Multi-Dimensional Visual Data Completion via Low-Rank Tensor Representation Under Coupled Transform

Jian-Li Wang, Ting-Zhu Huang, Xi-Le Zhao, Tai-Xiang Jiang, and Michael K. Ng, Senior Member, IEEE

Abstract—This paper addresses the tensor completion problem, which aims to recover missing information of multidimensional images. How to represent a low-rank structure embedded in the underlying data is the key issue in tensor completion. In this work, we suggest a novel low-rank tensor representation based on coupled transform, which fully exploits the spatial multi-scale nature and redundancy in spatial and temporal/spatial dimensions, leading to a better low tensor multi-rank approximation. More precisely, this representation is achieved by two-dimensional Fourier transform for the two spatial dimensions, one/two-dimensional Fourier transform for the temporal/spatial dimension, and then Karhunen–Loève transform (via singular value decomposition) for the transformed tensor. Based on this low-rank tensor representation, we formulate a novel low-rank tensor completion model for recovering missing information in multi-dimensional visual data, which leads to a convex optimization problem. To tackle the proposed model, we develop the alternating directional method of multipliers (ADMM) algorithm tailored for the structured optimization problem. Numerical examples on color images, multispectral images, and videos illustrate that the proposed method outperforms many state-of-the-art methods in qualitative and quantitative aspects.

Index Terms—2D framelet transform, multi-scale representation, tensor nuclear norm, tensor completion.

I. INTRODUCTION

As the high-dimensional extension of vector/matrix [1]–[3], tensor provides a more diverse and flexible representation for multi-dimensional visual data, which usually contains two spatial dimensions and another temporal (or spectral) dimension, such as color images [4], videos [5], [6], hyperspectral images [7]–[13], seismic data [14], etc. Unfortunately, due to hardware restrictions and various degradations, the obtained data are usually incomplete, which significantly degrades the visual quality and limits the subsequent processing tasks. The problem of recovering the missing information in multi-dimensional visual data from its small known observations is called tensor completion (TC) [15]–[18], which is a typical inverse problem in image processing. An effective restoration process generally relies on prior knowledge about the desired solution.

Low-rankness is a powerful tool to describe the internal redundancy of tensor, and how to exploit the embedded low-rank structure has been widely studied in TC [19]. Mathematically, a unified low-rank tensor completion (LRTC) model can be written as

$$\arg\min_{\mathcal{X}} \text{rank}(\mathcal{X})$$

s.t. $\mathcal{P}_\Omega(\mathcal{X}) = \mathcal{P}_\Omega(\mathcal{O})$,

where $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_k}$ is the underlying tensor, $\mathcal{O} \in \mathbb{R}^{n_1 \times \cdots \times n_k}$ is the observed data, $\Omega$ is the index set of the observed data, and $\mathcal{P}_\Omega(\cdot)$ is the sampling operator that retains the elements in $\Omega$ while making the others to be zeros. In LRTC, a central issue is the definition of the tensor rank, i.e., how to characterize the correlations along different dimensions within a data. However, it is a difficult task due to the complex algebraic structure of tensors, and there is no universally accepted solution to the definition of tensor rank [20]. Over the past decades, many research efforts have been devoted to this topic, such as CANDECOMP/PARAFAC (CP) rank [21], [22], Tucker rank [23], [24], and tubal rank [25], [26], all of which capture the tensor redundancy from their perspectives.

The CP decomposition [22] decomposes a tensor into a sum ofrank-one factors constructed by the vector outer product, and the CP-rank is defined as the smallest number of these factors among all such decompositions. However, calculating the CP-rank of a known tensor is an NP-hard problem [27], [28],
and there is generally no accurate algorithm for estimating CP-rank. The Tucker decomposition factorizes a tensor into a product of a core tensor and factor matrices, and the Tucker rank (also called as “n-rank”) [23], [24], [29] is a vector composed of the ranks of unfolding matrices of the target tensor along different dimensions [30]. However, Tucker rank suffers from a limitation that the global correlation within the tensor is destroyed by its unfolding scheme.

Recently, a new tensor decomposition method called the tensor singular value decomposition (t-SVD) [25], [26] was proposed for third-order tensors, which decomposes a three-dimensional tensor into the product of two orthogonal tensors and one f-diagonal tensor (see Section II for details). t-SVD has the advantage of characterizing the correlation along the third dimension via constructing group-rings along the tensor fibers, i.e., executing the 1D fast Discrete Fourier Transform (DFT) along the third mode. Based on the t-SVD framework, a new tensor tubal rank [31], [32] is defined as the number of non-zero tubes of the f-diagonal tensor in t-SVD. However, direct minimizing tubal rank is an NP-hard problem. As the convex surrogate of tensor tubal rank, tensor nuclear norm (TNN) [25], [33], [34] is considered for tubal rank minimization with a strong theoretical guarantee for LRTC.

Later works [35] have been proposed to improve the performance of TNN. Considering that TNN is a biased approximation [36]–[38] to the tubal rank, some works replace the nuclear norm with non-convex surrogates [36]–[38] to obtain a better low-rank approximation, including weighted tensor nuclear norm [39], Laplace function [40], and the partial sum of tubal nuclear norm [37]. Other works focus on a more effective characterization of the data correlation by finding a more suitable transform than DFT, such as invertible linear transform [41], [42], multiple linear transform [43], [44], and framelet transform [45].

Recently, deep learning (DL)-based completion methods [46]–[48] have been rapidly developed to learn deep image priors from a large number of example images and have shown promising performance due to its high capacity. However, most of these methods are designed for particular tasks, e.g., inpainting (tube-wise sampling) [46]–[48], and their performance is essentially dependent upon the diversity and volume of training datasets. Therefore, the lack of generalization hinders its direct application to the general tensor completion problem for diverse samplings and data compared with regularized LRTC methods.

The above-mentioned LRTC methods only consider the global data correlation (i.e., the low-rankness) but ignore the other significant properties, such as the spatial multi-scale nature and redundancy in the spatial and spectral/temporal dimensions. As a result, those methods can only recover the global cartoon but fail to repair specific textures, especially when the sampling rate is low. To fully exploit the intrinsic structure within multi-dimensional visual data, we propose a novel low-rank tensor representation under coupled transform, which can characterize the correlations along different dimensions in a unified framework. The main idea is to explore suitable transforms to decorrelate the spatial and temporal (or spectral) dimensions, achieving an enhanced low-rank representation of the underlying data. More precisely, the proposed representation involves three layers: $B = B_3 \circ B_2 \circ B_1$.

- In the first layer $B_1$, we use a two-dimensional framelet transform with abundant bases to describe the local spatial correlation, which has two advantages. First, with the help of the filters to construct the framelet system, the abundant spatial structures can be elaborately expressed with respect to different frequencies and spatial directions. Second, the multi-level nature of the framelet decomposition helps to capture the multi-scale features of the tensor, as shown in Fig. 1 (a).
- In the second layer $B_2$, we use a Fourier transform to characterize the global correlation of the temporal (or spectral) dimension. This layer characterizes the temporal (or spectral) global correlation by constructing group-rings along the corresponding dimensions and reveals the intrinsic low-rank property of the multi-scale representation coefficients achieved by the first layer, in which the spatial features of different directions and levels are gathered together with strong correlations.
- In the third layer $B_3$, we use a Karhunen–Loéve (KL) transform (via singular value decomposition), regarded as a special transform, to characterize the global spatial correlation (i.e., low-rankness) of the representation coefficients by the first two layers, which incorporates both the global correlation and local geometric details in a unified framework. The sparsity of the result after KL transform is indeed the rank of the spatial slice of the framelet and Fourier transformed result.

