Visual Tracking using Structural Local DCT Sparse Appearance Model with Occlusion Detection

B. K. Shreyamsha Kumar , M.N.S. Swamy , M. Omair Ahmad

© Springer Science+Business Media, LLC, part of Springer Nature 2018.

This is Author's Post-print that is published in Springer's Multimedia Tools and Applications and is available at https://link.springer.com (DOI:10.1007/s11042-018-6453-z). Permission to make digital or hard copies of all or part of this work for research or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for commercial purposes, or modification of the content of the paper are prohibited and require prior specific permission and/or a fee.

Cite this article as:
B.K. Shreyamsha Kumar, M.N.S. Swamy and M. Omair Ahmad, "Visual Tracking using Structural Local DCT Sparse Appearance Model with Occlusion Detection", Multimedia Tools and Appl., Vol. 78, Issue 6, pp. 7243-7266, 2019, DOI:10.1007/s11042-018-6453-z

Bibtex:
@article{LDSAMOD2019,
title = Visual Tracking using Structural Local DCT Sparse Appearance Model with Occlusion Detection,
author = {Shreyamsha Kumar, B. K. and Swamy, M. N. S. and Omair Ahmad, M.},
journal = {Multimedia Tools and Appl.},
pages = {7243–7266},
volume = {78},
number = {6},
month = {Mar},
year = {2019},
DOI = {10.1007/s11042-018-6453-z}
}

Multimedia Tools and Applications (2019) 78: 7243-7266
https://doi.org/10.1007/s11042-018-6453-z
Post-print

Visual Tracking using Structural Local DCT Sparse Appearance Model with Occlusion Detection

B. K. Shreyamsha Kumar · **M.N.S. Swamy** · **M. Omair Ahmad**

08-Aug-2018

Abstract In this paper, a structural local DCT sparse appearance model with occlusion detection is proposed for visual tracking in a particle filter framework. The energy compaction property of the 2D-DCT is exploited to reduce the size of the dictionary as well as that of the candidate samples so that the computational cost of l_1-minimization can be lowered. Further, a holistic image reconstruction procedure is proposed for robust occlusion detection and used for appearance model update, thus avoiding the degradation of the appearance model in the presence of occlusion/outliers. Also, a patch occlusion ratio is introduced in the confidence score computation to enhance the tracking performance. Quantitative and qualitative performance evaluations on two popular benchmark datasets demonstrate that the proposed tracking algorithm generally outperforms several state-of-the-art methods.

Keywords Visual Tracking · Local DCT Sparse Appearance Model · Holistic Image Reconstruction · Reconstruction Error · Occlusion Map · Observation Model Update.

1 Introduction

In the last two decades, visual object tracking has seen a flurry of research due to its wide range of real-life applications including vehicle navigation, robotics, human behavior analysis, action recognition, human computer interaction, video indexing and retrieval, medical imaging, security and surveillance [45]. In spite of this, it still remains a challenging problem due to the following reasons: (1) complexity in target searching, (2) intrinsic (e.g., pose changes, shape deformation) and extrinsic (e.g., varying viewpoints, rotation and scaling due to camera motion, illumination changes, occlusions, cluttered and moving backgrounds) object appearance variations [39, 44]. In order to carefully handle these appearance variations, a good appearance model that adapts to intrinsic appearance variations and be robust to extrinsic appearance variations is needed.

✉ M. Omair Ahmad
omair@encs.concordia.ca

Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec H3G 1M8 Canada

In general, there are three main components in visual tracking, namely, (1) an observation model, used to represent the tracked object and finds whether an observed image patch belongs to the object class or not, (2) a dynamic model, which describes the states of an object over time and predicts its likely state (e.g., Kalman filter [6], and particle filter [21]), and (3) an observation model update to adapt the appearance variations of the object. The proposed method contributes to all these three components and are as follows: (1) a new observation model based on 2D discrete cosine transform (DCT) features, (2) a modification of the observation likelihood of each particle to improve the tracking performance, and (3) reconstruction of the holistic image and its application to occlusion detection for the observation model update.

In the literature, the tracking algorithms are categorized as belonging to either generative or discriminative approaches based on the representation scheme used to model the appearance of the object. Discriminative methods extract information from both the target and the background to differentiate the target from the background (e.g., using boosting algorithms [2], semi-supervised learning [12] and support vector machines [41]). On the other hand, generative methods extract information only from the target region to model the object appearance and search for a region that is most similar to the target model. These methods are based on templates [3, 6, 24, 26], or local patches/fragments [1, 17, 31], or subspace models [29, 32, 33, 35] or local subspace models [28]. Since the generative methods consider information from the target region alone for object appearance, they are not efficient in cluttered environments, but they achieve higher generalization with limited data. In contrast, the discriminative methods perform better if the training set is large due to its capability of differentiating the target from the background. The advantages of these individual methods are exploited by collaborating both generative and discriminative methods for object appearance model in [48, 55].

Recently, Henriques *et al.* [15] have proposed a high speed tracking algorithm based on kernelized correlation filter (KCF), which exploits the properties of circulant data in the Fourier domain to diagonalize the elements of the circulant data. Further, Danelljan *et al.* [8] proposed a spatially regularized discriminative correlation filter (SRDCF) for tracking by introducing a spatial regularization in the learning process to penalize the correlation filter coefficients depending on their spatial location. Wang *et al.* [42] have proposed a deep learning tracker (DLT) by training a stacked denoising autoencoder network to learn generic image features from a large image dataset in an unsupervised fashion and then using those features for online tracking. Li *et al.* [19] proposed a tracking algorithm using four-layer convolutional neural network (CNN) to distinguish the target from its surroundings. In addition to that, they proposed a truncated structural loss function to maintain as many training samples as possible and reduce the risk of tracking error accumulation. Note that both the methods of [42] and [19] update the appearance models by fine-tunning the CNNs online. In [50], the interdependencies between the particles are exploited to jointly learn the particle representations for visual tracking. In [49], a circulant sparse tracker, which exploits the circulant structure of the target templates for visual tracking by efficiently solving the optimization problem in the Fourier domain, has been proposed. Zhang *et al.* [51] have proposed a correlation particle filter for visual tracking by exploiting the advantages of individual filters to handle the problems of occlusions, large-scale variations and high computational complexity. In [52, 53], by exploiting the interdependencies between different features, the authors have proposed a tracker in which the correlation filters are learned jointly. Further, Ou *et al.* [26] have proposed a visual tracking algorithm based on online representative sample selection scheme via non-negative least square to construct the templates, and then predict the optimal candidate using a score function. In deep relative tracking [10], robust visual features and an effective nonlinear ranking function are learned to exploit the relative relationship among image patches for object appearance modelling. A

part-to-target regression model has been proposed in [11] to exploit the context information and spatial structure of the parts to find the target location.

In this paper, a structural local 2D-DCT sparse appearance model is proposed to exploit the energy compaction property of 2D-DCT in the object appearance model by using only a few 2D-DCT coefficients. In addition, it is proposed to reconstruct a holistic image from the overlapped local patches that are obtained using the local patch dictionary and the sparse codes. Further, a robust occlusion map generation is proposed using the reconstructed holistic image, and the pooled feature vector. Also, it is proposed to compute the threshold for occlusion detection automatically for each sequence. In addition, the highest confident occlusion-free sample among the cumulated samples is used to reconstruct the image for the template update. Further, it is proposed to replace the template that contributes least in representing the previous tracking results with the reconstructed image obtained after incremental subspace learning. Also, it is proposed to use the patch occlusion ratio while computing the confidence of a candidate. Experiments conducted on two popular benchmark datasets with comparison to the state-of-the-art tracking methods bear out the competency and effectiveness of the proposed method for visual tracking.

The rest of this paper is organized as follows. Section 2 gives the background information and reviews the related work available in the literature on visual tracking. Section 3 describes the object representation using structural local 2D-DCT sparse appearance model. The holistic image reconstruction from an overlapped local patches is explained in Section 4 followed by the proposed tracking algorithm in Section 5. Experimental results for the two popular benchmark datasets are demonstrated and discussed in Section 6 followed by a conclusion in Section 7.

2 Background and Related Work

The success of sparse representation in vision applications, such as image denoising [9], image classification [46] and face recognition [43] has motivated Mei *et al.* to propose a l_1 tracker [24]. In l_1 tracker, a set of target templates and trivial templates are used to model the object appearance. The target location is determined by solving one l_1-minimization for each particle sample. Further, the minimum error bounded efficient l_1 tracker with occlusion detection [25] and the accelerated proximal gradient algorithm (L1APG) [3] were proposed to improve the l_1 tracker in terms of both the speed and accuracy. As most of these methods use holistic representation, they fail/drift during occlusions. Adam *et al.* [1] proposed a tracking method based on fragments, where each fragment is tracked by measuring the local regional similarity. Finally, the target location is found by using the vote maps of the tracked fragments. Jia *et al.* [17] proposed a visual tracking algorithm based on an adaptive structural local sparse appearance (ASLA) model by exploiting both the partial and the spatial information of the target. Similar to ASLA, Dai *et al.* [7] proposed a part-based sparsity model for visual tracking but with non-overlapped patches to model the object appearance. Then the target template set for each patch is updated dynamically. Further, a tracking algorithm based on support vector machine (SVM) is proposed by exploiting the aligned structural local sparse features [41]. Concurrently, a robust local sparse tracker with global consistency constraint is proposed in [47] to alleviate the problem of drifting when the target patch is similar to that of the background. Wang *et al.* [37] proposed a weighted local cosine similarity (WLCS) to measure the similarity between the target and the candidates, and then developed a tracking algorithm based on the local model. In [38], an inverse sparse tracking algorithm is proposed by employing a locally weighted distance metric to measure the similarity between the target and the candidates. As

the algorithm employs a local template update scheme, the unoccluded local parts are updated while the occluded ones are discarded during heavy occlusion.

