A DCT-based Tensor Completion Approach for Recovering Color Images and Videos from Highly Undersampled Data

Chenjian Pan$^1$ and Chen Ling$^1$ and Hongjin He$^2$ and Liquan Qi$^{3,4}$ and Yanwei Xu$^3$

$^1$Department of Mathematics, School of Science, Hangzhou Dianzi University, Hangzhou, 310018, China.
$^2$ School of Mathematics and Statistics, Ningbo University, Ningbo 315211, China.
$^3$Future Network Theory Lab, 2012 Labs Huawei Technologies Investment Co., Ltd, Shatin, New Territory, Hong Kong, China;
$^4$Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China

Recovering color images and videos from highly undersampled data is a fundamental and challenging task in face recognition and computer vision. By the multi-dimensional nature of color images and videos, in this paper, we propose a novel tensor completion approach, which is able to efficiently explore the sparsity of tensor data under the discrete cosine transform (DCT). Specifically, we introduce two DCT-based tensor completion models as well as two implementable algorithms for their solutions. The first one is a DCT-based weighted nuclear norm minimization model. The second one is called DCT-based $p$-shrinkage tensor completion model, which is a nonconvex model utilizing $p$-shrinkage mapping for promoting the low-rankness of data. Moreover, we accordingly propose two implementable augmented Lagrangian-based algorithms for solving the underlying optimization models. A series of numerical experiments including color image inpainting and video data recovery demonstrate that our proposed approach performs better than many existing state-of-the-art tensor completion methods, especially for the case when the ratio of missing data is high.

Index Terms—Tensor completion, $p$-shrinkage thresholding, weighted nuclear norm, discrete cosine transform, image inpainting.

I. INTRODUCTION

WITH the rapid developments of sensor technologies and video surveillance systems, the collected images and videos are naturally stored as multi-way arrays, which are also called tensors. As we know, the complete information of images and videos is very important to make a good decision for intelligent devices or data users. However, during the acquisition process, observed image and video data often contain missing entries (or pixels). In this situation, estimating these missing information of images and videos, which is also called tensor completion, is fundamental and challenging problem in the communities of image processing and computer vision, e.g., see [1]–[4] and references therein.

It is well-known that tensor is a higher-order extension of matrix, so the tensor completion is also a natural generalization of matrix completion. Generally speaking, when the missing entries are sparse in an incomplete matrix, we could efficiently explore the local information around the unknown components to get a relatively ideal completion to the matrix, e.g., see [5]–[7]. However, when dealing with a highly undersampled matrix, it is incredibly difficult, even if not impossible, to accurately estimate these missing entries from an incomplete matrix without any global or prior information. Comparatively, the complex structure of tensors makes that recovering the missing entries from a highly undersampled tensor puts forward more both theoretical and computational challenges at the interface of statistics and optimization, e.g., see [8]. In the past decades, it is well-documented that exploiting the inherent global information, e.g., low rank and sparsity, of the data is able to greatly improve the estimation quality of highly undersampled matrices and tensors, e.g., see [9]–[13], to name just a few.

As a direct extension of the low-rank matrix completion, the canonical low-rank tensor completion (LRTC) model is expressed mathematically as

$$
\min_{\mathcal{X}} \text{rank}(\mathcal{X}) \\
\text{s.t. } \mathcal{P}_{\Omega}(\mathcal{X}) = \mathcal{P}_{\Omega}(\mathcal{H}),
$$

where both $\mathcal{X}$ and $\mathcal{H}$ are $N$-th order tensors, $\text{rank}(\cdot)$ denotes the rank function, $\Omega$ is the index set corresponding to the observed entries of the incomplete tensor $\mathcal{H}$, and $\mathcal{P}_{\Omega}(\cdot)$ is the linear operator that keeps known elements in $\Omega$ while sets the others to be zeros. Indeed, directly minimizing the rank function of tensors, even for matrices, is an NP-hard problem (see [14], [15]). Moreover, unlike the unique definition of rank for matrices, there are diverse definitions for tensors such as CANDECOMP/PARAFAC (CP) rank, Tucker rank and Tensor Train (TT) rank (see [16], [17]). Consequently, the diversity of tensor ranks often makes engineers difficult to choose an appropriate candidate for characterizing the low-rankness of their problems.

Roughly speaking, most state-of-the-art tensor completion methods can be grouped into two categories. (i) The first type refers to the tensor nuclear norm minimization approach, which unfolds the underlying tensor as a series of matrices so that we could efficiently employ the matrix nuclear norm as an approximation to the rank of a tensor. Therefore, many efficient algorithms tailored for matrix completion could be naturally extended to tensor completion based on the so-called tensor nuclear norm definitions, e.g., see [9], [11], [18]–[23] and references therein. Mathematically, a seminal tensor nuclear norm minimization model introduced in [9] takes the
From the mathematical modeling perspective, the performance can be found in recent works, e.g., [22], [33], [36]–[40]. From the mathematical modeling perspective, the underlying images are mostly sparse after taking the TV operation. Consequently, we see that an $\ell_1$-norm is used to promote the sparsity of an image in the TV transform domain, which demonstrate that combining the low-rank and sparse prior information simultaneously can greatly improve the restoration quality. In the image processing literature, the well-known discrete cosine transform (DCT) has been widely used for the most popular image compression standard JPEG, since DCT is powerful to obtain a (at least approximately) sparse representation of an image. For example, we consider a widely used gray video, i.e., Suzie (see Fig. 1), which is a third order tensor with size $147 \times 176 \times 100$. By performing DCT on each dimension of such a gray video, a large number of elements of approach to zeros, thereby leading to a high level sparsity when setting a truncated number (TN), e.g., 0.05 and 0.1, to drop all values less than such a TN. However, the video recovered directly from the truncated tensor is still close to the original one, which means that a small sample information is possible to recover an ideal (at least identifiable) one. Such an experimental observation also encourages us to consider a sparsity regularization on the coefficient tensor after the DCT procedure. Here, we refer the reader to [41]–[47] for recent applications of DCT on image recovery. However, compared to the TV-based tensor completion model, to the best of our knowledge, DCT-based approaches received much less considerable attention on color images and videos inpainting from tensor completion angle. Hence, we are motivated to make a further study on showing the ability of the DCT technique for recovering images and videos from highly undersampled data.

\[
\min_{X} \sum_{n=1}^{N} \alpha_n \|X(n)\|_1, \\
\text{s.t.} \quad \Omega_\Omega(X) = \Omega_\Omega(H),
\]

where $\alpha_n \geq 0 \ (n \in [N] := \{1, 2, \cdots, N\})$ can be regarded as weight parameters often satisfying $\sum_{n=1}^{N} \alpha_n \approx 1$, matrix $X(n)$ corresponds to the mode-$n$ unfolding of tensor $X$ for every $n \in [N]$, and $\| \cdot \|_*$ represents the well-known nuclear norm referring to the sum of all singular values of a matrix. Note that a vector consisting of the ranks of unfolding matrices $X(n)$’s ($n = 1, 2, \cdots, N$) is called Tucker rank of tensor $X$ (see [16]). Hence, the objective function in (2) is indeed a convexly weighted sum of the components of Tucker rank. Compared to the direct vectorization and matricization approaches for tensor data, model (2) efficiently exploits the multi-mode structure, thereby possibly achieving a better approximation to the low-rankness of tensor data. Moreover, these unfolding completion models (i.e., tensor nuclear norm minimization models) are beneficial for algorithmic design due to the convexity of tensor nuclear norm functions. Recently, the state-of-the-art alternating direction method of multipliers (ADMM) and Douglas-Rachford splitting method have been successfully applied to tensor completion, e.g., see [9], [10], [24]–[26].

The second type of tensor completion approach is based on tensor decomposition. Actually, we observe that many tensor nuclear norm based models utilizes the so-called unfolding (or matricization) technique. However, Yuan and Zhang [27] showed that the matricization for higher-order tensor ignores the nature of tensor and potentially destroys some inherent properties of the data, thereby leading to suboptimal procedure, which encourages researchers to develop new completion methods from tensor decomposition perspectives. In the literature, some popular tensor decomposition techniques include CP decomposition, Tucker decomposition [16], T-SVD [28] and Tensor-Train decomposition [17]. Like the matrix decomposition, the aforementioned tensor decomposition forms have also been verified as powerful dimensionality reduction tools, which can promote the low-rankness of tensor data to some extent. Empirically, these tensor decomposition based completion models have received great successes in big data analysis, pattern recognition, and traffic data recovery, e.g., see [12], [13], [29]–[34] and references therein.

