Hierarchical Spatiotemporal Graph Regularized Discriminative Correlation Filter for Visual Object Tracking

Sajid Javed, Arif Mahmood, Jorge Dias, *Senior Member, IEEE*, Lakmal Seneviratne, *Senior Member, IEEE*, and Naoufel Werghi, *Senior Member, IEEE*

***Abstract*—Visual object tracking is a fundamental and challenging task in many high-level vision and robotics applications. It is typically formulated by estimating the target appearance model between consecutive frames. Discriminative correlation filters (DCFs) and their variants have achieved promising speed and accuracy for visual tracking in many challenging scenarios. However, because of the unwanted boundary effects and lack of geometric constraints, these methods suffer from performance degradation. In the current work, we propose hierarchical spatiotemporal graph-regularized correlation filters for robust object tracking. The target sample is decomposed into a large number of deep channels, which are then used to construct a spatial graph such that each graph node corresponds to a particular target location across all channels. Such a graph effectively captures the spatial structure of the target object. In order to capture the temporal structure of the target object, the information in the deep channels obtained from a temporal window is compressed using the principal component analysis, and then, a temporal graph is constructed such that each graph node corresponds to a particular target location in the temporal dimension. Both spatial and temporal graphs span different subspaces such that the target and the background become linearly separable. The learned correlation filter is constrained to act as an eigenvector of the Laplacian of these spatiotemporal graphs. We propose a novel objective function that incorporates these spatiotemporal constraints into the DCFs framework. We solve the objective function using alternating direction methods of multipliers such that each subproblem has a closed-form solution. We evaluate our proposed algorithm on six challenging benchmark datasets and compare it with 33 existing state-of-the art trackers. Our results demonstrate an excellent performance of the proposed algorithm compared to the existing trackers.**

***Index Terms*—Discriminative correlation filters (DCFs), graph regularization, visual object tracking (VOT).**

Manuscript received 23 September 2020; revised 18 March 2021; accepted 28 May 2021. Date of publication 7 July 2021; date of current version 17 October 2022. This work was supported by the Khalifa University of Science and Technology under Award RC1-2018-KUCARS. This article was recommended by Associate Editor W. Hu. *(Corresponding author: Sajid Javed.)*

Sajid Javed, Jorge Dias, Lakmal Seneviratne, and Naoufel Werghi are with the Department of Electrical and Computer Engineering, Khalifa University of Science and Technology, Abu Dhabi, UAE (e-mail: sajid.javed@ku.ac.ae; naoufel.werghi@ku.ac.ae).

Arif Mahmood is with the Department of Computer Science, Information Technology University, Lahore 25000, Pakistan.

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2021.3086194.

Digital Object Identifier 10.1109/TCYB.2021.3086194

I. Introduction

VISUAL object tracking (VOT) is one of the most fundamental tasks in vision and robotics having a wide range of applications across several domains, such as surveillance and security, autonomous driving, abnormality detection, medical imaging, and augmented reality [17], [19], [28]–[31], [33], [36], [70]. The major challenges encountered by VOT include significant scale and illumination variations, severe occlusion and background clutter, blurring because of fast motion, and deformation of the nonrigid targets [75]. To handle these challenges, numerous research directions have been investigated in recent years [10], [14], [24], [25], [34], [44], [57], [69], [74] and several review studies have also been presented [17], [32], [37], [38], [70]. Moreover, many challenging datasets have been proposed to facilitate evaluation and comparison of VOT methods [41], [42], [52], [60], [75], [76]. Despite a lot of research focus, VOT in challenging environments is still an open problem, which needs to be further investigated [17], [60].

Among the most popular tracking approaches, Discriminative Correlation Filters (DCFs) have attained significant attention because of their impressive performance in terms of speed and accuracy [17], [70]. In most of the DCF methods, an online correlation filter is trained from the region of interest in the current frame, which is then employed to track the target object in the subsequent frames by estimating the maximum response [3], [17]. Henriques *et al.* proposed kernelized correlation filter (KCF) which approximated a dense sampling scheme using a circulant matrix in which each row contains a circular shifted base sample [24]. The regression model was estimated in the Fourier domain with only a base sample; therefore, it achieved significant computational performance in both training and testing stages. Their method exploited a single channel kernel and enabled efficient learning and target detection between consecutive frames with fast Fourier transform (FFT). Galoogahi *et al.* [18] introduced multiple channels into the DCFs framework for more accurate VOT. However, the periodic assumption of the target training samples in these methods produces unwanted boundary effects leading to inaccurate image representation resulting in degraded VOT performance [10].

Several extensions of DCFs have been proposed to address the unwanted boundary effects problem [10], [39], [44], [59].

For example, Daneljan *et al.* [10] proposed a spatially regularized DCFs tracker to investigate this problem by proposing to learn filters from training examples with a large spatial support. Although VOT performance improved in many tracking scenarios, their method suffered from the computational complexity of the optimization function. Li *et al.* [44] addressed this lack by incorporating spatiotemporal regularization in the DCF objective function and solved the regression model using alternating direction methods of multipliers (ADMM). Galoogahi *et al.* [40] proposed a correlation filter to restrain the boundary effects and also proposed a background-aware correlation filter, which increases negative examples by sampling background patches around the target [39]. Muller *et al.* [59] proposed the context-aware DCFs method, which learns the filter by considering the contextual patches surrounding the target object, and achieves a good tradeoff between computational complexity and accuracy. Although, these approaches have reduced the boundary effects and produced encouraging results on many large-scale VOT datasets [42], most of these methods only focus on spatial dependency in every frame and update the correlation filter with a steady learning rate. Moreover, the constraints employed in these approaches are usually fixed for the target object and do not change during the tracking process; therefore, these approaches cannot fully exploit the diverse temporal appearance variations. Dai *et al.* [7] proposed adaptive spatially regularized correlation filters (ASRCF) to simultaneously optimize the filter coefficients and the spatial regularization weights. Huang *et al.* [27] proposed an aberrance repressed correlation filter (ARCF) by enforcing a restriction on the rate of alteration in response maps generated in the tracking phase. Li *et al.* [51] proposed an online and adaptive method to learn the spatiotemporal regularization term. The differences among recent DCFs-based VOT methods handling boundary effects are summarized in Table I in the supplementary material.

Inspired by the success of deep CNNs on a wide variety of visual-recognition tasks [64], several studies have also been proposed to incorporate deep features into the DCFs framework [11], [44], [57], [73]. For instance, Ma *et al.* [57] achieved improved accuracy by employing hierarchical CNN features with the DCFs. The DCFs are learned over the fine-, middle-, and coarse-level deep features to capture both spatial and semantic information. While inferring the target location, their method employed a coarse-to-fine search strategy on a multilevel response map. It has been observed that deep features-based DCF methods have outperformed the handcrafted features-based DCF methods on publicly available VOT datasets [17], [42]. Although feature-level fusion methods [10], [44], [57], [62] have been widely used to boost the VOT performance, the initial weights of coarse-level features are usually high resulting dominant role of semantic features, which may be justified because of more effectiveness of coarse-level information compared to fine level [57]. However, in these approaches, a transient drift may get amplified by the inadequate online update process. Therefore, the feature-level fusion approaches may fail to fully explore the true relationship of multilevel features [17], [70]. Also, relying on spatial feature fusion strategy limits the model diversity;

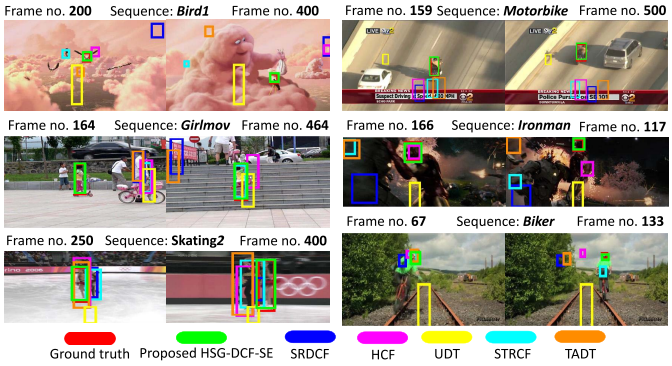

Fig. 1. Many existing tracking methods, including SRDCF [10], HCF [57], STRCF [44], UDT [72], and TADT [48], are not able to effectively handle VOT in the presence of challenging scenarios. For example, *Bird1* sequence suffers from motion blur, *Motorbike* and *Ironman* from illumination variations, *Girl2* from scale variation (SV) and out-of-plane rotation (OPR), *Skating2* from fast motion (FM) and OPR, and *Biker* from low resolution (LR) and OPR. These sequences are selected from the OTB100 dataset [75]. In contrast to the compared methods, the proposed hierarchical spatiotemporal graph regularized DCFs (HSG-DCF) with scale estimation (HSG-DCF-SE) using HOG features algorithm has better handled these challenges.

therefore, target appearance variations may not be appropriately handled in challenging tracking scenarios. Fig. 1 depicts instances of these limitations for the HCF method [57].

Manifold learning methods have also been employed to estimate the geometric and topological properties of the target object [20]. The spatial constraints preserve local structures while spatiotemporal constraints preserve global geometric structures embedded in high-dimensional spaces [35], [79], [85]. The spatial and temporal target structures may be considered as points on high-dimensional manifolds. It has been assumed that if two data samples are close in the intrinsic manifold of the data distribution, then the representations of these two points in a new space are also close to each other [22], which has often been achieved by using graph-based regularizations [1], [79]. This notion has also been employed for VOT in [26] and [53]. Inspired by these findings, we also propose graph-based regularization to improve the VOT performance by preserving local as well as global data structures embedded in high-dimensional manifolds.

In the current work, we address the aforementioned challenges by proposing the HSG-DCF tracker. In the proposed algorithm, a target object is represented using hierarchical deep features, and at each level of the hierarchy, a DCF is trained by jointly minimizing the sum of least squares loss with structural constraints enforced by spatial and temporal graphs. The spatial graph encodes local appearance variations of the target in the current frame while the temporal graph encodes global appearance variations over a temporal window. For this purpose, we propose the spatial graph to be constructed using deep features across different spatial components of the target object to capture target spatial structure such that the neighborhood connections are preserved, thus resulting in a more discriminative model. In order to capture temporal appearance variations of the target, we propose a temporal graph to be constructed using compressed information capturing variations of different target appearances in a temporal window. The temporal constraint preserves the relationship between global target

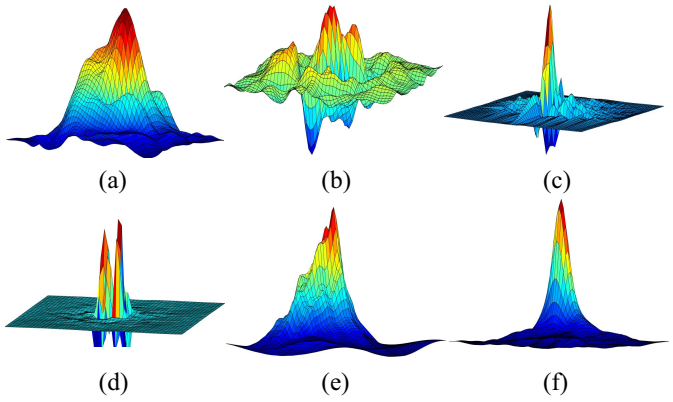

Fig. 2. Correlation filter response of the existing tracking methods, including HCF [57], CACF-MOSSE [59], SRDCF [10], STRCF [44], BACF [39], and the proposed HSG-DCF-SE on frame no. 117, *Ironman* sequence (Fig. 1). The correlation filter responses of compared methods show multiple peaks and lack of localization resulting in degraded VOT performance. Compared to these methods, the response map of the proposed HSG-DCF algorithm has a single peak with better localization resulting in improved VOT performance.

appearance at different time instances, which further enhances the tracking ability of the target object.

The resulting DCF is constrained to be aware of both the spatial and temporal target structure by enforcing it to be the eigenvectors of the spatiotemporal Laplacian matrices computed from both spatial and temporal graphs. Indeed, the eigenvectors of the Laplacian matrices contain the structure captured by the corresponding graphs. Therefore, enforcing a DCF to be the eigenvectors of these matrices ensures that the DCF will capture the target structure. As an example, the eigenvector corresponding to the minimum nonzero eigenvalue, also known as the Fiedler vector, defines two partitions of the graph based on the signs of its coefficients [58]. By incorporating these spectral clustering-based constraints into the DCF framework, the correlation filters are enabled to be aware of the target structure both in the spatial and the temporal domain. By encoding the target spatial and temporal structure, our tracker is able to better discriminate the target object from distractors as well as the background and thus, improving the tracking performance. We solve the proposed objective function using the ADMM method [4], because of its computational efficiency.