In the literature, only a few attempts have been made to exploit the properties of DCT for visual tracking [5, 20, 22] in spite of its success in a wide range of vision applications such as image retrieval [14], image fusion [30], face recognition [13, 34], video object segmentation [4] and video caption localization [54]. In both [22] and [5], the features extracted from the DCT coefficients are used to find the target location by measuring the similarity between the target and candidates, but there is no update of appearance model in [22]. At the same time, Li *et al.* [20] proposed a compact 3D-DCT based object representation and its incremental learning for robust visual tracking (IL3DDCT). The likelihood is evaluated using a signal reconstruction-based similarity measure. In contrast to these methods, which use only DCT for the appearance model, the proposed method exploits both the sparse representation and 2D-DCT to model the appearance of the object. Further, the proposed method uses the local appearance model in contrast to the holistic templates used in the above methods. Also, a robust occlusion detection and an observation model update is proposed to reduce the effects of occlusion/outliers on the tracking algorithm.

3 Structural Local 2D-DCT Sparse Appearance Model

The robustness and effectiveness of local representations, when the objects undergo pose change, deformation and partial occlusion [1, 7, 17], has motivated us to propose a structural local 2D-DCT sparse appearance model for visual tracking. The proposed algorithm has some similarity to ASLA [17] in the use of local sparse representation, but differs in the domain in which sparse representation is applied. ASLA directly uses the pixel intensities in the local patches for object appearance model, whereas the proposed method uses the DCT coefficients of those pixel intensities in the local patches. Further, the proposed method reconstructs an image from the overlapped patches and sparse codes to detect the occlusions. Even though ASLA can handle partial occlusions due to local representations, ASLA does not have any mechanism for occlusion detection. Also, the proposed method differs from ASLA in terms of the appearance model update. Most of the methods in the literature use pixel intensities in the holistic templates [3, 24] or in the local patches [7, 17, 47] for object appearance modeling using sparse representation. But the proposed method explores it in the transform domain (2D-DCT). It is well known that a fraction of the 2D-DCT coefficients are sufficient to represent an image with less visual distortion due to the energy compaction property of the 2D-DCT [27]. In this paper, the energy compaction property of the 2D-DCT is exploited by reducing the number of elements/coefficients in the candidate samples and the dictionary to lower the computational cost of l_1-minimization. Most of the local appearance models use the patch reconstruction error to detect the occlusion [7, 47]. In contrast, the proposed method uses the holistic reconstruction error to detect the occlusion. The holistic reconstruction error is obtained from the holistic image reconstructed from the overlapped local patches and the sparse codes.

For a given target candidate, the overlapped local patches $\mathbf{P}_i$ inside the target region are extracted with a spatial layout as shown in Fig. 1. Then, 2D-DCT of these patches are computed followed by zigzag scanning, whose order is akin to the one defined in [27]. This gives a matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N] \in \mathbb{R}^{d \times N}$ for a given candidate, where N denotes the number of local patches extracted within the target region and d is the number of pixels in a patch. As each fixed part of the target object is represented by one local patch, the complete holistic structure of a target candidate can be represented by all these N local patches with a fixed spatial relationship. Similar procedure is followed for every template in a given set of target templates

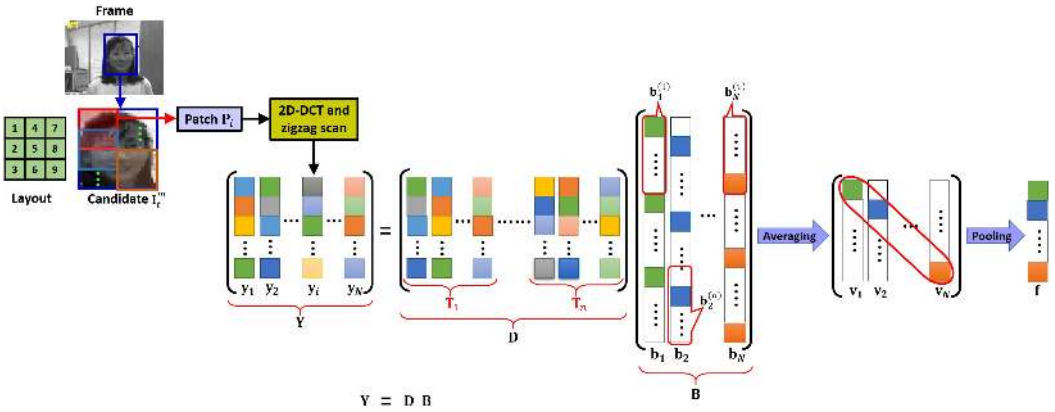


Fig. 1 Structural local 2D-DCT sparse appearance model: illustration of a local patch $\mathbf{P}_i$ extraction followed by 2D-DCT, zigzag scanning, averaging of sparse coefficients obtained via l_1-normalization and alignment pooling of features. 2D-DCT of each local patch (where the first patch is denoted in red, second one in blue, and the last one in brown rectangle) is sparsely represented by the patch dictionary $\mathbf{D}$ (in 2D-DCT domain) with a sparse vector. These sparse coefficients are averaged and pooled to represent a target object.

$\mathbf{T} = [\mathbf{T}_1, \mathbf{T}_2, ..., \mathbf{T}_n]$ to create a dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{(n \times N)}] \in \mathbb{R}^{d \times (n \times N)}$, where n is the number of target templates. Here, the local patches of the target templates are used as the dictionary atoms to encode the local patches inside the candidate regions, where all these local patches are in 2D-DCT domain. As these local patches are obtained across many templates, the resultant dictionary captures the generality of the different templates and hence, it is able to represent various forms of target parts [17, 31].

In sparse representation, only a few basis elements of the dictionary with different coefficients are sufficient to represent a local patch inside the target region and this is achieved by solving the following minimization problem:

$$\min_{\mathbf{b}_i} \|\mathbf{y}_i - \mathbf{D}\mathbf{b}_i\|_2^2 + \lambda \|\mathbf{b}_i\|_1, \quad \text{s.t. } \mathbf{b}_i \succeq 0, \tag{1}$$

where $\mathbf{y}_i \in \mathbb{R}^{d \times 1}$ represents the 2D-DCT of the i-th local patch, $\mathbf{b}_i \in \mathbb{R}^{(n \times N) \times 1}$ is the sparse code of that local patch, and the constraint $\mathbf{b}_i \succeq 0$ indicates that all the elements of $\mathbf{b}_i$ are non-negative. Now, the sparse codes of the given target candidate is given by $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_N]$. Further, the sparse coefficients of each local patch $\mathbf{b}_i$ are divided into several segments depending on the template that each element of the vector belongs to, i.e., $\mathbf{b}_i^\top = [\mathbf{b}_i^{(1)\top}, \mathbf{b}_i^{(2)\top}, , \mathbf{b}_i^{(n)\top}]$, where $\mathbf{b}_i^{(k)} \in \mathbb{R}^{N \times 1}$ indicates the k-th segment of the sparse coefficient vector $\mathbf{b}_i$ corresponding to the template $\mathbf{T}_k$ in the given target template set $\mathbf{T}$. From these segmented sparse coefficients $\mathbf{b}_i^{(k)}$, a normalized feature vector $\mathbf{v}_i \in \mathbb{R}^{N \times 1}$ for the i-th local patch is obtained as

$$\mathbf{v}_i = \frac{1}{G} \sum_{k=1}^{n} \mathbf{b}_i^{(k)}, \quad i = 1, 2, ..., N, \tag{2}$$

where G is a normalization term, which makes all the contributions from the templates sum to unity. Thus, for a given candidate, all the normalized feature vectors of the local patches within a candidate region form a square matrix $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_N] \in \mathbb{R}^{N \times N}$. Since a single local patch captures only some local appearance of the object, the whole object modeling requires pooling of information from these normalized feature vectors. Here, alignment pooling is chosen due

to its capability of using full structural information contained in the dictionary and precisely locating the target object [17]. Even though each local patch at a given position of the candidate is represented by patches at different positions of the templates, the local appearance of a patch with some appearance variation in a candidate is correctly represented by the patches at the same positions of the templates. That is, the top left corner patch of the object in Fig. 1 can be represented precisely by the top left corner patches of the templates. This is achieved by considering only the diagonal elements of the square matrix $\mathbf{V}$ as the pooled feature vector $\mathbf{f} \in \mathbb{R}^{N\times 1}$, given by

$$\mathbf{f} = \text{diag}(\mathbf{V}) \tag{3}$$

This feature vector $\mathbf{f}$ not only captures the target structure with a fixed spatial relationship but also reflects the similarity between the candidate and the target template.

4 Holistic Image Reconstruction from an Overlapped Local Patches

In the proposed method, the overlapped local patches are used for the object representation rather than holistic templates due to their robustness to pose change, deformation and occlusion. The overlapped local patches of size 16×16 with an overlap of 8 pixels are extracted from an image of size 32×32 as per the layout shown in Fig. 1. Now, the extracted local patches are concatenated as per their spatial relationship to obtain a overall block of size 48×48 (as shown in left part of Fig. 2). For clarity, the original image of size 32×32 is assumed to be divided into a sub-blocks of size 8×8 as shown in right part of Fig. 2. Now, the first 16×16 patch extracted from the top-left corner of the image, denoted as block **1** in left part of Fig. 2, is comprised of four 8×8 sub-blocks denoted as sub-block **A, B, E** and **F** in right part of Fig. 2. Similarly, the second block, denoted as block **2**, is comprised of four 8×8 sub-blocks **B, C, F** and **G**, and so on. Please note that some of the nomenclature/symbols used in this section may have different meaning outside this section.

