A Bottom-Up and Top-Down Integration Framework for Online Object Tracking

Meihui Li, Lingbing Peng, Tianfu Wu, *Member, IEEE*, Zhenming Peng, *Member, IEEE*

***Abstract*—Robust online object tracking entails integrating short-term memory based trackers and long-term memory based trackers in an elegant framework to handle structural and appearance variations of unknown objects in an online manner. The integration and synergy between short-term and long-term memory based trackers have yet studied well in the literature, especially in pre-training free settings. To address this issue, this paper presents a bottom-up and top-down integration framework. The bottom-up component realizes a data-driven approach for particle generation. It exploits a short-term memory based tracker to generate bounding box proposals in a new frame. In the top-down component, this paper presents a graph regularized sparse coding scheme as the long-term memory based tracker. The over-complete bases for sparse coding are composed of part-based representations learned from earlier tracking results and new observations to form a space with rich temporal context information. A particle graph is computed whose nodes are the bottom-up discriminative particles and edges are formed on-the-fly in terms of appearance and spatial-temporal similarities between particles. The particle graph induces a regularization term in optimizing the sparse coding coefficients for bottom-up particles. In experiments, the proposed method is tested on the widely used OTB-100 benchmark and the VOT2016 benchmark with better performance obtained than baselines including deep learning based trackers. In addition, the outputs from the top-down sparse coding are potentially useful for downstream tasks such as action recognition, multiple-object tracking, and object re-identification.**

***Index Terms*—Online object tracking, Bottom-up and Top-down, Graph regularized sparse coding, Alternating direction method of multipliers.**

I. Introduction

Object tracking is one of the research focuses in the field of computer vision due to its wide application for automated video analysis [57], [34], [1], [3], [51]. Despite many years of research, achieving high accuracy and stable tracking is still challenging in the unconstrained real-world scenarios. Nevertheless, the problem of object tracking is relatively easy for a human, even in an extremely complex environment. The well-known Atkinson-Shiffrin Memory Model (ASMM) [52] has mentioned that the memory storage of the human brain is divided into three stages in information processing: sensory memory, short-term memory, and long-term memory. The sensory memory converts the external stimuli to a biological signal, such as sound and image. Acceptable input information is then sent to the short-term store and is encoded as an instant target model. The long-term memory system focuses on processing long-term temporal changes via recalling historical input information.

By simulating the memory mechanism of the human brain, object tracking can be modeled as a continuous activity and entails the stored memory to estimate the target states in the current frame. Existing tracking algorithms are categorized into two distinct types: short-term memory based trackers [24], [23], [26], [9], [50] and long-term memory based trackers [45], [78], [29], [77], [76]. In general, the short-term memory based trackers are learned by the instant information and are able to produce an immediate response with a reasonable result in the case of smooth motion. However, when the target occlusion or target exit happens, the short-term information is invalid to describe the target status. Take the kernelized correlation filter [26] as an example, which has been successfully applied to realize high-speed tracking recently. In the update stage, the new classifier is learned by the training samples selected in the current frame to meet the circular structure, and the classifier is updated by putting more weights on recent frames. With the effect of previous frames decay exponentially over time, it is easy to cause tracking drift when the target undergoes occlusion in long-term tracking. An example of the tracking result obtained from KCF tracker is shown in Fig. 1 (b) and (c), in which the target is partially occluded by a pillar. In this situation, the response peak is corresponding to the distractors, while the real target position is located at the sidelobes of the response map.

Compared with the short-term memory based trackers, the long-term memory based trackers contain more historical information of the object which is effective to deal with complex target appearance changes. In [54], Ross et al. have presented to learn a low dimensional subspace representation of the target objects with incremental learning, in which the new target observations are updated in the appearance model while preserving most of the previous knowledge. The sparse coding method, which models the data vectors as sparse linear

M. Li and Z. Peng are supported by National Natural Science Foundation of China (61775030, 61571096), Sichuan Science and Technology Program (2019YJ0167) and Open Research Fund of Key Laboratory of Optical Engineering, Chinese Academy of Sciences (2017LBC003). M. Li is also supported in part by the financial support from China Scholarship Council. T. Wu is supported by ARO grant W911NF1810295 and NSF IIS-1909644. This work was partly done when M. Li was a visiting student at NC State University. *(Corresponding author: Zhenming Peng and Tianfu Wu.)*

M. Li, L. Peng and Z. Peng are with School of Information and Communication Engineering at the University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China (e-mail: aimee0208@outlook.com, lbpeng163@163.com, zmpeng@uestc.edu.cn).

M. Li and Z. Peng are with Laboratory of Imaging Detection and Intelligent Perception, University of Electronic Science and Technology of China, Chengdu 610054, China

T. Wu is with Department of Electrical and Computer Engineering and Visual Narrative Initiative at the North Caroline State University, Raleigh, NC 27606, USA (e-mail: tianfu_wu@ncsu.edu).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2020.2978623, IEEE Transactions on Multimedia

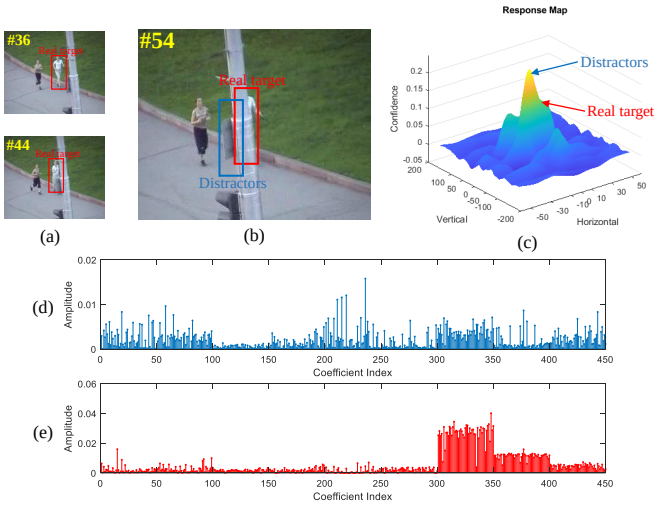

Fig. 1. Approximation of the discriminative particles by the top-down component. (a): The real target areas of sequence Jogging-1. (b): Real target and distractors in the 54^{th} $frame$. (c): Response map obtained from the bottom-up component. (d): Distractor approximated by the top-down component. (e): Real target approximated by the top-down component.

combinations of basis elements, has also been successfully used in long-term tracking. As Mei and Ling [45] suggested, the basis elements are composed of fixed target templates, varied target templates, and negative templates. The earlier tracking results are more accurate, so they should be stored longer and are collected in the fix target templates. The varied target templates are updated frequently with new observations to adapt inevitable appearance changes. Occlusion, noise and other challenging issues are addressed seamlessly through the negative templates. On this basis, the target is presented in the space with rich temporal context information. This method adopts the holistic target as target templates and is liable to cause tracking drift when similar objects or partial occlusion occur. To solve this problem, Jia et al. [29] proposed an adaptive local sparse appearance model, in which both the spatial and partial information of the target is considered via an alignment-pooling method. The sparse representation based trackers have shown a great advantage in dealing with target appearance changes, especially in pre-training free settings. However, the target model construction of the the long-term memory based tracker is time consuming. And it is unrealistic to extract particles with pixel-by-pixel and scale-by-scale sampling, which limits the tracking performance to a certain extent.

Accordingly, it is of great importance to integrate the short-term memory based tracker and long-term memory based tracker to handle object appearance variations efficiently and effectively. In [66], Wang et al. proposed a tracking framework with a re-detection module, which is composed of a short-term memory based tracker and a long-term memory based trackers. It exploits the short-term memory based tracker as the first detector and long-term memory based trackers as a re-detector. The re-detection module is activated if the first detection is failed, which is hard to identify while tracking on-the-fly. On the other hand, the performance of the long-term memory based tracker relies on the accuracy of particle sampling, which is fraught with strong uncertainty.

This paper proposes a bottom-up and top-down integration framework to make the short-term memory based tracker and long-term memory based trackers work cooperatively. The bottom-up component realizes a data-driven approach which guides the gaze to the visual attention areas according to the object feature analysis. We exploit the short-term memory based tracker as the bottom-up component to generate bounding box proposals that are to be carried forward to the top-down component. The areas corresponding to the response peaks and sidelobes are presented as discriminative particles in the new frame, in which both of the real target and other distractors are included, as shown in Fig. 1 (b). The top-down component is driven by the high-level cognitive knowledge and aims to approximate the proposals obtained from the bottom-up component by the long-term memory of the target status. A novel graph regularized sparse coding scheme is presented as the representation model of long-term tracking. We first compute a particle graph whose nodes are the discriminative particles and edges are formed in terms of appearance and spatial-temporal similarities between bottom-up particles. And the constraint mode of the sparse coefficients is induced by the particle graph. Moreover, part-based representations are exploited to model the particles, which aims to deal with target partial variations. The sparse coding results of the distractor and the real target are shown in Fig. 1 (d) and (e) respectively. For the distractor with the maximum response score in the bottom-up component, the energy of the coefficients is scattered in different dictionary entries. By comparison, in the representation of the real target, the non-zero elements of the coefficients are mainly distributed on the corresponding subdictionary.

The contribution of our proposed approach can be summarized in threefold.

1) A novel bottom-up and top-down integration tracking framework is proposed in this paper, which makes the short-term memory based tracker and long-term memory based tracker work cooperatively.

2) In the top-down component, we propose a graph regularized sparse coding scheme to exploit the appearance and spatial-temporal similarities between particles, which is suitable to yield the geometry structure of data with outliers and noises.

3) The proposed tracking approach is tested on the widely used OTB-100 benchmark and VOT2016 benchmark with better performance than the baselines including deep learning based trackers.

The rest of the paper is organized as follows. In Sec.II, we review some works that are closely related to ours. Sec.III talks about the formulation of the integration framework, the detailed description of the long-term memory based tracker with graph regularized sparse coding scheme is presented in Sec.III-B. Sec.IV is the overview and implementation details of the proposed tracking framework. Experimental results are reported and analyzed in Sect.V. The conclusion of the whole paper is summarized in Sect.VI.

II. Related works

In this section, we review two main research streams in visual object tracking which are closely related to our work: correlation tracking and target appearance modeling.

