This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2026.3679727

Separable Decomposition for Ragged Tensors

Yexun Hu, Tai-Xiang Jiang, Michael K. Ng, and Xi-Le Zhao

***Abstract*—Tensor decompositions have proven highly effective in the analysis and processing of multidimensional data. However, many real-world datasets exhibit highly irregular index patterns that deviate from regular tensor structures, rendering classical tensor decomposition methods inapplicable. In this paper, we introduce a CANDECOMP/PARAFAC (CP)-based geometry-aware separable decomposition framework for directly factorizing multidimensionally irregular tensor data, which we term *ragged tensors*. We model the valid domain of a ragged tensor using a binary weighting tensor and exploit the linkage between CP factor rows and corresponding valid elements to decouple the global objective into independent, well-conditioned subproblems. This structural decoupling enables a domain-adapted proximal alternating minimization scheme with closed-form stabilized updates, yielding an efficient and scalable solver along with a rigorous convergence guarantee to a critical point. We validate the proposed method on a range of challenging datasets, including multispectral and hyperspectral images as well as spatial transcriptomics data. Experimental results demonstrate that our approach consistently achieves superior accuracy and efficiency compared to competing baselines, highlighting the effectiveness of our modeling and optimization strategy for ragged tensor data.**

***Index Terms*—Tensor decomposition, ragged tensors, separable decomposition, proximal alternating minimization.**

I. Introduction

A Tensor, as a multidimensional array, naturally represents high-dimensional data encountered in a wide range of real-world applications, including hyperspectral images (HSIs) [1], videos [2], [3], seismic data [4], and DNA microarrays [5]. By leveraging the regular structure of tensor data, tensor decomposition methods aim to decompose these multidimensional arrays into a set of lower-dimensional factors, such as tensors, matrices, and vectors. Tensor decomposition methods have shown immense promise in analyzing and processing complex datasets, uncovering latent structures, and enabling efficient data representation.

The origins of tensor decomposition trace back to the pioneering work of Hitchcock in the 1920s [6], [7] and Cattell in 1944 [8]. Since then, extensive research has been conducted across diverse domains such as signal processing [9], machine learning [10], [11], and neuroscience [12], propelling advancements in tensor decomposition techniques. Notable methods include Tucker decomposition [13], [14], which factorizes a tensor into factor matrices and a core tensor, and CANDECOMP/PARAFAC (CP) decomposition [15], [16], which expresses a tensor as a sum of rank-one tensors formed through the outer product of factorized vectors. The tensor singular value decomposition (t-SVD) framework [17], [18], based on the tensor-tensor convolution, and tensor network decomposition [19], [20], [21], which represents higher-order tensors as interconnected small-scale sub-tensors, have also gained significant attention for their ability to effectively capture multilinear correlations in higher-order data. These methods have achieved remarkable success across a wide range of applications, showcasing their versatility and potential for addressing complex challenges in high-dimensional data analysis [22].

Despite the success of tensor decomposition methods, many real-world applications involve data that are not well-structured enough to be directly expressed in a regular tensor format. For instance, in visual object detection, objects of interest are rarely confined to rectangular shapes. Similarly, temporal health datasets in electronic health records (EHR) [23] often exhibit irregularities, such as inconsistent sampling rates or missing entries resulting from irregular patient visit schedules. These inherent complexities pose significant challenges to representing such data in regular tensor formats, severely limiting the applicability of traditional tensor decomposition methods to these scenarios. A more complex example can be found in spatial transcriptomics techniques [24], [25]. Fig. 1 illustrates the general workflow of the spatial transcriptomics technique. This cutting-edge approach analyzes gene expression while preserving the spatial architecture of tissues, enabling the study of how gene activity coordinates complex cellular organization in multicellular organisms[1]. The result is a spatially resolved transcriptomic map that captures the expression levels of thousands of genes in distinct regions of tissue, represented as spatial "spots" in Fig. 1. A key challenge arises from the inherent irregularity of the tissue itself. Unlike structured rectangular grids, the spatial arrangement of spots in spatial transcriptomics often follows a staggered pattern, commonly referred to as "orange crate packing." This irregular layout reflects not only the arrangement of spots but also the underlying variability in tissue morphology. Tissue samples are rarely uniform; their contours, spatial coverage, and internal cellular distribution often exhibit significant heterogeneity. This irregularity makes it impossible to directly represent the data as regular tensor slices. Moreover, attempting to adapt such irregular data into regular tensors, such as through zero-padding, introduces artificial distortions, disrupting the intrinsic spatial relationships. Such modifications can compromise

Yexun Hu, Tai-Xiang Jiang are with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu 611130, China (Emails: huyexun2022@163.com; taixiangjiang@gmail.com).

Michael K. Ng is with the Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong, (E-mail: michael-ng@hkbu.edu.hk).

Xi-Le Zhao is with the School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China (E-mail: xlzhao122003@163.com).

[1]A portion of Fig. 1 is schematically illustrated based on the Visium Spatial Gene Expression Reagent Kits for FFPE, Document Number CG000407, Rev E, 10x Genomics, (2023, September). For more details, please refer to https://cdn.10xgenomics.com/image/upload/v1695417748/support-documents/CG000407_VisiumSpatialGeneExpressionforFFPE_UserGuide_RevE.pdf.