To understand the reconstruction procedure, the 8×8 sub-blocks in Fig. 2 are divided into three groups depending on the number of overlaps they have with the neighboring 16×16 blocks. The first group consisting of four sub-blocks (**A, D, M** and **P**) has no overlaps with 16×16 blocks, and are denoted as no overlapping blocks (NOB) (blue color blocks in the right part of Fig. 2). The second group consisting of 8 sub-blocks (**B, C, E, H, I, L, N** and **O**) has an overlap with two 16×16 adjacent blocks, and are denoted as two overlapping blocks (TOB) (green color blocks in the reconstructed image of Fig. 2). Similarly, the last group consisting of four sub-blocks (**F, G, J** and **K**) has an overlap with four 16×16 adjacent blocks, and are denoted as four overlapping blocks (FOB) (orange color blocks in the reconstructed image of Fig. 2). Since the NOB group has no overlaps, the pixels in the sub-blocks **A, D, M** and **P** of the reconstructed image are copied directly from the *II*-quadrant of block **1**, *III*-quadrant of block **3**, *I*-quadrant of block **7** and *IV*-quadrant of block **9**, respectively. However, the TOB group has two overlapping blocks and hence the pixels in the sub-blocks **B, C, E, H, I, L, N** and **O** of the reconstructed image are computed from the corresponding two 16×16 overlapping blocks. For example, the sub-block **N** is reconstructed from *IV*-quadrant of block **7**, *I*-quadrant of block **8**. Therefore, all the pixels in the *IV*-quadrant of block **7** (denoted as P_N^7) and *I*-quadrant of block **8** (denoted as P_N^8) are used to find the pixel values in sub-block **N** (denoted as P_N) by

$$P_N = \frac{a\,P_N^7 + b\,P_N^8}{a+b}, \tag{4}$$

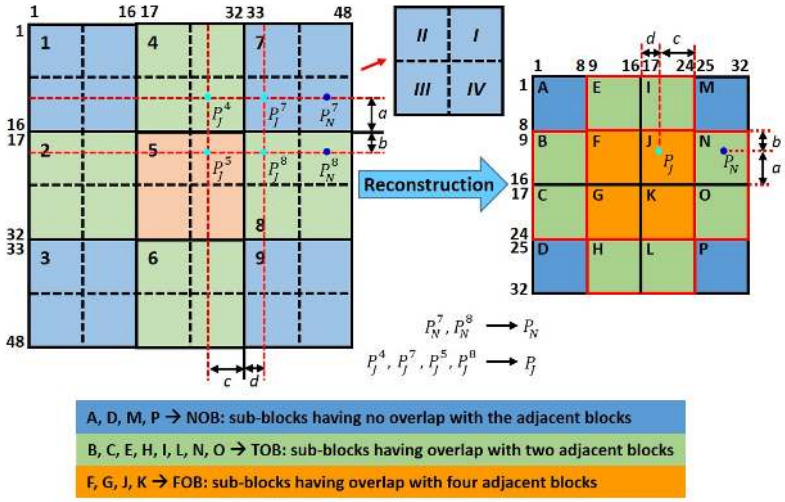


Fig. 2 Holistic image reconstruction from overlapped local patches.

where a and b are the distances as shown in Fig. 2. Similarly, all the pixels in the sub-blocks **F, G, J** and **K** belonging to FOB group are computed from the pixels of the corresponding four 16×16 overlapping blocks using

$$P_J = \frac{c}{c+d}\left(\frac{a\,P_J^4 + b\,P_J^5}{a+b}\right) + \frac{d}{c+d}\left(\frac{a\,P_J^7 + b\,P_J^8}{a+b}\right), \tag{5}$$

where a, b, c and d are the distances as shown in Fig. 2, P_J^4, P_J^5, P_J^7 and P_J^8 are the pixel values from the blocks **4, 5, 7** and **8**, respectively, and P_J is the reconstructed pixel belonging to block **J**. The reconstructed holistic image $\mathbf{I}_r$ of the *Jogging-2* and *Woman* sequences is shown in Fig. 3 to illustrate the effectiveness of the proposed image reconstruction without introducing any errors/artifacts during transition from one patch to another.

5 Proposed Tracking Algorithm

In the proposed framework, motion of the target is estimated using a Markov model with hidden state variables [16]. Let $\mathbf{x}_t$ represent a state variable describing the affine motion parameters of a target at time t. Given a set of observed images $\mathcal{I}_t = \{\mathbf{I}_1, ..., \mathbf{I}_t\}$ at time t, the posterior probability $p(\mathbf{x}_t|\mathcal{I}_t)$ can be estimated recursively by the Bayesian theorem,

$$p(\mathbf{x}_t|\mathcal{I}_t) \propto p(\mathbf{I}_t|\mathbf{x}_t) \int p(\mathbf{x}_t|\mathbf{x}_{t-1})\, p(\mathbf{x}_{t-1}|\mathcal{I}_{t-1})\, d\mathbf{x}_{t-1} \tag{6}$$

where $p(\mathbf{I}_t|\mathbf{x}_t)$ represents the observation model, and $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ represents the dynamic model. In this work, an affine transformation with six parameters is adopted to model the target state $\mathbf{x}_t = (x_t, y_t, \theta_t, s_t, \alpha_t, \phi_t)$, where $x_t, y_t, \theta_t, s_t, \alpha_t, \phi_t$ denote horizontal and vertical translations, rotation angle, scale, aspect ratio and skew direction at time t, respectively. The dynamic model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ describes the temporal correlation of states between two consecutive frames and is modeled by Gaussian distribution assuming the affine parameters to be independent. That is,

$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}_{t-1}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ denotes a diagonal covariance matrix, whose elements are the variances of the affine parameters. These affine parameters are used to crop a sub-image from the current frame and then normalized to the size $w \times h$. The dynamic model randomly selects M samples of the state variable $\mathbf{x}_t$ given the state $\mathbf{x}_{t-1}$ at $t-1$, which are used to generate candidate samples $\mathbf{I}_t^m$, where $m = 1, 2, ..., M$. Among these M states, the optimal state of the tracked target $\mathbf{x}_t$ at time t is determined by solving the following MAP estimation:

$$\hat{\mathbf{x}}_t = \arg\max_{\mathbf{x}_t^m} p(\mathbf{I}_t^m|\mathbf{x}_t^m)\, p(\mathbf{x}_t^m|\mathbf{x}_{t-1}),\ m = 1, 2, ..., M \tag{7}$$

where $\mathbf{x}_t^m$ denotes the m-th sample of the state $\mathbf{x}_t$ and $\mathbf{I}_t^m$ indicates an image sample observed by $\mathbf{x}_t^m$.

In the proposed method, the 2D-DCT of the overlapped local patches inside the target region are used to model the object appearance. These overlapped local patches can be represented by a very few low-frequency 2D-DCT coefficients, which can preserve the image information in a patch very well due to the energy compaction property of the 2D-DCT. Hence, only r lower frequency 2D-DCT coefficients are considered out of d coefficients in all the patches of the candidates and the dictionary while solving the l_1-minimization problem in Eq. (1). The confidence scores C^m for all the candidates are computed from the sparse codes using Eqs. (2), (3) as [17]

$$C^m = \sum_{i=1}^{N} \mathbf{f}_i^m \tag{8}$$

In visual tracking based on particle filter framework, the confidence of the each particle is given by its observation likelihood, defined as

$$p(\mathbf{I}_t^m|\mathbf{x}_t^m) \propto C^m \tag{9}$$

Finally, the optimal state of the target is estimated using Eq. (7). Further, the observation models are adapted to handle the appearance change of the target by incrementally updating the template set and dictionary, as discussed in the next subsection.

Algorithm 1 Proposed Tracking Algorithm

Input: Target object is labeled in the first frame and its initial state is $\mathbf{x}_0$, number of templates n, number of local patches N

1: Collect a set of n templates, $\mathbf{T} = [\mathbf{T}_1, \mathbf{T}_2, ..., \mathbf{T}_n]$, using kd-tree to model the object appearance.
2: From every template in $\mathbf{T}$, extract the N overlapped local patches $\mathbf{P}_i$, compute the 2D-DCT followed by zigzag scanning to create a dictionary $\mathbf{D}$.
3: **for** $t > n$ **do**
4: Sample M candidate states $\{\mathbf{x}_t^1, \mathbf{x}_t^2, ..., \mathbf{x}_t^M\}$ from $\mathbf{x}_{t-1}$.
5: For every candidate state $\mathbf{x}_t^m$, extract the corresponding image sample $\mathbf{I}_t^m$ and its local patch matrix in 2D-DCT domain $\mathbf{Y}_t^m$ as explained in section 3.
6: For all the candidate samples $\mathbf{I}_t^m$, compute sparse codes $\mathbf{b}_i$ for each local patch vector $\mathbf{y}_i$ in the candidate sample matrix $\mathbf{Y}_t^m$ using Eq. (1) by considering only r number of DCT coefficients in both candidates and the dictionary.
7: Compute the pooled feature vector $\mathbf{f}^m$ and the confidence score C^m for all the candidates using Eqs. (2), (3) and (15), respectively.
8: Find the optimal state of the tracked target $\hat{\mathbf{x}}_t$ using the Eqs. (9) and (7).
9: Update the template set $\mathbf{T}$ and dictionary $\mathbf{D}$ as described in section 5.1 for every 5 frames.
10: **end for**

Output: Target state $\hat{\mathbf{x}}_t$ at time t and the updated template set $\mathbf{T}$ and dictionary $\mathbf{D}$

5.1 Observation Model Update

The update of observation model is very much essential to handle the appearance variations of the object, but the update with imprecise samples will cause tracking drift due to model degradation. Therefore, the imprecise samples should be avoided during the model update. Even though ASLA is efficient during partial occlusion due to its local appearance model, its template update mechanism has a drawback during occlusion. That is, the tracking results, which are occluded, are employed directly for an incremental subspace learning thereby degenerating the PCA subspace. As the image reconstructed from the degenerated PCA subspace is used for the template update, there are chances of appearance model degradation (see Fig. 4a) resulting in a tracking drift. To address this issue, it is proposed to detect and generate a robust occlusion map, which is used to modify the occluded samples before the appearance model update.

