Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 19, No. 1, pp. 69–96, 2023

BLIND DECONVOLUTION FOR MULTIPLE OBSERVED IMAGES WITH MISSING VALUES*

Yihang Gao, Xuelei Lin and Michael K. Ng

Abstract: In this paper, we study the problem of blind deconvolution for multiple observed images with missing values. By making use of low rank tensor structure from multiple observed images, we formulate the proposed optimization model consisting of data-fitting term between the observed pixel values and blurred image pixels, low rank tensor completion for multiple observed images, and total variational regularization for the deconvoluted image. The multiple observed images form a third-order tensor where two modes are for spatial domain (under the convolution (blurring) operation) and the remaining mode is for temporal domain. Here we propose to employ the transformed tubal nuclear norm based on the transformation along temporal mode to regularize such observed low rank tensor in the model. We solve the resulting model by inexact proximal alternating minimization and show the convergence of the inexact iterations for the blind deconvolution problem. Numerical examples are reported to show the performance of the proposed model is better than that using single observed image only.

Key words: *tensor completion, blind deconvolution, incomplete observation, transformed tensor nuclear norm*

Mathematics Subject Classification: *49J52, 65F22, 65K10, 68U10, 94A08*

1 Introduction

In image processing, the image blurring is modeled as follows:

$$Y = K \star X + \Xi,$$

where Y is the blurred image with noise, K is the kernel blur, X represents the clean image, Ξ is the additive noise and "$\star$" is the convolution product. Mathematically, the convolution product of $K \in \mathbb{R}^{(2*k_v+1)\times(2*k_b+1)}$ and $X \in \mathbb{R}^{n_v \times n_b}$ is defined as

$$[K \star X](i,j) = \sum_{-k_v \le i' \le k_v,\ -k_b \le j' \le k_b} K(i',j') * X(i+i', j+j').$$

where the support of the kernel K is $(i',j') \in \{-k_v, \ldots, 0 \ldots, k_v\} \times \{-k_b, \ldots, 0 \ldots, k_b\}$. The task of image deconvolution is to recover the clean image X from the observed blurred image Y. In real world applications, K is usually unknown. In such situation, we call the deconvolution as blind deconvolution [19], which requires to recover both the clean image

*Research supported in part by HKRGC GRF 12200317, 12300218, 12300519, 17201020 and 17300021, NSFC-RGC HKU769-21, China Postdoctoral Science Foundation (Grant 2021M702281).

and the kernel. Because of its ill-posedness, the blind deconvolution problem is challenging in both implementation and theory. To reduce its ill-posedness, priors for both clean image and kernel are adopted in blind deconvolution models, see for example, [18, 9, 26, 28]. In [18], Krishnan and Fergus proposed a statistical model with hyper-Laplacian prior. Dong et.al. [9] put TV prior to the clean image and deblur the image in sparse representation. Wang and Ng's model [26] used TV prior and frame based prior for both clean image and blur. Some researchers also focus on recovering an image from several blurred images. Ward and Saleh [28] assumed the time correlation property to the blurs from Gaussian distributions, and used maximum likelihood estimation to recover the image from multiple blurred images.

Because of the availability of multi-dimensional data, low rank tensor completion from incomplete observed data is becoming more and more important in image processing [24, 32, 34, 15], machine learning [22, 33] and data science [2, 30]. CP rank [12] and Tucker rank [25] are the two most popular methods describing tensor rank in low rank tensor completion. Kilmer and Martin [16] proposed tubal rank based on the Fourier transform. Kernfield [14] and Song [24] found that other mathematical transformations can also be implemented in tensor tubal rank.

In our model, multiple blurred images (can be regarded as a tensor) are only partially observed. The slices of the blurred tensor come from a single clean image but are blurred by a sequence of continuously changing kernels. We aim to recover a clean image and a sequence of unknown kernels from the incompletely observed blurred tensor. This model is more challenging than classical blind deconvolution problems as only partial entries of the blurred data are known. It does have applications in real world. Researchers usually take several photos to a fixed scene with different camera settings for wider research use. However, blurred images can be produced out of our control. Camera shake and long distance between the scene and the camera can blur images [10]. In astronomy and remote sensing, atmospheric turbulence can result in blurred images [31]. Lens aberrations will inevitably create blur across images especially for wide apertures and slow adjustment of aperture size leads to continuous changing blurs [8]. During images transmission and storage, there usually exist data loss and corruption. Hence, our model has wide applications in real world.

To the best of our knowledge, it is the first model to do tensor based blind deconvolution to restore a sharp image from multiple blurred images with missing values. Given several incomplete blurred images, our model can be used for blind deconvolution: restoring the clean image X and estimating kernels $\mathcal{K}$, and tensor completion: recovering blurred images $\mathcal{Y}$.

The rest part of this paper is organized as follows. In section 2, some notations and backgrounds mainly about tensor computation are introduced based on Song's paper on transformed tubal nuclear norm (TTNN) [24], a more general form of tubal nuclear norm (TNN). The model is proposed in section 3 and corresponding PAM Iteration algorithm as well as ADMM solvers are developed in section 4. Global convergence of the model with inexact solutions is discussed in section 5. The inexact solution not only means the proximal property of PAM iteration, but also implies errors of the output in each subproblem because we inexactly solve subproblems by ADMM algorithms. Four numerical tests in section 6 show both the effectiveness and robustness of our model. Finally, some conclusions and future works are given in section 7.

2 Notation and Preliminaries

In this section, we review some notations and background of tensor computation and basic definition of semi-algebraic function for convergence analysis in section 5 . For more details

about tensors, please refer to [24, 16]. Tensors are denoted by calligraphy letters, e.g., $\mathcal{X}$. Matrices are denoted by capital letters, e.g., X and we use bold lowercase letters to represent vectors, e.g., $\mathbf{x}$. For a third-order tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, we denote its (i,j,k)-th entry as $\mathcal{X}_{ijk}$, and use the MATLAB notation $\mathcal{X}(i,:,:)$, $\mathcal{X}(:,i,:)$, $\mathcal{X}(:,:,i)$ to denote the i-th horizontal, lateral, and frontal slices, respectively. And for more convenience, we denote the t-th frontal slice of the tensor $\mathcal{X}$ as $\mathcal{X}_t$, i.e., $\mathcal{X}_t = \mathcal{X}(:,:,t)$. The Frobenius norm of a tensor, $||\mathcal{X}||_F^2 = \sum_{ijk} |\mathcal{X}(i,j,k)|^2$.

2.1 Tensor Product and TTNN

Let $\Phi \in \mathbb{C}^{n_3 \times n_3}$ be a unitary matrix with $\Phi\Phi^H = \Phi^H\Phi = I_{n_3}$. Denote $\hat{\mathcal{X}}_\Phi = \Phi[\mathcal{X}]$ as a tensor obtained via multiplying all tubes of the tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ by the unitary matrix Φ along the third dimension, i.e.,

$$\mathbf{vec}(\hat{\mathcal{X}}_\Phi(i,j,:)) = \Phi * \mathbf{vec}(\mathcal{X}(i,j,:)),$$

where **vec** is the vectorization operation and * denotes matrix-vector product. Moreover, we can return the tensor $\hat{\mathcal{X}}_\Phi$ back to $\mathcal{X}$ by multiplying all tubes of $\hat{\mathcal{X}}_\Phi$ by Φ^H along the third dimension, that is $\mathcal{X} = \Phi^H[\hat{\mathcal{X}}_\Phi]$. Let $\bar{\mathcal{X}}_\Phi$ be the block diagonal matrix of the tensor $\hat{\mathcal{X}}_\Phi$, where diagonals of the matrix are slices of the tensor $\hat{\mathcal{X}}_\Phi$, i.e.,

$$\bar{\mathcal{X}}_\Phi = blockdiag(\hat{\mathcal{X}}_\Phi) = \begin{pmatrix} \hat{\mathcal{X}}_{\Phi(1)} & & & \\ & \hat{\mathcal{X}}_{\Phi(2)} & & \\ & & \ddots & \\ & & & \hat{\mathcal{X}}_{\Phi(n_3)} \end{pmatrix}.$$

Meanwhile, we can also fold the block diagonal matrix to a tensor by 'fold' operation:

$$fold(blockdiag(\hat{\mathcal{X}}_\Phi)) = fold(\bar{\mathcal{X}}_\Phi) = \hat{\mathcal{X}}_\Phi.$$

Definition 2.1 (Φ-product [24])**.** The Φ-product of two tensors $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ and $\mathcal{Y} \in \mathbb{C}^{n_2 \times n_4 \times n_3}$, denoted by $\mathcal{X} \diamond_\Phi \mathcal{Y}$, is an $n_1 \times n_4 \times n_3$ tensor given by

$$\mathcal{Z} = \mathcal{X} \diamond_\Phi \mathcal{Y} = \Phi^H[fold(blockdiag(\hat{\mathcal{X}}_\Phi) \cdot blockdiag(\hat{\mathcal{Y}}_\Phi))].$$

This definition is actually compatible with t-product in [16] and t-product can be seen as a special case of the Φ- product because it's actually equivalent to let $\Phi = \frac{1}{\sqrt{n_3}} F_{n_3}$, where F_{n_3} is the discrete Fourier transform matrix.

Definition 2.2 (conjugate transpose of tensor [24])**.** The conjugate transpose of the tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ is the tensor $\mathcal{X}^H \in \mathbb{C}^{n_2 \times n_1 \times n_3}$ obtained by

$$\mathcal{X}^H = \Phi^H[fold(blockdiag(\hat{\mathcal{X}}_\Phi)^H)].$$

Definition 2.3 (identity tensor [16, 24])**.** The indentity tensor $\mathcal{I}_\Phi \in \mathbb{C}^{n \times n \times n_3}$ with respect to Φ, is defined as $\mathcal{I}_\Phi = \Phi[\mathcal{T}]$, with each frontal slice of $\mathcal{T} \in \mathbb{R}^{n \times n \times n_3}$ being the $n \times n$ identity matrix.

Definition 2.4 (unitary tensor [14, 24])**.** A tensor $\mathcal{Q} \in \mathbb{C}^{n \times n \times n_3}$ is unitary with respect to Φ-product if it satisfies the following:

$$\mathcal{Q}^H \diamond_\Phi \mathcal{Q} = \mathcal{Q} \diamond_\Phi \mathcal{Q}^H = \mathcal{I}_\Phi,$$

where $\mathcal{I}_\Phi$ is the identity tensor.

Theorem 2.5 (transformed tensor SVD [14, 24]). *Suppose that $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, then $\mathcal{X}$ can be factorized as*

$$\mathcal{X} = \mathcal{U} \diamond_\Phi \mathcal{S} \diamond_\Phi \mathcal{V}^H,$$

where $\mathcal{U} \in \mathbb{C}^{n_1 \times n_1 \times n_3}$, $\mathcal{V} \in \mathbb{C}^{n_2 \times n_2 \times n_3}$ are unitary tensors with respect to Φ-product, and $\mathcal{S} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ is a diagonal tensor (non-zero entries only appear at $(i,i,:)$-entry).

Definition 2.6 (transformed tubal nuclear norm [24]). The transformed tubal nuclear norm (TTNN) of a tensor $\mathcal{X} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$, denoted as $||\mathcal{X}||_{TTNN}$, is the sum of nuclear norms of all frontal slices of $\hat{\mathcal{X}}_\Phi$, that is

$$||\mathcal{X}||_{TTNN} = \sum_{i=1}^{n_3} ||\hat{\mathcal{X}}_{\Phi(i)}||_* = \sum_{i=1}^{n_3} ||\hat{\mathcal{S}}_{\Phi(i)}||_* = \sum_{i=1}^{n_3} sum(diag(\hat{\mathcal{S}}_{\Phi(i)})),$$

where sum operation is the summation of all elements of a vector, and diag extracts the diagonal elements of a matrix.

Song et.al. [24] also defined a transformed multirank function of a tensor $\mathcal{X}$, denoted as $rank_{sum}(\mathcal{X}) = \sum_{i=1}^{n_3} rank(\hat{\mathcal{X}}_{\Phi(i)})$. Lemma 1 in [24] states that $||\mathcal{X}||_{TTNN}$ is the convex envelope of the rank function $rank_{sum}(\mathcal{X})$ on the unit ball $\{\mathcal{X} |\ ||\mathcal{X}||_\Phi \leq 1\}$, where $||\mathcal{X}||_\Phi$ is the spectral norm of the matrix $\bar{\mathcal{X}}_\Phi$. The proof can be found in Appendix A of [24].

2.2 Semi-algebraic Function

Definition 2.7 (semi-algebraic function [13]). Let V be a finite dimensional real vector space and let $\varphi : V \to \mathbb{R}^n$ be a linear isomorphism.

(a) A subset $\mathcal{S} \subset \mathbb{R}^n$ is a semi-algebraic set if and only if there exists a finite number of real polynomial functions P_{ij} and Q_{ij}, satisfying

$$\mathcal{S} = \bigcup_{j=1}^{p} \bigcap_{i=1}^{q} \{x \in \mathbb{R}^n | P_{ij}(x) = 0, Q_{ij}(x) < 0\}.$$

(b) A subset $\mathcal{W} \subset \mathbb{R}^n$ is semi-algebraic if and only if $\varphi(\mathcal{W})$ is a semi-algebraic set in $\mathbb{R}^n$.