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2026.3679727

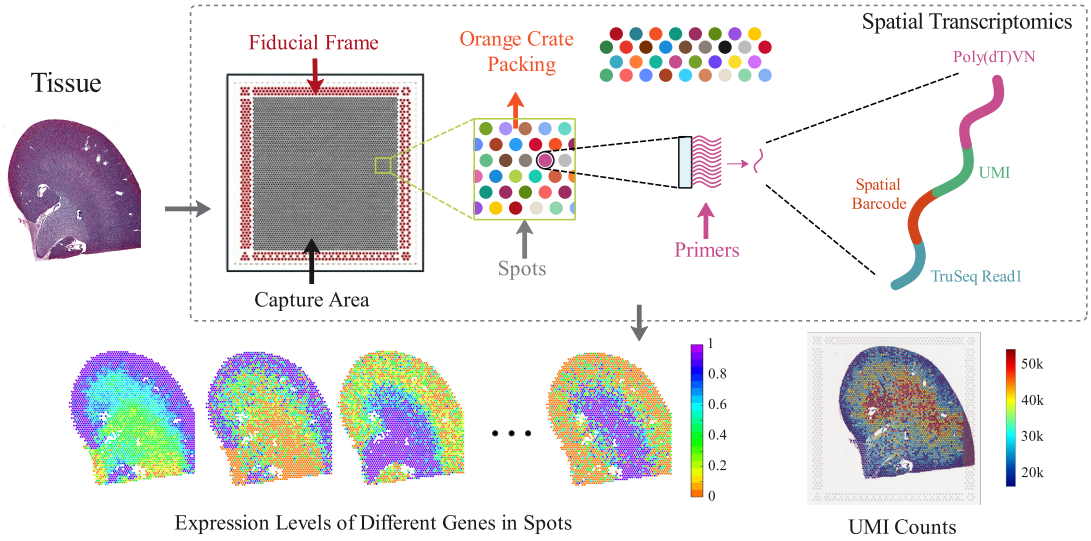


Fig. 1. Overview of the spatial transcriptomics workflow. The process begins by placing a fresh-frozen tissue section onto a capture slide containing numerous spots within a defined capture area. The fiducial frame surrounding the area is used to align the histological image of the tissue with the spatial data. Each spot in the grid, arranged in a row-wise staggered "orange crate packing" pattern, contains a unique set of primers. These primers are designed to capture messenger RNA (mRNA) from the overlying cells. Each primer consists of several key components: (i) a poly(dT)VN adaptor that ensures that only mRNA is reverse transcribed; (ii) a Unique Molecular Identifier (UMI) used for accurate transcript counting; and (iii) a spatial barcode, a unique sequence that encodes the spatial position of its spot of origin. After reverse transcription, sequencing, and data processing, the technique yields spatially resolved gene expression maps. These can be visualized, for example, as heatmaps showing the expression level of individual genes across all spots, or as a map of the total UMI counts that reflects the overall transcriptional activity within the tissue, preserving the original tissue's spatial structure and irregular arrangement pattern of sampling spots.

the accuracy and reliability of downstream analyses by misrepresenting the true structure of the data. Therefore, traditional tensor decomposition methods, which rely on regular data formats, are ill-suited for analyzing spatial transcriptomics data.

The gap between existing tensor decomposition methods and the challenges posed by real-world irregular data significantly restricts the applicability of traditional tensor decomposition methods. While techniques such as zero-padding can reshape irregular data into a regular tensor format, the introduction of artificial "zeros" often results in unavoidable biases that distort the decomposition process and compromise analytical accuracy. In the literature, numerous efforts have been made to address some irregular higher-order tensors. For instance, Ponnapalli *et al.* [26] developed a higher-order generalized singular value decomposition (HO-GSVD) for comparing global mRNA expression across multiple organisms, including 3167 genes from *Saccharomyces pombe*, 4772 genes from *Saccharomyces cerevisiae*, and 13,068 human genes. These datasets are structured as matrices with 17 columns each, corresponding to DNA microarray-measured mRNA expression from 17 equally spaced time points over two cell-cycle periods. Despite having the same number of columns, the matrices differ in row size based on the number of genes in each organism. Similarly, Chalise *et al.* [27] generalized classical nonnegative matrix factorization (NMF) [28] to handle a set of matrices with the same row size but varying column sizes. Their method enables integrative clustering of multiple high-dimensional molecular datasets. Additionally, several studies have applied PARAFAC2 decomposition [29] to address irregular temporal data, such as in health data analysis [23] and stock prediction [30]. However, these approaches are limited to irregular tensors composed of matrices that still share either the same row size or the same column size. A general framework for *multidimensionally* irregular data is crucial for real-world applications, yet remains absent.

In this paper, we refer to such multi-axis, coupled irregularity as a **ragged tensor** (see the formal definition in Eq. (3)). Unlike a conventional tensor supported on a full Cartesian grid, a ragged tensor is supported on a non-rectangular index set $\Omega \subseteq [I_1] \times \cdots \times [I_m]$, where the admissible index range along one mode may depend on coordinates in other modes. For example, in a video, a moving object has a non-rectangular spatial footprint and its location evolves over time; hence across (row, column, time), each frame contains a differently placed and differently sized set of valid pixels.

The problem of ragged tensor decomposition (RTD) poses challenges that are not directly addressed by classical tensor decomposition, nor by many existing "irregular tensor" models that mainly handle *slice-wise* irregularity (i.e., a collection of matrices sharing one aligned dimension while the other dimension varies, e.g., PARAFAC2 [29] and its variants). It is clear that such ragged tensors cannot be directly handled by traditional tensor decomposition methods and irregular tensor decomposition methods like [26], [29]. A possible workaround is to embed Ω into a minimal bounding rectangular tensor and fit a weighted (regular) tensor decomposition model that excludes out-of-domain indices via a binary weight tensor. However, the binary mask generally breaks the standard *unweighted* normal-equation/closed-form update structure underpinning efficient alternating least squares (ALS) updates when dealing with

factor updating subproblem, e.g., the standard matricized-tensor-times-Khatri-Rao-product (MTTKRP) structure in the CP factor updating. Then, the challenge turns out to be the expensive weighted least-squares subproblems or reliance on holistic first-order optimization (e.g., CP-WOPT [31]), which may converge slowly on highly irregular supports.

In this work, we propose a CP-based geometry-aware separable decomposition framework tailored for ragged tensors. We formulate the problem as a weighted least-squares objective over the ragged support. Crucially, while this objective shares the algebraic form of weighted CP [31], we exploit a structural advantage specific to ragged domains, i.e., the separability of the CP decomposition scheme naturally aligns with the ragged support. Specifically, each factor row interacts exclusively with the valid entries in its own support set. This insight allows us to decouple the coupled global weighted objective into a sequence of *row-separable* localized least-squares subproblems. Based on this structure, we develop a proximal alternating minimization (PAM) scheme featuring *stabilized, closed-form row updates*. Moreover, we establish convergence of the generated iterates to a critical point by verifying the assumptions of a standard PAM analysis framework [32], [33]. Extensive experiments on diverse datasets further validate the effectiveness of our modeling strategy and the efficiency and stability of the proposed solver.

Our main contributions are summarized as follows:

- **Modeling.** We formalize a CP-based decomposition for tensors with multi-axis coupled irregularity, extending beyond previous slice-wise irregular settings (e.g., PARAFAC2 [29]). By encoding the valid domain via a binary weight tensor, we cast the problem of RTD as a specialized weighted CP decomposition problem that explicitly respects the non-rectangular data geometry.
- **Algorithm.** We exploit the natural match between the separability of CP decomposition and the ragged support, and thereby decouple each CP factor update into row-separable localized least-squares subproblems. We instantiate a proximal alternating scheme with *stabilized closed-form row updates*, yielding faster and more stable practical convergence (than the previous holistic first-order optimization solver [31]). For rigor, we provide an iterate convergence guarantee to a critical point under a standard PAM framework [32], [33].
- **Experiments.** Comprehensive experiments on synthetic and real datasets demonstrate the effectiveness of the proposed modeling and the efficiency/stability of the solver. Notably, on highly irregular spatial transcriptomics data, our method achieves state-of-the-art recovery performance while being up to two orders of magnitude faster ($100\times$ speedup) than the representative weighted CP baseline, confirming the necessity of exploiting ragged geometry.

The remainder of this paper is structured as follows. In Section II, we introduce ragged tensors and review related tensor decomposition methods, highlighting their strengths and limitations. Section III introduces our separable decomposition model specifically designed for ragged tensors, along with the corresponding optimization algorithm and rigorous convergence analysis. In Section IV, we provide numerical examples to demonstrate the effectiveness and advantages of the proposed method. Finally, Section V concludes the paper with a summary of key findings and potential directions for future research.

A. Notations and Preliminaries

Throughout this work, we use the following conventions for notation:

- Lowercase letters, e.g., x, denote scalars.
- Boldface lowercase letters, e.g., $\mathbf{x}$, denote vectors.
- Boldface uppercase letters, e.g., $\mathbf{X}$, denote matrices.
- Calligraphic letters, e.g., $\mathcal{X}$, denote tensors.

