Visual Tracking via Nonnegative Multiple Coding

Fanghui Liu, Chen Gong, *Member, IEEE*, Tao Zhou, Keren Fu, Xiangjian He, *Senior Member, IEEE*, and Jie Yang

***Abstract*—It has been extensively observed that an accurate appearance model is critical to achieving satisfactory performance for robust object tracking. Most existing top-ranked methods rely on linear representation over a single dictionary, which brings about improper understanding on the target appearance. To address this problem, in this paper, we propose a novel appearance model named as "nonnegative multiple coding" (NMC) to accurately represent a target. First, a series of local dictionaries are created with different predefined numbers of nearest neighbors, and then the contributions of these dictionaries are automatically learned. As a result, this ensemble of dictionaries can comprehensively exploit the appearance information carried by all the constituted dictionaries. Second, the existing methods explicitly impose the nonnegative constraint to coefficient vectors, but in the proposed model, we directly deploy an efficient ℓ_2 norm regularization to achieve the similar nonnegative purpose with theoretical guarantees. Moreover, an efficient occlusion detection scheme is designed to alleviate tracking drifts, which investigates whether negative templates are selected to represent the severely occluded target. Experimental results on two benchmarks demonstrate that our NMC tracker are able to achieve superior performance to state-of-the-art methods.**

***Index Terms*—Approximated locality-constrained linear coding, nonnegative constraint, occlusion detection, visual tracking.**

I. Introduction

VISUAL tracking aims to track the interested object in a video sequence given its precise location at the first frame. It plays a significant role in computer vision with broadly applications, such as video surveillance, human interaction, and vehicle navigation [1], [2]. Although a large number of approaches such as those in [3]–[5] have been developed, it is still difficult to find a lasting solution to achieve a robust tracking performance due to its intrinsic factors (e.g. fast motion, rotation in-plane or out-of-plane, and shape deformation) and extrinsic factors (e.g. partial occlusions, illumination variation, and background clutter).

In visual tracking, satisfactory performance depends on the accurate appearance modeling for the target. According to the adopted appearance model, the tracking methods can be categorized into two classes: generative trackers [6], [7] and discriminative trackers [8]–[10]. The generative methods attempt to find the most similar candidate region to the target by the minimal reconstruction error; whereas discriminative methods cast the tracking problem as a binary classification problem [11], [12] to separate the foreground target from the background. Our method belongs to the former one, and in the next we will briefly review the typical generative trackers.

Extensive research in generative methods demonstrate that sparse representation has been successfully used in visual tracking. The rationale of sparse representation based methods is that the target can be sparsely represented by the atoms in a dictionary $\mathbf{T}$ (e.g. a target template set in visual tracking area) with a sparse coefficient vector. According to the adopted representation scheme, these trackers can be grouped into global template representation [13]–[15], local sparse model [16]–[18], joint sparse appearance model [19], [20], sparse collaborative model [21], and structured sparse model [22]. These methods learn the target representation from different cues and thus show promising performance against various challenging factors. To be specific, in a local sparse model [16], [17], the local patches within a possible target candidate are sparsely represented by the patches in a dictionary. The joint sparse appearance model [19], [20] exploits the intrinsic relationship among particles to represent the target jointly. The sparse collaborative model [21] combines the advantages of generative and discriminative methods, and thus is able to exploit both holistic templates and local representations to describe the target. Furthermore, the structural sparse appearance model [22] not only exploits the intrinsic relationship among target candidates through sparse representations, but also retains the spatial structure among the local patches within every target candidate.

The above sparse representation based approaches are often used to model the appearance of object through a linear representation way, in which the mapping relationship between the original sample space and coding space has been largely ignored. When the dictionary $\mathbf{T}$ is used to represent the target, the similarity between two samples should be reflected in both the corresponding dictionaries and coefficient vectors. To be specific, Local Coordinate Coding (LCC) [23] is proposed to investigate the relationship among data points. It uses a set

Manuscript received August 26, 2016; revised March 31, 2017 and May 12, 2017; accepted May 20, 2017. Date of publication May 26, 2017; date of current version November 15, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61572315, Grant 6151101179, and Grant 61602246, in part by 863 Plan of China under Grant 2015AA042308, and in part by the China Postdoctoral Science Foundation under Grant 2016M601597. This associate editor coordinating the review of this manuscript and approving it for publication was Prof. Abdulmotaleb El Saddik. *(Corresponding author: Jie Yang.)*

F. Liu, T. Zhou, and J. Yang are with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: lfhsgre@outlook.com; zhou.tao@sjtu.edu.cn; jieyang@sjtu.edu.cn).

C. Gong is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: chen.gong@njust.edu.cn).

K. Fu is with the College of Computer Science, Sichuan University, Chengdu 610065, China (e-mail: fkrsuper@gmail.com).

X. He is with the School of Computing and Communications, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: Xiangjian.He@uts.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2017.2708424

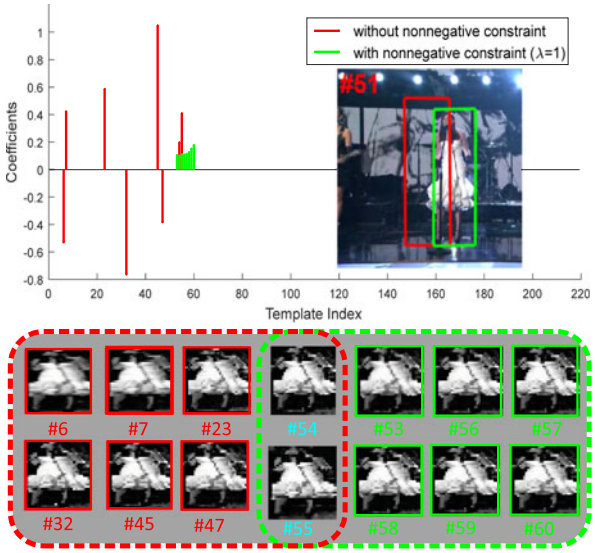


Fig. 1. Illustration of two tracking results in the sequence *Singer1*. The good tracking result (in green) is represented by eight positive template images (#53, #54, #55, #56, #57, #58, #59, and #60) with nonnegative coefficients. The bad tracking result (in red) is represented by eight selected positive template images (#6, #7, #23, #32, #45, #47, #54, and #55) with both positive and negative coefficients as illustrated by the figure in the top corner.

of anchor points to create a dictionary, and each data point is linearly represented by only a few anchor points. Locality-constrained Linear Coding (LLC) [24] investigates the similarity between data points and their corresponding dictionaries. These coding methods have been successfully incorporated into a visual tracking framework [25]–[27]. In [26], the authors apply the histograms of sparse coefficients encoded by the LLC algorithm and a local optimal searching scheme for object tracking. Unlike other works, LLC in [28] is employed for state search instead of appearance modeling. The sampling procedure is formulated as an optimization problem, and thus it can be efficiently solved on a convex hull.

However, there are two main drawbacks of these sparse coding based trackers. First, these methods simply use a fixed number of k-nearest neighbors to construct the local dictionary. This single dictionary cannot accurately and adequately capture the appearance of the target due to the insufficient emphasis on the target appearance information. As a result, they cannot tackle appearance variations of the target, which leads to tracking drifts easily. Furthermore, it is usually difficult to determine the number of neighbors k in advance, and an improper choice of k may degrade the tracking performance. Second, there is no nonnegative constraint on the coefficient vectors, where such constraint, as we shown in this paper, is essential for better tracking performance. Fig. 1 shows two tracking results reconstructed by two sets of selected positive templates in the LLC method with/without the nonnegative constraint. Although both the good tracking result (in a green bounding box) and the bad result (in a red bounding box) are represented by the positive template set with the indices ranging from #1 to #60, the bad result suffers severe drifts and contains much background information. This is because the bad tracking result is reconstructed by some positive template images with negative coefficients (e.g. #6, #32, and #47), whereas all the coefficients for the good result are nonnegative. Therefore, we argue that positive templates with negative coefficients would lead to an unreliable tracking result.

Based on the above observations, this paper develops a robust Nonnegative Multiple Coding (NMC) tracker by exploiting an ensemble of multiple dictionaries and the nonnegative constraint to accurately character the target appearance. The contributions of this paper are summarized as follows.

1) A series of local dictionaries are built with regards to different pre-defined numbers of nearest neighbors, and their related weights are automatically learned in our NMC model. Thereby, the obtained coefficient vectors and the ensemble of multiple dictionaries can be learned in a uniform framework.
2) We demonstrate through both theory and experiments that the incorporation of ℓ_2 norm regularization term to our NMC model is able to make the obtained coefficient vector nonnegative. It achieves the similar effect with the exact nonnegative constraint with the provable guarantee of the lower bounded regularization parameter.
3) The occlusion detection criterion is proposed to mitigate drifting problem which investigates whether the negative templates (i.e. background) are used to represent the target.

This paper is the extended version of our previous work [29]. The tracker described here differs from [29] in several aspects. Firstly, to update the positive template set, PCA-based vectors and additional trivial templates are used to represent the target by the approximated LLC method. The updating scheme is accomplished by gradually absorbing the latest positive samples and eliminating out-of-date samples. Secondly, we theoretically prove that the introduced ℓ_2 norm regularization is able to achieve nonnegative effects if its regularization parameter is larger than a lower bound. Thirdly, Histogram of Gradient (HOG) feature is introduced into our tracking framework to improve the performance. Lastly, we present more experimental results on two benchmark datasets, and investigate the parametric sensitivity and computational complexity analysis of the proposed tracker.

The remainder of the paper is organized as follows. Section II explains the details of the proposed Nonnegative Multiple Coding model. Section III introduces the proposed NMC tracker. Section IV shows the evaluation results of the proposed tracker with other state-of-the-art methods on two popular benchmarks. Finally, conclusion is given in Section V.

II. Nonnegative Multiple Coding Model

In this section, we first provide an overview of the LLC method that will be used in our NMC model, and then detail the employed nonnegative constraint and dictionary ensemble. Particularly, the nonnegative constraint is replaced by the ℓ_2 norm regularization term to achieve the similar nonnegative purpose with theoretical guarantees.