(c) A function $f : \mathcal{S} \subset \mathbb{R}^n \to \mathcal{W} \subset \mathbb{R}^m$ is semi-algebraic if and only if the graph $\{(x, f(x)) \in \mathbb{R}^{n+m} | x \in \mathcal{S}\}$ is a semi-algebraic set in $\mathbb{R}^{n+m}$.

(d) A function $f : \mathcal{S} \subset V \to \mathcal{W} \subset \mathbb{R}^m$ is semi-algebraic if and only if the composition $f \circ \varphi^{-1} : \varphi(\mathcal{S}) \to \mathcal{W}$ is semi-algebraic.

Proposition 2.8 (properties for semi-algebraic functions [4, 13, 14, 29]). *The followings are some properties of semi-algebraic functions:*

(a) *A subset of $\mathbb{R}$ is semi-algebraic if it is a finite union of points and open intervals (bounded or unbounded).*

(b) *Complements, finite unions, finite intersections, finite Cartesian products of semi-algebraic sets are semi-algebraic.*

(c) *Finite sums and finite products of semi-algebraic functions are semi-algebraic.*

(d) *The composition $f \circ g$ of semi-algebraic mappings $f : \mathcal{S} \to \mathcal{W}$ and $g : \mathcal{Z} \to \mathcal{S}$ is semi-algebraic.*

(e) *If $f : \mathcal{S} \to \mathcal{W}$ is semi-algebraic and $\mathcal{Z} \subseteq \mathcal{S}$ is a semi-algebraic subset, then $f(\mathcal{Z})$ is semi-algebraic subset.*

(f) *Polynomials defined on $\mathbb{R}^n$ are semi-algebraic.*

(g) *The absolute value function $f(x) = |x|$ is a semi-algebraic function defined on $\mathbb{R}$.*

(h) *If $f_i : \mathcal{S} \to \mathbb{R}$ is semi-algebraic for each $i = 1, ..., n$, then the function $f : x \in \mathcal{S} \to (f_1(x), ..., f_n(x)) \in \mathbb{R}^n$ is also semi-algebraic.*

(i) *The nuclear norm function $f_* : \mathbf{x} \in \mathbb{R}^n \to ||\mathbb{V}^{-1}(\mathbf{x})||_*$ is semi-algebraic, where $\mathbb{V}^{-1}$ defined in Definition 5.1 is a reshape operation from a vector to a matrix.*

It is easy to prove the following Proposition by Definition 2.7 and Proposition 2.8.

Proposition 2.9. *If $f : \mathcal{S} \to \mathcal{W}$ is a semi-algebraic function, and $\mathcal{Z}$ is a semi-algebraic subset of $\mathcal{S}$, then f is also a semi-algebraic function on $\mathcal{Z}$.*

3 The Proposed Model

In this paper, we are given several blurred images $\mathcal{Y} \in \mathbb{R}^{n_v \times n_h \times n_b}$ that are only partially observed ($\mathcal{Y}(\Omega)$ is observed), and we aim to recover the corresponding single sharp image $X \in \mathbb{R}^{n_v \times n_h}$, blurred images $\mathcal{Y}$ and blur kernels $\mathcal{K} \in \mathbb{R}^{(2*k_v+1)\times(2*k_h+1)\times k_b}$. $\mathcal{Y}$ is a tensor and its frontal slices are blurred images; $\mathcal{K}$ is a tensor and similarly, the frontal slices of it are the kernels to blurred images; matrix X is the sharp image. Here, we assume that the kernel is shift-invariant, meaning that one blurred image corresponds to one kernel, and the blurred image is the result of the convolution of a kernel and a sharp image. Therefore, for each $t = 1, ..., n_b$,

$$\mathcal{Y}_t = \mathcal{K}_t \star X,$$

where the operation '$\star$' is the convolution product. For a simpler representation, we can rewrite it into a tensor form,

$$\mathcal{Y} = \mathcal{K} \star X.$$

Then, $||\mathcal{Y} - \mathcal{K} \star X||_F^2$ is used as a data fitting term. Here, we adopt periodic boundary condition and FFTs for fast computation in convolution [21, 7]. Other boundary conditions such as Neumann boundary condition can also be used [21].

Zhou et.al. [35] showed the low rank property of blurred image compared with its sharp image. A blurred image can be regarded as local averaging of a sharp image, thus the values of neighboring pixels tend to have more similarities, resulting in the low rank property. For multiple blurred images (the tensor $\mathcal{Y}$), similar to low rank property of a single blurred image (a matrix), it is also of low tensor rank (we adopt transformed tubal rank in Definition 5). However, the optimization of tubal rank is NP-hard. Lu [20] and Zhang [32] et.al. used TNN as a regularization term for low rank of a tensor. The transformed tubal nuclear norm (TTNN) is adopted in our model because it is the convex envelope of transformed multirank function of a tensor [24] and it is more general than TNN. Therefore, we introduce $||\mathcal{Y}||_{TTNN}$ as a regularization term for the low tubal rank of multiple blurred images.

Blurred images are usually regarded as local averaging of the sharp image and the kernel acts as weights. Based on this, we add two widely accepted constraints [1] to the kernel: $\mathcal{K}_t \geq 0$ and $\sum_{i,j} \mathcal{K}_t(i,j) = 1$. For each kernel (denoted as $\mathcal{K}_t$), we do not expect them to be delta kernel which has one value on its central entry and others are zero. Tikhonov

regularization term [11] is considered for $\mathcal{K}$ to be away from delta kernel. That is, for each $t = 1, ..., n_b$,

$$Tik(\mathcal{K}_t) = ||\mathcal{K}_t||_F^2 = \sum_{i,j} \mathcal{K}_t(i,j)^2 \leq \sum_{i,j} \mathcal{K}_t(i,j) = 1 = ||\delta||_F^2.$$

In the tensor form,

$$Tik(\mathcal{K}) = ||\mathcal{K}||_F^2 = \sum_{t=1}^{T} ||\mathcal{K}_t||_F^2 \leq n_b.$$

If the kernel is close to the delta kernel, the blurred image will be clean enough so that we do not need to deblur. For example, for a Gaussian kernel, if its standard deviation is large, then the kernel should be smooth, leading to small Tikhonov values and more blurry images.

For a clean image, edges are much more clear and sharper than blurred image. Blurred images are usually with more noise and leading to the values of zigzag forms. To avoid the fluctuating values of the image, we introduce the TV regularization term [23] for sharp image X as many other similar deconvolution work. Moreover, we scale the values of image data $\mathcal{Y}$ and X to $[0, 1]$.

Based on the analysis above, our proposed model is as follows:

$$\begin{aligned} &\arg\min_{\mathcal{Y},\mathcal{K},X} \frac{1}{2}||\mathcal{Y} - \mathcal{K} \star X||_F^2 + \alpha||\mathcal{Y}||_{TTNN} + \frac{\beta}{2}||\mathcal{K}||_F^2 + \gamma(||D_1 X||_1 + ||X D_2^T||_1), \\ &s.t. \quad 0 \leq X, \mathcal{Y} \leq 1, \;\; P_\Omega(\mathcal{Y} - \mathcal{M}) = \mathbf{0}, \;\; \mathcal{K} \geq 0, \;\; \sum_{i,j} \mathcal{K}_t(i,j) = 1, \end{aligned} \tag{3.1}$$

where α, β and γ are given positive parameters of regularization terms and $\mathcal{M}$ is the partially observed tensor (blurred images) . Blurred images are given in $\mathcal{Y} \in \mathbb{R}^{n_v \times n_h \times n_b}$, and $X \in \mathbb{R}^{n_v \times n_h}$ is the clean image. P_Ω is a projection operation that preserves the entries of the tensor inside Ω and projects the entries outside Ω to 0, meaning that $\mathcal{Y}(\Omega) = \mathcal{M}(\Omega)$. Here, '$\star$' is the convolution product with periodic boundary condition. $\mathcal{K} \in \mathbb{R}^{(2*k_v+1)\times(2*k_h+1)\times k_b}$ denotes the kernels, and $k_b = n_b$ because each blurred image corresponds to one kernel. $(i, j, t) \in \{-k_v, ..., k_v\} \times \{-k_h, ..., k_h\} \times \{1, ..., n_b\}$ is the support of the kernel. The fourth term is the TV regularization term, where $D_1 \in \mathbb{R}^{n_v \times n_v}$ and $D_2 \in \mathbb{R}^{n_h \times n_h}$ denote backward partial difference operators with periodic boundary conditions along vertical and horizontal directions, respectively. More specifically, the periodic boundary condition for sharp image X is that

$$X(i,j) = X(i, j + n_h) = X(i + n_v, j) = X(i + n_v, j + n_h).$$

Therefore, the backward partial difference operators D_1 and D_2 are as follows:

$$D_1 = \begin{pmatrix} 1 & & & -1 \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix}_{n_v \times n_v}, \qquad D_2 = \begin{pmatrix} 1 & & & -1 \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix}_{n_h \times n_h}.$$

For the implementation, we rewrite the problem (3.1):

$$\arg\min_{\mathcal{Y},\mathcal{K},X} F(\mathcal{Y}, \mathcal{K}, X), \tag{3.2}$$

where

$$F(\mathcal{Y},\mathcal{K},X)=\frac{1}{2}||\mathcal{Y}-\mathcal{K}\star X||_F^2+\alpha||\mathcal{Y}||_{TTNN}+\frac{\beta}{2}||\mathcal{K}||_F^2+\delta_{\mathfrak{S}}(\mathcal{Y},\mathcal{K},X)$$

$$+\gamma(||D_1X||_1+||XD_2^T||_1),$$

$$\mathfrak{S}=\{(\mathcal{Y},\mathcal{K},X)\in\mathbb{R}^{n_v\times n_h\times n_b}\times\mathbb{R}^{(2*k_v+1)\times(2*k_h+1)\times k_b}\times\mathbb{R}^{n_v\times n_h}:0\le\mathcal{Y},X\le 1,$$

$$P_\Omega(\mathcal{Y}-\mathcal{M})=\mathbf{0},\ \ \mathcal{K}\ge 0,\ \sum_{i=-k_v}^{k_v}\sum_{j=-k_h}^{k_h}\mathcal{K}_t(i,j)=1,\ \ for\ \ each\ \ t=1,..,k_b\}$$

and $\delta_{\mathfrak{S}}$ is an indicator function defined on the whole space that

$$\delta_{\mathfrak{S}}(\mathcal{Y},\mathcal{K},X)=\begin{cases}0, & \text{if } (\mathcal{Y},\mathcal{K},X)\in\mathfrak{S},\\ +\infty, & \text{otherwise.}\end{cases}$$

4 PAM Iteration and the Gauss-Seidel based ADMM

It is difficult to obtain direct solutions to Problem (3.2). Instead, we adopt an inexact iterative method: Proximal Alternating Minimization (PAM) [3] to solve the problem. PAM applied to the problem at hand involves iteratively solving three subproblems, and that each of these subproblems is solved by an iterative scheme based on ADMM. The PAM iterations are indexed by i while the ADMM iterations are indexed by j.

At the $(i+1)$-th iteration step, the initial value of $(\mathcal{Y},\mathcal{K},X)$ is the solution from the i-th step:

$$\mathcal{Y}^{i+1}=\arg\min_{\mathcal{Y}}F(\mathcal{Y},\mathcal{K}^i,X^i)+\frac{\rho_1}{2}||\mathcal{Y}-\mathcal{Y}^i||_F^2, \tag{4.1}$$

$$\mathcal{K}^{i+1}=\arg\min_{\mathcal{K}}F(\mathcal{Y}^{i+1},\mathcal{K},X^i)+\frac{\rho_2}{2}||\mathcal{K}-\mathcal{K}^i||_F^2, \tag{4.2}$$

$$X^{i+1}=\arg\min_{X}F(\mathcal{Y}^{i+1},\mathcal{K}^{i+1},X)+\frac{\rho_3}{2}||X-X^i||_F^2, \tag{4.3}$$

where ρ_i $(i=1,2,3)$ are given positive parameters to constrain large changes of $(\mathcal{Y},\mathcal{K},X)$ between two steps.

In the following part, we consider numerical methods to solve subproblems (4.1)-(4.3) within PAM Iteration. Fast Fourier Transforms (FFTs) are used to reduce computational costs.