For the occlusion map generation, all the 2D-DCT coefficients are used in both the dictionary $\mathbf{D}$ and the tracked candidate matrix $\hat{\mathbf{Y}}$ to compute the sparse codes $\hat{\mathbf{B}}$ using Eq. (1). The local appearance of a patch in a target candidate is correctly represented by the patches at the same positions of the templates. Hence, the sparse codes corresponding to the respective patches are only used to compute the overlapped local patches. These overlapped patches are used to reconstruct the holistic image $\mathbf{I}_r$ as described in section 4 and then the holistic reconstruction error $\mathbf{E} = \hat{\mathbf{I}} - \mathbf{I}_r$ is computed, where $\hat{\mathbf{I}}$ is the target image of the tracking result. Further, the holistic reconstruction error $\mathbf{E}$ is used to generate a binary occlusion map $\mathbf{O}_1$ using Eq. (10) indicating one for occluded pixels and zero for non-occluded pixels.

$$\mathbf{O}_1 = \begin{cases} 1, & \text{if } |\mathbf{E}| \geq O_{Thr} \\ 0, & \text{otherwise,} \end{cases} \tag{10}$$

where O_{Thr} is a precomputed threshold that decides whether the pixel is occluded or not. In order to compute O_{Thr}, it is assumed that all the elements of the reconstruction error $\mathbf{E}$ will be in the "normal range" and follow the Gaussian distribution in the absence of occlusion. But during occlusions, the occluded elements of the reconstruction error $\mathbf{E}$ will probably exceed the "normal range". Therefore, by knowing the "normal range" of the reconstruction error in the initial frames (from 1 to n) of the respective sequence, the value of the O_{Thr} is computed as

$$O_{Thr} = \frac{c_1}{n} \sum_{f=1}^{n} \mathsf{std}\left(\mathbf{E}_f\right) \tag{11}$$

Here, each target template in a template set $\mathbf{T}$ is used as the candidate sample and the remaining $n-1$ templates are used as the dictionary in a round-robin fashion to compute O_{Thr}. In general, the occlusion is a large connected region as opposed to the random noises or object appearance variations, whose region is very small. Hence, the occlusion map is updated to retain only the large connected region by applying a morphological operations and a connected component analysis.

In order to increase the robustness of the occlusion detection method, it is proposed to identify the patch of the tracked candidate with no contribution from the respective patches of the dictionary $\mathbf{D}$ using the pooled feature vector $\hat{\mathbf{f}}$. If there is no contribution from the respective patches of the dictionary $\mathbf{D}$, the respective element of the pooled feature vector $\hat{\mathbf{f}}$ will be zero. That is, $\hat{\mathbf{f}}_i = 0$ indicates that there is no contribution from the i-th patch of all the templates in representing the i-th patch of the tracked candidate, and this happens when the i-th patch is occluded fully. Then, a binary occlusion map $\mathbf{O}_2$ is generated by indicating one in the respective pixel locations of the corresponding patch.

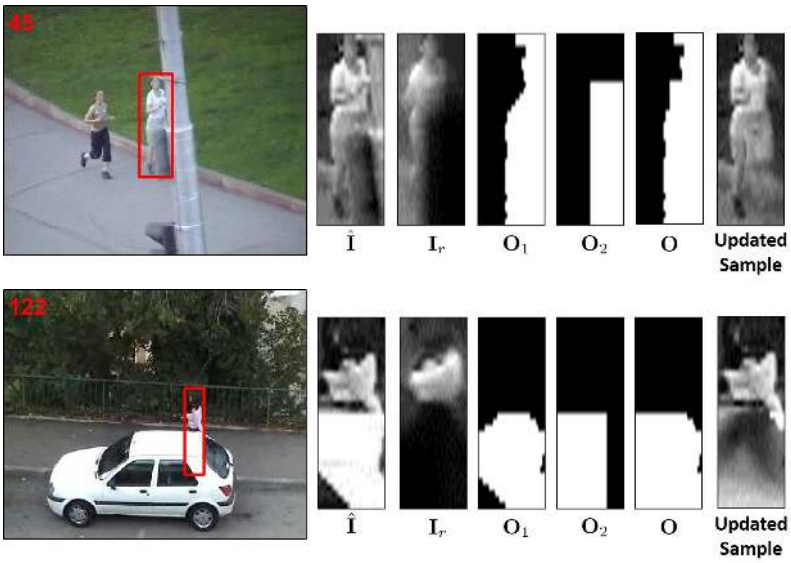


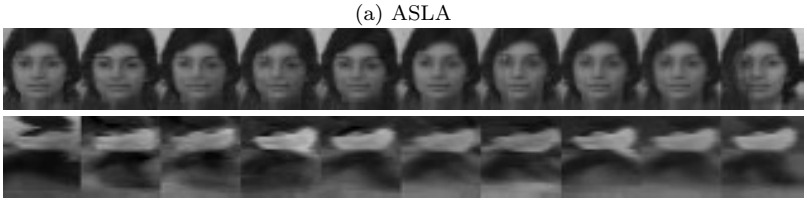
Fig. 3 Some representative cases of *Jogging-2* and *Woman* sequences to illustrate the effectiveness of the proposed holistic image reconstruction, the robust occlusion map generation ($\mathbf{O}_1$, $\mathbf{O}_2$, $\mathbf{O}$), and the updated sample.

Now, the two occlusion maps $\mathbf{O}_1$ and $\mathbf{O}_2$ are combined to generate a final occlusion map $\mathbf{O}$ by performing a logical-OR operation. This makes the occlusion map more robust, so that the chances of appearance model deterioration due to occlusion could be reduced. Fig. 3 shows the occlusion maps $\mathbf{O}_1$, $\mathbf{O}_2$ and $\mathbf{O}$ for *Jogging-2* (#45) and *Woman* (#122) sequences. Further, a occlusion ratio τ is computed as the ratio of the number of ones in $\mathbf{O}$ to the total number of elements in $\mathbf{O}$. This occlusion ratio τ indicates the amount of occlusion in the tracked candidate. The occlusion ratio τ decides whether the update of the observation model with the tracked sample is full or partial or not, with the help of two thresholds τ_1 and τ_2. In the absence of occlusion (when $\tau < \tau_1$), the tracked sample is used directly for the model update (full update). During partial occlusion (when $\tau_1 < \tau < \tau_2$), the occluded pixels in the tracked sample are replaced with the corresponding pixels from the previously updated mean $\boldsymbol{\mu}$ to get a updated sample. This updated sample is free from occlusion and is used in a model update. During severe occlusion (when $\tau > \tau_2$), the tracked sample is not used for model update. Fig. 3 shows the updated tracked sample, which is free from occlusion, for *Jogging-2* (#45) and *Woman* (#122) sequences. It is observed from Fig. 3 that the combined occlusion map $\mathbf{O}$ is more robust than the individual ones $\mathbf{O}_1$ and $\mathbf{O}_2$, and make the updated sample free from occlusion with the proposed appearance model update. The updated tracked samples, which are free from occlusion, are cumulated for an incremental subspace learning [29]. This incremental learning not only adapts to the target appearance variation but also preserves the common visual information in the collected observations. The proposed method uses trivial templates along with PCA basis vectors $\mathbf{U}$ to estimate the target $\mathbf{p}$, as given by

$$\mathbf{p} = \mathbf{U}\mathbf{q} + \mathbf{e} = [\ \mathbf{U} \quad \mathbf{I}\ ] \begin{bmatrix} \mathbf{q} \\ \mathbf{e} \end{bmatrix} \tag{12}$$

where $\mathbf{I}$ is the identity matrix representing trivial templates, $\mathbf{q}$ denotes the coefficients of the PCA basis vectors $\mathbf{U}$, and $\mathbf{e}$ represents the pixels in $\mathbf{p}$ that are outliers or corrupted. Unlike

(a) ASLA



(b) LDSAMOD

Fig. 4 Some representative cases of *Faceocc1* (#700) and *Woman* (#180) to illustrate the effectiveness of the proposed observation model update compared to that of ASLA.

ASLA, which uses the latest tracked sample as $\mathbf{p}$, the proposed method choses the updated tracked sample with highest confidence score among the cumulated samples as $\mathbf{p}$, which is free from occlusion. Note that when $\tau > \tau_2$, the tracked sample with severe occlusion is not cumulated and hence its chances of considering as $\mathbf{p}$ is zero. Since the error is arbitrary and sparse, Eq. (12) is solved by l_1-minimization, and an image $\hat{\mathbf{p}}$ is reconstructed from the PCA basis vectors $\mathbf{U}$ along with its coefficients $\mathbf{q}$. This ensures the reconstructed image is free from corruption and outliers, as the coefficients of trivial templates (due to noise or outlier) are excluded for image reconstruction. The reconstructed image $\hat{\mathbf{p}}$ is then used for updating both the template and the corresponding dictionary atoms in $\mathbf{D}$ as discussed in the next paragraph.

It is known that the sparse codes $\hat{\mathbf{B}}$ of the tracked candidate represent the contributions from all the patches of all the templates. Also, the local appearance of a patch with some appearance variation in the tracked candidate is correctly represented by the patches at the same positions of the templates. Hence, the contribution $\hat{\mathbf{h}}_k$ of each template in representing the tracking result $\hat{\mathbf{I}}$ is computed by considering only the contributions from the respective patches of the template using Eq. (13).

$$\hat{\mathbf{h}}_k = \sum_{i=1}^{N} \hat{\mathbf{B}}\left(N[k-1]+i, i\right), \quad k = 1, 2, ..., n, \tag{13}$$

While cumulating the tracked samples, the respective confidence $\hat{C}$ and the template contribution scores $\hat{\mathbf{h}}_k$ are also cumulated. From the cumulated confidence scores, the highest confidence score is found and its corresponding updated tracked sample is used as $\mathbf{p}$ in Eq. (12). Further, from the cumulated template contribution scores, the location $\hat{k}$ of the template to be replaced is found using the Eq. (14).

$$\hat{k} = \arg\min_{k} \sum_{t \in [t-4:t]} \mathbf{h}_k(t) \tag{14}$$

With this replacement strategy, the template, which is contributing least to the representation of the previous 5 tracking results, is replaced with the reconstructed image $\hat{\mathbf{p}}$. This is done with an assumption that the template with the least contribution score may have an old appearance of the object, which may be outdated, and hence cannot contribute significantly in representing the target candidate. After updating the template set with $\hat{\mathbf{p}}$, the corresponding

dictionary atoms are also updated. To illustrate the effectiveness of the proposed observation model update, the template dictionary for the proposed method and ASLA is shown in Fig. 4. From Fig. 4, it is observed that the template set of ASLA gets corrupted over the time, but this is not so with the proposed method. Unlike [24], which considers only the current tracking result (may be severely occluded) to find the template contribution score, the proposed method considers previous 5 tracking results (which are not severely occluded with $\tau < \tau_2$) to find the average template contribution score. As [24] directly uses the occluded/corrupted tracking result for the model update, there are chances of observation model degradation. However, in the proposed method, $\hat{\mathbf{p}}$ used for the model update is free from the occlusion and outliers/corruptions, as they are removed before and after an incremental subspace learning.