Given an m-th order tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_m}$, its mode-$k$ ($k = 1, 2, \cdots, m$) unfolding matrix is denoted by $\mathbf{X}_{(k)} \in \mathbb{R}^{N_k \times \prod_{j=1, j\neq k} N_j}$, where the $(n_1, n_2, \cdots, n_m)$-th tensor element $x_{n_1,n_2,\cdots,n_m}$ maps to the matrix $(n_k, 1+\sum_{j=1,j\neq k}^{m}(n_j - 1)\prod_{d=1,d\neq j}^{j-1} n_d)$-th element. Additionally, the tensor Frobenius norm of an m-th order tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_m}$ is defined as:

$$\|\mathcal{X}\|_{\mathrm{F}} := \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} = \sqrt{\sum_{n_1,n_2,\cdots,n_m} x^2_{n_1,n_2,\cdots,n_m}}.$$

The **Khatri–Rao product** [34] between two matrices, $\mathbf{A} \in \mathbb{R}^{N_1 \times k}$ and $\mathbf{B} \in \mathbb{R}^{N_2 \times k}$, is denoted by $\mathbf{A} \odot \mathbf{B} \in \mathbb{R}^{N_1 N_2 \times k}$ and is defined as:

$$\mathbf{A} \odot \mathbf{B} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1k}b_{1k} \\ a_{21}b_{11} & a_{22}b_{12} & \cdots & a_{2k}b_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_1 1}b_{11} & a_{N_1 2}b_{12} & \cdots & a_{N_1 k}b_{1k} \\ a_{11}b_{21} & a_{12}b_{22} & \cdots & a_{1k}b_{2k} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2k}b_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_1 1}b_{N_2 1} & a_{N_1 2}b_{N_2 2} & \cdots & a_{N_1 k}b_{N_2 k} \end{bmatrix}.$$

For a regular m-dimensional tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times \cdots \times N_m}$, the **CANDECOMP/PARAFAC (CP)** decomposition of $\mathcal{X}$ is classically expressed as:

$$\mathcal{X} = \sum_{z=1}^{Z} \lambda_z \mathbf{a}^1_{:,z} \circ \mathbf{a}^2_{:,z} \circ \cdots \circ \mathbf{a}^m_{:,z}. \tag{1}$$

where $\circ$ denotes the outer product of vectors, $\lambda_z \in \mathbb{R}$s are scalar weights, and $\mathbf{a}^j_{:,z} \in \mathbb{R}^{N_j}$ is the z-th factor vector along the j-th mode. The minimum $Z \in \mathbb{N}^+$ required to express $\mathcal{X}$ is called the CP rank. The CP decomposition can be written more compactly [35], [22], [36] as

$$\mathcal{X} = [\![\boldsymbol{\lambda}; \mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m]\!],$$

where $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \cdots, \lambda_Z] \in \mathbb{R}^Z$ is a vector of weights, and $\mathbf{A}_j \in \mathbb{R}^{N_j \times Z} = [\mathbf{a}^j_{:,1}, \cdots, \mathbf{a}^j_{:,z}, \cdots, \mathbf{a}^j_{:,Z}]$ for $j = 1, 2, \cdots, m$ is the factor matrix corresponding to the j-th mode. The mode-k matricized version is given by:

$$\mathbf{X}_{(k)} = \mathbf{A}_k \boldsymbol{\Lambda} \left(\mathbf{A}_m \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1 \right)^\top,$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with $\boldsymbol{\lambda}$ on its diagonal, i.e., $\boldsymbol{\Lambda} = \mathtt{diag}(\boldsymbol{\lambda})$, and $(\cdot)^\top$ denotes the matrix transpose.

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2026.3679727

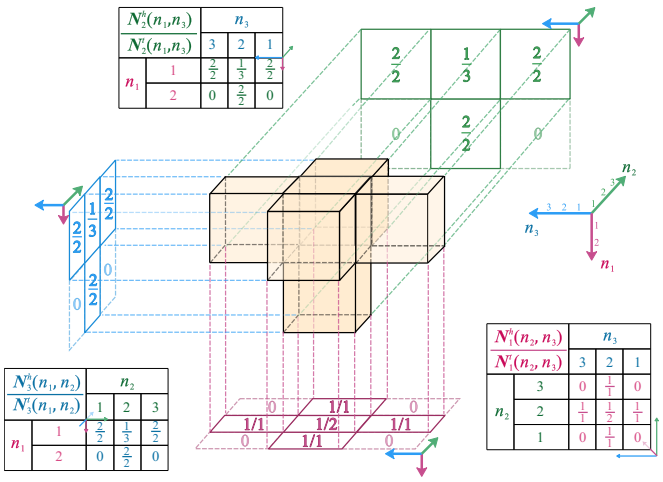


Fig. 2. An illustration of a third-order ragged tensor and its head and tail index tables along three dimensions as defined in (2), with its 1st, 2nd, and 3rd modes indicated by the red, green, and blue arrows, respectively.

II. Ragged Tensors

Let us introduce and define ragged tensors. To reflect its complex and variable geometric structure, a third-order ragged tensor $\mathcal{X}$ can be formally described as:

$$\left\{ x_{n_1,n_2,n_3} \;\middle|\; \begin{array}{l} \boldsymbol{N}_1^h(n_2,n_3) \le n_1 \le \boldsymbol{N}_1^t(n_2,n_3) \\ \boldsymbol{N}_2^h(n_1,n_3) \le n_2 \le \boldsymbol{N}_2^t(n_1,n_3) \\ \boldsymbol{N}_3^h(n_1,n_2) \le n_3 \le \boldsymbol{N}_3^t(n_1,n_2) \end{array} \right\}, \tag{2}$$

where $\boldsymbol{N}_j^h$ and $\boldsymbol{N}_j^t$ denote tables specifying the head (start) and tail (end) indices along the j-th dimension, given a specific set of indices for the other dimensions. A toy example illustrating such a ragged tensor is presented in Fig. 2.

This formulation can be generalized to the mth-order case ($m > 3$) as follows:

$$\left\{ x_{n_1,\cdots,n_m} \;\middle|\; \begin{array}{c} \mathcal{N}_1^h(n_2,\cdots,n_m) \le n_1 \le \mathcal{N}_1^t(n_2,\cdots,n_m) \\ \cdots \\ \mathcal{N}_j^h(n_1,\cdots n_{j-1},n_{j+1},\cdots,n_m) \le n_j \\ \le \mathcal{N}_j^t(n_1,\cdots,n_{j-1},n_{j+1},\cdots,n_m) \\ \cdots \\ \mathcal{N}_m^h(n_1,\cdots,n_{m-1}) \le n_m \le \mathcal{N}_m^t(n_1,\cdots,n_{m-1}) \end{array} \right\} \tag{3}$$

where $\mathcal{N}_j^h$ and $\mathcal{N}_j^t$ are tables that specify the head and tail indices for the j-th dimension, respectively, given the values of the remaining $m-1$ indices. Note that the dimensions of the j-th order are not fixed but depend on other dimensions. Intuitively, for each mode j and fixed values of the remaining indices, the valid indices along mode j form a contiguous interval $[\mathcal{N}_j^h(\cdot), \mathcal{N}_j^t(\cdot)]$. An entry $x_{n_1,\ldots,n_m}$ exists if and only if each n_j lies in its interval. The tables $\mathcal{N}_j^h, \mathcal{N}_j^t$ thus store per-slice start/end indices.