Fig. 1 gives an illustration of the proposed representation on the color image “Butterfly”, in which Fig. 1 (a) shows the third-layer structure of the proposed representation; Fig. 1 (b) shows the comparison of the accumulation energy ratio (AccEgy = $\sum_{i=1}^{k} \sigma_i^2 / \sum_{j} \sigma_j^2$, where $\sigma_i$ is the i-th singular value) of the proposed representation and t-SVD. Form Fig. 1 (b), we can observe that the proposed representation always achieve higher structural similarity (SSIM) values and lower relative squared error (RSE) than t-SVD on the same AccEgy, which implies that our representation coefficients exhibit higher sparsity, i.e., the representation is more efficient in terms of data compression.

Based on the above sparsity of the representation coefficients, we formulate a novel model for LRTC under coupled transform, termed as CT-LRTC, which is expressed as follows:

$$\min_{\mathbf{X}} \|B(\mathbf{X})\|_1$$

s.t. $P_\Omega(\mathbf{X}) = P_\Omega(O)$,

where $B = B_3 \circ B_2 \circ B_1$ is coupled transforms which involves three layer, i.e., the framelet transform, Fourier transform, and KL transform.

Our work also provides a unified framework that can accommodate the previous TNN-based methods, as well as a deconstruction perspective to view their inner mechanisms. In our framework, they can be viewed as a compound of transforms, such as FFT+KL (TNN) [33] and DCT+KL [41] (see Section II-B for more details).
The main contributions of this work are summarized as follows.

- We propose a novel low-rank tensor representation under coupled transform for multi-dimensional visual data, which provides an enhanced low-rank approximation and a novel perspective to exploit the implicit low-rank structure, i.e., being not directly low-rank in the original domain but low-rank in the transformed domain with well-chosen transformation.

- We formulate a novel low-rank tensor completion model for recovering missing information in multi-dimensional visual data based on the proposed low-rank tensor representation, and develop an efficient alternating directional method of multipliers (ADMM) [49] solving algorithm.

- Numerical examples on color images, multispectral images, and videos illustrate the superiority and effectiveness of the proposed CT-LRTC compared with many existing methods.

The rest of this paper is organized as follows. Section II gives some preliminaries of tensor used throughout this paper. Section III presents the proposed CT-LRTC and the corresponding ADMM solver in detail. Section IV displays experimental results to verify the effectiveness of the proposed CT-LRTC. Finally, Section V concludes our work.

II. PRELIMINARIES

In this section, we give some preliminary knowledge of tensor and t-SVD used throughout this paper.

A. Tensor Notations

For a better reading, we summarize some basic notations of tensors in Table I. Below we introduce some necessary definitions; see [41], [50] for more details.

![Figure 1](image1.png)

**Fig. 1.** An illustration of the proposed CT-LRTC on the color image “Butterfly”. (a) Shows the three-layer structure of the proposed representation. (b) Compares the accuracy of the proposed representation and t-SVD with respect to the change of AccEgy and shows the image constructed by AccEgy = 0.6.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x, x, X, X'</td>
<td>Scalar, vector, matrix, tensor.</td>
</tr>
<tr>
<td>T</td>
<td>The identity tensor, whose first frontal slice is the identity matrix, and the other frontal slices are zero matrices.</td>
</tr>
<tr>
<td>X^(i)</td>
<td>The i-th frontal slice of X'.</td>
</tr>
<tr>
<td>unfold_m(X)/X(n)</td>
<td>The mode-n unfolding of X'.</td>
</tr>
<tr>
<td>fold_m(X)</td>
<td>The folding operation of X along the mode-n.</td>
</tr>
<tr>
<td>X̂</td>
<td>The DFT result of X along its third dimension e.g., X̂ = DFT(X, 3).</td>
</tr>
<tr>
<td>|X|_F</td>
<td>The Frobenius norm of X, which is the square root of the squared sum of each element of X, i.e., |X|<em>F = \sqrt{\sum</em>{i,j,k} x_{ijk}^2}.</td>
</tr>
<tr>
<td>|X|_{TNN}</td>
<td>The TNN of X, which is the sum of the nuclear norm of all frontal slices of X̂, i.e., |X|<em>{TNN} = \sum</em>{i=1}^{n_3} |X^{(i)}|_2.</td>
</tr>
<tr>
<td>X^H</td>
<td>The conjugate transpose of X, which is obtained by conjugate transposing each frontal slice and then reversing the order of the transposed frontal slices 2 through n_3, i.e., X^{(i)}(1)H = (X^{(i)}H(1)) and X^{(n_3+2-i)}H = (X^{(i)}H(1)) (i = 2, · · · , n_3).</td>
</tr>
</tbody>
</table>

B. The t-SVD Framework

In this subsection, we briefly introduce the t-SVD framework since the proposed CT-LRTC is somewhat related to TNN. Here are some related definitions.

**Definition 2.1 (t-Product [41])**: The tensor-tensor product of X ∈ R^{n_1×n_2×n_3} and Y ∈ R^{n_2×n_4×n_3} is defined as

\[ X \ast Y = \text{Fold} (\text{bcirc} (X) \cdot \text{Unfold} (Y)) \in R^{n_1×n_4×n_3}, \]
where \( \text{bcirc}(\mathbf{X}) \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) is the following block circulant matrix

\[
\text{bcirc}(\mathbf{X}) := \begin{bmatrix}
X(1) & X(n_3) & \ldots & X(2) \\
X(2) & X(1) & \ldots & X(3) \\
\vdots & \vdots & \ddots & \vdots \\
X(n_3) & X(n_3-1) & \ldots & X(1)
\end{bmatrix},
\]

Unfold(\( \mathbf{Y} \)) is a matrix of size \( n_2 \times n_3 \times n_4 \):

\[
\text{Unfold}(\mathbf{Y}) = \begin{bmatrix}
Y(1) \\
Y(2) \\
\vdots \\
Y(n_3)
\end{bmatrix}, \quad \text{Fold}(\text{Unfold}(\mathbf{Y})) = \mathbf{Y}.
\]

When \( n_3 = 1 \), the t-product degenerates to a matrix-matrix product.

**Definition 2.2 (Orthogonal Tensor [50]):** A tensor \( \mathbf{Q} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) is called an orthogonal tensor, if it satisfies \( \mathbf{Q}^T \mathbf{Q} = \mathbf{Q} \mathbf{Q}^T = \mathbf{I} \).

**Definition 2.3 (f-Diagonal Tensor [50]):** A tensor is called an f-diagonal tensor, if each of its frontal slices is a diagonal matrix.

**Definition 2.4 (t-SVD [41]):** Every tensor \( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) can be factorized as

\[
\mathbf{X} = \mathbf{U} \ast \mathbf{S} \ast \mathbf{V}^T,
\]

where \( \mathbf{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) is an f-diagonal tensor, and \( \mathbf{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3} \) and \( \mathbf{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3} \) are two orthogonal tensors.

**Definition 2.5 (Tensor Multi Rank [33]):** The multi rank of \( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) is a vector \( \mathbf{r} \in \mathbb{R}^{n_3} \), whose i-th element equals to the rank of i-th frontal slice of \( \mathbf{X} \), i.e., \( r_i = \text{rank} (X(:, :, i)) \).