4.1 Solving $\mathcal{Y}$

For the subproblem (4.1) of $\mathcal{Y}$,

$$\begin{aligned}\mathcal{Y}^{i+1}&=\arg\min_{\mathcal{Y}}F(\mathcal{Y},\mathcal{K}^i,X^i)+\frac{\rho_1}{2}||\mathcal{Y}-\mathcal{Y}^i||_F^2\\&=\arg\min_{0\le\mathcal{Y}\le 1,P_\Omega(\mathcal{Y}-\mathcal{M})=0}\frac{1}{2}||\mathcal{Y}-\mathcal{K}^i\star X^i||_F^2+\alpha||\mathcal{Y}||_{TTNN}+\frac{\rho_1}{2}||\mathcal{Y}-\mathcal{Y}^i||_F^2\end{aligned}$$

is equivalent to

$$\arg\min_{0\le\mathcal{Y}\le 1,P_\Omega(\mathcal{Y}-\mathcal{M})=0}\frac{1}{2}||\mathcal{Y}-\mathcal{K}^i\star X^i||_F^2+\alpha||\mathcal{Q}_{\mathcal{Y}}||_{TTNN}+\frac{\rho_1}{2}||\mathcal{Y}-\mathcal{Y}^i||_F^2,$$

$$s.t. \quad \mathcal{Q}_{\mathcal{Y}} = \mathcal{Y}. \tag{4.4}$$

Let $\mathcal{Z}_{\mathcal{Y}}$ be the Lagrange multiplier for the Lagrangian function of the constrained optimization problem (4.4). The augmented Lagrangian function is

$$L_{\mathcal{Y}} = \frac{1}{2}||\mathcal{Y} - \mathcal{K}^i \star X^i||_F^2 + \alpha||\mathcal{Q}_{\mathcal{Y}}||_{TTNN} + \frac{\rho_1}{2}||\mathcal{Y} - \mathcal{Y}^i||_F^2$$

$$+\langle \mathcal{Z}_{\mathcal{Y}}, \mathcal{Y} - \mathcal{Q}_{\mathcal{Y}} \rangle + \frac{\mu_1}{2}||\mathcal{Y} - \mathcal{Q}_{\mathcal{Y}}||_F^2.$$

Then, ADMM iteration steps for (4.4) are as follows:

$$\mathcal{Q}_{\mathcal{Y}}^{j+1} = \arg\min_{\mathcal{Q}_{\mathcal{Y}}} \ \alpha||\mathcal{Q}_{\mathcal{Y}}||_{TTNN} + \frac{\mu_1}{2}||\mathcal{Y}^{i,j} - \mathcal{Q}_{\mathcal{Y}} + \frac{\mathcal{Z}_{\mathcal{Y}}^j}{\mu_1}||_F^2, \tag{4.5}$$

$$\mathcal{Y}^{i,j+1} = \arg\min_{0 \le \mathcal{Y} \le 1, P_\Omega(\mathcal{Y}-\mathcal{M})=0} \ \frac{1}{2}||\mathcal{Y} - \mathcal{K}^i \star X^i||_F^2 + \frac{\rho_1}{2}||\mathcal{Y} - \mathcal{Y}^i||_F^2 + \frac{\mu_1}{2}||\mathcal{Y} - \mathcal{Q}_{\mathcal{Y}}^{j+1} + \frac{\mathcal{Z}_{\mathcal{Y}}^j}{\mu_1}||_F^2, \tag{4.6}$$

$$\mathcal{Z}_{\mathcal{Y}}^{j+1} = \mathcal{Z}_{\mathcal{Y}}^j + \mu_1(\mathcal{Y}^{i,j+1} - \mathcal{Q}_{\mathcal{Y}}^{j+1}). \tag{4.7}$$

Initial values $\mathcal{Y}^{i,0}$ and $\mathcal{Z}_{\mathcal{Y}}^0$ are set as $\mathcal{Y}^i$ and $\mathbf{0}$ respectively; μ_1 is a given parameter. After repeating κ_1 times, $\mathcal{Y}^{i,\kappa_1}$ is the approximate solution of the subproblem, denoted as $\mathcal{Y}^{i+1}$. To solve (4.5), theorem 3 in [24] provides a method using transformed tensor SVD (2.5) for third-order tensors. For more details, please refer to theorem 1, theorem 3 and algorithm 1 in [24]. In this step, the major computational cost is from SVDs of $n_v \times n_h$ matrix for n_b times.

For (4.6), it's actually a single variable component-wise minimization problem, which can be easily solved with little computational costs. (4.7) is a gradient ascent procedure, which just involves elementary computation of tensors, and thus the computation is cheap. In summary, the main computational cost for solving $\mathcal{Y}$ is transformed tensor SVD, which means that we need to do SVDs of $n_v \times n_h$ matrix for n_b times.

4.2 Solving $\mathcal{K}$

For subproblem (4.2) of $\mathcal{K}$,

$$\begin{aligned}\mathcal{K}^{i+1} &= \arg\min_{\mathcal{K}} F(\mathcal{Y}^{i+1}, \mathcal{K}, X^i) + \frac{\rho_2}{2}||\mathcal{K} - \mathcal{K}^i||_F^2 \\ &= \arg\min_{\mathcal{K} \ge 0, \mathbf{1}^T\mathcal{K}_t(:)=1} \frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{K} \star X^i||_F^2 + \frac{\beta}{2}||\mathcal{K}||_F^2 + \frac{\rho_2}{2}||\mathcal{K} - \mathcal{K}^i||_F^2\end{aligned}$$

is equivalent to

$$\arg\min_{\mathcal{K} \ge 0, \mathbf{1}^T\mathcal{K}_t(:)=1} \frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{Q}_{\mathcal{K}} \star X^i||_F^2 + \frac{\beta}{2}||\mathcal{K}||_F^2 + \frac{\rho_2}{2}||\mathcal{K} - \mathcal{K}^i||_F^2,$$

$$s.t. \quad \mathcal{Q}_{\mathcal{K}} = \mathcal{K}. \tag{4.8}$$

Let $\mathcal{Z}_{\mathcal{K}}$ be the Lagrangian multiplier for the Lagrangian function of the constrained optimization problem (4.8). The augmented Lagrangian function is

$$L_{\mathcal{K}} = \frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{Q}_{\mathcal{K}} \star X^i||_F^2 + \frac{\beta}{2}||\mathcal{K}||_F^2 + \frac{\rho_2}{2}||\mathcal{K} - \mathcal{K}^i||_F^2$$

$$+\langle \mathcal{Z}_{\mathcal{Y}}, \mathcal{K} - \mathcal{Q}_{\mathcal{K}} \rangle + \frac{\mu_2}{2} ||\mathcal{K} - \mathcal{Q}_{\mathcal{K}}||_F^2.$$

Then, the ADMM iteration steps for (4.8) is as follows:

$$\mathcal{Q}_{\mathcal{K}}^{j+1} = \arg\min_{\mathcal{Q}_{\mathcal{K}}} \ \frac{1}{2} ||\mathcal{Y}^{i+1} - \mathcal{Q}_{\mathcal{K}} \star X^i||_F^2 + \frac{\mu_2}{2} ||\mathcal{K}^{i,j} - \mathcal{Q}_{\mathcal{K}} + \frac{\mathcal{Z}_{\mathcal{K}}^j}{\mu_2}||_F^2, \tag{4.9}$$

$$\mathcal{K}^{i,j+1} = \arg\min_{\mathcal{K} \geq 0, \mathbf{1}^T \mathcal{K}_t(:) = 1} \ \frac{\beta}{2} ||\mathcal{K}||_F^2 + \frac{\rho_2}{2} ||\mathcal{K} - \mathcal{K}^i||_F^2 + \frac{\mu_2}{2} ||\mathcal{K} - \mathcal{Q}_{\mathcal{K}}^{j+1} + \frac{\mathcal{Z}_{\mathcal{K}}^j}{\mu_2}||_F^2, \tag{4.10}$$

$$\mathcal{Z}_{\mathcal{K}}^{j+1} = \mathcal{Z}_{\mathcal{K}}^j + \mu_2 (\mathcal{K}^{i,j+1} - \mathcal{Q}_{\mathcal{K}}^{j+1}). \tag{4.11}$$

Initial values $\mathcal{K}^{i,0}$ and $\mathcal{Z}_{\mathcal{K}}^0$ are set as $\mathcal{K}^i$ and $\mathbf{0}$ respectively; μ_2 is a given parameter. After κ_2 times, $\mathcal{K}^{i,\kappa_2}$ is taken as the output of the subproblem, denoted as $\mathcal{K}^{i+1}$. For (4.9), it's easy to find that, for each slice, each element of $\mathcal{Q}_{\mathcal{K}_t} \star X$ is a linear combination of elements of $\mathcal{Q}_{\mathcal{K}_t}$, meaning that

$$[\mathcal{Q}_{\mathcal{K}_t} \star X](:) = A \mathcal{Q}_{\mathcal{K}_t}(:),$$

where $\mathcal{Q}_{\mathcal{K}_t}$ is the t-th frontal slice of tensor $\mathcal{Q}_{\mathcal{K}}$ and A is a $(n_v * n_h) \times ((2*k_v+1)*(2*k_h+1))$ matrix, depending on X. The matrix A can be cheaply calculated in the Fourier domain that positions of $\mathcal{K}$ and X in the convolution can be exchanged. After obtaining the matrix A, letting the first derivative of the objective function with respect to $\mathcal{Q}_{\mathcal{Y}}(:)$ to be zero, where the objective function has the similar form with a simple linear problem $\arg\min_{\mathbf{x}} ||A\mathbf{x} - \mathbf{b}||_2^2 + ||\mathbf{x} - \mathbf{y}||_2^2$. Then the problem transfers to an equivalent linear system of $\mathcal{Q}_{\mathcal{Y}}(:)$. The scale of the linear system is only $((2 * k_v + 1) * (2 * k_h + 1)) \times ((2 * k_v + 1) * (2 * k_h + 1))$, which is much smaller compared to $(n_v * n_h)$. Hence, it is still very fast to directly solve the $((2 * k_v + 1) * (2 * k_h + 1)) \times ((2 * k_v + 1) * (2 * k_h + 1))$ linear system by Gaussian elimination or Cholesky decomposition (the coefficient matrix is symmetric and positive semi-definite). And we need to solve this kind of linear systems for k_b times. To solve (4.10), we can simplify it to

$$\begin{aligned} \mathcal{K}^{i,j+1} &= \arg\min_{\mathcal{K} \geq 0, \mathbf{1}^T \mathcal{K}_t(:) = 1} \ \frac{\beta}{2} ||\mathcal{K}||_F^2 + \frac{\rho_2}{2} ||\mathcal{K} - \mathcal{K}^i||_F^2 + \frac{\mu_2}{2} ||\mathcal{K} - \mathcal{Q}_{\mathcal{K}}^{j+1} + \frac{\mathcal{Z}_{\mathcal{K}}^j}{\mu_2}||_F^2 \\ &= \arg\min_{\mathcal{K} \geq 0, \mathbf{1}^T \mathcal{K}_t(:) = 1} \ ||\mathcal{K} - \frac{1}{\beta + \rho_2 + \mu_2} (\rho_2 \mathcal{K}^i + \mu_2 \mathcal{Q}_{\mathcal{K}}^{j+1} - \mathcal{Z}_{\mathcal{K}}^j)||_F^2 \\ &= \arg\min_{\mathcal{K} \geq 0, \mathbf{1}^T \mathcal{K}_t(:) = 1} \ ||\mathcal{K} - \mathcal{W}||_F^2, \end{aligned} \tag{4.12}$$

where $\mathcal{W} = \frac{1}{\beta + \rho_2 + \mu_2} (\rho_2 \mathcal{K}^i + \mu_2 \mathcal{Q}_{\mathcal{K}}^{j+1} - \mathcal{Z}_{\mathcal{K}}^j)$. [27] provides a fast and direct algorithm for solving (4.12), which requires a sorting of each slice of $\mathcal{W}$. And this step is cheap for only $O(D * log(D) * k_b)$, where $D = (2 * k_v + 1) * (2 * k_h + 1)$. (4.11) is merely an element-wise computation of a $(2 * k_v + 1) \times (2 * k_h + 1) \times k_b$ tensor. In summary, the dominant computational cost is solving $((2 * k_v + 1) * (2 * k_h + 1)) \times ((2 * k_v + 1) * (2 * k_h + 1))$ linear system for k_b times.

4.3 Solving X

For subproblem (4.3) of X,

$$\begin{aligned} X^{i+1} &= \arg\min_{X} F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X) + \frac{\rho_3}{2}||X - X^i||_F^2 \\ &= \arg\min_{0\le X\le 1} \frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{K}^{i+1} \star X||_F^2 + \gamma(||D_1 X||_1 + ||X D_2^T||_1) \\ &\qquad + \frac{\rho_3}{2}||X - X^i||_F^2 \end{aligned}$$

is equivalent to

$$\arg\min_{0\le X\le 1} \frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{K}^{i+1} \star Q_X||_F^2 + \gamma(||Q_1||_1 + ||Q_2||_1) + \frac{\rho_3}{2}||X - X^i||_F^2,$$

$$s.t. \quad Q_X = X,\ Q_1 = D_1 Q_X,\ Q_2 = Q_X D_2^T,\ -1 \le Q_1, Q_2 \le 1. \tag{4.13}$$

Let Z_X, Z_1 and Z_2 be the Lagrangian multiplier for the Lagrangian function of the constrained optimization problem (4.13). The augmented Lagrangian function is

$$\begin{aligned} L_X = &\frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{K}^{i+1} \star Q_X||_F^2 + \gamma(||Q_1||_{l_1} + ||Q_2||_{l_1}) + \frac{\rho_3}{2}||X - X^i||_F^2 \\ &+\langle Z_X, X - Q_X\rangle + \langle Z_1, D_1 Q_X - Q_1\rangle + \langle Z_2, Q_X D_2^T - Q_2\rangle \\ &+\frac{\mu_3}{2}||X - Q_X||_F^2 + \frac{\mu_3}{2}||D_1 Q_X - Q_1||_F^2 + \frac{\mu_3}{2}||Q_X D_2^T - Q_2||_F^2. \end{aligned}$$

