IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 5, 2019, accepted March 18, 2019, date of publication March 20, 2019, date of current version April 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2906508

Learning Local Structured Correlation Filters for Visual Tracking via Spatial Joint Regularization

CHENGGANG GUO, DONGYI CHEN, AND ZHIQI HUANG
School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Corresponding author: Dongyi Chen (dychen@uestc.edu.cn)

This work was supported in part by the National Key R&D Program of China under Grant 2017YFB1002803, and in part by the National Natural Science Foundation of China under Grant 61572110.

ABSTRACT Robust visual tracking is a fundamental problem in the field of computer vision and has a wide range of practical applications. Recent progress in developing robust tracking methods are mainly made upon discriminative correlation filters (DCF). However, most DCF-based methods develop their trackers under the assumption of a holistic appearance model, ignoring the underlying spatial local structural information. In this paper, we introduce the tree-structured group sparsity regularization into the DCF-based formula. The correlation filter to be learned is divided into hierarchical local groups. The relationship between the response and the circularly shifted target appearance is regularized by applying the l_1-norm across the l_2-norm of the hierarchical local filter groups. Moreover, a local response consistency term is incorporated together with the structured sparsity to make each local filter group contributes equally to the final response. The accelerated proximal gradient method is employed to optimize this non-smooth composite regularization problem. Benefiting from the properties of circulant matrices, several key steps in the optimization process can be efficiently solved in the frequency domain. The experiments are conducted on four publicly available visual tracking benchmarks. Both quantitative and qualitative evaluations demonstrate that the proposed tracking method performs favorably against a number of state-of-the-art tracking methods.

INDEX TERMS Correlation filter, spatial regularization, structure sparsity, visual tracking.

I. INTRODUCTION

Robust visual tracking is a fundamental and challenging topic in the field of computer vision. It is the essential technique for a variety of applications, including vehicle navigation, security surveillance and augmented reality. In general object tracking, a visual tracker only acquires the initial position of the target object at the first frame of a video sequence, and then expects to robustly estimate the motion trajectory of this object. This problem is very challenging since the tracker can only learns from a limited set of reliable training samples in the first frame. Typically, trackers need to update themselves with unreliable online collected appearance samples to generalize models against time-varying noises, such as object deformations, occlusions, and appearance variations.

In recent years, discriminative correlation filters (DCFs) based tracking approaches have gain an increasing attention and continuing advances. Inspired by the initial success of applying correlation filters in visual tracking [1], a lot of improved DCF-based approaches have been proposed. The core idea of DCFs exploits the 2D extension of convolution theorem that the circular convolution of two feature maps is equivalent to the element-wise product in the Fourier domain. Thus, the correlation filter learning problem can be efficiently solved in the frequency domain through fast Fourier transform (FFT) operations. However, the underlying periodic assumption inside the circular convolution operation introduces undesired boundary effects. Existing DCF-based approaches mainly address this problem in two ways. Firstly, a spatial regularization strategy is introduced in [2]–[4] to suppress the boundary effects. Correlation coefficients corresponding to the background are largely penalized to a small value, which increases the discriminative power of the correlation model. But the coefficients corresponding to the target region are treated as a whole and local information is not explored. Secondly, a binary cropping operator is proposed in [5] to extract real negative samples of the target

The associate editor coordinating the review of this manuscript and approving it for publication was Naveed Akhtar.

size, instead of suppressing the circularly shifted background pixels on the entire feature map. It further enhances the quality of the learned model. However, the complex optimization process for this method limits further improvements on spatial local regularization of correlation filters.

In this paper, we follow the research line of using spatial weighting and present a spatial tree-structured joint regularization for visual tracking. In contrary to the existing methods that treat the spatial regularized filter as a whole, we consider exploiting local information inside the target region. As discussed in [6]–[9], it is insufficient for describing real world object simply using a single rigid model. Features grouped in local regions are generally more temporal stable to achieve a robust tracking. Furtherly, we introduce a tree-structured group sparse constraint to do adaptive hierarchical feature group selection for correlation template learning. The similar region-level feature selection idea for correlation tracking is also proposed in [10], where the feature selection strategy is achieved by imposing an elastic net regularization. But with a tree-structured group sparsity regularization [11], we are able to uncover a spatial structured information inside the target region. A hierarchical relationship between adjacent filter parts can be adaptively explored during the optimization. Moreover, a local response consistency constraint term [12] is incorporated together with the tree-structured regularization to make sure the filter would not over fit on the dominant response parts. To optimize this non-smooth composite regularization problem, we also present an efficient optimization algorithm based on the accelerated proximal gradient (APG) [13]. In brief, the contributions of this work can be summarized in three-fold:

Firstly, a spatial tree-structured joint regularization is introduced into the correlation filter framework. Since the structured sparsity regularization and local response consistency regularization are emphasized simultaneously, the dependency between local parts inside the target region are characterized. Based on this joint regularization, we propose to use local structured discriminative correlation filters (LSDCF) for visual tracking.

Secondly, the accelerated proximal gradient (APG) method is utilized to solve the proposed joint regularized problem. In optimizing the problem using APG, we decompose the non-smooth composite objective function into a smooth convex part and a non-smooth convex part. We calculate the sub-differential of the former part, and then connects the generalized gradient update step in the optimization process to the Moreau-Yosida regularization with the predefined tree structure that has an analytical solution [11]. Thus the convergence of the proposed algorithm can be ensured.

Thirdly, a robust visual tracking method based on the local structured correlation filters is presented with promising tracking performance. Experiments evaluated on recent tracking benchmarks show that the proposed method has competitive performance than existing DCF-based methods in terms of spatial regularization.

II. RELATED WORK

This section begins with a brief review of the DCF-based trackers proposed in recent years. Related improvements on spatial regularization and part-based trackers are discussed. Then other model based trackers, including distance metric learning, sparse coding, are also briefly reviewed. Since the proposed model utilizes a structured sparsity regularization, some classical structured sparsity algorithms are introduced.

A. DCF MODEL BASED TRACKERS

According to the competition results of the visual tracking challenges over the past five years [14], [15], discriminative correlation filters (DCFs) have become the most popular visual tracking framework and more than half of the top ranking trackers are improved DCF-based trackers. Learning with adaptive tracking using DCFs initially starts with the MOSSE tracker [1], which minimizes the sum of squared error between the output of convolution and the desired Gaussian shaped output. Based on the theory of circulant matrices, Henriques *et al.* [16] first proposed a kernelized correlation filter and corresponding closed-form solutions, and then extended their work to handle multi-channel features, such as histogram of oriented gradients (HOG) [17], and named KCF in [18]. Ma *et al.* [19] considered the hierarchical feature representation power of CNNs and proposed to learn adaptive correlation filters in a coarse-to-fine fashion for tracking. Despite the intensity-based tracker, color representations have also been considered. Danelljan *et al.* [20] validated the performance of different color features and demonstrated color names to be the best color representations. For the problem of scale changes during tracking, several scale adaptive methods [21], [22] have been developed. An explicit scale filter was proposed in [21] to train samples at a set of different scales. Li *et al.* [22] made the first attempt to handle the aspect ratio variation during tracking by integrating boundary and center correlation filters into a regularization term. To enable integration of multi-resolution feature maps, Qi *et al.* [23] proposed to hedge multiple weak trackers into a strong one for exploiting deep features from different CNN layers. Danelljan *et al.* [3] proposed to apply continuous convolution in learning correlation filters. Furtherly they proposed a factorized convolution operator [4] to reduce model parameters.

Following the discussion before, the boundary effects introduced by the periodic assumption greatly affect the discriminative power of the learned correlation filters. A spatial regularization for discriminative correlation filters (SRDCF) was proposed in [2] to suppress the boundary effect appeared in the implicitly generated negative samples. The background coefficients are largely penalized, so the model can focus on learning information from the center region of each negative sample. The method proposed in [5] inherits the idea of [24] that the target region of each circularly shifted negative samples is directly cropped through a masking matrix. With this operation, most implicitly generated negative samples are immune to boundary effects. Our method, on the other

hand, applies a masking matrix on the circularly shifted samples without cropping, which is equivalent to binary spatial weighting. In this way, the masking matrix in the objective function are more intuitive and convenient for exploring local information.

Since a single rigid appearance model is not sufficient to describe object deformations or occlusions, researchers have proposed a number of part-based trackers in recent years. Johnander *et al.* [6] proposed to learn a deformable filter represented as a linear combination of sub-filters. The sub-filter coefficients and their relative locations are jointly optimized in a unified framework. Liu *et al.* [7] proposed to independently use correlation filters as part classifiers and adaptively weighting each part response map using a smooth constraint. Li *et al.* [8] selected reliable patches under a sequential Monte Carlo framework and then estimated the target location through a Hough Voting-like scheme. A fully connected deformable local correlation filters was proposed in [9] to explicitly address the non-rigid deformations and occlusions in visual tracking scenarios. Different from the existing part-based trackers that explicitly model the relationship between parts, we aim to adaptively explore hierarchical local structure information inside the holistic correlation filter model, which can be regarded as a generalization of the idea in [12]. In [12], a 3×3 grid of local regions are divided and reliable weights for these regions are explicitly modeled. However, the third term in their objective is a squared l_2 regularization term that imposed on the foreground region as a whole. It is distinctly different from the ideas to be presented in our method.