When regarding color images and videos as general tensors, the aforementioned tensor completion approaches are certainly applicable to images and video inpainting problems. However, it often fails to achieve an ideal restoration quality for highly undersampled image data since only the low-rank property was exploited in general tensor completion models. As we know, the total variation (TV) regularization (see [35]) is a widely used tool in the community of image processing to preserve sharp discontinuities (edges) of an image, while removing noise and other unwanted fine scale detail. Hence, researchers judiciously incorporated TV regularization into low-rank tensor completion models for the purpose of exploiting the inherent structure of images. The promising numerical performance can be found in recent works, e.g., [22], [33], [36]–[40]. From the mathematical modeling perspective, the underlying images are mostly sparse after taking the TV

![Fig. 1](image1.png)

Fig. 1. The first row show the visualization of the sparsity of the gray video Suzie after DCT, where the white represents 0. The second row illustrates the comparisons of tenth frame of the original video Suzie with ones reconstructed from the truncated tensor.

In this paper, we introduce a DCT-based low-rank tensor completion approach for recovering color images and videos. Our contribution is four-fold.

- Most nuclear norm based tensor completion models equally treat all singular values of the unfolding matrices. For some real-world datasets, small singular values are often sensitive to the incomplete (or corrupted with noise) observed information, thereby possibly leading to a suboptimal low-rank tensor. In this situation, we shall propose a DCT-based weighted nuclear norm minimization
model, where the singular values have different weights to enhance the global low-rank structure.

- In many unfolding approaches, the most popular low-rank surrogate of tensors is the nuclear norm of unfolding matrices. Recently, both theoretical and empirical results show that the so-called $p$-shrinkage thresholding algorithm outperforms the classical iterative soft thresholding algorithm induced by nuclear norm for low-rank and sparse recovery problems, e.g., see [48]–[50]. Accordingly, the second contribution of this paper is to introduce a DCT-based $p$-shrinking tensor completion model, which is a nonconvex optimization problem due to the nonconvex penalty function associated to the $p$-shrinkage operator for promoting the low-rankness of the underlying tensor.

- We develop two implementable augmented Lagrangian-based algorithms for the underlying models, where all subproblems of our algorithms have closed-form solution.

- We conduct a series of experiments on color images and surveillance videos. Our computational results demonstrate that the proposed DCT-based approaches outperform many state-of-the-art tensor completion methods especially for highly undersampled cases.

The structure of this paper is as follows. In Section II, we summarize some notations and definitions including (weighted) nuclear norm of matrices, $p$-shrinkage mapping and proximal operators. In Section III, we will first introduce a unified DCT-based tensor completion approach. Then, we split this section into two parts and propose two kinds of DCT-based tensor completion models, respectively. Based on the augmented Lagrangian function, we also propose two implementable algorithms for the underlying tensor completion models. Numerically, in Section IV, we conduct the performance of our approaches on images and videos recovery from highly undersampled data. Finally, some concluding remarks are provided in Section V.

II. NOTATIONS AND PRELIMINARIES

In this section, we summarize some notations and definitions on $p$-shrinkage mapping and weighted nuclear norm that will be used throughout this paper.

Tensor is a multidimensional array, which is an extension of matrix. The space of all $N$-th order real tensors is denoted by $\mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, where the order of a tensor is also called way or mode. Given an $N$-th order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, we denote the $(i_1, i_2, \ldots, i_N)$-th component of $\mathcal{A}$ by $a_{i_1i_2\ldots i_N}$. So the $N$-th order tensor $\mathcal{A}$ is also denoted by $\mathcal{A} = (a_{i_1i_2\ldots i_N})$. Throughout this paper, tensors of order $N \geq 3$ are denoted by calligraphical letters, e.g., $\mathcal{A}, \mathcal{B}, \ldots$. Generally, we use capital letters (e.g., $A, B, \ldots$), boldfaced lowercase letters (e.g., $a, b, \ldots$), and lowercase letters (e.g., $a, b, \ldots$) to denote matrices, vectors, and scalars, respectively. For any two $N$-th order tensors $\mathcal{A} = (a_{i_1i_2\ldots i_N})$ and $\mathcal{B} = (b_{i_1i_2\ldots i_N})$, the inner product between $\mathcal{A}$ and $\mathcal{B}$ is given by

$$\langle \mathcal{A}, \mathcal{B} \rangle := \sum_{i_1,i_2,\ldots,i_N} a_{i_1i_2\ldots i_N} b_{i_1i_2\ldots i_N}.$$  

Consequently, the Frobenius norm of tensor $\mathcal{A}$ associated with the above inner product is given by $\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$.

Given an $N$-th order tensor $\mathcal{A}$, the mode-$n$ matricization (or unfolding) of $\mathcal{A}$ is denoted by $\mathcal{A}_{(n)}$, and the $(i_1, i_2, \ldots, i_N)$-th entry of tensor $\mathcal{A}$ is mapped to the $(i_n, j)$-th entry of matrix $A_{(n)}$ in the lexicographical order, where

$$j = 1 + \sum_{1 \leq l \leq N, l \neq n} (i_l - 1)J_l$$  

with $J_l = \prod_{1 \leq i \leq l-1, i \neq n} I_i$.

Given a matrix $A \in \mathbb{R}^{m \times n}$, we write its singular value decomposition (SVD) as $A = U \Lambda V^T$, where $U$ and $V$ are orthogonal matrices and $\Lambda$ is a diagonal matrix, i.e.,

$$\Lambda = \text{diag}(\sigma(A)) = \text{diag}(\sigma_1(A), \sigma_2(A), \cdots, \sigma_r(A))$$

with $r := \min\{m, n\}$, and the diagonals $\sigma_i(A)$’s are singular values of $A$ satisfying $\sigma_1(A) \geq \sigma_2(A) \geq \cdots \geq \sigma_r(A)$ without loss of generality. With the above preparation, we first recall the definition of (weighted) nuclear norm of a matrix (see [42]).

**Definition 2.1.** For any given matrix $A \in \mathbb{R}^{m \times n}$, its nuclear norm is defined by

$$\|A\|_* = \sum_{i=1}^{r} \sigma_i(A)$$

and the weighted nuclear norm is given by

$$\|A\|_{*,w} = \sum_{i=1}^{r} w_i \sigma_i(A)$$

where $r := \min\{m, n\}$ and $\sigma_i(A)$ is the $i$-th largest singular value of $A$, and $w = (w_1, \cdots, w_r)$ is a weight vector, which, as suggested in [19], is given by

$$w_i = \frac{\delta}{(\sigma_i(A) + \epsilon)}, \quad i = 1, 2, \ldots, r,$$

where $\delta \in \mathbb{R}^+$ is a constant and $\epsilon$ is a small value to avoid division by 0. In algorithmic implementation, we can iteratively update $w_i$’s.

Below, we recall the $p$-shrinkage mapping introduced in [49].

**Definition 2.2.** Given $a \in \mathbb{R}^n$, $\mu > 0$ and $p \leq 1$, the $p$-shrinkage thresholding operator is defined in component-wise by

$$\text{pshrink}(a, \mu, p) = \text{sign}(a) \odot \max \{ |a| - \mu |a|^{p-1}, 0 \}, \quad (3)$$

where ‘$\text{sign}(\cdot)$’ and ‘$\cdot \mid \cdot$’ are the sign function and absolute value function in component-wise, respectively, and ‘$\odot$’ represents the component-wise product between two vectors. In particular, when setting $p = 1$, the $p$-shrinkage operator (3) immediately reduces to the well-known soft-thresholding, which is denoted by

$$\text{shrink}(a, \mu) = \text{sign}(a) \odot \max \{ |a| - \mu, 0 \}. \quad (4)$$

Clearly, the smaller $p$ is, the less shrink($a, \mu, p$) shrinks large inputs. We refer the reader to [50] for an illustration to the philosophy of the $p$-shrinkage operator.