A. Review of LLC and Approximated LLC

The idea of the LLC method [24] implies that if two data points are close to each other in the original space, they should be represented by two similar dictionaries, and thus have similar

encoding coefficients. To improve its efficiency, LLC applies a locality constraint to select similar bases, and then it learns a linear combination of these bases in the local dictionary $\mathbf{B}$ with a coefficient vector $\mathbf{c}_i$ to represent a data point $\mathbf{y}_i \in \mathbb{R}^M$ as follows:

$$\min_{\mathbf{C}=[\mathbf{c}_1,\cdots,\mathbf{c}_N]} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{B}\mathbf{c}_i\|_2^2 + \lambda \|\mathbf{d}_i \odot \mathbf{c}_i\|_2^2$$
$$\text{s.t.} \qquad \mathbf{1}^\top \mathbf{c}_i = 1, \forall i = 1, 2, \cdots, N \tag{1}$$

where N is the number of data points, and the constraint $\mathbf{1}^\top \mathbf{c}_i = 1$ ensures shift-invariant where $\mathbf{1}$ denotes the all-one vector. The distance $\mathbf{d}_i = \exp(\frac{\text{dist}(\mathbf{y}_i, \mathbf{B})}{\sigma})$ with σ being a scale factor parameter is defined by $\text{dist}(\mathbf{y}_i, \mathbf{B}) = [\text{dist}(\mathbf{y}_i, \mathbf{b}_1), \text{dist}(\mathbf{y}_i, \mathbf{b}_2), \cdots, \text{dist}(\mathbf{y}_i, \mathbf{b}_N)]^\top$, where the distance metric $\text{dist}(\mathbf{y}_i, \mathbf{b}_j)$ represents the Euclidean distance between $\mathbf{y}_i$ and $\mathbf{b}_j$.

The approximated LLC method [24], as a simplified version of LLC, takes local sparsity into consideration. To be specific, it simply selects k nearest neighbors of $\mathbf{y}_i$ as the local dictionary $\mathbf{B}_i$ instead of using the metric $\text{dist}(\mathbf{y}_i, \mathbf{B})$. In this way, the approximated LLC method not only obtains local similarity but also achieves sparsity due to the judicious selection of k nearest neighbors. Hence, a test sample $\mathbf{y}_i$ can be sparsely represented by the linear combination of several local base vectors in the local dictionary $\mathbf{B}_i$ with the coefficient vector $\mathbf{c}_i$ as follows:

$$\min_{\mathbf{c}_i} \|\mathbf{y}_i - \mathbf{B}_i \mathbf{c}_i\|_2^2 \quad \text{s.t. } \mathbf{1}^\top \mathbf{c}_i = 1, \forall i = 1, 2, \cdots, N \tag{2}$$

of which the closed-form solution is $\mathbf{c}_i = [(\mathbf{B}_i^\top - \mathbf{1}\mathbf{y}_i^\top)(\mathbf{B}_i - \mathbf{y}_i \mathbf{1}^\top)]^{-1}\mathbf{1}$.

B. Nonnegative Constraint on Approximated LLC

Note that, in nonnegative matrix factorization (NMF) [30], a nonnegative data point is modeled as the nonnegative linear combination of a set of nonnegative bases, which could implicitly grab the structure information contained within data distribution. Therefore it is straightforward to require the obtained coefficients in our NMC model to be nonnegative. Specifically, the illustration in the introduction also demonstrates the superiority of adopting the nonnegative coefficients (see Fig. 1). In this paper, higher priority is given to adopt the nonnegative coefficients based on (2), which is written as

$$\min_{\mathbf{c}_i} \left\|\mathbf{y}_i - \mathbf{B}_i \mathbf{c}_i\right\|_2^2$$
$$\text{s.t. } \mathbf{1}^\top \mathbf{c}_i = 1,$$
$$\mathbf{c}_i \geq 0, \quad \forall i = 1, 2, \cdots, N. \tag{3}$$

Several optimization algorithms for such quadratic programming problem with linear constraint have been proposed, such as interior point method, accelerated proximal gradient method (APG) [15], and alternating direction method of multipliers (ADMM) [32]. However, (3) contains a nonnegative inequality constraint, so the above methods cannot yield a closed-form solution here. Consequently, above iterative algorithms are not very efficient for visual tracking. To tackle this problem, we introduce a conventional ℓ_2 norm regularization term to replace the nonnegative constraint to achieve the similar nonnegative effect. The rationality of such replacement will be theoretically demonstrated in Section II-D with provable guarantees. Herein, (3) is reformulated as

$$\min_{\mathbf{c}_i} \|\mathbf{y}_i - \mathbf{B}_i \mathbf{c}_i\|_2^2 + \lambda \|\mathbf{c}_i\|_2^2 \quad \text{s.t. } \mathbf{1}^\top \mathbf{c}_i = 1 \tag{4}$$

where λ is the tuning parameter. Let γ be the Lagrange multiplier for the shift-invariant constraint $\mathbf{1}^\top \mathbf{c}_i = 1$, then the related Lagrange function can be formulated as

$$\mathcal{L} = \mathbf{c}_i^\top (\mathbf{y}_i \mathbf{1}^\top - \mathbf{B}_i)^\top (\mathbf{y}_i \mathbf{1}^\top - \mathbf{B}_i)\mathbf{c}_i + \lambda \mathbf{c}_i^\top \mathbf{c}_i + \gamma(1 - \mathbf{1}^\top \mathbf{c}_i). \tag{5}$$

The partial derivative of $\mathcal{L}$ with respect to $\mathbf{c}_i$ is

$$\frac{\partial \mathcal{L}}{\partial \mathbf{c}_i} = 2(\mathbf{y}_i \mathbf{1}^\top - \mathbf{B}_i)^\top (\mathbf{y}_i \mathbf{1}^\top - \mathbf{B}_i)\mathbf{c}_i + 2\lambda \mathbf{c}_i - \gamma \mathbf{1}. \tag{6}$$

By using the Karush-Kuhn-Tucker condition, the analytic solution can be obtained as

$$\mathbf{c}_i = \frac{\gamma}{2}[(\mathbf{B}_i^\top - \mathbf{1}\mathbf{y}_i^\top)(\mathbf{B}_i - \mathbf{y}_i \mathbf{1}^\top) + \lambda \mathbf{I}]^{-1}\mathbf{1}. \tag{7}$$

Specifically, γ can be further obtained by the shift-invariant constraint. In this case, we can directly obtain the closed-form solution $\mathbf{c}_i$ without employing or designing any inefficient iterative method for this particular purpose.

C. Ensemble of Multiple Local Dictionaries

With regards to the existing problems, the single dictionary cannot faithfully represent the appearance of the target. In this paper, we mainly focus on designing an ensemble of dictionaries to sufficiently capture the appearance information revealed by the target. The local dictionaries $(\mathbf{B}_i^1, \mathbf{B}_i^2, \cdots, \mathbf{B}_i^m)$ for every sample $\mathbf{y}_i$ with different numbers of nearest neighbors are constructed by

$$\begin{cases} \min_{\mathbf{c}_i^1} \|\mathbf{y}_i - \mathbf{B}_i^1 \mathbf{c}_i^1\|_2^2 \quad \text{s.t. } \mathbf{c}_i^1 \geq 0, \ \ \mathbf{1}^\top \mathbf{c}_i^1 = 1; \\ \min_{\mathbf{c}_i^2} \|\mathbf{y}_i - \mathbf{B}_i^2 \mathbf{c}_i^2\|_2^2 \quad \text{s.t. } \mathbf{c}_i^2 \geq 0, \ \ \mathbf{1}^\top \mathbf{c}_i^2 = 1; \\ \cdots \\ \min_{\mathbf{c}_i^m} \|\mathbf{y}_i - \mathbf{B}_i^m \mathbf{c}_i^m\|_2^2 \ \text{ s.t. } \mathbf{c}_i^m \geq 0, \ \mathbf{1}^\top \mathbf{c}_i^m = 1. \end{cases} \tag{8}$$

where $\mathbf{B}_i^j \in \mathbb{R}^{M \times k_j}$ is established from the sample $\mathbf{y}_i$'s k_j nearest neighbors, and the associated coefficient vector is $\mathbf{c}_i^j \in \mathbb{R}^{k_j}$. After obtaining these coefficient vectors $(\mathbf{c}_i^1, \mathbf{c}_i^2, \cdots, \mathbf{c}_i^m)$, the next step is to accurately integrate these vectors into a final coefficient vector. By introducing a weight vector $\mathbf{w} = [w_1, w_2, \cdots, w_m]^\top$, which satisfies $\sum_{s=1}^m w_s = 1$ and $w_s \geq 0$ for $s = 1, 2, \cdots, m$, the weight vector $\mathbf{w}$ can be obtained by optimizing the following objective function:

$$\min_{\mathbf{w}} \left\|\mathbf{y}_i - \sum_{j=1}^{m} w_j (\mathbf{B}_i^j \mathbf{c}_i^j)\right\|_2^2$$
$$\text{s.t. } \ \mathbf{1}^\top \mathbf{w} = 1, \quad w_j \geq 0 \quad \forall j = 1, 2, \cdots, m. \tag{9}$$

Furthermore, by defining $\mathbf{D} = [\mathbf{B}_i^1 \mathbf{c}_i^1, \mathbf{B}_i^2 \mathbf{c}_i^2, \ldots, \mathbf{B}_i^m \mathbf{c}_i^m] \in \mathbb{R}^{M \times m}$, we formulate (9) as

$$\min_{\mathbf{w}} \left\|\mathbf{y}_i - \mathbf{D}\mathbf{w}\right\|_2^2 \ \text{ s.t. } \mathbf{1}^\top \mathbf{w} = 1, w_j \geq 0 \ \forall j. \tag{10}$$

This objective function shares the same expression as the approximate LLC problem with the nonnegative constraint in (3).

