Low-rank Tensor Tracking

Sajid Javed, Jorge Dias, and Naoufel Werghi
Khalifa University of Science and Technology
Abu Dhabi, United Arab Emirates
sajid.javed,jorge.dias,naoufel.werghi@ku.ac.ae

Abstract

Visual object tracking is an important step for many computer vision applications. Visual tracking becomes more challenging when the target object observes severe occlusion, lighting variations, background clutter, and deformation difficulties to name a few. In the literature, low-rank matrix decomposition methods have shown to be a potential solution for visual tracking in many complex scenarios. These methods first arrange the particles of the target object in a 2-D data matrix and then perform convex optimization to solve the low-rank objective function. However, these methods show performance degradation in the presence of the aforementioned challenges. Because these methods do not consider the intrinsic structure of the target particles, therefore, the object loses its spatial appearance or consistency. To address these challenges, we propose a new low-rank tensor decomposition model for robust object tracking. Our proposed low-rank tensor tracker considers the multi-dimensional data of target particles. We employ the recently proposed tensor-tensor product-based singular value decomposition and a new tensor nuclear norm to promote the intrinsic structure correlation among the target particles. Experimental evaluations on 20 challenging tracking sequences demonstrate the excellent performance of the proposed tracker as compared with state-of-the-art trackers.

1. Introduction

The visual object tracking is an important task for many computer vision applications comprising video surveillance [2], human activity recognition [49], robotics [44], and augmented reality [22]. The main goal is to estimate the position of the target object between consecutive video frames given its bounding box position in the first frame. Visual tracking becomes more challenging when a target object undergoes varying lighting conditions, background clutters, object deformation, in plane rotation, out of plane rotation, scale variation, occlusion, fast motion, and motion blur [52]. In the literature, many trackers have been proposed to handle object tracking problem [7, 28, 33, 43, 46, 52, 53, 59]. A number of potential datasets and survey studies have also been contributed for object tracking [10, 26, 27, 30, 32, 46, 51–53]. However, visual tracking is still a challenging problem because of the aforementioned tracking difficulties [52].

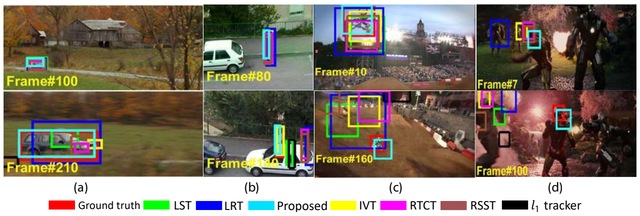

Figure 1. Comparisons of the proposed LLT tracker with different state-of-the-art subspace representation-based trackers including LST [21], LRT [58], IVT[43], l_1 tracker [40], RTCT [55], RSST [60], and MTT [57] on (a) `CarScale`, (b) `Woman`, (c) `MotorRolling`, and (d) `Ironman` sequences taken from OTB50 dataset [51]. These sequences present severe tracking difficulties of scale variations in (a) and (c), occlusion in (b), fast motion in (c), motion blur and illumination variation in (d). The proposed LLT tracker consistently performs well against these challenges.

Among the most popular tracking methods such as those presented in [32, 52], subspace learning trackers have enjoyed a great success in the past. David *et al.* proposed subspace learning tracker in which a low-dimensional subspace corresponding to target object is estimated using principal component analysis [43]. The tracker exhibited excellent performance however, due to the noise and gross corruption in the target object, the tracker caused performance degradation in the presence of occlusion since, the target appearance was not fully addressed (Figs. 1 (a)-(d)). To handle this challenge, many sparse representation-based trackers are proposed [1, 15, 21, 33, 34, 40, 41, 55, 57, 60]. These trackers learn the target appearance based on the linear combination of the dictionary atoms. For instance, Jia *et al.* proposed a local sparse representation tracker whereby each patch inside the target region is represented by a sparse linear combination of the dictionary elements [21]. Encouraging results were presented in sparse representation-based trackers [1, 15, 21, 34, 41] for many complex scenarios however some of these trackers showed performance degradation in the presence of scale variations and fast motion (Figs. 1 (a)-(d)). To handle these issues, global and joint sparse representation trackers have also been proposed which encode the spatial intrinsic structure of the target object [33, 40, 55, 57, 60].

Many trackers have also considered the correlations among the target particles [20, 54, 56, 58, 62]. For example, Zhang *et al.* proposed low-rank sparse tracker in which the global low-rank component is estimated using nuclear norm convex regularization among the target particles [58]. The temporal consistency and sparse representation assumption are also incorporated for consistent object tracking. Although good results are reported over many complex tracking sequences the tracker loses target in the presence of fast motion and motion blur sequences (Figs. 1 (a)-(d)). Moreover, time complexity also remained the major burden for these trackers because of the batch optimization. To address this lack, Zhang *et al.* proposed an online formulation which processes each particle sample per time instance [54]. Javed *et al.* recently proposed structural low-rank tracker where the Laplacian matrices were enforced into the low-rank component to promote the spatial and temporal structure of the target [20]. All these subspace-based trackers reported good results. However, the vectorization of the target particles in these methods disregards the intrinsic structural information leading to an implicit loss of the spatial coherence among the target objects. We hypothesize that vectorization scheme is adverse to robust tracking in the presence of the tracking challenges mentioned earlier, and therefore accountable for the performance degradation.

To address these challenges, we propose a new tracker, dubbed as Low-rank Tensor Tracking (LTT). The proposed tracker processes multidimensional input data comprising target particles for visual tracking. We first create a 3-D input data matrix of n target particles in the t-th frame using particle sampling technique [47]. We observe that there exists a correlation among all the tensor modes of the target particles. This multi-dimensional correlation reveals the intrinsic low-rank structure which is useful for modeling the target object appearance consistency. We employ recently proposed tensor nuclear norm based on tensor-tensor product and tensor singular value decomposition in the fast Fourier transform domain [16, 35, 36]. This tensor nuclear norm regularizes all tensor modes and hence estimates the low-rank tensor in a mixture of multiple subspaces effectively. To capture the final tensors, the proposed tracker uses an average pooling operation of folded tensors in all modes. Our proposed LTT tracker incorporating multi-dimensional spatial constraints into the objective function allows to track objects in many complicated tracking scenarios, even when the appearance of the target object varies because of occlusion, fast motion, and motion blur which is quite difficult for matrix-based subspace trackers (Figs. 1 (a)-(d)). We solve our proposed objective function using Alternating Direction Method of Multiplier (ADMM) optimization scheme with convergence guarantee [37]. Experiments on 20 complex tracking sequences demonstrate that the proposed tracker achieves the excellent performance as compared to state-of-the-art trackers including matrix-based low-rank trackers. To the best of our knowledge, this is the first work which explores the low-rank tensor for visual object tracking.

The remainder of this paper is organized as follows. In section II, we review related work. In section III, we describe the proposed LLT algorithm in detail. The experimental results are presented in Section IV. Finally, the conclusions are drawn in Section V.

2. Related Work

In the past decade, a plethora of methods have been proposed for visual object tracking [3, 6, 9, 13, 20, 38, 41, 60]. Interested readers are referred to survey studies for more details in [10, 30, 46, 51–53]. However, there is no unique tracker which is able to handle all aforementioned tracking challenges. In this work, we classify state-of-the-art trackers in to deep learning trackers [9, 38, 48], sparse representation-based trackers [21, 33, 34, 41, 58, 60], correlation filter-based trackers [3, 6, 13], and low-rank trackers [20, 58, 62].