**Definition 2.6 (Tensor Tubal Rank [33]):** The tubal rank of \( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) is defined as the number of non-vanishing tubes of \( \mathbf{S} \) in \( \mathbf{X} = \mathbf{U} \ast \mathbf{S} \ast \mathbf{V}^T \).

\[
\text{rank}_{\text{tubal}}(\mathbf{X}) = \sharp \{ i : \mathbf{S}(i, i, :) \neq \mathbf{0} \}.
\]

### III. Low-Rank Tensor Completion Under Coupled Transform

This section is divided into three parts. Subsection III-A gives the proposed CT-LRTC for tensor completion. Subsection III-B provides justification for several key ideas of the proposed transform. Subsection III-C provides the corresponding ADMM [49] algorithm with guaranteed convergence in details.

#### A. The Proposed CT-LRTC

Based on the above sparsity of the representation coefficients, we formulate a novel model for LRTC under coupled transforms, termed as CT-LRTC, which is as follows:

\[
\min_{\mathbf{X}} \| \mathbf{X} \|_1 \quad \text{s.t.} \quad \mathbf{P}_{\mathbf{Q}}(\mathbf{X}) = \mathbf{P}_{\mathbf{Q}}(\mathbf{O}),
\]

where \( \mathbf{X} \in \mathbb{R}^{n_1 \times \cdots \times n_k} \) is the underlying tensor, \( \mathbf{O} \in \mathbb{R}^{n_1 \times \cdots \times n_k} \) is the observed data, \( \mathbf{Q} \) is the index set of the observed data, \( \mathbf{P}_{\mathbf{Q}}(\cdot) \) is the sampling operator that remains the elements in \( \mathbf{Q} \) while making the others to be zeros, and \( \mathbf{B} = \mathbf{B}_3 \circ \mathbf{B}_2 \circ \mathbf{B}_1 \) is a coupled transform which involves three layer, i.e., the framelet transform, Fourier transform, and KL transform.

In the first layer \( \mathbf{B}_1 \), the framelet transform is conducted on the spatial slices, which can be expressed as a composition of two operators, i.e., \( B_1(\mathbf{X}) = \mathbf{CL}(\mathbf{W}(\mathbf{X})) \). Here, \( \mathbf{W} : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{m_1 \times n_2 \times n_3} \) is a framelet transform operator, i.e., \( \mathbf{W}(\mathbf{X}) = \text{fold}_4 ((\mathbf{WX}^{(3)})) \), where \( \mathbf{W} \in \mathbb{R}^{m_1 \times n_2 \times n_3} \) is a framelet transform matrix with \( m \) filters and \( l \) levels, and \( \mathbf{CL} : \mathbb{R}^{m_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{m_2 \times n_2 \times n_3} \) is a reshaping operator, i.e., rearranges \( \mathbf{W}(\mathbf{X}) \) in a diagonal manner which regards \( \mathbf{W}(\mathbf{X}) \) as \( \mathbb{R}^{m_2} \) k-order tensors. In fact, \( \mathbf{B}_1 : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{m_1 \times n_2 \times n_3} \) is a linear operator. Here, the two-dimensional framelet transform \( \mathbf{W} \) is equivalent to a collection of over-complete basis over \( \mathbb{L}^2(R) \) which satisfies \( \mathbf{W}^T \mathbf{W} = \mathbf{I} \) [51]–[53], where \( \mathbf{W}^T \) and \( \mathbf{I} \) are the inverse framelet transform and equivalent transform, respectively. The detailed description of the generation process of \( \mathbf{W} \) can be found in [54, 55].

In the second layer \( \mathbf{B}_2 \), the Fourier transform is implemented on the spectral or temporal mode through Matlab’s fft operation, which is a linear operator \( \mathbf{B}_2 : \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{C}^{n_1 \times n_2 \times n_3} \). In the third layer \( \mathbf{B}_3 \), the KL transform projects the data to the bases, which is the principal directions of the data itself \( \mathbf{B}_3 : \mathbb{C}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}^{n_1 \times n_2 \times n_3} \). As a contrast, Fourier transform and framelet transform only project the data to some fixed bases.

The compound of the KL transform, framelet transform, and Fourier transform can be viewed as a couple of the predefined transform and data-adaptive transform. Moreover, the coupled transform \( \mathbf{B} \) can be efficiently obtained by calculating a series of operations in Matlab; see Algorithm 1 for more details.

We discuss the connection between the proposed work and previous TNN-based methods. On the one hand, we provide a unified framework that can accommodate the previous TNN-based methods, as well as a deconstruction perspective to view their inner mechanisms. In our framework, they can be viewed as a sparse approximation under a compound of transforms, such as FFT+KL (TNN) [33] and DCT+KL [41]. On the other hand, our method can be seen as an improvement of the existing work, namely TNN in the framelet domain. Thus, for a third-order tensor \( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \), the proposed

---

**Algorithm 1 Coupled Transform for a Third-Order Tensor**

**Input:** \( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \).

1. **Step 1:** \( \mathbf{W}(\mathbf{X}) \leftarrow \text{fold}_4 ((\mathbf{WX}^{(3)})^T); \)
2. \( \mathbf{B}_1(\mathbf{X}) \leftarrow \mathbf{CL}(\mathbf{W}(\mathbf{X})); \)
3. **Step 2:** \( \mathbf{B}_2(B_1(\mathbf{X})) \leftarrow \text{fft}(B_1(\mathbf{X}), \{1, 3\}); \)
4. **Step 3:** \( \mathbf{S} \leftarrow B_2(B_1(\mathbf{X})); \)
5. for \( i = 1 \) to \( n_3 \)
6. \( [\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}(B_2(B_1(\mathbf{X}))(i, :, i)); \)
7. \( \mathbf{S}(i, :, j) = \mathbf{S}; \)
8. end

**Output:** \( B(\mathbf{X}) \leftarrow \mathbf{S}. \)
CT-LRTC is equivalent to
\[
\min_{\mathcal{X}} \| B_1(\mathcal{X}) \|_{\text{TNN}} \\
\text{s.t.} \quad \mathcal{P}_B(\mathcal{X}) = \mathcal{P}_B(\mathcal{O}),
\]
where $B_1$ is a linear operator which is a two-dimensional framelet transform, it is computed within a local spatial neighborhood of the original tensor. The following theorem shows that the proposed model is a convex optimization problem.

**Theorem 1:** The proposed CT-LRTC (1) is a convex optimization problem.

**Proof.** The key to proving the convexity of the proposed CT-LRTC (1) is that $\{ X | \mathcal{P}_B(\mathcal{X}) = \mathcal{P}_B(\mathcal{O}) \}$ is a convex set and $\| B_1(\mathcal{X}) \|_{\text{TNN}}$ is a convex function.

First, it is easy to check that set $\{ X | \mathcal{P}_B(\mathcal{X}) = \mathcal{P}_B(\mathcal{O}) \}$ is convex by the definition.

Second, we prove that the function $\| B_1(\mathcal{X}) \|_{\text{TNN}} : \mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{R}$ is convex. In fact, for $\forall \lambda_1, \lambda_2 \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, and $0 \leq \theta \leq 1$, we have
\[
\| B_1(\theta \lambda_1 + (1 - \theta) \lambda_2) \|_{\text{TNN}} = \| \theta B_1(\lambda_1) + (1 - \theta) B_1(\lambda_2) \|_{\text{TNN}} \\
\times (B_1 \text{ is a linear operator}) \\
\leq \theta \| B_1(\lambda_1) \|_{\text{TNN}} + (1 - \theta) \| B_1(\lambda_2) \|_{\text{TNN}} \\
\times (\text{TNN} \text{ is a convex function}).
\]
Thus, the function $\| B_1(\mathcal{X}) \|_{\text{TNN}}$ is convex by definition.