To explicitly handle SVs of the target object, we employ the simple strategy based on HOG features to estimate the target scale. The maximum filter response is estimated across a scale range, which is then implicitly refined by using three deep feature levels in the detection step. The use of HOG features has ensured low-computational complexity of the scale estimation (SE) step. The proposed algorithm with the SE step is referred as HSG-DCF-SE.

Despite the DCFs-based tracking being explored in numerous dimensions, the structure of the target object has not been fully exploited to obtain structure-aware correlation filters. More specifically, we propose DCFs that are consistent with the spatial and temporal structure of the target object. We define the target structure by capturing the relationship among different hierarchical features of the target both in spatial and temporal domains. The structural correlation filters proposed by Liu *et al.* [54] are the closet work to ours. In their approach, they considered dividing a target object into a set of patches and estimated a different DCF for each patch [54]. In contrast, we propose to capture the similarity of different target components by using two graphs. To the best of our knowledge, graph-based structure-aware DCFs have not been proposed before us.

The proposed HSG-DCF-SE algorithm is able to robustly track target in the presence of many challenging scenarios. For example, if the target appearance changes rapidly, it is difficult to handle using existing trackers (*Bird1*, *Ironman*, and *Biker* sequences in Fig. 1). A comparison of the DCF visualization is shown in Fig. 2 for the sequence *Ironman* selected from the OTB100 dataset [75]. Most of the response maps have a low signal-to-noise ratio where the signal is the maximum peak and the noise is the second highest peak. For the proposed algorithm, the signal-to-noise ratio is significantly higher than the compared methods. Also, the shape of the response map suggests a quick convergence to the optimal value, better localization, and less chances of the algorithm to be stuck in local maxima. The experimental evaluations on seven benchmark tracking datasets demonstrate an excellent performance of the proposed algorithm compared to the 33 existing state-of-the-art trackers. The main contributions of the current work are as follows.

1) We enable DCFs to capture the spatial target structure by integrating graph-based regularization into the DCFs framework.
2) We extend our algorithm to make the DCFs temporal target structure aware by extending the structural constraint in the temporal dimension. For this purpose, we compress the deep features in the temporal window using principal component analysis (PCA) and construct a graph capturing different temporal variations across target components. To the best of our knowledge, such graph-based spatiotemporal constraints have not been investigated before in the DCFs framework for VOT.
3) We propose a novel objective function that encodes spatiotemporal structural constraints into a DCFs optimization model. We jointly optimize the DCFs constraints and structural regularization using the ADMM method in a computationally efficient manner.
4) We performed extensive evaluations using seven publicly available tracking benchmark datasets, and we compared our algorithm with 33 existing state-of-the-art methods, and provide rigorous analysis of the results.

The remainder of this article is organized as follows. In Section II, we review related work. In Section III, we describe the HSG-DCF-SE algorithm in detail. The experimental results are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. Related Work

Over the past decade, VOT has remained an active research area, and numerous methods have been proposed [17], [23], [46], [47], [53]. The main focus of the current work is to

improve the VOT performance using the DCF framework; therefore, in this section, we mainly discuss the DCF-based methods. Interested readers may explore more details about the other tracking methods in recent studies [70]. We broadly categorize the DCFs-based VOT methods into classical DCFs [3], [24], [25], deep features-based DCFs [8], [12], [57], [62], and structural regularized DCFs [6], [10], [44], [54], [59].

Classical DCFs Methods: The classical DCF methods have been widely employed for VOT because of low computational cost and good performance. Bolme *et al.* [3] proposed MOSSE filters by minimizing the error between the actual and the desired correlation output on a set of gray-scale patches. By using circular correlation, the resulting filter was efficiently computed using FFTs and pointwise operations. Henriques *et al.* [24] extended the MOSSE filters and proposed the KCF method using HOG descriptors, and Zhang *et al.* [80] further improved the KCF for VOT. These tracking methods were limited to determine the target location, and observed degraded performance in the presence of SVs and target rotation. Therefore, lot of efforts have been put to address these issues using multidimensional features [13], context learning [82], scale estimation [9], and efficient filter mining [8]. Rout *et al.* [63] trained different orientation-specific filters using rotated target patches to address the orientation variations. Li and Zhu [49] proposed a scale adaptive feature fusion scheme to handle the fixed size template problem. Danelljan *et al.* [13] developed adaptive multiscale correlation filters using color attributes by mapping multichannel features into a Gaussian kernel space. Zhang *et al.* [82] modeled SVs using consecutive correlation responses by incorporating context information into a DCF framework. Zhu *et al.* [86] proposed a collaborative DCF method that combines multiscale KCF to handle SV using an online filter.

Most of these trackers aim an adaptive model and do not utilize long-term target appearance variations. As a result, these models are prone to target drift in the presence of occlusion and target disappearance. Moreover, these approaches are also unable to recover from tracking failures [70]. To address these limitations, Hong *et al.* [25] proposed a biology-inspired approach employing a set of cooperating long-term and short-term trackers. Ma *et al.* [56] also proposed a long-term tracker using an online random fern classifier to address these problems. Moreover, hierarchical spatiotemporal context-aware DCFs are also proposed for efficient VOT [74].

Deep Features-Based DCFs Methods: Many researchers have used deep feature representations for improving the VOT performance due to their robustness against photometric and geometric variations [8], [12], [57], [62]. For instance, DCF trackers show state-of-the-art performance when deep convolution features are used [12]. Mostly, pretrained deep networks are employed to obtain deep features of the target and the search space, and they are also used to develop scale-invariant VOT methods. Danelljan *et al.* [11] extended the spatially regularized correlation filter to use deep convolution features. They also proposed continuous convolution filters for tracking with multiscale deep features to account for appearance variation [12]. Ma *et al.* [57] estimated the position of the target by fusing the response maps obtained from the deep convolution features of various resolutions in a coarse-to-fine scheme. Qi *et al.* [62] tracked the target by employing an adaptive hedge method on the response maps obtained from deep features. Liu *et al.* [44] also incorporated deep features in the spatiotemporal DCFs. However, even though each correlation filter works fast, deep features have large dimensions to be handled in real time. Furthermore, to recognize scale changes of the target, correlation filter-based methods need to train scalewise filters or apply the same filter repeatedly, leading, thus, to a significant increase of the computational complexity. Valmadre *et al.* [68] have obtained computational efficiency by using end-to-end lightweight architectures. They proposed to implement DCF as a differentiable layer in a deep neural network enabling deep features tightly coupled with correlation filters.

Structural Regularized DCFs Methods: The circular correlation employed by classical DCF assumes the periodic target appearance model in both training and detection stages. This assumption results in unwanted boundary effects, which leads to an inaccurate target description that degrades the VOT performance [10], [70]. To address this problem, Liu *et al.* [54] proposed the part-based tracking method, which is robust against partial occlusion (POC) and better preserves the target structure. Li *et al.* [50] presented a method based on target patch reliability of being tracked and exploited the patch trajectories for VOT. Danelljan *et al.* [10] proposed spatial regularization in the DCF framework to penalize the filter coefficients in the background regions. Choi *et al.* [6] exploited spatial attention to weight the filter coefficients to handle undesired boundary effects. Han *et al.* [21] proposed a target state-aware correlation filter for improved VOT performance. The context-aware and temporal regularized DCFs are also proposed in [44] and [59], and recent improvements can be explored in [66] and [77].

These approaches have achieved encouraging VOT performance by enforcing either spatial or temporal structural constraints; however, these constraints result in a computational burden on the optimization models. Wang *et al.* [74] attempted to obtain real-time performance by exploiting spatiotemporal constraints; however, the performance is degraded in complex scenes due to the weak target features representation. In contrast to the aforementioned DCF methods, we propose the spatiotemporal structural regularized DCFs algorithm by incorporating graph-based constraints into the DCFs objective function. Our proposed algorithm is different from the previous methods [7], [27], [51], [66], [74], [77] because we consider the similarity relationship among the deep features extracted from different target components both in the spatial and temporal domain as shown in Table I in the supplementary material.

III. Proposed Methodology

The block diagram and notations of the proposed HSG-DCF algorithm are shown in Fig. 3 and Table I. Our proposed algorithm consists of three main steps: 1) deep features extraction; 2) the construction of spatiotemporal graphs; and 3) DCF

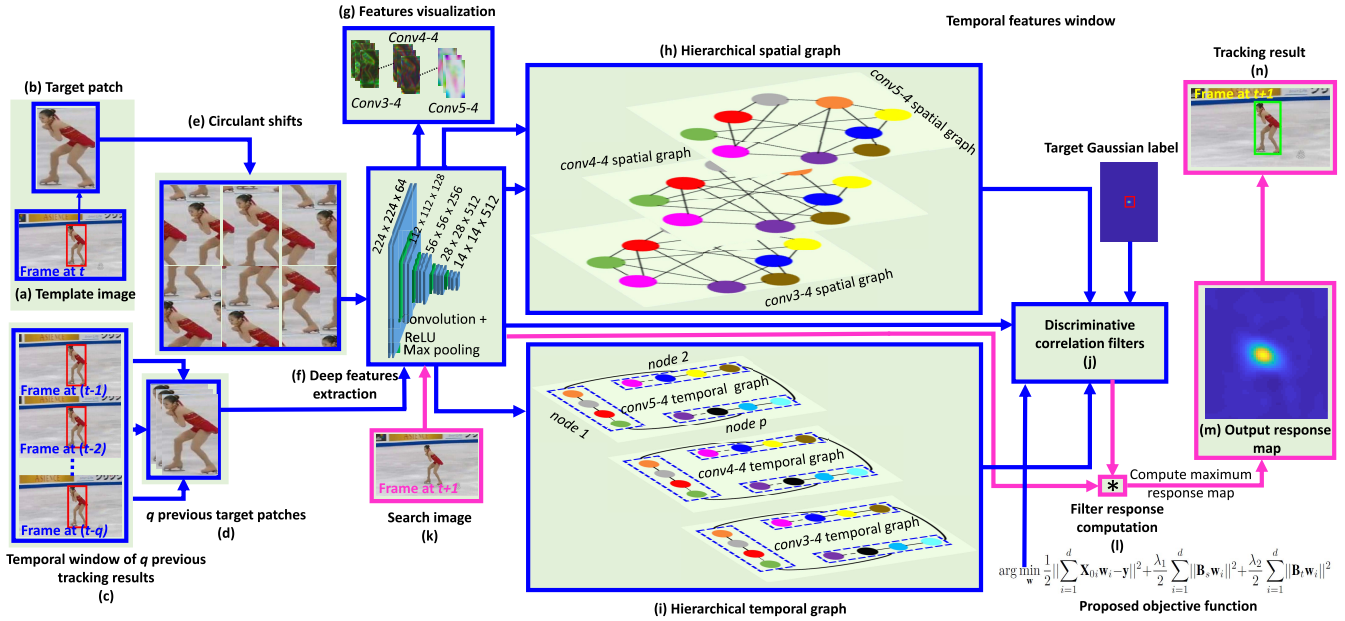

Fig. 3. Block diagram of the proposed HSG-DCF tracking algorithm. Steps (a) and (b) show the template image and cropped target patch, steps (c) and (d) show the corresponding temporal tracking window of q previous tracking observations with target patches, step (e) shows the circulant shifts of the target patch in (b), steps (f) and (g) show the deep features extraction and their visualizations, and step (h) shows the construction of spatial graphs for each deep feature hierarchy. The nodes in each spatial graph correspond to target spatial locations. Step (i) shows the construction of temporal graphs for each level of hierarchy. The nodes in each temporal graph correspond to the target spatiotemporal locations. Step (j) shows the proposed objective function where spatial and temporal graph-based regularizations are encoded. Steps (k)–(n) in pink color show the target detection on the test image, where step (k) shows a search image, step (l) is the computation of three filter response maps using convolution operator $*$, step (m) is the selection of maximum response map, and step (n) is the tracking output.