For clarity, the notations used for the ragged tensor and the proposed method are summarized in Table I.

A. Regular Tensor Decomposition

It is clear from Eq. (3) that if the head and tail indices are fixed to the full dimension boundaries (e.g., $\mathcal{N}_j^h \equiv 1$ and $\mathcal{N}_j^t \equiv N_j$), the ragged tensor degenerates into a standard regular tensor on a rectangular support. For such regularly structured data, a variety of tensor decomposition schemes have been studied. The most representative decomposition models include the CANDECOMP/PARAFAC (CP) decomposition [15], [16], which represents a tensor as a sum of rank-one components and serves as the basis for our proposed CP-based RTD formulation, and the Tucker decomposition [13], which factorizes a tensor into mode-wise factor matrices and a core tensor capturing multiway interactions. Beyond these, the tensor singular value decomposition (t-SVD) [17], [18], [37] extends matrix SVD to third-order tensors via the tensor-tensor product, with recent developments exploring function settings [38], [39] and geometric perspectives of the t-product framework [40]. In addition, tensor-network decompositions represent a high-order tensor via contractions of low-order cores (e.g., tensor train [20] and tensor ring [41]), building on early diagrammatic calculi and graphical notations [42], [43]. For interested readers, please refer to [22], [44] for a comprehensive overview or tutorial.

B. Irregular Tensor Decomposition

While the landscape of regular tensor decomposition methods is vast and well-explored, akin to stars scattered across the sky, the methods for decomposing irregular tensors are far less abundant, resembling oases in a desert.

Ponnapalli *et al.* [26] proposed the higher-order generalized singular value decomposition (HO-GSVD) for analyzing global mRNA expression across three organisms (*Saccharomyces pombe*, *Saccharomyces cerevisiae*, and humans). The datasets were represented as a set of matrices $\{\mathbf{R}^i\}$ with the same column size (17 time points in [26]) but varying row sizes (3167, 4772, and 13,068 for the three organisms). The HO-GSVD demonstrates that, under the assumption of full row rank for each $\mathbf{R}^i$, the decomposition can be expressed as:

$$\mathbf{R}^i = \mathbf{U}^i \mathbf{\Sigma}^i \mathbf{V}^\top,$$

where $\mathbf{U}^i$ are orthogonal matrices, $\mathbf{\Sigma}^i$ are diagonal matrices, and $\mathbf{V}$ is a nonsingular matrix shared by all $\mathbf{R}^i$s. By applying HO-GSVD, Ponnapalli *et al.* identified an approximately common HO-GSVD subspace, which revealed similar cell-cycle mRNA expression oscillations among the datasets. This pioneering work underscores the potential of irregular tensor decomposition techniques for uncovering meaningful patterns in datasets with inconsistent dimensions, thereby expanding the reach of tensor decomposition applications beyond the realm of regular tensor data.

Beyond the SVD, nonnegative matrix factorization (NMF) [28] has also proven essential in numerous applications. Classical NMF factorizes a nonnegative matrix $\mathbf{X} \in \mathbb{R}_+^{M\times N}$ into two nonnegative factor matrices $\mathbf{W} \in \mathbb{R}_+^{M\times R}$ and $\mathbf{H} \in \mathbb{R}_+^{R\times N}$, i.e., $\mathbf{X} = \mathbf{WH}$. In [27], Chalise *et al.* extended NMF to handle a set of matrices $\{\mathbf{R}^i\}$ that share the same row size but differ in column sizes, i.e., $\mathbf{R}^i \in \mathbb{R}_+^{M\times P_i}$ for all i. Their generalized factorization is given by:

$$\mathbf{R}^i = \mathbf{W}\mathbf{H}_i,$$

TABLE I
SUMMARY OF NOTATIONS.

Basic Notations			
Symbol	**Description**	**Symbol**	**Description**
$x \in \mathbb{R}$ $\mathbf{X} \in \mathbb{R}^{N_1 \times N_2}$	Scalar Matrix	$\mathbf{x} \in \mathbb{R}^{N}$ $\mathcal{X} \in \mathbb{R}^{N_1 \times \cdots \times N_m}$	Vector Tensor
$x_{n_1 \cdots n_m}$ or $\mathcal{X}(n_1, \ldots, n_m)$	The $(n_1, \cdots, n_m)$-th element of a m-order tensor $\mathcal{T}$	$\|\mathcal{X}\|_{\mathrm{F}} := \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} =$ $(\sum_{n_1, \cdots, n_m} x^2_{n_1, \cdots, n_m})^{1/2}$	Tensor Frobenius norm
$\mathbf{X}_{(j)}$	The mode-j unfolding of $\mathcal{X}$	$\mathbf{A} \odot \mathbf{B}$	The Khatri-Rao product between $\mathbf{A}$ and $\mathbf{B}$
Notations for the ragged tensor			
$\mathcal{N}^h_j(n_1, \cdots, n_{j-1}, n_{j+1}, \cdots, n_m)$	The head table for the tensor's j-th ragged dimension.	$\mathcal{N}^t_j(n_1, \cdots, n_{j-1}, n_{j+1}, \cdots, n_m)$	The table of end indices for the tensor's j-th ragged dimension.
Notations for the proposed Decomposition: $X_{n_1, n_2, \cdots, n_m} = \sum_{z=1}^{Z} \lambda_z \prod_{j=1}^{m} \mathbf{A}_j(n_j, z)$			
Z	The separable rank of tensor $\mathcal{X}$	$\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_Z]$	The vector of CP factor weights
$\boldsymbol{\Lambda} = \mathtt{diag}(\boldsymbol{\lambda})$	The diagonal matrix with $\boldsymbol{\lambda}$ on its diagonal	$\mathbf{a}^j_{:,z}$	The z-th factor vector along the j-th dimension
$\mathbf{A}_j = [\mathbf{a}^j_{:,1}, \ldots, \mathbf{a}^j_{:,z}, \ldots, \mathbf{a}^j_{:,Z}]$	The factor matrix whose columns are Z factor vectors along the j-th dimension	$\mathbf{a}^j_{p,:} = \mathbf{A}_j(p, :)$	The p-th row of the factor matrix $\mathbf{A}_j$
$a^j_{p,z} = \mathbf{a}^j_{:,z}(p) = \mathbf{a}^j_{p,:}(z)$ $= \mathbf{A}_j(p, z)$	The p-th element of $\mathbf{a}^j_{:,z}$ along the j-th dimension	$\mathcal{W}$	The weight tensor determining the area of a ragged tensor

where $\mathbf{H}_i \in \mathbb{R}_+^{R \times P_i}$ is the i-th the specific factor matrix. This framework enabled the integrative clustering of multiple high-dimensional molecular datasets, effectively addressing challenges posed by varying dimensionalities in the data. By accommodating irregular matrices, this generalized NMF approach demonstrated its versatility and established itself as a powerful tool for analyzing multi-modal data.