Sparse representation-based methods have achieved a great success for visual object tracking in the past few years [21, 29, 33, 34, 41, 58, 60]. These methods mainly represent the target appearance using a sparse linear combination of dictionary atoms. The dictionary is updated to cope with target appearance variations. In global methods, the holistic appearance of the target is modeled using sparse representation [29, 34, 41, 55, 58]. The global methods perform well in many tracking scenarios however; they cause performance degradation in the presence of occlusion. In local methods, local patches of the target particle are represented using sparse combination of dictionary atoms [21, 33]. However, local and global methods ignore the structural intrinsic relationship among the particle and their patches. In [60], a joint sparse representation tracker is proposed which jointly learns the local and global intrinsic relationships among the target particles and their local patches. Encouraging results are reported for many complex tracking scenarios as the deep features are also incorporated in this work.

Many deep learning methods have also been proposed to handle the visual tracking problem such as those reported in [9, 38, 48]. These methods use already pre-trained model for feature extraction and then employ either the classifier or learn the correlation filters for tracking. For instance,

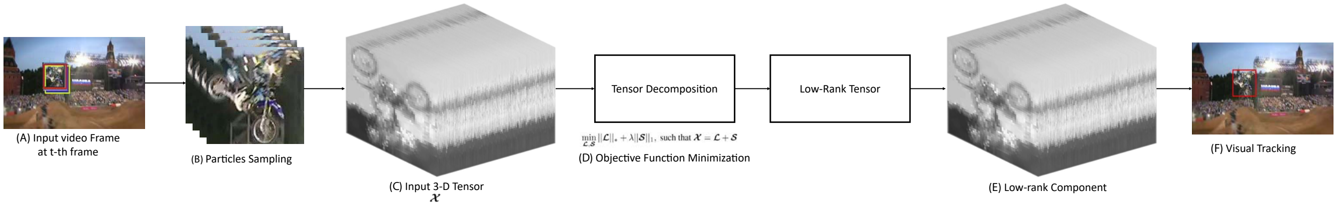
(A) Input video Frame at t-th frame
(B) Particles Sampling
(C) Input 3-D Tensor $\mathcal{X}$
Tensor Decomposition
$\min_{\mathcal{L},\mathcal{S}} ||\mathcal{L}||_* + \lambda||\mathcal{S}||_1$, such that $\mathcal{X} = \mathcal{L} + \mathcal{S}$
(D) Objective Function Minimization
Low-Rank Tensor
(E) Low-rank Component
(F) Visual Tracking

Figure 2. Schematic illustration of our proposed Low-Rank Tesnsor Tracking (LTT) algorithm. (A) shows an input image at time *t* with target particles bounding boxes. (B) particle sampling strategy and target particles are cropped to create input tensor $\boldsymbol{\mathcal{X}}$. (C) input tensor $\boldsymbol{\mathcal{X}}$. (D) shows the objective function optimization. (E) shows Low-rank tensor. (F) shows the result of visual object tracking using our proposed LTT tracker.

Wang and Yeung proposed a deep neural network model pre-trained on natural tiny images for visual tracking [48]. Ma *et al.* exploited deep features hierarchy using VGG-19 model [38]. The correlation filter response is computed for each convolutional features map and averaged to get the score for a target object. Danelljan *et al.* also exploited multiple convolutional features for visual tracking [9]. All these deep trackers show improved performance but they rely mainly on already pre-trained model since a large amount of annotated data is not available for training.

Correlation filter-based trackers have also shown to be a potential methods for tracking [3, 6, 13]. These methods estimate the fast Fourier transform in the target window. For instance, Bolme *et al.* used a luminance channel and learnt a sum of squared error filter for tracking [3]. Several other improvements are also proposed for tracking in [6,13]. These methods are quite simple and fast as compared to other trackers due to the computation of fast Fourier transform. However, these trackers drift due to the noise in the target appearance variations therefore these trackers mainly rely on the re-detection module [38]. Many low-rank trackers have also been proposed in the literature [20, 58, 62]. For instance, Zhang *et al.* were the first author to model the correlations among the target particles using low-rank computations [58]. Several other low-rank trackers are also proposed to improve the tracking performance [20, 62]. These trackers show good performance however; they ignore the intrinsic relationship among the target particles as each target particle is represented by a column vector which leads to the information loss and would cause performance degradation.

To address this lack, we propose low-rank tensor tracker which manipulates the tensor data by taking the advantage of its multi-dimensional structure and improves the performance in many tracking scenarios.

3. Proposed Methodology

The major steps of the proposed algorithm are shown in Fig. 2. Our proposed algorithm is dubbed as Low-rank Tensor Tracking (LTT). The LTT is based on the particle filter-based tracking approach in which we draw an n number of target particles from a given target object at t-th frame using particle sampling technique [47]. These n target particles are used to construct 3-way input tensor, $\boldsymbol{\mathcal{X}} = [\mathbf{X}_1, \mathbf{X}_2, \cdot\cdot\cdot, \mathbf{X}_n] \in \mathbb{R}^{n_1 \times n_2 \times n}$, where each 2-D matrix, $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$, denotes target particle with width n_1 and height n_2. These target particles in tensor $\boldsymbol{\mathcal{X}}$ are highly correlated with each other and therefore, they span a low-dimensional common subspace in a mixture of multiple low-rank subspaces. We aim to recover such a subspace representing target appearance by enforcing nuclear norm regularization in all modes of a tensor. In the later sections, we describe each step of the proposed LTT in more detail.

3.1. LTT Model

We pose the problem of visual object tracking as the problem of estimating low-rank component from target particles. Wright *et al.* has formulated the following convex optimization problem for low-rank matrix decomposition as [50]:

$$\min_{\mathbf{L},\mathbf{E}} ||\mathbf{L}||_* + \lambda_1||\mathbf{E}||_1, \text{ such that } \mathbf{X} = \mathbf{L} + \mathbf{E}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{d \times n}$ denotes input 2-D matrix comprising n target particles and $d = n_1 \times n_2$. $\mathbf{L} \in \mathbb{R}^{d \times n}$ is the low-rank representation which models the correlation among the target particles and $\mathbf{E} \in \mathbb{R}^{d \times n}$ is the sparse outlier comprising occlusions and noises. In (1), $||\cdot||_*$ denotes the nuclear norm (sum of the singular values), $||\cdot||_1$ is the l_1-norm (sum of the absolute values of all the entries), and $\lambda_1 = 1/\sqrt{\max(d, n)}$. The low-rank representation model defined by (1) has enjoyed a great success in the past years in a number of computer vision applications including background-foreground separation [17–19], salient object detection [42], image rain streaks removal [31], and medical imaging [4] to name a few.