For any given $a \in \mathbb{R}^n$, the proximal operator associated
with \( \theta : \mathbb{R}^n \to \mathbb{R} \) about \( \tau > 0 \) is defined as

\[
\text{prox}_{\theta, \tau}(a) := \arg \min_{x \in \mathbb{R}^n} \theta(x) + \frac{\tau}{2} \|x - a\|^2.
\]

Consequently, as shown in [51, Theorem 4] (see also [52]), for any \( \mu > 0 \), \( A \in \mathbb{R}^{m \times n} \) and a weight vector \( w \), the global optimal solution to the following optimization problem

\[
\min_X \|X\|_* + \frac{1}{2\mu} \|X - A\|_F^2
\]
is given by the weighted SVD thresholding

\[
X^* = U \text{wshrink}(A, \mu, w)V^T,
\]
where \( A = UAV^T \) is the SVD of \( A \) and

\[
\text{wshrink}(A, \mu, w) = \max \{\sigma(A) - \mu w, 0\}. \tag{5}
\]

Moreover, it follows from [48] that the \( p \)-shrinkage mapping defined in Definition 2.2 is also interpreted as the proximal operator of a penalty function \( \Phi^p(\cdot) : \mathbb{R}^n \to \mathbb{R} \), i.e.,

\[
\text{pshrink}(a, \mu, p) = \arg \min_{x \in \mathbb{R}^n} \Phi^p(x) + \frac{1}{2\mu} \|x - a\|^2,
\]
where \( \Phi^p(x) := \sum_{i=1}^n \phi^p_i(x_i) \) with \( \phi^p_i(x_i) \) being even, concave, nondecreasing and continuous on \([0, \infty)\), differentiable on \((0, \infty)\), nondifferentiable at 0 with the subdifferential being \( \partial \phi^p_i(0) = [-1, 1] \). We refer the reader to [48, Theorem 1] (also [50], [53]) for more details. Notice that the \( p \)-shrinkage operator is available to matrix variables. Specifically, for the optimization problem

\[
\min_X \Phi^p(X) + \frac{1}{2\mu} \|X - A\|_F^2, \tag{6}
\]
we have the globally optimal solution is

\[
X^* = U \text{pshrink}(A, \mu, p)V^T, \tag{7}
\]
where \( A = UAV^T \) is the SVD of \( A \) and

\[
\text{pshrink}(A, \mu, p) = \text{pshrink}(\sigma(A), \mu, p).
\]

III. MODELS AND ALGORITHM

In this section, we aim to develop a DCT-based tensor completion approach to images and videos recovery. Mathematically, we first introduce a unified DCT-based low-rank tensor completion model for \( N \)-th order tensors as follows:

\[
\min_{\mathcal{X}} \text{rank}(\mathcal{X}) + \lambda \|\mathcal{D}(\mathcal{X})\|_0 \tag{8}
\]

s.t. \( \mathcal{P}_1(\mathcal{X}) = \mathcal{P}_2(\mathcal{H}) \),

where \( \mathcal{D}(\cdot) \) denotes the multi-dimensional DCT operator, \( \|\cdot\|_0 \) represents the so-called \( \ell_0 \)-norm to characterize the sparsity of coefficients of DCT, and \( \lambda > 0 \) is a tuning parameter. Due to the appearances of low-rank function and \( \ell_0 \)-norm, the optimization model (8) is a highly non-convex and NP-hard problem. Usually, we can find an approximate solution of (8) by employing the (non-) convex relaxations of rank function and \( \ell_0 \)-norm, respectively. Therefore, in this section, we first introduce a DCT-based low-rank tensor completion model equipped with a weighted nuclear norm regularization term. Then, we propose a DCT-based \( p \)-shrinking low-rank tensor completion model, where the objective function incorporates a structured nonconvex part to promote the low-rankness of data. Additionally, we will propose two implementable augmented Lagrangian-based splitting methods to solve the underlying models.

A. A DCT-based weighted nuclear norm minimization model

As shown in the literature, directly minimizing the sum of the rank function and \( \ell_0 \)-norm of tensor \( \mathcal{X} \) in (8) is not an easy task. So, how to approximate the rank of tensors is crucial for improving the completion quality. In this subsection, we use the sum of the weighted nuclear norms of the unfolding matrices to approximate the rank of \( \mathcal{X} \) and employ the \( \ell_1 \)-norm to replace the \( \ell_0 \)-norm for promoting the sparsity of images under the DCT. Accordingly, we call the model a DCT-based weighted nuclear norm minimization model, which is expressed as the form

\[
\min_{\mathcal{X}} \sum_{i=1}^N \alpha_i \|X_{(i)}\|_* + \lambda \|\mathcal{D}(\mathcal{X})\|_1 \tag{9}
\]
s.t. \( \mathcal{P}_1(\mathcal{X}) = \mathcal{P}_2(\mathcal{H}) \).

Apparently, two parts of the objective function in model (9) are nonsmooth, but both have promising structures so that their proximal operators have explicit forms. To exploit the structure of model (9), we here first introduce auxiliary variables to separate the two nonsmooth terms in the objective function, thereby leading to a separable optimization model, i.e.,

\[
\min_{\mathcal{X}} \sum_{i=1}^N \alpha_i \|Y_{(i)}\|_* + \lambda \|\mathcal{D}(\mathcal{X})\|_1 \tag{10}
\]
s.t. \( \mathcal{X} = Y_i \) for all \( i \in [N] \),

\[
\mathcal{P}_1(\mathcal{X}) = \mathcal{P}_2(\mathcal{H}),
\]
where \( Y_{(i)} \) represents the mode-\( i \) unfolding of tensor \( Y_i \) for \( i \in [N] \). Clearly, model (10) is an equality-constrained optimization problem, then its augmented Lagrangian function reads as

\[
L(Y_i, X, T, S_i, Q) := \sum_{i=1}^N \left( \alpha_i \|Y_{(i)}\|_* + \langle S_i, X - Y_i \rangle + \frac{\beta}{2} \|X - Y_i\|_F^2 \right)
\]

\[
+ \lambda \|T\|_1 + \langle Q, T - \mathcal{D}(\mathcal{X}) \rangle + \frac{\beta}{2} \|T - \mathcal{D}(\mathcal{X})\|_F^2, \tag{11}
\]

where \( S_i \) and \( Q \) are Lagrangian multipliers associated to \( X_{(i)} = Y_{(i)} \) and \( T = \mathcal{D}(\mathcal{X}) \), respectively, and \( \beta \) are positive penalty parameters. Motivated by the alternating update spirit of ADMM, we accordingly update the variables in (11) via the order \( Y_i \to X \to T \to S_i \to Q \). Specifically, for given \( (X^k, T^k, S^k_i, Q^k) \), we update the \( (k+1) \)-th iterates via the following process:

- The update of \( Y_i \) (\( i \in [N] \)) reads as
  \[
  Y_i^{k+1} = \arg \min_{Y_i} L(Y_i, X^k, T^k, S^k_i, Q^k)
  \]
  \[
  = \arg \min_{Y_i} \left\{ \alpha_i \|Y_{(i)}\|_* + \frac{\beta}{2} \|Y_i - S^k_i\|_F^2 \right\}
  \]
  \[
  = \text{fold} \left( U_k \text{wshrink} \left( A^k, \frac{\alpha_i}{\beta_k}, w_i \right) V_k^T \right), \tag{12}
  \]
where ‘fold(·)’ corresponds to the inverse operator of mode-$i$ unfolding, ‘wshrink(·, ·)’ is given by (5) and

$$\hat{Y}_i^k = X^k + \frac{1}{\beta_k} S_i^k \quad \text{and} \quad \hat{Y}_{i(i)}^k = U_k A^k V_k^T.$$

- With the latest $Y_i^{k+1}$, we update $X^{k+1}$ via

$$X^{k+1} = \arg\min_{X} L(Y_i^{k+1}, X, T^k, S_i^k, Q^k)$$

$$= \arg\min_{X} \left\{ \frac{N}{2} \left( \|X - Y_i^{k+1} + \frac{1}{\beta_k} S_i^k \|_F^2 \right) + \frac{\beta_k}{2} \left( \|X - \Phi^{-1} (T^k + \frac{1}{\beta_k} Q^k) \|_F^2 \right) \right\}$$

$$= \frac{1}{N+1} \hat{X}^k,$$ (13)

where

$$\hat{X}^k = \sum_{i=1}^{N} \left( Y_i^{k+1} - \frac{1}{\beta_k} S_i^k \right) + \Phi^{-1} \left( T^k + \frac{1}{\beta_k} Q^k \right).$$

Due to the constraint $P_{\Omega}(X) = P_{\Omega}(\mathcal{H})$, we further implement

$$P_{\Omega}(X^{k+1}) = P_{\Omega}(\mathcal{H})$$ (14)

to get the new $X^{k+1}$.