Hence, an ℓ_2 norm regularization term can be added to (10) to achieve a similar nonnegative effect, that is

$$\min_{\mathbf{w}} \|\mathbf{y}_i - \mathbf{D}\mathbf{w}\|_2^2 + \beta\|\mathbf{w}\|_2^2 \ \text{ s.t. } \mathbf{1}^\top\mathbf{w} = 1,\ \forall j \tag{11}$$

where β is the regularization parameter. Similar to the solution of (4), the optimal $\mathbf{w}$ of (11) is $\mathbf{w} = [(\mathbf{D}^\top - \mathbf{1}\mathbf{y}_i^\top)(\mathbf{D} - \mathbf{y}_i\mathbf{1}^\top) + \beta\mathbf{I}]^{-1}\mathbf{1}$.

D. Theoretical Analysis on ℓ_2 Norm Regularization

As mentioned in Section II-B, the adopted ℓ_2 norm regularization shows the similar effect with the hard nonnegative constraint. In this section, we theoretically demonstrate the requirements for the nonnegative coefficient vector with theoretical guarantees. As both (4) and (11) incorporate the ℓ_2 norm to generate nonnegative coefficient vectors $\mathbf{c}_i$ and $\mathbf{w}$ respectively, here we study their general formation, namely

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{B}\mathbf{x}\|_2^2 + \alpha\|\mathbf{x}\|_2^2 \ \text{ s.t. } \mathbf{1}^\top\mathbf{x} = 1 \tag{12}$$

where $\mathbf{y}$ is a data point, $\mathbf{B} \in \mathbb{R}^{M\times k}$ is the base matrix, $\mathbf{x}$ is the coefficient vector, and α is the tuning parameter that represents λ and β in (4) and (11) respectively. Likewise, the optimizer of (12) is

$$\mathbf{x} = [(\mathbf{B}^\top - \mathbf{1}\mathbf{y}^\top)(\mathbf{B} - \mathbf{y}\mathbf{1}^\top) + \alpha\mathbf{I}]^{-1}\mathbf{1}\,. \tag{13}$$

By defining $\mathbf{A} = \mathbf{B} - \mathbf{y}\mathbf{1}^\top$ and $\mathbf{F} = (\mathbf{A}^\top\mathbf{A} + \alpha\mathbf{I}) \in \mathbb{R}^{k\times k}$, the solution in (13) can be reformulated as

$$\mathbf{x} = (\mathbf{A}^\top\mathbf{A} + \alpha\mathbf{I})^{-1}\mathbf{1} = \mathbf{F}^{-1}\mathbf{1}\,. \tag{14}$$

Specifically, when α is extremely large, the optimization problem in (12) approaches to the following problem:

$$\min_{\mathbf{x}} \ \alpha\|\mathbf{x}\|_2^2 \ \text{ s.t. } \mathbf{1}^\top\mathbf{x} = 1\,. \tag{15}$$

It is obvious that the minimizer of (15) is $x_1 = x_2 = \cdots = x_k = \frac{1}{k}$. This formula implies that when the tuning parameter α approaches to infinity, the elements in $\mathbf{x}$ tend to be nonnegative and equivalent. It means that, there must exist a lower bound of α that ensures the coefficient vector $\mathbf{x}$ to be nonnegative. Next the following theorem will demonstrate how to compute this bound.

Theorem 1: Let $\mathbf{A}^\top\mathbf{A}$ be a $k \times k$ matrix, then the coefficient vector $\mathbf{x} = (\mathbf{A}^\top\mathbf{A} + \alpha\mathbf{I})^{-1}\mathbf{1}$ in (13) is nonnegative if α satisfies

$$\alpha \geq (k+1)\left\|\mathbf{A}^\top\mathbf{A} - 2\Gamma\right\|_\infty + k\left\|\mathbf{A}^\top\mathbf{A}\right\|_\infty \tag{16}$$

where Γ is a diagonal matrix with its ith diagonal element $(\mathbf{A}^\top\mathbf{A})_{ii}$.

Proof: The proof is presented in Appendix A. ■

Theorem 1 theoretically demonstrates that replacing the nonnegative constraint by ℓ_2 norm regularization term is reasonable and effective if α is lower bounded. Therefore, in all experiments we simply set α (i.e. λ and β) to be its lower bound rather than empirically choosing this parameter.

III. Nonnegative Multiple Coding Tracker

Visual tracking problem is usually formulated as a particle filter framework [31], [35]. The implicit rationale behind particle filter is to estimate the posterior distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, which can be approximately computed by a finite set of randomly sampled particles. Given some observed image patches at the tth frame $\mathbf{y}_{1:t} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t\}$, the state of the target $\mathbf{x}_t$ can be estimated recursively by the following two stages:

$$\begin{aligned}\text{Prediction}:\ p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) &= \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) \\ &\quad \times\ p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\mathrm{d}\mathbf{x}_{t-1}, \\ \text{Updating}:\ p(\mathbf{x}_t|\mathbf{y}_{1:t}) &\propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1}).\end{aligned} \tag{17}$$

By defining the target state $\mathbf{x}_t = [l_x, l_y, \theta, s, \alpha, \phi]^\top$, where l_x, l_y, θ, s, α, and ϕ denote the translations in the direction of x and y, rotation angle, scale, aspect ratio, and skew respectively, we apply the affine transformation with the above six parameters to model the target motion. The motion model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ represents the state transition of the target between two consecutive frames. The observation model $p(\mathbf{y}_t|\mathbf{x}_t)$ denotes the likelihood of the observation $\mathbf{y}_t$ at state $\mathbf{x}_t$. In our method, the observation model is formulated by our NMC model, which reflects the similarity between a candidate sample and the target templates. The optimal state at the tth frame is obtained by maximizing the posterior probability

$$\mathbf{x}_t^* = \underset{\mathbf{x}_t^i}{\operatorname{argmax}}\, p(\mathbf{y}_t^i|\mathbf{x}_t^i)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}), i = 1, 2, \cdots, N \tag{18}$$

where $\mathbf{x}_t^i$ is the ith sample of the state $\mathbf{x}_t$, and $\mathbf{y}_t^i$ is a candidate sample predicted by $\mathbf{x}_t^i$.

Here we briefly introduce how to obtain the templates module in the beginning frames of a video sequence. To better capitalize on the distinction between the foreground and background, plenty of negative templates are collected apart from positive templates. In other words, the template set $\mathbf{T}$ contains positive and negative template sets, defined as $\mathbf{T}^{pos} = [\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_p]$ and $\mathbf{T}^{neg} = [\mathbf{T}_{p+1}, \mathbf{T}_{p+2}, \ldots, \mathbf{T}_{p+n}]$, where p and n denote the numbers of positive and negative templates respectively. The base dictionary $\mathbf{B}_i = [\mathbf{B}_i^{pos}, \mathbf{B}_i^{neg}] \in R^{M\times k}$ is built by selecting k nearest neighbors of a candidate sample $\mathbf{y}_i$ from the template set $\mathbf{T}$.

A. Observation Model

The proposed NMC model outputs a coefficient vector $\mathbf{c}$ to calculate the reconstruction error $\|\mathbf{y} - \mathbf{T}\mathbf{c}\|_2^2$ that represents the similarity between a candidate and the target. A small reconstruction error indicates that the test sample $\mathbf{y}$ is similar to the target with a high probability, and vice versa. The detailed implementation issues are explained as follows.

Under the aforementioned results, the local dictionary $\mathbf{B}_i^j \in \mathbb{R}^{M\times k_j}$ is established by k_j ($j = 1, 2, \cdots, m$) nearest neighbors of a candidate sample $\mathbf{y}_i$, which are selected from the template set $\mathbf{T}$. To record the selected templates in $\mathbf{B}_i^j$, we define a selection vector $\mathbf{d}_i^j \in \mathbb{R}^{p+n}$ of which the dimensionality equals to the number of templates. Thereby $\mathbf{c}_i^j \in \mathbb{R}^{k_j}$ in (8) is plugged into $\mathbf{d}_i^j$ where the locations of these k_j elements correspond to the indices of the selected templates in $\mathbf{T}$.

After that, once the weight vector $\mathbf{w}$ and the coefficient vector $\mathbf{d}_i^j$ are successfully obtained, the final coefficient vector $\mathbf{d}_i \in \mathbb{R}^{p+n}$ is a linear combination of $\mathbf{d}_i^j$ ($j = 1, 2, \cdots, m$) weighted by the elements in the vector $\mathbf{w}$. The final coefficient vector $\mathbf{d}_i$ is solved by Algorithm 1.

Algorithm 1: Algorithm for the final coefficient vector $\mathbf{d}_i$

Input: a candidate $\mathbf{y}_i$, local dictionaries: $\mathbf{B}_i^1, \mathbf{B}_i^2, \cdots, \mathbf{B}_i^m$.

Output: the final coefficient vector $\mathbf{d}_i$ related to $\mathbf{y}_i$.

Set: iteration number η.
for $j = 1 : m$ **do**
  Compute $\mathbf{c}_i^j$ by (4);
  Obtain the selection vector $\mathbf{d}_i^j$;
end
Solve the weight vector $\mathbf{w}$ by (11).
Obtain the final coefficient vector $\mathbf{d}_i = \sum_{j=1}^{m} w_j \mathbf{d}_i^j$.

In our method, $\mathbf{d}_i \in \mathbb{R}^{p+n}$ is divided into two parts: $\mathbf{d}_i = [\mathbf{d}_i^{pos}; \mathbf{d}_i^{neg}]$ with respect to $\mathbf{T}^{pos}$ and $\mathbf{T}^{neg}$ respectively, where $\mathbf{d}_i^{pos} \in \mathbb{R}^p$ and $\mathbf{d}_i^{neg} \in \mathbb{R}^n$. Then the observation model for the candidate $\mathbf{y}_i$[1] is formulated as

$$p(\mathbf{y}_i|\mathbf{x}_i) \propto exp(-\psi(\varepsilon_i^{pos} - \varepsilon_i^{neg})) \tag{19}$$

where $\varepsilon_i^{pos} = \|\mathbf{y}_i - \mathbf{T}^{pos}\mathbf{d}_i^{pos}\|_2^2$ is the reconstruction error between the candidate $\mathbf{y}_i$ and the positive template set $\mathbf{T}^{pos}$. Likewise, $\varepsilon_i^{neg} = \|\mathbf{y}_i - \mathbf{T}^{neg}\mathbf{d}_i^{neg}\|_2^2$ is the reconstruction error between the candidate $\mathbf{y}_i$ and the negative template set $\mathbf{T}^{neg}$. The parameter ψ is a normalization factor fixed to 2.5 in our experiments.