TABLE I
DESCRIPTION OF IMPORTANT SYMBOLS USED IN HSG-DCF

Symbols	Description
m, n	Height and width of the target object.
p	Size of feature vector ($p = m \times n$).
d	Number of channels in the l-th layer.
σ_s	Smoothing parameter of $\mathbf{G}_s^l$.
q	Number of previous tracking observation.
h	Number of nearest neighbors.
k	ADMM iterations index.
λ_1, λ_2	Hyper-parameters in Eq. (2).
$\mathbf{y} \in \mathbb{R}^p$	Groundtruth Gaussian response.
$\mathbf{w}_i \in \mathbb{R}^p$	Correlation filter for the i-th channel.
$\mathbf{A} \in \mathbb{R}^{m \times n}$	Target object region of interest.
$\mathbf{X} \in \mathbb{R}^{p \times d}$	Input feature matrix.
$\mathbf{M}_l \in \mathbb{R}^{p \times d \times q}$	Temporal features matrix at the l-th layer.
$\mathbf{X}_{0i} \in \mathbb{R}^{p \times p \times d}$	Circulant shifted version of $\mathbf{X}$ for the i-th channel.
$\mathbf{G}_s^l, \mathbf{G}_t^l$	Spatial and temporal graphs of the l-th layer.
$\mathbf{B}_s, \mathbf{B}_t \in \mathbb{R}^{p \times p}$	Matrices of encoding spatial and temporal graphs structure in Eq. (2).
$\mathbf{A}_s^l, \mathbf{A}_t^l \in \mathbb{R}^{p \times p}$	Spatial and temporal adjacency matrices of the $\mathbf{G}_s^l$ and $\mathbf{G}_t^l$.
$\mathbf{V}_s^l(i), \mathbf{V}_t^l(i) \in \mathbb{R}^d$	Spatial and temporal vertices of the $\mathbf{G}_s^l$, $\mathbf{G}_t^l$.
$\mathbf{S}_l, \mathbf{T}_l \in \mathbb{R}^{p \times p}$	Spatial and temporal graph-Laplacian matrices of $\mathbf{G}_s^l$ and $\mathbf{G}_s^l$.

objective function minimization. In the following sections, we explain each step of the proposed algorithm in detail.

A. Deep Features Extraction

Given the target object location in the first frame, we crop the region of interest $\mathbf{A} \in \mathbb{R}^{m \times n}$, where m and n denote the height and width of the target object. Similar to Ma *et al.* [57], using VGG-19 as features extractor [65], we extract deep features from the last three layers, including: 1) *conv3-4*; 2) *conv4-4*; and 3) *conv5-5* for the target object. We create our feature matrices $\mathbf{X}_l \in \mathbb{R}^{p \times d}$, where $p = m \times n$ and d is the number of channels in the lth layer of the VGG-19. In VOT, the target object may suffer from large appearance variations; therefore, the features using the output of *conv5-4* are able to discriminate the target even when it undergoes severe background changes while the features using the output of *conv4-4* and *conv3-4* encoding more spatial details are useful to localize the target.

B. Proposed HSG-DCF Model

The DCFs learn discriminative patterns of the target object and estimate its position in the subsequent frames by searching maximum correlation response. DCFs allow for dense sampling around the target at a very low computational cost, which is achieved by using all possible translations of the target within a search window as circulant shifts to form a data matrix $\mathbf{X}_0 \in \mathbb{R}^{p \times p \times d}$, where p is the size of the target patch. The circulant structure of this matrix facilitates a very efficient solution to the ridge regression problem in the Fourier domain [24]. The multichannel correlation filter for a particular layer can then be formulated as follows [18]:

$$\arg\min_{\mathbf{w}} \left\| \sum_{i=1}^{d} \mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y} \right\|^2 + \lambda_1 \sum_{i=1}^{d} \|\mathbf{w}_i\|^2 \tag{1}$$

where $\mathbf{w}_i \in \mathbb{R}^p$ is the correlation filter for the ith channel, $\mathbf{X}_{0i}$ contains all circulant shifts of the ith channel of the feature map, $\mathbf{y} \in \mathbb{R}^p$ is the vectorized Gaussian response, and λ_1 is the regularization parameter.

The DCFs formulation given by (1) observes undesirable boundary effects because of the periodic assumption of the target patch [10]. In addition, the minimization of the objective

function given by (1) has been considered as the minimization of $p \times d$ independent problems where each problem only minimizes a particular correlation filter coefficient [44]. However, different filter coefficients are not independent of each other because each correlation filter encodes a particular target structure. Therefore, one may preserve the relationships among different filter coefficients by enforcing intrinsic constraints based on the target object structure. The target structure can be preserved in the spatial domain considering spatial appearance variations as well as in the temporal domain by considering the temporal appearance variations of the target. Therefore, we propose to incorporate new constraints to enforce the spatiotemporal structure of the target within the DCFs framework to improve the VOT performance by alleviating the undesirable boundary effects. Our proposed objective function is then formulated as follows:

$$\arg\min_{\mathbf{w}} \frac{1}{2}\left\|\sum_{i=1}^{d}\mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y}\right\|^2 + \frac{\lambda_1}{2}\sum_{i=1}^{d}\|\mathbf{B}_s\mathbf{w}_i\|^2 + \frac{\lambda_2}{2}\sum_{i=1}^{d}\|\mathbf{B}_t\mathbf{w}_i\|^2 \tag{2}$$

where $\mathbf{B}_s$ encodes the spatial target structure and $\mathbf{B}_t$ encodes the temporal target appearance variations. λ_1 and λ_2 assign relative importance to the different terms in the regression model. Both matrices $\mathbf{B}_s$ and $\mathbf{B}_t$ are estimated from spatial and temporal graphs, and are further explained in the following sections.

C. Hierarchical Spatial Appearance Regularization

The hierarchical deep features are computed from a deep neural network and for each level of the hierarchy, a different spatial graph $\mathbf{G}_s^l$ is computed capturing the target appearance variations at that level.

Let $\mathbf{G}_s^l = (\mathbf{V}_s^l, \mathbf{A}_s^l)$ be an undirected weighted spatial graph at lth layer of the hierarchy, where $\mathbf{V}_s^l$ and $\mathbf{A}_s^l$ contain the vertices and the edge weighted adjacency matrix of the graph. Each vertex $\mathbf{V}_s^l(i) \in \mathbb{R}^d$ contains feature values across d channels corresponding to the ith target location and represented as a column vector in features matrix $\mathbf{X}_l$.

The motivation of the spatial appearance constraint comes from the observation that DCF preserves the target structure on the Riemann manifold [79]. That is, if two vertices $\mathbf{V}_s^l(i)$ and $\mathbf{V}_s^l(j)$ are close on the data manifold, then their corresponding coefficients in $\mathbf{w}_i$ should also be close. Here, we consider spatial closeness among the feature maps, encoded, in the graph $\mathbf{G}_s^l$ using the h-nearest neighbor strategy [61]. The first step involves searching for the closest neighbors for all the columns in the features matrix $\mathbf{X}_l$ based on the Euclidean distance, where each vertex is connected to its h-nearest neighbors, so that if $\mathbf{V}_s^l(i)$ and $\mathbf{V}_s^l(j)$ are in the h-nearest neighbors of each other, we set

$$A_s^l(i,j) = \exp\left(-\frac{\left\|\mathbf{V}_s^l(i) - \mathbf{V}_s^l(j)\right\|_2^2}{2\sigma_s^2}\right) \tag{3}$$

where σ_s is a normalizing parameter set as the average distance among the vertices in $\mathbf{G}_s^l$. If two vertices $\mathbf{V}_s^l(i)$ and $\mathbf{V}_s^l(j)$ are connected, then $A_s^l(i,j) > 0$, otherwise, $A_s^l(i,j) = 0$. Based on the weighted adjacency matrix $\mathbf{A}_s^l$, we compute the normalized spatial Laplacian matrix $\mathbf{S}_l$ of the graph $\mathbf{G}_s^l$ by

$$\mathbf{S}_l = \mathbf{I} - \mathbf{D}_s^{-\frac{1}{2}}\mathbf{A}_s^l\mathbf{D}_s^{-\frac{1}{2}} \tag{4}$$

where $\mathbf{I}$ is a $p \times p$ identity matrix and $\mathbf{D}_s$ is a $p \times p$ spatial degree matrix with its ith diagonal element being equal to the sum of the ith row of $\mathbf{A}_s^l$ (i.e., $\sum_j A_s^l(i,j)$) and all nondiagonal values are zero. The spatial Laplacian matrix $\mathbf{S}_l$ encodes the spatial structure of the target object. The eigenvectors of $\mathbf{S}_l$ act as cluster indicators in the graph $\mathbf{G}_s^l$. To encode this information in the correlation filter, we enforce $\mathbf{w}_i$ to act as the eigenvector of $\mathbf{S}_l$. To this end, we minimize the generalized eigenvalue problem $\mathbf{w}_i^\top \mathbf{S}_l \mathbf{w}_i$, which is independently minimized for each channel

$$\Theta_s^l = \sum_{i=1}^{d}\mathbf{w}_i^\top \mathbf{S}_l \mathbf{w}_i. \tag{5}$$

The normalized Laplacian matrix $\mathbf{S}_l$ can be symmetrically decomposed as

$$\mathbf{S}_l = \mathbf{Y}\Sigma\mathbf{Y}^\top = \left(\Sigma^{\frac{1}{2}}\mathbf{Y}^\top\right)^\top \Sigma^{\frac{1}{2}}\mathbf{Y}^\top = \mathbf{B}_s^{l\top}\mathbf{B}_s^l \tag{6}$$

where $\mathbf{Y}$ is a $p \times p$ orthonormal matrix with each column being an eigenvector of $\mathbf{S}_l$, and Σ is a $p \times p$ diagonal matrix with its diagonal element Σ_{ii} being a singular value of $\mathbf{S}_l$ (sorted as $0 \le \Sigma_{ii} \le \cdots \Sigma_{pp}$). The matrix $\mathbf{B}_s^l = \Sigma^{(1/2)}\mathbf{Y}^\top$ is computed using all eigenvectors of $\mathbf{S}_l$. It may be considered as a combination of scaled basis of the graph-Laplacian matrix $\mathbf{S}_l$ and thus, defining a manifold structure of the target object. Substituting $\mathbf{S}_l = \mathbf{B}_s^{l\top}\mathbf{B}_s^l$ in (5), we obtain

$$\Theta_s^l = \sum_{i=1}^{d}\mathbf{w}_i^\top \mathbf{B}_s^{l\top}\mathbf{B}_s^l\mathbf{w}_i = \sum_{i=1}^{d}\left\|\mathbf{B}_s^l\mathbf{w}_i\right\|^2. \tag{7}$$

The spatial structural constraint above can be interpreted as enforcing the correlation filter $\mathbf{w}_i$ in each channel to be orthogonal to the eigenvectors of $\mathbf{S}_l$. Assuming that the manifold spanned by the background patches will be different from the manifold spanned by the target object, therefore, such DCFs will be able to better discriminate the target object from its background, resulting in the improvement of VOT performance.

D. Hierarchical Temporal Appearance Regularization

The temporal appearance variations of the target object are often different from the background region and may be exploited to improve the VOT performance. Therefore, we propose to incorporate target temporal appearance variations into our proposed DCFs objective function by using a graph constructed over a temporal window of q previous tracking observations. The corresponding deep features are computed and a features matrix $\mathbf{M}_l \in \mathbb{R}^{p\times d\times q}$ is created. The feature matrix $\mathbf{M}_l$ is rearranged as a 2-D matrix $\mathbf{M}_l \in \mathbb{R}^{p\times(dq)}$, where each column of size dq is a spatiotemporal feature corresponding to a particular target location. In order to reduce

the computational complexity of the temporal graph construction, we employ PCA to reduce the dimensionality of spatiotemporal features [66].

Considering the hierarchy of deep features at each level, a different temporal graph $\mathbf{G}_t^l$ is computed capturing the target temporal appearance variations at that level. Let $\mathbf{G}_t^l = (\mathbf{V}_t^l, \mathbf{A}_t^l)$ be an undirected weighted temporal graph at the lth layer of the hierarchy. Each vertex $\mathbf{V}_t^l(i)$ corresponds to a spatiotemporal feature contained as a column vector in the feature matrix $\mathbf{M}_l$.