In case of visual tracking, the low-rank representation $\mathbf{L}$ is then used to estimate the observation likelihood model $p(\mathbf{x}_t^i|\mathbf{z}_t^i)$ for sample state $\mathbf{z}$ at a time instance t for i-th particle in the particle filter framework [47]. The reconstruction error is computed between the target particles and the low-rank representation as:

$$p(\mathbf{x}_t^i|\mathbf{z}_t^i) = \exp(-||\mathbf{x}_t^i - \mathbf{l}_t^i||_2^2). \tag{2}$$

One major shortcoming of model (1) is that it can only processes 2-D data matrix which causes performance degradation in the case of object tracking since it ignores the difference of the intrinsic structure correlation among the target particles.

In this work, we address this problem by estimating the correlation between all modes of a 3-D tensor $\boldsymbol{\mathcal{X}}$. Liu *et al.* has proposed the following low-rank tensor objective function as [35, 36]:

$$\min_{\boldsymbol{\mathcal{L}},\boldsymbol{\mathcal{E}}} ||\boldsymbol{\mathcal{L}}||_* + \lambda||\boldsymbol{\mathcal{E}}||_1, \text{ such that } \boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{E}}, \tag{3}$$

where $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ is the low-rank tensor representation of target particles, $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ is the sparse tensor which models occlusions, and $\lambda_1 = 1/\sqrt{\max(n_1, n_2)n}$. $||\boldsymbol{\mathcal{L}}||_*$ is the tensor nuclear norm which is computed in the fast Fourier transform (see the definition in Sec. 3.2) and $||\boldsymbol{\mathcal{E}}||_1$ is the l_1-norm of the tensor. We observed that the extension of low-rank matrix model (1) into low-rank tensor model (3) significantly improves the performance of the visual object tracking. In order to optimize the proposed LLT objective function (3), we first present notations and tensor algebra and then we explain the optimization of Eq. (3) using ADMM method [37].

3.2. Notations and Preliminaries

In this section, we define basic notations of tensor and several algebric structures of 3-way tensors. We use the same notations as defined in [25].

We denote tensors by bold face calligraphic letters, e.g., $\boldsymbol{\mathcal{X}}$, matrices by bold face capital letters, e.g., $\mathbf{X}$, vectors by bold face lower case letters $\mathbf{x}$, and scalars by lower case letters, e.g., x. An n-way tensor is a multi-dimensional array and it is represented by $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ and its (i, j, k)-th entry is denoted by $\boldsymbol{\mathcal{X}}_{ijk}$ or x_{ijk}. The slice of a tensor is a 2-D section defined by fixing all but two indices. The i-th horizontal, lateral, and frontal slice of a tensor $\boldsymbol{\mathcal{X}}$ is denoted by $\boldsymbol{\mathcal{X}}(i, :, :)$, $\boldsymbol{\mathcal{X}}(:, i, :)$, and $\boldsymbol{\mathcal{X}}(:, :, i)$, respectively. The i-th frontal slice $\boldsymbol{\mathcal{X}}(:, :, i)$ of a tensor $\boldsymbol{\mathcal{X}}$ is also denoted by $\mathbf{X}^{(i)}$. Similarly, the fiber is a 1-D section defined by fixing all indices but one also known as tube of $\boldsymbol{\mathcal{X}}$. The $\boldsymbol{\mathcal{X}}(:, i, j)$, $\boldsymbol{\mathcal{X}}(i, j, :)$, and $\boldsymbol{\mathcal{X}}(i, :, j)$, respectively, denote the mode-1, mode-2, and mode-3 fibers of $\boldsymbol{\mathcal{X}}$ [24, 35, 36]. The inner product between $\mathbf{X}$ and $\mathbf{Y}$ is defined by $< \mathbf{X}, \mathbf{Y} >=$ $\text{Tr}(\mathbf{X}^* \, \mathbf{Y})$, where $\mathbf{X}^*$ denotes the conjugate transpose of matrix $\mathbf{X}$ and $\text{Tr}(\cdot)$ denotes the trace of a matrix. The Frobenius norm of a tensor $\boldsymbol{\mathcal{X}}$ is $||\boldsymbol{\mathcal{X}}||_F = \sqrt{(\sum_{i,j,k}|x_{ijk}|^2)}$, and l_1-norm of $\boldsymbol{\mathcal{X}}$ is $||\boldsymbol{\mathcal{X}}||_1 = \sum_{i,j,k}|x_{ijk}|$.

We denote $\bar{\boldsymbol{\mathcal{X}}}$ as the result of discrete fast Fourier transform along the 3-rd dimension, e.g., $\bar{\boldsymbol{\mathcal{X}}} = \texttt{fft}(\boldsymbol{\mathcal{X}}, [\,], 3)$. In the same manner, we can also compute $\boldsymbol{\mathcal{X}}$ as the result of inverse fast Fourier transform as $\boldsymbol{\mathcal{X}} = \texttt{ifft}(\bar{\boldsymbol{\mathcal{X}}}, [\,], 3)$. The block diagonal matrix $\bar{\mathbf{X}}$ of tensor $\bar{\boldsymbol{\mathcal{X}}}$ can be written as $\bar{\mathbf{X}} = \texttt{bdiag}(\bar{\boldsymbol{\mathcal{X}}})$ with its each block in its diagonal as the frontal slice $\bar{\mathbf{X}}^{(i)}$ of $\bar{\boldsymbol{\mathcal{X}}}$ as [24]:

$$\bar{\mathbf{X}} = \texttt{bdiag}(\bar{\boldsymbol{\mathcal{X}}}) = \begin{bmatrix} \bar{\mathbf{X}}^{(1)} & & & \\ & \bar{\mathbf{X}}^{(2)} & & \\ & & \ddots & \\ & & & \bar{\mathbf{X}}^{(n)} \end{bmatrix} \tag{4}$$

Tensor-Tensor Product [24]: To define a new tensor-tensor product, we first define three main block-based operators. For tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$, its block circulant matrix $\texttt{bcirc}(\boldsymbol{\mathcal{X}}) \in \mathbb{R}^{n_1 n \times n_2 n}$ can be defined as:

$$\texttt{bcirc}(\boldsymbol{\mathcal{X}}) = \begin{bmatrix} \mathbf{X}^{(1)} & \mathbf{X}^{(n)} & \dots & \mathbf{X}^{(2)} \\ \mathbf{X}^{(2)} & \mathbf{X}^{(1)} & \dots & \mathbf{X}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{X}^{(n)} & \mathbf{X}^{(n-1)} & \dots & \mathbf{X}^{(1)} \end{bmatrix} \tag{5}$$

The $\texttt{unfold}$ operation also known as block-vectorization, and its opposite operator $\texttt{fold}$ can be defined as:

$$\texttt{unfold}(\boldsymbol{\mathcal{X}}) = \begin{bmatrix} \mathbf{X}^{(1)} \\ \mathbf{X}^{(2)} \\ \vdots \\ \mathbf{X}^{(n)} \end{bmatrix}, \texttt{fold}(\texttt{unfold}(\boldsymbol{\mathcal{X}})) = \boldsymbol{\mathcal{X}}. \tag{6}$$

Finally, the tensor-tensor product (t-product) $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{n_1 \times n_4 \times n}$, between any two 3-way tensors $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ and $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{n_2 \times n_4 \times n}$ can be defined as:

$$\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} * \boldsymbol{\mathcal{Y}} = \texttt{fold}\{\texttt{bcirc}(\boldsymbol{\mathcal{X}})\texttt{unfold}(\boldsymbol{\mathcal{Y}})\}. \tag{7}$$

It should be noted that the tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$, can be treated as $n_1 \times n_2$ matrix with each entry as a tube in the third dimension. This t-product is analogous to the matrix-matrix product except that the circular convolution replaces the product operation between the elements. The t-product in the original domain corresponds to the matrix multiplication of the frontal slices in the fourier domain [24, 35, 36].