B. Occlusion Detection Scheme

In tracking process, severe occlusion is one of the most significant issues which lead to tracking drifts or failures. Therefore, designing a suitable occlusion detection module is important to ensure accurate appearance modeling and avoid unexpected tracking drifts.

Generally, the target without occlusions will be entirely represented by positive templates. If the target suffers from severe occlusions, negative templates will be selected to represent the target with a high probability. Such scenario motivates us to propose an efficient approach to detect severe occlusions. The occlusion detection criterion is mainly based on whether negative templates are selected to represent the tracking target. If several negative templates are used to reconstruct the target, the target probably suffers from serious occlusions, and vice versa. In the proposed NMC tracker, if more than one negative template (the number of selected negative templates is denoted as $\mathrm{Num}(\mathrm{neg}_*)$) are used to represent the target, the target in the current frame is regarded as "occluded". In this case, the model updating should be strictly prohibited, otherwise the appearance model would be contaminated by inaccurate positive templates.

Fig. 2 shows an intuitive example of an occluded target represented by the templates. At frame #528, the target is recognised as "occluded" because a considerable amount of negative templates (i.e. 4) are participated in constructing the target. In this case, the current tracking result cannot be used for model updating as explained above. Apart from this qualitative explanation, the quantitative results are also provided to verify such criterion in Section IV-D.

[1]Note that the time index t is omitted for simplicity.

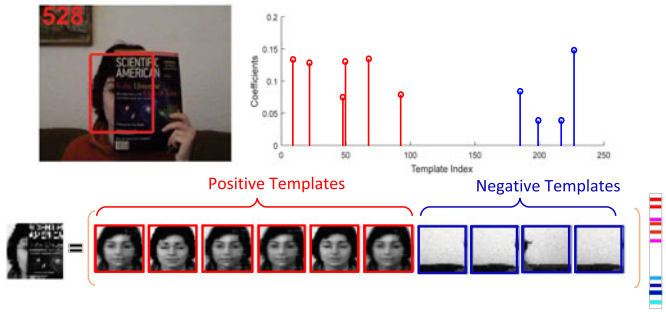


Fig. 2. Occluded target in the sequence *FaceOcc1* represented by six positive templates and four negative templates with nonnegative coefficients.

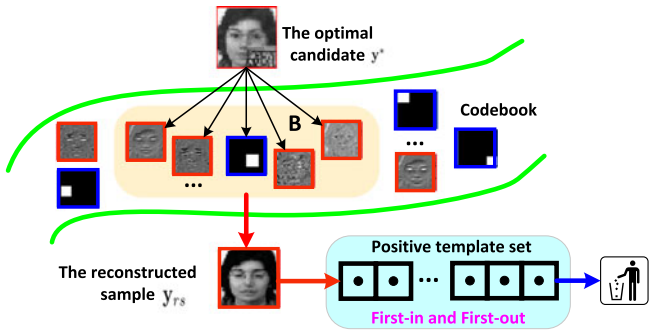


Fig. 3. Template updating scheme for positive template set.

C. Template Updating Scheme

Usually, tracking with fixed templates is prone to fail in dynamic scenes due to the inevitable appearance changes. However, constantly updating the template set easily leads to error accumulation and tracking drifts. According to this, we propose an incremental updating scheme for the positive template set based on the approximated LLC method. The flowchart of this updating scheme is shown in Fig. 3. Similar to [37], the estimated target can be modeled by a linear combination of PCA base vectors and additional trivial templates. In the proposed NMC tracker, the local dictionary $\mathbf{B}$ is composed of k nearest neighbors of the optimal candidate $\mathbf{y}^*$ from the codebook as shown in Fig. 3, which arrives at

$$\mathbf{y}^* = \mathbf{B}\mathbf{c} = \mathbf{U}\mathbf{x} + \mathbf{e} = \underbrace{[\mathbf{U} \quad \mathbf{I}]}_{\triangleq \mathbf{B}} \begin{bmatrix} \mathbf{x} \\ \mathbf{e} \end{bmatrix} \tag{20}$$

where $\mathbf{y}^*$ is the current optimal candidate sample, $\mathbf{U}$ is the matrix composed of PCA base vectors in the local dictionary $\mathbf{B}$, $\mathbf{x}$ is the corresponding PCA coefficient vector, and $\mathbf{e}$ is assumed to be sparse noise. The coefficient vector $\mathbf{c}$ can be obtained by (4) in the proposed NMC model. In this formula, we use the reconstructed sample $\mathbf{y}_{rs} = \mathbf{U}\mathbf{x}$ for positive template set updating rather than the current optimal candidate $\mathbf{y}^*$. Compared to directly using the current tracking result $\mathbf{y}^*$ with the reconstruction error smaller than a pre-defined threshold as in [29], in the proposed template updating scheme, the appearance model can effectively avoid being contaminated when the optimal candidate $\mathbf{y}^*$ is slightly occluded. For example, in Fig. 3, the reconstructed sample $\mathbf{y}_{rs}$ is much more accurate than the optimal candidate $\mathbf{y}^*$.

Algorithm 2: Algorithm for the NMC tracker

Initialization: Extract templates $\mathbf{T}$ in the first 5 frame.
for $t = 6$ *to the end of the sequence* **do**
  N particles $\mathbf{Y}_{1:N}$ are sampled;
  for $i = 1 : N$ **do**
    Construct local dictionaries $\mathbf{B}_i^1$, $\mathbf{B}_i^2$,...,$\mathbf{B}_i^m$ for each candidate $\mathbf{y}_i$;
    Obtain $\mathbf{d}_i$ by **Algorithm 1**;
    Calculate the likelihood of $\mathbf{y}_i$ by (19) in the observation model;
  end
  Obtain the optimal candidate $\mathbf{y}^*$ and the reconstructed sample $\mathbf{y}_{rs}$ by (20);
  Update: **for** *each* 5 *frames* **do**
    Update the negative template set $\mathbf{T}^{neg}$;
    if $Num(neg_*) \leq 1$ **then**
      Update the positive template set;
    end
  end
end

Furthermore, to save the storage space and reduce the computational cost, the "First-in and First-out" procedure is used to maintain the number of positive templates. In this way, the newly reconstructed sample $\mathbf{y}_{rs}$ is added to the template set, while the oldest positive template is thrown away.

If the occlusion is not detected, the positive and negative template sets are normally updated every 5 frames by our template updating scheme. Once an occlusion is detected, the positive template set should stop updating, while the negative template set is still regularly updated for every 5 frames.

Finally, the flowchart of the NMC tracker is summarized in Algorithm 2.

IV. Experiments

This section evaluates the proposed NMC tracker on two benchmarks including Princeton Tracking Benchmark (PTB) [38] and Object Tracking Benchmark (OTB) [39]. We perform an extensive comparison of our NMC tracker with the existing state-of-the-art trackers on two benchmarks. The compared methods include SST [22], KCF [10], LNLT [25], and IMT [40]. In addition, the results on parameter sensitivity and computational complexity analyses are also provided.

Setup: The proposed NMC tracker was implemented in Matlab on a PC with Intel Xeon E5506 CPU (2.13 GHz) and 24 GB memory. The following parameters were used for our tests: each sample (i.e. the patch of image) was normalized to 32×32 pixels ($M = 1024$); the number of positive templates and negative templates p and n were set to 100 and 150 respectively; three local dictionaries (i.e. $m = 3$) were selected with three different numbers of nearest neighbors $k_1 = 5$, $k_2 = 8$, and $k_3 = 10$; the tuning parameters λ and β for ℓ_2 norm regularization were chosen by (16); In the template updating scheme, the number of PCA base vectors was set to 16, and eight nearest neighbors from the codebook were selected to represent the optimal candidate $\mathbf{y}^*$.

Evaluation metrics: In PTB and OTB, two fundamental metrics namely mean center location error (CLE) and the Pascal VOC Overlap Ratio (VOR) [41] are used to evaluate the tracking performances of all the compared methods. CLE is defined as the pixel distance between the centroid of the tracking result and the ground truth, in which small CLE value indicates a precise tracking result. The overlap ratio measures the overlapping rate between the bounding box yielded by a tracker and the ground truth box. It is defined as $o = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)}$, where R_T and R_G denote the area of bounding boxes of the tracker and ground truth, respectively.

A. Results on PTB

The Princeton Tracking Benchmark contains 95 video clips collected by both RGB and depth sensors from a standard Microsoft Kinect 1.0. These sequences are grouped into the following aspects: target type (human, animal and rigid), target size (large and small), movement (slow and fast), presence of occlusion, and motion type (passive and active). For fair comparison, the proposed tracker is only compared with RGB competitors without considering the depth information. The nine colored trackers includes KCF [10], Struck [42], VTD [43], CT [44], TLD [45], MIL [46], SemiB [47], RGBdet [38] and OF [38].

The average overlap ratio and ranking results of these methods are shown in Table I. One can see that our method ranks first, and it is followed by the KCF tracker. From the target type aspect, the proposed NMC tracker performs better on human and animal target than KCF method due to an accurate appearance model. Besides, the proposed tracker achieves promising performance in terms of target size, fast movement, occlusion, and active motion when compared with other methods.

B. Results on OTB

Object Tracking Benchmark includes 29 trackers evaluated on 51 sequences with accurate bounding box annotations. Two evaluation criteria are used in OTB: precision plot and success plot. Both two criteria show the percentage of successfully tracked frames based on the two metrics CLE and VOR mentioned above under different thresholds. In precision plot, the tracking result in a frame is considered successful if the CLE falls below a pre-defined threshold. In success plot, the target in a frame is declared to be successfully tracked if the current VOR exceeds a certain threshold. Two ranking metrics are used to evaluate all compared trackers: one is the Area Under the Curve (AUC) metric at the success plot, and the other is the precision score at threshold of 20 pixels for the precision plot.