A. Correlation tracking

Correlation tracking methods have gained much attention recently [9], [26], [7]. Like many other discriminative models, in the correlation filter based trackers, a linear or kernelized classifier is trained to find the location with maximum response in a searching window. In 2014, Henriques et al. exploited the special property of circular correlation in the frequency domain to speed up the training process and proposed the KCF tracker [26]. After that, many extensions [15], [38], [40], [48] have been presented to solve the inherent problems of the KCF tracker. To address the problems induced by the periodic assumption, Danelljan et al. introduced a spatial regularization component that penalizes filter coefficients residing outside the target region [15]. In [38], Li et al. proposed a scaled adaptive tracker with multiple features (SAMF) to tackle the problem of fixed template size in the KCF tracker. To deal with tracking situations with partial occlusion, Liu et al. applied the part-based strategy in the correlation filter model and proposed the structured KCF tracker [40]. In [48], a context-aware correlation filter tracking was proposed. In this tracker, several context patches are sampled around the object of interest and are regarded as hard negative samples, aiming to incorporate information about global context to the learned filter. In most traditional correlation filter based trackers, a singled and fixed Gaussian function is utilized as the target response. However, this framework is sensitive to the case where circular shifts do not reliably approximate translations. To solve this problem, Bibi et al. [8] designed an adaptive target response in the context of correlation tracking, in which the best filter and target response are learned and updated jointly.

B. Target appearance modeling

Rich feature types have been utilized in tracking. In the correlation tracking framework, raw pixel feature, Hog feature and color feature are three commonly used hand-crafted features to construct object descriptors. The raw pixel feature is firstly adopted in the DSST tracker [9] and CSK tracker [25]. With the generalizations of linear correlation filters to multiple channels, more modern features are allowed to leverage [26]. In the ACFN tracker [11], a 6-dimensional color feature is built in the RGB and Lab space and is utilized in an attentional mechanism driven correlation filter. In the CSR-DCF tracker [41], standard Hog and colornames features are employed as independent feature channels to obtain the channel-wise correlation outputs. Inspired by the successful Viola-Jones pedestrian detection approach [61], haar features have also been employed in many approaches, such as the Struck tracker [24], MIL tracker [4], OAB tracker [23]. This feature element has almost become a standard for tracking-by-detection approaches. With the development of deep neural networks in many computer vision fields, CNN features have also been applied in the tracking task [2], [55], [22]. In 2015, Ma et al. interpreted hierarchical convolutional features as a non-linear counterpart of an image pyramid representation and exploited these multiple levels of abstraction for visual tracking [43]. In this method, the deep image features are obtained offline from a pre-trained convolutional network. While trackers in this style include the DeepSRDCF tracker [14], FCNT tracker [64], C-COT tracker [16], CNN-SVM tracker [27], etc.

To combine the discriminative power of CNN features with the correlation filter, an online end-to-end convolutional deep learning is proposed by Valmadre et al. [60]. In the same year, Song et al. [58] proposed a convolutional residual learning scheme for tracking (CREST), in which the discriminative correlation filters are reformulated as a one layer convolutional network to generate response maps. In [70], Yun et al. proposed the AdNet tracker. The pre-training of actions is done by utilizing deep reinforcement learning and supervised learning. In the multiple object tracking field, Milan et al. employed the recurrent neural networks to perform prediction, data association and state update [46]. Girdhar et al. employed the Mask R-CNN for pose estimation in work [21]. In 2019, Gao et al. proposed a graph convolutional tracking (GCT) [19]. This tracker jointly incoperates two types of graph convolutional networks into a siamese network for target appearance modeling. However, these trackers need pre-training settings which is not suitable for the real-scenario application. On the other hand, the tracking speed and memory occupation are sacrificed to achieve higher accuracy.

To capture particular visual information in a certain scenario, statistical learning schemes were applied to some tracking approaches [37]. Specifically, the sparse representation, an effective mathematical model using statistical learning techniques, has played an important role in tracking these years. Mei and Ling [45] first formulated tracking as a sparse approximation problem. To find the tracking target in a new frame, each target candidate is sparsely represented in the space spanned by target template and trivial template. Zhang et al. further proposed a multi-task sparse learning method for object tracking, in which particle representations are jointly sparse while respecting the underlying relationship between particles [75], [76]. A multi-view multi-task sparse tracker [28] was proposed by Hong et al. Multiple types of features are employed to introduce a multiple-view representation for robust appearance modeling. In 2019, Li et al. proposed a locally weighted reverse sparse model, in which the spatio-temporal relationships of different templates were considered in the appearance modeling process [36].

III. Integration framework

A. Bottom-up component

Different off-the-shelf short-term memory based trackers can be utilized as the bottom-up component. We revisit the widely used circulant tracker [26] due to its high efficiency in discriminative learning. The kernelized correlation filter ω is trained with a base sample p centered around the target and all its circular shifts $p_{i,j}$ $(i = 1, 2, \dots I, j = 1, 2, \dots J)$

with Gaussian function label $l_{i,j}$. The training problem can be modeled as minimizing the following objective function with respect to ω.

$$\omega = \arg\min_{\omega} \sum_{i,j} \left|F\left(\phi\left(p_{i,j}\right)\right) \cdot F\left(\omega\right) - F\left(l_{i,j}\right)\right|^2 + \lambda \|F\left(\omega\right)\|^2, \tag{1}$$

where ϕ is the kernel function, λ is a regularization parameter and F represents the Fourier transform.

With the base candidate patch q and all its circular shifts $q_{i,j}\,(i = 1, 2, \ldots I, j = 1, 2, \ldots J)$, the regression function of all candidates are calculated with Eq.2.

$$r = F^{-1}\left(F\left(\omega\right) \odot F\left(\left\langle \phi\left(q\right), \phi\left(p\right)\right\rangle\right)\right), \tag{2}$$

where F^{-1} denotes the inverse Fourier transform, $\odot$ represents element-wise product, r is the obtained response score.

We adopt a 3-dimensional scale space correlation filter for joint translation-scale detection [13]. In this method, the translation estimation and scale estimation are calculated simultaneously. The scaled correlation filter $\Omega = \left\{\omega^1, \omega^2, \ldots \omega^S\right\}$ has S scale numbers and is trained on S base samples with different scale sizes. Accordingly, the scaled response map R^s is calculated by Eq. 3.

$$R^s = F^{-1}\left(F\left(\omega^s\right) \odot F\left(\left\langle \phi\left(Q^s\right), \phi\left(P^s\right)\right\rangle\right)\right), \tag{3}$$

where s represents the scale index, P^s and Q^s represent the base training sample and the base candidate sample with scale index s respectively.

B. Top-down component

Different from the bottom-up component which regards tracking as a classification task, the top-down component focuses on constructing the appearance model of the target in a space with long-term temporal context information. Spatial-temporal structure of the target is important for tracking. For more completely perform such high-level information to construct the appearance model of the target, we present a graph regularized sparse coding scheme.

In the general form of sparse representation, candidate image y_i is reconstructed by a linear combination of dictionary atoms and the coefficients are constrained by an l_1 norm. We have,

$$\arg\min_{x_i} \frac{1}{2} \|y_i - Dx_i\|_2^2 + \alpha \|x_i\|_1, \tag{4}$$

in which x_i is the coefficient vector; α is the penalization parameter; D is an artificially defined dictionary, which is composed of fixed target template, earlier tracking results and new observations. The representations of $y_1, \ldots y_N$ are treated independently and equally in the model above.

Given a set of candidate images $Y = \{y_1, \ldots y_N\} \in R^{d \times N}$, where d represents the feature dimension of candidate images and N represents the candidate image number, a graph G is constructed whose nodes are the candidate items, whose edges reflect the similarities of two nodes. If there is a connection between two candidates y_p and y_q in graph G, the dictionary atoms for participating the sparse coding of these two candidates should be the same. This property is regularized by a total variation constraint $\sum_{e=(p,q)\in E} r_{p,q} \left|x_p - x_q\right|$, which yields a sparse solution in coefficient differences between highly inter-correlated inputs. $r_{p,q}$ is the edge weight. We adopt the l_1 norm to restrain the coefficient difference and enforce a row-wise structured sparse solution for similar candidate images, which has been shown more robust to data outliers and noises than the Laplacian based constraint [69].

The second graph regularized term is formulated as the weighted fidelity term $\|(DX - Y)\, W\|_2^2$, which is utilized to exploit the spatial-temporal relationship among successive frames. Regarding of the moving continuity and the small change of object's features in frames, we assume that the candidate which is sampled closer to the tracking center in the last frame and has similar appearance model with the last tracking result is more likely to be a better choice. This assumption is referred to adding the node weight W of graph G to the fidelity term, which allows larger penalty degrees of reconstruction error for the candidates with good spatial-temporal coherent and vice versa.

In view of target partial corruption and partial appearance changes, part-based representations are further introduced in the long-term memory based tracker. Each particle is divided into several local square blocks by a sliding window with a fixed size $\sqrt{d} \times \sqrt{d}$. We denote $y_i^{(k)}$ as the k^{th} part of particle y_i and $t_i^{(k)}$ as the k^{th} part of template t_i. The dictionary D is defined as $D = \left\{T^{(1)}, T^{(2)}, \ldots T^{(K)}\right\}$, where $T^{(k)} = \left\{t_1^{(k)}, t_2^{(k)}, \ldots t_M^{(k)}\right\}$ is the set of the k^{th} local part of the M templates. The graph regularized sparse coding is applied for each particle part individually with the dictionary shared. And the part-based objective function is defined as follows.

$$\begin{aligned} \arg\min_{X^{(k)}} \tfrac{1}{2} \left\|\left(DX^{(k)} - Y^{(k)}\right) W^{(k)}\right\|_2^2 + \alpha \left\|X^{(k)}\right\|_1 \\ + \gamma \sum_{e=(p,q)\in E} r_{p,q}^{(k)} \left|x_p^{(k)} - x_q^{(k)}\right|. \end{aligned} \tag{5}$$

C. Component integration

Generally, the short-term memory based tracker estimates a unique localization of the target on the highest peak of the response map, which is defined as unimodal detection [65]. The unimodal detection is easy to cause tracking drift when occlusion or similar object appears in the tracking process. As shown in Fig. 1, both of the real target and semantic background distractors generate response peaks in the challenging situations with partial occlusion, and the peak values of the distractors may even exceed that of the real target, which results in the decrease of the distinguishability ability of the short-term memory based tracker. We propose to exploit the short-term memory based tracker as the bottom-up component to realize a data-driven approach for particle generation. For a local area with a unique peak, a non-maximum suppression method is applied to find the most discriminative areas and filter out redundant bounding boxes. The final discriminative

particles are selected based on repeated operations of all such local areas.

In the top-down component, a particle graph is constructed to exploit the high-level cognitive knowledge of bottom-up particles. The integration of these two components forms the main contribution of this paper. Given a set of particles generated from the bottom-up component $Y = \{y_1, \dots y_N\} \in R^{d \times N}$, where d represents the particle feature dimension and N represents the particle number, a particle graph G is constructed whose nodes are the particles items, node weights and edge weights are computed by the following rules.