In the proposed method, instead of using the sum of pooled features to find the confidence score C^m by giving equal weights to each patch (as in ASLA), it is proposed to compute the confidence score C^m using weighted sum of the pooled features. This is done by assigning different weights to each patch depending on its patch occlusion ratio τ_{p_i}. Now the confidence score C^m is rewritten as

$$C^m = \sum_{i=1}^{N} \tau_{p_i} \mathbf{f}_i^m \tag{15}$$

where τ_{p_i} is the ratio of occluded pixels to that of total pixels in a patch.

6 Experimental Results

The proposed algorithm is implemented in MATLAB and its performance is evaluated using 50 and 60 challenging sequences available in the OTB-50 [44] and VOT2016 [18] datasets, respectively. These sequences cover most of the real-life challenging situations in object tracking, such as motion blur due to fast movement, pose variation, complex background, varying lighting conditions, low contrast, scale change, heavy occlusion, in-plane and out-of-plane rotation. In the proposed method, each image observation is resized to 32×32 pixels and then local patches of size 16×16 are extracted with an overlap of 8 pixels. Therefore, each target region is cut into $N = 9$ overlapping patches. In the proposed method, SPAMS package [23] is used for l_1-minimization and the regularization constant λ is set to 0.01. 10 eigenvectors are used in an incremental subspace learning and the observation model is updated for every 5 frames. Considering the trade-off between effectiveness in tracking and computational efficiency, 600 particles are sampled using a particle filter. The constant c_1 used to compute threshold O_{Thr} in (11) is set to 4. The occlusion ratio thresholds τ_1 and τ_2 are set to 0.1 and 0.65, respectively. In the initial 10 frames, kd-tree is used to obtain the tracking results, and from these tracking results $n = 10$ target templates are extracted for the generation of the dictionary $\mathbf{D}$. The number of DCT coefficients considered in each patch, r, is set to 64 while computing the confidence score C^m for all the candidates, except during occlusions when $\tau > \tau_2$, it is set to 256. By reducing the size of the dictionary as well as that of the candidate samples, the proposed method has achieved a speed of 2.18 fps (including the time required for the computation of the 2D-DCT and the proposed occlusion detection) as compared with 1.86 fps required by ASLA. This is a 17.2% increase in the speed compared to that of ASLA in spite of having to compute the 2D-DCT coefficients and the occlusion detection.

The performance of the proposed method is evaluated against several recent state-of-the-art algorithms based on the particle filter framework for a fair comparison. The considered algorithms are IVT [29], L1 accelerated proximal gradient (L1APG) [3], ASLA [17], sparse prototype tracker (SPT) [39], weighted residual minimization in PCA subspace for visual tracking

(WRMPCA) [33], visual tracking via bilateral 2DPCA and robust coding (B2DPCA) [32], visual tracking via least soft-threshold squares (LSST) [40], locally weighted inverse sparse tracker (LWIST) [38], visual tracking via weighted local cosine similarity (WLCS) [37], robust object tracking via probability continuous outlier model (PCOM) [36], IL3DDCT [20] and DLT [42]. Note that the codes of the trackers are downloaded from the respective authors' website and evaluated on both the OTB-50 and VOT2016 benchmark sequences for a fair comparison with the proposed method. For these evaluations, the parameter $\mathbf{\Sigma}$ (diagonal covariance matrix) of the particle filter is set to be like that used in OTB-50 [44], i.e. $\mathbf{\Sigma} = (4, 4, 0.01, 0.0, 0.005, 0)^2$.

6.1 Performance Evaluation Methodology

In general, two frame-based metrics, namely, overlap rate (OR) and center location error (CLE), are employed to evaluate the tracker in a given frame. Based on these basic metrics, the OTB-50 and VOT2016 methodologies derive other performance measures to analyze the tracking performance.

In OTB-50, the performance of a tracker for a given sequence is evaluated using the *success rate* and the *precision score*. The former is the ratio of successful frames whose OR is larger than a given threshold to the total frames in a sequence, whereas the later is the percentage of frames whose CLE is less than a given threshold distance of the ground truth. By using multiple thresholds, two curves are obtained showing how the threshold value affects the *success rate* and the *precision score*, and are called as *success plot* and *precision plot*, respectively, for a given sequence. Further, these *success* and *precision curves* are averaged over all the sequences to obtain the overall *success* and *precision plots*, respectively. In order to quantify the overall performance of a tracker, the area under curve (AUC) of the *success plot* or the *precision score* for the threshold of 20 pixels, is employed [44].

In VOT2016, the performance of a tracker is analyzed using the accuracy (A) and robustness (R). The accuracy is the average overlap between the predicted and ground truth bounding boxes during successful tracking periods. On the other hand, the robustness measures the number of times the tracker fails to track. In VOT2016, whenever a tracker predicts a bounding box with zero overlap with the ground truth, a failure is detected and the tracker is re-initialized. All the trackers are evaluated 15 times on each sequence and then per-frame accuracy is obtained as an average over these runs. Averaging per-frame accuracies gives per-sequence accuracy, while per-sequence robustness is computed by averaging failure rates over different runs [18].

6.2 Performance Evaluation on OTB-50

The performance of the proposed method is evaluated on the OTB-50 benchmark [44] consisting of 50 sequences with fully annotated attributes and compared with the state-of-the-art tracking algorithms using one-pass evaluation (OPE). The one-pass evaluation (OPE) uses the ground truth object location in the first frame and evaluates the tracker based on the average precision score or success rate. The *precision* and *success plots* of OPE for the various trackers averaging over the OTB-50 benchmark sequences are shown in Fig. 5. In *precision plot*, the *precision score* for the threshold of 20 pixels is used to rank the tracker. Whereas in *success plot*, AUC is used to rank the overall performance of the tracker. Both the *precision score* and AUC values have been shown along with the tracker name in the respective plots of Fig. 5. From Fig. 5, it is observed that the proposed method (LDSAMOD) outperforms the state-of-the-trackers ASLA, DLT, IL3DDCT, WRMPCA, IVT, LSST, L1APG and PCOM by 3.6%, 8%, 23.6%,

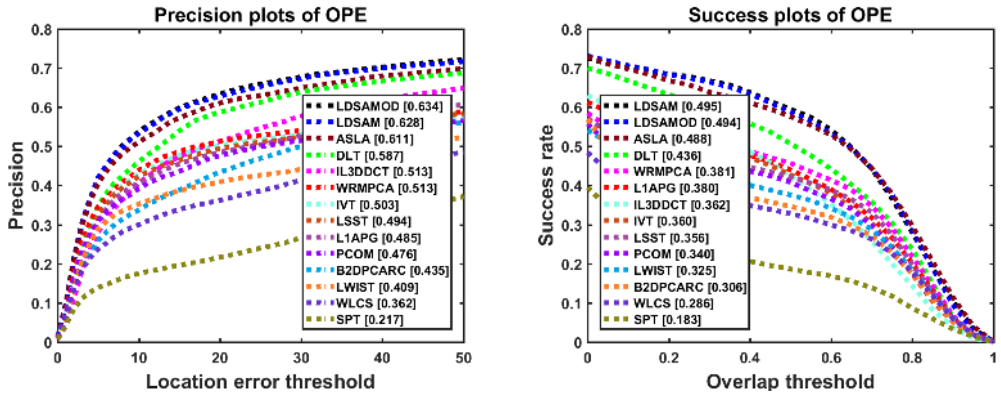


Fig. 5 Overall performance evaluation of the proposed method using *success* and *precision plots* of OPE.

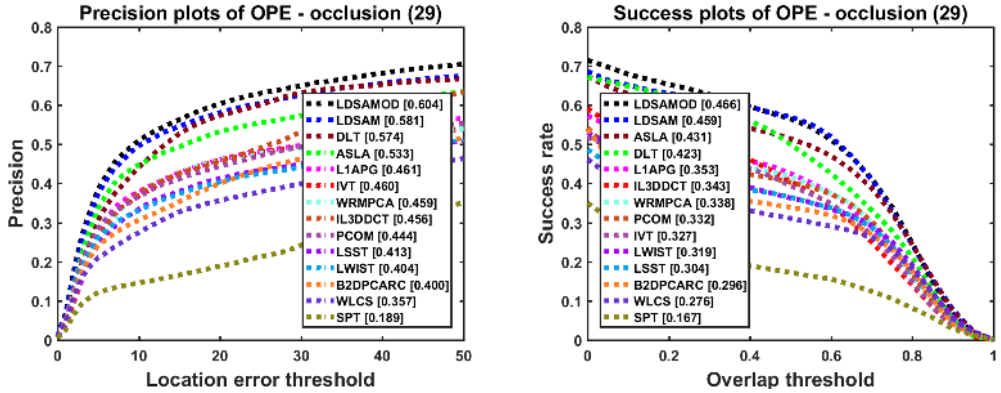


Fig. 6 Performance evaluation of the proposed method using *success* and *precision plots* of OPE for sequences having occlusion. The values appearing in the title denotes the number of sequences associated with the respective attribute.

23.6%, 26%, 28.3%, 30.7% and 33.2%, respectively, in terms of *precision score*. A similar trend in performance is also observed in *success scores*. Overall, LDSAMOD and its previous version (LDSAM) [31] provide the best performance to that of the other methods in terms of *precision* and *success score*. The proposed method, LDSAMOD, is an improvement of LDSAM [31] with a robust occlusion detection, which is used to update the appearance model as well as to find the confidence score. Further, the OPE performance comparison of the proposed method for sequences having occlusion against the other trackers is shown in Fig. 6. From the Fig. 6, it is observed that the proposed method outperforms the state-of-the-trackers DLT, ASLA, L1APG, IVT, WRMPCA, IL3DDCT, PCOM and LSST by 5.2%, 12.9%, 23.6%, 31%, 31.6%, 32.4%, 36% and 46.2%, respectively, in terms of *precision score*. This trend in performance holds true for *success score* also. For sequences having occlusion, it is observed that the proposed method and its previous version have shown better performance than that of the other trackers, both in terms of *precision* and *success scores*.