Tensor Transpose [24]: The transpose of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$, is an $n_2 \times n_1 \times n$ tensor which is obtained by transposing each frontal slice of $\boldsymbol{\mathcal{X}}$ and then reversing the order of the transposed frontal slices 2 through n.

Identity Tensor [24]: The identity tensor can be denoted

Algorithm 1: Tensor SVD Algorithm.

```
Input: Input tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$.
Output: $\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}$.
1. Compute $\bar{\boldsymbol{\mathcal{X}}} = \texttt{fft}(\boldsymbol{\mathcal{X}}, [], 3)$
2. for $k = 1 : n$ do
    3. Compute $[\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}] = \texttt{SVD}(\bar{\boldsymbol{\mathcal{X}}}^{(k)})$.
    4. $\bar{\boldsymbol{\mathcal{U}}}^{(k)} = \boldsymbol{\mathcal{U}}$, $\bar{\boldsymbol{\mathcal{S}}}^{(k)} = \boldsymbol{\mathcal{S}}$, $\bar{\boldsymbol{\mathcal{V}}}^{(k)} = \boldsymbol{\mathcal{V}}$.
end
5. $\boldsymbol{\mathcal{U}} = \texttt{ifft}(\bar{\boldsymbol{\mathcal{U}}}, [\,], 3)$, $\boldsymbol{\mathcal{S}} = \texttt{ifft}(\bar{\boldsymbol{\mathcal{S}}}, [\,], 3)$,
$\boldsymbol{\mathcal{V}} = \texttt{ifft}(\bar{\boldsymbol{\mathcal{V}}}, [\,], 3)$.
6.Return $\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}$.
```

as $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{n_1 \times n_1 \times n}$ and it is defined as a tensor whose first frontal slice is the $n_1 \times n_1$ identity matrix and all other frontal slices are zero.

Orthogonal Tensor [24]: The tensor $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{n_1 \times n_1 \times n}$, is said to be orthogonal, if and only if $\boldsymbol{\mathcal{Q}}^\top * \boldsymbol{\mathcal{Q}} = \boldsymbol{\mathcal{Q}} * \boldsymbol{\mathcal{Q}}^\top = \boldsymbol{\mathcal{I}}$, where $*$ is the t-product.

f-Diagonal Tensor [24]: A tensor is said to be f-diagonal if each of its frontal slices is equal to a diagonal matrix.

Tensor Singular Value Decomposition (t-SVD) [24]: The t-SVD of tensor $\boldsymbol{\mathcal{X}}$ can be factorized as follows:

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^\top, \tag{8}$$

where $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{n_1 \times n_1 \times n}$ and $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{n_2 \times n_2 \times n}$ are orthogonal tensors, respectively. $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ is a f-diagonal tensor and * denotes the t-product. This t-product can be computed more effieciently in the fast Fourier transform domain as presented in Alg. 1 of t-SVD.

Tensor Nuclear Norm (TNN) [16,35,36,45,61]: The TNN of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ is denoted by $||\boldsymbol{\mathcal{X}}||_*$ and it can be defined as the average of the nuclear norms of all the frontal slices of $\bar{\boldsymbol{\mathcal{X}}}$ as:

$$||\boldsymbol{\mathcal{X}}||_* = \frac{1}{n}\sum_{j=1}^{n}||\bar{\mathbf{X}}^{(j)}||_* = \frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{\min(n_1,n_2)}|\boldsymbol{\mathcal{S}}(i,i,j)|, \tag{9}$$

where $\boldsymbol{\mathcal{S}}$ is the singular values computed by Alg. 1. It should be noted that the TNN in (9) is defined in the fast Fourier domain. It is also closely related to the nuclear norm of the block circulant matrix in the original domain as:

$$\begin{aligned} ||\boldsymbol{\mathcal{X}}||_* &= \frac{1}{n}\sum_{j=1}^{n}||\bar{\mathbf{X}}^{(j)}||_* = \frac{1}{n}||\bar{\mathbf{X}}||_* \\ &= \frac{1}{n}||(\mathbf{F}_n \otimes \mathbf{I}_{n_1}).\texttt{bcirc}(\boldsymbol{\mathcal{X}}).(\mathbf{F}_n^{-1} \otimes \mathbf{I}_{n_2})||_* \\ &= \frac{1}{n}||\texttt{bcirc}(\boldsymbol{\mathcal{X}})||_*. \end{aligned} \tag{10}$$

where $\mathbf{F}_n$ denotes the $n \times n$ discrete Fourier transform matrix and $\otimes$ denotes the Kronecker product. The above Eq. (10) preserves more spatial intrinsic relationship among the tensor modes and therefore improves the tracking performance.

3.3. LTT Optimization

LTT model (3) is the convex optimization problem and therefore, we can solve it using the standard ADMM method [5, 37]. The proof of the global convergence can be found in [37,61]. To solve it, we first formulate its augmented Lagrangian function as:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{E}}, \mu) = ||\boldsymbol{\mathcal{L}}||_* + \lambda||\boldsymbol{\mathcal{E}}||_1 + \\ < \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{E}} > + \frac{\mu}{2}||\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{E}}||_F^2, \end{aligned} \tag{11}$$

where $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{n_1 \times n_2 \times n}$ is the Lagrangian multiplier and $\mu > 0$ is the penalty scalar for violating the linear constraints. Problem (11) can now be solved using divide and conquer strategy by solving one variable and fixing others. The optimization process will converge to the optimal solution when μ^k increases and bounds in the k-th iteration [36,37,61].

Updating $\boldsymbol{\mathcal{L}}$: Consider the following sub-problem from (11) by fixing $\boldsymbol{\mathcal{S}}$ as:

$$\begin{aligned} \boldsymbol{\mathcal{L}}^{k+1} &= \arg\min_{\boldsymbol{\mathcal{L}}} \mathcal{L}(\boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{E}}^k, \boldsymbol{\mathcal{Y}}^k, \mu^k) \\ &= \arg\min_{\boldsymbol{\mathcal{L}}} ||\boldsymbol{\mathcal{L}}||_* + \frac{\mu^k}{2}||\boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{Z}}_1||_F^2 \\ &= \min_{\boldsymbol{\mathcal{L}}} \tau||\boldsymbol{\mathcal{L}}||_* + \frac{1}{2}||\boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{Z}}_1||_F^2, \end{aligned} \tag{12}$$

where $\boldsymbol{\mathcal{Z}}_1 = \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{E}}^k + \frac{1}{\mu^k}\boldsymbol{\mathcal{Y}}^k$ and $\tau > 0$. The $\boldsymbol{\mathcal{L}}^{k+1}$ has a following closed-form solution by the tensor singular value convoluting as [16, 36, 37, 61]:

$$\boldsymbol{\mathcal{L}}^{k+1} = \boldsymbol{\mathcal{C}}_\tau(\boldsymbol{\mathcal{Z}}_1) = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{C}}_\tau(\boldsymbol{\mathcal{S}}) * \boldsymbol{\mathcal{V}}^\top \tag{13}$$

where $\boldsymbol{\mathcal{U}}$, $\boldsymbol{\mathcal{S}}$, and $\boldsymbol{\mathcal{V}}$ is the tensor singular value decomposition of $\boldsymbol{\mathcal{Z}}_1$ and $\boldsymbol{\mathcal{C}}_\tau(\boldsymbol{\mathcal{S}}) = \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{J}}$, where $\boldsymbol{\mathcal{J}}$ is an $n_1 \times n_2 \times n$ f-diagonal tensor whose diagonal element in the Fourier domain is $\bar{\boldsymbol{\mathcal{J}}}(i,i,j) = \left(1 - \frac{\tau}{\bar{\boldsymbol{\mathcal{S}}}^j(i,i)}\right)_+$, where t_+ is the positive part of t.