- For the $T$-subproblem, we have

$$T^{k+1} = \arg\min_{T} L(Y_i^{k+1}, X^{k+1}, T, S_i^k, Q^k)$$

$$= \arg\min_{T} \left\{ \lambda \|T\|_1 + \frac{\beta_k}{2} \|T - \hat{T}^k\|_F^2 \right\}$$

$$= \text{shrink} \left( \frac{\hat{T}^k}{\lambda}, \frac{\beta_k}{2} \right),$$ (15)

where ‘shrink(·, ·)’ is given by (4) and

$$\hat{T}^k = \Phi(X^{k+1}) - \frac{1}{\beta_k} Q^k.$$

- With the above latest $(Y_i^{k+1}, X^{k+1}, T^{k+1})$, we update the two Lagrangian multipliers $S_i^{k+1}$ and $Q^{k+1}$ via

$$S_i^{k+1} = S_i^k + \beta_k (X^{k+1} - Y_i^{k+1})$$

(16)

and

$$Q^{k+1} = Q^k + \beta_k (T^{k+1} - \Phi(X^{k+1})).$$ (17)

respectively.

With the above preparations, we formally summarize the updating schemes for model (10) in Algorithm 1.

**Algorithm 1 ADMM for Model (10)**

**Input:** Initial starting points $X^0, Y^0, T^0, S_i^0, Q_i^0$ and $\beta_0 > 0$.

1. Update $Y_i^{k+1}$ simultaneously via (12) for $i = [N]$;
2. Update $X^{k+1}$ via (13) and (14);
3. Update $T^{k+1}$ via (15);
4. Update $S_i^{k+1}$ simultaneously via (16) for $i = [N]$;
5. Update $Q_i^{k+1}$ via (17);
6. Update $\beta_{k+1} = p \beta_k$ with $p > 1$;
7. Until a termination criterion is fulfilled.

**Output:** $X^*$.

shrinking low-rank penalty and DCT-driven sparse regularization, and propose the following tensor completion model:

$$\min_{X} \sum_{i=1}^{N} \alpha_i \Phi_p(X_{i(i)}) + \lambda \| \Phi(X) \|_1$$ (18)

s.t. $P_{\Omega}(X) = P_{\Omega}(\mathcal{H}),$

where $p \leq 1$ and $\Phi_p(·)$ is a nonconvex penalty function defined in (6). Compared to model (9), the only difference is that we here employ the $\Phi_p(X_{i(i)})$ to replace the weighted nuclear norm term $\|X_{i(i)}\|_{w}$. So, both (9) and (18) share the same structure except the low-rank-inducing term. Hence, letting us abuse the symbols used in (10) for notational simplicity, then model (18) can be rewritten as the following separable optimization problem, i.e.,

$$\min_{X} \sum_{i=1}^{N} \alpha_i \Phi_p(Y_{i(i)}) + \lambda \| \Phi(X) \|_1$$ s.t. $X = Y_i$, for all $i \in [N],$

$$T = \Phi(X),$$

$$P_{\Omega}(X) = P_{\Omega}(\mathcal{H}).$$ (19)

Correspondingly, the augmented Lagrangian function for (19) reads as

$$L(Y_i, X, T, S_i, Q)$$

$$:= \sum_{i=1}^{N} \left( \alpha_i \Phi_p(Y_{i(i)}) + \langle S_i, X - Y_i \rangle + \frac{\beta}{2} \|X - Y_i\|_F^2 \right) + \lambda \|T\|_1 + \langle Q, T - \Phi(X) \rangle + \frac{\beta}{2} \|T - \Phi(X)\|_F^2.$$ (20)

Based upon (20), by invoking the update idea of Algorithm 1, i.e., $Y_i \rightarrow X \rightarrow T \rightarrow S_i \rightarrow Q$, we derive the detailed iterative schemes for solving model (19).

- Update $Y_i^{k+1}$ via

$$Y_i^{k+1} = \arg\min_{Y_i} L(Y_i, X^k, T^k, S_i^k, Q^k)$$

$$= \arg\min_{Y_i} \left\{ \alpha_i \Phi_p(Y_{i(i)}) + \frac{\beta}{2} \|Y_i - S_i^k\|_F^2 \right\}$$

$$= \text{fold} \left( \hat{U}_k \hat{Y}_i, \frac{\alpha_i}{\beta_k} p, \frac{\alpha_i}{\beta_k} T^k \right),$$ (21)

where ‘fold(·, ·, ·)’ is given by (3) and

$$\hat{Y}_i^k = X^k + \frac{1}{\beta_k} S_i^k \quad \text{and} \quad \hat{Y}_{i(i)}^k = U_k A^k V_k^T.$$

- Update $X^{k+1}$ via

$$X^{k+1} = \arg\min_{X} L(Y_i^{k+1}, X^{k}, T^k, S_i^k, Q^k)$$

**B. A DCT-based p-shrinking tensor completion model**

In model (9), we employ the sum of the weighted nuclear norms of the unfolding matrices $X_{i(i)}$ to approximate the low-rankness of the data. However, we do not know whether such an approximation is enough ideal for color images and videos tensor data. Recently, the so-called $p$-shrinking algorithm is widely used in low-rank and sparse recovery problems. Hence, in this subsection, we further consider the combination of $p$-
\[ \frac{1}{N + 1} \tilde{X}^k \]  
(22)

and

\[ \mathcal{P}_\Omega(X^{k+1}) = \mathcal{P}_\Omega(H), \]  
(23)

where

\[ \tilde{X}^k = \sum_{i=1}^{N} \left( y_i^{k+1} - \frac{1}{\beta_k} \sigma_i^k \right) + \mathcal{P}^{-1}(\mathcal{T}^{k} + \frac{1}{\beta_k} Q^k). \]  

- Update \( \mathcal{T}^{k+1} \) via

\[ \mathcal{T}^{k+1} = \arg \min_{\mathcal{T}} \mathcal{L}(\mathcal{Y}^{k+1}, X^{k+1}, \mathcal{T}, \mathcal{S}^k, Q^k) \]

\[ = \text{shrink} \left( \mathcal{P}(X^{k+1}) - \frac{1}{\beta_k} Q^k, \frac{\lambda}{\beta_k} \right). \]  
(24)

- Finally, update the two Lagrangian multipliers \( \mathcal{S}^k_{i+1} \) and \( \mathcal{Q}^{k+1} \) via (16) and (17), respectively.

Formally, we summarize the iterative schemes for (19) in Algorithm 2.

Algorithm 2 ADMM for Model (19).

**Input:** Initial starting points \( X^0, Y^0, T^0, S_0, Q_0 \).

1. Update \( Y_i^{k+1} \) simultaneously via (21) for \( i \in [N] \); 
2. Update \( X^{k+1} \) via (22) and (23); 
3. Update \( T^{k+1} \) via (24); 
4. Update \( \mathcal{S}^{k+1} \) simultaneously via (16) for \( i \in [N] \); 
5. Update \( Q^{k+1} \) via (17); 
6. Update \( \beta_{k+1} = \beta_k, \theta > 1 \); 
7. Until a termination criterion is fulfilled.

**Output:** \( X^* \).

Below, we follow the way used in [54] to prove the convergence result of Algorithm 1 for problem (10) under some assumptions. Since Algorithm 2 share the almost same iterative schemes expect the \( \mathcal{Y} \)-subproblem, we skip the convergence analysis of Algorithm 2 for model (19) for the conciseness of the paper.

**Lemma 3.1** ([55]). The function of the weighted sum of singular values, i.e., \( f(A) = \|A\|_{s,\omega} = \sum_{i=1}^{r} w_i \sigma_i(A) \) for matrix \( A \in \mathbb{R}^{n \times m} \), is Lipschitz continuous.