Edge construction rule. To reduce the impact of node number on relationship measurement, an adaptive edge construction rule is presented which contains an upper bound of the edge number and a lower bound of the edge weight. When the node number is small, only the minimal weight bound will be activated to keep connection among very related nodes. When the node number is large, both the number and weight bound will take effect on edge reservation and removal, which can enhance the effect of the most related neighbors. The edge weight between node y_i and node y_j is formulated as the similarity of particle feature and is calculated according to Eq.6.

$$r_{i,j} = \begin{cases} 0, if\ r_{i,j} < th_w\ or\ id\,(r_{i,j}) > th_n \\ e^{-\frac{\|y_i - y_j\|_2^2}{\sigma^2}}, otherwise \end{cases} \tag{6}$$

in which σ is a normalization parameter, th_w and th_n represent the threshold of edge weight and edge number respectively. $id\,(r_{i,j})$ is the index of the edge weight $r_{i,j}$ in a descend order.

Node weight evaluation rule. The heat kernel weighting scheme [5] is adopted to exploit the spatial-temporal information of these particles. The target center position and target feature of the last frame are recorded as p_{t-1} and y_{t-1} respectively. According to the node feature y_i and node position p_i of the current node, the location offset factor w_{i_loc} and feature variation factor of node y_i are defined by Eq.7.

$$\begin{aligned} w_{i\_loc} &= e^{-\frac{\|p_i - p_{t-1}\|_2^2}{\sigma^2}} \\ w_{i\_app} &= e^{-\frac{\|y_i - y_{t-1}\|_2^2}{\sigma^2}} \end{aligned} \tag{7}$$

And the final node weight $w_i^{(k)}$ is the combination of these two factors:

$$w_i^{(k)} = w_{i\_loc}^{(k)} \times w_{i\_app}^{(k)} \tag{8}$$

Consider the problem of mapping the graph G to the sparse representation, a reasonable criterion for representing multiple particles is to minimize the following objective function:

$$\begin{aligned} &\arg\min_X \tfrac{1}{2} \left\| (DX - Y)\, W \right\|_2^2 + \alpha \|X\|_1 \\ &+\gamma \sum_{e=(p,q) \in E} r_{p,q} \left| x_p - x_q \right| \end{aligned} \tag{9}$$

$D \in R^{d \times M}$ is the dictionary where each column represents a template vector. $X \in R^{M \times N}$ is the sparse coefficient matrix. N and M represent the number of candidate images and template images respectively. α and γ control the relative weight of two sparse constraints.

IV. Online object tracking with integration framework

A. Overview of the integration tracking framework

The overview of the proposed tracking framework is illustrated in Fig. 2. When a new frame arrives, a classifier with short-term memory of the tracking object is used to generate bounding box proposals (i.e., discriminative particles) within the searching window. Next, each particle is divided into several local image blocks, which are then represented individually by a stored template set of long-term object appearance. The representation results of all parts are combined together to determine the final tracking result.

Based on the graph regularized sparse coding scheme, each particle is encoded by a sparse coefficient group $X_i = \left\{ x_i^{(1)}, x_i^{(2)}, \dots x_i^{(k)}, \dots x_i^{(K)} \right\}$, in which k represents the k^{th} local part of the represented particle. To exploit both the sparse coefficients energies and probabilities of all part, the likelihood of each particle is calculated by Eq.10.

$$p_i = \sum_k E_r^{(k)} \times E_e^{(k)} \tag{10}$$

where $E_r^{(k)}$ is the coefficient probabilities defined by the residual error. And $E_e^{(k)}$ is the coefficient energy ratio, which is defined in Eq. 11.

B. Dictionary initialization and update

The initial templates are obtained from the rectangle areas sampled around the target in the first frame, which is manually calibrated according to the groundtruth data. In the tracking process, the object appearance will not remain the same all the time, as a result, the representation model with the initial templates could be inaccurate to describe tracking objects after a period of time. Generally, we need to update the template set by later tracking results to capture the appearance variations caused by target pose changes or external interference. For the update strategy, there are two essential factors that need to be confirmed: when to update and which to update. In other words, the first thing is to judge the reliability of the obtained tracking result and further determine whether it should be added to the template set. The second thing is to select which old templates should be replaced in the update process due to the fix template number in a sparse representation.

In our method, we adopt the thought of sample validation method proposed in [67] to evaluate the reliability of the tracking results. In view of the classification task which is based on the sparse representation, a valid test image should have a sparse representation whose nonzero entries concentrate mostly on the corresponding objects. Apply this thought to our update strategy, to represent the k^{th} image block of the tracking result, the nonzero entries should concentrate mostly on the atoms in sub-dictionary $D^{(k)}$. We define a coefficient energy ratio $E_e^{(k)}$ to describe such a characteristic:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2020.2978623, IEEE Transactions on Multimedia

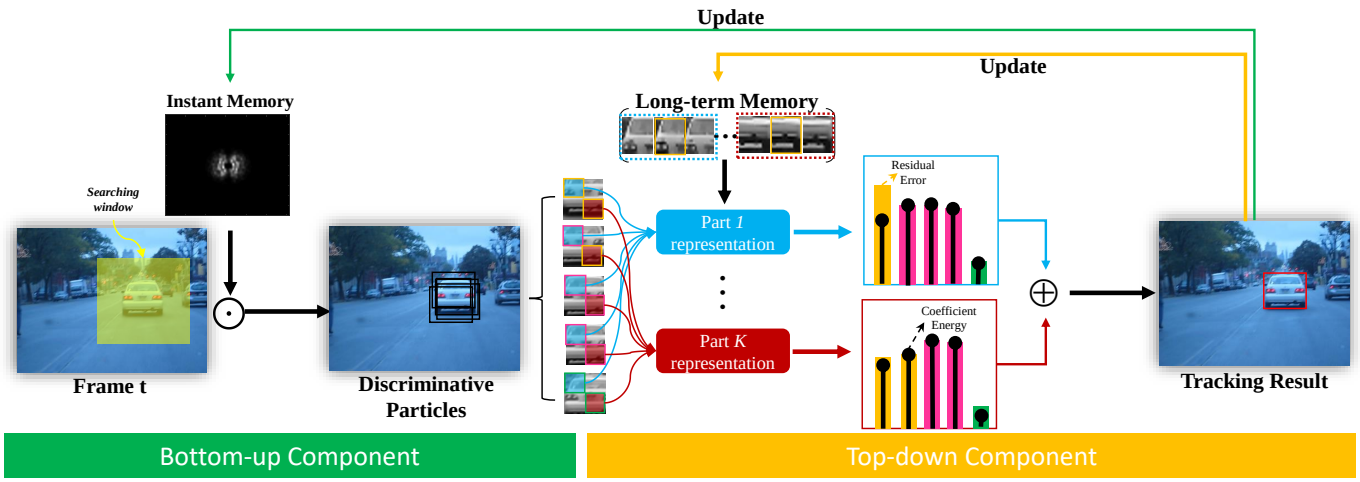

Fig. 2. Overview of the proposed integration tracking framework. Our method contains a bottom-up component, which applies the instant memory (a classifier) to generate bounding box proposals, and a top-down component to model each particle part individually with an over-complete dictionary that contains long-term memory of the tracking objects. The representation model is based on a novel graph regularized sparse coding scheme, in which the underlying relationship of bottom-up particles is utilized as high-level cognitive information in the representation process. And the final tracking result is inferred based on the sparse coding coefficients energies and probabilities of all parts. (See text for details.)

$$E_e^{(k)} = \frac{\|x\left((k-1)\,M+1:kM\right)\|_2}{\|x\|_2} \tag{11}$$

where x is the obtained sparse coefficient of the current tracking result. M is the column number of the local based sub-dictionary. Given a threshold E_{th}, if $E_e^{(k)} > E_{th}$, it indicates the tracking result is reliable and can be used to replace an old template which contributes most to the reconstruction process; if $E_e^{(k)} \leq E_{th}$, it means the tracking result can not be used to update dictionary.

C. Efficient optimization for graph regularized sparse coding

In the graph regularized sparse representation model, we need to solve the objective function shown in Eq.5. First, the differential term in Eq.5 can be rewritten as follows:

$$\sum_{e=(p,q)\in E} r_{p,q}^{(k)} \left| x_p^{(k)} - x_q^{(k)} \right| = \left\| X^{(k)} H^{(k)} \right\|_1 \tag{12}$$

in which $H^{(k)} \in R^{N\times E}$ is the vertex-edge matrix:

$$H_{i,e=(p,q)}^{(k)} = \begin{cases} r_{p,q}^{(k)} & if \quad i=p, \\ -r_{p,q}^{(k)} & if \quad i=q, \\ 0 & otherwise. \end{cases} \tag{13}$$

It can be seen from Eq.12 that H is a sparse matrix with $2\times E$ non-zero elements, E is the total edge number. Thus, we can rewrite the objective function in Eq.9 as:

$$\begin{aligned} &\arg\min_{X^{(k)}} \tfrac{1}{2} \left\| \left(DX^{(k)} - Y^{(k)}\right) W \right\|_2^2 + \alpha \left\| X^{(k)} \right\|_1 \\ &+\gamma \left\| X^{(k)} H^{(k)} \right\|_1, k = 1, 2, \ldots K \end{aligned} \tag{14}$$

Here, we propose to use the ADMM (Alternating Direction Method of Multipliers) to solve the above optimization problem. According to [20], Eq.14 can be rewritten as the equality-constrained problem by introducing an auxiliary variable Z:

$$\begin{aligned} &\min_{X^{(k)}} \tfrac{1}{2} \left\| \left(DX^{(k)} - Y^{(k)}\right) W \right\|_2^2 + \alpha \left\| X^{(k)} \right\|_1 + \gamma \left\| Z^{(k)} \right\|_1 \\ &s.t.\ X^{(k)} H^{(k)} - Z^{(k)} = 0 \end{aligned} \tag{15}$$

This can be solved by minimizing the augmented Lagrangian function $L_\rho\left(X^k, Z^k, \lambda\right)$:

$$\begin{aligned} &L_\rho\left(X^{(k)}, Z^{(k)}, \lambda\right) \\ &= \tfrac{1}{2} \left\| \left(DX^{(k)} - Y^{(k)}\right) W \right\|_2^2 + \alpha \left\| X^{(k)} \right\|_1 + \gamma \left\| Z^{(k)} \right\|_1 \\ &+\lambda^T \left(X^{(k)} H^{(k)} - Z^{(k)}\right) + \tfrac{\rho}{2} \left\| X^{(k)} H^{(k)} - Z^{(k)} \right\|_2^2 \end{aligned} \tag{16}$$

In Eq.16, λ is the dual vector, $\rho > 0$ is the penalty parameter. Due to the separable structure of the objective function L_ρ, we can simplify the problem by minimizing L_ρ with respect to variables $X^{(k)}$, $Z^{(k)}$ and λ separately. The ADMM algorithm alternately solves one variable by keeping others fixed, and iterating. The detailed steps of solving the graph-guided structured sparse representation model by ADMM algorithm is summarized in Algorithm 1.