Similar to the spatial graph, we compute Euclidean distances among the columns of $\mathbf{M}_l$ and consider k-nearest neighbors for the construction of temporal graph. Based on the estimated temporal adjacency matrix $\mathbf{A}_t^l$ [similar to (3)], we compute the normalized temporal Laplacian matrix $\mathbf{T}_l$ of the graph $\mathbf{G}_t^l$ as $\mathbf{T}_l = \mathbf{I} - \mathbf{D}_t^{-\frac{1}{2}} \mathbf{A}_t^l \mathbf{D}_t^{-\frac{1}{2}}$, where $\mathbf{D}_t$ is a $p \times p$ temporal degree matrix. Since the temporal Laplacian matrix $\mathbf{T}_l$ encodes the temporal structure of the target object, its eigenvectors encode the structure of the target object based on temporal appearance variations. The temporal Laplacian matrix can also be decomposed using SVD as $\mathbf{T}_l = \mathbf{Y}\Sigma\mathbf{Y}^\top = (\Sigma^{(1/2)}\mathbf{Y}^\top)^\top \Sigma^{\frac{1}{2}}\mathbf{Y}^\top = \mathbf{B}_t^{l\top}\mathbf{B}_t^l$. Similar to spatial structural constraints, the temporal appearance constraint is then given as follows:

$$\Theta_t^l = \sum_{i=1}^{d} \mathbf{w}_i^\top \mathbf{B}_t^{l\top} \mathbf{B}_t^l \mathbf{w}_i = \sum_{i=1}^{d} \left\| \mathbf{B}_t^l \mathbf{w}_i \right\|^2 \tag{8}$$

where $\mathbf{B}_t^l$ is a basis of the temporal manifold containing the target object variations. The temporal appearance constraints enforce the correlation filter to be orthogonal to the manifold basis. Thus, discriminating the target temporal variations from the background temporal variations.

E. Objective Function Minimization

We optimize the HSG-DCF model (2) using the ADMM method by solving one variable and fixing others [4]. We first introduce two auxiliary variables as a spatial filter $\mathbf{g}^s = \mathbf{w}$ and a temporal filter $\mathbf{g}^t = \mathbf{w}$ to make the objective function separable. The constrained optimization problem is then formulated as follows:

$$\arg\min_{\mathbf{w}} \left\| \sum_{i=1}^{d} \mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y} \right\|^2 + \frac{\lambda_1}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_s \mathbf{g}_i^s \right\|^2 + \frac{\lambda_2}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_t \mathbf{g}_i^t \right\|^2. \tag{9}$$

The Lagrangian form of model (9) is then formulated as follows:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathbf{g}^s, \mathbf{s}, \mathbf{g}^t, \mathbf{r}) = {} & \frac{1}{2} \left\| \sum_{i=1}^{d} \mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y} \right\|^2 + \frac{\lambda_1}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_s \mathbf{g}_i^s \right\|^2 \\ & + \frac{\lambda_2}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_t \mathbf{g}_i^t \right\|^2 + \sum_{i=1}^{d} \left(\mathbf{w}_i - \mathbf{g}_i^s\right)^\top \mathbf{s}_i \\ & + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^s \right\|^2 \\ & + \sum_{i=1}^{d} \left(\mathbf{w}_i - \mathbf{g}_i^t\right)^\top \mathbf{r}_i + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^t \right\|^2 \end{aligned} \tag{10}$$

where $\mathbf{s}$ and $\mathbf{r}$ are the Lagrangian multipliers and γ is a penalty factor. Putting $\mathbf{h} = (1/\gamma)\mathbf{s}$ and $\mathbf{m} = (1/\gamma)\mathbf{r}$, the above equation can be written as

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathbf{g}^s, \mathbf{s}, \mathbf{g}^t, \mathbf{r}) = {} & \frac{1}{2} \left\| \sum_{i=1}^{d} \mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y} \right\|^2 + \frac{\lambda_1}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_s \mathbf{g}_i^s \right\|^2 \\ & + \frac{\lambda_2}{2} \sum_{i=1}^{d} \left\| \mathbf{B}_t \mathbf{g}_i^t \right\|^2 + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^s + \mathbf{h}_i \right\|^2 \\ & + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^t + \mathbf{m}_i \right\|^2. \end{aligned} \tag{11}$$

Then, each subproblem $\mathbf{w}$, $\mathbf{g}^s$, $\mathbf{g}^t$, $\mathbf{m}$, and $\mathbf{h}$ can be solved efficiently using ADMM.

Solving Subproblem $\mathbf{w}$: By fixing other variables in (11) excluding $\mathbf{w}$, the subproblem $\mathbf{w}^{(k+1)}$ at the $(k+1)$th iteration can be written as

$$\begin{aligned} \mathbf{w}^{(k+1)} = {} & \arg\min_{\mathbf{w}} \frac{1}{2} \left\| \sum_{i=1}^{d} \mathbf{X}_{0i}\mathbf{w}_i - \mathbf{y} \right\|^2 \\ & + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^s + \mathbf{h}_i \right\|^2 \\ & + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \mathbf{w}_i - \mathbf{g}_i^t + \mathbf{m}_i \right\|^2. \end{aligned} \tag{12}$$

Using Parseval's theorem, the above equation can be rewritten in the Fourier domain as

$$\begin{aligned} & \operatorname*{argmin}_{\widehat{\mathbf{w}}} \frac{1}{2} \left\| \sum_{i=1}^{d} \widehat{\mathbf{x}}_i \odot \widehat{\mathbf{w}}_i - \widehat{\mathbf{y}} \right\|^2 + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \widehat{\mathbf{w}}_i - \widehat{\mathbf{g}}_i^s + \widehat{\mathbf{h}}_i \right\|^2 \\ & \quad + \frac{\gamma}{2} \sum_{i=1}^{d} \left\| \widehat{\mathbf{w}}_i - \widehat{\mathbf{g}}_i^t + \widehat{\mathbf{m}}_i \right\|^2 \end{aligned} \tag{13}$$

where $\mathbf{x}_i$ is the ith column vector of deep features matrix $\mathbf{X}_l$ and $\widehat{\mathbf{w}}$ denotes the DFT of the filter $\mathbf{w}$. From the above equation, we can see that the jth element of the label $\widehat{\mathbf{y}}$ only depends on the jth element of the filter $\widehat{\mathbf{w}}$ and sample $\widehat{\mathbf{x}}$ across all d channels. Therefore, it can be further decomposed into p subproblems. Let $\mathbf{x}^j, \mathbf{w}^j, \mathbf{g}^{sj}, \mathbf{g}^{tj}, \mathbf{h}^j, \mathbf{m}^j \in \mathbb{R}^d$ be the vectors consisting of the jth elements of $\mathbf{x}$, $\mathbf{w}$, $\mathbf{g}^s$, $\mathbf{g}^t$,$\mathbf{h}$, and $\mathbf{m}$ along all d channels. Each subproblem is given by

$$\begin{aligned} & \min_{\widehat{\mathbf{w}}^j} \left\| \widehat{\mathbf{x}}^{j\top} \widehat{\mathbf{w}}^j - \widehat{\mathbf{y}}(j) \right\|^2 + \gamma \left\| \widehat{\mathbf{w}}^j - \widehat{\mathbf{g}}^{sj} + \widehat{\mathbf{h}}^j \right\|^2 \\ & \quad + \gamma \left\| \widehat{\mathbf{w}}^j - \widehat{\mathbf{g}}^{tj} + \widehat{\mathbf{m}}^j \right\|^2. \end{aligned} \tag{14}$$

By taking the derivative with respect to $\widehat{\mathbf{w}}^j$ and setting it zero, we can obtain a closed-form solution

$$\begin{aligned} \widehat{\mathbf{w}}^j &= \left(\widehat{\mathbf{x}}^j \widehat{\mathbf{x}}^{j\top} + 2\gamma \mathbf{I} \right)^{-1} \alpha \\ \alpha &= \widehat{\mathbf{x}}^j \widehat{\mathbf{y}}(j) + \gamma \left(\widehat{\mathbf{g}}^{sj} - \widehat{\mathbf{h}}^j + \widehat{\mathbf{g}}^{tj} - \widehat{\mathbf{m}}^j \right). \end{aligned} \tag{15}$$

Algorithm 1: Pseudocode of HSG-DCF Tracker

Input: Video with region of interest $\mathbf{A} \in \mathbb{R}^{m\times n}$.
Initialization: Input features matrix $\mathbf{X}_l \in \mathbb{R}^{p\times d}$, q tracking temporal window, features matrix $\mathbf{M}_l \in \mathbb{R}^{p\times d\times q}$, λ_1, λ_2, $\gamma = 10$, $\gamma_{\max} = 100$, $\rho = 1.2$, $\mathbf{w}^0 = 0$, $\mathbf{r}^0 = 0$, and $\mathbf{s}^0 = 0$.
Compute $\mathbf{B}_s$ and $\mathbf{B}_t \in \mathbb{R}^{p\times p}$ using Eqs. (3)-(6).
while *not converged* ($k = 0, 1, ..$) **do**
1. Compute $\mathbf{w}^{k+1}$ using (16).
2. Compute $\mathbf{g}^{s(k+1)}$ using (18)
3. Compute $\mathbf{g}^{t(k+1)}$ using (20).
5. Update $\mathbf{h}^{(k+1)}$ using (21).
7. Update $\mathbf{m}^{(k+1)}$ using (21).
8. Update $\gamma^{(k+1)}$ using (21)

end
Output: $\mathbf{w}, \mathbf{g}^s, \mathbf{g}^t$
Use $\mathbf{w}$ in (22) to get each layer response map.
Find maximum across all maps for target localization.

Since $\widehat{\mathbf{x}}^j\widehat{\mathbf{x}}^{j\top}$ is a rank-1 matrix, (15) can be solved more efficiently using the Sherman–Morrison formula [44]. We have

$$\widehat{\mathbf{w}}^j = \frac{1}{2\gamma}\left(\mathbf{I} - \frac{\widehat{\mathbf{x}}^j\widehat{\mathbf{x}}^{j\top}}{2\gamma + \widehat{\mathbf{x}}^{j\top}\widehat{\mathbf{x}}^j}\right)\alpha. \tag{16}$$

Note that (16) only contains the vector multiply-add operation and, thus, can be computed efficiently. The filter $\mathbf{w}$ can then be obtained by the inverse DFT of $\widehat{\mathbf{w}}$.

Solving Subproblem g^s: In (11), fixing other variables excluding $\mathbf{g}^s$, the subproblem $\mathbf{g}^{s(k+1)}$ at the $(k+1)$th iteration can be written as

$$\mathbf{g}^{s(k+1)} = \underset{\mathbf{g}^s}{\operatorname{argmin}} \frac{\lambda_1}{2}\sum_{i=1}^{d}\left\|\mathbf{B}_s\mathbf{g}_i^s\right\|^2 + \frac{\gamma}{2}\sum_{i=1}^{d}\left\|\mathbf{w}_i - \mathbf{g}_i^s + \mathbf{h}_i\right\|^2. \tag{17}$$

By taking the derivative and setting it to zero, each element of $\mathbf{g}^s$ can be computed independently, and thus, the closed-form solution of $\mathbf{g}^s$ can be computed by

$$\mathbf{g}^{s(k+1)} = \left(\tilde{\mathbf{B}}_s\tilde{\mathbf{B}}_s^\top + \lambda_1\mathbf{I}\right)^{-1}\left(\gamma\tilde{\mathbf{w}} + \gamma\tilde{\mathbf{h}}\right) \tag{18}$$

where $\tilde{\mathbf{B}}_s$ represents the $dp \times dp$ diagonal matrix concatenated with d diagonal matrices Diag($\mathbf{B}_s$). The vectors $\tilde{\mathbf{w}}$ and $\tilde{\mathbf{h}}$ denote the concatenated vectors of $\mathbf{w}_i$ and $\mathbf{h}_i$ accross d-channels.

Solving Subproblem g^t: In (11), fixing other variables excluding $\mathbf{g}^t$, the subproblem $\mathbf{g}^{t(k+1)}$ at the $(k+1)$th iteration can be written as

$$\mathbf{g}^{t(k+1)} = \underset{\mathbf{g}^t}{\operatorname{argmin}} \frac{\lambda_2}{2}\sum_{i=1}^{d}\left\|\mathbf{B}_t\mathbf{g}_i^t\right\|^2 + \frac{\gamma}{2}\sum_{i=1}^{d}\left\|\mathbf{w}_i - \mathbf{g}_i^t + \mathbf{m}_i\right\|^2. \tag{19}$$

By taking the derivative and setting it zero, each element of $\mathbf{g}^t$ can be computed independently, and thus, the closed-form solution of $\mathbf{g}^t$ is given by

$$\mathbf{g}^t = \left(\tilde{\mathbf{B}}_t\tilde{\mathbf{B}}_t^\top + \lambda_2\mathbf{I}\right)^{-1}\left(\gamma\tilde{\mathbf{w}} + \gamma\tilde{\mathbf{m}}\right) \tag{20}$$

where $\tilde{\mathbf{B}}_t$ represents the $dp \times dp$ diagonal matrix concatenated with d diagonal matrices Diag($\mathbf{B}_t$).