B. OTHER MODEL BASED TRACKERS

Besides developing DCF based models, researchers in the visual tracking community have proposed many tracking methods based on other theories. Zhang *et al.* [25] proposed to integrate the image-to-imageSet distance metric learning into visual tracking to take full advantage of all training samples. Wu *et al.* [26] proposed an online multiple instance metric learning algorithm that learns a discriminative and adaptive metric. Zhang *et al.* [27] proposed an interestingly biologically inspired appearance model that is motivated in part by the success of the hierarchical organization of the primary visual cortex.

Mei and Ling [28] first introduced sparse representation theory into the field of visual tracking. Multi-task sparse learning problem, such as [29]–[31], are formulated to learn the sparse representations of all candidate samples jointly. In the work of [32], the proposed structural sparse appearance model emphasis on the spatial layout structure among local patches inside each target candidate region. Similar ideas are proposed in [33] that exploits the integration of spatial context information into a unified sparse framework. Zhang *et al.* [34] proposed to relax the sparsity constraint using a weighted least squares method to handle appearance variations and use structurally random projection to reduce computation complexity.

C. STRUCTURED SPARSITY ALGORITHMS

In many practical cases, prior knowledge of data structures can be beneficial for solving interpretable sparse variables and achieving better performance. Based on this observation, l_1 regularization (or Lasso) [35] have been extended to the group Lasso [36], [37] that a group structured dependency exists among sparse coefficients. Since the group lasso penalty only produces sparsity at the group level, Friedman *et al.* [38] proposed a sparse group lasso penalty that yields sparse solutions both at group level and individual variable level. Simon *et al.* [39] adopt the sparse group lasso penalty in a regression model and proposed an block-wise descent optimization algorithm based on the accelerated generalized gradient descent. Nevertheless, the group lasso and sparse group lasso are restricted to the non-overlapping groups of features, which is not flexible for some applications. Thus several attempts have focused on developing group lasso with potential overlaps [40], [41]. The composite absolute penalties (CAP) family is introduced by Zhao *et al.* [42], which allows given grouping and hierarchical relationships between the variables to be expressed. After observing the limitation of traditional lasso, Zhang *et al.* [43] proposed a discriminative lasso to select features that are strongly correlated with the response and less correlated with each other.

The tree structured group lasso is considered recently to encode variables at multiple granularity. Specifically, a multi-level grouping structure is encoded as a tree over the variables, where each leaf node represents an individual variable and internal node represents the cluster of a certain leaf nodes. Kim and Xing [44] presented the tree structured group lasso for learning sparse multi-task regression. Liu and Ye [11] developed an analytical solution for the Moreau-Yosida regularization associated with the grouped tree structure. The tree structure is intuitively suitable for image analysis that features from hierarchical spatial locality can form a 2D spatial tree structure. Luo *et al.* [45] presented a tree-structured nuclear norm approximation for robust face recognition. Li *et al.* [46] utilized the tree-structured group joint sparse representation to combine multi-level cues for image illumination estimation. In this paper, we present a local structured object tracker based on the spatial tree structured joint regularization for the first time.

III. PROPOSED METHOD

In this section, we first briefly review the SRDCF model [2], and then describe the proposed local structured discriminative correlation filters (LSDCF) in detail. Finally, an optimization algorithm based on the accelerated proximal gradient method is derived.

A. REVISITING SPATIAL REGULARIZED DCF

Let $\mathbf{y} \in \mathbb{R}^{K\times 1}$ denote the Gaussian shaped response, and $\mathbf{x} = \left[\mathbf{x}_1^\mathsf{T}, \mathbf{x}_2^\mathsf{T}, \cdots, \mathbf{x}_C^\mathsf{T}\right]^\mathsf{T} \in \mathbb{R}^{CK\times 1}$ be the input training sample with C feature channels. To make the description of

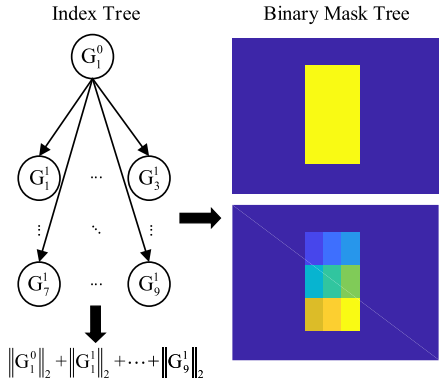

FIGURE 1. The tree structure defined by an index tree, a depth of $d = 1$ and a 3x3 grid of local regions for example. In practice, we generate a binary mask tree and select each local feature group by applying pixel-wise masking. Each color block in the figure represents a local filter group.

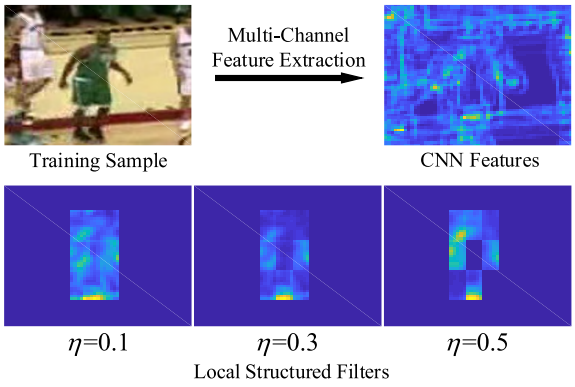

FIGURE 2. A training example shows the effect of the regularization parameter η. The background region around the target in the learned filters is masked to zero. The proposed algorithm adaptively explores hierarchical local structure information inside the holistic correlation filter model.

the algorithm concise, the input training sample is a one-dimensional vector with a dimension of K, but in practice, we extract a two-dimensional feature map for each sample. The desired correlation filter $\mathbf{f} \in \mathbb{R}^{CK\times 1}$ also consists of one sub filter $\mathbf{f}_c$ per feature channel. The SRDCF minimizes a l_2-norm squared error between the response $\mathbf{y}$ and the circular convolution response using the following regularized objective:

$$\arg\min_{\mathbf{f}} \sum_{j=1}^{T} \alpha^j \left\| \sum_{c=1}^{C} \mathbf{x}_c^j * \mathbf{f}_c - \mathbf{y}^j \right\|_2^2 + \sum_{c=1}^{C} \|\mathbf{w} \circ \mathbf{f}_c\|_2^2 \quad (1)$$

Here, the operator $*$ denotes the circular convolution, the operator $\circ$ denotes the Hadamard product, $\mathbf{w}$ are the regularization weights that determine the importance of the filter coefficients depending on spatial locations. Although the SRDCF learns a more discriminative correlation filters by introducing the spatial regularization, two drawbacks can be noted. First, the relationship between the features of circularly shifted samples have not been taken into account. The l_2-norm regularization term, i.e. $\sum_c \|\mathbf{w} \circ \mathbf{f}_c\|_2^2 = \sum_c \sum_k \left[\mathbf{w}_k \circ \mathbf{f}_{c,k}\right]^2$, constrains each feature dimension separately during the minimization of (1). Second, some types of prior information about the inner structure of the target have not been exploited, considering that this may make the learned filter more robust. As a result, we introduce a tree-structured regularization term to group local features and form a hierarchical structure inside the target region.

B. MODELING LOCAL STRUCTURED DCF

A tree structure is described first, as illustrated in Fig. 1. We generate the index tree for the foreground object at the first frame of a video sequence, as defined in [11]. This tree has a depth of d. Let G_j^i be the j-th node at the i-th level, and nodes from the same level have non-overlapping indices, i.e. $G_j^i \bigcap G_k^i = \varnothing, j \neq k, 1 \leq j, k \leq n_i, \forall i = \{1, \cdots, d\}$, n_i gives the number of nodes at i-th level. By constructing such a hierarchical tree, the structure inside the target region can be represented. In practice, $\mathbf{I}_{G_j^i} \in \mathbb{R}^{K\times 1}$ is a binary mask that preserves feature maps at the local feature group G_j^i. $\mathbf{I}_j^i = diag(\mathbf{I}_{G_j^i}) \oplus \cdots \oplus diag(\mathbf{I}_{G_j^i}) \in \mathbb{R}^{CK\times CK}$ denotes a block diagonal binary matrix, $\mathbf{I}^i = \sum_{j=1}^{n_i} \mathbf{I}_j^i$ contain all the nodes at depth i.