Combining the above facts, the proposed CT-LRTC (1) is a convex optimization problem.

### B. The Coupled Transform

Below, we provide explanations for several key ideas of the proposal.

1) **Role of Different Transforms:** The framelet transform captures local spatial correlation, since it is computed within a local spatial neighborhood of the original tensor. The Fourier transform is believed to exploit the global temporal (or spectral) correlation, since its computation refers to the whole temporal (or spectral) vectors. The KL transform reflects the global spatial correlation (i.e., low-rankness), since it is computed within a spatial slice of the framelet and Fourier transformed result. The sparsity of the result after KL transform is indeed the rank of the spatial slice of the framelet and Fourier transformed result.

These three transforms are inherently related, and they have collaborated. As we can see, the framelet transform is applied on the spatial slices while the Fourier transform is conducted along the third mode (temporal or spectral mode). The carrying out of framelet and Fourier transforms relatively independent but spatial and temporal (or spectral) are coupled. Meanwhile, the KL transform is based on the bases related to the data distribution, being different from the fixed bases provided by framelet and Fourier transforms.

2) **Spatial and Spectral Decorrelation:** First, the framelet transform has been widely used for image processing (Please see [51], [54] for example) with its abundant bases which are greatly suitable for the spatial geometric structures, textures, and details existed in a majority of natural images. Second, along the temporal (or spectral) dimension, this framelet transformed result is further decomposed into low-frequency components and high-frequency components with respect to the Fourier bases. Thus, after the framelet and Fourier transforms, the multi-dimensional images are thoroughly decomposed with respect to the semi-orthogonal framelet bases along the spatial dimensions and the orthogonal Fourier bases along the temporal (or spectral) dimension. The introduced framelet transform faithfully exploits the spatial multi-scale nature widely existed in imaging data, especially for implicit spatial low-rank images, which results in more sparse coefficients after KL transform (i.e., more low-rankness) as compared with coupled Fourier and KL transforms (i.e., TNN).

3) **Enhanced Low-Rank Representation:** The proposed low-rank tensor representation based a coupled transform faithfully exploits the spatial multi-scale nature widely existed in imaging data, and this is expected to help generating a better low tensor multi-rank approximation. Fig. 2 plots the AccEgy with respect to the percentage of singular values of the original tensor and the framelet transformed tensor on different types of multi-dimensional data. An immediate observation is that the framelet transform together with Fourier transform significantly reformed singular values for different multi-dimensional data and the energy of the singular value of the transformed tensor is more concentrated after the framelet transform. More distinctly, as pointed out by the auxiliary dashed lines, after the framelet and Fourier transforms, the transformed data could occupy 95% of the whole energy with a smaller proportion of singular values, compared with the case in which only the Fourier transform is involved (i.e., TNN). Therefore, we can use a better tensor low multi-rank approximation to achieve the same AccEgy. The above numerical experiments justify our contribution.

### C. ADMM-Based Optimization Algorithm

We develop an ADMM [49] algorithm to solve the proposed CT-LRTC (1). By introducing an auxiliary variable $\mathcal{Y} = B_1(\mathcal{X})$, we rewrite the original problem (1) as the equivalent constrained optimization problem
\[
\min_{\mathcal{X}, \mathcal{Y}} \ l_\beta(\mathcal{X}) + \| \mathcal{Y} \|_{\text{TNN}} \\
\text{s.t.} \quad \mathcal{Y} = B_1(\mathcal{X}),
\]
where $l_\beta(\mathcal{X})$ is an indicator function defined as
\[
l_\beta(\mathcal{X}) = \begin{cases} 
0, & \mathcal{X} \in \Phi, \\
\infty, & \text{otherwise},
\end{cases}
\]
where \( \Phi := \{ X \in \mathbb{R}^{n_1 \times n_2 \times n_3}, \ P_\Omega(X) = P_\Omega(O) \} \). The corresponding augmented Lagrangian function is
\[
L_\beta(X, \mathcal{Y}, \Lambda) = I_\Omega(X) + \| \mathcal{Y} \|_{\text{TNN}} + \frac{\beta}{2} \| B_1(X) - \mathcal{Y} + \frac{\Lambda}{\beta} \|_F^2,
\]
where \( \Lambda \in \mathbb{R}^{m_1 \times m_2 \times n_3} \) is the Lagrangian multiplier and \( \beta \) is a positive penalty parameter. In the end, ADMM iterates as follows:
\[
\begin{align*}
\mathcal{Y}^{k+1} &= \arg\min_{\mathcal{Y}} \| \mathcal{Y} \|_{\text{TNN}} + \frac{\beta}{2} \| B_1(X^k) - \mathcal{Y} + \frac{\Lambda^k}{\beta} \|_F^2, \\
X^{k+1} &= \arg\min_{X} L_\beta(X, \mathcal{Y}^{k+1}, \Lambda^k), \\
\Lambda^{k+1} &= \Lambda^k + \beta (B_1(X^{k+1}) - Y^{k+1}).
\end{align*}
\]

Below, we give the details of updating each minimizing subproblem.

**Step 1.** The \( \mathcal{Y} \)-subproblem at the \( k \)-th iteration is
\[
\mathcal{Y}^{k+1} = \arg\min_{\mathcal{Y}} \| \mathcal{Y} \|_{\text{TNN}} + \frac{\beta}{2} \| B_1(X^k) - \mathcal{Y} + \frac{\Lambda^k}{\beta} \|_F^2.
\]

A closed-form solution of (4) can be obtained by tensor singular value thresholding (t-SVT) operator \([56], [57]\), i.e.,
\[
\mathcal{Y}^{k+1} = \text{t-SVT}_\beta(B_1(X^k) + \frac{\Lambda^k}{\beta}),
\]
and its computational complexity is \( O(m_1^2 n_2 n_3 \log(n_3) + m_2^2 n_1 n_3 \min(n_1 n_2^2, n_2^2 n_1^2)) \).

**Step 2.** The \( X \)-subproblem at the \( k \)-th step is
\[
X^{k+1} = \arg\min_{X} I_\Omega(X) + \frac{\beta}{2} \| B_1(X) - \mathcal{Y}^{k+1} + \frac{\Lambda^k}{\beta} \|_F^2,
\]
\[
= \arg\min_{X} I_\Omega(X) + \frac{\beta}{2} \| \text{vec}(W X^\top) - (C^\top (C X^{k+1} - \frac{\Lambda^k}{\beta}) (4))^\top \|_F^2.
\]

To solve (6), we introduce the following theorem.

**Theorem 2:** Let \( E \) be a semi-orthogonal matrix, i.e., \( E^\top E = I \), where \( I \) is the identity matrix, then
\[
E^\top \mathcal{Y} = \arg\min_Z \| EZ - \mathcal{Y} \|_F^2.
\]

**Proof.** The derivation process is as follows
\[
\begin{align*}
& \arg\min_Z \| EZ - \mathcal{Y} \|_F^2 \\
& = \arg\min_Z \| EZ \|_F^2 - 2 < EZ, \mathcal{Y} > + \| \mathcal{Y} \|_F^2 \\
& = \arg\min_Z \| Z \|_F^2 - 2 < Z, E^\top \mathcal{Y} > + \| E^\top \mathcal{Y} \|_F^2 \\
& = \arg\min_Z \| Z - E^\top \mathcal{Y} \|_F^2.
\end{align*}
\]

Thus, \( E^\top \mathcal{Y} = \arg\min_Z \| Z - E^\top \mathcal{Y} \|_2 \).