Similarly, the variables $\mathbf{h}$, $\mathbf{m}$, and γ can be updated iteratively in (11) as

$$\begin{aligned}\mathbf{h}^{(k+1)} &= \mathbf{w}^{(k+1)} - \mathbf{g}^{s(k+1)} + \mathbf{h}^{(k)}\\ \mathbf{m}^{(k+1)} &= \mathbf{w}^{(k+1)} - \mathbf{g}^{t(k+1)} + \mathbf{m}^{(k)}\\ \gamma^{(k+1)} &= \min\left(\gamma^{\max}, \rho\gamma^k\right)\end{aligned} \tag{21}$$

where ρ is a scalar term. Algorithm 1 summarizes the optimization procedure.

Scale Estimation: A simple approach to handle SVs is to train CFs at multiple resolutions and then, maximum response is used to estimate the best target scale [10], [39]. However, in contrast to these approaches, we propose a two-step process, which is more efficient as well as SE is more refined. Our approach consists of a coarse and a fine search. In the coarse search, we explicitly search for the target scale by training five correlation filters at different scales using HOG features. In this approach, we only estimate a coarse scale of the target by using the maximum filter response. This strategy is computationally attractive because the HOG features have smaller computational cost and low dimensionality compared to the deep features. However, the SE in this step is a coarse scale that further needs to be refined. In the fine scale search, we further refine the scale search by computing CFs at three different scales of deep features extracted from three different VGG-19 layers as discussed above. The best fine scale is estimated by choosing the maximum response across these three different sets of CFs. The coarse SE may be considered as an explicit estimation step, while the fine SE is an implicit step embedded within our proposed algorithm.

Model Update: We train our filters with an online template update strategy as employed by other CF-based trackers such as those proposed in [27] and [39]. The template update scheme of the model is as follows: $\widehat{\mathbf{x}}_f = (1-\eta)\widehat{\mathbf{x}}_{(f-1)} + \eta\widehat{\mathbf{x}}_f$, where $\widehat{\mathbf{x}}_f$ and $\widehat{\mathbf{x}}_{(f-1)}$ are a template model at frame f and $(f-1)$, respectively, and η is the online learning rate. We observe that the update scheme makes our model effective for pose and lighting variations.

Detection Step: Given an image patch in the next frame, the correlation response at the lth layer R_l is then computed by

$$R_l = \mathcal{F}^{-1}\left(\sum_{i=1}^{d}\widehat{\mathbf{w}}_i \odot \bar{\widehat{\mathbf{x}}}_i\right) \tag{22}$$

where $\bar{\widehat{\mathbf{x}}}_i$ is the complex conjugate of $\widehat{\mathbf{x}}_i$, which is the Fourier transform of the input feature representation at the ith channel at the lth layer, $\mathbf{x}_i$. The maximum response is then computed over all the convolutional layers to obtain the resulting response map [57].

IV. Experimental Evaluations

The performance of the proposed algorithms is evaluated on seven challenging datasets, including: 1) OTB50 [76]; 2) OTB100 [75]; 3) Temple-Colors 128 (TC-128) [52]; 4) UAV123 [60]; 5) VOT2017 [41]; 6) VOT2018-LT [42]; and 7) LaSOT dataset [15]. The description of each dataset is

TABLE II
DETAILS OF THE DATASETS USED IN EXPERIMENTAL EVALUATIONS

Datasets Name	OTB50 [76]	OTB100 [75]	TC128 [52]	UAV123 [60]	VOT2017 [41]	VOT2018-LT[42]	LaSOT [15]
Total Sequences	51	100	129	123	60	35	1400
Minimum Frames	71	71	71	109	41	1389	1000
Maximum Frames	3872	3872	3872	3085	1500	29700	11397
Total Frames	29491	59040	55346	112578	21973	146847	3.52M
Average Resolution	356×530	356×530	458×731	1280×720	465×758	468×785	632×1089

presented in Table II. These datasets comprise of the following tracking challenges: occlusion (Occ), background clutter (BC), SV, deformation (DEF), in-plane rotation (IPR), OPR, out of view (OV), illumination variation (IV), FM, motion blur (MB), and LR. The UAV123 and LaSOT datasets contain additional challenges, including the aspect ratio change (ARC), full occlusion (FOC), POC, similar object (SOB), camera motion (CM), and viewpoint change (VC).

For the proposed algorithm, two main variants are evaluated, including: 1) HSG-DCF with deep features and 2) HSG-DCF-SE using HOG features. The performance of the proposed trackers HSG-DCF and HSG-DCF-SE is compared with 33 existing state-of-the-art trackers divided into four categories as follows.

1) *DCFs with handcrafted features*, including BACF [39], SRDCF [10], STRCF [44], MEEM [81], MUSTer [25], DSST [9], LCT [56], STAPLE [2], and MCCT [73].
2) *DCFs with deep features*, including HCF [57], HCFT [55], HDT [62], DeepSRDCF [11], DeepSTRCF [44], CCOT [12], ECO [8], MCPF [84], PTAV [16], DeepMCCT [73], RPCF [66], ASRCF [7], and GFS-DCF [77].
3) *End-to-end DCFs*, including FCNT [71], CFNET [68], TADT [48], UDT [72], and TRACA [5].
4) *Other methods*, including DSTN [67], GradNet [45], SPLT [78], DGL [43], DeepRSLT [34], and DeepRSST [83].

The VOT performance evaluation is measured using the precision and success rates (SRs) [75] for OTB50, OTB100, TC128, UAV123, and LaSOT datasets. The precision rate (PR) is defined as the percentage of frames with the Euclidean distance between the predicted and ground-truth target location less than 20 pixels threshold [75]. The SR is defined as the percentage of frames with an overlap ratio $[(b_1 \cap b_2)/(b_1 \cup b_2)] > 0.5$ [75], where b_1 and b_2 are the predicted and the ground-truth bounding boxes, respectively. By varying the threshold from 0 to 1, the success plots are generated and the area under the curve is estimated.

A. Experimental Settings

Our proposed HSG-DCF-SE model (2) requires only two regularization parameters λ_1 and λ_2. In our experiments, we performed sensitivity analysis on the OTB50 dataset to estimate appropriate values of λ_1 and λ_2 as shown in Fig. 4. For each parameter, we define a discrete set of values, $\Omega = \{0.06, 0.09, 0.1, 0.2, 0.3, 0.5, 1, 5\}$. For a particular value of λ_1, the SR is computed by varying λ_2 over this set, and the maximum SR is plotted in Fig. 4. For OTB50 sequences, average SR is computed by taking average over 50 sequences for each combination of λ_1 and λ_2. We empirically find that the parameters, $\lambda_1 = 0.2$ and $\lambda_2 = 0.5$, are the best combination, which are then used in all experiments.

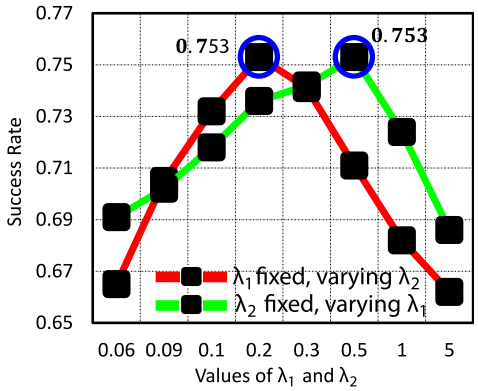

Fig. 4. Sensitivity analysis for the selection of $\lambda_1 = 0.2$ and $\lambda_2 = 0.5$. Blue circles denote the best performance by fixing λ_1 and λ_2.

TABLE III
PERFORMANCE IMPROVEMENT IN STATE-OF-THE-ART TRACKERS USING OUR PROPOSED TEMPORAL GRAPH-BASED REGULARIZATION ON OTB-100 AND TC-128 DATASETS. SRDCF-TC STANDS FOR SRDCF TRACKER WITH TEMPORAL CONSTRAINTS AND SIMILARLY BACF-TC STANDS FOR BACF TRACKER WITH OUR TEMPORAL CONSTRAINTS. THE PERFORMANCE IS REPORTED IN TERMS OF AUC

Datasets	SRDCF [10]	BACF [39]	SRDCF-TC	BACF-TC
OTB-100	0.597	0.629	0.634	0.658
TC-128	0.509	0.519	0.542	0.553

We crop the square region centered at the target, in which the side length of the region is $\sqrt{5mn}$ ($m \times n$ represents the width and the height of the target) and the input features are weighted by a cosine window to reduce the boundary discontinuities as suggested by SRDCF [10]. For scale estimation, HOG-based DCF is used as discussed above. We use 4×4 cell size for HOG features with 31 dimensions. The number of scales is set to 5 with a scale step of 1.01 [39]. For the construction of spatial and temporal graphs, we used $h = 15$ nearest neighbors. For the $\mathbf{G}_t^l$ construction, a temporal window of size $q = 20$ previous target objects is used. PCA is employed to compress the features to 100 dimensions. The optimization hyperparameters are set as $\gamma = 10$, $\gamma^{\max} = 100$, and $\rho = 1.2$, as suggested in [44]. The model learning rate is set as $\eta = 0.012$ [39], [62] and the number of ADMM iterations $k = 3$ is used in all our experiments.

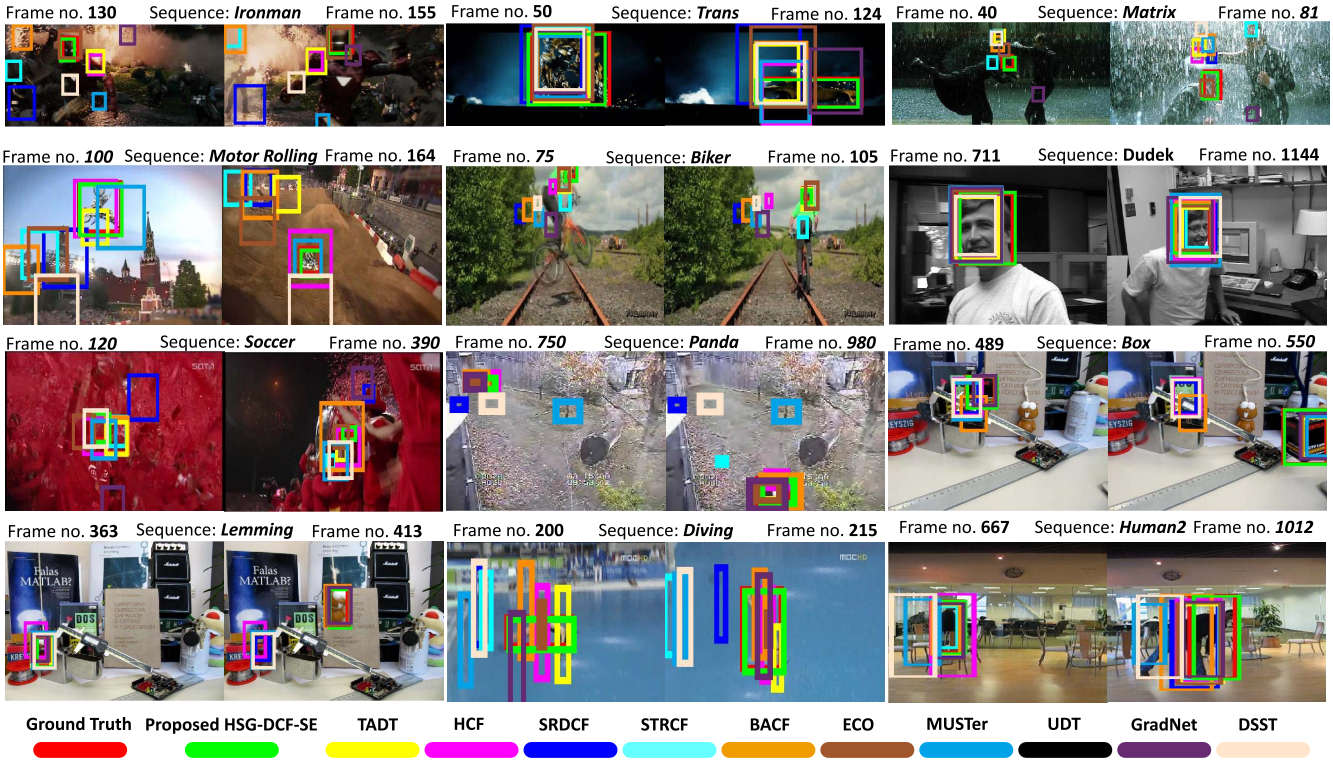

Fig. 5. Visual results of the proposed HSG-DCF-SE tracker and its comparison with current state-of-the-art trackers, including TADT [48], HCF [57], SRDCF [10], STRCF [44], BACF [39], ECO [8], MUSTer [25], UDT [72], GradNet [45], and DSST [9] on 12 challenging sequences selected from OTB50 [76] and OTB100 [75] datasets. Frame indexes and sequence names are shown for each sequence. Our proposed HSG-DCF-SE tracker has consistently performed well against these challenges as compared to the other trackers. (a) Location error threshold (b) Overlap threshol (c) Location error threshold (d) Overlap threshol

B. Ablation Study

We perform the following ablation studies, including VOT performance comparison of different components (Table II in the supplementary material), VOT attributes-based performance comparison (Table III in the supplementary material), performance comparison between different feature configurations (Fig. 1 in the supplementary material), and performance and speed comparison at varying ADMM iterations (Table IV in the supplementary material). For more details, see Section II in the supplementary material.