Since \( \|A\|_{s,\omega} \) is nonconvex and nonsmooth, the typical subdifferential for convex function would be empty set. Therefore, we use the Clarke subdifferential (see Definition 3.2 in [56]), which is a general case of the subdifferential and denoted by \( \partial_C(\cdot) \). Regardless of nonconvexity or nonsmoothness, the Clarke subdifferential always exists for locally Lipschitz continuous functions. Also, \( \partial_C\|A\|_{s,\omega} \) is well defined, since \( \|A\|_{s,\omega} \) is a Lipschitz continuous function (see Lemma 3.1).

**Lemma 3.2.** Suppose that both \( \{S_k\} := \{S_1^k, \ldots, S_N^k\} \) and \( \{Q^k\} \) are bounded, the sequences \( \{Y^k\} := \{Y_1^k, \ldots, Y_N^k\} \) and \( \{T^k\} \) produced by Algorithm 1 are bounded.

**Proof.** From the definition of \( L(Y, X, T, S, Q) \), we have

\[ L(Y^k, X^k, T^k, S^k, Q^k) - L(Y^k, X^k, T^k, S^{k-1}, Q^{k-1}) \]

\[ = \sum_{i=1}^{N} (s_i^k - s_i^{k-1}, x^k - y_i^k) + (q^k - q^{k-1}, t^k - \mathcal{P}(x^k)). \]

By invoking \( \beta_k = \theta \beta_{k-1} \) with \( \theta > 1 \), we immediately prove \( \sum_{i=1}^{\infty} \beta_i + \beta_{k-1} < \infty \). As a consequence of (26), the sequence \( \{L(Y^k, X^k, T^k, S^k, Q^k)\} \) is upper-bounded due to the boundedness of \( \{S^k\} \) and \( \{Q^k\} \). Rearranging terms of \( \{L(Y^k, X^k, T^k, S^k, Q^k)\} \) immediately yields

\[ \sum_{i=1}^{N} \left( \alpha_i \|y_i^k\|_{s,\omega} + \lambda \|t^k\|_1 \right) = L(Y^k, X^k, T^k, S^{k-1}, Q^{k-1}) - \frac{\beta_{k-1}}{2} \left( \|Q^k\|_F^2 - \|Q^{k-1}\|_F^2 \right) \]

\[ - \frac{\beta_{k-1}}{2} \sum_{i=1}^{N} \left( \|s_i^k\|_F^2 - \|s_i^{k-1}\|_F^2 \right), \]

which, together with the boundedness of the sequences \( \{L(Y^k, X^k, T^k, S^k, Q^k)\} \), implies that \( \{Y_i^{k+1}\} \) and \( \{T^k\} \) for all \( i \in [N] \) are bounded, since \( \|\cdot\|_{s,\omega} \) and \( \|\cdot\|_1 \) are nonnegative.

**Theorem 3.3.** Let \( \{L(Y^k, X^k, T^k, S^k, Q^k)\} \) be a sequence generated by Algorithm 1. Suppose that both \( \{S^k\} \) and \( \{Q^k\} \) are bounded, and further satisfy \( \lim_{k \to \infty} \|S^{k+1} - S^k\|_F = 0 \) and \( \lim_{k \to \infty} \|Q^{k+1} - Q^k\|_F = 0 \), respectively. Then,\n
(i) \( \{X^k\}, \{Y^k\}, \{T^k\} \) are bounded Cauchy sequences.

(ii) any accumulation point \( \{(X^\infty, X^\infty, \Omega^\infty, S^\infty, Q^\infty)\} \) satisfies the KKT condition of problem (10).

**Proof.** For any \( i \in [N] \), it easily follows from the iterative scheme (16) and \( \lim_{k \to \infty} \beta_k = \infty \) that the sequence \( \{X^k\} \) is bounded due to Lemma 3.2. Moreover, by the properties of \( \{S^k\} \) and \( \{Q^k\} \), we have

\[ \begin{align*}
\lim_{k \to \infty} \|X^{k+1} - X^k\|_F &= 0, \\
\lim_{k \to \infty} \|T^{k+1} - T^k\|_F &= 0,
\end{align*} \]

(27)
which further implies that \( \{Y^k\} \) and \( \{T^k\} \) approach to feasible solutions.

Now, we show that \( \{X^k\} \), \( \{Y^k\} \), \( \{T^k\} \) are Cauchy sequences. Here, we only show \( \{Y^k\} \) being a Cauchy sequence, while the other sequences can be proved in a similar way.

Invoking the first-order optimality condition of the \( Y^k \)-subproblem yields
\[
0 \in \alpha_i \partial \delta \|Y^k\|_{*,w} - \beta_k \left( X^k(i) - Y^k(i) + \frac{1}{\beta_k} S^k(i) \right),
\]
which, together with (16), leads to
\[
X^k(i) - Y^k(i) + \frac{1}{\beta_k} S^k(i) = 0.
\]

By the boundeness of both \( \{Y^k\} \) and \( \{S^k\} \) and properties of \( \| \cdot \|_{*,w} \), it follows from (30) that \( \|X^k - X^k+1\|_{F} = O(\beta_k^{-1}) \), which, together with \( \sum_{i=0}^{\infty} \beta_k = \frac{\beta_k}{\beta_k^{(1-\epsilon)} - \infty} \), implies that \( \{X^k\} \) is a Cauchy sequence, and it immediately has a limit point.

Let \( X^\infty, Y^\infty, T^\infty, S^\infty, Q^\infty \) be the limit points of \( \{X^k\}, \{Y^k\}, \{T^k\}, \{S^k\}, \{Q^k\} \), respectively. It first follows from (27) that
\[
X^\infty = Y^\infty, (i = 1, \ldots, N) \quad \text{and} \quad T^\infty = \partial (X^\infty).
\]

Rearranging (28) arrives at
\[
0 \in \alpha_i \partial \delta \|Y^k\|_{*,w} - S^k(i) + \beta_k \left( X^k(i) - X^k+1(i) \right)
= \alpha_i \partial \delta \|Y^k\|_{*,w} - S^k(i) - \beta_k \left( X^k - X^k+1 \right).
\]

Consequently, taking limit on (29) immediately yields \( S^\infty(i) = \alpha_i \partial \delta \|Y^\infty\|_{*,w} \).

By the first-order optimality condition of the \( X^k \)-subproblem, we have
\[
\sum_{i=1}^{N} \left( S^k(i) + \beta_k (X^k+1(i) - Y^k(i)) \right)
+ \beta_k \left( X^k(i) - T^k(i) \right)
= 0.
\]

Taking \( k \to \infty \) on the above two equalities leads to
\[
\sum_{i=1}^{N} \partial \delta(X^\infty) \quad \text{and} \quad \partial \delta(X^\infty) = \partial \delta(H). \]

Finally, the first-order optimality condition of the \( T^k \)-subproblem reads as
\[
0 \in \partial \lambda \|T^k\|_1 + \beta_k \left( T^k - \partial (X^k) + \beta_k^{-1} Q^k \right)
= 0 \in \partial \lambda \|T^k\|_1 + Q^k + \beta_k \left( T^k - \partial (X^k) \right)
= 0 \in \partial \lambda \|T^k\|_1 + Q^k.
\]

Consequently, we have \( -Q^* \in \partial \lambda \|T^*\|_1 \) when \( k \to \infty \). Therefore, we conclude that the accumulation point satisfies the KKT condition of model (10).

Hereafter, we only give the time complexity analysis of Algorithm 1 and omit the analysis of Algorithm 2 since both of them share the almost iterative scheme and the same SVD.