By using **Theorem 2**, the corresponding (6) equals to
\[
X^{k+1} = \arg\min_{X} I_\Omega(X) + \frac{\beta}{2} \| X(3) - (C^\top (X^{k+1} - \frac{\Lambda^k}{\beta}) (4))^\top \|_F^2.
\]

**Algorithm 2** Low-Rank Tensor Completion Under Coupled Transform

**Input:** The observed data \( O \in \mathbb{R}^{n_1 \times n_2 \times n_3} \), the index set \( \Omega \), and the parameter \( \beta \);

**Initialization:** \( X^{(0)} = O; \mathcal{Y}^{(0)} = \Lambda^{(0)} = \text{zeros}(m_1^2 n_1 \times m_2^2 n_2 \times n_3); \ W \) and \( k = 0 \).

1. while \( \| X^{k+1} - X^k \|_F > 10^{-3} \) and \( k \leq 200 \) do
2. Update \( \mathcal{Y}^{(k+1)} \) by solving (5);
3. Update \( X^{(k+1)} \) by solving (7);
4. Update \( \Lambda^{(k+1)} \) by solving (8);
5. end while

**Output:** The completed tensor \( X \).

By minimizing the \( \mathcal{X} \)-subproblem, we have \( I_\Omega(X) = 0 \). Thus, the solution of \( \mathcal{X} \)-subproblem can be obtained by
\[
\begin{align*}
\mathcal{P}_\Omega(X^{k+1}) &= \mathcal{P}_\Omega(O), \\
\mathcal{P}_\Omega^c(X^{k+1}) &= \mathcal{P}_\Omega^c(\text{fold}_3((C^\top (C X^{k} - \frac{\Lambda^k}{\beta}) (4))^\top W)),
\end{align*}
\]
where \( \Omega^c \) denotes the complement of set \( \Omega \), and its computational complexity is \( O(m_1^2 n_1 n_3) \).

**Step 3.** updating the multiplier \( \Lambda \) by
\[
\Lambda^{k+1} = \Lambda^k + \beta (B_1(X^{k+1}) - Y^{k+1}).
\]

The overall ADMM iteration strategy is summarized in **Algorithm 2**. The total computational complexity at each iteration is \( O(m_1^2 n_1 n_3 \log(n_3) + m_2^2 n_1 n_3 \min(n_1 n_2^2, n_2^2 n_1^2)) \).

**Theorem 3:** The sequences generated from the proposed ADMM algorithm converges to the minimizer of the convex optimization problem (1).

**Proof.** Since \( B_1 \) is a linear operator, \( B_1(X) \) can be written as a matrix-vector product, i.e., \( B_1 X \), where \( X \) denotes the vectorization of \( X \). The linear constraints can be re-formulated as the following matrix-vector product:
\[
y + (-B_1) x = 0,
\]
where \( I \) and \( y \) denote the identity matrix and the vectorization of \( \mathcal{Y} \), respectively.

We separate all the variables into two groups, \( \mathcal{X} \) and \( \mathcal{Y} \), decompose the objective function as \( f + g \) with \( f = I_\Omega(X) \) and \( g = \| \mathcal{Y} \|_{\text{TNN}} \), both of which are convex functions. Therefore, the problem (2) fits the framework of ADMM [48], and the convergence of the algorithm is theoretically guaranteed, i.e., the sequence \( \{X^k, Y^k\} \) generated from the proposed ADMM algorithm is convergent.

**IV. EXPERIMENTAL RESULTS**

In this section, we test the performance of the proposed CT-LRTC and compare it with state-of-the-art tensor completion methods. Subsection IV-A provides some experimental settings. Subsection IV-B tests the proposed CT-LRTC in synthetic data completion. Subsections IV-C-IV-E present experimental results on multi-dimensional visual data, including color images, multispectral images, grayscale videos, and color videos. Finally, some discussions are given in Subsection IV-F.
A. Experimental Setting

1) Parameter Setting: The proposed method involves the following parameters: m and l in the framelet transform matrix W, and the positive penalty parameter \( \beta \) in the augmented Lagrangian function (3). The parameter \( m \) controls the type of the selected filter, including the Haar wavelet, piece-wise linear B-spline, and piece-wise cubic B-spline, which are briefly referred to as “Haar”, “Linear”, and “Cubic”, severally. The parameter \( l \) controls the level of framelet transform, and we set it to the range of \([1, 3]\) with increments 1. The effects of \( m \) and \( l \) are discussed in subsection IV-F. The parameter \( \beta \) controls the speed of convergence, and we fine-tuned it in the range of \([0.01, 0.1]\) with increments 0.01 to obtain the highest peak signal-to-noise ratio (PSNR) [58] value.

2) Compared Methods: We compare the proposed method with several existing approaches, including TNN [33], TNN-DCT [41], t-TNN [59], PSTNN [37], and TRLRF [60]. For a fair comparison, parameters in the compared method are manually adjusted according to the authors’ suggestions to obtain the highest PSNR value. Table II compares the computational complexity of different methods.

3) Data Generation and Experimental Environment: In order to test the generalization, we consider four types of multi-dimensional visual data: color images, multispectral images, grayscale videos, and color videos. All data are normalized to \([0, 1]\). We also consider two types of sampling: the element-wise sampling and the tube-wise sampling as shown in Fig. 3. In the element-wise sampling, we generate incomplete data by sampling elements of data randomly, with different sampling rates (SRs) \( \in \{5\%, \ 10\%, \ 20\%\} \). As for tube-wise sampling, we generate incomplete data by deleting elements of the original data with three different masks (text, graffiti, and grid mask). All tests are implemented under Windows 10 and MATLAB R2018a running on a desktop with an Intel(R) Core (TM) i9-9900K CPU at 3.60GHz and 32GB RAM.

4) Evaluation Indices: We consider the following indices for quantitative evaluation: PSNR and the structural similarity index [58] (SSIM). PSNR and SSIM are defined as

\[
\text{PSNR} = 20 \log_{10} \sqrt{\frac{n_1 \times n_2}{\|X - \hat{X}\|_F}},
\]

and

\[
\text{SSIM} = \frac{(2\mu_X \cdot \mu_{X*} + c_1)(2\sigma_{XX} + c_2)}{(\mu_X^2 + \mu_{X*}^2 + c_1)(\sigma_X^2 + \sigma_{X*}^2 + c_2)}.
\]

Here, \( X^* \in \mathbb{R}^{n_1 \times n_2} \) is the recovered image, \( X \) is the ground truth image, \( \mu_X \) and \( \mu_{X*} \), \( \sigma_X \) and \( \sigma_{X*} \) are the mean values and standard variances of \( X \) and \( X^* \), respectively, \( \sigma_{XX} \) is the covariance of \( X \) and \( X^* \), and \( c_1, c_2 > 0 \) are constants. For a third-order tensor, the PSNR and SSIM values are calculated by averaging the PSNR and SSIM values of all frontal slices.