Then, the ADMM iteration steps for (4.13) is as follows:

$$Q_1^{j+1} = \arg\min_{-1\le Q_1\le 1} \gamma||Q_1||_{l_1} + \frac{\mu_3}{2}||D_1 Q_X^j - Q_1 + \frac{Z_1^j}{\mu_3}||_F^2, \tag{4.14}$$

$$Q_2^{j+1} = \arg\min_{-1\le Q_2\le 1} \gamma||Q_2||_{l_1} + \frac{\mu_3}{2}||Q_X^j D_2^T - Q_2 + \frac{Z_2^j}{\mu_3}||_F^2, \tag{4.15}$$

$$\begin{aligned} Q_X^{j+1} = \arg\min_{Q_X} &\frac{1}{2}||\mathcal{Y}^{i+1} - \mathcal{K}^{i+1} \star Q_X||_F^2 + \frac{\mu_3}{2}\Big[||X^{i,j} - Q_X + \frac{Z_X^j}{\mu_3}||_F^2 \\ &+ ||D_1 Q_X - Q_1^{j+1} + \frac{Z_1^j}{\mu_3}||_F^2 + ||Q_X D_2^T - Q_2^{j+1} + \frac{Z_2^j}{\mu_3}||_F^2\Big], \end{aligned} \tag{4.16}$$

$$X^{i,j+1} = \arg\min_{0\le X\le 1} \frac{\rho_3}{2}||X - X^i||_F^2 + \frac{\mu_3}{2}||X - Q_X^{j+1} + \frac{Z_X^j}{\mu_3}||_F^2, \tag{4.17}$$

$$Z_1^{j+1} = Z_1^j + \mu_3(D_1 Q_X^{j+1} - Q_1^{j+1}), \tag{4.18}$$

$$Z_2^{j+1} = Z_2^j + \mu_3(Q_X^{j+1} D_2^T - Q_2^{j+1}), \tag{4.19}$$

$$Z_X^{j+1} = Z_X^j + \mu_3(X^{i,j+1} - Q_X^{j+1}). \tag{4.20}$$

Initial values $X^{i,0}$ and Q_X are both set to X^i; Z_1^0, Z_2^0, and Z_X^0 are all set as $\mathbf{0}$; μ_3 is a given parameter. After repeating the iteration for κ_3 times, X^{i,κ_3} is close to the real solution of the subproblem, and thus outputting it as X^{j+1}. Problem (4.14) is the element-wise minimization problem of the form $\arg\min_{-1\le x\le 1} |x| + \lambda(x-y)^2$, which can be easily solved

and the solution is $x^* = \min\{\max\{\bar{x}, -1\}, 1\}$, where $\bar{x} = sign(y)\max\{|y| - \frac{1}{2\lambda}, 0\}$. (4.15) is the similar to (4.14), and thus can be solved by the same way. In the Fourier domain, (4.16) becomes an element-wise quadratic problem without any constraints, which can be easily solved and inverse FFTs can return solution from the Fourier domain back to real space. (4.17) is actually a component-wise minimization for each element of the matrix X. (4.18)-(4.20) are just simple matrix multiplication and elementary computation. Noting that D_1 and D_2 are circulant matrix, so for faster computation, $D_1 Q_X$ and $Q_X D_2^T$ are calculated in Fourier domain, avoiding loop operation.

In the next section, we will discuss about the convergence of inexact solutions of subproblems (4.1)-(4.3). Inexact solutions of each subproblem of PAM iteration converge to a critical point of the objective function F. Although F is convex separately in $\mathcal{Y}$, $\mathcal{K}$ and X, it's nonconvex jointly for those 3 variables. It's hard to achieve global minima in nonconvex optimization and convergence to critical points is within our expectation. Repeating κ_1, κ_2 and κ_3 times respectively in the above ADMM iteration processes will definitely lead to inexact solutions with errors to (4.1)-(4.3). That's the reason for Assumption 5.2 in the convergence analysis.

5 Convergence of PAM Iteration with Inexact Solutions

For convergence analysis, we first rewrite the PAM Iteration (4.1)-(4.3) as a vector-form.

Definition 5.1 (some projections of tensors and vectors)**.** For a positive integer n, define $\chi(n) = \{1, ..., n\}$,

1. For positive integers n_1, n_2 and n_3, define bijections $\mathbb{V}[n_1, n_2, n_3] : \mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \to \mathcal{X}(:) \in \mathbb{R}^{n_1 n_2 n_3 \times 1}$ and $\mathbb{P}[n_1, n_2, n_3] : (i, j, t) \in \chi(n_1) \times \chi(n_2) \times \chi(n_3) \to \chi(n_1 n_2 n_3)$ satisfying that $\mathcal{X}(i, j, t) = [\mathbb{V}(\mathcal{X})](\mathbb{P}(i, j, t))$. $\mathbb{V}$ projects a tensor to its corresponding vector, and $\mathbb{P}$ project entries of the tensor to the vector.

2. The inverse projection $\mathbb{V}^{-1}$ and $\mathbb{P}^{-1}$ project a vector to its corresponding tensor.

We denote $\mathbb{V}_1 = \mathbb{V}[n_v, n_h, n_b]$ and $\mathbb{P}_1 = \mathbb{P}[n_v, n_h, n_b]$ for tensor $\mathcal{Y} \in \mathbb{R}^{n_v \times n_h \times n_b}$ that

$$\mathbb{V}_1(\mathcal{Y}) = \mathcal{Y}(:)$$

and

$$\mathcal{Y}(i, j, t) = [\mathcal{Y}(:)](\mathbb{P}_1(i, j, t)).$$

Similarly, we define $\mathbb{V}_2 = \mathbb{V}[(2 * k_v + 1), (2 * k_h + 1), k_b]$ and $\mathbb{P}_2 = \mathbb{P}[(2 * k_v + 1), (2 * k_h + 1), k_b]$ for kernels $\mathcal{K} \in \mathbb{R}^{(2*k_v+1)\times(2*k_h+1)\times k_b}$; and projection function $\mathbb{V}_3 = \mathbb{V}[n_v, n_h]$ and $\mathbb{P}_3 = \mathbb{P}[n_v, n_h]$ for matrix $X \in \mathbb{R}^{n_v \times n_h}$. Let $\mathbb{V}_1(\mathcal{Y}) = \mathbf{y}$, $\mathbb{V}_2(\mathcal{K}) = \mathbf{k}$ and $\mathbb{V}_3(X) = \mathbf{x}$; and $N_1 = n_v * n_h * n_b$, $N_2 = (2 * k_v + 1) * (2 * k_h + 1) * k_b$, $N_3 = n_v * n_h$.

Using these projection functions and corresponding inverse functions, we can rewrite the problem (3.2) to a model of vectors.

$$\arg\min_{\mathbf{v}} \tilde{F}(\mathbf{v}), \tag{5.1}$$

where $\mathbf{v} = (\mathbf{y}, \mathbf{k}, \mathbf{x})$ and

$$\begin{aligned}
&\tilde{F}(\mathbf{v}) = \Psi(\mathbf{v}) + g_1(\mathbf{y}) + g_3(\mathbf{x}) + \delta_{\hat{\mathfrak{S}}}(\mathbf{v}),\\
&\Psi(\mathbf{v}) = \frac{1}{2}||\mathbb{V}_1^{-1}(\mathbf{y}) - \mathbb{V}_2^{-1}(\mathbf{k}) \star \mathbb{V}_3^{-1}(\mathbf{x})||_F^2 + \frac{\beta}{2}||\mathbf{k}||_2^2,\\
&g_1(\mathbf{y}) = \alpha||\mathbb{V}_1^{-1}(\mathbf{y})||_{TTNN},\\
&g_3(\mathbf{x}) = \gamma(||h_1(\mathbf{x})||_1 + ||h_2(\mathbf{x})||_1),\\
&h_1(\mathbf{x}) = D_1(\mathbb{V}_3^{-1}(\mathbf{x})),\\
&h_2(\mathbf{x}) = (\mathbb{V}_3^{-1}(\mathbf{x}))D_2^T,\\
&\hat{\mathfrak{S}} = \{(\mathbf{y}, \mathbf{k}, \mathbf{x}) \in \mathbb{R}^{N_1 \times N_2 \times N_3} : \ \mathbf{y} \in \mathbb{R}^{N_1}, \ \mathbf{y}(\mathbb{P}_1(\Omega)) = [\mathbb{V}_1(\mathcal{M})](\mathbb{P}_1(\Omega)), \ 0 \le \mathbf{y} \le 1,\\
&\qquad \mathbf{k} \in \mathbb{R}^{N_2}, \ \mathbf{k} \ge 0, \ \mathbf{1}^T\mathbf{k}_t = 1, \ \mathbf{k} = [\mathbf{k}_1, ..., \mathbf{k}_{k_b}], \ \mathbf{x} \in \mathbb{R}^{N_3}, \ 0 \le \mathbf{x} \le 1\}.
\end{aligned}$$

It's easy to verify that the problem (3.2) is equivalent to problem (5.1). Therefore, in the convergence analysis, we mainly focus on the vectorized problem (5.1). And the PAM Iteration procedure (4.1)-(4.3) can also be rewritten in form of vectors:

$$\mathbf{y}^{i+1} = \arg\min_{\mathbf{y}} \tilde{F}(\mathbf{y}, \mathbf{k}^i, \mathbf{x}^i) + \frac{\rho_1}{2}||\mathbf{y} - \mathbf{y}^i||_2^2, \tag{5.2}$$

$$\mathbf{k}^{i+1} = \arg\min_{\mathbf{k}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}, \mathbf{x}^i) + \frac{\rho_2}{2}||\mathbf{k} - \mathbf{k}^i||_2^2, \tag{5.3}$$

$$\mathbf{x}^{i+1} = \arg\min_{\mathbf{x}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}) + \frac{\rho_3}{2}||\mathbf{x} - \mathbf{x}^i||_2^2. \tag{5.4}$$

It's not difficult to verify the following Proposition by the Definition of PLSC function in [17].

Proposition 5.2. *$\tilde{F}(\mathbf{v})$ is a Proper Lower Semi-Continuous (PLSC) function.*

A necessary condition [3] for x to be the minimizer of a PLSC function is that

$$0 \in \partial f(x),$$

and a point satisfying the above condition is called the critical point of f, where ∂ is sub-differential operation.

We first make some assumptions for inexact solutions. If $(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1})$ is the solution to subproblems (5.2)-(5.4), then

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \frac{\rho_1}{2}||\mathbf{y}^{i+1} - \mathbf{y}^i||_2^2 \le \tilde{F}(\mathbf{y}^i, \mathbf{k}^i, \mathbf{x}^i), \tag{5.5}$$

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \frac{\rho_2}{2}||\mathbf{k}^{i+1} - \mathbf{k}^i||_2^2 \le \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i), \tag{5.6}$$

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \frac{\rho_3}{2}||\mathbf{x}^{i+1} - \mathbf{x}^i||_2^2 \le \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i), \tag{5.7}$$

and

$$0 \in \partial_{\mathbf{y}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i), \tag{5.8}$$

$$0 \in \partial_{\mathbf{k}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i), \tag{5.9}$$

$$0 \in \partial_{\mathbf{x}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i), \tag{5.10}$$

where ∂ is sub-differential. In real applications, our obtained solutions of (5.2)-(5.4) usually do not satisfy conditions (5.5)-(5.10), i.e., our solutions may have errors with respect to exact solutions, due to the fact that we solve subproblems inexactly by ADMM algorithm. So we further assume that the solutions satisfy the following conditions.

Assumption 5.1. There exists non-negative constants $\epsilon_1 \in (0, \rho_1)$, $\epsilon_2 \in (0, \rho_2)$, $\epsilon_3 \in (0, \rho_3)$, and b_1, b_2, b_3, such that for each iteration,

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \frac{\rho_1}{2}||\mathbf{y}^{i+1} - \mathbf{y}^i||_2^2 - \tilde{F}(\mathbf{y}^i, \mathbf{k}^i, \mathbf{x}^i) \leq \frac{\epsilon_1}{2}||\mathbf{y}^{i+1} - \mathbf{y}^i||_2^2, \tag{5.11}$$

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \frac{\rho_2}{2}||\mathbf{k}^{i+1} - \mathbf{k}^i||_2^2 - \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) \leq \frac{\epsilon_2}{2}||\mathbf{k}^{i+1} - \mathbf{k}^i||_2^2, \tag{5.12}$$

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \frac{\rho_3}{2}||\mathbf{x}^{i+1} - \mathbf{x}^i||_2^2 - \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) \leq \frac{\epsilon_3}{2}||\mathbf{x}^{i+1} - \mathbf{x}^i||_2^2, \tag{5.13}$$

and

$$\exists\ \xi_1^{i+1} \in \partial_{\mathbf{y}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i), \tag{5.14}$$

$$\exists\ \xi_2^{i+1} \in \partial_{\mathbf{k}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i), \tag{5.15}$$

$$\exists\ \xi_3^{i+1} \in \partial_{\mathbf{x}} \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i), \tag{5.16}$$

with

$$||\xi_1^{i+1}||_2 \leq b_1||\mathbf{y}^{i+1} - \mathbf{y}^i||_2, \tag{5.17}$$

$$||\xi_2^{i+1}||_2 \leq b_2||\mathbf{k}^{i+1} - \mathbf{k}^i||_2, \tag{5.18}$$

$$||\xi_3^{i+1}||_2 \leq b_3||\mathbf{x}^{i+1} - \mathbf{x}^i||_2. \tag{5.19}$$

In the remaining part, we give a proof that if our obtained inexact solutions of PAM Iteration (5.2)-(5.4) satisfy the conditions (5.11)-(5.19), our solutions will always converge to a critical point. However, we cannot guarantee the global minimum because of the non-convexity of the object function.

Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a PLSC function, and for $-\infty < \eta_1 < \eta_2 < +\infty$, we denote $[\eta_1 < f < \eta_2] = \{x \in \mathbb{R}^n | \eta_1 < f(x) < \eta_2\}$.

Proposition 5.3 ([5, 6])**.** *A real valued semi-algebraic function is a Kurdyka-Lojasiewicz (KL) function, i.e. it satisfies KL property [3] at each point in its domain.*

Lemma 5.4. *$\tilde{F}$ is semi-algebraic on $\hat{\mathfrak{S}}$. Combined with Proposition 5.3, $\tilde{F}$ has KL property at each point $\mathbf{v} \in \hat{\mathfrak{S}}$.*

Proof. The main idea of the proof is that we first prove that the set $\hat{\mathfrak{S}}$ is semi-algebraic, and the objective function $\tilde{F}$ is semi-algebraic on $\mathbb{R}^N$. By Proposition 2.9, we complete the proof. Here are the details of the proof.

From (5.1),

$$\tilde{F}(\mathbf{v}) = \Psi(\mathbf{v}) + g_1(\mathbf{y}) + g_3(\mathbf{x}) + \delta_{\hat{\mathfrak{S}}}(\mathbf{v}),$$

where $\mathbf{v} = (\mathbf{y}, \mathbf{k}, \mathbf{x})$. And

$$\hat{\mathfrak{S}} = \hat{\mathfrak{S}}_1 \times \hat{\mathfrak{S}}_2 \times \hat{\mathfrak{S}}_3,$$

where

$$\hat{\mathfrak{S}}_1 = \{\mathbf{y} \in \mathbb{R}^{N_1} |\ 0 \leq \mathbf{y} \leq 1,\ \mathbf{y}(\mathbb{P}_1(\Omega)) = [\mathbb{V}_1(\mathcal{M})](\mathbb{P}_1(\Omega))\}, \tag{5.20}$$

$$\hat{\mathfrak{S}}_2 = \{\mathbf{k} \in \mathbb{R}^{N_2} |\ \mathbf{k} \geq 0,\ \mathbf{1}^T\mathbf{k}_t = 1,\ \mathbf{k} = [\mathbf{k}_1, ..., \mathbf{k}_{k_b}]\}, \tag{5.21}$$

$$\hat{\mathfrak{S}}_3 = \{\mathbf{x} \in \mathbb{R}^{N_3} |\ 0 \leq \mathbf{x} \leq 1\}. \tag{5.22}$$

$\hat{\mathfrak{S}}_1$ is the Cartesian products of points and intervals $[0,1]$, where interval $[0,1]$ can also be regarded as the union of open interval $(0,1)$ and points $\{0,1\}$. Hence, $\hat{\mathfrak{S}}_1$ is semi-algebraic subset of $\mathbb{R}^{N_1}$. $\hat{\mathfrak{S}}_3$ is the Cartesian products of intervals $[0,1]$. Hence, $\hat{\mathfrak{S}}_3$ is a semi-algebraic subset of $\mathbb{R}^{N_3}$.

We can rewrite the

$$\hat{\mathfrak{S}}_2 = \hat{\mathfrak{S}}_{2,1} \times \hat{\mathfrak{S}}_{2,2} \times ... \times \hat{\mathfrak{S}}_{2,k_b},$$

where $\hat{\mathfrak{S}}_{2,t} = \{\mathbf{k}_t \in \mathbb{R}^{N_{2,t}} | \mathbf{k}_t \geq 0, \mathbf{1}^T\mathbf{k}_t = 1\}$ and $N_{2,t} = N_2/k_b = (2 * k_v + 1) * (2 * k_h + 1)$. Denote an operation $\mathfrak{I}(\mathbf{k}_t) = \mathbf{1}^T\mathbf{k}_t$, and it is a polynomial of $\mathbf{k}_t \in \mathbb{R}^{N_{2,t}}$. Thus from the Proposition 2.8 (**f**), the operation $\mathfrak{I}$ is semi-algebraic on $\mathbb{R}^{N_{2,t}}$. According to Definition 2.7 (**c**), $\{(\mathbf{k}_t, \mathfrak{I}(\mathbf{k}_t)) | \mathbf{k}_t \in \mathbb{R}^{N_{2,t}}\}$ is semi-algebraic. Also, it's easy to verify that the set $\mathbb{R}_+^{N_{2,t}} \times \{1\}$ is semi-algebraic by Proposition 2.8 (**a**), (**b**). Denote $\bar{\mathfrak{S}}_{2,t} = \{(\mathbf{k}_t, \mathfrak{I}(\mathbf{k}_t)) | \mathbf{k}_t \in \mathbb{R}^{N_{2,t}}\} \cap (\mathbb{R}_+^{N_{2,t}} \times \{1\})$, and it is semi-algebraic. Let

$$\mathfrak{T}_{n+1}^n(\mathfrak{S}) = \{(x_1, ..., x_n) | x = (x_1, ..., x_n, x_{n+1}) \in \mathfrak{S}\}.$$

Note that, for each t, by Tarski-Seidenberg Theorem (Theorem 2.2 in [4]),

$$\hat{\mathfrak{S}}_{2,t} = \mathfrak{T}_{n+1}^n(\bar{\mathfrak{S}}_2)$$

is semi-algebraic, meaning that $\hat{\mathfrak{S}}_2$ is semi-algebraic by Proposition 2.8 (**b**). Therefore, $\hat{\mathfrak{S}}$ is a semi-algebraic subset of $\mathbb{R}^N$.

For the objective function $\tilde{F} = \Psi(\mathbf{v}) + g_1(\mathbf{y}) + g_3(\mathbf{x}) + \delta_{\hat{\mathfrak{S}}}(\mathbf{v})$. $\mathbb{V}_i^{-1}$ $(i = 1, 2, 3)$ are linear mappings from a vector to a tensor, and convolution '$\star$' is a linear operator, thus $\Psi(\mathbf{v}) = \frac{1}{2}||\mathbb{V}_1^{-1}(\mathbf{y}) - \mathbb{V}_2^{-1}(\mathbf{k}) \star \mathbb{V}_3^{-1}(\mathbf{x})||_F^2 + \frac{\beta}{2}||\mathbf{k}||_2^2$ is actually a polynomial of $\mathbf{v} = (\mathbf{y}, \mathbf{k}, \mathbf{x})$. $g_1(\mathbf{y}) = ||\mathbb{V}_1^{-1}(\mathbf{y})||_{TTNN} = \sum_{j=1}^{n_b} ||[\mathbb{V}_1^{-1}(\mathbf{y})]_{\Phi(j)}||_*$, and we can see that for each j, $[\mathbb{V}_1^{-1}(\mathbf{y})]_{\Phi(j)}$ is the polynomials of $\mathbf{y}$, and from Proposition 2.8 (**i**), $||[\mathbb{V}_1^{-1}(\mathbf{y})]_{\Phi(j)}||_*$ is semi-algebraic function defined on $\mathbb{R}^{N_1}$. Thus, $g_1(\mathbf{y})$ is semi-algebraic. $g_3(\mathbf{x}) = \gamma(||h_1(\mathbf{x})||_1 + ||h_2(\mathbf{x})||_1)$ and $h_1(\mathbf{x}) = D_1(\mathbb{V}_3^{-1}(\mathbf{x}))$, $h_2(\mathbf{x}) = (\mathbb{V}_3^{-1}(\mathbf{x}))D_2^T$. It's easy to prove that h_1 and h_2 are polynomials of $\mathbf{x}$, and by Proposition 2.8 (**g**), g_3 is semi-algebraic on $\mathbb{R}^{N_3}$. As for $\delta_{\hat{\mathfrak{S}}}$, the graph $\{(\mathbf{v}, \delta_{\hat{\mathfrak{S}}}(\mathbf{v})) | \mathbf{v} \in \mathbb{R}^N\} = (\hat{\mathfrak{S}} \times \{0\}) \cup ((\mathbb{R}^N \setminus \hat{\mathfrak{S}}) \times \{+\infty\})$ is semi-algebraic because $\hat{\mathfrak{S}}$ is semi-algebraic subset of $\mathbb{R}^N$. Therefore, the objective function $\tilde{F}(\mathbf{v})$ is semi-algebraic on $\mathbb{R}^N$.

With Proposition 2.9, the objective function $\tilde{F}$ is semi-algebraic on $\hat{\mathfrak{S}}$. By Proposition 5.3, $\tilde{F}$ has KL property at each point $\mathbf{v} \in \hat{\mathfrak{S}}$. □

Lemma 5.5 ([4]). *For a PLSC function $f : \mathbb{R}^n \to \mathbb{R}$, suppose that there is a sequence $\{x^k\}_{k\in\mathbb{N}}$ satisfying the following three conditions:*

H1. *(sufficient decrease condition). For each $k \in \mathbb{N}$, there exists a fixed constant $\mathfrak{a} > 0$,*

$$f(x^{k+1}) + \mathfrak{a}||x^{k+1} - x^k||_2^2 \leq f(x^k).$$

H2. *(relative error condition). For each $k \in \mathbb{N}$, there exists a fixed constant $\mathfrak{b} > 0$, and $\exists w^{k+1} \in \partial f(x^{k+1})$, such that*

$$||w^{k+1}|| \leq \mathfrak{b}||x^{k+1} - x^k||_2.$$

H3. *(continuity condition) There exists a subsequence $\{x^{k_j}\}_{j\in\mathbb{N}}$, and x^* such that*

$$x^{k_j} \to x^* \quad \textit{and} \quad f(x^{k_j}) \to f(x^*) \quad \textit{as} \quad j \to +\infty.$$

*And if f has the KL property at x^* in **H3.**, then*

- $x^k \to x^*$.
- $0 \in \partial f(x^*)$, *meaning that* x^* *is a critical point of* f.
- *The sequence* $\{x^k\}_{k\in\mathbb{N}}$ *has a finite length, i.e.,*

$$\sum_{k=0}^{+\infty} ||x^{k+1} - x^k||_2^2 < +\infty.$$

The following theorem proves that if we inexactly solve each subproblem in PAM Iteration with errors satisfying Assumption 5.1, the sequence we obtained will still converge to a critical point of the objective function $\tilde{F}$.

Theorem 5.6 (global convergence of inexact solutions with errors)**.** *Let* $\mathbf{v}^i = (\mathbf{y}^i, \mathbf{k}^i, \mathbf{x}^i)$ *is the sequence generated by PAM Iteration (5.2)-(5.4) with errors satisfying (5.11)-(5.19) in Assumption 5.1. Then there exists a critical point* $\mathbf{v}^*$*, such that*

- $\mathbf{v}^i \to \mathbf{v}^*$;
- $\mathbf{0} \in \partial\tilde{F}(\mathbf{v}^*)$;
- $\{\mathbf{v}^i\}_{i\in\mathbb{N}}$ *has a finite length, i.e.,*

$$\sum_{i=0}^{+\infty} ||\mathbf{v}^{i+1} - \mathbf{v}^i||_2^2 < +\infty.$$

Proof. From (5.11)-(5.13) in Assumption 5.1,

$$\begin{aligned}
&\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \frac{\rho_1}{2}||\mathbf{y}^{i+1} - \mathbf{y}^i||_2^2 - \tilde{F}(\mathbf{y}^i, \mathbf{k}^i, \mathbf{x}^i) \le \frac{\epsilon_1}{2}||\mathbf{y}^{i+1} - \mathbf{y}^i||_2^2,\\
&\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \frac{\rho_2}{2}||\mathbf{k}^{i+1} - \mathbf{k}^i||_2^2 - \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) \le \frac{\epsilon_2}{2}||\mathbf{k}^{i+1} - \mathbf{k}^i||_2^2,\\
&\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \frac{\rho_3}{2}||\mathbf{x}^{i+1} - \mathbf{x}^i||_2^2 - \tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) \le \frac{\epsilon_3}{2}||\mathbf{x}^{i+1} - \mathbf{x}^i||_2^2,
\end{aligned}$$

where $\epsilon_1 \in (0, \rho_1)$, $\epsilon_2 \in (0, \rho_2)$, $\epsilon_3 \in (0, \rho_3)$. Summing up the above three inequalities, we can obtain

$$\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \frac{\rho_\epsilon}{2}||\mathbf{v}^{i+1} - \mathbf{v}^i||_2^2 \le \tilde{F}(\mathbf{y}^i, \mathbf{k}^i, \mathbf{x}^i),$$

where $\rho_\epsilon = \min\{\rho_1 - \epsilon_1, \rho_2 - \epsilon_2, \rho_3 - \epsilon_3\}$. Therefore, the condition **H1.** in the Lemma 5.5 is satisfied.