Updating $\boldsymbol{\mathcal{E}}$: Consider the following sub-problem in (11) by fixing $\boldsymbol{\mathcal{L}}$ as:

$$\begin{aligned} \boldsymbol{\mathcal{E}}^{k+1} &= \arg\min_{\boldsymbol{\mathcal{S}}} \mathcal{L}(\boldsymbol{\mathcal{L}}^{k+1}, \boldsymbol{\mathcal{E}}, \boldsymbol{\mathcal{Y}}^k, \mu^k) \\ &= \arg\min_{\boldsymbol{\mathcal{E}}} \lambda||\boldsymbol{\mathcal{E}}||_1 + \frac{\mu^k}{2}||\boldsymbol{\mathcal{F}} - \boldsymbol{\mathcal{Z}}_2||_F^2, \end{aligned} \tag{14}$$

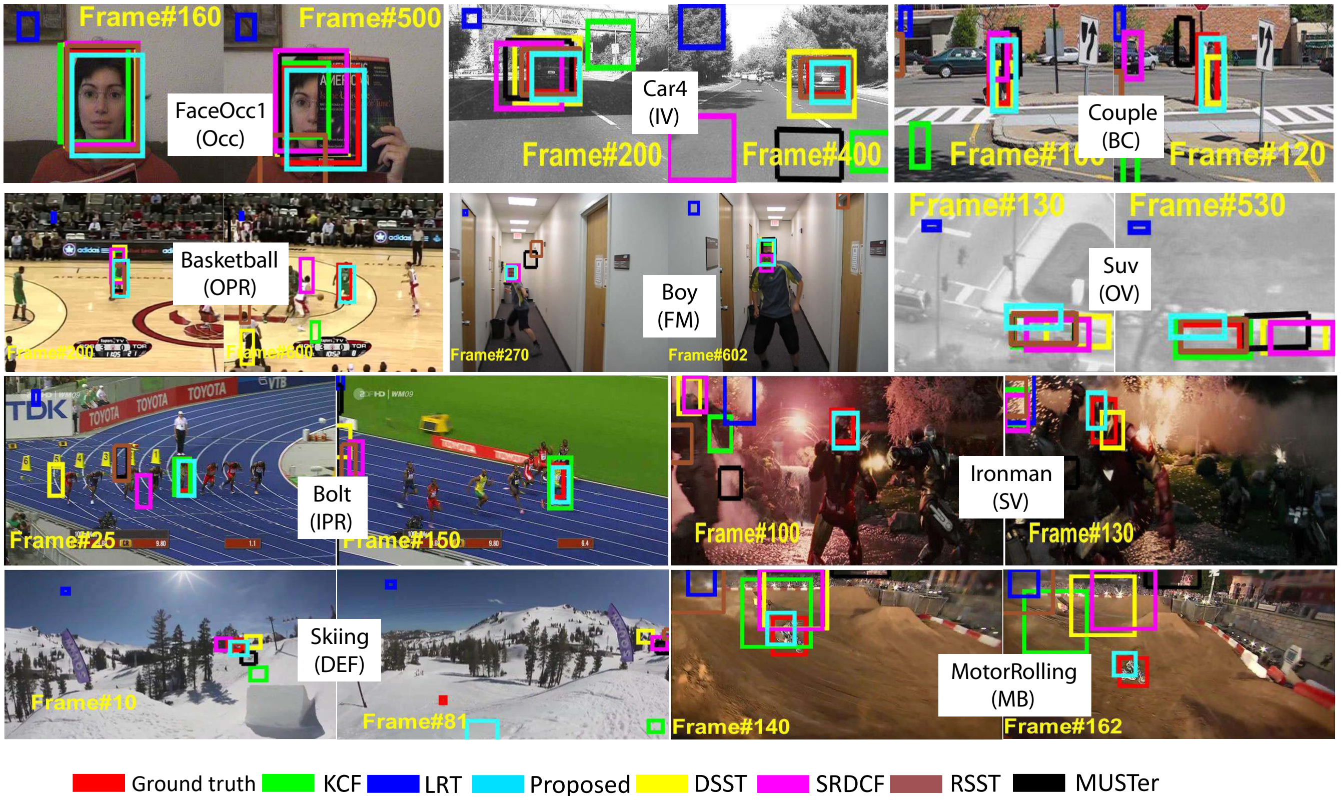

Figure 3. Visual results of the propsoed LTT tracker and its comparison with other state-of-the-art trackers including KCF [13], LRT [58], DSST [6], SRDCF [8], RSST [60], and MUSTer [14]. Frame indexes with associated tracking challenge and sequence names are shown in each row in black and yellow colors. Our proposed LTT tracker has consistently performed well against these sequences as compared to other trackers. The proposed tracker has consistently performed better than other trackers in these complex tracking scenarios.

where $\mathcal{Z}_2 = \mathcal{X} - \mathcal{B}^{k+1} + \frac{1}{\mu^k}\mathcal{Y}^k$. The closed-form solution of $\mathcal{E}^{k+1}$ can be derived as [36,61]:

$$\mathcal{E}^{k+1} = \texttt{shrink}\Big(\frac{1}{\mu^k}\mathcal{Y}^k + \mathcal{X} - \mathcal{B}^{k+1}, \frac{\lambda}{\mu^k}\Big), \tag{15}$$

where $\texttt{shrink}(c, \tau)$ (for any $c, \tau \in \mathbb{R}$) , is the element-wise soft thresholding operation [11, 16, 36, 61] defined as:

$$\texttt{shrink}(c, \tau) = \texttt{sign}(c)\max(|c| - \tau, 0). \tag{16}$$

The proof for both solutions (12) and (14) can be found in [11, 16, 36, 37, 61]. Similarly, the Lagrangiang multiplier can be updated as:

$$\mathcal{Y}^{k+1} = \mathcal{Y}^k + \mu_k(\mathcal{X} - \mathcal{L}^{k+1} - \mathcal{E}^{k+1}) \tag{17}$$

The sequences $\bar{\mathcal{L}}$ and $\bar{\mathcal{E}}$ generated by the ADMM model (11) converges to the optimal solution when $||\mathcal{L}^{k+1} - \mathcal{L}^k||_\infty \leq \eta$, $||\mathcal{E}^{k+1} - \mathcal{E}^k||_\infty \leq \eta$, $||\mathcal{L}^{k+1} - \mathcal{E}^{k+1} - \mathcal{X}|| \leq \eta$, where $||\cdot||_\infty$ dentoes the infinity norm and it is equal to $||\mathcal{X}||_\infty = \max_{ijk}|x_{ijk}|$ and $\eta = 1e - 8$ [36].