1) Update step for $X^{(k)}$

The first sub optimization problem is the minimization with respect to variable $X^{(k)}$. We define the objective function in Step 1 as $f\left(X^{(k)}\right)$:

$$\begin{aligned} &f\left(X^{(k)}\right) = \tfrac{1}{2} \left\| \left(DX^{(k)} - Y^{(k)}\right) W \right\|_2^2 \\ &+\alpha \left\| X^{(k)} \right\|_1 + \tfrac{\rho}{2} \left\| X^{(k)} H^{(k)} - \left(Z^{(k)}\right)^t \right\|_2^2 \\ &+\left(\lambda^t\right)^T \left(X^{(k)} H^{(k)} - \left(Z^{(k)}\right)^t\right) \end{aligned} \tag{17}$$

Let $u^t = \lambda^t/\rho$, Eq.17 can be rewritten in an easier way:

$$\begin{aligned} &f\left(X^{(k)}\right) = \tfrac{1}{2} \left\| \left(DX^{(k)} - Y^{(k)}\right) W \right\|_2^2 \\ &+\tfrac{\rho}{2} \left\| X^{(k)} H^{(k)} - \left(Z^{(k)}\right)^t + u^t \right\|_2^2 \\ &+\alpha \left\| X^{(k)} \right\|_1 \end{aligned} \tag{18}$$

Algorithm 1 Algorithm1 ADMM(Alternating Direction Method of Multipliers) Algorithm for GRS model

Input: Dictionary $D \in R^{d\times(K\times M)}$, represented matrix $Y^{(k)} \in R^{d\times N}$, vertex-edge matrix $H^{(k)} \in R^{N\times E}$, model parameters ρ, α, γ.
Initialization: $\left(X^{(k)}\right)^0$, $\left(Z^{(k)}\right)^0$, λ^0.
while not converged **do**
step1: Fix the others and update $X^{(k)}$ by:
$\left(X^{(k)}\right)^{t+1} = \arg\ \min_{X^{(k)}} \frac{1}{2}\left\|\left(DX^{(k)} - Y^{(k)}\right)W\right\|_2^2$
$+\frac{\rho}{2}\left\|X^{(k)}H^{(k)} - \left(Z^{(k)}\right)^t\right\|_2^2 + \alpha\left\|X^{(k)}\right\|_1$
$+(\lambda^t)^T\left(X^{(k)}H^{(k)} - \left(Z^{(k)}\right)^t\right)$
step2: Fix the others and update $Z^{(k)}$ by:
$\left(Z^{(k)}\right)^{t+1} = \arg\ \min_{Z^{(k)}} \frac{\rho}{2}\left\|\left(X^{(k)}\right)^{t+1}H^{(k)} - Z^{(k)}\right\|_2^2$
$+(\lambda^t)^T\left(\left(X^{(k)}\right)^{t+1}H^{(k)} - Z^{(k)}\right) + \gamma\left\|Z^{(k)}\right\|_1$
step3: Update the multipliers λ by:
$\lambda^{t+1} = \lambda^t + \left(\left(X^{(k)}\right)^{t+1}H^{(k)} - \left(Z^{(k)}\right)^{t+1}\right)$
end while
Output: Sparse coefficient matrix $X^{(k)}$.

The first two terms in $f\left(X^{(k)}\right)$ is convex and differentiable. We define a smooth function $h\left(X^{(k)}\right)$ to represent them:

$$\begin{aligned} h\left(X^{(k)}\right) &= \tfrac{1}{2}\left\|\left(DX^{(k)} - Y^{(k)}\right)W\right\|_2^2 \\ &+\tfrac{\rho}{2}\left\|X^{(k)}H^{(k)} - \left(Z^{(k)}\right)^t + u^t\right\|_2^2 \end{aligned} \tag{19}$$

The approximation of $f\left(X^{(k)}\right)$ at point $\left(X^{(k)}\right)^t$ can be calculated as:

$$\begin{aligned} &f\left(X^{(k)}\right) = h\left(\left(X^{(k)}\right)^t\right) + \left\langle X^{(k)} - \left(X^{(k)}\right)^t, \nabla h\left(\left(X^{(k)}\right)^t\right)\right\rangle \\ &+\tfrac{L}{2}\left\|X^{(k)} - \left(X^{(k)}\right)^t\right\|_2^2 + \alpha\left\|X^{(k)}\right\|_1 \\ &= \tfrac{1}{2}\left\|X^{(k)} - \left(X^{(k)}\right)^t + \tfrac{1}{L}\nabla h\left(\left(X^{(k)}\right)^t\right)\right\|_2^2 + \tfrac{\alpha}{L}\left\|X^{(k)}\right\|_1 \\ &= \tfrac{1}{2}\left\|X^{(k)} - \left(\left(X^{(k)}\right)^t - \tfrac{1}{L}\nabla h\left(\left(X^{(k)}\right)^t\right)\right)\right\|_2^2 + \tfrac{\alpha}{L}\left\|X^{(k)}\right\|_1 \end{aligned} \tag{20}$$

where $L > 0$ is the Lipschitz constant. Let $v = \left(X^{(k)}\right)^t - \frac{1}{L}\nabla f\left(\left(X^{(k)}\right)^t\right)$, the objective function $f\left(X^{(k)}\right)$ can be expressed as:

$$f\left(X^{(k)}\right) = \frac{1}{2}\left\|X^{(k)} - v\right\|_2^2 + \frac{\alpha}{L}\left\|X^{(k)}\right\|_1 \tag{21}$$

It can be verified that the approximate solution $\left(X^{(k)}\right)^{t+1}$ is given by:

$$\begin{aligned} \left(X^{(k)}\right)^{t+1} &= \arg\min_{X^{(k)}} f\left(X^{(k)}\right) \\ &= sign\left(v_j\right)\max\left(0, \left|v_j - \tfrac{\alpha}{L}\right|\right) \end{aligned} \tag{22}$$

2) Update step for $Z^{(k)}$
The second sub problem is shown in Step 2. We define:

$$\begin{aligned} g\left(Z^{(k)}\right) &= \tfrac{\rho}{2}\left\|\left(X^{(k)}\right)^{t+1}H^{(k)} - Z^{(k)}\right\|_2^2 \\ &+(\lambda^t)^T\left(\left(X^{(k)}\right)^{t+1}H^{(k)} - Z^{(k)}\right) + \gamma\left\|Z^{(k)}\right\|_1 \end{aligned} \tag{23}$$

The same with the update process in solving $X^{(k)}$, the scaled dual vector $u^t = \lambda^t/\rho$ is added in Eq. (16). Thus,

$$\begin{aligned} &g\left(Z^{(k)}\right) \\ &= \tfrac{1}{2}\left\|\left(\left(X^{(k)}\right)^{t+1}H^{(k)} + u^t\right) - Z^{(k)}\right\|_2^2 + \tfrac{\gamma}{\rho}\left\|Z^{(k)}\right\|_1 \end{aligned} \tag{24}$$

Let $\kappa = \left(X^{(k)}\right)^{t+1}H^{(k)} + u^t$ Therefore, the approximation solution $\left(Z^{(k)}\right)^{t+1}$ is given by:

$$\begin{aligned} \left(Z^{(k)}\right)^{t+1} &= \arg\min_{Z^{(k)}} g\left(Z^{(k)}\right) \\ &= sign\left(\kappa_j\right)\max\left(0, \left|\kappa_j - \tfrac{\gamma}{\rho}\right|\right) \end{aligned} \tag{25}$$

V. Experimental evaluations

A. Experiment settings

In our experiment, the padding size is 3 times of the target size. In the bottom-up component, the regularization parameter and learning rate are set the same with the KCF tracker. Color-name features and HOG features are utilized to generate response maps. We use 17 number of scales with a scale factor of 1.006. For the top-down component,50 positive samples are selected to initialize the dictionary in the first frame. Each candidate image is resized to 32×32 pixels and is divided into 16×16 local image blocks with 8 pixels$'$ patch step size. The penalty factors of the l_1 term and differential sparse term are set to 0.01 and 0.03 respectively. Our experiment is implemented in MATLAB on a laptop with an Intel Core i7-6700HQ 2.60GHz CPU and 16G RAM.

B. Overall performance

1) Object Tracking Benchmark (OTB)

This benchmark contains 100 test sequences. In the tracking process, for each sequence, only the target location of the first frame is manually labeled. Two basic metrics, center error and overlap rate are employed to evaluate the performance of each tracker. Based on these two metrics, the benchmark result is reported as the precision plots and success plots respectively. The precision plots show the position error between the predicting bounding boxes and the ground truth bounding boxes. Its performance score is the distance precision at a threshold of 20 pixels. The success plots take the position and scale variation into account. Its performance score is the area under curve value.

We evaluate the proposed algorithm on OTB-100 with comparisons to 10 state-of-the-art trackers, including 1 correlation filter based trackers: SRDCF [15]; 2 sparse representation based trackers: SCM [78] and MTT [75]; 2 trackers with CNN features: CNN-SVM [27] and CNT [72], 2 tracking-by-detection based trackers: STRUCK [15] and TLD [30] and 3 deep learning based trackers: GCT [19], CREST [58] and AdNet [70]. From the one pass evaluation (OPE) results shown in Fig. 3, we can see that the proposed tracker ranks 1 among all the trackers on the distance precision and ranks 3 on the success rate. Overall, the success plots and precision plots shown in Fig. 3 demonstrate that our tracker performs favorably against other state-of-the-art-trackers.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2020.2978623, IEEE Transactions on Multimedia

We also report the precision and success plots on 8 challenging attributes specified by the benchmark, which are shown in Fig. 4 and Fig. 5. The results indicate that our tracker is effective in handling background clutter and occlusion. It is mainly because the integration of the short-term memory and long-term memory information. The bottom-up component can generate discriminative particles for input to the top-down component. Useless particles are eliminated from the searching area and the remaining particles can have a wide cover range. Then the top-down component studies the appearance and position information of the discriminative particles by using the long-term and part-based dictionary and selects the particle with the maximum likelihood score. When the target is reappear or clear, the long-term memory based tracker can re-detect the target accurately according to the GRS model. In the remaining situations, the performance of the proposed tracker is not as good as the GCT tracker and is similar to other two deep learning based trackers, CREST and AdNet. Meanwhile, our proposed method is significantly superior to trackers without pre-training settings. When the appearance of target changes constantly, the normal hand-crafted features, such as color-name features, HOG features will become unstable to describe the target. This is a common problem for the existing pre-training free trackers. Our tracker reduces this limitation by re-identifying the target using the GRS model, in which many high level information such as context information and spatial-temporal continuity is considered in the target appearance modeling. However, the feature problem will still reduce the efficiency of particle generating and lead to tracking drift. More stable feature representation for short-term memory based tracker will be considered as our future work.

