRL achieves an improvement of 10.3% and 4.2% than the second best performing tracking method in the DEF scenario. It demonstrates that the proposed CRL method can robustly handle target appearance deformation in the real-world environment.

For a more intuitive understanding of the performance of the proposed CRL method, we plot raw tracking results of the top 10 trackers on various image sequences in Fig. 5. The targets to be tracked in these sequences undergo heavy occlusion, scale variation, illumination variation, appearance deformation and so on. For example, the target person is completely obscured by a pole in several frames in the sequence Jogging. But the proposed CRL method is able to relocate the target after the target becomes visible. In the sequence Human, the proposed CRL method successfully continually tracks the target person even though the appearance feature of the target changes as time goes on.

4) Running Speed Evaluation

The computation cost is of vital importance for a visual tracker due to the real time processing demands of visual tracking tasks. However, despite several previous attempts to speed up sparse decomposition calculations, existing sparse trackers still lack real-time processing speed for promising performance. Since the circulant structure is introduced to transform computation into the frequency domain, the proposed CRL methods achieves a good balance of real time speed and robust tracking performance. To make a comparison with existing sparse trackers, running speed (average frame per second, FPS) for each sparse tracker is listed in Table 2. The speed results for CST-HOG, RSST-Deep, ISGL are taken from corresponding papers. As for SCM, ASLA, MTT, L1APG, we adopt the codes provided from the OTB 2013 benchmark and report average running speed evaluated on the benchmark sequences. The speed of LSK is taken from OTB 2015 [25] because of its

Fig. 4 Attribute based precision plots and success plots on the OTB 2015 benchmark. Experimental results on 8 different attributes are presented.

Fig. 5 Tracking results of the top 10 performing trackers on 10 image sequences: *Woman*, *Sylvester*, *Singler1*, *Shaking*, *Liquor*, *Lemming*, *Jogging*, *Ironman*, *Human9* and *Human6*. Frame Indexes are shown in the top left of each figure in yellow color.

Table 2 Running speed comparison of sparse trackers in average frames per second (FPS)

	CRL	SCM	ASLA	LSK	MTT	LIAPG	CST-HOG	RSST-Deep	ISGL
FPS	35.8	0.19	2.8	5.5	1.1	4.7	2.2	1.8	2.5

Fig. 7 Some failure tracking cases caused by large in-plane rotation angle, fast object motion and low resolution of target image patch.

Fig. 6 The effects of varied λ values.

dependence on an outdated Matlab runtime. It shows in Table 2 that the proposed CRL method can run above 30 fps without code optimization and thus outperforms existing sparse trackers in terms of running speed.

5) Effects of The Parameter

In the proposed reverse sparse representation formulation, the parameter λ controls the regularization of coefficient sparsity. The effects of this parameter can be understood more clearly in the Eq. (7), which is used as a threshold in the coordinate-wise soft thresholding operator. If the value of λ is too large, $\mathbf{b}$ tends to zero, and the sparsity will be over-emphasized. If the value of λ is too small, $\mathbf{b}$ tends to $\mathbf{c}$ and may lead to an unstable solution. We report the tracking accuracies with different λ values in Fig. 6.

6) Limitations and Failure Cases

In the attribute-based evaluation presented in Fig. 4, it shows that the proposed tracking method can effectively deal with many challenging factors. However, our tracker still cannot robustly tracking objects in some challenging scenarios, such as in-plane rotation (IPR), fast motion (FM), and low resolution (LR). Figure 7 illustrates some tracking sequences that the proposed tracker fails to track. It suggests that the proposed method have limited ability to extract rota-

tion invariant information and low resolution information. A more robust feature description and adaptive motion model should be considered in the future.

5. Conclusions

In this paper, we propose a robust sparse tracking method based on circulant reverse lasso model. By integrating the properties of circulant structure, densely sampled target candidates are implicitly generated and embedded into a reverse sparse model to encode a single correlation template. The optimization for reverse sparse model can be efficiently solved in the frequency domain, which enables the proposed tracker to run in real-time. A scale-aware template updating strategy is combined with the correlation filter template learning to take into account both appearance deformations and scale variations. Experimental results compared with state-of-the-art sparse trackers on challenging tracking benchmarks demonstrate the effectiveness and robustness of the proposed algorithm.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (no.61572110) and National Key Research & Development Plan of China (no.2016YFB1001401).