In addition to the ablation study of the proposed algorithm, we also perform experiments by incorporating the graph-based temporal regularization on two existing SOTA trackers, including: 1) SRDCF [10] and 2) BACF [39]. The purpose of this study is to demonstrate the capability of the temporal regularization for performance improvement of existing trackers. Table III shows the performance obtained by SRDCF-TC and BACF-TC trackers compared to the original versions on two datasets. On the OTB-100 dataset, SRDCF-TC has achieved 3.7% improvement while BACF-TC has obtained 2.9% performance gain. On the TC-128 dataset, SRDCF-TC has obtained 3.3% improvement while BACF-TC has obtained 3.4% performance gain.

C. Qualitative Results

To evaluate the performance of the proposed HSG-DCF-SE tracker, we present rigorous results on key frames of 12 challenging sequences selected from the OTB100 dataset (Fig. 5), and 12 sequences from TC128 and UAV123 datasets [Fig. 2 (supplementary material)]. The bounding boxes of the tracked objects are overlaid on the input images and the comparisons are shown with ten existing trackers, including TADT, HCF, BACF, SRDCF, STRCF, ECO, UDT, MUSTer, GradNet, and DSST. See Section III in the supplementary material for more discussion. Overall, HSG-DCF-SE has performed much better than the compared trackers in all sequences, which can be attributed to spatial and temporal graph-based regularizations in the proposed objective function.

D. Quantitative Evaluations

1) Results on OTB50 Dataset: Fig. 6(a) and (b) shows the comparative performance in terms of precision and success plots of the proposed trackers with other state-of-the-art trackers on the OTB50 dataset. In terms of precision plot, HSG-DCF-SE has obtained 94.7% accuracy, which is 1.7% better than the second best performer ECO (93.0% accuracy). Most trackers obtaining precision of more than 85.0% are based on deep features and deep neural networks. In contrast to that, HSG-DCF has obtained 92.9% precision, which is also better than many deep trackers.

In terms of success plot, HSG-DCF-SE obtained 75.3% accuracy, which is 4.40% better than the second best performer ECO (70.9%). It can be observed that most of the trackers using deep features, including DeepSRDCF, etc., have performed better than end-to-end training-based trackers, including GradNet and FCNT. This fact justifies the use

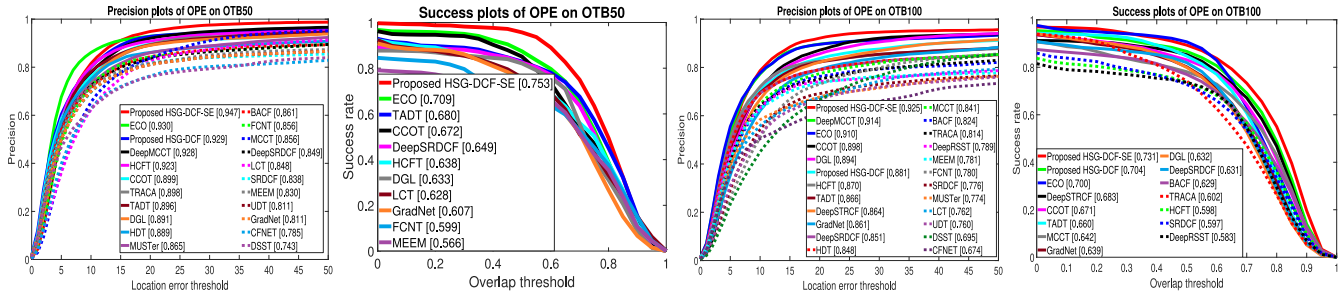

Fig. 6. Precision and success plots using OPE of the proposed HSG-DCF-SE and HSG-DCF trackers against other state-of-the-art trackers on OTB50 and OTB100 datasets. The legend of precision plot contains threshold scores at 20 pixels, while the legend of SR contains area-under-the-curve score for each tracker. The proposed tracker, HSG-DCF-SE, performs better against the state-of-the-art trackers. (a) Location error threshold. (b) Overlap threshold

of deep features in our proposed tracker. The HCFT tracker with SE has obtained 63.8% SR, which is 11.50% less than the HSG-DCF-SE tracker. This huge difference highlights the significance of our proposed spatial and temporal graph-based constraints for improving the VOT performance.

2) Results on OTB100 Dataset: Fig. 6(c) and (d) shows the performance comparison of the proposed trackers with current state-of-the-art trackers on the OTB100 dataset. In terms of precision plot [Fig. 6(c)], the HSG-DCF-SE tracker has obtained 92.5% accuracy, which is better than DeepMCCT, ECO, and other trackers. Our proposed HSG-DCF tracker has obtained 88.1% accuracy, which is competitive with many state-of-the-art trackers. An overview of the performance comparison shows that most of the deep features-based trackers perform better than the end-to-end deep trackers.

In terms of success plot [Fig. 6 (d)], HSG-DCF-SE tracker has obtained 73.1% accuracy, which is 3.1% better than ECO. The HSG-DCF tracker has obtained 70.4% accuracy and outperformed ECO and DeepSTRCF trackers. It shows that in addition to using deep features and scale estimation, the spatial and temporal graph-based constraints generalize better on large datasets and show performance boost compared to the baseline HCFT (59.8%).

3) Attribute-Based Performance Evaluation on the OTB100 Dataset: We also performed the attribute-based performance evaluation on the OTB100 dataset containing 11 different challenges, including IV, SV, Occ, DEF, MB, FM, IPR, OPR, OV, BC, and LR. Table V (supplementary material) shows this comparison in terms of PR and SR with existing state-of-the-art trackers. In terms of the PR comparison, the proposed HSG-DCF-SE tracker achieves the best results under 4 out of 11 attributes while in terms of SR comparison, HSG-DCF-SE has achieved the best performance under 9 out of 11 attributes. For more details, see Section IV in the supplementary material.

4) Results on TC128 Dataset: Table IV shows the performance comparison of the proposed HSG-DCF-SE tracker with existing state-of-the-art trackers on the TC128 dataset. In terms of PR, the HSG-DCF-SE tracker has obtained the best performance of 84.8%, which is 4.8% and 4.9% better than the ECO and DeepMCCT trackers. In terms of SR, the proposed HSG-DCF-SE tracker has obtained best SR of 69.4%, which is 8.9% and 9.3% better than the ECO and DeepMCCT trackers. The TC128 dataset is more challenging compared to OTB50 and OTB100 datasets; therefore, the performance of all compared trackers has reduced. These results demonstrate the advantages of incorporating spatial and temporal appearance consistency constraints. The best performance of the proposed tracker suggests that it can handle various challenging factors more effectively than the compared trackers.

TABLE IV
PERFORMANCE COMPARISON WITH STATE-OF-THE-ART TRACKERS ON TC-128 DATASET. THE PERFORMANCE IS REPORTED IN TERMS OF PR AT A THRESHOLD OF 20 PIXELS AND AUC FOR SR

Trackers	PR	AUC	Trackers	PR	AUC
HSG-DCF-SE	**0.848**	**0.694**	HSG-DCF	0.794	**0.661**
ECO	**0.800**	0.605	DeepMCCT	0.799	0.586
MCCT	0.720	0.551	MCPF	0.776	0.544
CCOT	0.774	0.597	DeepSTRCF	0.753	0.601
TADT	0.751	0.562	DeepSRDCF	0.740	0.536
PTAV	0.741	0.544	SRDCF	0.696	0.509
STAPLE	0.665	0.498	HCF	0.703	0.488
UDT	0.717	0.541	BACF	0.660	0.519
HDT	0.686	0.452	LCT	0.606	0.455
DSST	0.534	0.405	CFNET	0.607	0.456
STRCF	0.712	0.553	MUSTer	0.636	0.471
MEEM	0.639	0.459	DGL	0.748	0.535
GradNet	0.746	0.556			

5) Results on UAV123 Dataset: Many state-of-the-art trackers have not evaluated their performance on the UAV123 dataset. We, therefore, compare our performance with only those ten trackers who reported results on this dataset. Table V shows the performance comparison in terms of PR and SR of the proposed HSG-DCF-SE tracker with existing state-of-the-art trackers, including GCT, SRDCF, STRCF, MEEM, BACF, MUSTer, DSST, ECO, MCCT, and Staple. In terms of PR, the HSG-DCF-SE tracker has obtained the best performance of 83.6% and HSG-DCF has obtained 80.2%. Among the compared trackers, ECO has obtained the best performance of 74.1%, which is 9.5% less than HSG-DCF-SE. In terms of SR, the HSG-DCF-SE tracker has obtained best performance of 78.2% and HSG-DCF has obtained 76.1%, which is better than the ECO (52.5%) tracker.

TABLE V
PERFORMANCE COMPARISONS WITH EXISTING STATE-OF-THE-ART TRACKERS ON UAV123 DATASET. THE PERFORMANCE IS REPORTED IN TERMS OF PR AT A THRESHOLD OF 20 PIXELS AND AUC FOR SR

Measures	Proposed HSG-DCF-SE	HSG-DCF	GCT	SRDCF	STRCF	MEEM	BACF	MUSTer	DSST	ECO	MCCT	Staple
PR	**0.836**	**0.802**	0.732	0.676	0.681	0.627	0.592	0.591	0.586	0.741	0.662	0.666
SR	**0.782**	**0.761**	0.508	0.464	0.481	0.392	0.396	0.391	0.356	0.525	0.458	0.450

6) Results on VOT2017/VOT2018 Short-Term Datasets: We have evaluated our proposed tracker HSG-DCF-SE on short-term challenge of the VOT2018 dataset [42], which are the same sequences as in the VOT2017 dataset [41]. The 60 sequences contain more deformations and noise compared to the aforementioned datasets. The main aim is to evaluate tracking performance such that if a failure happens that tracker is reinitialized.

Following the protocols defined in VOT2017/VOT2018 [42], we used three primary measures, including: 1) expected average overlap (EAO); 2) robustness (R); and 3) accuracy (A), to compare the performance of different trackers. The EAO estimates the average overlap a tracker is expected to obtain on a large set of short-length sequences with the same visual properties as a given dataset. The robustness measures the number of times a tracker fails (loss the target) during tracking while accuracy is the average overlap between the ground truth and estimated bounding box during the successful tracking periods.

Table VI compares the tracking performance of the proposed tracker with 12 existing state-of-the-art trackers that participated in the VOT2017/VOT2018 challenge. In terms of EAO, both HSG-DCF-SE and GFS-DCF trackers have obtained the best score of 39.0% while HSG-DCF is the nearest competitor obtaining 37.0%. In terms of accuracy measure, the HSG-DCF-SE tracker has attained the best accuracy of 57.0%, which is 6.0% better than the second best performing tracker GFS-DCF (51.0%). In terms of robustness, GFS-DCF and DeepSTRCF trackers have obtained better performance of 14.0% and 21.0% while our proposed HSG-DCF-SE tracker remained the third best tracker in terms of robustness. It is because the proposed tracker does not have the redetection strategy, which would have increased all measures, especially the robustness score.

7) Results on VOT2018-LT Dataset: The VOT2018-LT dataset is used to evaluate the long-term performance of different trackers [42]. We evaluate the performance of the trackers using precision (Pr), recall (Re), and F-score as defined in the VOT2018-LT evaluation protocol [42] and shown in Table VII. In the VOT2018-LT dataset, the highest F-score achieved by a particular tracker is used to rank different trackers; therefore, in Table VII, the highest F-score and the corresponding precision and recall values are shown. It can be noticed that our proposed HSG-DCF-SE tracker has achieved the best F-score of 69.0%. Besides the proposed tracker, TADT, UDT, and CCOT trackers have also achieved comparative performance of 68.0%, 68.0%, and 62.0%, respectively.