6.3 Performance Evaluation using VOT2016

The average accuracy and robustness are used to evaluate the performance of the proposed method using VOT2016 benchmark [18] consisting of 60 sequences, which are per-frame annotated with several visual attributes. Further, the tracking results are ranked according to accuracy and robustness performance metrics, and are named as accuracy rank (A-Rank) and robustness rank (R-Rank), respectively. Table 1 shows the A-Rank and overlap comparison of the proposed method with that of the recent state-of-the-art tracking algorithms averaging over the VOT2016 sequences. These sequences contain challenging situations such as camera motion, illumination change, motion change, occlusion and size change. Similarly, Table 2 shows the performance comparison of the proposed method using R-Rank and failures averaging over the same challenging sequences. The last six columns of these two tables show the respective measures using different averaging methodologies. The mean and weighted mean are the averages of attributes in an equal or weighted manner, whereas pooled corresponds to the per-frame averaging of the super-sequence obtained by concatenating all of the sequences [18]. The best three results are shown in **(red, bold)**, (violet, underline) and *(blue, italic)* fonts for better comparison of the proposed tracker with the other state-of-the-art trackers. Note that as the trackers with statistically equivalent results are merged while ranking, the different trackers may have same A-Rank and R-Rank [18]. From Table 1, it is observed that the proposed method stands first in all the challenging sequences in terms of overlap, except for occlusion where it stood second, and illumination change, where it stood third. Also, it is observed in Table 2 that the proposed method ranks first in all the challenging sequences in terms of failures, except for motion change, where it stood second, and illumination change, where it stood fourth. Further, the overall performance of the proposed method is superior to that of other methods in terms of mean, weighted mean and pooled averaging of overlap and failures.

6.4 Qualitative Evaluation

For qualitative evaluation of the trackers, some tracking results on a subset of the OTB-50 benchmark sequences are shown in Fig. 7. In Fig. 7, the tracking results of all the trackers on the six exemplar image frames are shown for each selected sequence and these six frames are selected at regular intervals without any bias. The proposed LDSAMOD tracker successfully tracks the target in the all the frames of the *Doll*, *Faceocc2*, *Dudek*, *Fish*, *Girl*, *Freeman3*, *Jogging-2*, *Singer1*, *Walking2* and *Woman* sequences, which contain most of the real-time challenges such as pose change, partial occlusion, illumination change, scale change and out-of-plane rotation. This indicates the strong capabilities of the proposed method in handling these challenges. The proposed tracker has slightly drifted away in middle few frames (#163 to #412) of the *Girl* sequence and then starts tracking afterwards. It is also observed that the previous version of the proposed method, LDSAM, performs better in all the sequences except *Walking2* and *Woman* sequences, where the sequence undergoes severe occlusion along with scale and appearance change, in spite of using local appearance model. Similarly, even with local appearance model, ASLA fails to track the object in *Faceocc2*, *Jogging-2*, *Walking2* and *Woman*, where the sequence undergoes partial/severe occlusion. These failures in both ASLA/LDSAM is because of the appearance model update with the imprecise tracked samples without removing the occlusion. IVT tracks the object in most of the sequences except *Doll*, *Girl*, *Freeman3*, *Jogging-2* and *Woman* sequences, where as L1APG tracks the object completely in *Girl* and *Walking2* sequences and drifts away in *Faceocc2*, *Dudek* and *Fish* sequences in the last few frames. Further, WRMPCA fails to track the object completely in *Doll*, *Girl*,

Table 1 Accuracy Rank and Average Overlap Comparison of the Proposed Method with that of the Compared Trackers. **(red, bold)**, (violet, underline) and *(blue, italic)* indicate First, Second and Third Rankings, respectively.

Accuracy	label camera motion		label empty		label illum. change		label motion change		label occlusion		label size change		Mean		Weighted mean		Pooled	
Trackers	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap	A-Rank	Overlap
IVT	**1.00**	0.41	**1.00**	0.48	6.00	0.50	**1.00**	0.36	**1.00**	0.33	**1.00**	0.37	1.83	0.41	1.83	0.41	**1.00**	0.42
L1APG	**1.00**	0.43	**1.00**	0.50	10.00	0.45	**1.00**	0.37	**1.00**	0.35	5.00	0.33	3.17	0.40	3.17	0.41	**1.00**	0.42
ASLA	**1.00**	0.44	**1.00**	0.49	**1.00**	0.61	**1.00**	*0.38*	**1.00**	*0.36*	**1.00**	0.41	**1.00**	*0.45*	**1.00**	*0.44*	**1.00**	*0.44*
WRMPCA	**1.00**	0.40	**1.00**	0.49	5.00	0.52	**1.00**	0.37	**1.00**	0.31	**1.00**	0.37	1.67	0.41	1.67	0.41	**1.00**	0.42
B2DPCARC	**1.00**	0.42	**1.00**	0.47	*3.00*	0.54	**1.00**	0.36	**1.00**	0.35	**1.00**	0.37	*1.33*	0.42	*1.33*	0.41	**1.00**	0.42
SPT	*12.00*	0.31	11.00	0.36	9.00	0.45	*7.00*	0.27	3.00	0.29	*6.00*	0.31	8.00	0.33	8.00	0.32	11.00	0.32
WLCS	**1.00**	0.41	**1.00**	*0.50*	**1.00**	0.63	**1.00**	0.34	**1.00**	0.33	**1.00**	0.40	**1.00**	0.43	**1.00**	0.42	**1.00**	0.43
LWIST	**1.00**	0.42	**1.00**	0.49	**1.00**	**0.65**	**1.00**	0.34	**1.00**	0.34	**1.00**	*0.41*	**1.00**	0.44	**1.00**	0.42	**1.00**	0.43
PCOM	**1.00**	0.41	**1.00**	0.48	*3.00*	0.57	**1.00**	0.36	**1.00**	0.36	**1.00**	0.41	*1.33*	0.43	*1.33*	0.42	**1.00**	0.43
DLT	**1.00**	*0.44*	**1.00**	0.50	**1.00**	0.58	**1.00**	0.39	**1.00**	**0.38**	**1.00**	0.42	**1.00**	0.45	**1.00**	0.44	**1.00**	0.45
LSST	**1.00**	0.40	**1.00**	0.44	2.00	0.55	**1.00**	0.32	**1.00**	0.34	**1.00**	0.37	1.17	0.40	1.17	0.39	**1.00**	0.40
LDSAM	6.00	0.33	*12.00*	0.36	13.00	0.31	5.00	0.28	**1.00**	0.31	11.00	0.25	8.00	0.31	8.00	0.31	11.00	0.33
LDSAMOD	**1.00**	**0.45**	**1.00**	**0.51**	**1.00**	*0.61*	**1.00**	**0.39**	**1.00**	0.36	**1.00**	**0.42**	**1.00**	**0.46**	**1.00**	**0.45**	**1.00**	**0.46**

Table 2 Robustness Rank and Average Failures Comparison of the Proposed Method with that of the Compared Trackers. **(red, bold)**, (violet, underline) and *(blue, italic)* indicate First, Second and Third Rankings, respectively.

Robustness	label camera motion		label empty		label illum. change		label motion change		label occlusion		label size change		Mean		Weighted mean		Pooled	
Trackers	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures	R-Rank	Failures
IVT	4.00	114.67	*5.00*	59.40	**1.00**	9.47	3.00	101.87	*4.00*	40.00	*4.00*	52.47	3.50	62.98	3.50	78.10	*4.00*	264.47
L1APG	4.00	140.80	**1.00**	49.13	*4.00*	11.60	5.00	99.13	2.00	38.60	*4.00*	54.27	3.33	65.59	3.33	82.89	*4.00*	267.27
ASLA	**1.00**	92.80	3.00	*45.80*	**1.00**	**7.73**	3.00	*79.27*	2.00	33.93	2.00	40.20	*2.00*	*49.96*	*2.00*	*61.84*	2.00	*208.27*
WRMPCA	4.00	106.60	3.00	52.47	2.00	10.47	3.00	91.93	*4.00*	38.20	2.00	48.33	3.00	58.00	3.00	71.51	*4.00*	242.60
B2DPCARC	*6.00*	126.93	6.00	62.20	*4.00*	10.67	5.00	100.67	7.00	43.93	8.00	59.20	6.00	67.27	6.00	83.66	7.00	278.73
SPT	12.00	183.27	12.00	107.27	10.00	12.33	12.00	135.27	*4.00*	54.07	12.00	72.60	10.33	94.13	10.33	120.29	13.00	414.47
WLCS	*6.00*	164.73	11.00	89.53	**1.00**	9.13	9.00	124.13	*4.00*	46.67	8.00	60.87	6.50	82.51	6.50	105.86	11.00	355.27
LWIST	*6.00*	131.27	9.00	81.13	**1.00**	8.53	5.00	115.13	*4.00*	43.07	5.00	55.27	5.00	72.40	5.00	91.28	8.00	307.53
PCOM	*6.00*	127.00	*5.00*	57.67	**1.00**	*8.87*	*4.00*	102.33	*4.00*	41.20	*4.00*	53.73	4.00	65.13	4.00	81.61	5.00	273.93
DLT	**1.00**	*94.73*	**1.00**	42.20	**1.00**	10.07	**1.00**	**66.53**	2.00	*35.40*	2.00	*43.33*	1.33	48.71	1.33	60.06	**1.00**	201.47
LSST	*6.00*	135.60	9.00	77.87	*4.00*	11.40	5.00	115.20	*4.00*	45.60	9.00	61.47	6.17	74.52	6.17	93.11	9.00	315.20
LDSAM	11.00	167.13	11.00	114.20	**1.00**	10.00	11.00	131.40	*4.00*	51.67	12.00	76.60	8.33	91.83	8.33	117.10	11.00	396.53
LDSAMOD	**1.00**	**90.80**	**1.00**	**40.53**	**1.00**	9.13	**1.00**	73.73	**1.00**	**28.27**	**1.00**	**34.53**	**1.00**	**46.17**	**1.00**	**57.60**	**1.00**	**192.47**

Freeman3, *Jogging-2* and *Woman* sequences, where as B2DPCA fails to estimate the scale and location of the object in most of the sequences except *Doll* and *Jogging-2*. Also, LWIST tracks the object completely in *Fish*, *Jogging-2*, *Singer1* and *Walking2* sequences, whereas WLCS tracks the object in *Faceocc2*, *Singer1* and *Walking2* sequences. Further, LSST fails to estimate the scale and location of the object in most of the sequences except *Singer1* and *Freeman3* sequences. SPT fails to track the object in all the sequences, whereas DLT tracks the object successfully in most of the sequences except *Doll*, *Girl* and *Jogging-2*. Further, PCOM fails to estimate the scale and location of the object in most of the sequences except *Singer1* and *Walking2* sequences.