For the convenience of derivation, we define the circulant matrix of a training sample instead of using the circular convolution operator. Let $\mathbf{X}_c = \left[\mathbf{x}_{c,\triangle_{(0)}}, \mathbf{x}_{c,\triangle_{(1)}}, \cdots, \mathbf{x}_{c,\triangle_{(K-1)}}\right] \in \mathbb{R}^{K\times K}$ denote the circulant matrix of the c-th feature channel, $\triangle_{(k)}$ means a k-step discrete circular shift to the base feature descriptor $\mathbf{x}_{c,\triangle_{(0)}}$. $\mathbf{X} = \left[\mathbf{X}_1^\mathsf{T}, \mathbf{X}_2^\mathsf{T}, \cdots, \mathbf{X}_C^\mathsf{T}\right]^\mathsf{T} \in \mathbb{R}^{CK\times K}$ represents the concatenation of all C feature channels. $\mathbf{f} = \left[\mathbf{f}_1^\mathsf{T}, \mathbf{f}_2^\mathsf{T}, \cdots, \mathbf{f}_C^\mathsf{T}\right]^\mathsf{T} \in \mathbb{R}^{CK\times 1}$ denotes the correlation filter to be learned. We minimize the following optimization problem:

$$\begin{aligned} \tilde{\mathbf{f}} &= \arg\min_{\mathbf{f}} F(\mathbf{f}; \mathbf{X}) \\ &= \arg\min_{\mathbf{f}} L(\mathbf{f}; \mathbf{X}) + \lambda S(\mathbf{f}; \mathbf{X}) + \eta\Omega(\mathbf{f}) \end{aligned} \quad (2)$$

where

$$L(\mathbf{f}; \mathbf{X}) = \frac{1}{2} \left\| \mathbf{y} - \mathbf{X}^\mathsf{T}\mathbf{I}^d\mathbf{f} \right\|_2^2 \quad (3)$$

$$S(\mathbf{f}; \mathbf{X}) = \sum_{i,j}^{n_d} \left\| \mathbf{X}^\mathsf{T}\mathbf{I}_i^d\mathbf{f} - \mathbf{X}^\mathsf{T}\mathbf{I}_j^d\mathbf{f} \right\|_2^2 \quad (4)$$

$$\Omega(\mathbf{f}) = \sum_{i=0}^{d} \sum_{j=1}^{n_i} \left\| \mathbf{I}_j^i\mathbf{f} \right\|_2 \quad (5)$$

λ is the weight parameter corresponding to the S regularization term. η controls the group-wise sparsity. From the later described optimization algorithm it can be found that the larger the value of the η, the more local filter groups will be penalized to zero. The effect of the value η with different value settings are illustrated in the Fig. 2.

The objective function (2) of LSDCF consists of three terms. The L part is the loss cost defined by the Euclidean error between the Gaussian shaped response and the multi-channel circular convolution response. The leaf node of the

tree is applied to mask the learned filter so that only the target region affects the convolution output. From another perspective, $\mathbf{X}^{\mathsf{T}}\mathbf{I}^{d}\mathbf{f} = \left(\mathbf{I}^{d}\mathbf{X}\right)^{\mathsf{T}}\mathbf{f}$ means that the binary mask is applied to every circularly shifted sample. The boundary effect is largely reduced since the background around the target region is masked to zero. As a result, the filter is trained from real negative training samples that are densely generated from the circular convolution operator. The S part is the smooth term adopted from [12] to make each local feature group contribute equally to the final response. The last part is the tree-structured group sparsity regularization that applies l_2-norm over each local feature group $\mathbf{I}_j^i\mathbf{f}$ and then applies the l_1-norm across all the local feature groups to promote sparsity. Noting that this term is distinctly different with the squared l_2 regularization used in [12]. In [12], the foreground region is regularized as a whole. But here we apply a l_2 regularization for each local region and finally applies the l_1-norm across all the local regions, which is a structured regularization term. In sum, we expect to combine the structured sparsity inducing regularization and local response consistency regularization to learn robust correlation filters for visual tracking.

C. OPTIMIZATION FOR LSDCF

Accelerated proximal gradient method (APG) is applied to solve the proposed non-smooth composite regularization problem (1). The APG was originally proposed by Nesterov *et al.* [13] and then extended for the convex-concave optimization by Tseng [47]. Here, we mainly adopt the optimization procedure described by Chen *et al.* [48] which considers multi-task learning problem with the $L_{1,\infty}$ regularizer.

Firstly, we derive the generalized gradient update step used in the APG optimization process. Let $LS(\mathbf{f};\mathbf{X}) = L(\mathbf{f};\mathbf{X}) + \lambda S(\mathbf{f};\mathbf{X})$ denotes the smooth convex part in the objective function and the rest term $\Omega(\mathbf{f})$ is the 'simple' non-smooth convex part. In a majorization minimization scheme, we majorize the $LS(\mathbf{f};\mathbf{X})$ centered at a point $\mathbf{f}_{(t)}$ by

$$LS(\mathbf{f}) \le LS\left(\mathbf{f}_{(t)}\right) + \left(\mathbf{f}-\mathbf{f}_{(t)}\right)^{\mathsf{T}}\nabla LS\left(\mathbf{f}_{(t)}\right) + \frac{H}{2}\left\|\mathbf{f}-\mathbf{f}_{(t)}\right\|_2^2. \tag{6}$$

By adding the tree-structured spatial regularized term Ω into (6), we have

$$M_H\left(\mathbf{f},\mathbf{f}_{(t)}\right) = LS\left(\mathbf{f}_{(t)}\right) + \left(\mathbf{f}-\mathbf{f}_{(t)}\right)^{\mathsf{T}}\nabla LS\left(\mathbf{f}_{(t)}\right) + \frac{H}{2}\left\|\mathbf{f}-\mathbf{f}_{(t)}\right\|_2^2 + \eta\Omega(\mathbf{f}). \tag{7}$$

$\nabla LS\left(\mathbf{f}_{(t)}\right)$ denotes the sub-differential of $LS(\mathbf{f};\mathbf{X})$ at $\mathbf{f}_{(t)}$.The generalized gradient update step is defined as minimizing $M_H(\cdot)$:

$$m_H\left(\mathbf{f}_{(t)}\right) = \arg\min_{\mathbf{f}} M_H\left(\mathbf{f},\mathbf{f}_{(t)}\right) \tag{8}$$

Equation (8) is equivalent to the following:

$$m_H\left(\mathbf{f}_{(t)}\right) = \arg\min_{\mathbf{f}} \frac{1}{2}\left\|\mathbf{f}-\mathbf{b}\right\|_2^2 + \frac{\eta}{H}\Omega(\mathbf{f}) \tag{9}$$

Algorithm 1 Optimization for the Moreau-Yosida Regularization With Tree-Structured Regularization

Initialization: $\mathbf{b} = \mathbf{f}_{(t)} - \frac{1}{H}\nabla LS\left(\mathbf{f}_{(t)}\right)$,$\tilde{\eta} = \frac{\eta}{H}$, the index tree with all the nodes $\left\{G_j^i | i = 0, 1, \cdots, d; j = 1, 2, \cdots, n_i\right\}$.

1: Define a working variable $\mathbf{v}^{(d+1)} = \mathbf{b}$
2: **for** $i = d$ to 0 **do**
3: **for** $j = 1$ to n_i **do**

$$\mathbf{v}_{G_j^i}^{(i)} = \begin{cases} \mathbf{0}, & \left\|\mathbf{v}_{G_j^i}^{(i+1)}\right\|_2 \le \tilde{\eta} \\ \dfrac{\left\|\mathbf{v}_{G_j^i}^{(i+1)}\right\|_2 - \tilde{\eta}}{\left\|\mathbf{v}_{G_j^i}^{(i+1)}\right\|_2}\mathbf{v}_{G_j^i}^{(i+1)}, & \left\|\mathbf{v}_{G_j^i}^{(i+1)}\right\|_2 > \tilde{\eta} \end{cases} \tag{10}$$

4: **end for**
5: **end for**

Output: $m_H\left(\mathbf{f}_{(t)}\right) = \mathbf{v}^0$

where $\mathbf{b} = \mathbf{f}_{(t)} - \frac{1}{H}\nabla LS\left(\mathbf{f}_{(t)}\right)$ is an auxiliary variable. This is actually the Moreau-Yosida regularization associated with the non-smooth grouped structure regularization. The work of [11] has given an analytical solution for this kind of problem, which is shown in Algorithm 1. In the Algorithm 1, the working variable is updated in the reverse breadth-first order along the index tree. At the node G_j^i, the l_2-norm of the local feature group $\mathbf{v}_{G_j^i}$ is compared with the threshold $\tilde{\eta}$. The l_2-norm of each local feature group will shrink by at most $\tilde{\eta}$ if it is not directly penalized to zero.

Secondly, we derive the differential of the smooth convex part used in the generalized gradient update step. The differential of the LS part can be decomposed into calculations on each feature channel:

$$\begin{aligned}\nabla LS(\mathbf{f}) &= \nabla L(\mathbf{f}) + \lambda\nabla S(\mathbf{f}) \\ &= \left[\nabla LS(\mathbf{f}_1)^{\mathsf{T}}, \cdots, \nabla LS(\mathbf{f}_C)^{\mathsf{T}}\right]^{\mathsf{T}}\end{aligned} \tag{11}$$

For the L part in $\nabla LS(\mathbf{f}_c)$, it can be derived as:

$$\begin{aligned}\nabla L(\mathbf{f}_c) &= \mathbf{I}^d(c)\mathbf{X}_c\left(\mathbf{X}^{\mathsf{T}}\mathbf{I}^d\mathbf{f}-\mathbf{y}\right) \\ &= \mathbf{I}^d(c)\mathcal{F}^{-1}\left(\hat{\mathbf{x}}_c \circ \Delta\hat{\mathbf{y}}_L\right)\end{aligned} \tag{12}$$

where $\Delta\hat{\mathbf{y}}_L = \sum_{c=1}^{C}\hat{\mathbf{x}}_c^* \circ \mathcal{F}\left(\mathbf{I}^d(c)\mathbf{f}_c\right) - \hat{\mathbf{y}}$.