The main time cost of Algorithm 1 is consumed by performing SVD and multi-dimensional DCT. In each iteration, the complexity of subproblem \( Y^k+1 \) (\( i \in [N] \)) is
\[
O \left( \sum_{i=1}^{N} (I_i)^2 \times \prod_{j \neq i} (I_j) \right).
\]

The complexities of subproblems \( X^k+1 \) and \( T^k+1 \) are both \( O (\Pi_{i=1}^{N} I_i \times \log \Pi_{i=1}^{N} I_i) \). In addition, the complexities to update \( S^k+1 \) (\( i \in [N] \)) and \( Q^k+1 \) are \( O (N \Pi_{i=1}^{N} I_i) \) and \( O (\Pi_{i=1}^{N} I_i) \). So, the time complexity of each iteration is
\[
O \left( \sum_{i=1}^{N} (I_i)^2 \times \prod_{j \neq i} (I_j) + \prod_{i=1}^{N} I_i \times \log \Pi_{i=1}^{N} I_i \right).
\]

Thus, the total time complexity of Algorithm 1 is
\[
O \left( t \left( \sum_{i=1}^{N} (I_i)^2 \times \prod_{j \neq i} (I_j) + \prod_{i=1}^{N} I_i \times \log \Pi_{i=1}^{N} I_i \right) \right), \quad \text{where} \quad t \quad \text{is the number of iterations}.
\]

IV. NUMERICAL EXPERIMENTS

In this section, we are concerned with the numerical performance of our approach proposed in Section III on images and videos recovery. Here, we will consider two kinds of images and videos datasets: 1) RGB images; 2) surveillance videos. All algorithms were implemented in MATLAB R2018b (64bit) and experiments were conducted on a laptop computer with Intel(R) Core(TM) i7-7500 CPU @ 2.70GHz and 8GB memory. Throughout this section, we denote Algorithms 1 and 2 by ‘DCT-WNN’ and ‘DCT-IpST’ for simplicity, respectively. Moreover, we also compare the proposed algorithms with five state-of-the-art tensor completion approaches as follows:

- HaLRTC [9]: High accuracy low-rank tensor completion, which is based on the direct nuclear norms of mode-i unfolding matrices and ADMM algorithmic framework.
- STDC [13]: Simultaneous tensor decomposition and completion, which combines a rank minimization technique with Tucker model decomposition and is solved by an inexact augmented Lagrangian method.
- TNNL1 [43]: Using the truncated tensor nuclear norm for low-rank approximation, and a sparse regularization term combined with the 3-D DCT bases.
- IpST [50]: Iterative p-shrinkage thresholding algorithm for solving low Tucker rank tensor recovery problem, which only employs a nonconvex penalty function \( \Phi_p(\cdot) \) given in (6) to replace the traditional nuclear norm of unfolding matrices.
- TNN-3DTV [39]: Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm, which utilizes the 3D total variation regularization to exploit the structure of images.

For the fair comparison, we take
\[
\text{RelCha} = \frac{\|X^k+1 - X^k\|_F}{\|X^\text{true}\|_F} \leq 10^{-4},
\]
as the stopping criterion for all methods, where \( X^\text{true} \) is the true tensor. Moreover, we use the Peak Signal-to-Noise Ratio (PSNR), i.e.,
\[
\text{PSNR} = 10 \log_{10} \left( \frac{\|X^\text{max}\|}{\|X^* - X^\text{true}\|_F^2} \right)
\]
to measure the quality of the restored tensor data by an algorithm, where \( \|\cdot\| \) denotes the number of elements in the complementary set of \( \Omega \), \( X^\text{max} \) represents the largest element of \( X^\text{true} \), and \( X^* \) corresponds to the restored tensor. Since

\footnote{Matlab code: http://mp.cs.nthu.edu.tw/}
there are some parameters in models and algorithms, for both models (10) and (19), we set \((\alpha_1, \alpha_2, \alpha_3) = (1/3, 1/3, 10^{-3})\) for image inpainting and \((\alpha_1, \alpha_2, \alpha_3) = (1/3, 1/3, 1/3)\) for video inpainting, respectively. Moreover, we take \(\lambda = 0.05\) and \(\lambda = 10^{-2}\) for (10) and (19), respectively. For the algorithmic parameters, we set \(p = 0.2\) for DCT-IpST throughout the experiments. In addition, we set \(\varrho = 1.2\) and \(\beta_0 = 10^{-5}\), respectively. All parameters of the other compared algorithms were taken as the default values used in the paper.

A. RGB image inpainting

In this subsection, we consider the RGB image inpainting problem, where RGB images namely includes Red, Green, and Blue channels and the number of channels corresponds to the mode a third order tensor. Here, we conduct the numerical performance our approach on ten widely used images, which are summarized in Fig. 2. These images have no visual features such as low-rank and sparsity except the ‘facade’ image. However, as shown in Fig. 3, the sparsity is apparent when applying DCT to the images. Hence, Fig. 3 sufficiently supports that the main idea of this paper is reasonable.

Fig. 2. Ten RGB images for experiments.

![Fig. 2. Ten RGB images for experiments.](image)

Fig. 3. Sparsity of RGB images after DCT.

![Fig. 3. Sparsity of RGB images after DCT.](image)

Now, we are concerned with the inpainting performance of our approach on the case where the pixels of images are missed in a random way. In Fig. 4, we first consider four scenarios on the sample ratio \(('sr')\), i.e., \(sr = \{5\%, 10\%, 20\%, 30\%\}\), for five images (i.e., ‘airplane’, ‘barbara’, ‘facade’, ‘house’, ‘lena’). Clearly, the recovered images by our methods, i.e., DCT-WNN and DCT-IpST, are better than the other five methods for the scenarios where \(sr\)'s are lower than 10\% (see the first three columns of Fig. 4). To investigate the full sensitivity of these methods to \(sr\), we further consider ten scenarios on \(sr\)'s from 5\% to 90\% for the images in Fig. 2 and plot PSNR values with respect to \(sr\)'s in Fig. 5. It can be easily seen from Fig. 5 that both DCT-WNN and DCT-IpST outperform the other five methods in terms of taking higher PSNR values for all scenarios.

Below, we are interested in the numerical performance on structured missing scenarios such as slices missing or shape/text mask (i.e., downsampling operator). Specifically, we consider five scenarios as shown by the first row of Fig. 6. Obviously, we see from rows 2-7 in Fig. 6 that our DCT-WNN and DCT-IpST have surprising abilities to recover the structured missing images. Moreover, for showing details of all methods, we further report the RSE and PSNR values, number of iterations (Iter for short) and computing time in seconds.
frames, it is obvious that both DCT-WNN and DCT-IpST can successfully obtain relatively ideal videos even for highly undersampled cases.

In summary, the above computational results show that the proposed DCT-WNN and DCT-IpST are powerful to complete different types of downsampled images and videos datasets. Also, these results support the novelty of our DCT-based approach for image and video inpainting problems.

V. Conclusion

In this paper, we introduced a unified DCT-based low-rank tensor completion approach, which includes two structured optimization models for recovering images and videos from highly undersampled data. By introducing auxiliary variables, we gainfully separated the two nonsmooth terms appeared in the objective functions, thereby being of benefit for designing easily implementable algorithms for the underlying separable models. A series of computational experiments on RGB images and surveillance videos datasets demonstrated that our approach performs better than some state-of-the-art tensor completion approaches.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Yufan Li for sharing their code of the paper [50]. C. Ling and H. He were supported in part by National Natural Science Foundation of China (Nos. 11971138 and 11771113) and Natural Science Foundation of Zhejiang Province (Nos. LY19A010019, LY20A010018, and LD19A010002).

REFERENCES

[1] N. Komodakis, “Image completion using global optimization,” in Computer Vision and Pattern Recognition, vol. 1, 2006, pp. 442–452.
[2] N. Hao, M. E. Kilmer, K. Braman, and R. C. Hoover, “Facial recognition using tensor-tensor decompositions,” SIAM J. Imaging Sci., vol. 6, no. 1, pp. 437–463, 2013.
[3] A. Criminisi, P. Perez, and K. Toyama, “Region filling and object removal by exemplar-based image inpainting,” IEEE Trans. Image Process., vol. 13, no. 9, pp. 1200–1212, 2004.
[4] X. Geng, K. Smith-Miles, Z. Zhou, and L. Wang, “Face image modeling by multilinear subspace analysis with missing values,” IEEE Trans. Syst. Man Cybern. Part B Cybern., vol. 41, no. 3, pp. 881–892, 2011.
[5] A. Efros and T. Leung, “Texture synthesis by non-parametric sampling,” in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, 1999, pp. 1033–1038.
## Table I