B. Synthetic Data Completion

To verify our motivation, we conducted the experiments with respect to the tensor multi-rank and the sampling rate (SR) by the TNN-based tensor completion method and the proposed CT-LRTC. We conduct 50 independent experiments on random color image patch of size \( 64 \times 64 \times 3 \) and calculate the success rate, where one test is successful if the relative square error of the recovered tensor \( \hat{X} \) and the ground-truth tensor \( X \), i.e., \( \frac{\|X - \hat{X}\|_F^2}{\|X\|_F^2} \) is less than \( 10^{-3} \). Fig. 4 shows the success rates for various multi-ranks and different SRs. We can observe that the performance of the proposed CT-LRTC shows larger reddish-brown areas than the TNN-based method, implying higher success rates.
C. Color Image Completion

Ten benchmark color images\(^1\) are shown in Fig. 5 for color image completion. We conduct two types of experiments: the first five benchmark images of size \(512 \times 512 \times 3\) with element-wise sampling, and the last five benchmark images of size \(500 \times 500 \times 3\) with tube-wise sampling (i.e., color image inpainting). Here, we test three different masks: text, graffiti, and grid mask.

1) The Element-Wise Sampling: Fig. 6 shows the recovered results by different methods on color images under SR = 20%. As one can see, the recovered results obtained by TNN only recover the coarse structure, which produce evident blur and artifacts. We can observe that other TNN-based methods (i.e., TNN-DCT, t-TNN, and PSTNN) also suffer from such problems. The recovered results obtained by TRLRF recover some details and do not alleviate the blurriness compared to TNN-based methods. The underlying reason is that only the global low-rankness prior is not sufficient to recover the potential image. In comparison, the proposed CT-LRTC provides the most visually pleasing results with clear and sharp spatial details, due to CT-LRTC can capture the intrinsic structure of the multi-dimensional visual data represented in a multi-scale manner.

Table III presents the PSNR/SSIM values and average CPU time (in minutes) of the recovered results by different methods on color images under different SRs. We can observe that the proposed CT-LRTC consistently outperforms the compared ones in terms of both PSNR and SSIM values and the CPU time is comparable with the other compared methods. More precisely, CT-LRTC outperforms the second-best methods about 3 dB in PSNR and 0.1 in SSIM. Such an improvement is mainly attributed to that the use of multi-scale representation by CT-LRTC to implicitly utilize the spatial information of images, thereby promoting the improvement of PSNR and SSIM values.

2) The Tube-Wise Sampling: Fig. 7 shows the recovered results by different methods for color image inpainting. Notice that the magnified map of a patch and the corresponding error map (difference from the ground truth) are displayed below each image. The corresponding PSNR and SSIM values are displayed above the image. Clearly, CT-LRTC obtains the most visually satisfying results among the competing methods, and the PSNR and SSIM results are higher than the corresponding compared methods in all cases. In particular, the advantages of the proposed CT-LRTC are particularly evident in the text and mesh masking problem.

D. Multispectral Image Completion

In this subsection, we perform LRTC on five multispectral images (MSIs) from the CAVE dataset,\(^2\) i.e., Clay, Balloons, Feathers, Cd, and Beads. All MSIs are of size \(512 \times 512 \times 31\) with SR = 5%, 10%, and 20%.

\(^1\)Available at http://sipi.usc.edu/database/database.php

\(^2\)Available at http://www.cs.columbia.edu/CAVE/databases/multispectral/
Fig. 7. The recovered results by different methods on the five masked images. The corresponding PSNR and SSIM values are displayed above each image. For better visual comparison, under each image, we display the magnified map of a patch and the corresponding error map (difference from the ground truth). Error maps with less color information indicate better restoration performance.

Fig. 8 shows the pseudo-color image (composed of the 1st, 2nd, and 31st bands) of the recovered results by different methods on MSIs under SR = 10%, where the pixel intensity is readjusted for better viewing. We observe that TNN, TNN-DCT, PSTNN, and TRLRF cannot recover all the elements; t-TNN performs slightly better while the proposed CT-LRTC fills almost all elements. Regarding visual quality, the proposed CT-LRTC produces the closest result to the ground truth, which is superior to other compared methods.

Table IV lists the quantitative indexes (PSNR/SSIM) and average CPU time of the recovered results by different methods on MSIs under different SRs. The proposed CT-LRTC obtains the highest PSNR and SSIM values in almost all cases under different MRs. In addition, Fig. 9 displays the PSNR values of each frontal slice of multispectral image Clay. As observed, in all frontal slices, the PSNR values of the proposed CT-LRTC are much higher than those of the other compared methods.
Fig. 8. The pseudo-color image (composed of the 1st, 2nd, and 31st bands) of the recovered results by different methods on MSIs under SR = 10%. From top to bottom: Balloons, Feathers, and Cd, respectively. The intensity of the pixels in the shown image is readjusted for better display.

Fig. 9. The PSNR values of all frontal slices obtained by different methods on the multispectral image Clay and the grayscale video Suzie under different SRs.

E. Video Completion

In this subsection, we test four grayscale videos with different SRs: Birdhouse of size 700 × 860 × 30, Suzie of size 480 × 700 × 30, Bus of size 288 × 352 × 30, and Hall of size 144 × 176 × 100. And we also test three color videos with different SRs: Bus of size 288 × 352 × 3 × 30, Container of size 144 × 176 × 3 × 30, and Coastguard of size 144 × 176 × 3 × 30. These videos are available online http://trace.eas.asu.edu/yuv/.
Fig. 10. The 15-th frame of the recovered results by different methods on grayscale videos under SR = 10%. From top to bottom: Birdhouse, Suzie, Bus, and Hall, respectively.

Fig. 11. The 15-th frame (color figure) of the recovered results by different methods on color videos under SR = 5%. From top to bottom: Bus, Container, and Coastguard, respectively. For better viewing, under each image, we display the magnified map of a patch and the corresponding error map (difference from the ground truth). Error maps with less color information indicate better restoration performance.

1) Grayscale Video Completion: Fig. 10 shows the 15-th frame of the recovered examples by different methods on grayscale videos under SR = 10%, and the corresponding PSNR, SSIM values, and average CPU time are summarized in Table V. Clearly, CT-LRTC obtains the most visually pleasant results among the compared methods. In the red regions of Fig. 10, we can see that the recovered results by the proposed CT-LRTC are more clear and sharper in spatial texture compared with the other methods. Table V shows that the proposed CT-LRTC obtains the highest quantitative indexes in all testing videos. Fig. 9 displays the PSNR values of each frontal slice of Suzie, and the quantitative indexes are consistent with the above results.

2) Color Video Completion: Compared with the other TNN-based methods, the proposed CT-LRTC can also be used for color videos completion by simply changing $B_2$ into a two-dimensional Fourier transform. In order to facilitate comparison, for the other TNN-based methods, we use a reshaping operation to convert the color video (fourth-order) into third-order data, and then input it into the model for solution.
Table III

<table>
<thead>
<tr>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barbars</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNN</td>
<td>16.22</td>
<td>0.234</td>
<td>21.55</td>
<td>0.47</td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>16.16</td>
<td>0.219</td>
<td>19.34</td>
<td>0.36</td>
</tr>
<tr>
<td>t-TNN</td>
<td>17.19</td>
<td>0.266</td>
<td>19.39</td>
<td>0.27</td>
</tr>
<tr>
<td>PSTNN</td>
<td>16.97</td>
<td>0.256</td>
<td>21.65</td>
<td>1.88</td>
</tr>
<tr>
<td>TRFLF</td>
<td>17.10</td>
<td>0.244</td>
<td>19.69</td>
<td>6.58</td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>22.81</td>
<td>0.573</td>
<td>24.96</td>
<td>4.30</td>
</tr>
</tbody>
</table>