For the S part in $\nabla LS(\mathbf{f}_c)$, it can be derived as:

$$\begin{aligned}\nabla S(\mathbf{f}_c) &= 4n_d\sum_{i=1}^{n_d}\mathbf{I}_i^d(c)\mathbf{X}_c\mathbf{X}^{\mathsf{T}}\mathbf{I}_i^d\mathbf{f} - 4\mathbf{I}^d(c)\mathbf{X}_c\mathbf{X}^{\mathsf{T}}\mathbf{I}^d\mathbf{f} \\ &= 4n_d\sum_{i=1}^{n_d}\mathbf{I}_i^d(c)\mathcal{F}^{-1}\left(\hat{\mathbf{x}}_c \circ \sum_{c=1}^{C}\hat{\mathbf{x}}_c^* \circ \mathcal{F}\left(\mathbf{I}_i^d(c)\mathbf{f}_c\right)\right) \\ &\quad - 4\mathbf{I}^d(c)\mathcal{F}^{-1}\left(\hat{\mathbf{x}}_c \circ \sum_{c=1}^{C}\hat{\mathbf{x}}_c^* \circ \mathcal{F}\left(\mathbf{I}^d(c)\mathbf{f}_c\right)\right)\end{aligned} \tag{13}$$

Algorithm 2 Accelerated Proximal Gradient Optimization for the Proposed LSDCF Method

Initialization: $H_0 > 0, \xi > 0$, the initial filter $\mathbf{f}_{(0)} \in \mathbb{R}^{CK\times 1}$, the working variable $\mathbf{v}_{(0)} = \mathbf{f}_{(0)}$,$\alpha_0 = 1$, the iterator $t = 0$ and the maximum iterations t_{max}.

1: **repeat**
2: $H = H_t$;
3: **while** $F(m_H(\mathbf{v}_{(t)})) > M_H(m_H(\mathbf{v}_{(t)}), \mathbf{v}_{(t)})$ **do**
4: $H = \xi H$;
5: **end while**
6: $H_{t+1} = H$;
7: $\mathbf{f}_{(t+1)} = m_{H_{t+1}}(\mathbf{v}_{(t)})$;
8: $\alpha_{t+1} = \frac{2}{t+3}$;
9: $\mathbf{v}_{(t+1)} = \mathbf{f}_{(t+1)} + \frac{1-\alpha_t}{\alpha_t}\alpha_{t+1}\left(\mathbf{f}_{(t+1)} - \mathbf{f}_{(t)}\right)$;
10: $t = t + 1$;
11: **until** $t = t_{max}$

Output: $\tilde{\mathbf{f}} = \mathbf{f}_{(t_{max})}$

where the variable with a hat $\hat{\cdot}$ denotes the DFT transform $\hat{\mathbf{v}} = \mathcal{F}(\mathbf{v})$ and the $\mathcal{F}^{-1}$ denotes the inverse DFT operation. Through (12) and (13) we may find that the correlation filters are bond with the binary mask to ensure the defined spatial structure is incorporated in the gradient updating process. The property of circulant matrix is fully exploited by applying DFT transforms. As such, the main computation cost for matrix multiplication during the APG iterations can be greatly reduced by performing elementwise multiplication in the frequency domain.

Thirdly, Algorithm 2 summarizes the proposed learning procedure for LSDCF. It can be found in the Step 3-5 that the APG algorithm searches for a suitable H value to make the majorization inequality hold. And the Step 9 is a momentum update process with varying step size. In the next section, we will elaborate the process of target localization using the proposed LSDCF method.

D. LSDCF BASED TRACKERS

In the training phase, we generate an index tree based on the object bounding box given by the first frame of the input video sequence. The tree structure is constructed by dividing the feature maps into a set of feature submaps, e.g. a 3×3 grid, and repeating such a process for d times. A new training sample patch centered at the target location is cropped, and then feature maps for this sample patch are extracted. For the case of integrating multi-resolution feature maps, we develop two trackers based on the proposed LSDCF model, namely LSDCFd and LSDCFc.

LSDCFd denotes a basic LSDCF tracker that independently trains discrete correlation filters at each feature resolution. We perform a late fusion strategy in the detection phase, which is to superimpose the response maps for each feature resolution in the frequency domain. Further scale filtering and displacement searching are performed on the final fusion response map.

LSDCFc represents a continuous LSDCF tracker that learns correlation filters in the continuous domain. The multi-resolution feature maps are first interpolated by a predefined interpolation function. The joint optimization process also uses Algorithm 2. It should be noted that the summation symbols in (12) and (13) refer to summing all feature channels of all feature resolutions. In the detection phase, continuous response maps for each feature resolution are computed and fused for further process.

We perform model updates in each frame and combine the newly learned filters with the previously learned filters in a moving average manner:

$$\bar{\mathbf{f}} = (1-\alpha)\bar{\mathbf{f}} + \alpha\tilde{\mathbf{f}} \tag{14}$$

where $\bar{\mathbf{f}}$ is the updated template and α is the learning rate. We find this to be a simple but effective strategy for the proposed method. The LSDCF method can focus on learning enough discriminative features at each frame and maintain a robust appearance template over the temporal span in an accumulating manner. It is worth noting that a sparse update mechanism combined with multiple frames learning have been proved to be more effective than performing dense update on every single frame [4]. But in order to succinctly highlight the proposed method, we regard this as a useful extension and have not directly integrated it into the proposed tracking method. In the detection phase, image patches at S different scales (or resolutions) are cropped centered around the target position of last frame. The updated template $\bar{\mathbf{f}}$ is then applied to each scale independently:

$$R_s = p_s\mathcal{F}^{-1}\left\{\sum_{c=1}^{C}\hat{\mathbf{x}}_{s,c}^{*}\circ\hat{\bar{\mathbf{f}}}_c\right\}, \quad s = 1, 2, \cdots, S \tag{15}$$

where p_s is the scale penalty factor that constrains rapid scale changes in the detection process, $\hat{\mathbf{x}}_{s,c}$ denotes the c-th channel feature map at scale s. A Gaussian-like motion window is applied to reweight the response maps to smooth the target motion trajectory. The location and scale of the target are obtained by finding the peak score in all the S response maps $\{R_s|s = 1, 2, \cdots, S\}$.

IV. EXPERIMENTAL RESULTS

In this section, we present extensive experimental results. First, we introduce the overall experimental setup. Second, we analysis the impacts of different key parameter settings on the proposed LSDCF method. Third, we provide quantitative, qualitative and attributes comparisons on four public datasets with a number of state-of-the-art trackers.

A. EXPERIMENTAL SETUP

Our tracking method is implemented with Matlab 2017a, and has an average speed of 3.8FPS without code optimization. All experiments are evaluated on a desktop PC with Intel Core i7-8700K CPU @ 3.7GHz, 16GB RAM and a single NVIDIA GTX 1080Ti GPU.

1) GENERAL PARAMETER SETTING

The key parameters, APG optimization maximum iterations t_{max}, the regularization parameters λ and η, are evaluated on the OTB2013 dataset in the following section. The rest parameters are empirically determined and fixed for all the test sequences as follows. The image patch cropping size is determined by multiplying the scale factor by the first frame target size, where the scale factor is set to 3. The area of the cropped image patch is then scaled into the closed interval $[250^2, 300^2]$. Then the cropped image patch is fed into the 2-layer and 5-layer AlexNets trained by the CFnet [49] to extract deep features. For example, if the size of a cropped patch is 250×250, then the sizes of the extracted features are 55×55 and 47×47 (i.e. value K). We apply L2 normalization to the extracted features. The extracted features are not weighted by a cosine window in our practice. The scale pyramid S is set to 3 and the scale penalty p_s is 0.98. The template learning rate α is 0.005. In the Algorithm 2, ξ is set to 1.1. H_0 is set to 500 for LSDCFd and 70 for LSDCFc. The optimization for LSDCFd is always started from $\mathbf{f}_{(0)} = \mathbf{0}$. As for LSDCFc, we initialize $\mathbf{f}_{(0)}$ with the filter learned in the previous frame.

2) EVALUATION DATASET AND METRICS

To evaluate our tracking algorithm, we conduct experiments with four popular visual tracking benchmarks, namely the OTB 2013 [15] (includes 51 sequences), OTB 2015 [14] (includes 100 sequences), VOT 2017 [50], [51] (includes 60 sequences), and TC128 [52] (includes 128 sequences). Each sequence in the OTB dataset are annotated with different attributes to facilitates analyzing the specific performance of each tracker. These attributes include illumination variation (IV), out-of-plane rotation (OPR), scale variation (SV), occlusion (OCC), deformation (DEF), motion blur (MB), fast motion (FM), in-plane rotation (IPR), out of view (OV), background clutter (BC), low resolution (LR). For the comparison on OTB datasets, trackers are quantitatively evaluated by plotting the precision plots and the success plots under the one pass evaluation (OPE) criterion. The precision plot gives the percentage of frames whose location error is within the given Euclidean distance threshold. The success plot gives the percentage of frames where the overlap score is greater than the given threshold. Average distance precision at threshold = 20 pixels and average area under curve (AUC) are applied in the precision plot and success plot to rank trackers, respectively. For comparisons on the VOT dataset, we measure the performance in terms of Expected Average Overlap (EAO). For comparsions on the TC128 dataset, the performance metrics are the same to those used by the OTB dataset.