The PARAFAC2 decomposition [29], introduced by Harshman for factor analytic applications, is a powerful tool for analyzing a collection of matrices $\mathbf{X}_k \in \mathbb{R}^{N_k \times L}$ $(k = 1, \cdots, K)$, where the number of rows N_k can vary across matrices. The ability to handle varying sizes in one mode is the primary advantage of PARAFAC2 decomposition. Given R as the number of dimensions for the decomposition, the PARAFAC2 model is expressed as:

$$\mathbf{X}_k \approx \mathbf{U}_k \mathbf{S}_k \mathbf{V}^\top, \quad k = 1, 2, \cdots, K,$$

where $\mathbf{U}_k$ is an $N_k \times R$ matrix, $\mathbf{S}_k$ is an $R \times R$ diagonal matrix, and $\mathbf{V}$ is an $L \times R$ factor matrix shared across all k. This flexibility in accommodating matrices with varying row sizes has made PARAFAC2 decomposition particularly valuable for irregular data analysis. Recently, it has been applied across diverse fields, including robust health data analysis [23] and accurate stock prediction [30]. These applications demonstrate the effectiveness of PARAFAC2 in handling datasets where traditional tensor decomposition methods fall short due to irregularities in data structure.

Notably, HO-GSVD [26], intNMF [27], and PARAFAC2 [29] or its variants, all consider irregular tensors whose size varies along a single mode. This limits their application scope in many tasks. Alternatively, many works factorize such data and build connections by enforcing that they share the same factors, such as link prediction in heterogeneous data [45], and stock prediction with multimodal data [46]. While these methods have achieved considerable success, to the best of our knowledge, there remains an absence of a tensor decomposition framework that directly handles the variable geometry of ragged tensors as formulated in Eq. (3). This gap poses a significant challenge to the analysis of datasets with such highly irregular structures.

C. Sparse and Weighted CP Decompositions

A natural strategy to handle ragged data is to embed the irregular domain into a minimal bounding rectangular tensor and apply a weighted decomposition model that excludes out-of-domain indices. This formulation draws a close connection to two well-studied paradigms on *regular* tensors: (i) CP algorithms for sparse tensors and (ii) weighted CP formulations for incomplete data.

Sparse tensor decomposition primarily targets computational efficiency for large-scale *regular* tensors that contain many meaningful zero-valued entries. Such methods exploit sparse storage and kernels to accelerate key operations (e.g., MTTKRP) by avoiding arithmetic on explicitly stored zeros [47], [48], [49], [50]. Other works address the semantic ambiguity of zeros in implicit feedback and assign positive weights to zero entries so that they still influence the factorization [51], [52]. Related approaches impose sparsity priors directly on factor matrices via regularization [53], [54], [55]. Importantly, these methods typically assume the tensor exists on a full rectangular grid, where zeros are treated as *valid* numerical observations that constrain the fitted factors, unavoidably introducing biases when applied to ragged data.

Regarding weighted tensor decomposition [56], [31], [57], our focus is on CP-type formulations. The method most relevant to the ragged setting is CP-WOPT [31], which fits a weighted CP model using a binary mask to ignore missing entries. While algebraically feasible for ragged tensors, such generic weighted solvers typically employ holistic optimization strategies, such as first-order nonlinear conjugate gradient

(NCG) methods over full factor matrices. Crucially, such solvers do not explicitly leverage the separability of the CP model combined with the ragged domain geometry, and holistic first-order optimization often leads to slow convergence or instability (See discussion in Section III-D).

Consequently, a fundamental gap remains in the literature. On one hand, standard sparse methods are semantically ill-suited for ragged tensors, as they enforce a rectangular model that treats non-existent regions as valid zero-valued data. On the other hand, generic weighted solvers (like CP-WOPT), while algebraically capable of handling the ragged tensor, are algorithmically inefficient because they treat irregularity merely as a nuisance to be masked rather than a structural property to be leveraged. These limitations suggest that the variable geometry of ragged tensors requires a tailored decomposition strategy that actively exploits the ragged support. This motivates the separable decomposition framework presented in the following section.

III. Separable Decomposition for Ragged Tensors

Although directly processing a ragged tensor with conventional methods designed for standard tensors is challenging, we make it possible by exploiting the separable nature of the CP decomposition.

A. Separable Decomposition

Reviewing the CP decomposition from the separable view, it can be element-wisely formulated as:

$$x_{n_1,n_2,\cdots,n_m} = \sum_{z=1}^{Z} \lambda_z a^1_{n_1,z} a^2_{n_2,z} \cdots a^m_{n_m,z}, \tag{4}$$

where $x_{n_1,n_2,\cdots,n_m}$ denotes the $(n_1, n_2, \cdots, n_m)$-th entry of $\mathcal{X}$ and $a^j_{n_j,z}$ denote the n_j-th element of $\mathbf{a}^j_{:,z}$ for $j = 1, 2, \cdots, m$. According to (4), each entry $x_{n_1,n_2,\cdots,n_m}$ can be represented as a combination of Z values, where each value involves the product of the scalar λ_z and m variables $a^1_{n_1,z}, a^2_{n_2,z}, \cdots, a^m_{n_m,z}$ based on a separable decomposition framework. This can be interpreted as $x_{n_1,n_2,\cdots,n_m}$ being directly governed by the combination of Z products of variables $\{a^j_{n_j,z}\}_{j=1,z=1}^{j=m,z=Z}$, which implies a unique separability, that naturally matches the irregular geometry of ragged tensors, we can utilize.

We first define the index set of entries for a ragged tensor as $\Omega \triangleq \{(n_1, \cdots, n_m) \mid n_j \in \mathbb{N}^+$ and $\mathcal{N}^h_j(n_1, \cdots, n_{j-1}, n_{j+1}, \cdots, n_m) \leq n_j \leq \mathcal{N}^t_j(n_1, \cdots, n_{j-1}, n_{j+1}, \cdots, n_m)$ for $j = 1, \cdots, m\}$. Let

$$N_j = \max_{\{n_1,\cdots,n_{j-1},n_{j+1},\cdots,n_m\}} \mathcal{N}^t_j(n_1,\cdots,n_{j-1},n_{j+1},\cdots,n_m)$$

for $j = 1, 2, \cdots, m$. The variables $\{a^j_{n_j,z}\}_{j=1}^m$ can be organized into matrices $\mathbf{A}_j$s, i.e.,

$$\mathbf{A}_j \in \mathbb{R}^{N_j \times Z} = \begin{bmatrix} a^j_{1,1} & a^j_{1,2} & \cdots & a^j_{1,Z} \\ a^j_{2,1} & a^j_{2,2} & \cdots & a^j_{2,Z} \\ \vdots & \vdots & \ddots & \vdots \\ a^j_{N_j,1} & a^j_{N_j,2} & \cdots & a^j_{N_j,Z} \end{bmatrix}, \tag{5}$$

for $j = 1, 2, \cdots, m$. Thus, our separable decomposition model for ragged tensors is formulated as

$$\min_{\{\mathbf{A}_j\}_{j=1}^m, \boldsymbol{\lambda}} \sum_{(n_1,\cdots,n_m)\in\Omega} \left| x_{n_1,\cdots,n_m} - \sum_{z=1}^{Z} \lambda_z \prod_{j=1}^{m} \mathbf{A}_j(n_j, z) \right|^2 \tag{6}$$