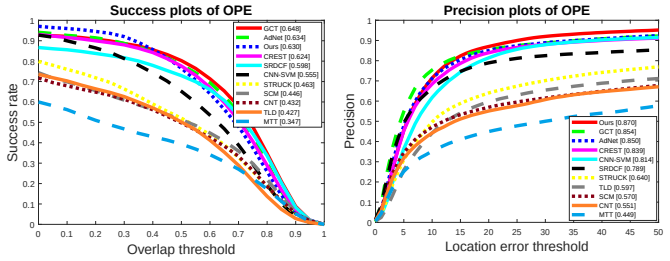

Fig. 3. Plots of one pass evaluation (OPE) on the OTB-100 benchmark. The performance scores in the success plot and precision plot are the area under curve (AUC) value and precision at a threshold of 20 pixels, respectively.

2) The VOT2016 benchmark

The VOT2016 benchmark contains 60 video sequences. Unlike OTB where a tracker is initialized at the beginning of a sequence and left to track until the end, the VOT challenges apply a reset-based methodology, in which trackers are re-initialized five frames after failure. Three measures are identified by Cehovin et al. [10] to analysis tracking behavior in the reset-based experiment: 1) accuracy (A), 2) robustness (R) and 3) expected average overlap (EAO). The accuracy is the average overlap value of the predicted and ground truth bounding boxes during success tracking periods. The robustness measures the number of tracking failures. The third measure, EAO, is a combination of the raw values of per-frame accuracies and failures in a principled manner. The computing details of EAO are stated in [32].

We compare the proposed tracker with 29 related and state-of-the-art tracking methods in the vot2016 challenge. Sixteen trackers are based on the correlation filter framework with hand-crafted features, ACT [18], ART-DSST [31], ColorKCF [62], [26], DFST [53], DSST2014 [13], FCF [31], GCF [33], [12], KCF2014 [26], OEST [44], [68], [71], SAMF2014 [38], sKCF [47], SRDCF [15], SSKCF [35], [26], staple [6], STC [73] and SWCF [26], [31]. Five trackers apply CNN features into correlation filter, CCOT [16], DDC [31], deepMKCF [56], [59], deepSRDCF [14], and RFD-CF2 [43]. The remaining eight trackers are selected from other tracking frameworks, MDNet-N [49] and SiamAN [7] are based on convolutional neural networks architecture, MIL [4] and STRUCK2014 [24] are in the tracking-by-detection framework, DPT [42] and GGTv2 [17], [38] are two part-based trackers, DFT [31] is based on distributed fields, IVT [54] is based on sub-space learning. The MDNet-N tracking method is an extension of MDNet, which is the winner of vot2015. The CCOT method is the winner of vot2016.

The tracking results are summarized by the sequence pooled AR plot and expected average overlap ranks, as shown in Fig. 6. From the AR plot, we can see that in the success tracking period, FCF tracker, SSKCF tracker and our proposed tracker achieve the highest tracking accuracy. Meanwhile, the CCOT tracker, DeepSRDCF and DDC tracker have the least failure times of all these 30 trackers. In the expected average overlap graph, our tracker ranks 6, which indicates that targets usually lost after a relatively long period from the beginning. The raw accuracy and robustness values and ranks of the top 15 trackers (according to EAO) are shown in Tab. I. We also show the tracking performance of our tracker with respect to different visual attributes, which is shown in Tab. II. Our tracker ranks 3 and 11 in the overall accuracy and robustness comparison experiments respectively. Specifically, the proposed tracker performs well in the occlusion situation, which has been demonstrated to be the most difficult attribute in the vot2016 challenge[31].

	EAO	A	A_{rank}	R	R_{rank}
CCOT	0.331	0.541	8	0.788	1
STAPLE	0.295	0.547	4	0.686	7
DDC	0.293	0.542	7	0.708	4
SSKCF	0.277	0.550	2	0.689	6
DeepSRDCF	0.276	0.529	11	0.722	2
OURS	0.265	0.548	3	0.633	11
MDNET	0.257	0.542	6	0.714	3
FCF	0.251	0.556	1	0.633	12
SRDCF	0.247	0.536	9	0.657	8
RFD_CF2	0.241	0.477	15	0.689	5
GGTV2	0.238	0.516	12	0.624	14
DPT	0.236	0.494	14	0.613	15
SIAM_AN	0.235	0.534	10	0.630	13
DeepMKCF	0.232	0.544	5	0.656	9
ColorKCF	0.226	0.504	13	0.642	10

TABLE I

THE TABLE SHOWS EXPECTED AVERAGE OVERLAP (EAO), ACCURACY AND ROBUSTNESS VALUES (A,R) AND RANKS (A_{rank}, R_{rank}) OF THE TOP 15 TRACKERS IN TERMS OF EAO ON THE VOT2016 BENCHMARK.

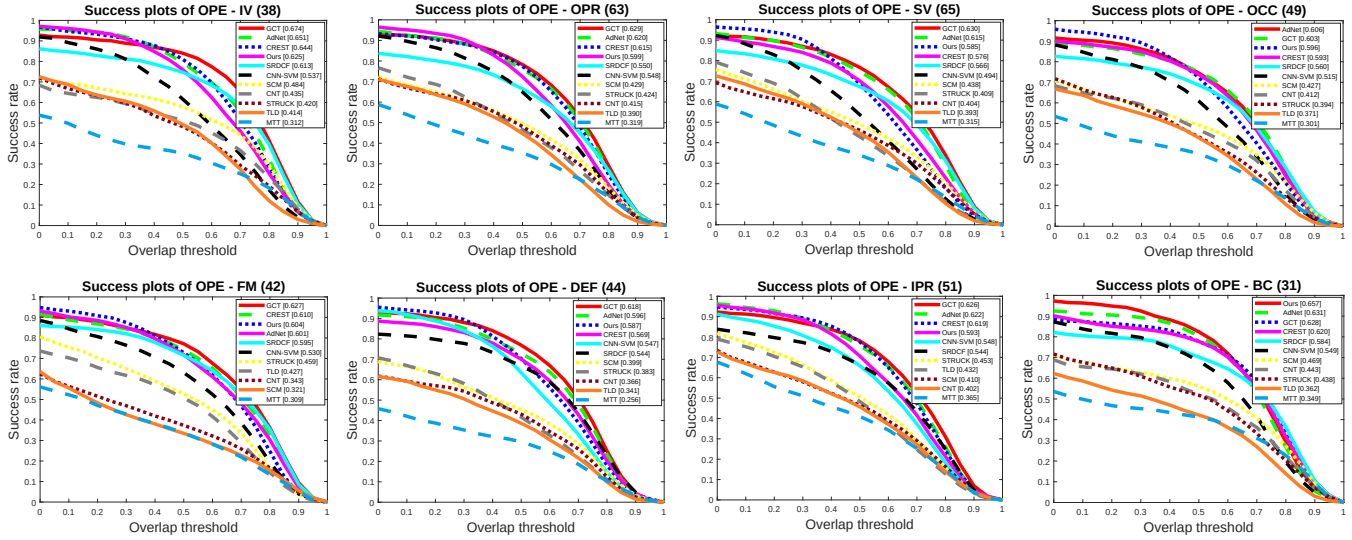

Fig. 4. Success plots of one pass evaluation (OPE) on the OTB-100 benchmark with different challenging attributes, including illumination variation (IV), out-of-plane rotation (OPR), scale variation (SV), occlusion (OCC), fast motion (FM), deformation (DEF), in-plane rotation (IPR) and background clutter (BC).

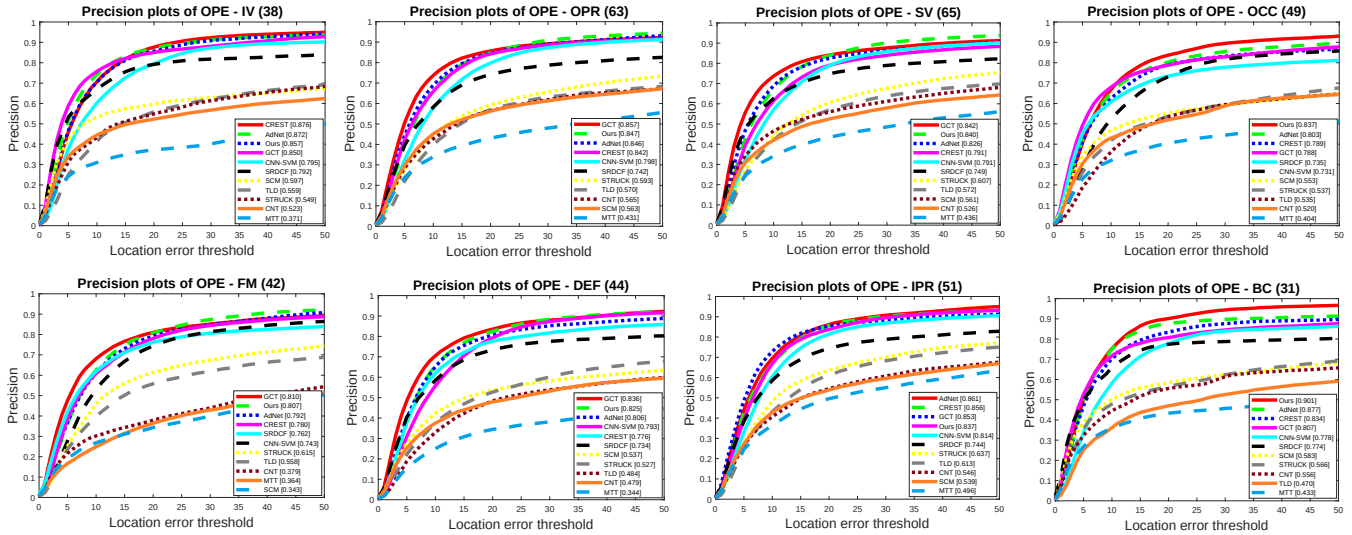

Fig. 5. Precision plots of one pass evaluation (OPE) on the OTB-100 benchmark with different challenging attributes, including illumination variation (IV), out-of-plane rotation (OPR), scale variation (SV), occlusion (OCC), fast motion (FM), deformation (DEF), in-plane rotation (IPR) and background clutter (BC).