B. PARAMETER ANALYSIS

In this section, we validate the impacts of several key parameters on the OTB2013 dataset to avoid overfitting the parameters on all the video benchmarks. Instead of doing a time-consuming grid search, we take the strategy that the later parameter is linearly searched after the optimal value for the previous one is determined. Here we conduct detailed parameter analysis for the LSDCFd tracker. As for LSDCFc tracker, the evaluation procedure is basically the same. The initial set of key parameters for LSDCFd tracker is set to $\{t_{max} = 10, \eta = 0.01, \lambda = 1\}$.

TABLE 1. Key parameter settings for LSDCFd and LSDCFc.

	LSDCFd	LSDCFc
t_{max}	[16 16]	[10 3][a]
η	0.002	0.5
λ	1.1	1

[a]The first number in the bracket indicates the t_{max} used in the first frame, and the second number indicates the t_{max} used since the second frame.

Fig. 3(a) gives the impact of the maximum iterations for APG optimization. It can be found that the choice of maximum iterations affects the proposed tracker performance with a performance gain of at least 10%, comparing $t_{max} = 2$ and $t_{max} = 4$. As the number of iterations increases, the accuracy of the proposed method also increases, and tends to saturate after a certain number of iterations. From Fig. 3(a), the best performance is achieved around $t_{max} = 16$. Thus we fix the maximum iterations to be 16 for the LSDCFd tracker. Fig. 3(b) gives the impact of the regularization parameter η. This parameter controls the sparsity threshold in the (10). The lower the parameter η, the more groups can adaptively shrink in their respective proportions in the optimization process. In contrast, the higher the parameter η, the more groups will be penalized to zero, leading to a more sparse correlation filter. From the evaluation results in Fig. 3(b) we may find that a very sparse correlation filter is not the optimal solution for visual tracking, since the precision curve declines as the value of η becomes larger. This may suggest that the local feature groups with low response scores also contain valuable information and contribute to the generalization ability of the correlation filter. Thus we fix the η to be 0.002. The evaluation results of the regularization parameter λ is shown in Fig. 3(c). We note that $\lambda = 0$ implies that the objective function is optimized without the local response consistency part. From the evaluation results in Fig. 3(c), the best performance is achieved at $\lambda = 1.1$. Finally, Table 1 summarizes the key parameter settings for both two trackers.

C. PERFORMANCE EVALUATION ON OTB BENCHMARKS

On the OTB benchmarks, we compare the proposed LSDCF method with the reported top 2 trackers in OTB2013, including Struck [53], TLD [54], and 15 state-of-the-art methods, including KCF [18], CFNet-2 [49], SAMF [55], DSST [56], TGPR [57], SRDCF [2], DeepSRDCF [58], BACF [5], RSSTDeep [32], RPT [8], MEEM [59], CSR-DCF [60], Staple [61], MUSTer [62], SRDCFdecon [63]. Note that we only employ the codes released by the authors or the raw results released along with the papers for fair comparison.

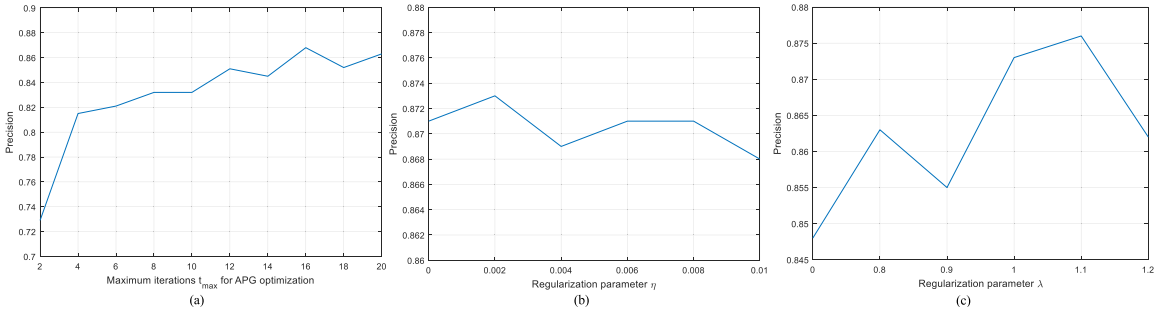

FIGURE 3. Impacts of (a) the maximum iterations t_{max}, (b) the regularization parameter η, and (c) the regularization parameter λ on the OTB2013 benchmark. As the number of iterations increases, the accuracy of the proposed method also increases, and tends to saturate after a certain number of iterations. The regularization parameters η and λ also have impacts on the performance of the proposed LSDCFd tracker.

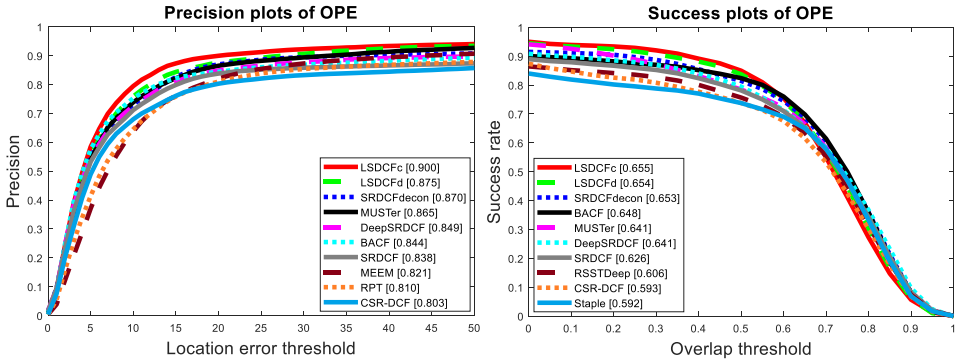

FIGURE 4. Overall performance comparison on the OTB 2013 benchmark using precision plot and success plot under OPE criterion. For clarity, only the top 10 trackers are displayed. Our method performs favorably against the state-of-the-art trackers in terms of precision rate and AUC rate.

1) QUANTITATIVE PERFORMANCE ON OTB2013

We first illustrate the comparison of the proposed trackers, the continuous version LSDCFc and the discrete version LSDCFd, with the selected 17 state-of-the-art trackers in the Fig. 4. It is shown that the proposed LSDCF method outperforms different variants of the SRDCF method and other state-of-the-art trackers in terms of precision rate and success rate. In the SRDCF method, the regularization weights smoothly penalize the filter coefficients depending on their spatial locations. The DeepSRDCF method investigates the use of convolutional layer activations for DCF based tracking. However, our proposed LSDCFc tracker effectively explores the inner structure of the holistic DCF model and outperforms the SRDCF, DeepSRDCF by 6.2%/2.9% and 5.1%/1.4% in terms of precision and success rate, respectively. This suggests that the proposed method is more robust in handling complex scenarios. Comparing to BACF, our LSDCFc tracker achieves tracking performance gains of 5.6% and 0.7% in precision rate and success rate, respectively. Comparing to one of the part-based methods, RPT, our method also shows a superior tracking performance (+9.0%/ + 7.8%).

Furtherly, we present the precision plots and success plots of OPE for 9 attributes in Fig. 5. From the figure plots it shows that the proposed LSDCFc and LSDCFd trackers rank top in these challenge scenarios. Excluding our LSDCFd tracker, LSDCFc outperforms the second best method in 7 attribute comparisons, which is listed as follows: OPR (+2.6%/ + 0.5%), SV (+8.1%/ + 4.7%), OCC (+2.3%/ + 0.2%), FM (+2.0%/+1.0%), OV (+0.5%/+0.4%), BC (+2.9%/+ 1.2%) and LR (+14.9%/ + 11.7%). In particular, the large margin of improvement in the LR scenario demonstrates that the proposed LSDCF method can robustly handle low resolution target by using a relative small padding margin and learning from local target appearances. For the factors of IV and IPR, our LSDCFc tracker performs better than the second best tracker in the precision plots (+2.1%, +2.5%), but performs slightly worse than the best tracker in the success plots (−0.4%, −1.7%), respectively. In these two challenging scenarios, our LSDCFc tracker can effectively locate the target but does not adapt well to the target scale change, resulting in a slightly worse bounding box overlap score.