4. Experimental Evaluations

We evaluated the performance of our proposed LTT tracker on 20 challenging video sequences including *Bird2*, *Car4*, *Bolt*, *FaccOcc1*, *Girl2*, *Lemming*, *Suv*, *Tiger1*, *David2*, *Dog*, *Board*, *Coupon*, *Skater*, *Subway*, *Toy*, *Couple*, *Box*, *MotorRolling*, *Ironman*, and *Biker* taken from OTB13 [51] and OTB15 [52] datasets. These sequences pose the 11 different tracking challenges including Occlusion (Occ), Fast Motion (FM), Illumination Variation (IV), Scale Variation (SV), In-Plane Rotation (IPR), Out of Plane Rotation (OPR), Background Clutter (BC), Low Resolution (LR), Motion Blur (MB), Out-of-View (OV), and Deformation (Def). These sequences also contain ground truth object locations and varying resolutions.

We compared the performance of the proposed LTT tracker with 10 existing current state-of-the-art trackers including KCF [13], LRT [58], DSST [6], SRDCF [8], RSST [60], MUSTer [14], LCT [39], DLT [48], BACF [23], and Struck [12]. For comparison purpose, we used the publicly available source codes and we use the implementations provided by the original author's. The number of particles in our LTT tracker is set as $n = 1000$. All the ex-

Table 1. Comparison of the average Center Location Error (CLE) in pixels of 10 different trackers on 20 sequences. Top and second best performing tracker is shown in red and blue colors, respectively.

Categories	Bird2	Car24	Bolt	FaccOcc1	Girl2	Lemming	Suv	Tiger1	David2	Dog	Board	Coupon	Skater	Subway	Toy	Couple	Box	MotorRolling	Ironman	Biker	Average
LCT	2.9	6.5	7.2	8.7	4.1	62.3	193	19.8	6.2	7.0	7.2	7.5	10.7	5.2	2.5	6.6	7.1	8.8	3.5	4.5	19.0
RSST	**1.1**	3.5	4.0	4.2	4.7	75.8	**37.0**	**4.7**	9.1	9.8	6.4	9.2	**10.6**	3.1	2.5	**3.8**	**5.1**	**1.6**	9.6	9.9	10.7
BACF	2.5	3.4	**3.5**	**1.9**	2.8	59.3	174.2	12.4	140	6.8	**2.5**	**6.2**	12.8	**2.8**	**1.5**	4.1	**3.1**	3.4	**1.7**	**3.6**	22.4
KCF	6.0	9.8	8.0	7.7	3.9	**15.3**	194.9	21.1	7.6	112.5	12.8	10.2	18.6	6.3	2.8	4.6	3.8	3.6	4.2	5.6	22.7
MUSTer	2.2	**1.8**	3.7	2.6	**2.4**	23.0	129.3	7.1	7.2	**8.0**	4.1	9.0	15.1	3.6	2.2	5.1	6.2	2.2	3.2	3.7	12.0
Struck	2.7	4.8	6.8	5.3	5.2	106.1	129.7	16	38.8	11.9	15.7	9.3	10.9	375	4.4	11.2	7.6	3.3	2.9	3.9	38.5
DLT	2.7	2.2	4	2.4	**2.4**	109.6	84.4	11.5	**5.9**	61.5	4.2	178.3	12.4	5.2	3.6	6.6	8.1	1.9	4.2	6.8	25.7
LRT	5.2	9.6	5.5	2.3	7.9	9.9	66.3	11.6	6.6	10.2	4.4	3.7	20.4	15.6	7.4	7.1	9.9	1.8	5.1	3.1	10.3
DSST	1.4	3.2	4.5	2.8	4.5	7.5	39.6	5.6	6.1	11.5	4.7	11.2	3.6	3.1	2.8	6.2	5.5	5.6	4.4	4.2	**6.7**
SRDCF	2.1	2.5	4.2	2.1	2.5	16.3	41.2	6.8	7.2	11.1	4.6	7.6	11.1	3.5	1.9	9.1	7.9	6.1	6.4	4.5	7.6
Proposed	**1.0**	**1.2**	**1.7**	**0.9**	**1.2**	**4.8**	**31.1**	**1.2**	**2.7**	**3.2**	**0.5**	**2.7**	**4.1**	**0.6**	**0.2**	**3.6**	4.8	**1.4**	**2.3**	**3.1**	**3.6**

Table 2. Comparison of the average Overlap Score (OS) of 10 seven different trackers on 20 sequences. Top and second best performing tracker is shown in red and blue colors, respectively.

Categories	Bird2	Car24	Bolt	FaccOcc1	Girl2	Lemming	Suv	Tiger1	David2	Dog	Board	Coupon	Skater	Subway	Toy	Couple	Box	MotorRolling	Ironman	Biker	Average
LCT	0.74	0.74	0.67	0.67	0.75	0.14	0.16	0.61	0.59	**0.73**	0.67	**0.75**	0.65	0.70	0.79	0.72	0.66	0.76	0.78	0.72	0.65
RSST	**0.83**	0.78	0.61	0.80	0.71	0.19	0.33	**0.74**	0.54	0.68	0.36	0.71	**0.67**	0.74	0.78	0.70	0.68	0.75	0.75	0.74	0.65
BACF	0.76	0.75	0.74	0.65	0.76	0.23	0.13	0.72	0.16	**0.73**	**0.84**	**0.78**	0.57	0.82	0.74	0.75	0.70	0.69	0.72	0.69	0.64
KCF	0.60	0.48	0.53	0.62	0.78	0.42	0.14	0.61	0.49	0.04	0.36	0.72	0.54	0.67	0.76	0.70	0.62	0.76	0.75	0.71	0.56
MUSTer	0.78	**0.87**	0.73	0.76	0.79	0.33	0.1	**0.79**	0.49	0.72	0.74	0.75	0.51	**0.76**	**0.80**	0.74	0.64	0.72	0.78	0.74	0.67
Struck	0.74	0.49	0.52	0.61	0.75	0.15	0.15	0.62	0.43	0.63	0.36	0.75	0.63	0.01	0.72	0.69	0.64	0.78	0.79	0.75	0.56
DLT	0.71	0.73	0.69	**0.82**	**0.84**	0.17	0.13	0.68	**0.61**	0.13	0.54	0.04	0.59	0.70	0.73	0.71	0.71	0.74	0.73	0.73	0.58
LRT	0.23	0.31	0.41	0.33	0.34	0.41	0.46	0.40	0.39	0.45	0.34	0.36	0.42	0.49	0.51	0.33	0.42	0.22	0.16	0.29	0.36
DSST	0.65	0.67	**0.80**	0.67	0.82	**0.82**	**0.79**	0.74	0.56	0.65	0.74	0.72	0.62	0.70	0.78	**0.81**	**0.83**	**0.79**	**0.80**	**0.76**	**0.73**
SRDCF	0.71	0.72	0.78	0.72	0.80	**0.86**	**0.75**	0.78	0.60	0.68	0.78	0.70	0.65	0.72	0.75	0.75	0.68	0.72	0.77	0.74	**0.73**
Proposed	**0.88**	**0.89**	**0.82**	**0.84**	**0.91**	0.64	0.44	**0.79**	**0.64**	**0.77**	**0.87**	**0.78**	**0.70**	**0.77**	**0.83**	**0.76**	**0.72**	**0.81**	**0.84**	**0.80**	**0.77**

periments are carried out on a machine with an Intel core i7 4.0 GHz CPU with 64 GP RAM. We report the execution time of the proposed LTT tracker on aforementioned 20 sequences. The proposed tracker is able to track a target object at 8.85 frames per second where the optimization methods convereges within 6 iterations for each frame. The computational complexity of the proposed tracker is $O(n_1 n_2 n log n + n_1 n_2^2 n)$ per-iteration.