$tags$	all	$cam.$	$ill.$	$mot.$	$occ.$	$scal.$	$emp.$
A	0.548	0.530	0.575	0.551	0.517	0.541	0.508
R	0.633	0.639	0.798	0.448	0.443	0.397	0.524
A_{rank}	3	12	11	3	1	5	6
R_{rank}	11	11	11	13	6	15	4

TABLE II

TRACKING PERFORMANCE OF OUR TRACKER ON VOT2016 WITH RESPECT TO THE FOLLOWING VISUAL ATTRIBUTES: ALL SCENES (all), CAMERA MOTION ($cam.$), ILLUMINATION CHANGE ($ill.$), MOTION CHANGE ($mot.$), OCCLUSION ($occ.$), SCALE CHANGE ($scal.$) AND NO ASSIGNED ($emp.$). ACCURACY AND ROBUSTNESS SCORES AND RANKS ARE ILLUSTRATED IN THE TABLE.

C. Ablation Study

In this sub-section, we present several expanded experiments to have a deeper understanding of our proposed tracker. The first group of experiment is shown in Fig. 7 and Fig. 8. Apart from the proposed integration framework 'Integration (OURS)', three contrast algorithms are created, including 'Bottom-up Component (SKF)', 'Top-down Component (GRS+PF)' and 'Integration (No NMS)'. In the 'Bottom-up Component (SKF)', the scale adaptive kernelized correlation filter based tracker (SKF) with HOG features is employed as the bottom-up component and no top-down component is contained in the tracking framework. In other words, only the bottom-up component is utilized to generate response scores in the searching area. Since only the short-term memory is utilized, this tracker is prone to drift when the target undergoes severe deformation or occlusion. Examples have been shown in Fig. 8, the blue bounding boxes represent the tracking result of the 'Bottom-up Component (SKF)' and drift away from the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2020.2978623, IEEE Transactions on Multimedia

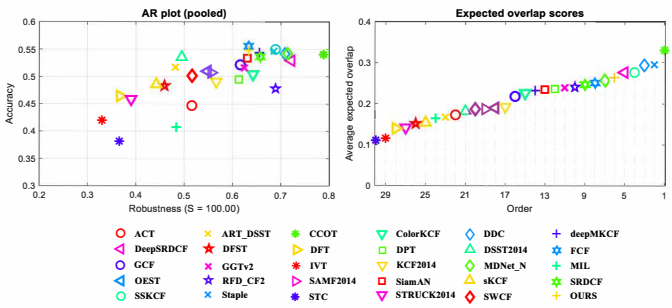

Fig. 6. Pooled AR plot and expected average overlap ranks on VOT2016. The sensitivity parameter for calculating robustness is defined as 100.

target in the sequence Girl and Bolt when occlusion happens. In the 'Top-down Component (GRS+PF)', the proposed graph regularized sparse coding model is employed as the top-down component. Compared with the proposed integration framework in which candidates are generated from the bottom-up component, candidates in this tracker are randomly selected around the center location of the target in the last frame without any prior information. As shown in the sequence Girl in Fig. 8, the pink bounding boxes drift away from the target in frame 441 due to the inappropriate size of the sampled candidates. The number of candidate sampling is 300 in this experiment. By comparison, the generated discriminative particles by bottom-up component is usually less than 50 in the proposed method, which demonstrate the effectiveness of the bottom-up component in the integration framework. The 'Integration (No NMS)' employed the integration framework, however, in this tracker, only the first 30 areas with larger response scores are selected as the discriminative particles. By contrast, in the proposed tracker, a larger range of candidate areas are provided with the use of non-maximum suppression. The comparison results shown in Fig. 7 and Fig. 8 demonstrate the superiority of NMS in discriminative particles' selection.

To validate the effectiveness of the graph regularized sparse (GRS) scheme, which is the representation model of our proposed long-term memory based tracker, we apply it in the particle filter framework and name it as GRS+PF. In this tracker, 300 particles are sampled randomly, other parameters of GRS model are set to the same with the proposed tracker. We compare the GRS+PF tracker with other 7 state-of-the-art sparse trackers: adaptive structural local sparse tracker (ASLA) [29], L1 tracker (L1APG) [45], low rank sparse tracker (LRST) [74], multi-task sparse learning based tracker (MTT) [75], sparse colloborative model based tracker (SCM) [78], local sparse tracker with K-selection (LSK) [39] and sparse prototypes based trackers (SPT) [63]. From Fig. 9, we can see that the GRS+PF tracker outperforms other sparse representation based trackers, which indicates the superiority of the proposed GRS model.

D. Qualitative comparison

The qualitative results shown in Fig. 10 intuitively show the tracking performance of different algorithms over different challenging attributes.

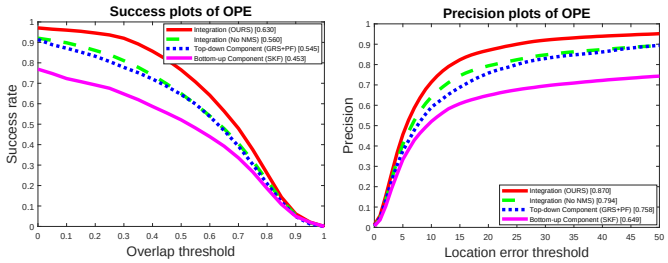

Fig. 7. Self-comparison results of experiments on OTB-100. The scores in the success plots indicate the area-under-curve (AUC) values. The scores in the precision plots indicate the distance precision at a threshold of 20 pixels.

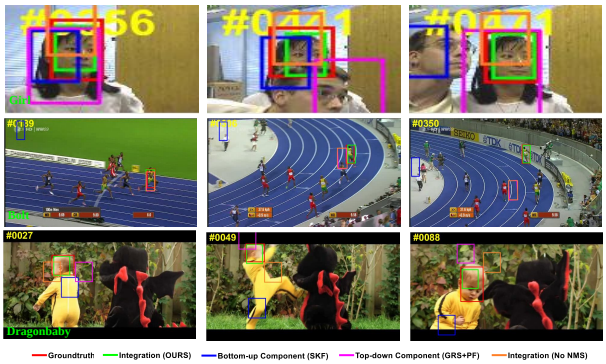

Fig. 8. Self-comparison results of experiments on representative sequences in OTB-100, including Girl, Bolt and Dragonbaby from top to down. The red and green bounding boxes are the groundtruth data and the proposed integration tracking framework, respectively. Other bounding boxes represent tracking results of the self-comparison experiments.

*1) **I**llumination **V**ariation and **OCC**lusion:* Fig. 10(a) shows the tracking results of 9 trackers when the targets are under occlusion and illumination variation. In Bird2 and Kitesurf, the targets, a bird and a human head are intermittently obscured by other objects. Many trackers (AdNet, GCT, SRDCF, STRUCK, MTT, CNN-SVM, TLD and CNT) drift away from these two targets at the 49^{th}, 79^{th} frame of Bird2 and the 55^{th}, 84^{th} frame of Kitesurf respectively. In Human4, Lemming and Basketball, when the targets are severely occluded (frame 354, frame 361 and frame 97 respectively), only the GCT, CNN-SVM, SRDCF, and our proposed methods locate the target successfully. When the target in Human4 is occluded once again at frame 631, both GCT and CNN-SVM

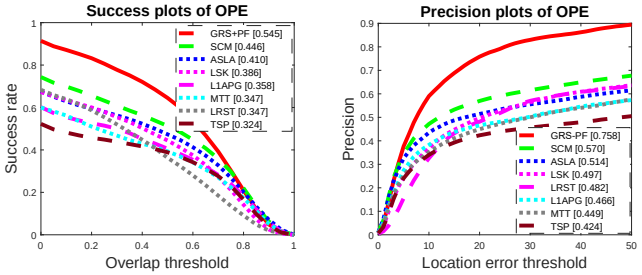

Fig. 9. Verification experiment on OTB-100 for the GRS model. The success plots and precision plots of the GRS+PF tracker and other 7 sparse representation based trackers are shown. The performance score for each tracker is illustrated in the legend.

trackers fail down. Illumination change and occlusion occur simultaneously in Soccer, Tiger1, and Tiger2 at frame 106, 87 and 236 respectively. Only the proposed method successfully tracks the object in all these three situations. The experiment results shown in Fig. 10(a) demonstrate that our tracker can effectively handle complex scenes of illumination variation and occlusion, especially when two challenging attributes occur at the same time.

*2) **S**cale **V**ariation and **DEF**ormation:* We show the tracking results of 8 sequences with deformation and scale variation in Fig. 10(b). In Couple, Human6, Human9, and Skating2-1, the deformation of targets is caused by pedestrian movement. We can see from the tracking results that only the proposed method and SRDCF tracker locate the target accurately and steadily. In Skiing, the target is deformed and the target size is changed during the air leap. Only the proposed method and four convolution based trackers: AdNet tracker, GCT tracker, CREST tracker and CNN-SVM tracker locate the target successfully. In Dog, Dudek and Panda, size variation occurs to the target gradually. We add a scale pyramid to the searching window, so the proposed method performs better than KCF in this situation. In addition, GCT, AdNet and Crest trackers also achieve good performance.

*3) Motion **B**lur, **F**ast **M**otion and **B**ackground **C**lutters:* Representative sequences with motion blur, fast motion and background clutters are shown in Fig. 10(c). In Blurcar1, Blurcar2, Blurcar3, and Blurcar4, the camera or target has an abrupt motion and results in the blurred appearance of cars. TLD, MTT and CNT trackers fail to locate the targets when motion blur occurs in frame 943, 534, 359, 393 respectively. In Biker, the target moves in a very high speed in frame 77, only the proposed method tracks the target successfully. In Board and Clifbar, in addition to motion blur, complex backgrounds also interfere with target tracking. In frame 291 of Clifbar, the GCT, MTT, TLD and CNT methods misjudge the hand as the target. Meanwhile, the CREST, SCM, CNN-SVM and STRUCK trackers are influenced by the clutter background. Only the proposed method and the SRDCF tracker find the real target in examples of both Board and Clifbar. The target jumps up and down in Jumping, causing a blurred appearance of the target. The AdNet, CNT, TLD, SCM and MTT trackers fail down in this sequence.

*4) **I**n-**P**lane **R**otation and **O**ut-of-**P**lane **R**otation:* We summarize two rotation types of moving targets in Fig. 10(d). Examples show that the rotation of human head is usually involved in this situation. In David, Toy, Trellis, and Football1, slight turn happens to the targets in frame 459, 243, 547, and 65. Six trackers, CNN-SVM, SCM, TLD, MTT, CNT, and STRUCK lost the targets when rotations occur. In Freeman3, Freeman4, Shaking, and Surfer, the targets have a larger rotation. Other trackers, such as SRDCF, CNN-SVM, and STRUCK fail down in at least one video. Only the proposed method and AdNet tracker perform steadily in all these rotation situations.