From (5.14)-(5.16), there exists

$$\begin{aligned}
\xi_1^{i+1} &\in \partial_{\mathbf{y}}\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i)\\
&= \nabla_{\mathbf{y}}\Psi(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \partial_{\mathbf{y}} g_1(\mathbf{y}^{i+1}) + \partial_{\mathbf{y}}\delta_{\widehat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i),\\
\xi_2^{i+1} &\in \partial_{\mathbf{k}}\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i)\\
&= \nabla_{\mathbf{k}}\Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \partial_{\mathbf{k}}\delta_{\widehat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i),\\
\xi_3^{i+1} &\in \partial_{\mathbf{x}}\tilde{F}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i)\\
&= \nabla_{\mathbf{x}}\Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \partial_{\mathbf{x}} g_3(\mathbf{x}^{i+1}) + \partial_{\mathbf{x}}\delta_{\widehat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i).
\end{aligned}$$

Thus, there exists

$$\begin{aligned}
&\zeta_1^{i+1} \in \partial_{\mathbf{y}} g_1(\mathbf{y}^{i+1}) + \partial_{\mathbf{y}} \delta_{\hat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i),\\
&\zeta_2^{i+1} \in \partial_{\mathbf{k}} \delta_{\hat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i),\\
&\zeta_3^{i+1} \in \partial_{\mathbf{x}} g_3(\mathbf{x}^{i+1}) + \partial_{\mathbf{x}} \delta_{\hat{\mathfrak{S}}}(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}),
\end{aligned}$$

such that

$$\xi_1^{i+1} = \nabla_{\mathbf{y}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \zeta_1^{i+1} + \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i), \tag{5.23}$$

$$\xi_2^{i+1} = \nabla_{\mathbf{k}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \zeta_2^{i+1} + \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i), \tag{5.24}$$

$$\xi_3^{i+1} = \nabla_{\mathbf{x}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \zeta_3^{i+1} + \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i). \tag{5.25}$$

Note that

$$\delta_{\hat{\mathfrak{S}}}(\mathbf{v}) = \delta_{\hat{\mathfrak{S}}_1}(\mathbf{y}) + \delta_{\hat{\mathfrak{S}}_2}(\mathbf{k}) + \delta_{\hat{\mathfrak{S}}_3}(\mathbf{x}),$$

then we have $\partial_{\mathbf{y}} \delta_{\hat{\mathfrak{S}}}(\mathbf{v}) = \partial_{\mathbf{y}} \delta_{\hat{\mathfrak{S}}_1}(\mathbf{y})$, $\partial_{\mathbf{k}} \delta_{\hat{\mathfrak{S}}}(\mathbf{v}) = \partial_{\mathbf{k}} \delta_{\hat{\mathfrak{S}}_2}(\mathbf{k})$, $\partial_{\mathbf{x}} \delta_{\hat{\mathfrak{S}}}(\mathbf{v}) = \partial_{\mathbf{x}} \delta_{\hat{\mathfrak{S}}_3}(\mathbf{x})$.

The set $\hat{\mathfrak{S}}$ is bounded and closed, and the function $\Psi(\mathbf{v}) = \frac{1}{2}||\mathbb{V}_1^{-1}(\mathbf{y}) - \mathbb{V}_2^{-1}(\mathbf{k}) \star \mathbb{V}_3^{-1}(\mathbf{x})||_F^2 + \frac{\beta}{2}||\mathbf{k}||_2^2$ is a polynomial of $\mathbf{v} = (\mathbf{y}, \mathbf{k}, \mathbf{x})$, therefore, $\nabla_{\mathbf{v}} \Psi(\mathbf{v})$ is Lipschitz continuous on the set $\hat{\mathfrak{S}}$, i.e., there exits a positive constant C, such that for any $\mathbf{v}, \mathbf{w} \in \hat{\mathfrak{S}}$,

$$||\nabla \Psi(\mathbf{v}) - \nabla \Psi(\mathbf{w})||_2 \leq C||\mathbf{v} - \mathbf{w}||_2. \tag{5.26}$$

Let $w^{i+1} = (w_1^{i+1}, w_2^{i+1}, w_3^{i+1})$ and

$$\begin{aligned}
w_1^{i+1} &= \nabla_{\mathbf{y}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \zeta_1^{i+1},\\
w_2^{i+1} &= \nabla_{\mathbf{k}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \zeta_2^{i+1},\\
w_3^{i+1} &= \nabla_{\mathbf{x}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \zeta_3^{i+1},
\end{aligned}$$

then $w^{i+1} \in \partial_{\mathbf{v}} \tilde{F}(\mathbf{v})$. Substituting the ζ^{i+1} in above equations by (5.23)-(5.25), we obtain the following:

$$\begin{aligned}
w_1^{i+1} &= \nabla_{\mathbf{y}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) - \nabla_{\mathbf{y}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^i, \mathbf{x}^i) + \xi_1^{i+1} - \rho_1(\mathbf{y}^{i+1} - \mathbf{y}^i),\\
w_2^{i+1} &= \nabla_{\mathbf{k}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) - \nabla_{\mathbf{k}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^i) + \xi_3^{i+1} - \rho_2(\mathbf{k}^{i+1} - \mathbf{k}^i),\\
w_3^{i+1} &= \nabla_{\mathbf{x}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) - \nabla_{\mathbf{x}} \Psi(\mathbf{y}^{i+1}, \mathbf{k}^{i+1}, \mathbf{x}^{i+1}) + \xi_3^{i+1} - \rho_3(\mathbf{x}^{i+1} - \mathbf{x}^i).
\end{aligned}$$

Therefore,

$$\begin{aligned}
||w^{i+1}||_2 &\leq C||\mathbf{k}^{i+1} - \mathbf{k}^i||_2 + 2C||\mathbf{x}^{i+1} - \mathbf{x}^i||_2 + b_1||\mathbf{y}^{i+1} - \mathbf{y}^i|| + b_2||\mathbf{k}^{i+1} - \mathbf{k}^i||\\
&\quad + b_3||\mathbf{y}^{i+1} - \mathbf{y}^i|| + \rho_1||\mathbf{y}^{i+1} - \mathbf{y}^i|| + \rho_2||\mathbf{k}^{i+1} - \mathbf{k}^i|| + \rho_3||\mathbf{y}^{i+1} - \mathbf{y}^i||\\
&\leq 2C||\mathbf{v}^{i+1} - \mathbf{v}^i||_2 + \rho_b||\mathbf{v}^{i+1} - \mathbf{v}^i||_2 = (2C + \rho_b)||\mathbf{v}^{i+1} - \mathbf{v}^i||_2,
\end{aligned}$$

where $\rho_b = \max\{\rho_1 + b_1, \rho_2 + b_2, \rho_3 + b_3\}$. The first inequality is because of the (5.26) and (5.17)-(5.19) in Assumption 5.1. Therefore, the condition **H2.** in the Lemma 5.5 is satisfied.

Since $\{\mathbf{v}^i\}_{i \in \mathbb{N}}$ is a bounded subset of $\mathbb{R}^N$, it's relative compact, meaning that there exists a subsequence $\{\mathbf{v}^{i_j}\}_{j \in \mathbb{N}} \to \mathbf{v}^*$, as $j \to +\infty$. $\{\mathbf{v}^{i_j}\}_{j \in \mathbb{N}} \subseteq \hat{\mathfrak{S}}$ and $\hat{\mathfrak{S}}$ is a closed set, thus

$\mathbf{v}^* \in \hat{\mathfrak{S}}$, and $\tilde{F}(\mathbf{v}^{i_j}) \to \tilde{F}(\mathbf{v}^*)$, as $j \to +\infty$. Therefore, the condition **H3.** in the Lemma 5.5 is satisfied.

By Lemma 5.4, $\tilde{F}$ has KL property at each point of $\hat{\mathfrak{S}}$, thus at $\mathbf{v}^*$. Hence, the proof is completed by Lemma 5.5. □

Back to the tensor computation rather than the vector form, the Assumption 5.1 can be rewritten as the following Assumption 5.2.

Assumption 5.2. There exists same non-negative constants in Assumption 5.1, $\epsilon_1 \in (0, \rho_1)$, $\epsilon_2 \in (0, \rho_2)$, $\epsilon_3 \in (0, \rho_3)$, and b_1, b_2, b_3, such that for each iteration,

$$F(\mathcal{Y}^{i+1}, \mathcal{K}^i, X^i) + \frac{\rho_1}{2}||\mathcal{Y}^{i+1} - \mathcal{Y}^i||_F^2 - F(\mathcal{Y}^i, \mathcal{K}^i, X^i) \leq \frac{\epsilon_1}{2}||\mathcal{Y}^{i+1} - \mathcal{Y}^i||_F^2, \tag{5.27}$$

$$F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X^i) + \frac{\rho_2}{2}||\mathcal{K}^{i+1} - \mathcal{K}^i||_F^2 - F(\mathcal{Y}^{i+1}, \mathcal{K}^i, X^i) \leq \frac{\epsilon_2}{2}||\mathcal{K}^{i+1} - \mathcal{K}^i||_F^2, \tag{5.28}$$

$$F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X^{i+1}) + \frac{\rho_3}{2}||X^{i+1} - X^i||_F^2 - F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X^i) \leq \frac{\epsilon_3}{2}||X^{i+1} - X^i||_F^2, \tag{5.29}$$

and

$$\exists\ \xi_1^{i+1} \in \partial_{\mathcal{Y}} F(\mathcal{Y}^{i+1}, \mathcal{K}^i, X^i) + \rho_1(\mathcal{Y}^{i+1} - \mathcal{Y}^i), \tag{5.30}$$

$$\exists\ \xi_2^{i+1} \in \partial_{\mathcal{K}} F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X^i) + \rho_2(\mathcal{K}^{i+1} - \mathcal{K}^i), \tag{5.31}$$

$$\exists\ \xi_3^{i+1} \in \partial_X F(\mathcal{Y}^{i+1}, \mathcal{K}^{i+1}, X^{i+1}) + \rho_3(X^{i+1} - X^i), \tag{5.32}$$

with

$$||\xi_1^{i+1}||_F \leq b_1||\mathcal{Y}^{i+1} - \mathcal{Y}^i||_F, \tag{5.33}$$

$$||\xi_2^{i+1}||_F \leq b_2||\mathcal{K}^{i+1} - \mathcal{K}^i||_F, \tag{5.34}$$

$$||\xi_1^{i+1}||_F \leq b_3||X^{i+1} - X^i||_F. \tag{5.35}$$

Corollary 5.7 (global convergence of inexact iteration)**.** *Let $(\mathcal{Y}^i, \mathcal{K}^i, X^i)$ be the sequence generated by PAM Iteration (4.1)-(4.3) with errors satisfying (5.27)-(5.35) in Assumption 5.2. Then there exists $(\mathcal{Y}^*, \mathcal{K}^*, X^*)$, such that*

- $(\mathcal{Y}^i, \mathcal{K}^i, X^i) \to (\mathcal{Y}^*, \mathcal{K}^*, X^*)$;
- $0 \in \partial F(\mathcal{Y}^*, \mathcal{K}^*, X^*)$;
- $\{\mathcal{Y}^i, \mathcal{K}^i, X^i\}_{i \in \mathbb{N}}$ *has a finite length, i.e.,*

$$\sum_{i=0}^{+\infty} \sqrt{||\mathcal{Y}^{i+1} - \mathcal{Y}^i||_F^2 + ||\mathcal{K}^{i+1} - \mathcal{K}^i||_F^2 + ||X^{i+1} - X^i||_F^2} < +\infty.$$

The Corollary 5.7 demonstrates that if the sequence generated by PAM Iteration (4.1)-(4.3) with errors satisfying (5.27)-(5.35) in Assumption 5.2, then our sequence will converge to the critical point of the objective function F. Moreover, due to the non-convexity of the object function in $(\mathcal{Y}, \mathcal{K}, X)$, the global minimum is hard to achieve.

6 Numerical Experiments

In this section, numerical experiments are conducted and numerical results show both the effectiveness and the robustness of the proposed model for tensor completion and blind deconvolution. We adopt both the discrete Fourier transform matrix (DFT) and the discrete cosine transform (DCT) matrix as the transform matrix Φ in TTNN.

6.1 Parameters setting

Fixed hyperparameters. The observation ratio is defined as $\rho = \frac{|\Omega|}{n_v n_h n_b}$, where $|\Omega|$ denotes the number of elements in Ω. In our numerical experiments, the observed data are sampled uniformly. All images tested in this section are of size 251×251. Usually the model should be insensitive to the parameters ρ_1, ρ_2 and ρ_3 (defined in (4.1) - (4.3)), thus we set them all to be 10^{-6}, i.e., $\rho_1 = \rho_2 = \rho_3 = 10^{-6}$. We adopt the following relative error as stopping criterion for each iteration step,

$$\max\{\frac{||\mathcal{Y}^{i+1} - \mathcal{Y}^i||_F^2}{||\mathcal{Y}^{i+1}||_F^2}, \frac{||\mathcal{K}^{i+1} - \mathcal{K}^i||_F^2}{||\mathcal{K}^{i+1}||_F^2}, \frac{||X^{i+1} - X^i||_F^2}{||X^{i+1}||_F^2}\} < 5 \times 10^{-7}.$$

The maximum number of PAM iteration for our model is set to 500. In ADMM iteration, we set $\kappa_1 = \kappa_2 = \kappa_3 = 2$ for lower computational costs.

Tunable hyperparameters. In our proposed model (3.1), hyperparameters α, β and γ play essential roles in recovering clean images. Proper choices of penalty parameters μ_1, μ_2 and μ_3 for augmented Lagrangian methods also contribute to empirical performances of the proposed algorithm. In each experiment, we tune the above hyperparameters by grid search. Surprisingly, we find that with fixed image size, blur kernels and observation ratios, the numerically optimal hyperparameters are close even for different images, which implies the practical feasibility of the proposed model. Hyperparamters for test 1 (applicable to all images with the same setting) are $(\alpha, \beta, \gamma, \mu_1, \mu_2, \mu_3) = (1e{-}1, 8e{-}1, 15e{-}5, 15, 7e2, 7e{-}3)$. For test 2 to 4, we tune hyperparameters based on those in test 1.