Table IV

<table>
<thead>
<tr>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clay</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNN</td>
<td>39.20</td>
<td>0.951</td>
<td>43.53</td>
<td>0.953</td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>40.36</td>
<td>0.970</td>
<td>44.29</td>
<td>0.985</td>
</tr>
<tr>
<td>t-TNN</td>
<td>37.80</td>
<td>0.893</td>
<td>42.58</td>
<td>0.978</td>
</tr>
<tr>
<td>PSTNN</td>
<td>40.19</td>
<td>0.965</td>
<td>43.97</td>
<td>0.982</td>
</tr>
<tr>
<td>TRFLF</td>
<td>41.86</td>
<td>0.962</td>
<td>43.75</td>
<td>0.976</td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>44.71</td>
<td>0.988</td>
<td>48.16</td>
<td>0.993</td>
</tr>
</tbody>
</table>

Table V

<table>
<thead>
<tr>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balloons</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNN</td>
<td>34.99</td>
<td>0.924</td>
<td>39.61</td>
<td>0.967</td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>36.05</td>
<td>0.939</td>
<td>40.87</td>
<td>0.974</td>
</tr>
<tr>
<td>t-TNN</td>
<td>38.37</td>
<td>0.966</td>
<td>43.21</td>
<td>0.980</td>
</tr>
<tr>
<td>PSTNN</td>
<td>36.16</td>
<td>0.929</td>
<td>40.94</td>
<td>0.960</td>
</tr>
<tr>
<td>TRFLF</td>
<td>36.43</td>
<td>0.932</td>
<td>40.13</td>
<td>0.963</td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>41.47</td>
<td>0.985</td>
<td>45.07</td>
<td>0.992</td>
</tr>
</tbody>
</table>

Table VI

<table>
<thead>
<tr>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feathers</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNN</td>
<td>29.28</td>
<td>0.812</td>
<td>33.10</td>
<td>0.897</td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>30.33</td>
<td>0.850</td>
<td>34.51</td>
<td>0.926</td>
</tr>
<tr>
<td>t-TNN</td>
<td>30.94</td>
<td>0.855</td>
<td>35.86</td>
<td>0.942</td>
</tr>
<tr>
<td>PSTNN</td>
<td>29.98</td>
<td>0.808</td>
<td>34.66</td>
<td>0.903</td>
</tr>
<tr>
<td>TRFLF</td>
<td>31.33</td>
<td>0.826</td>
<td>33.70</td>
<td>0.895</td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>33.77</td>
<td>0.938</td>
<td>37.14</td>
<td>0.966</td>
</tr>
</tbody>
</table>

Table VII

<table>
<thead>
<tr>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cd</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TNN</td>
<td>28.39</td>
<td>0.839</td>
<td>32.22</td>
<td>0.911</td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>29.19</td>
<td>0.881</td>
<td>32.65</td>
<td>0.924</td>
</tr>
<tr>
<td>t-TNN</td>
<td>31.58</td>
<td>0.899</td>
<td>34.50</td>
<td>0.938</td>
</tr>
<tr>
<td>PSTNN</td>
<td>29.72</td>
<td>0.847</td>
<td>32.87</td>
<td>0.911</td>
</tr>
<tr>
<td>TRFLF</td>
<td>30.66</td>
<td>0.870</td>
<td>33.96</td>
<td>0.902</td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>35.56</td>
<td>0.954</td>
<td>37.90</td>
<td>0.970</td>
</tr>
</tbody>
</table>

Fig. 11 shows the 15-th frame of the recovered examples by different methods on color videos under SR = 10%, and the corresponding PSNR, SSIM values, and average CPU time are summarized in Table VI. It is obvious that CT-LRTC visually outperforms the other compared methods in keeping details of the recovered images and achieves the highest quantitative results under different SRs.

Fig. 12. The relative error curves of the ADMM algorithm on multispectral image Clay with SR = 10% (left) and SR = 20% (right), respectively.

F. Discussion

In this subsection, we give some discussions.

1) A Rigorous Comparison: In order to further verify the effectiveness of the proposed method, we add a rigorous comparison with the compared methods about missing percentage from 10% to 90% with an interval of 20%. Table VII lists the quantitative indexes (PSNR/SSIM) by different methods on the color image Flower, the multispectral image Cd, and the grayscale video Hall under different SRs. We can observe that the proposed CT-LRTC obtains visually satisfying results and consistently outperforms the other compared methods in terms of both PSNR and SSIM values, especially when the sampling is extremely low. These observations correspond exactly to the relationships we mentioned earlier.

2) Framelet Setting: We discuss the effects of two important parameters of the framelet transform, which are the parameter $m$ that controls the selectio n of filter and the parameter $l$ that controls the decomposition level. Table VIII shows the quantitative indexes of the proposed CT-LRTC with different framelet settings on the color image peppers under SR = 5%. From Table VIII, we can see that CT-LRTC is not very sensitive to changes in the framelet transform parameters, but they directly affect the computational complexity of the proposed algorithm.

3) Numerical Convergence of the Algorithm: In Fig. 12, we illustrate the convergence behavior of the proposed ADMM solver. Here, the relative error in each iteration is defined as $\|X^{k+1} - X^k\|_F / \|X^k\|_F$, where $X^{k+1}$ and $X^k$ are the two successive reconstructed tensors. Fig. 12 suggests a strong convergence behavior of the proposed ADMM solver; it can reach a relative decrease of $10^{-2}$ in 30 iterations, and the relative error is decreased with the increase of iterations so that higher accuracy can be obtained.
Fig. 13. The recovered results by different methods on color image Airplane of size 512 \times 512 \times 3 under SR = 20\%. (a) The ground truth; (b) The observed image; (c-f) The recovered results by CT-LRTC without framelet and Fourier transforms, CT-LRTC without Fourier transform, CT-LRTC without framelet transform, and the proposed CT-LRTC, respectively. The corresponding PSNR and SSIM values are displayed above the image.

Fig. 14. The recovered results by different methods for diverse sampling and data. From top to bottom: the color images Sculpture and Beach from the testing dataset with tube-wise sampling; the color images Cherry and Sailboat not from the testing dataset with tube-wise sampling; and the color images Peppers and Fruits not from the testing dataset with element-wise sampling. For better visualization, under each image, we show enlargements of a demarcated patch and the corresponding error map (difference from the ground-truth). Error maps with less color information indicate better restoration performance.

4) Effects of the Three Transforms: Numerically, to further clarify that the inner-correlation between these three transforms are mutually collaborative, we have added experiments to test the performance of different combinations of these three transforms in Fig. 13. From Fig. 13, we can observe that the recovered results with KL and Framelet + KL are color distorted, which implies the necessity of Fourier transform in the spectral (i.e., color) fidelity. The recovered results with KL and FFT+KL lose some image details and textures, especially in the tail, which implies the necessity of Framelet transform in the spatial fidelity. Although the coupled transform may adversely affect real-time performance, the proposed CT-LRTC has the best visual performance in the spatial and spectral fidelity, which validates our motivation.