8) Results on LaSOT Dataset: The LaSOT dataset is comparatively a large-scale dataset consisting of 1400 sequences. We evaluate our trackers on the test set consisting of 280 videos [15]. Fig. 7(a) and (b) shows the performance comparison of the proposed HSG-DCF-SE with existing state-of-the-art trackers in terms of precision and success plots using OPE. In terms of precision, HSG-DCF-SE has obtained 35.6% accuracy, which is 5.5% better than the second best ECO tracker (30.1% accuracy). All of the compared trackers, except ECO, have obtained precision score less than 30.0%, which demonstrates that they were not able to handle long-term tracking challenges.

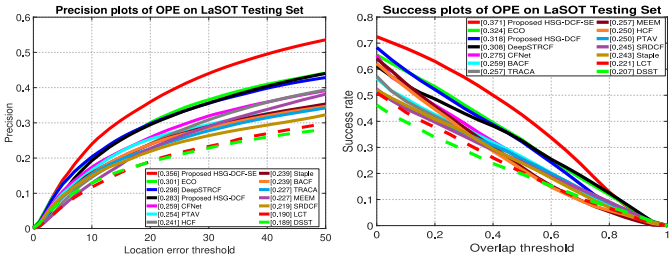

Fig. 7. Precision and success plots using OPE of the proposed HSG-DCF-SE and HSG-DCF trackers against other state-of-the-art trackers on LaSOT dataset [15]. The legend of precision plot contains threshold scores at 20 pixels, while the legend of SR contains area-under-the-curve score for each tracker.

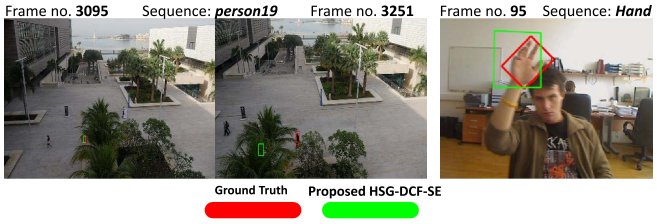

Fig. 8. Failure cases of the proposed tracker in the case of FOC and rotated bounding box challenges. Sequence *person19* is taken from the UAV123 dataset while sequence *Hand* is taken from the VOT2017 dataset.

In terms of success, HSG-DCF-SE has obtained 37.1% accuracy, which is 4.7% better than ECO (32.4%) and 6.3% better than DeepSTRCF. The HSG-DCF tracker has obtained 31.8% accuracy and outperformed DeepSTRCT, BACF, and TRACA trackers. These results demonstrate that the proposed spatial and temporal graph-based regularization assisted our trackers for performance improvement over long-term tracking challenges.

E. Failure Cases

In Fig. 8, we demonstrate two different failure cases of the proposed tracker. In sequence *person19*, our tracker was not able to track the person in the case of long-term and full occlusion. It is because, there is no target redetection module in our proposed tracker. In sequence *Hand*, the intersection over

TABLE VI
PERFORMANCE EVALUATION ON THE VOT2017/VOT2018 DATASETS [41], [42] IN TERMS OF EXPECTED AVERAGE OVERLAP (EAO), ACCURACY (A), AND ROBUSTNESS (R). THE BEST TWO RESULTS ARE SHOWN IN RED AND BLUE COLORS, RESPECTIVELY

Measures	ECO	CCOT	Staple	DeepSRDCF	DSST	DeepSTRCF	DSTN	GradNet	DeepRSLT	HSG-DCF	RPCF	GFS-DCF	ASRCF	HSG-DCF-SE
EAO↑	0.28	0.26	0.16	0.15	0.07	0.34	0.24	0.29	0.30	**0.37**	0.31	**0.39**	0.32	**0.39**
A↑	0.48	0.49	0.53	0.49	0.39	0.52	0.50	0.50	0.50	**0.56**	0.50	0.51	0.49	**0.57**
R↓	0.27	0.31	0.68	0.70	1.45	**0.21**	0.67	0.37	0.35	0.34	0.23	**0.14**	0.23	0.28

TABLE VII
PERFORMANCE COMPARISON ON VOT2018-LT DATASET [42] IN TERMS OF PRECISION (PR), RECALL (RE), AND F-SCORE

Measures	PTAV	SPLT	CCOT	DeepSRDCF	DeepSTRCF	UDT	TADT	HSG-DCF	HSG-DCF-SE
F-score↑	0.48	0.61	0.62	0.57	0.51	0.62	**0.68**	0.66	**0.69**
Pr↑	0.59	0.63	0.59	0.41	0.52	**0.82**	0.71	0.74	**0.78**
Re↑	0.40	0.60	**0.66**	**0.93**	0.51	0.50	**0.66**	0.60	0.63

union of the proposed tracker significantly reduced because of the rotated ground-truth bounding box. The proposed tracker only generates axis aligned bounding boxes, which results in degraded performance if the target object undergoes IPR with rotated bounding box challenge. Inclusion of target orientation detection module would have improved the performance in such scenarios.

F. Computational Complexity and Execution Time

We evaluate the execution time and computational complexity of the proposed trackers, which mainly depend on the optimization process and graph construction. We used FLANN libraries for the graph construction using the nearest neighbor strategy [61]. The spatial graph $\mathbf{G}_s^l$ complexity is $O(pd\log(p))$ and temporal graph $\mathbf{G}_t^l$ is $O(pd\log(d))$, where $p = m \times n$ is the number of pixels and d is the number of channels in each deep features hierarchy.

Since (13) is separable in each pixel location, we solve p subproblems and each is a system of linear equations with d variables. Each subproblem can be solved in $O(d)$ using the Sherman–Morrison formula. Thus, the complexity of solving $\hat{\mathbf{w}}$ is $O(dp)$. Taking the DFT and inverse DFT into account, the complexity of solving $\mathbf{w}$ is $O(dp\log(p))$. The computational cost for solving both $\mathbf{g}^s$ and $\mathbf{g}^t$ is $O(dp)$. Hence, the complexity of our HSG-DCF is $O(kdp\log(p))$, where k is the number of ADMM iterations.

The execution time of the proposed trackers is measured on a PC with an Intel core i7 4.0 GHz, Titan Xp GPU, and 64-GB RAM. The proposed HSG-DCF-SE tracker is able to track a target object at 5.64 frames per second while the proposed HSG-DCF tracker can track target object at 8.28 frames per second for the OTB100 dataset. Similarly, the HOG version of our proposed trackers, HSG-DCF-HOG-SE and HSG-DCF-HOG, takes 14.91 and 20.28 FPS to track the target object, as in Table IV (supplementary material). Compared to that the other trackers, including HCF, HCFT, HDT, Staple, LCT, SRDCF, DeepSTRCF, BACF, DeepSRDCF, STRCF, and DeepMCCT have reported 10.4, 6.70, 10, 80, 20.7, 5.62, 5.3, 26.7, 0.2, 24.3, and 8.0 frames per second, respectively, on their machines using the OTB100 dataset. Although, because of the difference of hardware used by each author, a direct comparison may not be very meaningful, and our processing speed shows the practical significance of the proposed algorithms.

V. CONCLUSION

In this work, a new set of constraints based on spatial and temporal consistency of the target object is proposed in the DCF framework to handle challenging VOT scenarios. The proposed spatial constraint incorporates the spatial structure of the target object by constructing a dense graph across different target components based on hierarchical deep features. The other constraint incorporates the temporal appearance variations of the target object in a temporal window into the DCF framework. The temporal graph is also constructed using hierarchical deep features where PCA is applied to compress the dimensionality of the feature vector. A pair of spatial and temporal graphs is computed at each resolution level of the deep features. The proposed objective function containing spatial and temporal graph-based constraints in the DCF framework is solved using the ADMM optimization method and a closed-form solution of each subproblem is derived in an efficient manner. At each level, we independently compute the constrained correlation filter response and a maximum is seeked across all levels. The SE is performed by computing maximum filter response at five different scales using HOG features. The proposed tracker, called HSG-DCF-SE, has shown significant performance improvement on seven challenging datasets compared to 33 existing state-of-the-art trackers. In the future, we aim to incorporate saliency-based target object specific constraints into the DCF framework with a redetection strategy to further boost the VOT performance.

REFERENCES

[1] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, no. 85, PP. 2399–2434, 2006.
[2] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. Torr, "Staple: Complementary learners for real-time tracking," in *Proc. IEEE CVPR*, 2016, pp. 1401–1409.
[3] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE CVPR*, 2010, pp. 2544–2550.
[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[5] J. Choi *et al.*, "Context-aware deep feature compression for high-speed visual tracking," in *Proc. IEEE CVPR*, 2018, pp. 479–488.

[6] J. Choi, H. J. Chang, J. Jeong, Y. Demiris, and J. Y. Choi, "Visual tracking using attention-modulated disintegration and integration," in *Proc. IEEE CVPR*, 2016, pp. 4321–4330.

[7] K. Dai, D. Wang, H. Lu, C. Sun, and J. Li, "Visual tracking via adaptive spatially-regularized correlation filters," in *Proc. IEEE CVPR*, 2019, pp. 4670–4679.

[8] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "Eco: Efficient convolution operators for tracking," in *Proc. IEEE CVPR*, 2017, pp. 6931–6939.

[9] 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.

[10] M. Danelljan, G. H´ager, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE ICCV*, 2015, pp. 4310–4318.

[11] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE ICCV-W*, 2015, pp. 621–629.

[12] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. ECCV*, 2016, pp. 472–488.

[13] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. IEEE CVPR*, 2014, pp. 1090–1097.

[14] C. Deng, Y. Han, and B. Zhao, "High-performance visual tracking with extreme learning machine framework," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2781–2792, Jun. 2020.

[15] H. Fan *et al.*, "Lasot: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 5374–5383.

[16] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *Proc. IEEE ICCV*, 2017, pp. 5487–5495.

[17] M. Fiaz, A. Mahmood, S. Javed, and S. K. Jung, "Handcrafted and deep trackers: Recent visual object tracking approaches and trends," *ACM Comput. Surveys*, vol. 52, no. 2, pp. 1–44, 2019.

[18] H. K. Galoogahi, T. Sim, and S. Lucey, "Multi-channel correlation filters," in *Proc. IEEE ICCV*, 2013, pp. 3072–3079.

[19] J. H. Giraldo, S. Javed, and T. Bouwmans, "Graph moving object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Dec. 9, 2020, doi: 10.1109/TPAMI.2020.3042093.

[20] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. (CVPR)*, vol. 2, 2006, pp. 1735–1742.

[21] Y. Han, C. Deng, B. Zhao, and D. Tao, "State-aware anti-drift object tracking," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4075–4086, Aug. 2019.

[22] X. He, D. Cai, Y. Shao, H. Bao, and J. Han, "Laplacian regularized Gaussian mixture model for data clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 9, pp. 1406–1418, Sep. 2011.

[23] Z. He, S. Yi, Y.-M. Cheung, X. You, and Y. Y. Tang, "Robust object tracking via key patch sparse representation," *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 354–364, Feb. 2017.

[24] 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.

[25] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, "Multi-store tracker (MUSTer): A cognitive psychology inspired approach to object tracking," in *Proc. IEEE CVPR*, 2015, pp. 749–758.

[26] H. Hu, B. Ma, J. Shen, and L. Shao, "Manifold regularized correlation object tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1786–1795, May 2018.

[27] Z. Huang, C. Fu, Y. Li, F. Lin, and P. Lu, "Learning aberrance repressed correlation filters for real-time UAV tracking," in *Proc. IEEE ICCV*, 2019, pp. 2891–2900.

[28] S. Javed, A. Mahmood, S. Al-Maadeed, T. Bouwmans, and S. K. Jung, "Moving object detection in complex scene using spatiotemporal structured-sparse RPCA," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 1007–1022, Feb. 2019.

[29] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung, "Spatiotemporal low-rank modeling for complex scene background initialization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 6, pp. 1315–1329, Jun. 2018.

[30] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung, "Background-foreground modeling based on spatiotemporal sparse subspace clustering," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5840–5854, Dec. 2017.

[31] S. Javed, A. Mahmood, J. Dias, and W. Naoufel, "CS-RPCA: Clustered sparse rpca for moving object detection," in *Proc. IEEE ICIP*, 2020, pp. 3209–3213.

[32] S. Javed, A. Mahmood, J. Dias, and N. Werghi, "Structural low-rank tracking," in *Proc. IEEE AVSS*, 2019, pp. 1–8.

[33] S. Javed *et al.*, "Cellular community detection for tissue phenotyping in colorectal cancer histology images," *Med. Image Anal.*, vol. 63, Jul. 2020, Art. no. 101696.