2) QUANTITATIVE PERFORMANCE ON OTB2015

To evaluate the proposed LSDCF method more comprehensively, we conduct more experiments on a larger benchmark, namely OTB 2015. Fig. 6 shows the OPE precision plot and success plot of the top 10 best performing tracking methods. In this comparison, the proposed LSDCFc tracker ranks the first place and shows a comparable or better performance than the selected state-of-the-art trackers. It should be noted

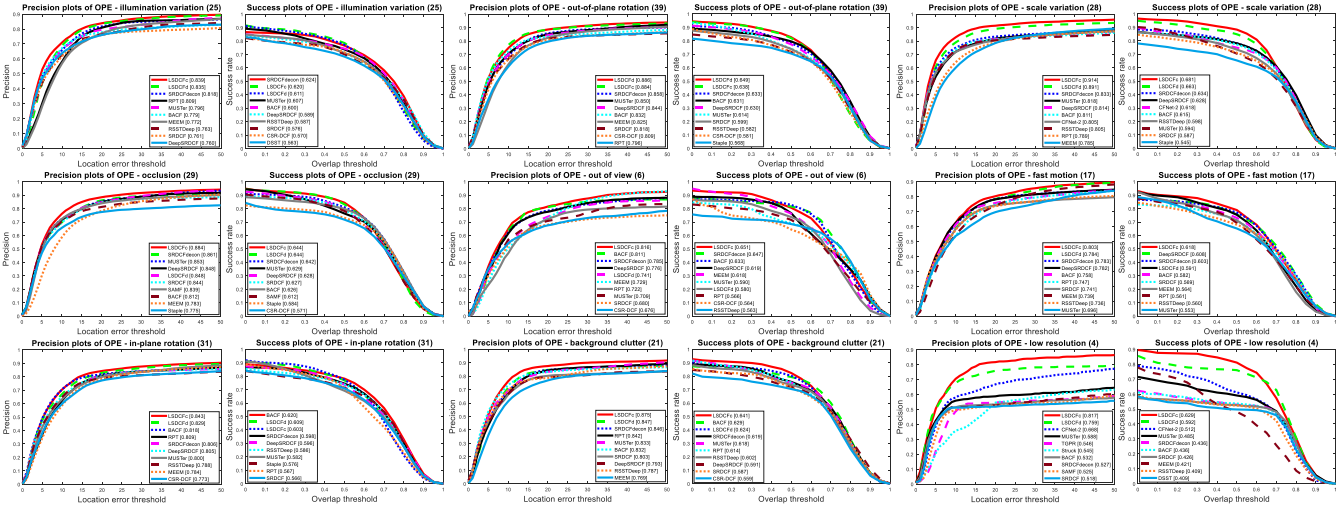

FIGURE 5. Precision plots and success plots of 9 attributes (IV, OPR, SV, OCC, OV, FM, IPR, BC, LR) on the OTB 2013 benchmark. Our method ranks top on most of the attribute plots.

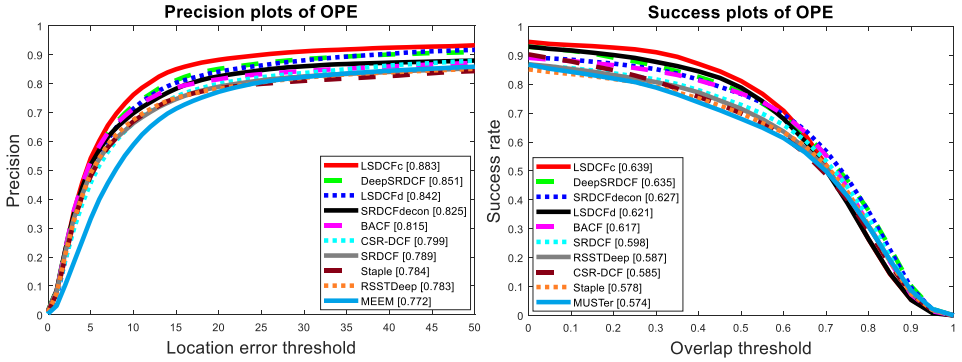

FIGURE 6. Overall performance comparison on the OTB 2015 benchmark using precision plot and success plot under OPE criterion. For clarity, only the top 10 trackers are displayed. In general, our LSDCF tracker performs favorably against the state-of-the-art trackers in terms of precision rate and AUC rate.

that the proposed LSDCF method adopts the same moving average update scheme, and extracts the same deep features as the CFNet tracking framework, but achieves a large margin of improvement (+12.3%/ + 6.6%). This suggests that the proposed method learns a more effective discriminative correlation model by considering local structural information.

Similarly, we also illustrate the performance of the trackers on different attribute challenges in Fig. 7 and Table 2. In Table 2, we compare the proposed two trackers with holistic model based methods SRDCF, DeepSRDCF and SRDCFdecon. It can be found that the LSDCFc tracker performs the best in most of the challenge scenarios. From the figure plots we may find that the proposed LSDCFc tracker can effectively cope with the challenges of IV (+2.6%/ + 0.5%), OPR (+3.4%/ + 0.4%), SV (+5.3%/ + 1.5%), OCC (+2.8%/ + 1.7%), DEF (+5.1%/ + 1.7%), FM (+1.6%/ + 2.3%), OV (+2.5%/+3.6%) and LR (+10.5%/+6.5%), and ranks the first place among these challenges under both evaluation metrics. The most significant improvement is achieved in the LR scenario with a 10.5% in precision rate increment and 6.5% in success rate increment, compared to the selected trackers. This is consistent with the experimental results evaluated on the OTB 2013 benchmark. Compared to the evaluation results on OTB 2013, our method ranks highest in the OTB2015 DEF scenario with more video sequences, indicating that the proposed LSDCF method can robustly handle target deformations in more challenging scenarios.

3) QUALITATIVE PERFORMANCE ON CHALLENGE SEQUENCES

In this section, we discuss the qualitative performance of the proposed LSDCFc tracker in six challenging attributes. Tracking results of 11 top performing trackers are plotted in 12 video sequences, as illustrated in Fig. 8.

a: ILLUMINATION VARIATION

The first row of Fig. 8 shows the tracking results in sequence *Ironman* and *Matrix*. In these two sequences, it shows significant illumination changes which severely affect the extracted target appearances. The MEEM tracker shows a gradual drift

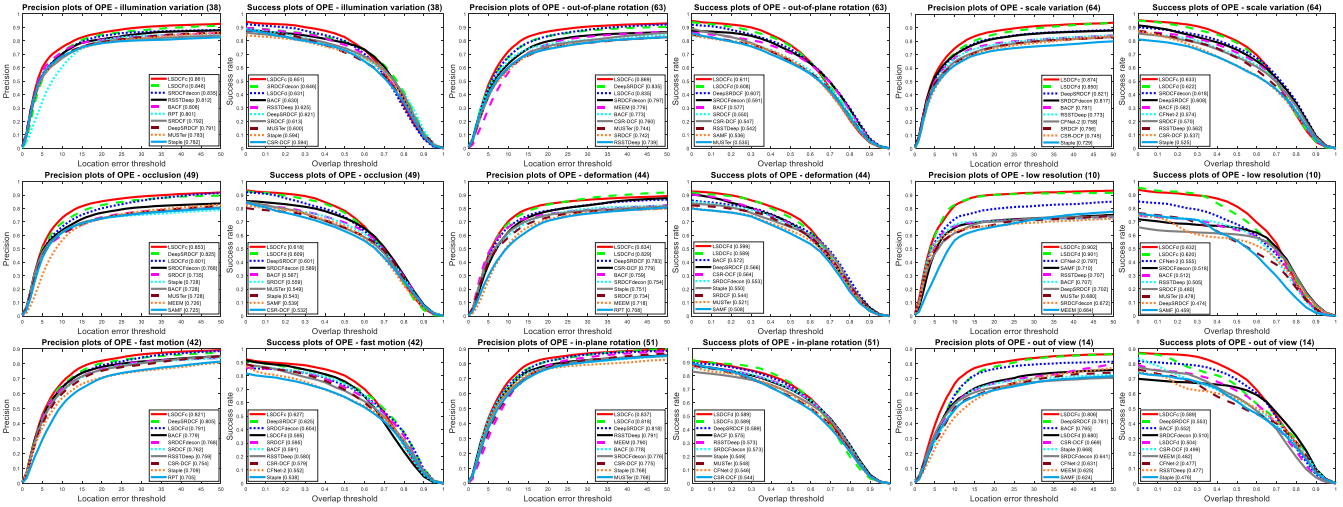

FIGURE 7. Precision plots and success plots of 9 attributes (IV, OPR, SV, OCC, DEF, LR, FM, IPR, OV) on the OTB 2015 benchmark. Our proposed two trackers, LSDCFc and LSDCFd, both rank top on most of the attribute plots.

TABLE 2. Precision (top) and success rate (bottom) of the evaluated trackers. Top two trackers of each column are shown in red and blue.