5) Compared With Deep Learning: To further illustrate that the proposed CT-LRTC has good generalization compared with some DL-based completion methods, some experiments are given. Here, we choose “Generative Image Inpainting with Contextual Attention” (termed as CA) [46] as the compared method, whose main idea is to utilize the surrounding image features as references during network training to make better predictions. The model is a feed-forward, fully convolutional neural network which is trained on the color image dataset with tube-wise sampling. We consider two color images from the testing dataset with tube-wise sampling, two color images not from the testing dataset with tube-wise sampling, and two color images not from the testing dataset with element-wise sampling.
TABLE V
THE PSNR/SSIM VALUES AND AVERAGE CPU TIME (IN MINUTES) OF THE RECOVERED RESULTS BY DIFFERENT METHODS ON GRAYSCALE VIDEOS UNDER DIFFERENT SRS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDER FONTS

<table>
<thead>
<tr>
<th>Video</th>
<th>SR</th>
<th>Method</th>
<th>5% PSNR</th>
<th>5% SSIM</th>
<th>10% PSNR</th>
<th>10% SSIM</th>
<th>20% PSNR</th>
<th>20% SSIM</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Birdhouse</td>
<td>TNN</td>
<td>23.74</td>
<td>0.467</td>
<td>24.79</td>
<td>0.526</td>
<td>26.97</td>
<td>0.625</td>
<td>7.86</td>
<td></td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>23.25</td>
<td>0.475</td>
<td>24.85</td>
<td>0.530</td>
<td>27.02</td>
<td>0.629</td>
<td>5.22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t-TNN</td>
<td>24.53</td>
<td>0.563</td>
<td>26.08</td>
<td>0.635</td>
<td>28.09</td>
<td>0.733</td>
<td>5.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSTNN</td>
<td>23.31</td>
<td>0.413</td>
<td>25.07</td>
<td>0.523</td>
<td>27.17</td>
<td>0.630</td>
<td>58.01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>23.43</td>
<td>0.492</td>
<td>24.74</td>
<td>0.594</td>
<td>25.70</td>
<td>0.659</td>
<td>64.72</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>26.85</td>
<td>0.673</td>
<td>27.93</td>
<td>0.703</td>
<td>29.78</td>
<td>0.770</td>
<td>58.77</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE VI
THE PSNR/SSIM VALUES AND AVERAGE CPU TIME (IN MINUTES) OF THE RECOVERED RESULTS BY DIFFERENT METHODS ON COLOR VIDEOS UNDER DIFFERENT SRS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDER FONTS

<table>
<thead>
<tr>
<th>Video</th>
<th>SR</th>
<th>Method</th>
<th>5% PSNR</th>
<th>5% SSIM</th>
<th>10% PSNR</th>
<th>10% SSIM</th>
<th>20% PSNR</th>
<th>20% SSIM</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Suzie</td>
<td>TNN</td>
<td>28.94</td>
<td>0.728</td>
<td>30.80</td>
<td>0.775</td>
<td>33.28</td>
<td>0.841</td>
<td>5.41</td>
<td></td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>29.44</td>
<td>0.741</td>
<td>31.28</td>
<td>0.788</td>
<td>33.57</td>
<td>0.848</td>
<td>4.20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t-TNN</td>
<td>30.35</td>
<td>0.771</td>
<td>32.26</td>
<td>0.823</td>
<td>34.58</td>
<td>0.880</td>
<td>4.14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSTNN</td>
<td>29.40</td>
<td>0.725</td>
<td>31.10</td>
<td>0.779</td>
<td>33.30</td>
<td>0.841</td>
<td>20.01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>29.40</td>
<td>0.742</td>
<td>31.07</td>
<td>0.789</td>
<td>32.14</td>
<td>0.830</td>
<td>51.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>32.14</td>
<td>0.826</td>
<td>33.54</td>
<td>0.855</td>
<td>35.60</td>
<td>0.896</td>
<td>38.72</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE VII
THE PSNR/SSIM VALUES OF THE RECOVERED RESULTS BY DIFFERENT METHODS UNDER DIFFERENT SRS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDER FONTS

<table>
<thead>
<tr>
<th>Video</th>
<th>SR</th>
<th>Method</th>
<th>5% PSNR</th>
<th>5% SSIM</th>
<th>10% PSNR</th>
<th>10% SSIM</th>
<th>20% PSNR</th>
<th>20% SSIM</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bus</td>
<td>TNN</td>
<td>19.70</td>
<td>0.383</td>
<td>21.23</td>
<td>0.486</td>
<td>23.21</td>
<td>0.635</td>
<td>2.25</td>
<td></td>
</tr>
<tr>
<td>TNN-DCT</td>
<td>19.84</td>
<td>0.388</td>
<td>21.14</td>
<td>0.485</td>
<td>23.07</td>
<td>0.622</td>
<td>0.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>t-TNN</td>
<td>20.87</td>
<td>0.488</td>
<td>22.45</td>
<td>0.569</td>
<td>24.85</td>
<td>0.748</td>
<td>1.14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSTNN</td>
<td>19.81</td>
<td>0.387</td>
<td>21.86</td>
<td>0.549</td>
<td>25.44</td>
<td>0.813</td>
<td>8.21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRLRF</td>
<td>19.47</td>
<td>0.396</td>
<td>20.50</td>
<td>0.489</td>
<td>22.51</td>
<td>0.612</td>
<td>11.18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CT-LRTC</td>
<td>21.84</td>
<td>0.557</td>
<td>23.06</td>
<td>0.636</td>
<td>25.10</td>
<td>0.752</td>
<td>7.03</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE VIII
THE PSNR/SSIM/TIME (IN SECONDS) VALUES OF THE PROPOSED CT-LRTC ON THE COLOR IMAGE Peppers UNDER DIFFERENT FRAMELET SETTINGS WITH SR = 5%. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDER FONTS

<table>
<thead>
<tr>
<th>Filters</th>
<th>Index</th>
<th>Level = 1</th>
<th>Level = 2</th>
<th>Level = 3</th>
<th>Level = 4</th>
<th>Level = 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Haar</td>
<td>PSNR</td>
<td>23.40</td>
<td>0.507</td>
<td>0.714</td>
<td>0.742</td>
<td>0.756</td>
</tr>
<tr>
<td>Time(s)</td>
<td>146</td>
<td>189</td>
<td>231</td>
<td>288</td>
<td>357</td>
<td>357</td>
</tr>
<tr>
<td>Linear</td>
<td>PSNR</td>
<td>21.89</td>
<td>0.623</td>
<td>0.748</td>
<td>0.748</td>
<td>0.756</td>
</tr>
<tr>
<td>Time(s)</td>
<td>245</td>
<td>313</td>
<td>481</td>
<td>635</td>
<td>804</td>
<td>804</td>
</tr>
<tr>
<td>Cubic</td>
<td>PSNR</td>
<td>22.89</td>
<td>0.725</td>
<td>0.776</td>
<td>0.776</td>
<td>0.776</td>
</tr>
<tr>
<td>Time(s)</td>
<td>625</td>
<td>823</td>
<td>932</td>
<td>1024</td>
<td>1206</td>
<td>1206</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper, we proposed a novel low-rank tensor representation under the coupled transforms, which can characterize both the global correlation and local geometric details in a unified framework, and obtain a better low multi-rank approximation. The proposed low-rank tensor representation can be formulated via the two-dimensional framelet transform, Fourier transform, and Karhunen–Loève transform (via singular value decomposition). Further, we formulated a novel model for LRTC (named as CT-LRTC) by promoting the proposed low-rank tensor representation. Then, we developed an efficient ADMM algorithm to optimize the proposed CT-LRTC, in which every variable has a closed-form solution in each iteration. Finally, experimental examples of real-world imaging data illustrated that the proposed CT-LRTC outperforms many existing approaches in both qualitative and quantitative aspects. In future work, we will work hard to speed up CT-LRTC, establish the theoretical justification, and expand the low-rank tensor representation under coupled transforms to other applications, such as denoising [61], [62] and subspace clustering [63], [64].
Authorized licensed use limited to: University of Electronic Science and Tech of China. Downloaded on March 31,2021 at 01:57:43 UTC from IEEE Xplore. Restrictions apply.