To achieve a more concise representation, we introduce a weight tensor $\mathcal{W}$ where

$$\mathcal{W}_{n_1,n_2,\cdots,n_m} = \begin{cases} 1, & \text{if } (n_1, n_2, \cdots, n_m) \in \Omega, \\ 0, & \text{otherwise.} \end{cases}$$

thereby clearly distinguishing the valid and invalid regions of ragged tensors using weights of one and zero, respectively. Then (6) can be compactly written as

$$\min_{\{\mathbf{A}_j\}_{j=1}^m, \boldsymbol{\lambda}} \left\| \mathcal{W} \circledast \left(\tilde{\mathcal{X}} - [\![\boldsymbol{\lambda}; \mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m]\!] \right) \right\|_{\mathrm{F}}^2, \tag{7}$$

where $\circledast$ denotes the element-wise (Hadamard) multiplication, and $\tilde{\mathcal{X}}$ is a virtual regular tensor that constitutes the smallest bounding box enclosing the ragged tensor. Those elements in $\tilde{\mathcal{X}}$ corresponding to the zero value in $\mathcal{W}$ serve only to pad the tensor to a hyper-rectangular shape for algebraic compatibility.

It is clear that for a regular tensor, all the entries of $\mathcal{W}$ are 1 and (7) degenerates to the classical CP decomposition for regular tensors.

B. Optimization

In this subsection, we propose a proximal alternating minimization (PAM) algorithm [58] to optimize (7). The factor matrices $\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m$ are updated alternately. In the PAM scheme, the subproblem for updating $\mathbf{A}_j$ $(j = 1, 2, \cdots, m)$, with other variables fixed at the k-th iteration is formulated as

$$\mathbf{A}_j^{k+1} = \arg\min_{\mathbf{A}_j} \left\| \mathbf{W}_{(j)} \circledast \left(\tilde{\mathbf{X}}_{(j)} - \mathbf{A}_j \mathbf{D}_j \right) \right\|_{\mathrm{F}}^2 + \rho \|\mathbf{A}_j - \mathbf{A}_j^k\|_{\mathrm{F}}^2, \tag{8}$$

where $\mathbf{D}_j = \boldsymbol{\Lambda}(\mathbf{A}_m^{k+1} \odot \cdots \odot \mathbf{A}_{j+1}^{k+1} \odot \mathbf{A}_{j-1}^{k} \odot \cdots \odot \mathbf{A}_1^{k})^\top$, the superscript k denotes the iteration number, ρ is a non-negative parameter, $\odot$ is the Khatri–Rao product, and $\mathbf{W}_{(j)}$ is the mode-j unfolding matrix of $\mathcal{W}$.

Solving (8) directly is challenging on account of the presence of a weight matrix. The presence of $\mathbf{W}_{(j)}$ (via the Hadamard product) disrupts the clean block structure of the matricized-tensor-times-Khatri-Rao-product (MTTKRP), causing the entries within the factor matrix to become algebraically entangled. This coupling explains why generic masked least-squares objectives (e.g., CP-WOPT [31]) are typically solved with holistic first-order schemes (such as the nonlinear conjugate gradient (NCG) method) rather than simple alternating least squares. However, specifically for ragged tensors, the separability of the CP decomposition and the optimization objective in (8) enables a geometry-aware optimization approach.

Since the weighting matrix $\mathbf{W}_{(j)} \in \mathbb{R}^{N_j \times \Pi_{i\neq j} N_i}$ is element-wisely multiplied by $\left(\tilde{\mathbf{X}}_{(j)} - \mathbf{A}_j \mathbf{D}_j \right)$ in (8) and the squared Frobenius norm implies separability across rows, we can decouple the optimization of $\mathbf{A}_j$ into independent row-wise

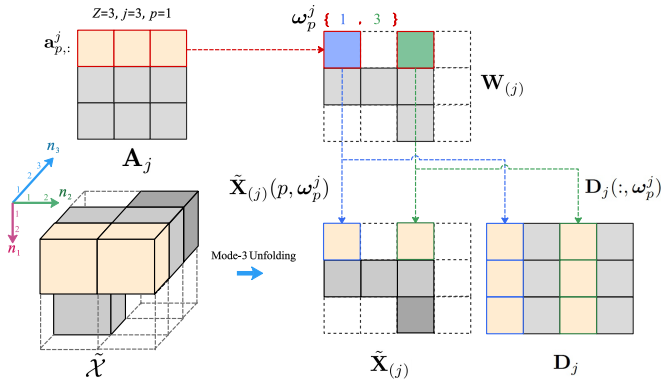


Fig. 3. Illustration of a single row-wise update for our separable factorization. The toy example is of the p-th row-wise update for $\mathbf{A}_j$ with p=1, j=3, Z=3. The mask row yields ω^j_p, which navigates the gather of $\tilde{\mathbf{X}}_{(j)}(p,\omega^j_p)$ and $\mathbf{D}_j(:,\omega^j_p)$. All irrelevant entries (gray) are ignored, making the computation geometry-aware and efficient.

subproblems. First, we rewrite $\mathbf{A}_j \in \mathbb{R}^{N_j \times Z}$ in the row vector form as

$$\mathbf{A}_j = \begin{bmatrix} \mathbf{a}^j_{1,:} \\ \mathbf{a}^j_{2,:} \\ \vdots \\ \mathbf{a}^j_{N_j,:} \end{bmatrix},$$

where $\mathbf{a}^j_{p,:} \in \mathbb{R}^{1\times Z}$ is the p-th row of $\mathbf{A}_j$ $(p = 1, 2, \cdots, N_j)$. Similarly, $\mathbf{D}_j \in \mathbb{R}^{Z \times \Pi_{i\neq j} N_i}$ can be also rewritten in column vector form as

$$\mathbf{D}_j = \left[\mathbf{d}^j_{:,1}, \mathbf{d}^j_{:,2}, \cdots, \mathbf{d}^j_{:,\Pi_{i\neq j} N_i}\right],$$

where $\mathbf{d}^j_{:,q} \in \mathbb{R}^{Z\times 1}$ is the q-th column of $\mathbf{D}_j$ $(q = 1, 2, \cdots, \Pi_{i\neq j} N_i)$.