### Numerical Results of Images with Structured Missing Pixels as Tested in Fig. 6.

| Method        | RSE (Facade-1) | PSNR (Facade-1) | Iter (Time) | RSE (Lena-1) | PSNR (Lena-1) | Iter (Time) | RSE (Baboon) | PSNR (Baboon) | Iter (Time) |
|---------------|----------------|-----------------|-------------|--------------|--------------|-------------|--------------|--------------|-------------|
| HaLRTC        | 0.497          | 11.80           | 1 (0.06)    | 0.508        | 11.62        | 177 (7.90)  | 0.245        | 17.33        | 183 (8.72)  |
| STDC          | 0.165          | 21.36           | 50 (17.04)  | 0.508        | 11.61        | 50 (3.36)   | 0.608        | 9.43         | 50 (5.61)   |
| TTNNL1        | 0.295          | 16.33           | 38 (5.44)   | 0.262        | 17.35        | 49 (6.89)   | 0.817        | 18.92        | 49 (7.15)   |
| IpST          | 0.497          | 11.80           | 2 (0.16)    | 0.508        | 11.66        | 81 (7.60)   | 0.204        | 18.92        | 81 (7.60)   |
| TNN-3DTV      | 0.215          | 19.07           | 200 (75.49) | 0.262        | 17.38        | 200 (73.40) | 0.187        | 19.76        | 200 (75.25) |
| DCT-IpST      | 0.147          | 21.76           | 89 (12.26)  | 0.262        | 17.38        | 89 (12.26)  | 0.147        | 21.76        | 89 (12.26)  |
| DCT-WNN       | 0.095          | 26.13           | 50 (8.50)   | 0.262        | 17.38        | 50 (8.50)   | 0.119        | 24.21        | 50 (8.50)   |

Fig. 6. Images recovered by the seven algorithms for the cases with random missing information. The top row corresponds to the observed incomplete images, where the first facade image is corrupted by dropping ten block slices information along mode-1; the second facade image is corrupted by dropping ten block slices while dropping 90% pixels in a random way; the first Lena image first randomly drops 90% pixel and then adds a word mask; the second Lena image is corrupted by adding a text mask and corrupted by drop 70% pixels in a random way; the last Baboon image is corrupted by dropping one slice for every two slices along mode-1 and mode-2. From the second row to bottom: Images recovered by HaLRTC, STDC, TTNNL1, IpST, TNN-3DTV, DCT-IpST and DCT-WNN, respectively.

[6] D. Coupier, A. Desolneux, and B. Ycart, “Image denoising by statistical area thresholding,” in Mustererkennung, 21 Dagm-symposium, Bonn, September, 1999.

[7] C. Guillemot and O. Le Meur, “Image inpainting: Overview and recent advances,” IEEE Signal Process Mag., vol. 31, pp. 127–144, 2013.

[8] Q. Song, H. Ge, J. Caverlee, and X. Hu, “Tensor completion algorithms in big data analytics,” ACM Trans. Knowl. Discovery Data, vol. 13, pp. 1–48, 2019.

[9] J. Liu, P. Musialski, P. Wonka, and J. Ye, “Tensor completion for estimating missing values in visual data,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 1, pp. 208–220, 2013.

[10] S. Gandy, B. Recht, and I. Yamada, “Tensor completion and low-n-rank tensor recovery via convex optimization,” Inverse Probl., vol. 27, p. 025010, 2011.

[11] C. Lu, J. Feng, Z. Lin, and S. Yan, “Exact low tubal rank tensor recovery from gaussian measurements,” in Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. International Joint Conferences on Artificial Intelligence Organization, 7
| Videos Datasets | Method      | \( sr = 0.05 \) | \( sr = 0.10 \) | \( sr = 0.20 \) | \( sr = 0.30 \) |
|----------------|-------------|----------------|----------------|----------------|----------------|
|                |             | PSNR | RSE  | PSNR | RSE  | PSNR | RSE  | PSNR | RSE  | PSNR | RSE  |
| Hall Monitor   | HaLRTC      | 8.39 | 0.645 | 13.69 | 0.351 | 23.21 | 0.117 | 26.52 | 0.080 |
|                | STDC        | 9.05 | 0.611 | 20.28 | 0.165 | 32.08 | 0.042 | 34.99 | 0.030 |
| 144 \times 176 \times 100 | TTNL1        | 22.69 | 0.124 | 27.09 | 0.075 | 25.11 | 0.093 | 27.63 | 0.070 |
|                | IpST        | 5.09 | 0.944 | 5.39  | 0.912 | 30.39 | 0.051 | 34.03 | 0.033 |
|                | TNN-3DTV    | 21.20 | 0.147 | 22.52 | 0.126 | 25.11 | 0.093 | 27.63 | 0.070 |
|                | DCT-IpST    | 25.85 | 0.086 | 30.78 | 0.049 | 34.70 | 0.031 | 36.84 | 0.024 |
|                | DCT-WNN     | 25.79 | 0.087 | 31.56 | 0.045 | 35.23 | 0.029 | 37.24 | 0.023 |
| Suzie          | HaLRTC      | 7.25 | 0.974 | 7.49  | 0.948 | 7.99  | 0.894 | 8.37  | 0.837 |
| 144 \times 176 \times 100 | STDC        | 11.52 | 0.609 | 24.75 | 0.130 | 30.82 | 0.064 | 32.14 | 0.054 |
|                | TTNL1       | 24.03 | 0.139 | 27.92 | 0.075 | 32.08 | 0.042 | 34.99 | 0.030 |
|                | IpST        | 7.69 | 0.926 | 27.89 | 0.090 | 30.39 | 0.051 | 34.03 | 0.033 |
|                | TNN-3DTV    | 26.85 | 0.109 | 28.87 | 0.079 | 31.39 | 0.059 | 33.25 | 0.047 |
|                | DCT-IpST    | 26.95 | 0.100 | 30.30 | 0.068 | 32.88 | 0.051 | 34.34 | 0.042 |
|                | DCT-WNN     | 28.41 | 0.084 | 31.51 | 0.045 | 34.86 | 0.028 | 37.34 | 0.023 |
| Akiyo          | HaLRTC      | 6.76 | 0.974 | 6.99  | 0.948 | 28.66 | 0.078 | 31.57 | 0.056 |
| 144 \times 176 \times 100 | STDC        | 10.78 | 0.627 | 21.86 | 0.172 | 27.37 | 0.091 | 29.40 | 0.072 |
|                | TTNL1       | 26.41 | 0.116 | 29.88 | 0.078 | 35.32 | 0.041 | 38.78 | 0.028 |
|                | IpST        | 7.14 | 0.932 | 28.37 | 0.081 | 33.28 | 0.046 | 37.68 | 0.027 |
|                | TNN-3DTV    | 24.23 | 0.130 | 25.98 | 0.106 | 28.73 | 0.077 | 31.20 | 0.058 |
|                | DCT-IpST    | 28.27 | 0.081 | 32.51 | 0.050 | 37.47 | 0.028 | 40.62 | 0.019 |
|                | DCT-WNN     | 30.22 | 0.075 | 34.81 | 0.044 | 39.80 | 0.024 | 42.89 | 0.017 |
| Seafish        | HaLRTC      | 7.70 | 0.974 | 7.93  | 0.948 | 8.44  | 0.894 | 9.02  | 0.836 |
| 185 \times 290 \times 300 | STDC        | 11.32 | 0.658 | 17.40 | 0.322 | 20.10 | 0.240 | 23.28 | 0.165 |
|                | TTNL1       | 17.45 | 0.306 | 20.68 | 0.211 | 25.26 | 0.124 | 28.63 | 0.084 |
|                | IpST        | 17.06 | 0.332 | 19.68 | 0.243 | 23.27 | 0.162 | 26.36 | 0.113 |
|                | TNN-3DTV    | 18.34 | 0.286 | 19.49 | 0.251 | 21.59 | 0.196 | 23.59 | 0.156 |
|                | DCT-IpST    | 19.05 | 0.264 | 23.31 | 0.161 | 28.41 | 0.089 | 31.15 | 0.065 |
|                | DCT-WNN     | 19.81 | 0.233 | 23.90 | 0.145 | 27.87 | 0.092 | 30.24 | 0.070 |

Fig. 7. Sparsity of videos data after DCT.

[12] P. Zhou, C. Y. Lu, Z. C. Lin, and C. Zhang, “Tensor factorization for low-rank tensor completion,” IEEE Trans. Image Process., vol. 27, no. 3, pp. 1152–1163, 2018.

[13] Y. L. Chen, C. T. Hsu, and H. Y. Liao, “Simultaneous tensor decomposition and completion using factor priors,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 3, pp. 577–591, 2014.

[14] E. J. Candès and B. Recht, “Exact matrix completion via convex optimization,” Found. Comput. Math., vol. 9, no. 6, pp. 717–722, 2009.

[15] C. J. Hillar and L. H. Lim, “Most tensor problems are NP-hard,” J. ACM, vol. 6, pp. 1–39, 2013.