1) Overall Performance: To show the overall performance of the proposed NMC tracker, we compare it with some other state-of-the-art methods on success plots and precision plots. The results in Fig. 4 show that the top 10 trackers on success plot are IMT [40], LNLT [25], SCM [21], SST [22], Struck [42], TLD [45], ASLA [16], CXT [48] and our two methods with HOG feature and raw grayscale feature, respectively. The proposed tracker with HOG feature ranks the first on success plot and precision plot. Besides, the NMC tracker with raw grayscale feature also achieves comparable performance with other state-of-the-art methods.

TABLE I
RESULTS ON THE PRINCETON TRACKING BENCHMARK: SUCCESS RATES AND RANKINGS (IN PARENTHESES) FOR DIFFERENT SEQUENCE CATEGORIZATIONS

Method	Avg. Rank	target type			target size		movement		occlusion		motion type	
		human	animal	rigid	large	small	slow	fast	yes	no	passive	active
Ours(HOG)	1.36(1)	0.46(1)	0.52(1)	0.59(2)	0.52(1)	0.58(1)	0.59(2)	0.49(1)	0.44(1)	0.67(2)	0.62(2)	0.50(1)
KCF [10]	1.63(2)	0.42(2)	0.50(2)	0.65(1)	0.48(2)	0.55(2)	0.65(1)	0.47(2)	0.41(2)	0.68(1)	0.65(1)	0.47(2)
Struck [42]	3.73	0.35(3)	0.47(4)	0.53(5)	0.45(3)	0.44(4)	0.58(3)	0.39(3)	0.30(5)	0.64(3)	0.54(5)	0.41(3)
VTD [43]	4.91	0.31(6)	0.49(3)	0.54(4)	0.39(4)	0.46(5)	0.57(4)	0.37(4)	0.28(7)	0.63(4)	0.55(4)	0.38(4)
RGBdet [38]	4.72	0.27(4)	0.41(6)	0.55(3)	0.32(8)	0.46(3)	0.51(7)	0.36(5)	0.35(3)	0.47(8)	0.56(3)	0.34(5)
MIL [46]	6.09	0.32(8)	0.37(7)	0.38(7)	0.37(6)	0.35(7)	0.46(5)	0.31(6)	0.26(4)	0.49(6)	0.40(7)	0.34(6)
TLD [45]	6.91	0.29(7)	0.35(8)	0.44(6)	0.32(7)	0.38(6)	0.52(8)	0.30(8)	0.34(6)	0.39(5)	0.50(9)	0.31(8)
CT [44]	7.0	0.31(5)	0.47(5)	0.37(8)	0.39(5)	0.34(8)	0.49(6)	0.31(7)	0.23(9)	0.54(7)	0.42(8)	0.34(7)
SemiB [47]	8.64	0.22(9)	0.33(9)	0.33(9)	0.24(9)	0.32(9)	0.38(9)	0.24(9)	0.25(8)	0.33(9)	0.42(6)	0.23(9)
OF [38]	10	0.18(10)	0.11(10)	0.23(10)	0.2(10)	0.17(10)	0.18(10)	0.19(10)	0.16(10)	0.22(10)	0.23(10)	0.17(10)

The best two results are highlighted by green and RED, respectively.

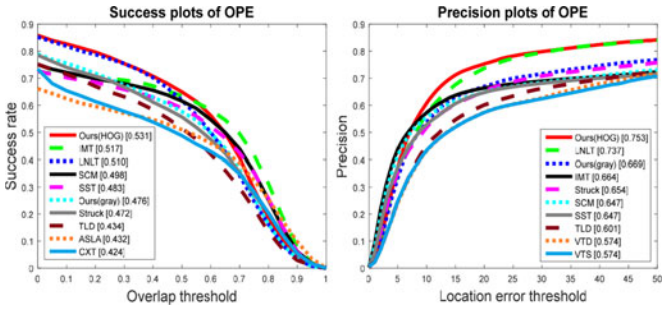


Fig. 4. Success plot and precision plot. The performance score for each tracker is shown in the legend.

Apart from the above methods, a representative deep learning based tracker (DLT) [49] is also incorporated for comparisons and its performance is recorded in Table II. It can be observed that the NMC tracker performs better than DLT with 3.8% improvement on average VOR.

2) Attribute-Based Performance Analysis: The sequences in the OTB dataset are annotated by several challenging factors in visual tracking. To prove the superiority of the proposed algorithm, we plot the success rates of various methods on the subsets with eight main attributes (*Occlusion*, *Motion Blur*, *Deformation*, *Illumination Variation*, *In-plane Rotation*, *Out-of-plane Rotation*, *Background Clutter*, and *Out of View*). The results shown in Fig. 5 illustrate that the proposed NMC tracker with HOG feature ranks first or second on these eight attributes. Specifically, on *Deformation* attribute, our proposed method ranks the first and obtains 11% improvement than the second best tracker on success plot due to an accurate appearance model. Besides, the occlusion detection scheme prevents the proposed tracker from updating the positive templates when the target is heavily occluded. The satisfactory performance on *Occlusion* attribute verifies that the template updating scheme introduced in this paper is effective.

3) Qualitative Analysis: Fig. 6 shows the qualitative results comparing three best baseline trackers on several representative frames. It can be observed that the proposed tracker is robust to the occlusions (*Suv* and *Jogging.2*), effectively grabs the appearance changes (*Basketball*, *Freeman1*, and *David3*), and resists on undesirable illumination variations (*Tiger2*) when compared with other baseline methods in these sequences. Overall, the proposed NMC tracker performs well on these challenging sequences, and it can be attributed to an accurate appearance model learned from an ensemble of multiple local dictionaries, and an effective model updating scheme to avoid degrading the appearance model.

C. Selection of Regularization Parameters

In this subsection, we firstly analyze the superiority of the adopted regularization term over the exact nonnegative constraint on the final tracking performance, and then investigate the influence of different regularization parameters to the tracking results.

Table II compares our NMC tracker with the "Nonnegative" method using the exact nonnegative constraint. Apart from high computational complexity, this setting also yields an inferior performance with 1.1% reduction on success plot than the proposed NMC tracker. In contrast, the introduced ℓ_2 regularization term is able to effectively avoid over-fitting and make the coefficient vector more stable.

Besides, to quantitatively analyze the influence of the two regularization parameters λ and β to the final tracking performance, we provide the corresponding tracking results on OTB under different selections of λ and β. Fig. 7 shows the influence of the tuning parameter λ with different values 0, 0.01, 0.1, 1, 10 on success plot and precision plot. We see that without any regularization term (i.e. $\lambda = \beta = 0$), the success rate and precision are extremely low, and these indicate that the tracker is very fragile. When λ ranges from 0.01 to 10 (β is fixed to 0.1), the success rate and precision steadily increase to the summit at $\lambda = 1$ and then fall down to a low record. If λ is too small (e.g. $\lambda \in [0.01, 0.1]$), we see that the variation of this parameter plays a little role in determining the final tracking performance. If λ is too large, it will lead to over-fitting with a large reconstruction error.

The above experimental results suggest that the fixed values of λ and β will degrade the tracking performance. Therefore, in our method the parameter λ is adaptively selected by (16), which brings about 2.2% improvement when compared with the manually tuned $\lambda = 1$ revealed by the success plot. Therefore the automatic way for setting λ presented in Section II-D is reasonable and effective.

D. Key Component Validation

This subsection validates the effectiveness of several key components in our tracker.

1) Validation of Occlusion Detection Scheme: The only difference between the "NoOcc" method and the proposed NMC

TABLE II
AVERAGE CENTER LOCATION ERRORS (CLE) AND AVERAGE VOC OVERLAP RATIO (VOR) ON OTB WITH 51 TASKS

Method	Ours(Gray)	DLT [49]	Nonnegative	M = 24 × 24	M = 36× 36	$k_1 = 5$	$k_2 = 8$	$k_3 = 10$	$\mathcal{K}_1 = \{3,5,8\}$	$\mathcal{K}_2 = \{8,10,15\}$
average CLE(pixels)	48.5	61.7	55.3	67.2	52.4	66.3	65.0	78.7	65.1	64.5
average VOR(%)	47.6%	43.8%	46.5%	44.2%	47.1%	43.2%	44.8%	42.7%	44.8%	43.4%

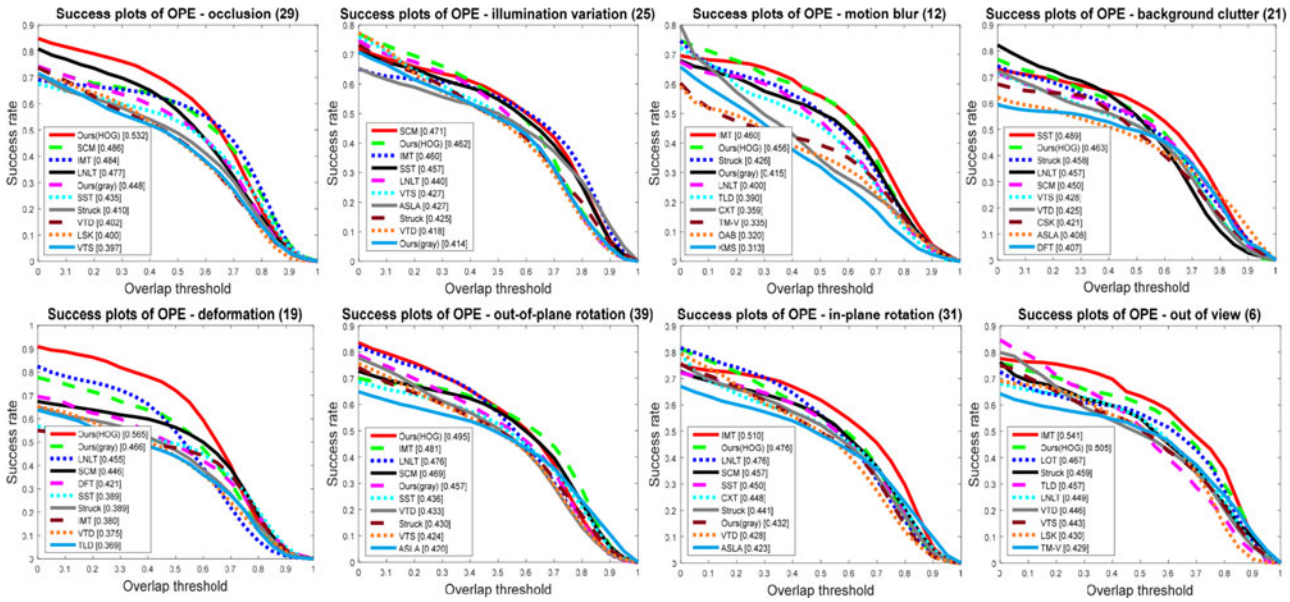


Fig. 5. Curves of success rates of various compared methods on eight main attributes.