Evaluation. In order to evaluate the performance of our model, we use PSNR values to measure the quality of the restored sharp image X and blurred images $\mathcal{Y}$. If tensor $\hat{\mathcal{Y}}$ is an approximation to tensor $\mathcal{Y}$, then the PSNR values for $\hat{\mathcal{Y}}$ and $\mathcal{Y}$ is as follows,

$$\text{MSE} = \frac{1}{n_v n_h n_b} \sum_{(i,j,t)} |\mathcal{Y}(i,j,t) - \hat{\mathcal{Y}}(i,j,t)|^2,$$

$$\text{PSNR} = 10 \cdot \log_{10}(\frac{\max(\mathcal{Y})^2}{\text{MSE}}).$$

Initial values. Initial values for the PAM iterations are

$$\mathcal{Y}^0(i,j,t) = \begin{cases} \frac{1}{2}, & \text{if } (i,j,t) \notin \Omega, \\ \mathcal{M}(i,j,t), & \text{if } (i,j,t) \in \Omega. \end{cases}$$

$$X^0 = \mathcal{Y}(:,:,1),$$

$$\mathcal{K}(i,j,t) = \frac{1}{(2k_v+1)(2k_h+1)}, \quad -k_v \le i \le k_v, \quad -k_h \le j \le k_h, \quad 1 \le t \le k_b.$$

Kernels. In the follwoing tests, we use Gaussian kernels with different standard deviations, as is defined in (6.1).

$$G(i,j) = \begin{cases} exp(-\frac{i^2+j^2}{2\sigma^2}), & \text{if } |i| \le k_v \text{ and } |j| \le k_h, \\ 0, & \text{otherwise.} \end{cases} \tag{6.1}$$

We assume that for the blurred images from the same source, the size of blurs should be the same, i.e., k_v and k_h are fixed for all blurred images in experiments, but with different σ. Therefore, we set $k_v = k_h = 5$ in the tests.

Transformations for tensor nuclear norm. In each case, we use discrete Fourier transform (DFT) matrix $\Phi_1 = \frac{1}{\sqrt{k_b}} F_{k_b}$ and discrete cosine transform (DCT) matrix $\Phi_2 = C_{k_b}$ as two transform matrix for TTNN. In [24], Song et.al. proposed a kind of data dependent transform matrix Φ, however, in our tests, it's not appropriate to adopt the strategy because our observations have missing values.

6.2 Single observed image with incomplete data

To show the effectiveness of our model of integrating information from different blurred images, we compare it with a "downgraded" method that restores a sharp image from single incomplete blurred image. The model adopts low rank prior, Tikhonov prior and low total variation prior to the blurred image $Y \in \mathbb{R}^{n_v \times n_h}$, the kernel $K \in \mathbb{R}^{k_v \times k_h}$ and the sharp image $X \in \mathbb{R}^{n_v \times n_h}$ respectively. It is a special case of our model that $n_b = k_b = 1$, thus we call it **one-dimension model**. Therefore, the model is

$$\begin{aligned} &\arg\min_{Y,K,X} \frac{1}{2}||Y - K \star X||_F^2 + \alpha||Y||_* + \frac{\beta}{2}||K||_F^2 + \gamma(||D_1 X||_1 + ||X D_2^T||_1), \\ &s.t. \quad 0 \le X, Y \le 1, \;\; P_\Omega(Y - M) = \mathbf{0}, \;\; K \ge 0, \;\; \sum_{i,j} K(i,j) = 1, \end{aligned} \tag{6.2}$$

where α, β and γ are given positive parameters of regularization terms. Ω denotes the observed entries of the blurred image and M is the observed data, satisfying that $P_\Omega(Y) = P_\Omega(M)$ and the values of matrix M outside Ω is 0. Here, '$\star$' is the convolution product with periodic boundary condition. The procedure of solving this optimization problem is similar to but much easier than solving our model.

6.3 Numerical results

There are several tests to show the effectiveness and robustness of our model. The blurred images are generated by the convolution of Gaussian kernels and the corresponding sharp images. In test 6.3.1, we select four different clean images: "Pepper", "Akita", "Cameraman" and "Barbara" for numerical tests. We compare the restored results with different observation ratios. To show the performance of our model in integrating information from different blurred images, we also compare it with the state-of-the-art method of recovering a sharp image from single incomplete blurred image. For brevity, we only present the best restored images and their PSNR values as results of the one-dimension method, although it does blind deconvolution to all obtained blurred images. In test 6.3.2, to further reveal the effectiveness of our model in extracting information, we test on different number of blurred images with fixed range of blur kernels. In test 6.3.3, we enlarge the range of blurs to further investigate the properties of our model. Finally, small perturbation is considered in observation ratio, indicating the robustness of our model with respect to observation ratio, please see test 6.3.4.

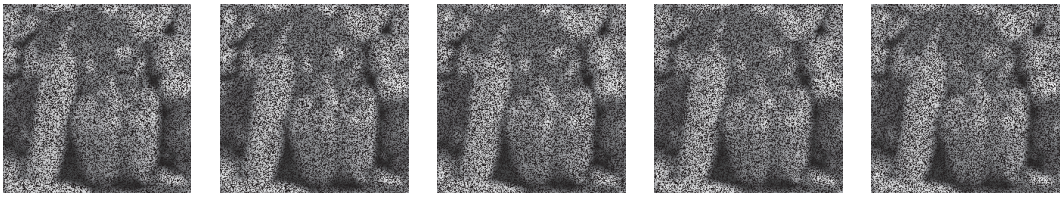
6.3.1 Test 1: performance for different images

In Figure 1, the "Pepper" images are blurred by Gaussian kernels with $\sigma \in \{3, 3.5, 4, 4.5, 5\}$. Figure 2 and Figure 3 show the observed images with observation ratio $\rho = 0.6$ and $\rho = 0.8$ respectively.

The clean image is shown in Figure 4a. Figure 4b is the restored image with $\rho = 0.6$ and the transformation matrix Φ being a discrete Fourier transformation matrix in TTNN.

(a) Blurred image 1 (b) Blurred image 2 (c) Blurred image 3 (d) Blurred image 4 (e) Blurred image 5

Figure 1: "Pepper" blurred images with continuously changing kernel blurs.



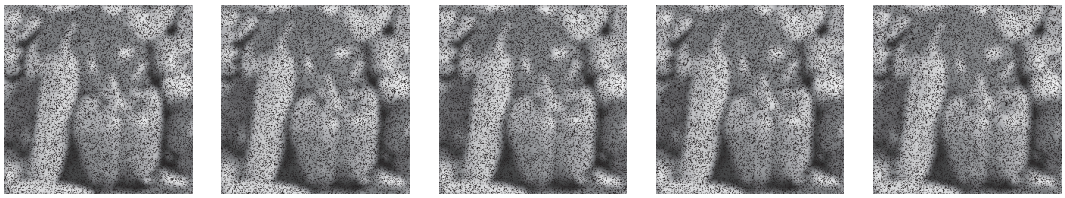
(a) Observation 1 (b) Observation 2 (c) Observation 3 (d) Observation 4 (e) Observation 5

Figure 2: "Pepper" observed images with $\rho = 0.6$.

Figure 4c shows the restored image with $\rho = 0.6$ and the transformation matrix Φ being a discrete cosine transformation matrix in TTNN. Figure 4d and Figure 4e are the restored image with $\rho = 0.8$ and the transformation matrix Φ being a discrete Fourier and cosine transformation matrix in TTNN respectively. Comparing Figure4b and Figure 4c with Figure 4d and Figure 4e, we can conclude that more observed data (larger observation ratio ρ) can result in better restoration. From the PSNR values of the restored images, discrete cosine transform matrix (DCT), as a transformed matrix in TTNN, performs better than DFT in our model for image restoration. To show the effectiveness of our model, we also use a blind deconvolution method to recover the sharp image from single incomplete blurred image. Because most of the methods use fully observed data, which are different from our model, we compare our model with the one-dimension method mentioned in section 6.2. The results of the one-dimension method is shown is in Figure 6. We zoom in the region of pepper stem with complicated details. The visual quality of zoomed images in Figure 5 is much better than images in Figure 6, demonstrating the effectiveness of our model that it performs better than restoring clean image from single incomplete blurred image and it can integrate information from all blurred images.

Numerical results of "Akita" images are shown in Figure 7. Figure 7b is the restored image by our proposed model with DCT as transformed matrix and $\rho = 0.6$. Image in Figure 7d is recovered by one-dimension method with $\rho = 0.6$. The restored image by our model has higher PSNR value than one-dimension method. We also zoom in the region of Akita's nose and whiskers, as is shown is in Figure 7c and Figure 7e. Figure 7c is closer to the zoomed clean image in Figure 7a and has more well recovered details. However, it's difficult to recognize Akita's whiskers in Figure 7e and the "Akita" image's nose is also of low resolution with obvious corners of pixels. The visual quality of zoomed images obtained by our model is much better than images recovered by the one-dimension method, indicating the effectiveness of our model.

The "Cameraman" images recovered by our model have higher PSNR values than that of one-dimension model as is shown in Figure 8. We zoom in the face and camera regions



(a) Observation 1 (b) Observation 2 (c) Observation 3 (d) Observation 4 (e) Observation 5

Figure 3: "Pepper" observed images with $\rho = 0.8$.

(a) Clean image (b) Restored image, Fourier transform, $\rho = 0.6$, PSNR=30.6261 (c) Restored image, cosine transform, $\rho = 0.6$, PSNR=31.0788 (d) Restored image, Fourier transform, $\rho = 0.8$, PSNR=31.0991 (e) Restored image, cosine transform, $\rho = 0.8$, PSNR=31.3448

Figure 4: "Pepper" clean and restored images.

(a) Zoomed clean image (b) Zoomed restored image, Fourier transform, $\rho = 0.6$ (c) Zoomed restored image, cosine transform, $\rho = 0.6$ (d) Zoomed restored image, Fourier transform, $\rho = 0.8$ (e) Zoomed restored image, cosine transform, $\rho = 0.8$

Figure 5: "Pepper" zoomed clean and restored images.

(a) Restored image, $\rho = 0.6$, $PSNR = 27.5730$ (b) Zoomed restored image, $\rho = 0.6$ (c) Restored image, $\rho = 0.8$, $PSNR = 28.5924$ (d) Zoomed restored image, $\rho = 0.8$

Figure 6: "Pepper" restored and zoomed images by single blurred image only.

of the "Cameraman" image with full of details. Our model can recover most of the details but the one-dimension model does not behave so well.

(a) Zoomed clean image (b) Restored image, cosine transform, $\rho = 0.6$, PSNR=34.4482 (c) Zoomed restored image, cosine transform, $\rho = 0.6$ (d) Restored image, one-dimension method, $\rho = 0.6$, PSNR=31.5432 (e) Zoomed restored image, one-dimension method, $\rho = 0.6$

Figure 7: "Akita" zoomed clean and restored images.

(a) Zoomed clean image (b) Restored image, cosine transform, $\rho = 0.6$, PSNR=29.2544 (c) Zoomed restored image, cosine transform, $\rho = 0.6$ (d) Restored image, one-dimension method, $\rho = 0.6$, $PSNR = 26.3951$ (e) Zoomed restored image, $\rho = 0.6$

Figure 8: "Cameraman" clean image and the restored images.

(a) Zoomed clean image (b) Restored image, cosine transform, $\rho = 0.6$, PSNR=28.6385 (c) Zoomed restored image, cosine transform, $\rho = 0.6$ (d) Restored image, one-dimension method, $\rho = 0.6$, $PSNR = 26.3631$ (e) Zoomed restored image, $\rho = 0.6$

Figure 9: "Barbara" zoomed clean and restored images.

The "Barbara" images recoveries are displayed in Figure 9. The images restored by our model have higher PSNR values than that of one-dimension model. The face and headscarf regions with complicated details and stripes are zoomed, referred to Figure 9a, Figure 9c and Figure 9e. Although our model performs better than the one-dimension method in the zoomed regions, it does not perform so well because of the complicated details especially with much texture in "Barbara" image.

Table 1 lists the numerical results of all four kinds of images. The left hand side of the table shows the PSNR values of the restored sharp image X and the right hand side

		PSNR of restored sharp images			PSNR of completed blurred images		
		DFT	DCT	one-dim	DFT	DCT	one-dim
Pepper	$p = 0.6$	30.6261	**31.0788**	27.5730	66.7739	**69.8342**	54.8405
	$p = 0.8$	31.0991	**31.3448**	28.5924	70.5208	**76.3178**	66.2547
Akita	$p = 0.6$	34.2476	**34.4482**	31.5432	69.0501	**69.8394**	54.3027
	$p = 0.8$	34.5492	**34.7374**	32.0961	74.5742	**77.4138**	64.9377
Cameraman	$p = 0.6$	29.0374	**29.2544**	26.3951	64.8718	**69.5704**	56.7223
	$p = 0.8$	29.3584	**29.4952**	27.0570	73.5482	**79.1875**	65.8105
Barbara	$p = 0.6$	28.3122	**28.6385**	26.3631	68.2208	**68.8145**	57.5101
	$p = 0.8$	29.0014	**29.0080**	27.3385	71.6434	**76.1636**	69.1548

Table 1: PSNR values of restored sharp images and incomplete blurred images

shows the PSNR values of the recovered blurred images $\mathcal{Y}$. In four images, our model performs better than one-dimension method from PSNR values of restored sharp images and discrete cosine transform (DCT) as the transformed matrix Φ is better than discrete Fourier transform (DFT) in our experiments. Better performance of DCT is within our expectations because DCT in TTNN assumes the reflective boundary conditions (also known as Neumann boundary conditions) for blurred images (a tensor) on third dimension. Also, high PSNR values of completed blurred images $\mathcal{Y}$ shows the excellent performance of our model in low rank tensor completion.