Attributes	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
LSDCFc	**0.861**	**0.869**	**0.874**	**0.853**	**0.834**	0.820	**0.821**	**0.837**	**0.806**	**0.892**	**0.902**
LSDCFd	**0.846**	**0.835**	**0.850**	0.801	**0.829**	0.787	0.791	**0.818**	0.680	0.808	**0.901**
SRDCF [2]	0.792	0.742	0.756	0.735	0.734	0.782	0.762	0.745	0.597	0.775	0.631
DeepSRDCF [58]	0.791	0.835	0.821	**0.825**	0.783	**0.834**	**0.805**	0.818	**0.781**	0.841	0.702
SRDCFdecon [63]	0.835	0.797	0.817	0.768	0.754	**0.826**	0.768	0.776	0.641	**0.850**	0.672
Attributes	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
LSDCFc	**0.651**	**0.611**	**0.633**	**0.618**	**0.589**	0.642	**0.627**	**0.589**	**0.589**	**0.638**	**0.620**
LSDCFd	0.631	**0.608**	**0.622**	**0.609**	**0.599**	0.600	0.595	**0.589**	0.504	0.598	**0.632**
SRDCF [2]	0.613	0.550	0.570	0.559	0.544	0.610	0.595	0.544	0.460	0.583	0.480
DeepSRDCF [58]	0.621	0.607	0.608	0.601	0.566	**0.656**	**0.625**	0.589	**0.553**	0.627	0.474
SRDCFdecon [63]	**0.646**	0.591	0.618	0.589	0.553	**0.653**	0.604	0.573	0.510	**0.641**	0.518

in the *Ironman* sequence. The RSSTDeep tracker drifts away from the target rapidly in the *Matrix* sequence. Comparing to the other trackers, our tracker has the smallest drift distance from the target center and performs the best. Part of the reason is that our tracker exploits the deep features that extract mid-level structural information. At the same time, the conservative template update strategy make the illumination variation affect less on the learned correlation model.

b: OUT-OF-PLANE ROTATION

The second row of Fig. 8 shows the tracking results in sequence *Box* and *Football*. In these two sequences, the target often rotates out of the image plane and gives unstable appearances. In the *Box* sequence, the MEEM, RSSTDeep and MUSTer trackers all track the foreground part that appears in previous frames and do not locate well to the newly emerging part. In the *Football* sequence, some trackers are misled under the situation that background objects share a highly similar appearance with the rotated target. In both cases, our tracker shows a more robust tracking performance. We attribute this to the adoption of local structures inside the holistic model that reinforces the response from local stable features.

c: SCALE VARIATION

The third row of Fig. 8 shows the tracking results in sequence *Human3* and *Human9*. Despite the targets experience great scale change during the movement, the simple but powerful multiscale search strategy adopted in the proposed method shows a robustness. Since the scale changes of the target are not well handled, the models learned by the rest trackers, such as SRDCF and DeepSRDCF, quickly degenerate and eventually lose the target.

d: DEFORMATION

The fourth row of Fig. 8 shows the tracking results in sequence *Girl2* and *Skater2*. Usually, the target deforms locally and resulting in an increase in the representation error of the rigid appearance model. It shows in the two sample sequences that the RPT, MEEM trackers all successfully tracks the deforming target. But the proposed tracker achieves a more precise bounding box prediction.

e: LOW RESOLUTION

The fifth row of Fig. 8 shows the tracking results in sequence *Biker* and *Skiing*. From the quantitatively results on attributes comparison in previous section it shows that the proposed

FIGURE 8. Tracking results of the top 11 trackers on the 12 video sequences: *Ironman*, *Matrix*, *Box*, *Football*, *Human3*, *Human9*, *Girl2*, *Skater2*, *Biker*, *Skiing*, *MotorRolling*, and *Jump*. The frame index is shown in yellow color at the top left of each frame. Each row in the figure illustrates the tracking results of two sequences with the same dominant attribute.

method achieves the largest improvement in this scenario. The tracking results illustrated in sequence *Biker* and *Skiing* also confirm this conclusion. These two sequences are quite challenging for visual trackers since the targets experiences SV, OCC, FM, OPR, OV, LR. However, the proposed tracker effectively extracts information from limited image pixels and robustly handles the target movement, which may benefit from using a relative small padding margin and learning from local target appearances.

f: IN-PLANE ROTATION

The last row of Fig. 8 shows the tracking results in sequence *MotorRolling* and *Jump*. From these two sequences it shows that the proposed tracker performs the best in handling the rotated targets. This also suggests that the proposed method learns an effective discriminative correlation model by adopting the proposed spatial joint regularization.

D. PERFORMANCE EVALUATION ON VOT2017 BENCHMARK

Fig. 9 shows the ranking results in terms of expected average overlap (EAO) in VOT2017, from which we can observe that our proposed LSDCFc tracker exceed the reported state-of-the-art bound set by the VOT 2017 report. Compared to the 51 trackers reported in the VOT2017, our tracker achieves a state-of-the-art performance and is superior to the holistic model SRDCF, indicating that the proposed method enables the tracker to benefit from processing local information.

TABLE 3. Comparison with some recent trackers on Temple Color-128 benchmark. Precision (at 20 pixels threshold) and success rate (AUC score) are displayed.

Trackers	Precision	Success rate	Year
SRDCF [2]	0.689	0.505	2015
DeepSRDCF [58]	0.733	0.532	2015
SRDCFdecon [63]	0.722	0.530	2016
FST-ML [64]	0.605	0.433	2018
CCT [65]	0.706	0.522	2019
OAPT [66]	0.715	0.508	2018
LRWR [67]	0.734	0.529	2019
SOSCF [68]	0.759	0.549	2019
MCPF [69]	0.774	0.545	2017
CCOT [3]	0.774	0.569	2016
LSDCFd	0.725	0.518	
LSDCFc	0.757	0.529	

E. PERFORMANCE EVALUATION ON TC128 BENCHMARK

Furtherly we compare the proposed two trackers, LSDCFd and LSDCFc, with some recent state-of-the-art trackers on the Temple-Color-128 benchmark, as listed in Table 3. It shows in the table that our tracker LSDCFc ranks the fourth place in this comparison list. The proposed LSDCFc tracker outperforms the baseline tracker SRDCF and its two variants

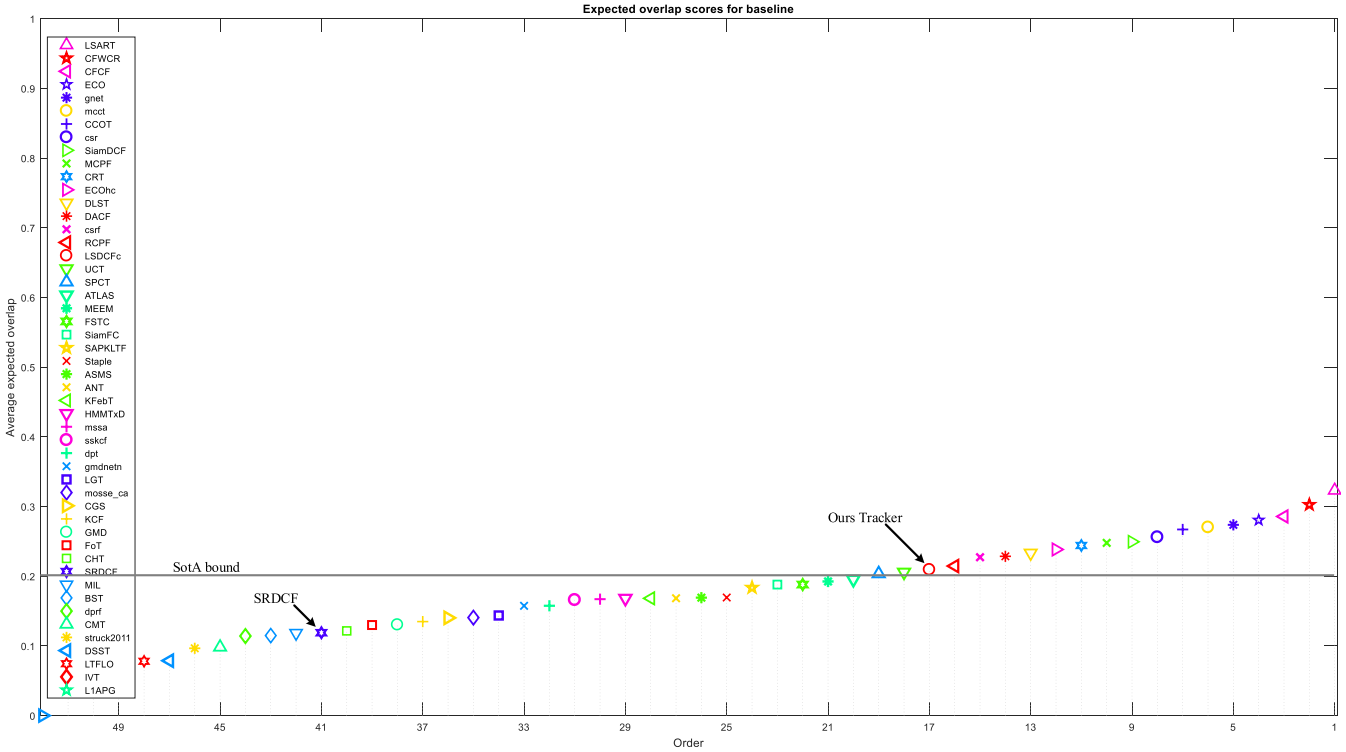

FIGURE 9. Expected average overlap (EAO) graph with trackers ranked from right to left on VOT2017. Our proposed LSDCFc tracker exceed the state-of-the-art bound announced by the report of VOT2017.

in precision and achieves a comparable performance in success rate. Meanwhile, our tracker LSDCFc gains of 4.2% in precision and 2.1% in success rate compared to the recent part-based tracker OAPT [66]. Overall, our proposed tracker achieves competitive results against the state-of-the-art trackers on the Temple-Color-128 benchmark.

V. CONCLUSION

In this paper, we propose a robust visual tracking method based on the spatial joint regularization, which combines the tree-structured group sparsity regularization and the local response consistency regularization. In our tracker, the dependency between hierarchical local filter parts inside the target region are characterized by adopting the proposed spatial joint regularization. The accelerated proximal gradient method is employed to optimized this joint regularization problem. Extensive experiments demonstrate that, under the challenging scenarios such as illumination and scale variation, rotation, deformation, low resolution, the proposed tracking method achieves higher precision or better robustness compared to the state-of-the-art methods.