[34] S. Javed, A. Mahmood, D. Jorge, and N. Werghi, "Robust structural low-rank tracking," *IEEE Trans. Image Process.*, vol. 29, pp. 4390–4405, 2020.

[35] S. Javed, A. Mahmood, N. Rajpoot, J. Dias, and N. Werghi, "Spatially constrained context-aware hierarchical deep correlation filters for nucleus detection in histology images," *Med. Image Anal.*, May 2021, Art. no. 102104.

[36] S. Javed, A. Mahmood, N. Werghi, K. Benes, and N. Rajpoot, "Multiplex cellular communities in multi-gigapixel colorectal cancer histology images for tissue phenotyping," *IEEE Trans. Image Process.*, vol. 29, pp. 9204–9219, Sep. 2020.

[37] S. Javed, X. Zhang, J. Dias, L. D. Seneviratne, and N. Werghi, "Spatial graph regularized correlation filters for visual object tracking." in *Proc. SoCPaR*, 2020, pp. 186–195.

[38] S. Javed, X. Zhang, L. D. Seneviratne, J. Dias, and N. Werghi, "Deep bidirectional correlation filters for visual object tracking," in *Proc. IEEE Inf. Fus.*, 2020, pp. 1–8.

[39] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. IEEE ICCV*, 2017, pp. 1144–1152.

[40] H. K. Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proc. IEEE CVPR*, 2015, pp. 4630–4638.

[41] M. Kristan *et al.*, "The visual object tracking VOT2017 challenge results," in *Proc. IEEE ICCV-W*, 2017, pp. 1949–1972.

[42] M. Kristan *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *ECCV-W*, 2018, pp. 3–53.

[43] C. Li, L. Lin, W. Zuo, J. Tang, and M.-H. Yang, "Visual tracking via dynamic graph learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2770–2782, Nov. 2019.

[44] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang, "Learning spatial-temporal regularized correlation filters for visual tracking," in *Proc. IEEE CVPR*, 2018, pp. 4904–4913.

[45] P. Li, B. Chen, W. Ouyang, D. Wang, X. Yang, and H. Lu, "Gradnet: Gradient-guided network for visual object tracking," in *Proc. IEEE ICCV*, 2019, pp. 6161–6170.

[46] S. Li, S. Zhao, B. Cheng, and J. Chen, "Noise-aware framework for robust visual tracking," *IEEE Trans. Cybern.*, early access, Jun. 10, 2020, doi: 10.1109/TCYB.2020.2996245.

[47] S. Li, S. Zhao, B. Cheng, E. Zhao, and J. Chen, "Robust visual tracking via hierarchical particle filter and ensemble deep features," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 1, pp. 179–191, Jan. 2020.

[48] X. Li, C. Ma, B. Wu, Z. He, and M.-H. Yang, "Target-aware deep tracking," in *Proc. IEEE CVPR*, 2019, pp. 1369–1378.

[49] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Proc. ECCV*, 2014, pp. 254–265.

[50] Y. Li, J. Zhu, and S. C. H. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *IEEE CVPR*, 2015, pp. 353–361.

[51] Y. Li, C. Fu, F. Ding, Z. Huang, and G. Lu, "Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2020, pp. 11920–11929.

[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] G. Liu, "Robust visual tracking via smooth manifold kernel sparse learning," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 2949–2963, Nov. 2018.

[54] S. Liu, T. Zhang, X. Cao, and C. Xu, "Structural correlation filter for robust visual tracking," in *Proc. IEEE CVPR*, 2016, pp. 4312–4320.

[55] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Robust visual tracking via hierarchical convolutional features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2709–2723, Nov. 2019.

[56] C. Ma, J.-B. Huang, X. Yang, and Y. Ming-Hsuan, "Adaptive correlation filters with long-term and short-term memory for object tracking," *Int. J. Comput. Vision*, vol. 126, no. 8, pp. 771–796, 2018.
[57] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE ICCV*, 2015, pp. 3074–3082.
[58] A. Mahmood, A. Mian, and R. Owens, "Semi-supervised spectral clustering for image set classification," in *Proc. IEEE CVPR*, 2014, pp. 121–128.
[59] M. Mueller, N. Smith, and G. Bernard, "Context-aware correlation filter tracking," in *Proc. IEEE CVPR*, 2017, pp. 1387–1395.
[60] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. ECCV*, 2016, pp. 445–461.
[61] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2227–2240, Nov. 2014.
[62] Y. Qi *et al.*, "Hedged deep tracking," in *Proc. IEEE CVPR*, 2016, pp. 4303–4311.
[63] L. Rout, P. M. Raju, D. Mishra, and R. K. S. S. Gorthi, "Learning rotation adaptive correlation filters in robust visual object tracking," in *Proc. ACCV*, 2018, pp. 646–661.
[64] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[65] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: arXiv:1409.1556.
[66] Y. Sun, C. Sun, D. Wang, Y. He, and H. Lu, "Roi pooled correlation filters for visual tracking," in *Proc. IEEE CVPR*, 2019, pp. 5783–5791.
[67] Z. Teng, J. Xing, Q. Wang, B. Zhang, and J. Fan, "Deep spatial and temporal network for robust visual object tracking," *IEEE Trans. Image Process.*, vol. 29, pp. 1762–1775, Sep. 2019.
[68] J. Valmadre, L. Bertinetto, J. F. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proc. IEEE CVPR*, 2017, pp. 5000–5008.
[69] G. S. Walia, H. Ahuja, A. Kumar, N. Bansal, and K. Sharma, "Unified graph-based multicue feature fusion for robust visual tracking," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2357–2368, Jun. 2020.
[70] J. Wang, L. Zheng, M. Tang, and J. Feng, "A comparison of correlation filter-based trackers and struck trackers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 9, pp. 3106–3118, Sep. 2020.
[71] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE ICCV*, 2015, pp. 3119–3127.
[72] N. Wang, Y. Song, C. Ma, W. Zhou, W. Liu, and H. Li, "Unsupervised deep tracking," in *Proc. IEEE CVPR*, 2019, pp. 1308–1317.
[73] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *Proc. IEEE CVPR*, 2018, pp. 4844–4853.
[74] W. Wang, K. Zhang, M. Lv, and J. Wang, "Hierarchical spatiotemporal context-aware correlation filters for visual tracking," *IEEE Trans. Cybern.*, early access, Jan. 30, 2020, doi: 10.1109/TCYB.2020.2964757.
[75] 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.
[76] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE CVPR*, 2013, pp. 2411–2418.
[77] T. Xu, Z.-H. Feng, X.-J. Wu, and J. Kittler, "Joint group feature selection and discriminative filter learning for robust visual object tracking," in *Proc. IEEE ICCV*, 2019, pp. 7949–7959.
[78] B. Yan, H. Zhao, D. Wang, H. Lu, and X. Yang, "'skimming-perusal'tracking: A framework for real-time and robust long-term tracking," in *Proc. IEEE ICCV*, 2019, pp. 2385–2393.
[79] M. Yin, J. Gao, and Z. Lin, "Laplacian regularized low-rank representation and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 504–517, Mar. 2016.
[80] B. Zhang *et al.*, "Output constraint transfer for kernelized correlation filter in tracking," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 4, pp. 693–703, Apr. 2017.
[81] J. Zhang, S. Ma, and S. Sclaroff, "Meem: robust tracking via multiple experts using entropy minimization," in *Proc. ECCV*, 2014, pp. 188–203.
[82] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Proc. ECCV*, 2014, pp. 127–141.
[83] 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. 2019.
[84] T. Zhang, C. Xu, and M.-H. Yang, "Learning multi-task correlation particle filters for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 365–378, Feb. 2019.
[85] Z. Zhang and K. Zhao, "Low-rank matrix approximation with manifold regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1717–1729, Jul. 2013.
[86] G. Zhu, J. Wang, Y. Wu, and H. Lu, "Collaborative correlation tracking." in *Proc. BMVC*, 2015, p. 184.

Sajid Javed received the B.Sc. degree in computer science from the University of Hertfordshire, Hatfield, U.K., in 2010, and the combined master's and Ph.D. degree in computer science from Kyungpook National University, Daegu, South Korea, in 2017.

He is an Assistant Professor of Computer Vision with the Electrical and Computer Engineering Department, Khalifa University of Science and Technology, Abu Dhabi, UAE. Prior to that, he was a Research Scientist with Khalifa University Center for Autonomous Robotics System, Abu Dhabi, from 2019 to 2021. Before joining Khalifa University, Abu Dhabi, he was a Research Fellow with the University of Warwick, Coventry, U.K., from 2017 to 2018, where he worked on histopathological landscapes for better cancer grading and prognostication. His research interests include visual object tracking in the wild, multiobject tracking, background–foreground modeling from video sequences, moving object detection from complex scenes, and cancer image analytics, including tissue phenotyping, nucleus detection, and nucleus classification problems. His research themes involve developing deep neural networks, subspace learning models, and graph neural networks.

Arif Mahmood received the M.Sc. and Ph.D. degrees in computer science from the Lahore University of Management Sciences (LUMS), Lahore, Pakistan, in 2003 and 2011.

He is a Professor with the Department of Computer Science, Information Technology University, Lahore, Pakistan, and the Director of Computer Vision Lab. He is actively working on cancer grading and prognostication using histology images, predictive autoscaling of services hosted on the cloud and the fog infrastructures, and ocean color monitoring using remote sensing. He has also worked as a Research Assistant Professor with the School of Mathematics and Statistics (SMS), University of the Western Australia (UWA), Perth, WA, Australia. In SMS, he worked on community detection in social and scientific networks. Before that, he was a Research Assistant Professor with the School of Computer Science and Software Engineering, UWA, and performed research on face recognition, object classification, and action recognition. His current research directions are person pose detection and segmentation, crowd counting and flow detection, background–foreground modeling in complex scenes, object detection, human-object interaction detection, and abnormal event detection.

Jorge Dias (Senior Member, IEEE) B.Sc., M.Sc., Ph.D., and Habilitation degrees in electrical engineering from the University of Coimbra, Coimbra, Portugal, in 1984, 1988, 1994, and 2011, respectively.

He is a Professor of ECE/Robotics with Khalifa University, Abu Dhabi, UAE. His research is in the area of computer vision and robotics and has contributions on the field since 1984. He has several publications in international journals, books, and conferences. He has been a Principal Investigator for several international research projects. He published several articles in the area of computer vision and robotics that include more than 80 publications in international journals, one published book, 15 books chapters, and more than 280 articles in international conferences with referee.

Lakmal Seneviratne (Senior Member, IEEE) received the B.Sc. (Eng.) and Ph.D. degrees in mechanical engineering from King's College London, London, U.K., in 1980 and 1986, respectively.

He is currently a Professor of Mechanical Engineering and the Founding Director of the Centre for Autonomous Robotic Systems, Khalifa University, Abu Dhabi, UAE. He is also the Technical Director of the Mohammed Bin Zayed International Robotics Challenge. Prior to joining Khalifa University, he was a Professor of Mechatronics, the Founding Director of the Centre for Robotics Research, and the Head of the Division of Engineering, King's College London. His main research interests are centred on robotics and automation, with particular emphasis on increasing the autonomy of robotic systems interacting with complex dynamic environments. He has published over 400 peer-reviewed publications on these topics.

Prof. Seneviratne is a member of the Mohammed Bin Rashid Academy of Scientists in the UAE.

Naoufel Werghi (Senior Member, IEEE) received the Diploma degree in electrical engineering from the École Nationale Ingénieurs Monastir, Monastir, Tunisia, in 1992, the M.Sc. degree in instrumentation and control from the University of Rouen, Mont-Saint-Aignan, France, in 1993, and the Ph.D. degree in robotic vision in 1996 and the Habilitation degree in computer vision from the University of Strasbourg, Strasbourg, France.

He has been a Research Fellow with the Division of Informatics, University of Edinburgh, Edinburgh, U.K., and a Lecturer with the Department of Computer Sciences, University of Glasgow, Glasgow, U.K. He is currently an Associate Professor with the Electrical Engineering and Computer Science Department, Khalifa University for Science and Technology, Abu Dhabi, UAE. He has been a Visiting Professor with the University of Louisville, Louisville, KY, USA; University of Florence, Florence, Italy; University of Lille, Lille, France; and Korean Advanced and Institute of Sciences and Technology, Daejeon, South Korea. His main research area is 2D/3D image analysis and interpretation, where he has been leading several funded projects related to biometrics, medical imaging, remote sensing, and intelligent systems.

Dr. Werghi is an Associate Editor of the *Eurasip Journal for Image and Video Processing*. He is a member of the IEEE Signal Processing Society and IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.