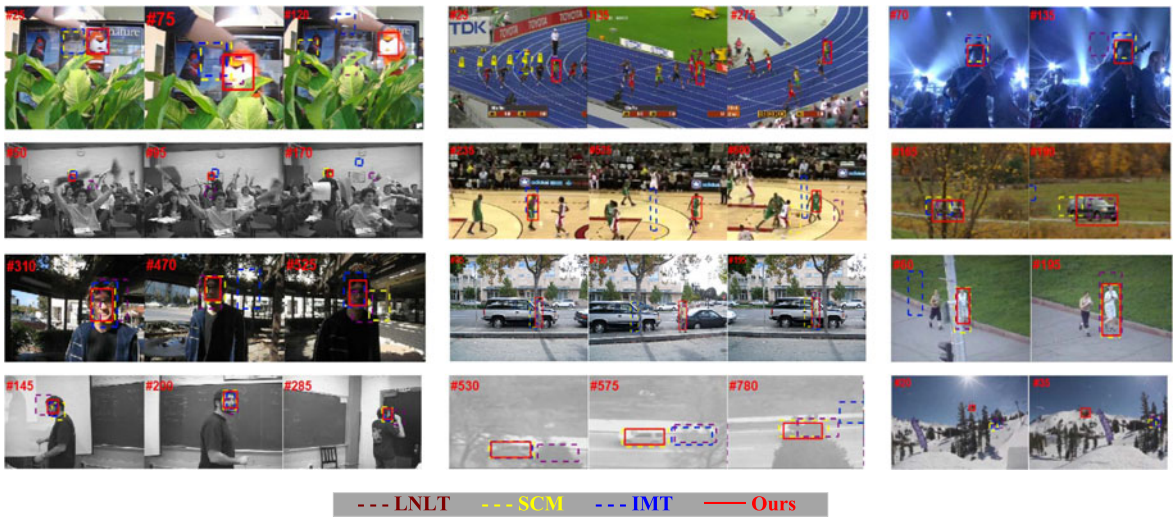


Fig. 6. Comparison of our method with three best baseline trackers on the challenging frames. Subfigures from left to right, top to bottom: *Tiger2*, *Bolt*, and *Shaking*; *Freeman4*, *Basketball*, and *Carscale*; *Trellis*, *David3*, and *Jogging.2*; *Freeman1*, *Suv*, and *Skiing*.

tracker is that the "NoOcc" method does not leverage the aforementioned occlusion detection scheme. Fig. 8 shows that the average overlap rate of the "NoOcc" method decreases dramatically from 47.6% to 43.1%, and the average center location error significantly rises from 48.5 to 70.4. Consequently, we see that the developed occlusion detection scheme plays an important role in boosting the tracking performance.

Furthermore, we also change the range of the number of negative templates $\text{Num}(\text{neg}_*)$ (from 0 to 6) to see its effect on the final tracking performance. In Fig. 8, as the threshold of the number of negative templates increases, the success rate and precision rapidly increase to the summit at $\text{Num}(\text{neg}_*) = 2$, and then steadily fall down to a low record. If this threshold is too small, the updating scheme will fail and does not accommodate

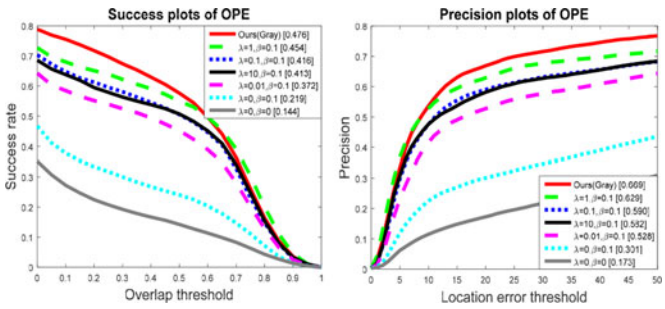


Fig. 7. Success plot and precision plot. The success rate and precision of our proposed tracker under different selections of λ and β.

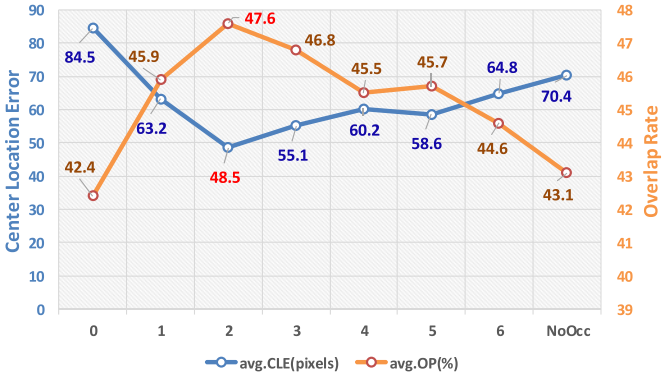


Fig. 8. Tracking performance with varying negative templates threshold. The average CLE and average OP versus the threshold of the number of negative templates [Num(neg_*)] are illustrated.

variations of the target. In contrast, when the threshold is large, the occlusion detection scheme would pale in importance and even cannot prevent the severe occluded target from updating the appearance model.

2) Numbers of Nearest Neighbors: Finding an appropriate k is important because this parameter affects the representation ability of a local dictionary. If k is selected too small, it is prone to noise, leading to degradation of the similarity between candidate samples. A large k might incorporate dissimilar candidates to construct the local dictionary, which would cause error accumulation and degrade the tracking accuracy.

We first verify the effectiveness of the ensemble strategy, and the corresponding results of different numbers of nearest neighbours for dictionary construction are shown in Table II. We can see that the proposed tracker with an ensemble of local dictionaries outperforms all settings with a single local dictionary (e.g. $k_1 = 5, k_2 = 8$, and $k_3 = 10$), therefore our ensemble strategy can fully exploit the advantage of each local dictionary in the ensemble, and this is the key reason for our tracker to obtain very impressive tracking results.

In addition, we also compare the results corresponding to our selected numbers of neighbors (i.e. $\mathcal{K} = \{5, 8, 10\}$) with other numbers of neighbors including $\mathcal{K}_1 = \{3, 5, 8\}$ and $\mathcal{K}_2 = \{8, 10, 15\}$. From the results shown in the last two columns in Table II, we see that an ensemble with small k-nearest neighbors (e.g. $\mathcal{K}_1 = \{3, 5, 8\}$) can hardly tackle noise and is not robust to challenging factors. The ensemble with large k-nearest neighbors (e.g. $\mathcal{K}_2 = \{8, 10, 15\}$) models the target's appearance

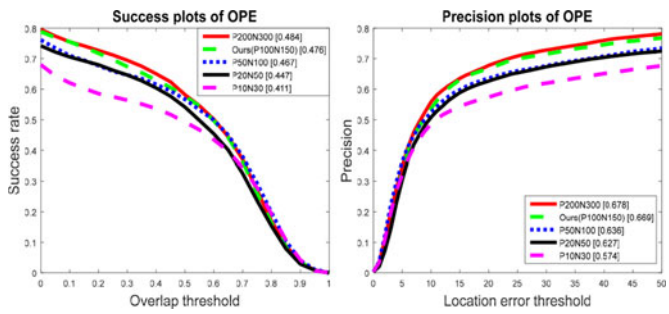


Fig. 9. Success plot and precision plot. The success rate and precision of our proposed tracker under different numbers of positive and negative templates.

inaccurately, leading to an unreliable tracking performance. Accordingly, a suitable ensemble should take diverse k-nearest neighbors into consideration to fully exploit their individual advantages, such as $\mathcal{K} = \{5, 8, 10\}$. Table II shows that the NMC tracker with the selection of $\mathcal{K} = \{5, 8, 10\}$ achieves the highest record on average VOR among all the compared parametric settings, which suggests that the selected $\mathcal{K} = \{5, 8, 10\}$ in this paper is effective.

3) Numbers of Positive and Negative Templates: We investigate how the numbers of positive and negative templates sampled at the beginning of a video sequence affect the final tracking performance. The corresponding success rate and precision are shown in Fig. 9, in which "P200N300" denotes that the number of positive and negative templates p and n are set to be 200 and 300, respectively. It can be observed that the insufficient number of positive templates (e.g., "P10N30") would incur unsatisfactory performance with a low record, namely 41.1% on success rate and 57.4% on precision. This is because that eight templates are selected from only ten positive templates in this case, and thus the output coefficient vector by such setting does not achieve sparse effect.

By contrast, numerous positive templates (e.g., "P200N300", "P50N100") sampled at the beginning frames do not have significant effects on the final tracking performance. Specifically, we can see that the "P200N300" setting achieves 48.4% on average overlap rate, which is slightly higher than our method because much more information has been taken into consideration. However, because superabundant templates are involved in the "P200N300" setting, it arrives at 0.9fps (frame per second), which is twice slower than our NMC tracker.

4) Size of the Image Patch: In our experiments, the image patch is normalized to $M = 32 \times 32$. Here we analyze the influence of different feature dimensions (i.e., the size of an image patch) on the final tracking performance. Table II shows the results of two settings with different sizes, where the "$M = 36 \times 36$" method leads to slight fluctuation on the average CLE and VOR. In addition, the performance of "$M = 24 \times 24$" method dramatically decreases from 47.6% to 44.2% on average VOR due to its insufficient feature representation.