REFERENCES

[1] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2544–2550.

[2] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4310–4318.

[3] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Computer Vision—ECCV*. New York, NY, USA: Springer, 2016, pp. 472–488.

[4] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6931–6939.

[5] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Oct. 2017, pp. 21–26.

[6] J. Johnander, M. Danelljan, F. S. Khan, and M. Felsberg, "DCCO: Towards deformable continuous convolution operators for visual tracking," in *Computer Analysis of Images and Patterns*. New York, NY, USA: Springer, 2017, pp. 55–67.

[7] T. Liu, G. Wang, and Q. Yang, "Real-time part-based visual tracking via adaptive correlation filters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4902–4912.

[8] Y. Li, J. Zhu, and S. C. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 353–361.

[9] A. Lukežič, L. Č. Zajc, and M. Kristan, "Deformable parts correlation filters for robust visual tracking," *IEEE Trans. Cybern.*, vol. 48, no. 6, pp. 1849–1861, Jun. 2018.

[10] Y. Sui, G. Wang, and L. Zhang, "Correlation filter learning toward peak strength for visual tracking," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1290–1303, Apr. 2018.

[11] J. Liu and J. Ye, "Moreau-yosida regularization for grouped tree structure learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1459–1467.

[12] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Correlation tracking via joint discrimination and reliability learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 489–497.

[13] Y. Nesterov, "Gradient methods for minimizing composite functions," *Math. Program.*, vol. 140, no. 1, pp. 125–161, 2013.

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

[15] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2411–2418.
[16] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Computer Vision—ECCV*. New York, NY, USA: Springer, 2012, pp. 702–715.
[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, no. 1, pp. 886–893.
[18] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.
[19] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3074–3082.
[20] M. Danelljan, F. S. Khan, M. Felsberg, and J. van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1090–1097.
[21] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Discriminative scale space tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1561–1575, Aug. 2017.
[22] F. Li, Y. Yao, P. Li, D. Zhang, W. Zuo, and M.-H. Yang, "Integrating boundary and center correlation filters for visual tracking with aspect ratio variation," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2001–2009.
[23] Y. Qi *et al.*, "Hedging deep features for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.
[24] H. K. Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4630–4638.
[25] S. Zhang, Y. Qi, F. Jiang, X. Lan, P. C. Yuen, and H. Zhou, "Point-to-set distance metric learning on deep representations for visual tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 187–198, Jan. 2018.
[26] Y. Wu, B. Ma, M. Yang, J. Zhang, and Y. Jia, "Metric learning based structural appearance model for robust visual tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 5, pp. 865–877, May 2014.
[27] S. Zhang, X. Lan, H. Yao, H. Zhou, D. Tao, and X. Li, "A biologically inspired appearance model for robust visual tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2357–2370, Oct. 2017.
[28] X. Mei and H. Ling, "Robust visual tracking using ℓ_1 minimization," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Oct. 2009, pp. 1436–1443.
[29] H. Fan and J. Xiang, "Robust visual tracking with multitask joint dictionary learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 5, pp. 1018–1030, May 2017.
[30] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2042–2049.
[31] 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.
[32] T. Zhang, C. Xu, and M. H. Yang, "Robust structural sparse tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 473–486, Feb. 2018.
[33] P. Feng *et al.*, "Sparse representation combined with context information for visual tracking," *Neurocomputing*, vol. 225, pp. 92–102, Feb. 2017.
[34] S. Zhang, H. Zhou, F. Jiang, and X. Li, "Robust visual tracking using structurally random projection and weighted least squares," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1749–1760, Nov. 2015.
[35] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. Ser. B Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[36] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Roy. Statist. Soc., B (Statist. Methodol.)*, vol. 68, no. 1, pp. 49–67, 2006.
[37] L. Meier, S. van de Geer, and P. Bühlmann, "The group lasso for logistic regression," *J. Roy. Stat. Soc. B, Stat. Methodol.*, vol. 70, no. 1, pp. 53–71, Feb. 2008.
[38] J. Friedman, T. Hastie, and R. Tibshirani. (2010). "A note on the group lasso and a sparse group lasso." [Online]. Available: https://arxiv.org/abs/1001.0736
[39] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A sparse-group lasso," *J. Comput. Graph. Statist.*, vol. 22, no. 2, pp. 231–245, May 2012.
[40] J. Liu and J. Ye. (2010). "Fast overlapping group lasso." [Online]. Available: https://arxiv.org/abs/1009.0306
[41] N. Rao, R. Nowak, C. Cox, and T. Rogers. (2014). "Classification with sparse overlapping groups." [Online]. Available: https://arxiv.org/abs/1402.4512
[42] P. Zhao, G. Rocha, and B. Yu, "The composite absolute penalties family for grouped and hierarchical variable selection," *Ann. Statist.*, vol. 37, no. 6A, pp. 3468–3497, 2009.
[43] Z. Zhang *et al.*, "Discriminative lasso," *Cognit. Comput.*, vol. 8, no. 5, pp. 847–855, Oct. 2016.
[44] S. Kim and E. P. Xing, "Tree-guided group lasso for multi-task regression with structured sparsity," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2010, pp. 543–550.
[45] L. Luo, L. Chen, J. Yang, J. J. Qian, and B. Zhang, "Tree-structured nuclear norm approximation with applications to robust face recognition," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5757–5767, Dec. 2016.
[46] B. Li, W. Xiong, W. Hu, and B. Funt, "Multi-cue illumination estimation via a tree-structured group joint sparse representation," *Int. J. Comput. Vis.*, vol. 117, no. 1, pp. 21–47, 2016.
[47] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *SIAM J. Optim.*, to be published.
[48] X. Chen, W. Pan, J. T. Kwok, and J. G. Carbonell, "Accelerated gradient method for multi-task sparse learning problem," in *Proc. ICDM*, Dec. 2009, pp. 746–751.
[49] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5000–5008.
[50] M. Kristan *et al.*, "A novel performance evaluation methodology for single-target trackers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2137–2155, Nov. 2016.
[51] M. Kristan *et al.*, "The visual object tracking VOT2017 challenge results," in *Proc. ICCV Workshops*, Oct. 2017, pp. 1949–1972.
[52] P. Liang, E. Blasch, and H. Ling, "Encoding color information for visual tracking: Algorithms and benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5630–5644, Dec. 2015.
[53] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 263–270.
[54] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 49–56.
[55] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 254–265.
[56] M. Danelljan, G. Haumlger, F. Shahbaz Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–5.
[57] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with Gaussian processes regression," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 188–203.
[58] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, Dec. 2015, pp. 621–629.
[59] 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.
[60] A. Lukežic, T. Vojír, L. C. Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4847–4856.
[61] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1401–1409.
[62] Z. Hong, C. Zhe, C. Wang, M. Xue, D. Prokhorov, and D. Tao, "MUlti-store tracker (MUSTer): A cognitive psychology inspired approach to object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 749–758.
[63] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1430–1438.
[64] S. Zhang, W. Lu, W. Xing, and L. Zhang, "Using fuzzy least squares support vector machine with metric learning for object tracking," *Pattern Recognit.*, vol. 84, pp. 112–125, Dec. 2018.
[65] D. Li, G. Wen, Y. Kuai, and F. Porikli, "Beyond feature integration: A coarse-to-fine framework for cascade correlation tracking," *Mach. Vis. Appl.*, vol. 5, pp. 1–10, Feb. 2019.

[66] X. Wang, Z. Hou, W. Yu, L. Pu, Z. Jin, and X. Qin, "Robust occlusion-aware part-based visual tracking with object scale adaptation," *Pattern Recognit.*, vol. 81, pp. 456–470, Sep. 2018.
[67] B. Jiang, Y. Zhang, J. Tang, B. Luo, and C. Li, "Robust visual tracking via laplacian regularized random walk ranking," *Neurocomputing*, vol. 339, pp. 139–148, Apr. 2019.
[68] M. Lee, T. Kim, Y. Ban, E. Song, and S. Leea, "Sampling operator to learn the scalable correlation filter for visual tracking," *IEEE Access*, vol. 7, pp. 11554–11546, 2019.
[69] T. Zhang, C. Xu, and M.-H. Yang, "Multi-task correlation particle filter for robust object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4335–4343.

CHENGGANG GUO received the M.S. degree in communication and information systems from the South China University of Technology, Guangzhou, China, in 2015. He is currently pursuing the Ph.D. degree in computer application technology with the University of Electronic Science and Technology of China, Chengdu, China. His current research interests include visual tracking and augmented reality.

DONGYI CHEN received the M.S. degree in computer and automation and the Ph.D. degree in electronic information engineering from Chongqing University, in 1985 and 1997, respectively. He completed his Postdoctoral Research at the Department of Electrical and Computer Engineering, University of Toronto, from 1997 to 1999. He was a Visiting Professor with the School of Computing, Georgia Institute of Technology, from 2002 to 2005. He is currently a Professor with the School of Automation Engineering, University of Electronic Science and Technology of China. His research interests include augmented reality, wearable computing, and wireless sensor networks.

ZHIQI HUANG received the Ph.D. degree in measuring and testing technologies and instruments from the University of Electronic Science and Technology of China, in 2009, where he is currently an Associate Professor. His current research interests include wearable computing, human–computer interaction, and reliability engineering.