E. Speed Analysis

Table III reports the tracking speed of 6 trackers. In the speed analysis of VOT2016 benchmark, we compare our tracker with other 4 trackers with better or equivalent AR performance. The Staple tracker runs 80 fps, which is fast than others. CCOT and DDC method use CNN features to improve the tracking accuracy at the cost of running speed. They run 0.55 and 0.16 fps respectively. Our tracker and SRDCF run 3.10 and 7.3 fps respectively. In OTB-100, our tracker runs 3.20 frames per second (fps). Among all the methods in OTB-100 experiment, our tracker performs best, SRDCF and SCM have the second best performance in the correlation tracking and sparse representation based tracking respectively. SRDCF runs 5 fps and SCM runs 0.36 fps, the speed of our tracker is in between. It is true that the top-down component of our tracker is time consuming. The complexity of GRS model is $O\left((K \times M)^2 N/\varepsilon\right)$, in which $K \times M$ and N are the number of dictionary atoms and discriminative particles respectively, ε is the convergence accuracy.

		CCOT	Staple	DDC	SRDCF	SCM	OURS
fps	OTB-100	-	-	-	5	0.36	3.20
	VOT2016	0.55	80	0.16	7.3	-	3.10

TABLE III
SPEED REPORTS OF TRACKERS. THE RESULT IS MEASURED BY THE AVERAGE FRAME PER SECOND (FPS). SRDCF AND SCM ARE CONTRAST METHODS IN THE OTB-100 EXPERIMENT. CCOT, STAPLE, DDC AND SRDCF ARE CONTRAST METHODS IN VOT2016 BENCHMARK.

F. Discussion

Apart from the good performance in online object tracking, the outputs of the top-down component of our tracker are potential useful for other downstream tasks. Two video sequences are shown in Fig. 11, each of which illustrates four groups of dictionary entries of the long-term memory based tracker corresponding to different tracking periods. In the first video, the girl in white T-shirt is marked as the target. For the target area with blue mask, different action poses of the leg are recorded in the dictionary. And the appearance of dictionary entries is quite different in the four tracking periods. By comparison, for the target area with red mask, we can see that the appearance of this part remains unchanged in the moving process. In the second video, the target areas with blue and red masks remain unchanged in the first three periods. In the last period, considerable changes have taken place in the appearance of dictionary entries. For the target part with blue mask, the shape of the target arm has a change compared with the previous entries. For the target part with red mask, the recording of the dictionary entries changes from the chest to back. Consequently, the dictionary stores a long-term memory of the tracking objects in the tracking process, which is crucial for action recognition and analysis.

VI. CONCLUSION

In this paper, we have presented a novel integration tracking framework to make the short-term memory based tracker and long-term memory based tracker work cooperatively. Different from the unique detection of other short-term memory based trackers, in our method, the short-term memory based tracker is exploited to generate multiple discriminative particles with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2020.2978623, IEEE Transactions on Multimedia

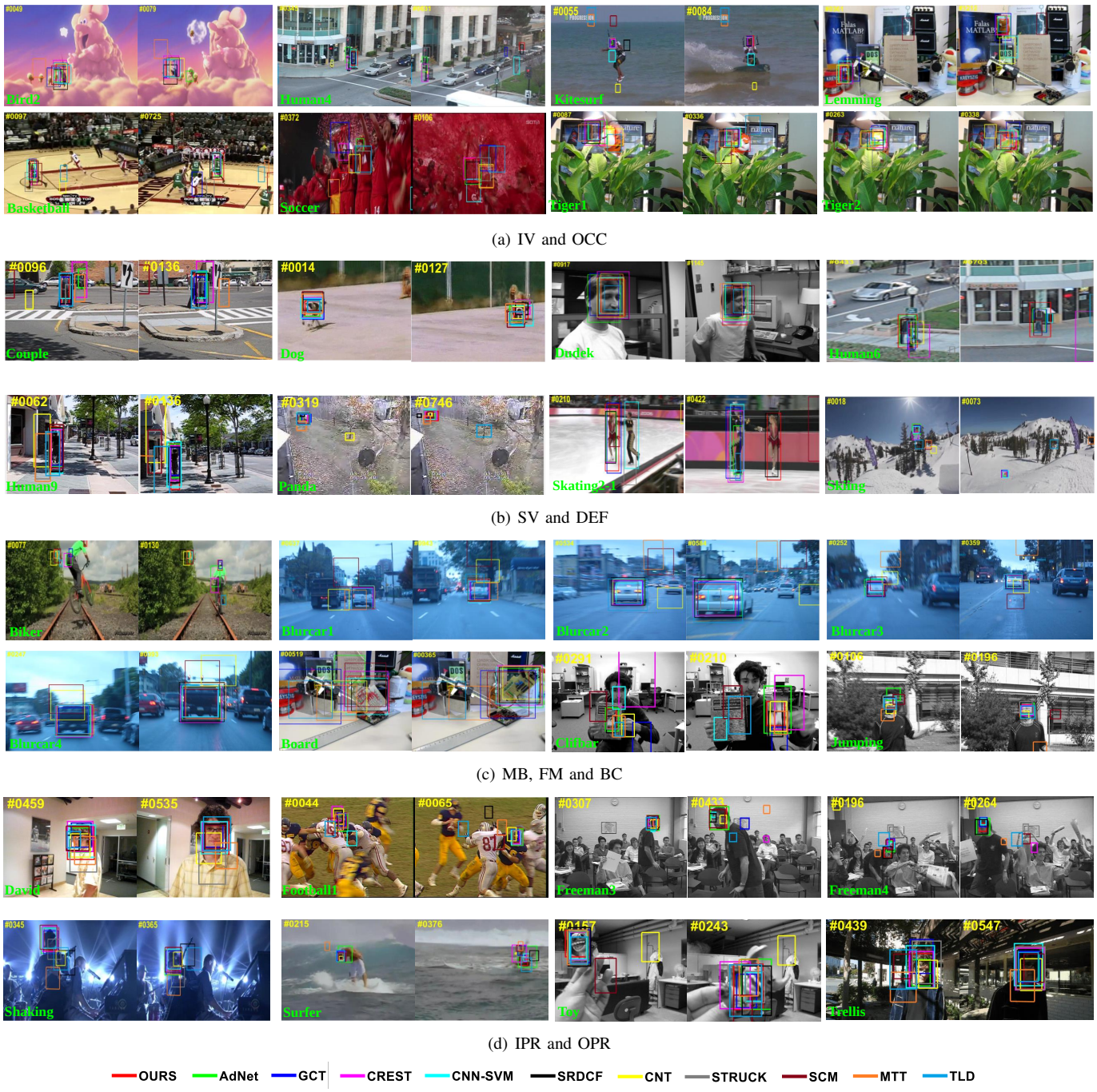

Fig. 10. Qualitative tracking results on OTB-100 of different trackers labeled with different colors over 9 challenging attributes.

the guidance of response map, which avoids the interference of other distractors to a certain extent. A graph regularized sparse coding scheme is proposed as the long-term tracking algorithm, which has shown great superiority in the sparse representation model. The graph regularization enhances the effect of the most informative neighbors among the pair-wise relationship, on the other hand, ensures the spatial-temporal coherent among patches in successive frames. The outputs of the short-term memory based tracker are input to the long-term memory based tracker and are regarded as the graph nodes of the sparse representation model. The overall performance of the proposed tracker is evaluated on OTB100 and VOT2016 benchmarks, which performs favorably against state-of-the-art trackers, especially in the occlusion situation. In addition, the outputs from the top-down sparse coding have shown to be potentially useful for downstream tasks, such as action recognition, multiple-object tracking and object re-identification.

REFERENCES

[1] M. V. Afonso, J. C. Nascimento, and J. S. Marques. Automatic estimation of multiple motion fields from video sequences using a region

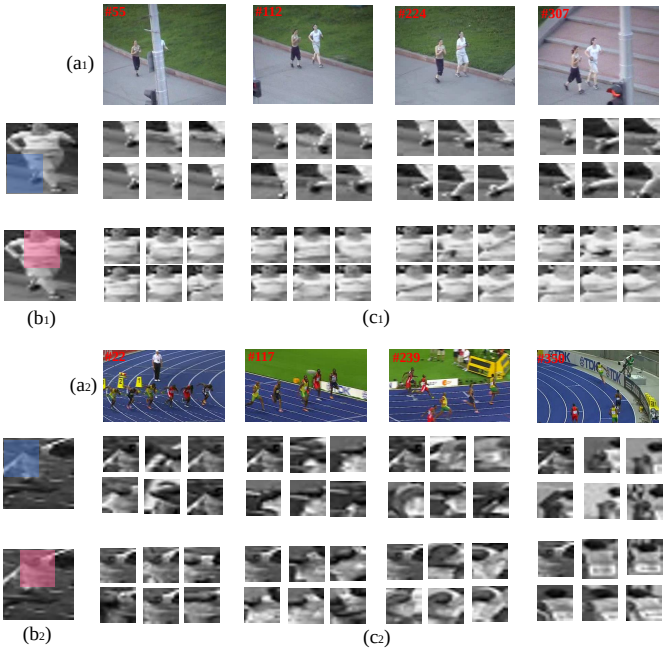

Fig. 11. Illustration of dictionary entries of the long-term memory based tracker. Two sequence videos (Jogging-1 and Bolt OTB-100) are shown as valid test examples. (a): Sequence frames of different tracking period. (b): The initial target to be tracked, the blue and red masks illustrate two selected local parts to be represented. (c): The dictionary entries for target representation.

matching based approach. *IEEE Transactions on Multimedia*, 16(1):1–14, 2014.

[2] P. Agrawal, R. Girshick, and J. Malik. Analyzing the performance of multilayer neural networks for object recognition. In *European conference on computer vision*, pages 329–344. Springer, 2014.

[3] D. Androutsos and R. Phan. Robust semi-automatic depth map generation in unconstrained images and video sequences for 2d to stereoscopic 3d conversion. *IEEE Transactions on Multimedia*, 16(1):122–136, 2013.

[4] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1619–1632, 2011.

[5] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in neural information processing systems*, pages 585–591, 2002.

[6] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr. Staple: Complementary learners for real-time tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1401–1409, 2016.

[7] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016.

[8] A. Bibi, M. Mueller, and B. Ghanem. Target response adaptation for correlation filter tracking. In *European conference on computer vision*, pages 419–433. Springer, 2016.

[9] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2544–2550. IEEE, 2010.

[10] L. Čehovin, A. Leonardis, and M. Kristan. Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing*, 25(3):1261–1274, 2016.