Considering the p-th row of $\mathbf{W}_{(j)}$, the minimization problem in (8) can be decoupled as

$$\mathbf{a}^{j,k+1}_{p,:} = \arg\min_{\mathbf{a}^j_{p,:}} \left\| \mathbf{W}_{(j)}(p,:) \circledast \left(\tilde{\mathbf{X}}_{(j)}(p,:) - \mathbf{a}^j_{p,:}\mathbf{D}_j\right)\right\|^2_{\mathrm{F}} + \rho \left\|\mathbf{a}^j_{p,:} - \mathbf{a}^{j,k}_{p,:}\right\|^2_{\mathrm{F}}. \tag{9}$$

where $\mathbf{X}(\boldsymbol{i}_{\text{row}}, \boldsymbol{j}_{\text{column}})$ extracts rows indexed by $\boldsymbol{i}_{\text{row}}$ and columns indexed by $\boldsymbol{j}_{\text{column}}$ from the matrix $\mathbf{X}$. Then, to be more compact, denoting the indices of non-zero entries in the p-th row of $\mathbf{W}_{(j)}$ as $\boldsymbol{\omega}^j_p$, we reduce (9) to the following form

$$\mathbf{a}^{j,k+1}_{p,:} = \arg\min_{\mathbf{a}^j_{p,:}} \left\| \tilde{\mathbf{X}}_{(j)}(p,\boldsymbol{\omega}^j_p) - \mathbf{a}^j_{p,:}\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\right\|^2_{\mathrm{F}} + \rho \left\|\mathbf{a}^j_{p,:} - \mathbf{a}^{j,k}_{p,:}\right\|^2_{\mathrm{F}}. \tag{10}$$

To clarify this step, an illustrative example is presented in Figure 3.

Then, the optimal solution of (10) is given by

$$\mathbf{a}^{j,k+1}_{p,:} = \left(\tilde{\mathbf{X}}_{(j)}(p,\boldsymbol{\omega}^j_p)\left(\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\right)^\top + \rho\mathbf{a}^{j,k}_{p,:}\right) \left(\rho\mathbf{I} + \mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\left(\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\right)^\top\right)^{-1}, \tag{11}$$

Algorithm 1 Proximal alternating minimization for the CP-based separable ragged tensor decomposition (RTD-PAM)

Input: The observed m-th order ragged tensor $\mathcal{X}$ defined in (3), and the CP-rank Z.

Initialization: $\tilde{\mathcal{X}} \in \mathbb{R}^{N_1\times N_2\times\cdots\times N_m}$, $\mathbf{A}_j = \texttt{rand}(N_j \times Z)$, $\boldsymbol{\lambda} = \texttt{ones}(Z \times 1)$.

while not converged and $k \leq 1000$ **do**
  for $j = 1$ to m **do**
    for $p = 1$ to N_j **do**
      Updating the p-th row of $\mathbf{A}_j^{k+\frac{1}{2}}$ with (11).
    end for
    Updating $\mathbf{A}_j^{k+1}$ with normalized columns referring to (13).
    for $z = 1$ to Z **do**
      Updating the z-th element of $\boldsymbol{\lambda}$ with $\lambda_z\|\mathbf{A}_j^{k+\frac{1}{2}}(:,z)\|_2$.
    end for
  end for
end while

Output: The factors $\mathbf{A}_j$ $(j = 1, 2, \cdots, m)$ and $\boldsymbol{\lambda}$.

where $\mathbf{I}$ is the identity matrix. The matrix $\mathbf{A}_j$ $(j = 1, 2, \cdots, m)$ can be updated as

$$\mathbf{A}_j^{k+\frac{1}{2}}(p,:) = \mathbf{a}^{j,k+1}_{p,:} \text{ for } p = 1, 2, \cdots, N_j. \tag{12}$$

Finally, the columns of $\mathbf{A}_j^{k+\frac{1}{2}}$ are then normalized as

$$\mathbf{A}_j^{k+1}(:,z) = \frac{\mathbf{A}_j^{k+\frac{1}{2}}(:,z)}{\left\|\mathbf{A}_j^{k+\frac{1}{2}}(:,z)\right\|_2} \text{ for } z = 1, 2, \cdots, Z. \tag{13}$$

Simultaneously, the z-th element of $\boldsymbol{\lambda}$ is updated with $\lambda_z \left\|\mathbf{A}_j^{k+\frac{1}{2}}(:,z)\right\|_2$ for $z = 1, 2, \cdots, Z$. The pseudo-code of our optimization algorithm is summarized in Algorithm 1. The main cost of Algorithm 1 is to compute $\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\left(\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\right)^\top$ to solve (11) for all rows of $\mathbf{A}_j^{k+1/2}$, and the per-iteration complexity of Algorithm 1 is $O(N_1\cdots N_m Z^2)$.

We note that after decoupling the objective function $\left\|\mathbf{W}_{(j)} \circledast \left(\tilde{\mathbf{X}}_{(j)} - \mathbf{A}_j\mathbf{D}_j\right)\right\|^2_{\mathrm{F}}$ in the form shown in (9), other algorithms can also be employed to solve the CP-type ragged tensor decomposition in (7). For example, the hierarchical alternating least squares (HALS) algorithm in [59] or the block coordinate descent (BCD) algorithm in [60] can be utilized. The choice of the PAM algorithm stems from the decouplable nature of the PAM framework, which enables the optimization subproblem at each alternating step to be decomposed into a collection of smaller, more manageable sub-tasks. This property aligns naturally with the separability of CP decomposition. Leveraging this, our method employs an element-wise optimization scheme, enabling it to flexibly handle the geometric structure in ragged tensors. Meanwhile, in terms of implementation, another advantage of the PAM framework is that the proximal term $\rho\|\mathbf{A}_j - \mathbf{A}_j^k\|^2_{\mathrm{F}}$ improves the stability of the solution process by ensuring the positive definiteness of the matrix $(\rho\mathbf{I} + \mathbf{D}_j(:,\boldsymbol{\omega}^j_p)\mathbf{D}_j(:,\boldsymbol{\omega}^j_p)^\top)$.

C. Convergence Analysis

In this part, we provide the theoretical guarantee for the convergence of our PAM-based algorithm. For convenience, we denote

$$L(\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m) = \left\| \mathcal{W} \circledast \left(\tilde{\mathcal{X}} - [\![\boldsymbol{\lambda}; \mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m]\!] \right) \right\|_{\mathrm{F}}^2, \tag{14}$$

and

$$\begin{aligned} \mathbf{A}_j^{k+1} = \arg\min_{\mathbf{A}_j} \big\{ & M_j(\mathbf{A}_j | \mathbf{A}_j^k) \\ & := L(\mathbf{A}_{1:j-1}^{k+1}, \mathbf{A}_j, \mathbf{A}_{j+1:m}^k) + \rho \|\mathbf{A}_j - \mathbf{A}_j^k\|_{\mathrm{F}}^2 \big\} \end{aligned} \tag{15}$$

for $j = 1, ..., m$, where $\mathbf{A}_{i:j}^k$ $(i \leq j)$ denotes $\mathbf{A}_i^k, \mathbf{A}_{i+1}^k, \cdots, \mathbf{A}_j^k$ for simplicity. To simplify the analysis, we analyze the case in which $\boldsymbol{\lambda}$'s values are fixed as 1 and columns in $\mathbf{A}_j$s $(j = 1, 2, \cdots, m)$ are not normalized. Thus, in this scenario, the sequence $\{\mathbf{A}_1^k, \mathbf{A}_2^k, \cdots, \mathbf{A}_m^k\}$ with respect to k generated by Algorithm 1 is equal to the sequence generated by (15). Next, we establish the global convergence of the sequence generated by (15) as follows.