E. Computational Complexity

We present the computational complexity analysis of the proposed algorithm when compared to the existing IVT and ℓ_1

tracker. Assume that the size of the base matrix is $M \times k$, and the iteration number is η, the computation in the IVT method involves matrix-vector multiplication and the complexity is $\mathcal{O}(Mk)$. The computational complexity of the ℓ_1 tracker for computing the sparse coefficients using the Lasso algorithm [50] is $\mathcal{O}(M^2 + Mk)$. For our method, the base matrix is obtained from k-nearest neighbors, and the computational complexity for each candidate is $\mathcal{O}(k^2)$. Regarding finding the closed-form solution for each candidate in (13), the complexity of matrix-vector multiplication is also $\mathcal{O}(Mk)$, and inverting a $k \times k$ matrix is $\mathcal{O}(k^3)$. Thus the complexity of total operation is $\mathcal{O}(k^3 + Mk)$. Although the complexity is mathematically cubic to k, the k in our tracker is set to very small values such as 5, 8 and 10, so the practical complexity is not as high as theoretically suggested.

V. Conclusion

In this paper, we have proposed a novel Nonnegative Multiple Coding (NMC) tracker that uses an ensemble of dictionaries to accurately character the target appearance. In addition, the nonnegative constraint on the coefficient vector is replaced by an efficient ℓ_2 regularization term to achieve the similar nonnegative effect. Such approximation has been demonstrated by the theoretical analysis and experimental results. Thereby, the obtained coefficient vector and the nonnegative constraint have greatly enhanced the representation ability for appearance modeling of the target, which effectively improves the tracking performance. The experimental results on two benchmarks have verified the rationality of such ℓ_2 norm regularization approximation and the ensemble strategy, and also demonstrated that the proposed NMC tracker achieves a favorable performance over the state-of-the-art trackers.

Appendix A
Proof of Theorem 1

This section provides a proof of Theorem 1.

Proof: Let $\mathbf{A}^\top\mathbf{A} = \mathbf{U}^\top\Lambda\mathbf{U}$, where $\mathbf{U}$ is an orthogonal matrix. The diagonal matrix Λ is defined as $\Lambda = \mathrm{diag}(\mu_1, \mu_2, \cdots, \mu_k)$, of which the elements are eigenvalues of $\mathbf{A}^\top\mathbf{A}$. Then $\mathbf{F}$ and $\mathbf{F}^{-1}$ can be reformulated as

$$\begin{cases} \mathbf{F} = \mathbf{A}^\top\mathbf{A} + \alpha\mathbf{I} = \mathbf{U}^\top(\Lambda + \alpha\mathbf{I})\mathbf{U} = \sum_{i=1}^{k}(\mu_i + \alpha)\mathbf{u}^{(i)\top}\mathbf{u}^{(i)} \\ \mathbf{F}^{-1} = (\mathbf{A}^\top\mathbf{A} + \alpha\mathbf{I})^{-1} = \mathbf{U}^\top(\Lambda + \alpha\mathbf{I})^{-1}\mathbf{U} = \sum_{i=1}^{k}\frac{\mathbf{u}^{(i)\top}\mathbf{u}^{(i)}}{\mu_i + \alpha} \end{cases} \tag{21}$$

where $\mathbf{u}^{(i)} \in \mathbb{R}^{1\times k}$ is the ith row of the orthogonal matrix $\mathbf{U}$, and $(\Lambda + \alpha\mathbf{I})^{-1}$ is a diagonal matrix of which the ith diagonal element is $\frac{1}{\mu_i + \alpha}$. As a result, the coefficient vector $\mathbf{x}$ can be rewritten as

$$\mathbf{x} = \mathbf{F}^{-1}\mathbf{1} = \sum_{i=1}^{k}\frac{1}{\mu_i + \alpha}\mathbf{u}^{(i)\top}\mathbf{u}^{(i)}\mathbf{1}. \tag{22}$$

Note that $\mathbf{F}^{-1}\mathbf{F} = \mathbf{I}$ and $\mathbf{u}^{(i)}\mathbf{u}^{(i)\top} = 1$, we have

$$\sum_{i=1}^{k}\mathbf{u}^{(i)\top}\mathbf{u}^{(i)}\mathbf{1} = \mathbf{u}^{(1)\top}\mathbf{u}^{(1)}\mathbf{1} + \cdots + \mathbf{u}^{(k)\top}\mathbf{u}^{(k)}\mathbf{1} = \mathbf{1}. \tag{23}$$

Hence, the ith element of $\mathbf{x}$ can be analytically expressed by

$$x_i = \sum_{r=1}^{k}\frac{1}{\mu_r + \alpha}u_{ri}\sum_{j=1}^{k}u_{rj}, \text{where} \sum_{r=1}^{k}u_{ri}\sum_{j=1}^{k}u_{rj} = 1. \tag{24}$$

Consequently, our target is to prove that x_i in (24) shown bottom of the perious page is not less than 0 when α satisfies the condition in (16). Without loss of generality, we assume that the first ν elements in (24) are smaller than zero, that is $u_{qi}\sum_{j=1}^{k}u_{qj} < 0$ for every $q = 1, 2, \cdots, \nu$. The last $k - \nu$ elements are nonnegative, namely $u_{qi}\sum_{j=1}^{k}u_{qj} \geq 0$ for $q = \nu+1, \nu+2, \cdots, k$. Therefore $x_i \geq 0$ is reformulated to (25), as shown at the bottom of the page, where $\mathcal{P} = \sum_{r=\nu+1}^{k}\frac{1}{\mu_r+\alpha}u_{ri}\sum_{j=1}^{k}u_{rj}$ and $\mathcal{Q} = \sum_{r=1}^{\nu}\frac{-1}{\mu_r+\alpha}u_{ri}\sum_{j=1}^{k}u_{rj}$.

Furthermore, by defining that $\mu_s = \min\{\mu_1, \mu_2, \cdots, \mu_\nu\}$ and $\mu_l = \max\{\mu_{\nu+1}, \mu_{\nu+2}, \cdots, \mu_k\}$, $\mathcal{P}$ and $\mathcal{Q}$ can be bounded by

$$\begin{cases} \mathcal{P} \geq \frac{1}{\mu_l+\alpha}\underbrace{\left(u_{(\nu+1)i}\sum_{j=1}^{k}u_{(\nu+1)j} + \cdots + u_{ki}\sum_{j=1}^{k}u_{kj}\right)}_{\triangleq\mathcal{P}_1} \\ \mathcal{Q} \leq \frac{1}{\mu_s+\alpha}\underbrace{\left(-u_{1i}\sum_{j=1}^{k}u_{1j} - \cdots - u_{\nu i}\sum_{j=1}^{k}u_{\nu j}\right)}_{\triangleq\mathcal{Q}_1} \end{cases} \tag{26}$$

where $\mathcal{P}_1 = \sum_{r=\nu+1}^{k}u_{ri}\sum_{j=1}^{k}u_{rj} \geq 0$, $\mathcal{Q}_1 = -\sum_{r=1}^{\nu}u_{ri}\sum_{j=1}^{k}u_{rj} \geq 0$, and we can easily obtain $\mathcal{P}_1 - \mathcal{Q}_1 = 1$ from (24). As a result

$$\mathcal{P} \geq \frac{1}{\mu_l + \alpha}\mathcal{P}_1 \geq \frac{1}{\mu_s + \alpha}\mathcal{Q}_1 \geq \mathcal{Q} \tag{27}$$

$$\underbrace{\frac{-1}{\mu_1 + \alpha}u_{1i}\sum_{j=1}^{k}u_{1j} + \cdots + \frac{-1}{\mu_\nu + \alpha}u_{\nu i}\sum_{j=1}^{k}u_{\nu j}}_{\triangleq\mathcal{Q}}$$

$$\leq \underbrace{\frac{1}{\mu_{\nu+1} + \alpha}u_{(\nu+1)i}\sum_{j=1}^{k}u_{(\nu+1)j} + \frac{1}{\mu_{\nu+2} + \alpha}u_{(\nu+2)i}\sum_{j=1}^{k}u_{(\nu+2)j} + \cdots + \frac{1}{\mu_k + \alpha}u_{ki}\sum_{j=1}^{k}u_{kj}}_{\triangleq\mathcal{P}} \tag{25}$$

so the tuning parameter α is bounded as follows:

$$\alpha \geq -\mu_s \mathcal{P}_1 + \mu_l \mathcal{Q}_1. \tag{28}$$

Furthermore, we divide the proof into two cases.

Case 1: $\mu_s \geq \mu_l$, $-\mu_s \mathcal{P}_1 + \mu_l \mathcal{Q}_1 \leq -\mu_l \mathcal{P}_1 + \mu_l \mathcal{Q}_1 = -\mu_l$.

In this case, the lower bound of α can be chosen as $\alpha \geq -\mu_l$, which is always satisfied due to $\alpha \geq 0$.