For quantitative comparison, we used two well-known tracking measures including Centre Location Error (CLE) and Overalp Score (OS) for 20 video sequences. The CLE is the Euclidean distance while OS is the intersection over union between the ground truth and the predicted bounding boxes.

4.1. Qualitative Results

Figure 3 shows the visual results of the proposed tracker as a bounding boxes on 10 tracking sequences and its visual comparison with 6 current state-of-the-art trackers on some specific key frames. These frames present tracking challenges to the object trackers making it very difficult to locate the target object. Thus, the visual analysis in these key frames is important to evaluate the qualitative performance of the trackers.

The proposed tracker perfectly handled scenes with IV (*Car4* sequence), BC (*Couple* sequence), Def (*Skiing* sequence), Occ (*FaceOcc1* sequence), OPR (*BasketBall* seuqence), FM (*Boy* sequence), OV (*Suv* sequence), IPR (*Bolt* sequence), SV (*Ironman* sequence), and MB (*Motor-Rolling* sequence) challenges as compared to the second best performing trackers DSST, KCF, and MUSTer. The excellent performance of LLT tracker suggests that it is able to mitigate outlier effects by estimating the correlation among all tensor modes into low-rank objective function. This brings biggest performance gain for achieving state-of-the-art tracking performance.

In addition to it, the following components are also beneficial in enhancing tracking performance. First, incorporating 3-D tensor structure into the low-rank representation is robust to object Def and partial Occ. Second, encoding t-SVD based nuclear norm constraints is important for handling drastic appearance changes and obtaining reliable tracking results. Finally, estimating target appearance subspace from mixture of subspaces makes low-rank learning very robust against rapid illumination variations and target drifting.

4.2. Qualitative Results

Tables 1 and 2 show the qualitative performance of the proposed tracker with 10 current state-of-the-art trackers in terms of average CLE and OS on 20 videos. These sequences present 11 complex tracking attributes as mentioned above. On the average, the proposed LTT tracker have performed better than the compared trackers in terms of both precision and success measures. For instance, LTT is 3%, 4% and 4%, 4% better than the second best performing trackers SRDCF and DSST in terms of precision and success measure.

In Tables 1 and 2, four videos including *car24*, *Box*, *MotorRolling*, and *Ironman*, presented IV tracking attribute. In terms of average CLE (Table 1), the proposed LTT tracker outperformed MUSTer and RSST trackers on two sequences with an average CLE of 1.2 and 1.4, respectively. While, in terms of average OS (Table 2), the proposed LTT tracker achieved the best performance than MUSTer and DSST trackers on three sequences with an average OS of 0.89, 0. 81, and 0.80, respectively.

The sequences *Biker, Box, Bird2, FaceOcc1, Lemming, Suv, and Girl2,* respectively, underwent Occ tracking attribute. The proposed LTT tracker achieved the best performance in terms of average CLE (Table 1) of 3.1, 1.0, 0.9, 4.8, 31.1, and 1.2, respectively, on six sequences than the second best trackers RSST, DLT, BACF, and MUSTer. On the other hand, the proposed tracker obtained significant improvements of 0.88, 0.84, 0.80, and 0.91, respectively, on 4 sequences than the compared trackers in term of average OS (Table 2). In case of SV tracking challenge (sequences *Biker, Box, Couple*, and *Dog*, respectively), the proposed LTT tracker achieved the best performance of 3.1, 2.7, and 3.2, respectively, in terms of average CLE than all of the compared trackers as shown in Table 1. In contrast, the LTT tracker could only obtained the best performance of 0.80 and 0.77, respectively, on two sequences in terms of average OS than the second best performing trackers in Table 2. For BC tracking attribute (sequences *Couple, Board*, and *Coupon*, respectively), the proposed LTT tracker obtained the best performance of 3.6, 0.5, and 2.7, respectively, in terms of average CLE than all of the compared trackers in Table 1. While, the LTT tracker achieved noticeable improvements of 0.87 and 0.78, respectively, on two BC sequences in terms of average OS in Table 2. For FM tracking attribute (sequences *Ironman, Tiger1*, and *Toy*, respectively), the proposed LTT tracker obtained the best performance of 0.2 and 1.2, respectively, on two sequences in terms of average CLE as shown in Table 1. In terms of average OS (Table 2), the LTT tracker achieved a comparative performance of 0.84, 0.83, and 0.79, respectively, on all FM sequences than compared trackers. The sequences *David2, Suv, Tiger1,* and *Toy* respectively, presented IPR challenge, where the proposed LTT tracker obtained an average CLE (Table 1) of 2.7, 31.1, 1.2, and 0.2, respectively, which is significantly better than other top performing trackers such as DLT, BACF, DSST, and SRDCF, respectively. In terms of average OS (Table 2), LTT obtained noticeable performance of 0.64, 0.79, and 0.83, respectively, on three IPR sequences than compared trackers.

Similarly, the proposed tracker also obtained comparative performance in terms of average CLE and OS on the remaining tracking attributes of OPR, MB, OV, Def, and LR, respectively. The best performance gained by the propsoed tracker is because of the inclusion of correlation among all the tensor modes of the input particles using nuclear norm regularization.

5. Conclusion

In this paper, a new framework is proposed for visual object racking with complex scenarios. The proposed algorithms have shown improved performance on many challenging sequences thanks to the new tensor nuclear norm constraints on the low-rank tensor objective function. The newly formulated tensor algebra and its nuclear norm regularization estimated over the frontal slices of input tensor models robust and holistic enforcement of the spatial appearance consistency in the low-rank target representation. The proposed objective function is solved efficiently in a batch manner using linearized ADMM optimization technique. The experimental evaluations on 20 sequences demonstrated the excellent performance and the superiority of the proposed algorithms in the presence of a wide variety of challenging scenarios when compared with 10 current state-of-the-art trackers using publicly available dataset. In the future, we aim to develop saliency-fused optimization algorithm to maintain the re-detection procedure in order to increase the robustness of the tracker to handle more challenging scenarios.

Acknowledgement

This publication is based upon work supported by the Khalifa University of Science and Technology under Award No. RC1-2018-KUCARS.

References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE T-PAMI*, 33(8):1619–1632, 2011.

[2] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *IEEE CVPR*, pages 3457–3464, 2011.

[3] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In *IEEE CVPR*, 2010.

[4] T. Bouwmans, S. Javed, H. Zhang, Z. Lin, and R. Otazo. On the applications of robust pca in image and video processing. *P-IEEE*, 106(8):1427–1457, 2018.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *FATML*, 3(1):1–122, 2011.