References

- [1] X. Mei and H. Ling, "Robust visual tracking using ℓ_1 minimization," 2009 IEEE 12th International Conference on Computer Vision, pp.1436–1443, IEEE, 2009.
- [2] H. Fan and J. Xiang, "Robust visual tracking with multitask joint dictionary learning," IEEE Trans. Circuits Syst. Video Techn., vol.27, no.5, pp.1018–1030, 2017.
- [3] S. Zhang, H. Yao, X. Sun, and X. Lu, "Sparse coding based visual

- tracking: Review and experimental comparison," *Pattern Recognition*, vol.46, no.7, pp.1772–1788, 2013.
- [4] T. Zhang, A. Bibi, and B. Ghanem, "In defense of sparse tracking: Circulant sparse tracker," *Proc. IEEE conference on computer vision and pattern recognition*, pp.3880–3888, 2016.
 - [5] Y. Zhou, J. Han, X. Yuan, Z. Wei, and R. Hong, "Inverse sparse group lasso model for robust object tracking," *IEEE Trans. Multimedia*, vol.19, no.8, pp.1798–1810, 2017.
 - [6] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Trans. Image Process.*, vol.23, no.5, pp.2356–2368, 2014.
 - [7] T. Zhang, C. Xu, and M.H. Yang, "Robust structural sparse tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.
 - [8] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," *European conference on computer vision*, pp.864–877, Springer, 2012.
 - [9] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1830–1837, IEEE, 2012.
 - [10] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model," *2012 IEEE Conference on Computer vision and pattern recognition (CVPR)*, pp.1822–1829, IEEE, 2012.
 - [11] S. Yi, Z. He, X. You, and Y.-M. Cheung, "Single object tracking via robust combination of particle filter and sparse representation," *Signal Processing*, vol.110, pp.178–187, 2015.
 - [12] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," *2012 IEEE conference on Computer vision and pattern recognition (CVPR)*, pp.2042–2049, IEEE, 2012.
 - [13] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Low-rank sparse learning for robust visual tracking," *European conference on computer vision*, pp.470–484, Springer, 2012.
 - [14] Z. Hong, X. Mei, D. Prokhorov, and D. Tao, "Tracking via robust multi-task multi-view joint sparse representation," *Proc. IEEE international conference on computer vision*, pp.649–656, 2013.
 - [15] T. Zhang, K. Jia, C. Xu, Y. Ma, and N. Ahuja, "Partial occlusion handling for visual tracking via robust part matching," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.1258–1265, 2014.
 - [16] P. Feng, C. Xu, Z. Zhao, F. Liu, C. Yuan, T. Wang, and K. Duan, "Sparse representation combined with context information for visual tracking," *Neurocomputing*, vol.225, pp.92–102, 2017.
 - [17] D. Wang, H. Lu, and M.-H. Yang, "Online object tracking with sparse prototypes," *IEEE Trans. Image Process.*, vol.22, no.1, pp.314–325, 2013.
 - [18] B. Liu, J. Huang, L. Yang, and C. Kulikowsk, "Robust tracking using local sparse appearance model and k-selection," *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1313–1320, IEEE, 2011.
 - [19] B. Zhuang, H. Lu, Z. Xiao, and D. Wang, "Visual tracking via discriminative sparse similarity map," *IEEE Trans. Image Process.*, vol.23, no.4, pp.1872–1881, 2014.
 - [20] W. Zhong, H. Lu, and M.-H. Yang, "Robust object tracking via sparsity-based collaborative model," *2012 IEEE Conference on Computer vision and pattern recognition (CVPR)*, pp.1838–1845, IEEE, 2012.
 - [21] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," *Proc. IEEE conference on computer vision and pattern recognition*, pp.2411–2418, 2013.
 - [22] P. Li, D. Wang, L. Wang, and H. Lu, "Deep visual tracking: Review and experimental comparison," *Pattern Recognition*, vol.76, pp.323–338, 2018.
 - [23] A.W. Smeulders, D.M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, no.1, p.1, 2013.
 - [24] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin Zajc, T. Vojir, G. Häger, A. Lukežič, A. Eldesokey, and G. Fernandez, "The visual object tracking vot2017 challenge results," 2017.
 - [25] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.37, no.9, pp.1834–1848, 2015.
 - [26] C. Sun, D. Wang, and H. Lu, "Occlusion-aware fragment-based tracking with spatial-temporal consistency," *IEEE Trans. Image Process.*, vol.25, no.8, pp.3814–3825, 2016.