7 Conclusion

In this paper, a generative tracking algorithm using a structural local DCT sparse appearance model with occlusion detection has been proposed. The energy compaction property of the 2D-DCT has been exploited to reduce the size of the dictionary and the candidate samples, which in turn, lowers the computational cost of l_1-minimization. Also, it has been proposed to reconstruct the holistic image from the overlapped local patches obtained from the patch dictionary and the sparse codes. Further, a robust occlusion map generation has been proposed using the reconstructed image and the tracked candidate. Also, it has been proposed to find the threshold for occlusion detection automatically for each sequence. In addition, the highest confident occlusion-free sample among the cumulated samples has been used to reconstruct the image for the template update. Further, it has been proposed to compute the patch occlusion ratio, and has been used in the confidence score computation by weighting the pooled features. Finally, the tracking result has been obtained by the MAP estimation. Extensive experiments have been conducted on the two popular tracking benchmark datasets, OTB-50 and VOT2016, to analyze the performance of the proposed method. The quantitative and qualitative performance of the proposed method has been compared with that of several recent state-of-the-art algorithms using these benchmark datasets, and it has been shown that the proposed method is competitive for most of the challenging sequences.

Acknowledgment

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, the Regroupement Stratégique en Microsystèmes du Québec (ReSMiQ), and Ministère de l'Éducation, de l'Enseignement Supérieur et de la Recherche (MEESR) du Québec.

The authors would like to thank the authors of [3,17,20,29,36–40,42] who made their codes available for comparison with the proposed method.

References

1. Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 798–805 (2006)
2. Babenko, B., Yang, M.H., Belongie, S.: Visual tracking with online multiple instance learning. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 983–990 (2009)
3. Bao, C., Wu, Y., Ling, H., Ji, H.: Real time robust L1 tracker using accelerated proximal gradient approach. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 1830–1837 (2012)

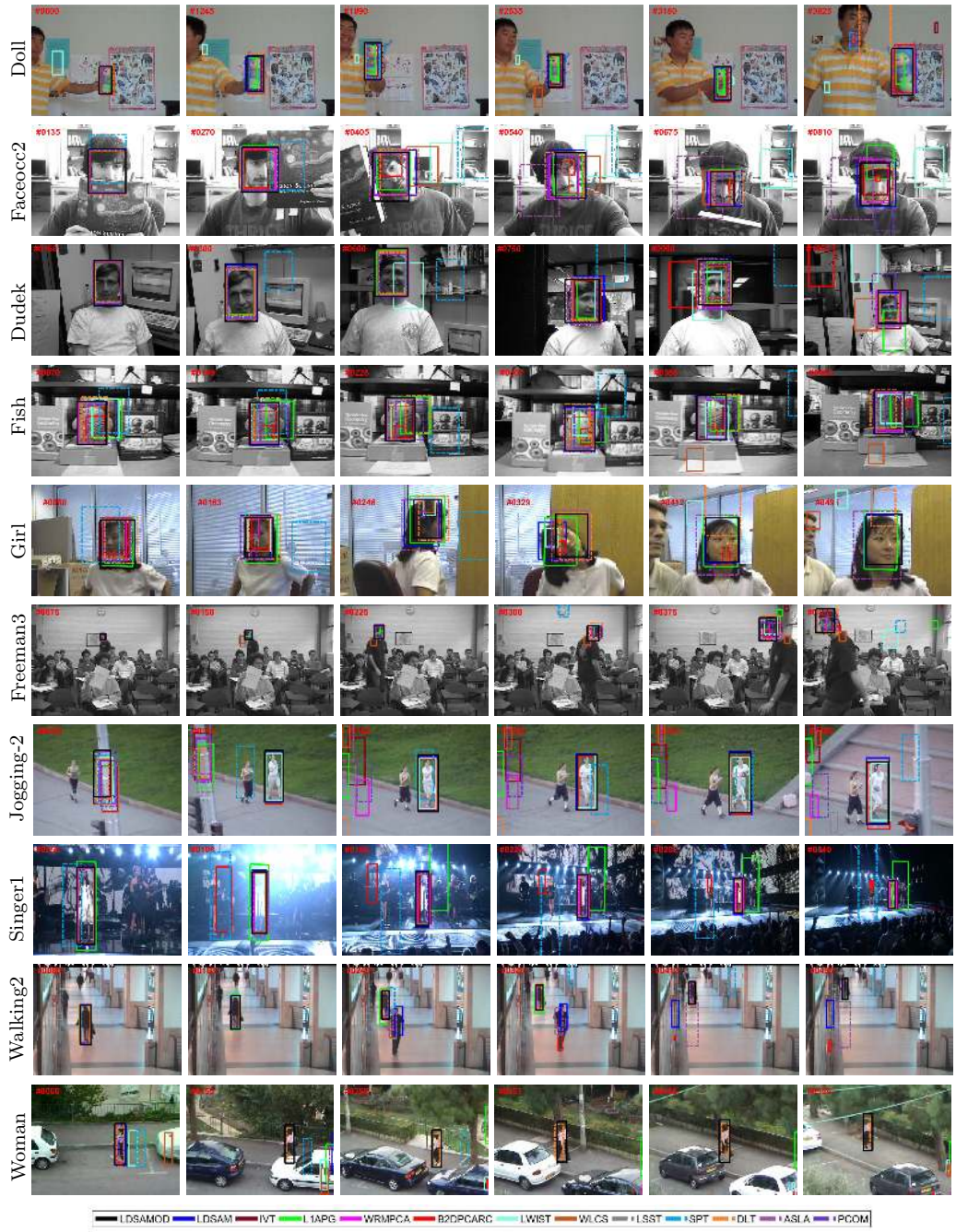


Fig. 7 Examples of tracking results of the compared methods on the ten OTB-50 benchmark sequences.

4. Chen, D., Liu, Q., Sun, M., Yang, J.: Mining appearance models directly from compressed video. IEEE Trans. on Multimedia **10**(2), 268–276 (2008)

5. Chen, H., Zhang, W., Zhao, X., Tan, M.: DCT representations based appearance model for visual tracking. In: Proc. of the IEEE Int. Conf. on Robotics and Biometrics (ROBIO), pp. 1614–1619 (2014)
6. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **25**(5), 564–577 (2003)
7. Dai, P., Luo, Y., Liu, W., Li, C., Xie, Y.: Robust visual tracking via part-based sparsity model. In: Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 1803–1806 (2013)
8. Danelljan, M., Hger, G., Khan, F.S., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: Proc. of the IEEE Int. Conf. on Comput. Vision (ICCV), pp. 4310–4318 (2015)
9. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. on Image processing **15**(12), 3736–3745 (2006)
10. Gao, J., Zhang, T., Yang, X., Xu, C.: Deep relative tracking. IEEE Trans. on Image Processing **26**(4), 1845–1858 (2017)
11. Gao, J., Zhang, T., Yang, X., Xu, C.: P2T: Part-to-target tracking via deep regression learning. IEEE Trans. on Image Processing **27**(6), 3074–3086 (2018)
12. Grabner, H., Leistner, C., Bischof, H.: Semi-supervised on-line boosting for robust tracking. In: Proc. of European Conf. on Comput. Vision (ECCV), pp. 234–247 (2008)
13. Hafed, Z.M., Levine, M.D.: Face recognition using the discrete cosine transform. Int. J. Comput. Vision **43**(3), 167–188 (2001)
14. He, D., Gu, Z., Cercone, N.: Efficient image retrieval in DCT domain by hypothesis testing. In: Proc. of the IEEE Int. Conf. on Image Processing (ICIP), pp. 225–228 (2009)
15. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **37**(3), 583–596 (2015)
16. Isard, M., Blake, A.: Condensation: Conditional density propagation for visual tracking. Int. J. Comput. Vision **29**(1), 5–28 (1998)
17. Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 1822–1829 (2012)
18. Kristan, M., Leonardis, A., Matas, J., Felsberg, M., Pflugfelder, R.: The visual object tracking VOT2016 challenge results. In: Proc. of European Conf. on Comput. Vision (ECCV), pp. 1–45 (2016)
19. Li, H., Li, Y., Porikli, F.: Deeptrack: Learning discriminative feature representations online for robust visual tracking. IEEE Trans. on Image Processing **25**(4), 1834–1848 (2016)
20. Li, X., Dick, A., Shen, C., Hengel, A., Wang, H.: Incremental learning of 3D-DCT compact representations for robust visual tracking. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **35**(4), 863–881 (2013)
21. Li, Y., Ai, H., Yamashita, T., Lao, S., Kawade, M.: Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **30**(10), 1728–1740 (2008)
22. Lin, C., Pun, C.M.: Tracking object using particle filter and DCT features. In: Proc. of Int. Conf. on Advances in Comput. Science and Engineering, pp. 167–169 (2013)
23. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online learning for matrix factorization and sparse coding. J. Mach. Learn. Res. **11**, 19–60 (2010)
24. Mei, X., Ling, H.: Robust visual tracking using L1 minimization. In: Proc. of the IEEE Int. Conf. on Comput. Vision (ICCV), pp. 1436–1443 (2009)
25. Mei, X., Ling, H., Wu, Y., Blasch, E., Bai, L.: Minimum error bounded efficient L1 tracker with occlusion detection. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 1257–1264 (2011)
26. Ou, W., Yuan, D., Liu, Q., Cao, Y.: Object tracking based on online representative sample selection via non-negative least square. Multimedia Tools and Appl. **77**(9), 10,569–10,587 (2018)
27. Pennerbaker, W., Mithchell, J.: JPEG: Still image data compression standard. Springer Science & Business Media (1992)
28. Qu, P.: Visual tracking with fragments-based PCA sparse representation. Int. J. of Signal Processing, Image Processing and Pattern Recogn. **7**(2), 23–34 (2014)
29. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. Int. J. Comput. Vision **77**, 125–141 (2008)
30. Shreyamsha Kumar, B.K., Swamy, M.N.S., Omair Ahmad, M.: Multiresolution DCT decomposition for multifocus image fusion. In: Proc. of the IEEE Canadian Conf. on Electrical and Comput. Engineering (CCECE), pp. 1–4 (2013). DOI 10.1109/CCECE.2013.6567721
31. Shreyamsha Kumar, B.K., Swamy, M.N.S., Omair Ahmad, M.: Structural local DCT sparse appearance model for visual tracking. In: Proc. of the IEEE Int. Symp. on Circuits and Systems (ISCAS), pp. 1194–1197 (2015). DOI 10.1109/ISCAS.2015.7168853
32. Shreyamsha Kumar, B.K., Swamy, M.N.S., Omair Ahmad, M.: Visual tracking via bilateral 2DPCA and robust coding. In: Proc. of the IEEE Canadian Conf. on Electrical and Comput. Engineering (CCECE), pp. 1–4 (2016). DOI 10.1109/CCECE.2016.7726647