[6] M. Danelljan, G. Häger, F. Khan, and M. Felsberg. Accurate scale estimation for robust visual tracking. In *BMVC*, 2014.

[7] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg. Discriminative scale space tracking. *IEEE T-PAMI*, 39(8):1561–1575, 2017.

[8] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg. Learning spatially regularized correlation filters for visual tracking. In *IEEE CVPR*.

[9] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *ECCV*, pages 472–488, 2016.

[10] M. Fiaz, A. Mahmood, S. Javed, and S. K. Jung. Handcrafted and deep trackers: A review of recent object tracking approaches. *ACM Surveys*, 2018.

[11] J. Friedman, T. Hastie, H. Höfling, R. Tibshirani, et al. Pathwise coordinate optimization. *Ann. of App. Stats.*, 1(2):302–332, 2007.

[12] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr. Struck: Structured output tracking with kernels. *IEEE T-PAMI*, 38(10), 2016.

[13] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Highspeed tracking with kernelized correlation filters. *IEEE T-PAMI*, 37(3):583–596, 2015.

[14] 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 *IEEE CVPR*, 2015.

[15] Z. Hong, X. Mei, D. Prokhorov, and D. Tao. Tracking via robust multi-task multi-view joint sparse representation. In *IEEE CVPR*.

[16] W. Hu, Y. Yang, W. Zhang, and Y. Xie. Moving object detection using tensor-based low-rank and saliently fused-sparse decomposition. *IEEE T-IP*, 26(2):724–737, 2016.

[17] 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 T-IP*, 28(2):1007–1022, 2018.

[18] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung. Spatiotemporal low-rank modeling for complex scene background initialization. *IEEE T-CSVT*, 28(6):1315–1329, 2016.

[19] S. Javed, A. Mahmood, T. Bouwmans, and S. K. Jung. Background–foreground modeling based on spatiotemporal sparse subspace clustering. *IEEE Transactions on Image Processing*, 26(12):5840–5854, 2017.

[20] S. Javed, A. Mahmood, N. Werghi, and D. Jorge. Structural low-rank tracking. In *IEEE AVSS*, 2019.

[21] X. Jia, H. Lu, and M.-H. Yang. Visual tracking via adaptive structural local sparse appearance model. In *IEEE CVPR*, 2012.

[22] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IEEE-ACM IWAR, year=1999*.

[23] H. Kiani Galoogahi, A. Fagg, and S. Lucey. Learning background-aware correlation filters for visual tracking. In *IEEE ICCV*, 2017.

[24] M. E. Kilmer and C. D. Martin. Factorization strategies for third-order tensors. *Lin. Alg. and Its App.*, 435(3):641–658, 2011.

[25] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[26] M. Kristan, A. Leonardis, J. Matas, and M. Felsberg. The visual object tracking vot2016 challenge results. In *Springer ECCVW*, 2016.

[27] A. Li, M. Lin, Y. Wu, M.-H. Yang, and S. Yan. Nus-pro: A new visual tracking challenge. *IEEE T-PAMI*, 38(2):335–349, 2016.

[28] C. Li, L. Lin, W. Zuo, J. Tang, and M.-H. Yang. Visual tracking via dynamic graph learning. *IEEE T-PAMI*, 2018.

[29] H. Li, C. Shen, and Q. Shi. Real-time visual tracking using compressive sensing. In *IEEE CVPR*, 2011.

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

[31] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown. Rain streak removal using layer priors. In *IEEE CVPR*, 2016.

[32] P. Liang, E. Blasch, and H. Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE T-IP*, 24(12).

[33] B. Liu, J. Huang, C. Kulikowski, and L. Yang. Robust visual tracking using local sparse appearance model and k-selection. *IEEE T-PAMI*, 35(12):2968–2981, 2013.

[34] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski. Robust and fast collaborative tracking with two stage sparse optimization. In *ECCV*, 2010.

[35] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan. Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization. In *IEEE CVPR*, 2016.

[36] C. Lu, J. Feng, W. Liu, Z. Lin, S. Yan, et al. Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE T-PAMI*, 2019.

[37] C. Lu, J. Feng, S. Yan, and Z. Lin. A unified alternating direction method of multipliers by majorization minimization. *IEEE T-PAMI*, 40(3):527–541, 2017.

[38] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang. Hierarchical convolutional features for visual tracking. In *IEEE ICCV*, 2015.

[39] C. Ma, X. Yang, C. Zhang, and M.-H. Yang. Long-term correlation tracking. In *IEEE CVPR*, 2015.

[40] X. Mei and H. Ling. Robust visual tracking and vehicle classification via sparse representation. *IEEE T-PAMI*, 33(11):2259–2272, 2011.

[41] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai. Minimum error bounded efficient l1 tracker with occlusion detection. In *IEEE CVPR*, 2011.

[42] H. Peng, B. Li, H. Ling, W. Hu, W. Xiong, and S. J. Maybank. Salient object detection via structured matrix decomposition. *IEEE T-PAMI*, 39(4):818–832, 2016.

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

[44] O. G. Selfridge, R. S. Sutton, and A. G. Barto. Training and tracking in robotics.

[45] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller. Tensor-based formulation and nuclear norm regularization for multienergy computed tomography. *IEEE T-IP*, 23(4):1678–1693, 2014.

[46] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking: An experimental survey. *IEEE T-PAMI*, 36(7):1442–1468, 2014.

[47] A. Smith. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.

[48] N. Wang and D.-Y. Yeung. Learning a deep compact image representation for visual tracking. In *A-NIPS*, 2013.

[49] P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Learning to track for spatio-temporal action localization. In *IEEE CVPR*.

[50] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *A-NIPS*, 2009.

[51] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *IEEE CVPR*, 2013.

[52] Y. Wu, J. Lim, and M.-H. Yang. Object tracking benchmark. *IEEE T-PAMI*, 37(9):1834–1848, 2015.

[53] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM-CS*, 38(4):13, 2006.

[54] C. Zhang, R. Liu, T. Qiu, and Z. Su. Robust visual tracking via incremental low-rank features learning. *Neurocomputing*, 131:237–247, 2014.

[55] K. Zhang, L. Zhang, and M.-H. Yang. Real-time compressive tracking. In *ECCV*, 2012.

[56] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Low-rank sparse learning for robust visual tracking. In *ECCV*, 2012.

[57] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. In *IEEE CVPR*, 2012.

[58] T. Zhang, S. Liu, N. Ahuja, M.-H. Yang, and B. Ghanem. Robust visual tracking via consistent low-rank sparse learning. *IJCV*, 111(2):171–190, 2015.

[59] T. Zhang, C. Xu, and M.-H. Yang. Learning multi-task correlation particle filters for visual tracking. *IEEE T-PAMI*, 41(2):365–378, 2019.

[60] T. Zhang, C. Xu, and M.-H. Yang. Robust structural sparse tracking. *IEEE T-PAMI*, 41(2):473–486, 2019.

[61] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *IEEE CVPR*, pages 3842–3849, 2014.

[62] T. Zhou, F. Liu, H. Bhaskar, and J. Yang. Robust visual tracking via online discriminative and low-rank dictionary learning. *IEEE T-C*, 48(9):2643–2655, 2018.