 - [27] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient ℓ_1 tracker with occlusion detection," *2011 IEEE conference on Computer vision and pattern recognition (CVPR)*, pp.1257–1264, IEEE, 2011.
 - [28] D. Wang, H. Lu, Z. Xiao, and M.-H. Yang, "Inverse sparse tracker with a locally weighted distance metric," *IEEE Trans. Image Process.*, vol.24, no.9, pp.2646–2657, 2015.
 - [29] M. Danelljan, A. Robinson, F.S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," *European Conference on Computer Vision*, pp.472–488, Springer, 2016.
 - [30] M. Danelljan, G. Bhat, F.S. Khan, and M. Felsberg, "Eco: Efficient convolution operators for tracking," *Proc. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp.21–26, 2017.
 - [31] A. Lukežic, T. Vojir, L.C. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," *CVPR*, p.3, 2017.
 - [32] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," *Proc. IEEE International Conference on Computer Vision*, pp.4310–4318, 2015.
 - [33] C. Sun, D. Wang, H. Lu, and M.H. Yang, "Learning spatial-aware regressions for visual tracking," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp.8962–8970, 2018.
 - [34] H. Fan and J. Xiang, "Robust visual tracking via local-global correlation filter," *AAAI*, pp.4025–4031, 2017.
 - [35] T. Zhang, S. Liu, C. Xu, B. Liu, and M.-H. Yang, "Correlation particle filter for visual tracking," *IEEE Trans. Image Process.*, vol.27, no.6, pp.2676–2687, 2018.
 - [36] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol.3, no.1, pp.1–122, 2011.
 - [37] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P.H.S. Torr, "End-to-end representation learning for correlation filter based tracking," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.5000–5008, IEEE, 2017.
 - [38] S. Hare, A. Saffari, and P.H.S. Torr, "Struck: Structured output tracking with kernels," *2011 International Conference on Computer Vision (ICCV)*, pp.263–270, 2011.
 - [39] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.49–56, IEEE, 2010.
 - [40] T.B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1177–1184, IEEE, 2011.
 - [41] J. Kwon and K.M. Lee, "Visual tracking decomposition," *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1269–1276, IEEE, 2010.
 - [42] J. Kwon and K.M. Lee, "Tracking by sampling trackers," *2011 IEEE International Conference on Computer Vision (ICCV)*, pp.1195–1202, IEEE, 2011.
 - [43] J.F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," *European conference on computer vision*, pp.702–715, Springer, 2012.

- [44] J.F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.37, no.3, pp.583–596, 2015.
- [45] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," *European Conference on Computer Vision*, pp.254–265, 2014.
- [46] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," *British Machine Vision Conference*, Nottingham, Sept. 1-5, 2014.
- [47] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with gaussian processes regression," *European Conference on Computer Vision*, pp.188–203, Springer, 2014.

Chenggang Guo received the M.S. degree in communication and information systems from South China University of Technology in 2015. He is currently a Ph.D. student at the University of Electronic Science and Technology of China. His current research interests include object detection and tracking.

Dongyi Chen received the M.S. and Ph.D. degrees in computer and automation, and electronic information engineering from Chongqing University in 1985 and 1997, respectively. He is a professor at the School of Automation Engineering, University of Electronic Science and Technology of China. His post-doctoral research was at the Department of Electrical and Computer Engineering from University of Toronto during 1997-1999. He was a visiting professor at the School of Computing at the Georgia Institute of Technology during 2002-2005. His research interests include augmented reality, wearable computing, and wireless sensor networks.

Zhiqi Huang received his Ph.D. degree in measuring and testing technologies and instruments in 2009. He is an associate professor at the University of Electronic Science and Technology of China. His current research interests include wearable computing, human-computer interaction, and reliability engineering.