In next several parts, we adopt DCT as the transformed matrix in TTNN.

6.3.2 Test 2: performance for different numbers of observed images

In this part, we show the effectiveness of our model with different number of input blurred images. The test images are same as in Figure 1. For more convenience, we fix the observation ratio $\rho = 0.6$ and the standard deviation of Gaussian kernels σ ranges from $[3, 5]$. Furthermore, the standard deviations are equally distributed in interval $[3, 5]$, i.e., if given $n_b = k_b$ blurred images, in matlab language, $\sigma \in linspace(3, 5, n_b)$, where

$$\sigma_1 = 3, \quad \sigma_{k_b} = 5, \quad \sigma_{i+1} - \sigma_i = \sigma_i - \sigma_{i-1}.$$

Numerical results are displayed is Figure 10. PSNR values increase dramatically when the number of input blurred images change from 3 to 5. But for more than 5 input blurred images in our model, the PSNR value moves more and more slowly. That's because with our setting that standard deviations are equally spaced in the interval $[3, 5]$, more input images means more similarities between neighboring images, leading to the growing amount of redundant information. In conclusion, too many similar blurred images will not improve the quality of the restored image because they merely have limited information.

Hence, to improve the restoration, we can enlarge the range of standard deviations of Gaussian blurs. See in the next test.

6.3.3 Test 3: performance for different blurred images

From Figure 10, with 10 input blurred images, our model has extracted most of the information from blurred images whose standard deviations of Gaussian kernels ranging from 3 to 5. If we enlarge the range of standard deviations of Gaussian blurs, e.g., interval $[2, 8]$, and 10 standard deviations are equally spaced in this range, the numerical results are shown in the following figure.

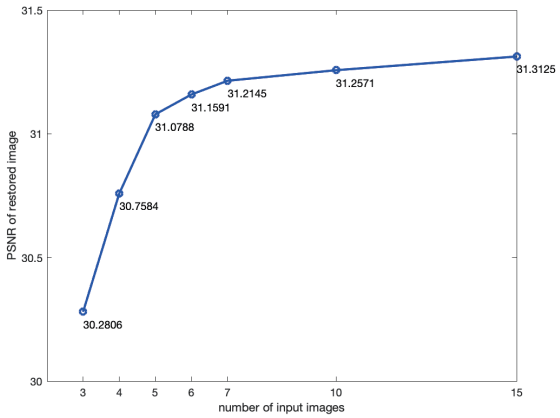
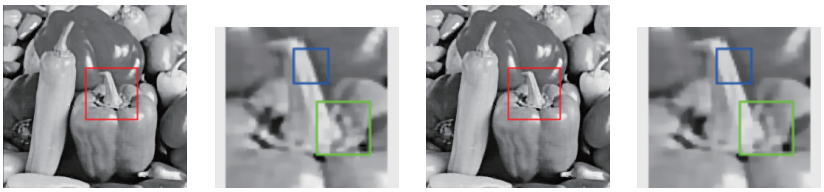


Figure 10: PSNR values of restored images by our model with different number of input blurred images.



(a) Restored image, σ range from 2 to 8, 10 images, $\rho = 0.6$, PSNR=31.8857

(b) Zoomed image

(c) Restored image, σ range from 3 to 5, 10 images, $\rho = 0.6$, PSNR=31.2571

(d) Zoomed image

Figure 11: Restored images and Zoomed regions

The restored image whose blur kernels range from 2 to 8, has larger PSNR values. We zoom in the region of the stem of the pepper labeled by red rectangle and zoomed region is shown in Figure 11b and Figure 11d. It's easy to find that the blue region of Figure 11d has more what zigzag pixels than the region in Figure 11b. In the green region, Figure 11b also has more details than Figure 11d. Both PSNR values and recovered details demonstrate the effectiveness of our model of extracting information from blurred images.

6.3.4 Test 4: performance for different sampling ratios ρ

In this test, we slightly change the observation ratio of each blurred image to show the robustness of our model w.r.t. the observation ratio ρ. In each test, 5 blurred images are used and the setting of Gaussian blurs is the same as Test 2, i.e., the standard deviations are equally spaced from 3 to 5. We first define original mode and three modes of changing observation ratio. In original mode, we set observation ratio $\rho = 0.6$ for each blurred image. For mode 1, observation ratios are monotonically increasing, i.e.,

$$\rho = [\tilde{\rho}_1, ..., \tilde{\rho}_5] = [0.56, 0.58, 0.60, 0.62, 0.64],$$

where, $\tilde{\rho}_i$ denotes the observation ratio of the i-th blurred image. For mode 2, observation ratios are monotonically decreasing, i.e.,

$$\rho = [\tilde{\rho}_1, ..., \tilde{\rho}_5] = [0.64, 0.62, 0.60, 0.58, 0.56].$$

And for mode 3, observation ratios are changing more complicated, for example,

$$\rho = [\tilde{\rho}_1, ..., \tilde{\rho}_5] = [0.60, 0.62, 0.58, 0.64, 0.56].$$

Numerical results are shown in Figure 2. Images recovered by our model with four kinds of observation modes have very similar PSNR values, meaning that our model is not sensitive to small changes of observation ratios and doesn't require each blurred images to have the same observation ratio strictly. In conclusion, our model is robust with respect to observation ratio that small perturbations of observation ratio don't influence its performance.

	original	mode 1	mode 2	mode 3
PSNR	31.0788	31.1078	31.1577	31.0544

Table 2: PSNR values of images in four observation modes

7 Conclusion

In this paper, we have studied a model that can do blind deconvolution and low rank tensor completion with incomplete observations for third order tensors. The model is mainly for recovering one sharp image from multiple corresponding blurred images with missing values. TTNN, a more general form of TNN, is adopted in our model for low rank tensor completion based on Song's recent work [24]. PAM and ADMM iterations are developed to solve the optimization problem. Moreover, an inexact solution convergence analysis is established. And the numerical experiments are presented to show both the effectiveness and robustness of our proposed model that it can integrate information from different blurred images.

Different unitary matrix can be adopted in TTNN and will have different performances in our model. Numerical experiments in this paper show better performance of DCT than DFT as the transformed matrix in TTNN. It is interesting to propose a model whose transformed matrix can be determined by observations rather than predefined because there are too many proper unitary matrices but difficult to find a relatively optimal one without much numerical testing.

References

[1] F. Abboud, E. Chouzenoux, J.-C. Pesquet, J. H. Chenot and L. Laborelli, An alternating proximal approach for blind video deconvolution, *Signal Processing: Image Communication* 70 (2019) 21–36.

[2] E. Acar, D.M. Dunlavy and T.G. Kolda, Link prediction on evolving data using matrix and tensor factorizations, in: *2009 IEEE International Conference on Data Mining Workshops*, IEEE, 2009, pp. 262–269.

[3] H. Attouch, J. Bolte, P. Redont and A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality, *Mathematics of operations research* 35 (2010) 438–457.

[4] H. Attouch, J. Bolte and B.F. Svaiter, Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods, *Mathematical Programming* 137 (2013) 91–129.

[5] J. Bolte, A. Daniilidis and A. Lewis, The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems, *SIAM Journal on Optimization*, 17 (2007) 1205–1223.

[6] J. Bolte, A. Daniilidis,A. Lewis and M. Shiota, Clarke subgradients of stratifiable functions, *SIAM Journal on Optimization* 18 (2007) 556–572.

[7] N.K. Bose and K.J. Boo, High-resolution image reconstruction with multisensors, *International Journal of Imaging Systems and Technology* 9 (1998) 294–304.

[8] J. Chung, J. Kim, X. Ou, R. Horstmeyer and C. Yang, Wide field-of-view fluorescence image deconvolution with aberration-estimation from Fourier ptychography, *Biomedical optics express* 7 (2016) 352–368.

[9] W. Dong, L. Zhang,G. Shi and X. Wu, Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization, *IEEE Transactions on image processing*, 20 (2011) 1838–1857.

[10] R. Fergus, B. Singh, A. Hertzmann, S.T. Roweis and W.T. Freeman, Removing camera shake from a single photograph, in: *ACM SIGGRAPH 2006 Papers*, pp. 787–794.

[11] C.W. Groetsch, The Theory of Tikhonov Regularization for Fredholm Equations, Pitman Publication, Boston, 1984.

[12] F.L. Hitchcock, The expression of a tensor or a polyadic as a sum of products, *Journal of Mathematics and Physics* 6 (1927) 164–189.

[13] M. Karow, Geometry of spectral value sets, Verlag nicht ermittelbar, 2003.

[14] E. Kernfeld, M. Kilmer and S. Aeron, Tensor–tensor products with invertible linear transforms, *Linear Algebra and its Applications* 485 (2015) 545–570.

[15] M.E. Kilmer, K. Braman, N. Hao and R.C. Hoover, Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging, *SIAM Journal on Matrix Analysis and Applications*, 34 (2013) 148–172.

[16] M.E. Kilmer and C.D. Martin, Factorization strategies for third-order tensors, *Linear Algebra and its Applications* 435 (2011) 641–658.

[17] K.C. Kiwiel, Convergence and efficiency of subgradient methods for quasiconvex minimization, *Mathematical programming* 90 (2001) 1–25.

[18] D. Krishnan and R. Fergus, Fast image deconvolution using hyper-Laplacian priors, in: *Advances in Neural Information Processing Systems*, 2009, pp. 1033–1041.

[19] D. Kundur and D. Hatzinakos, A novel blind deconvolution scheme for image restoration using recursive filtering, *IEEE Transactions on Signal Processing* 46 (1998) 375–390.

[20] 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: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5249–5257.

[21] M. Ng, R.H. Chan and W.-C. Tang, A fast algorithm for deblurring models with Neumann boundary conditions, *SIAM Journal on Scientific Computing* 21 (1999) 851–866.

[22] B. Romera-Paredes, H. Aung, N. Bianchi-Berthouze and M. Pontil, Multilinear multitask learning, in: *International Conference on Machine Learning*, 2013, pp. 1444–1452.

[23] L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D: nonlinear phenomena* 60 (1992) 259–268.

[24] G. Song, M.K. Ng and X. Zhang, Robust tensor completion using transformed tensor singular value decomposition, *Numerical Linear Algebra with Applications* 27 (2020): e2299.

[25] L.R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika* 31 (1966) 279–311.

[26] W. Wang and M.K. Ng, *On algorithms for automatic deblurring from a single image*, *Journal of Computational Mathematics* 30 (2012) 80–100.

[27] W. Wang and M.A. Carreira-Perpinán, *Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application*, arXiv preprint arXiv:1309.1541, 2013.

[28] R.K. Ward and B.E.A. Saleh, Image restoration under random time-varying blur, Applied Optics 26 (1987) 4407–4412.

[29] Y. Xu and W. Yin, A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion, *SIAM Journal on Imaging Sciences* 6 (2013) 1758–1789.

[30] Z. Xu, F. Yan, and Q. Yuan, Infinite Tucker decomposition: Nonparametric Bayesian models for multiway data analysis, arXiv preprint arXiv:1108.6296, 2011.

[31] L. Yan, M. Jin, H. Fang, H. Liu and T. Zhang, Atmospheric-turbulence-degraded astronomical image restoration by minimizing second-order central moment, *IEEE Geoscience and Remote Sensing Letters* 9 (2012) 672–676.

[32] X. Zhang and M. K. Ng, A corrected tensor nuclear norm minimization method for noisy low-rank tensor completion, *SIAM Journal on Imaging Sciences*, 12 (2019) 1231–1273.

[33] Q. Zhao, G. Zhou, L. Zhang, A. Cichocki and S.-I. Amari, Bayesian robust tensor factorization for incomplete multiway data, *IEEE transactions on neural networks and learning systems*, 27 (2015) 736–748.

[34] P. Zhou, C. Lu, Z. Lin and C. Zhang, Tensor factorization for low-rank tensor completion, *IEEE Transactions on Image Processing*, 27 (2017) 1152–1163.

[35] X. Zhou, C. Yang, H. Zhao and W. Yu, Low-rank modeling and its applications in image analysis, *ACM Computing Surveys (CSUR)* 47 (2014) 1–33.

Manuscript received 21 October 2021
revised 13 December 2021
accepted for publication 1 January 2022

Yihang Gao
Department of Mathematics, The University of Hong Kong
E-mail address: gaoyh@connect.hku.hk

Xuelei Lin
School of Science, Harbin Institute of Technology
Shenzhen 518055, China
E-mail address: linxuelei@hit.edu.cn

Michael K. Ng
Department of Mathematics, The University of Hong Kong
E-mail address: mng@maths.hku.hk