Case 2: $\mu_s < \mu_l$

$$\begin{aligned} -\mu_s \mathcal{P}_1 + \mu_l \mathcal{Q}_1 &= -\mu_s + (\mu_l - \mu_s)\mathcal{Q}_1 \\ &\leq -\mu_s + (\mu_l - \mu_s)(\mathcal{Q}_1 + \mathcal{P}_1) \\ &\leq -\mu_s + (\mu_l - \mu_s)k. \end{aligned} \tag{29}$$

In (29), by utilizing the inequalities of arithmetic and geometric means [33], we have $\mathcal{P}_1 \leq \sqrt{k}\sum_{j=\nu+1}^{k} u_{ji}$, and $\mathcal{Q}_1 \leq \sqrt{k}\sum_{j=1}^{\nu} u_{ji}$, which leads to $(\mathcal{P}_1 + \mathcal{Q}_1) \leq k$. Thus the lower bound of α is chosen as

$$\alpha \geq -\mu_s + (\mu_l - \mu_s)k. \tag{30}$$

Furthermore, we discuss the range of eigenvalues μ_s and μ_l to obtain a more elegant bound of α. Based on the Gerschgorin's theorem [34], the bounds of μ_s and μ_l satisfy

$$\begin{cases} |\mu_s - (\mathbf{A}^\top \mathbf{A})_{ss}| \leq \sum_{j\neq s}^{k} |(\mathbf{A}^\top \mathbf{A})_{sj}| \\ |\mu_l - (\mathbf{A}^\top \mathbf{A})_{ll}| \leq \sum_{j\neq l}^{k} |(\mathbf{A}^\top \mathbf{A})_{lj}| \end{cases}. \tag{31}$$

By plugging (31) into (30), the right-hand side of (30) is transformed to

$$\begin{aligned} &-\mu_s + (\mu_l - \mu_s)k \leq \sum_{j\neq s}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{sj}\right| - (\mathbf{A}^\top \mathbf{A})_{ss} + k(\mathbf{A}^\top \mathbf{A})_{ll} \\ &\quad - k(\mathbf{A}^\top \mathbf{A})_{ss} + k\sum_{j\neq s}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{sj}\right| + k\sum_{j\neq l}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{lj}\right| \\ &= (k+1)\left\{\sum_{j\neq s}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{sj}\right| - (\mathbf{A}^\top \mathbf{A})_{ss}\right\} + k\sum_{j=1}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{lj}\right| \\ &= (k+1)\left\{\sum_{j=1}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{sj}\right| - 2(\mathbf{A}^\top \mathbf{A})_{ss}\right\} + k\sum_{j=1}^{k} \left|(\mathbf{A}^\top \mathbf{A})_{lj}\right| \\ &\leq (k+1)\left\|\mathbf{A}^\top \mathbf{A} - 2\Gamma\right\|_\infty + k\left\|\mathbf{A}^\top \mathbf{A}\right\|_\infty \end{aligned} \tag{32}$$

which concludes the proof. ■

ACKNOWLEDGEMENT

The authors would like to thank Prof. X. Huang and Dr. M. Zareapoor from Shanghai Jiao Tong University for their proofreading and the anonymous reviewers for their insightful comments.

REFERENCES

[1] Y. Wu, J. Lim, and M. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[2] N. Wang, J. Shi, D. Yeung, and J. Jia, "Understanding and diagnosing visual tracking systems," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3101–3109.

[3] Y. Yuan, H. Yang, Y. Fang, and W. Lin, "Visual object tracking by structure complexity coefficients," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1125–1136, Aug. 2015.

[4] J. Zhang, S. Ma, and S. Sclaroff, "MEEM: Robust tracking via multiple experts using entropy minimization," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 188–203.

[5] C. Gong *et al.*, "PageRank tracker: From ranking to tracking," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 882–893, Jun. 2014.

[6] D. Wang, H. Lu, and M. Yang, "Online object tracking with sparse prototypes," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 314–325, Jan. 2013.

[7] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with gaussian process regression," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 188–203.

[8] D. Chen, Z. Yuan, G. Hua, Y. Wu, and N. Zheng, "Description-discrimination collaborative tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 345–360.

[9] S. Zhang, X. Yu, Y. Sui, S. Zhao, and L. Zhang, "Object tracking with multi-view support vector machines," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 265–278, Mar. 2015.

[10] J. 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, Mar. 2015.

[11] C. Gong *et al.*, "Multi-modal curriculum learning for semi-supervised image classification," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3249–3260, Jul. 2016.

[12] C. Gong, D. Tao, W. Liu, L. Liu, and J. Yang, "Label propagation via teaching-to-learn and learning-to-teach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1452–1465, Jun. 2016.

[13] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2259–2272, Nov. 2011.

[14] T. Zhang, A. Bibi, and B. Ghanem, "In defense of sparse tracking: Circulant sparse tracker," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2016, pp. 3880–3888.

[15] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust ℓ_1 tracker using accelerated proximal gradient approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 1830–1837.

[16] X. Jia, H. Lu, and M. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 1822–1829.

[17] Q. Wang, F. Chen, W. Xu, and M. Yang, "Online discriminative object tracking with local sparse representation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. Workshops*, Jun. 2012, pp. 425–432.

[18] B. Ma *et al.*, "Visual tracking using strong classifier and structural local sparse descriptors," *IEEE Trans. Multimedia*, vol. 17, no. 10, pp. 1818–1828, Oct. 2015.

[19] Z. Hong, X. Mei, D. Prokhorov, and D. Tao, "Tracking via robust multi-task multi-view joint sparse representation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 649–656.

[20] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via structured multi-task sparse learning," *Int. J. Comput. Vis.*, vol. 101, no. 2, pp. 367–383, 2013.

[21] W. Zhong, H. Lu, and M. Yang, "Robust object tracking via sparse collaborative appearance model," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2356–2368, May 2014.

[22] T. Zhang *et al.*, "Structural sparse tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2015, pp. 150–158.

[23] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 2223–2231.

[24] J. Wang *et al.*, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 3360–3367.

[25] B. Ma, H. Hu, J. Shen, Y. Zhang, and F. Porikli, "Linearization to nonlinear learning for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4400–4407.

[26] B. Liu, J. Huang, C. Kulikowski, and L. Yang, "Robust visual tracking using local sparse appearance model and k-selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2968–2981, Dec. 2013.

[27] F. Liu, T. Zhou, K. Fu, and J. Yang, "Kernelized temporal locality learning for real-time visual tracking," *Pattern Recog. Lett.*, vol. 90, pp. 72–79, 2017.

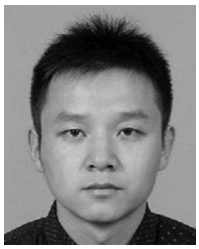
[28] G. Wang *et al.*, "Visual tracking via sparse and local linear coding," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3796–3809, Nov. 2015.
[29] F. Liu, T. Zhou, J. Yang, and I. Gu, "Visual tracking via nonnegative regularization multiple locality coding," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2015, pp. 72–80.
[30] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 556–562.
[31] Y. Wu, B. Shen, and H. Ling, "Visual tracking via online nonnegative matrix factorization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 374–383, Mar. 2014.
[32] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Opt.*, vol. 1, no. 3, pp. 123–231, 2013.
[33] B. Long and Y. Chu, "Optimal inequalities for generalized logarithmic, arithmetic, and geometric means," *J. Inequalities Appl.*, vol. 2010, no. 1, pp. 1, 2010.
[34] C. Carstensen, "Inclusion of the roots of a polynomial based on gerschgorin's theorem," *Numer. Math.*, vol. 59, no. 1, pp. 349–360, 1991.
[35] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient ℓ_1 tracker with occlusion detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 1257–1264.
[36] D. Ross, J. Lim, R. Lin, and M. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, pp. 125–141, 2008.
[37] D. Wang, H. Lu, and M. Yang, "Least soft-threshold squares tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2371–2378.
[38] S. Song and J. Xiao, "Tracking revisited using RGBD camera: Unified benchmark and baselines," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 233–240.
[39] Y. Wu, J. Lim, and M. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2411–2418.
[40] J. Yoon, M. Yang, and K. Yoon, "Interacting multiview tracker," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 5, pp. 903–917, May 2016.
[41] M. Everingham, L. Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[42] S. Hare, A. Saffari, and P. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 263–270.
[43] J. Kwon and K. Lee, "Visual tracking decomposition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun.2010, pp. 1269–1276.
[44] K. Zhang, L. Zhang, and M. Yang, "Fast compressive tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2002–2015, Oct. 2014.
[45] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.
[46] B. Babenko, M. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.
[47] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised boosting online boosting for robust tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 234–247.
[48] T. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 1177–1184.
[49] N. Wang and D. Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 809–817.
[50] M. Park and T. Hastie, "ℓ_1 regularization path algorithm for generalized linear models," *J. Roy. Stat. Soc.*, vol. 69, no. 4, pp. 659–677, 2007.

Fanghui Liu received the B.S. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2014, and is currently working toward the Ph.D. degree at the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, under the supervision of Prof. J. Yang.

His research areas mainly include computer vision and machine learning with respect to visual tracking, kernel learning, and Bayesian learning.

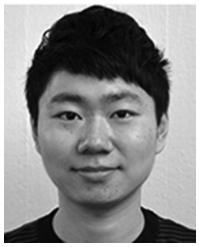
Chen Gong (M'17) received the dual Ph.D. degree from Shanghai Jiao Tong University, Shanghai. China, and the University of Technology Sydney, Sydney, Australia, in 2016, under the supervision of Prof. J. Yang and Prof. D. Tao, respectively.

He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. He has authored or coauthored more than 30 technical papers in journals and conferences such as the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CYBERNETICS, the Conference on Computer Vision and Pattern Recognition, the Association for the Advancement of Artificial Intelligence, and the International Joint Conference on Artificial Intelligence. His current research interests include machine learning, data mining, and learning-based vision problems.



Tao Zhou received the M.S. degree in computer application technology from Jiangnan University, Wuxi, China, in 2012, and the Ph.D. degree in pattern recognition and intelligent system from the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, in 2016.

His current research interests include object detection, visual tracking, and machine learning.



Keren Fu received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2011, and dual Ph.D. degrees from Shanghai Jiao Tong University, Shanghai, China, and Chalmers University of Technology, Gothenburg, Sweden, in 2016, under the joint supervision of Prof. J. Yang and Prof. I. Y.-H. Gu.

He is currently with the College of Computer Science, Sichuan University, Chengdu, China. His current research interests include visual computing, saliency analysis, and machine learning.

Xiangjian He (M'99–SM'05) received the Ph.D. degree in computer science from the University of Technology Sydney (UTS), Ultimo, Australia, in 1999.

He is currently a Full Professor and the Director of the Computer Vision and Pattern Recognition Laboratory, Global Big Data Technologies Centre, Ultimo, NSW, Australia, and a Co-Leader of the Network Security Research Team with the Centre for Real-Time Information Networks, UTS.

Prof. He is a Member of the IEEE Signal Processing Society Student Committee. He was a Guest Editor for various international journals. He has many high-quality publications and was the recipient of various research grants, including four National Research Grants by the Australian Research Council as a Chief Investigator.

Jie Yang received the Ph.D. degree from the Department of Computer Science, Hamburg University, Hamburg, Germany, in 1994.

He is currently a Professor with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China. He has led many research projects (e.g., National Science Foundation, 863 National High Tech. Plan), had one book published in Germany, and authored or coauthored more than 300 journal papers. His major research interests are object detection and recognition, data fusion and data mining, and medical image processing.