[11] J. Choi, H. J. Chang, S. Yun, T. Fischer, Y. Demiris, J. Y. Choi, et al. Attentional correlation filter network for adaptive visual tracking. In *CVPR*, volume 2, page 7, 2017.

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893. IEEE Computer Society, 2005.

[13] M. Danelljan, G. Häger, F. Khan, and M. Felsberg. Accurate scale estimation for robust visual tracking. In *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[14] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Convolutional features for correlation filter based visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 58–66, 2015.

[15] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4310–4318, 2015.

[16] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *European Conference on Computer Vision*, pages 472–488. Springer, 2016.

[17] D. Du, H. Qi, L. Wen, Q. Tian, Q. Huang, and S. Lyu. Geometric hypergraph learning for visual tracking. *IEEE transactions on cybernetics*, 47(12):4182–4195, 2017.

[18] M. Felsberg. Enhanced distribution field tracking using channel representations. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 121–128, 2013.

[19] J. Gao, T. Zhang, and C. Xu. Graph convolutional tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4649–4659, June 2019.

[20] E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson. Optimal parameter selection for the alternating direction method of multipliers (admm): quadratic problems. *IEEE Transactions on Automatic Control*, 60(3):644–658, 2014.

[21] R. Girdhar, G. Gkioxari, L. Torresani, M. Paluri, and D. Tran. Detect-and-track: Efficient pose estimation in videos. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 350–359, June 2018.

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[23] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via online boosting. In *British Machine Vision Conference*, volume 1, page 6, 2006.

[24] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr. Struck: Structured output tracking with kernels. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2096–2109, 2016.

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

[26] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596, 2015.

[27] S. Hong, T. You, S. Kwak, and B. Han. Online tracking by learning discriminative saliency map with convolutional neural network. In *International Conference on Machine Learning*, pages 597–606, 2015.

[28] Z. Hong, X. Mei, D. Prokhorov, and D. Tao. Tracking via robust multi-task multi-view joint sparse representation. In *Proceedings of the IEEE international conference on computer vision*, pages 649–656, 2013.

[29] X. Jia, H. Lu, and M.-H. Yang. Visual tracking via adaptive structural local sparse appearance model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1822–1829. IEEE, 2012.

[30] Z. Kalal, K. Mikolajczyk, J. Matas, et al. Tracking-learning-detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1409, 2012.

[31] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, and L. Cehovin. The visual object tracking vot2016 challenge results. In *European conference on computer vision*, pages 777–823, 2016.

[32] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, and R. Pflugfelder. The visual object tracking vot2015 challenge results. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 1–23, 2015.

[33] M. Kristan, J. Perš, V. Sulič, and S. Kovačič. A graphical model for rapid obstacle image-map estimation from unmanned surface vehicles. In *Asian Conference on Computer Vision*, pages 391–406. Springer, 2014.

[34] K. Kurzhals, M. Hlawatsch, C. Seeger, and D. Weiskopf. Visual analytics for mobile eye tracking. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):301–310, 2017.

[35] J.-Y. Lee and W. Yu. Visual tracking by partition-based histogram backprojection and maximum support criteria. In *IEEE International Conference on Robotics and Biomimetics*, pages 2860–2865. IEEE, 2011.

[36] M. Li, Z. Peng, Y. Chen, X. Wang, L. Peng, Z. Wang, G. Yuan, and Y. He. A novel reverse sparse model utilizing the spatio-temporal relationship of target templates for object tracking. *Neurocomputing*, 323:319–334, 2019.

[37] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel. A survey of appearance models in visual object tracking. *ACM transactions on Intelligent Systems and Technology (TIST)*, 4(4):58, 2013.

[38] Y. Li and J. Zhu. A scale adaptive kernel correlation filter tracker with feature integration. In *European conference on computer vision*, pages 254–265. Springer, 2014.

[39] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust tracking using local sparse appearance model and k-selection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1313–1320. IEEE, 2011.

[40] S. Liu, T. Zhang, X. Cao, and C. Xu. Structural correlation filter for robust visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4312–4320, 2016.

[41] A. Lukezic, T. Vojir, L. C. Zajc, J. Matas, and M. Kristan. Discriminative correlation filter with channel and spatial reliability. In *CVPR*, volume 6, page 8, 2017.

[42] A. Lukežič, L. Č. Zajc, and M. Kristan. Deformable parts correlation filters for robust visual tracking. *IEEE transactions on cybernetics*, 48(6):1849–1861, 2018.

[43] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang. Hierarchical convolutional features for visual tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 3074–3082, 2015.

[44] C. Ma, X. Yang, C. Zhang, and M.-H. Yang. Long-term correlation tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5388–5396, 2015.

[45] X. Mei and H. Ling. Robust visual tracking and vehicle classification via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 33(11):2259–2272, 2011.

[46] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler. Online multi-target tracking using recurrent neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*, pages 4225–4232, 2017.

[47] A. S. Montero, J. Lang, and R. Laganiere. Scalable kernel correlation filter with sparse feature integration. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 587–594. IEEE, 2015.

[48] M. Mueller, N. Smith, and B. Ghanem. Context-aware correlation filter tracking. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, page 6, 2017.

[49] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4293–4302, 2016.

[50] Z. Peng, Q. Zhang, and A. Guan. Extended target tracking using projection curves and matching pel count. *Optical Engineering*, 46(6):066401, 2007.

[51] Z. Qi, Y. Zhou, S. Shuang, G. Liang, and H. Ni. Heart rate extraction based on near-infrared camera: Towards driver state monitoring. *IEEE Access*, PP(99):1–1, 2018.

[52] J. G. Raaijmakers and R. M. Shiffrin. Models of memory. *Stevens' handbook of experimental psychology: Memory and cognitive processes*, 2, 2002.

[53] G. Roffo and S. Melzi. Online feature selection for visual tracking. 2016.

[54] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental learning for robust visual tracking. *International journal of computer vision*, 77(1-3):125–141, 2008.

[55] M. Schwarz, H. Schulz, and S. Behnke. Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1329–1335. IEEE, 2015.

[56] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[57] K. Somandepalli, N. Kumar, T. Guha, and S. S. Narayanan. Unsupervised discovery of character dictionaries in animation movies. *IEEE Transactions on Multimedia*, 20(3):539–551, 2018.

[58] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. Lau, and M.-H. Yang. Crest: Convolutional residual learning for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2555–2564, 2017.

[59] M. Tang and J. Feng. Multi-kernel correlation filter for visual tracking. In *Proceedings of the IEEE international conference on computer vision*, pages 3038–3046, 2015.

[60] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr. End-to-end representation learning for correlation filter based tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2805–2813, 2017.

[61] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2):153–161, 2005.

[62] T. Vojir, J. Noskova, and J. Matas. Robust scale-adaptive mean-shift for tracking. *Pattern Recognition Letters*, 49:250–258, 2014.

[63] D. Wang, H. Lu, and M.-H. Yang. Online object tracking with sparse prototypes. *IEEE transactions on image processing*, 22(1):314–325, 2013.

[64] L. Wang, W. Ouyang, X. Wang, and H. Lu. Visual tracking with fully convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 3119–3127, 2015.

[65] M. Wang, Y. Liu, and Z. Huang. Large margin object tracking with circulant feature maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4021–4029, 2017.

[66] N. Wang, W. Zhou, and H. Li. Reliable re-detection for long-term tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[67] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–227, 2009.

[68] H. Wu, A. C. Sankaranarayanan, and R. Chellappa. Online empirical evaluation of tracking algorithms. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1443–1458, 2010.

[69] X. Yang, T. Zhang, C. Xu, and M. Xu. Graph-guided fusion penalty based sparse coding for image classification. In *Pacific-Rim Conference on Multimedia*, pages 475–484. Springer, 2013.

[70] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Young Choi. Action-decision networks for visual tracking with deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2711–2720, 2017.

[71] J. Zhang, S. Ma, and S. Sclaroff. Meem: robust tracking via multiple experts using entropy minimization. In *European conference on computer vision*, pages 188–203. Springer, 2014.

[72] K. Zhang, Q. Liu, Y. Wu, and M.-H. Yang. Robust visual tracking via convolutional networks without training. *IEEE Transactions on Image Processing*, 25(4):1779–1792, 2016.

[73] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang. Fast visual tracking via dense spatio-temporal context learning. In *European conference on computer vision*, pages 127–141. Springer, 2014.

[74] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Low-rank sparse learning for robust visual tracking. In *European conference on computer vision*, pages 470–484. Springer, 2012.

[75] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. In *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on*, pages 2042–2049. IEEE, 2012.

[76] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via structured multi-task sparse learning. *International journal of computer vision*, 101(2):367–383, 2013.

[77] Y. Zhou, J. Han, X. Yuan, Z. Wei, and R. Hong. Inverse sparse group lasso model for robust object tracking. *IEEE Transactions on Multimedia*, 19(8):1798–1810, 2017.

[78] B. Zhuang, H. Lu, Z. Xiao, and D. Wang. Visual tracking via discriminative sparse similarity map. *IEEE Transactions on Image Processing*, 23(4):1872–1881, 2014.

Meihui Li received her bachelor's degree in Information Display and Optoelectronic Technology from University of Electronic and Science Technology of China (UESTC), Chengdu, Sichuan, China, in 2014. She is now a Ph.D. candidate in Signal and information processing at UESTC. She has been with the Department of Electrical and Computer Engineering of North Carolina State University in 2017. Her research interests include computer vision and machine learning.

Lingbing Peng received the B.Sc. degree in communications engineering from the University of Electronic Science and Technology of China, China, in 2012, where she is currently pursuing the Ph.D. degree in Communications and information systems. Her research interests include signal processing, image processing, and target recognition and tracking.

Tianfu Wu is an assistant professor in the Department of Electrical and Computer Engineering at NC state university (NCSU). He received his Ph.D. in statistics from UCLA under the supervision by Prof. Song-Chun Zhu. His research focuses on computer vision, often motivated by the task of building explainable and improvable visual Turing test and robot autonomy through life-long communicative learning. To accomplish his research goals, he is interested in pursuing a unified framework for machines to ALTER (Ask, Learn, Test, Explain and Refine) recursively in a principled way.

Zhenming Peng (M'06) received his Ph.D. degree in geo-detection and information technology from the Chengdu University of Technology, Chengdu, China, in 2001. From 2001 to 2003, he was a Post-Doctoral Researcher with the Institute of Optics and Electronics, Chinese Academy of Sciences, Chengdu, China. He is currently a Professor with the University of Electronic Science and Technology of China, Chengdu. His research interests include image processing, signal processing, and target recognition and tracking. Prof. Peng is a member of Institute of Electrical and Electronics Engineers (IEEE), Optical Society of America (OSA), China Optical Engineering Society (COES), and Chinese Society of Astronautics (CSA).