33. Shreyamsha Kumar, B.K., Swamy, M.N.S., Omair Ahmad, M.: Weighted residual minimization in PCA subspace for visual tracking. In: Proc. of the IEEE Int. Symp. on Circuits and Systems (ISCAS), pp. 986–989 (2016). DOI 10.1109/ISCAS.2016.7527408

34. Uzair, M., Mahmood, A., Mian, A.S.: Hyperspectral face recognition using 3D-DCT and partial least squares. In: Proc. of British Machine Vision Conference (BMVC), pp. 1–10 (2013)

35. Wang, D., Lu, H.: Object tracking via 2DPCA and l_1-regularization. Signal Processing Letters **19**(11), 711–714 (2012)

36. Wang, D., Lu, H., Bo, C.: Fast and robust object tracking via probability continuous outlier model. IEEE Trans. on Image Processing **24**(12), 5166–5176 (2015)

37. Wang, D., Lu, H., Bo, C.: Visual tracking via weighted local cosine similarity. IEEE Trans. on Cybernetics **45**(9), 1838–1850 (2015)

38. Wang, D., Lu, H., Xiao, Z., Yang, M.H.: Inverse sparse tracker with a locally weighted distance metric. IEEE Trans. on Image Processing **24**(9), 2646–2657 (2015)

39. Wang, D., Lu, H., Yang, M.H.: Online object tracking with sparse prototypes. IEEE Trans. on Image Processing **22**(1), 314–325 (2013)

40. Wang, D., Lu, H., Yang, M.H.: Robust visual tracking via least soft-threshold squares. IEEE Trans. on Circuits and Systems for Video Technology **26**(9), 1709–1721 (2016)

41. Wang, F., Zhang, J., Guo, Q., Liu, P., Tu, D.: Robust visual tracking via discriminative structural sparse feature. In: Proc. of the Chinese Conf. on Image and Graphics Technologies, pp. 438–446 (2015)

42. Wang, N., Yeung, D.Y.: Learning a deep compact image representation for visual tracking. In: Proc. of Advances in Neural Information Processing Systems (NIPS), pp. 809–817 (2013)

43. Wright, J., Yang, A., Ganesh, A., Sastry, S., Ma, Y.: Robust face recognition via sparse representation. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **31**(2), 210–227 (2009)

44. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 2411–2418 (2013)

45. Yang, H., Shao, L., Zheng, F., Wang, L., Song, Z.: Recent advances and trends in visual tracking: A review. Neurocomputing **74**(18), 3823–3831 (2011)

46. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 1794–1801 (2009)

47. You, X., Li, X., He, Z., Zhang, X.: A robust local sparse tracker with global consistency constraint. Signal Processing **111**, 308–318 (2015)

48. Zhang, H., Tao, F., Yang, G.: Robust visual tracking based on structured sparse representation model. Multimedia Tools and Appl. **74**(3), 1021–1043 (2015)

49. Zhang, T., Bibi, A., Ghanem, B.: In defense of sparse tracking: Circulant sparse tracker. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 3880–3888 (2016)

50. Zhang, T., Ghanem, B., Liu, S., Ahuja, N.: Robust visual tracking via multi-task sparse learning. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 2042–2049 (2012)

51. Zhang, T., Liu, S., Xu, C., Liu, B., Yang, M.H.: Correlation particle filter for visual tracking. IEEE Trans. on Image Processing **27**(6), 2676–2687 (2018)

52. Zhang, T., Xu, C., Yang, M.H.: Multi-task correlation particle filter for robust object tracking. In: Proc. of the IEEE Conf. on Comput. Vision and Pattern Recogn. (CVPR), pp. 4819–4827 (2017)

53. Zhang, T., Xu, C., Yang, M.H.: Learning multi-task correlation particle filters for visual tracking. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) pp. 1–14 (2018)

54. Zhong, Y., Zhang, H., Jain, A.K.: Automatic caption localization in compressed video. IEEE Trans. on Pattern Anal. and Mach. Intell. (PAMI) **22**(4), 385–392 (2000)

55. Zhuang, B., Wang, L., Lu, H.: Visual tracking via shallow and deep collaborative model. Neurocomputing **218**, 61–71 (2016)

B. K. Shreyamsha Kumar received B.E. degree in electronics and communication engineering from Bangalore University, India, in 2000, and the M.Tech degree in Industrial Electronics from National Institute of Technology Karnataka, Surathkal, India, in 2004. He is currently pursuing Ph.D. degree in electrical and computer engineering with Concordia University, Montreal, QC, Canada. He has been a Research Associate with the Signal Processing Group, Concordia University, since Sept 2012. Before joining Concordia, he was with Central Research Laboratory (A Corporate Research Facility of Bharat Electronics, India) as a Member (Research Staff) from Oct 2004 to Aug 2012. He is a recipient of R&D Excellence Award conferred by Bharat Electronics. His research interests include visual tracking, computer vision, image fusion, image denoising, image encryption and document image processing. He has served as a reviewer for several peer-reviewed journals and major conferences.

M.N.S. Swamy received the B.Sc. (Hons.) degree in mathematics from Mysore University, India, in 1954, the Diploma degree in electrical communication engineering from the Indian Institute of Science, Bangalore, in 1957 and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Saskatchewan, Saskatoon, Canada, in 1960 and 1963, respectively. He was conferred in 2009 the title of Honorary Professor at National Chiao Tong University in Taiwan. He is presently a Research Professor and the Director of the Center for Signal Processing and Communications in the Department of Electrical and Computer Engineering at Concordia University, Montreal, QC, Canada, where he served as the Chair of the Department of Electrical Engineering from 1970 to 1977, and Dean of Engineering and Computer Science from 1977 to 1993. During that time, he developed the Faculty into a research-oriented one from what was primarily an undergraduate Faculty. Since July 2001, he holds the Concordia Chair (Tier I) in Signal Processing. He has also taught in the Electrical Engineering Department of the Technical University of Nova Scotia, Halifax, and the University of Calgary, Calgary, as well as in the Department of Mathematics at the University of Saskatchewan. He has published extensively in the areas of number theory, circuits, systems and signal processing, and holds five patents. He is the coauthor of nine books and three book chapters. He was a founding member of Micronet from its inception in 1990 as a Canadian Network of Centers of Excellence until its expiration in 2004, and also its coordinator for Concordia University. Dr. Swamy is a Fellow of the Institute of Electrical and Electronics Engineers, Fellow of the Institute of Electrical Engineers (United Kingdom), the Engineering Institute of Canada, the Institution of Engineers (India), and the Institution of Electronic and Telecommunication Engineers (India). He was inducted in 2009 to the Provosts Circle of Distinction for career achievements. He has served the IEEE in various capacities such as the President-Elect in 2003, President in 2004, Past-President in 2005, Vice President (Publications) during 20012002, Vice-President in 1976, Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I from June 1999 to December 2001, Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS during June 1985 to May 1987, Program Chair for the 1973 IEEE CAS Symposium, General Chair for the 1984 IEEE CAS Symposium, Vice-Chair for the 1999 IEEE Circuits and Systems (CAS) Symposium, and a member of the Board of Governors of the CAS Society. He is the recipient of many IEEE-CAS Society awards, including the Education Award in 2000, Golden Jubilee Medal in 2000, and the 1986 Guillemin-Cauer Best Paper Award. He is the Editor-in-Chief of the journal Circuits, Systems and Signal Processing (CSSP) since 1999. Recently CSSP has instituted a best paper award in his name.

M. Omair Ahmad received the B.Eng. degree from Sir George Williams University, Montreal, QC, Canada, and the Ph.D. degree from Concordia University, Montreal, QC, Canada, both in electrical engineering. From 1978 to 1979, he was a Faculty Member with the New York University College, Buffalo, NY, USA. In September 1979, he joined the Faculty of Concordia University as an Assistant Professor of computer science. He joined the Department of Electrical and Computer Engineering, Concordia University, where he was the Chair with the department from June 2002 to May 2005 and is currently a Professor. He holds the Concordia University Research Chair (Tier I) in Multimedia Signal Processing. He has published extensively in the area of signal processing and holds four patents. His current research interests include the areas of multidimensional filter design, speech, image and video processing, non-linear signal processing, communication DSP, artificial neural networks, and VLSI circuits for signal processing. He was a Founding Researcher at Micronet from its inception in 1990 as a Canadian Network of Centers of Excellence until its expiration in 2004. Previously, he was an Examiner of the order of Engineers of Quebec. Dr. Ahmad was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART I: FUNDAMENTAL THEORY AND APPLICATIONS from June 1999 to December 2001. He was the Local Arrangements Chairman of the 1984 IEEE International Symposium on Circuits and Systems. In 1988, he was a member of the Admission and Advancement Committee of the IEEE. He has served as the Program Co-Chair for the 1995 IEEE International Conference on Neural Networks and Signal Processing, the 2003 IEEE International Conference on Neural Networks and Signal Processing, and the 2004 IEEE International Midwest Symposium on Circuits and Systems. He was a General Co-Chair for the 2008 IEEE International Conference on Neural Networks and Signal Processing. He is the Chair of the Montreal Chapter IEEE Circuits and Systems Society. He is a recipient of numerous honors and awards, including the Wighton Fellowship from the Sandford Fleming Foundation, an induction to Provosts Circle of Distinction for Career Achievements, and the Award of Excellence in Doctoral Supervision from the Faculty of Engineering and Computer Science of Concordia University. Dr. Ahmad is a Fellow of the Institute of Electrical and Electronics Engineers.