[16] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” SIAM Rev., vol. 51, pp. 455–500, 2009.

[17] I. Oseledets, “Tensor-train decomposition,” SIAM J. Sci. Comput., vol. 33, no. 5, pp. 2295–2317, 2011.

[18] T. X. Jiang, T. Z. Huang, X. L. Zhao, T. Y. Ji, and L. J. Deng, “Matrix factorization for low-rank tensor completion using framelet prior,” Inf. Sci., vol. 436-437, pp. 403–417, 2018.

[19] K. Hosono, S. Ono, and T. Miyata, “Weighted tensor nuclear norm minimization for color image denoising,” in 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 3081–3085.

[20] L. T. Huang, H. C. So, Y. Chen, and W. Q. Wang, “Truncated nuclear norm minimization for tensor completion,” in IEEE Sensor Array and Multichannel Signal Processing Workshop, 2014.

[21] J. A. Bengua, H. Phien, H. D. Tuan, and M. N. Do, “Efficient tensor completion for color image and video recovery: Low-rank tensor train,” IEEE Trans. Image Process., vol. 26, no. 5, pp. 2466–2479, 2017.

[22] T. Y. Ji, T. Z. Huang, X. L. Zhao, T. H. Ma, and G. Liu, “Tensor completion using total variation and low-rank matrix factorization,” Inf. Sci., vol. 326, pp. 243–257, 2016.
Fig. 8. Visualizations of recovered video datasets by the seven algorithms. From top to bottom: frames correspond to Hall Monitor, Suize, Akiyo, and Seafish with sr = 5%, 10%, 20% and 30%, respectively. From left to right: clean frames, observed frames, and frames recovered by HaLRTC, STDC, TTNNL1, IpST, TNN-3DTV, DCT-IpST and DCT-WNN, respectively.

minimization and its applications to tensor recovery,” Inf. Sci., vol. 503, pp. 109–128, 2019.

[24] Z. Zhang and S. Aeron, “Exact tensor completion using t-SVD,” IEEE Trans. Signal Process., vol. 65, no. 6, pp. 1511–1526, 2017.

[25] 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 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 5249–5257.

[26] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, “Tensor robust principal component analysis with a new tensor nuclear norm,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, pp. 925–938, 2020.

[27] M. Yuan and C. H. Zhang, “On tensor completion via nuclear norm minimization,” Found. Comput. Math., vol. 16, pp. 1031–1068, 2016.

[28] M. Kilmer and C. Martin, “Factorization strategies for third-order tensors,” Linear Algebra Appl., vol. 435, no. 3, pp. 641-658, 2011.

[29] Q. B. Zhao, L. Q. Zhang, and A. Cichocki, “Bayesian CP factorization of incomplete tensors with automatic rank determination,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1751–1763, 2015.

[30] M. Filipović and A. Jukić, “Tucker factorization with missing data with application to low-n-rank tensor completion,” Multidim. Syst. Sign. Process., vol. 26, no. 3, pp. 677–692, 2015.

[31] T. Yokota, Q. Zhao, and A. Cichocki, “Smooth PARAFAC decomposition for tensor completion,” IEEE Trans. Signal Process., vol. 64, no. 20, pp. 5423–5436, 2016.

[32] Y. Wu, H. Tan, Y. Li, J. Zhang, and X. Chen, “A fused CP factorization method for incomplete tensors,” IEEE Trans. Neural. Netw. Learn. Syst., vol. 30, no. 3, pp. 751–764, 2019.

[33] X. Lin, M. Ng, and X. Zhao, “Tensor factorization with total variation and tikhonov regularization for low-rank tensor completion in imaging data,” J. Math. Imaging Vis., vol. 62, pp. 900–918, 2020.

[34] Y. Liu, F. Shang, W. Fan, J. Cheng, and H. Cheng, “Generalized higher order orthogonal iteration for tensor learning and decomposition,” IEEE Trans. Neural. Netw. Learn. Syst., vol. 27, no. 12, pp. 2551–2563, 2016.

[35] L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” Physica D, vol. 60, pp. 227–238, 1992.

[36] V. N. Varghees, M. S. Manikandan, and R. Gini, “Adaptive mri image denoising using total-variation and local noise estimation,” in Proceedings of the 2012 International Conference on Advances in Engineering, Science and Management (ICAESM), 2012, pp. 506–511.

[37] M. Qin, Z. Li, S. Chen, Q. Guan, and J. Zheng, “Low-rank tensor completion and total variation minimization for color image inpainting,” IEEE Access, vol. 8, pp. 53 049–53 061, 2020.

[38] X. Zhao, X. Nie, Y. Zheng, T. Ji, and T. Huang, “Low-rank tensor completion via tensor nuclear norm with hybrid smooth regularization,” IEEE Access, vol. 7, pp. 131 888–131 901, 2019.

[39] F. Jiang, X. Liu, H. Lu, and R. Shen, “Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm for color image inpainting,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 1363–1367.

[40] D. Qiu, M. Bai, M. Ng, and X. Zhang, “Robust low-rank tensor completion via transformed tensor nuclear norm with total variation regularization,” Neurocomputing, vol. 435, pp. 197–215, 2021.

[41] Y. Abe and Y. Iiguni, “Image restoration from a downsampled image using the DCT,” Signal Processing, vol. 87, pp. 2370–2380, 2007.
“Exploiting global low-rank structure and local sparsity nature for tensor completion,” *IEEE Trans. Cybern.*, vol. 49, no. 11, pp. 3898–3910, 2019.

[43] Z. Han, C.S. Leung, L.T. Huang, and H. So, “Sparse and truncated nuclear norm based tensor completion,” *Neural Process. Lett.*, vol. 45, pp. 729–743, 2017.

[44] Y. Li, L. Shen, and B. Suter, “Adaptive inpainting algorithm based on DCT induced wavelet regularization,” *IEEE Trans. Image Process.*, vol. 22, pp. 752–763, 2013.

[45] B. Madathil and S. George, “DCT based weighted adaptive multi-linear data completion and denoising,” *Neurocomputing*, vol. 318, pp. 120–136, 2018.

[46] Y. Wang, C. Xu, S. You, C. Xu, and D. C. Tao, “DCT regularized extreme visual recovery,” *IEEE Trans. Image Process.*, vol. 26, pp. 3360–3371, 2018.

[47] Y. Wang and Q. Zhu, “Signal loss recovery in DCT-based image and video codecs,” in *Visual Communications and Image Processing ’91: Visual Communication*, K.-H. Tzou and T. Koga, Eds., vol. 1605, International Society for Optics and Photonics. SPIE, 1991, pp. 667 – 678.

[48] R. Chartrand, “Shrinkage mappings and their induced penalty functions,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1026–1029.

[49] R. Chartrand, “Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data,” in *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2009, pp. 262–265.

[50] K. Shang, Y. F. Li, and Z. H. Huang, “Iterative p-shrinkage thresholding algorithm for low Tucker rank tensor recovery,” *Inf. Sci.*, vol. 482, pp. 374–391, 2019.

[51] S. Gaïffas and G. Lecué, “Weighted algorithms for compressed sensing and matrix completion,” 2011, https://arxiv.org/abs/1107.1638.

[52] C. Lu, J. Tang, S. Yan, and Z. Lin, “Nonconvex nonsmooth low rank minimization via iterative reweighted nuclear norm,” *IEEE Trans. Image Process.*, vol. 25, pp. 829–839, 2016.

[53] J. Woodworth and R. Chartrand, “Compressed sensing recovery via nonconvex shrinkage penalties,” *Inverse Probl.*, vol. 32, p. 075004, 2016.

[54] T. Oh, Y. Tai, J. Bazin, H. Kim, and I. Kweon, “Partial sum minimization of singular values in robust PCA: Algorithm and applications,” *IEEE Trans. Image Process.*, vol. 38, pp. 744–758, 2016.

[55] Y. Mu, P. Wang, L.F. Lu, X.Y. Zhang, and L.Y. Qi, “Weighted tensor nuclear norm minimization for tensor completion using tensor-SVD,” *Pattern Recognit. Lett.*, vol. 130, pp. 4–11, 2020.

[56] A. Bagirov, N. Karmitsa, and M. Mäkelä, *Introduction to Nonsmooth Optimization: Theory, Practice and Software*. Switzerland: Springer, 2014.