Theorem 1. *The sequence* $\{\mathbf{A}_1^k, \mathbf{A}_2^k, \cdots, \mathbf{A}_m^k\}$ *generated by* (15) *is bounded, and it converges to a critical point of* $L(\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_m)$ *in* (14)*.*

The alternative updating in (15) for $j = 1, ..., m$ with respect to $k = 1, 2, \cdots$, is factually a special instance of Algorithm 4 proposed in [33]. Therefore, the proof of Theorem 1 confirms Theorem 6.2 in [33], and the roadmap of the proof follows the line of verifying the following conditions:

(i) the Kurdyka-Łojasiewicz (K-Ł) property of L at each point;
(ii) the sufficient decrease condition ((64) in [33]); and
(iii) the relative error condition ((65) and (66) in [33]).

To favor more friendly readability, we only mention the road map in the main manuscript and present the detailed proof of Theorem 1 in Supplementary Materials.

D. Comparison with Sparse and Weighted CP Decompositions

While Eq. (7) is algebraically a masked weighted-CP least-squares objective, treating a ragged tensor as a "regular tensor with missing values" overlooks how an irregular domain interacts with the separability of CP, which in turn affects how efficiently and stably the objective can be optimized. To make this distinction concrete, we use a minimal ragged matrix example (extendable in a straightforward way to higher-order tensors).

1) A Toy Ragged Example and Two Pitfalls of Padding-and-Masking: Consider a 2×2 ragged matrix $\mathbf{X}$ whose valid support is

$$\Omega = \{(1,1), (2,2)\}$$

with values

$$x_{11} = 1, \; x_{22} = 5.$$

The off-diagonals $(1,2)$ and $(2,1)$ are *non-existent* (out of domain). We aim to find a rank-1 factorization $\mathbf{a}\mathbf{b}^\top$ with $\mathbf{a}, \mathbf{b} \in \mathbb{R}^2$.

a) Bias from unmasked zero-padding: When considering the ragged matrix $\mathbf{X}$ as a sparse matrix, one typically treats non-existent entries as zero. If we zero-pad the ragged matrix into

$$\tilde{\mathbf{X}} = \begin{bmatrix} 1 & 0 \\ 0 & 5 \end{bmatrix},$$

the factorization attempts to fit the padded zeros at $(1,2)$ and $(2,1)$ as in-domain observations. Exploiting the sparse storage or kernels to accelerate key operations by avoiding arithmetic on explicitly stored zeros does not change this. Mathematically, a rank-1 model requires $a_1 b_1 \approx 1$ and $a_2 b_2 \approx 5$. This implies $(a_1 b_2)(a_2 b_1) = (a_1 b_1)(a_2 b_2) \approx 5$. However, the padded zeros require $a_1 b_2 \approx 0$ and $a_2 b_1 \approx 0$, creating a contradiction. Consequently, unmasked padding introduces artificial constraints on out-of-domain locations and may inflate the apparent rank and bias the fitted factors.

b) Flat directions under masking/weighting: To remove the padding bias, weighted methods (e.g., CP-WOPT [31]) use a binary mask to ignore padded entries. Let

$$\mathbf{W} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The weighted objective

$$\|\mathbf{W} \circledast (\mathbf{a}\mathbf{b}^\top - \tilde{\mathbf{X}})\|_{\mathrm{F}}^2 \tag{16}$$

correctly constrains only $a_1 b_1 = 1$ and $a_2 b_2 = 5$. However, this introduces a new challenge of *ill-conditioning*. The masked off-diagonals are unconstrained, allowing $a_1 b_2$ and $a_2 b_1$ to diverge or vanish arbitrarily. More concretely, consider the solution family

$$\mathbf{a} = \begin{bmatrix} t \\ 1/t \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 1/t \\ 5t \end{bmatrix}.$$

For any $t \neq 0$, the loss is exactly zero. This non-isolated solution set induces flat/poorly-scaled directions (degenerate curvature) in the objective landscape. We remark that this degeneracy is a property of the masked weighted least-squares objective itself; hence it is present regardless of the solver. The practical difference is how the solver handles these flat/poorly-scaled directions.

2) Holistic Masked Update vs. Separable PAM: CP-WOPT [31] is a representative approach that solves the weighted least-squares formulation via holistic first-order optimization (e.g., nonlinear conjugate gradient) over the full factor matrices. On highly irregular supports, the presence of nearly-flat directions can yield poorly scaled search directions, so holistic updates may progress slowly or oscillate along flat manifolds in practice.

In contrast, our separable PAM scheme explicitly leverages the natural match between CP separability and the ragged domain that each factor row is linked only to its valid support. Accordingly, we decouple each factor update into separable row-wise subproblems (Eqs. (9)-(11)). Moreover, the proximal term makes each row-wise quadratic subproblem strongly convex. That is, the coefficient matrix (e.g., $\rho\mathbf{I} + \mathbf{D}\mathbf{D}^\top$ in our derivation) is strictly positive definite for $\rho > 0$. This yields a unique and numerically stable closed-form solution

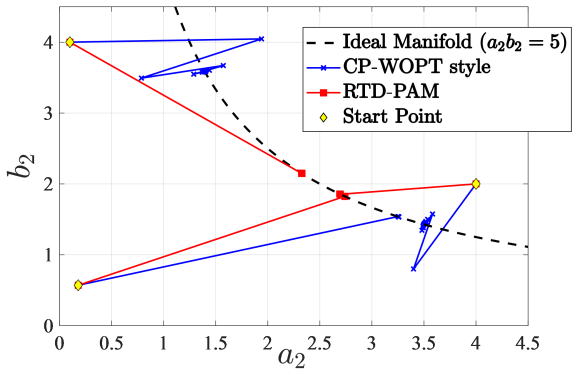


Fig. 4. Optimization trajectories on the toy problem of Eq. (16): a CP-WOPT-style holistic first-order update (blue) versus our RTD-PAM (red). The holistic update may exhibit oscillatory progress along a flat manifold, while the proposed row-wise proximal updates reach the optimal manifold more directly.

for every row update, even when the masked objective exhibits degenerate curvature.

To visualize this effect, we provide an illustrative 2D trajectory plot for the toy ragged instance of the weighted objective (16), comparing[2] (i) a holistic first-order solver following the CP-WOPT principle and (ii) our row-wise proximal updates. Figure 4 shows that the holistic update (blue) can wander along the flat hyperbola $a_2b_2 = 5$, whereas our separable PAM (red) rapidly reaches a point on the optimal manifold due to decoupled closed-form row updates. This qualitative behavior is consistent with (and helps explain in part) the stability and runtime advantages reported in our experiments (see Section IV).

IV. Numerical Experiments

In this section, we present comprehensive experiments to systematically evaluate the effectiveness, robustness, and generalization of our proposed method on various datasets. First, we validate the fundamental performance and noise robustness of our algorithm using synthetic data. These experiments are conducted in a controlled setting characterized with a mild irregularity, where the tensor size varies along only one mode. Second, we compare our method against several state-of-the-art approaches across different real-world tasks, including segment completion for natural and multispectral images, as well as denoising for hyperspectral remote sensing data. Finally, to demonstrate its practical utility in cutting-edge scientific applications, we apply our method to two spatial transcriptomics datasets. This application evaluates our method's ability to handle complex, non-meshgrid data, and showcases its generalization capabilities for challenging biological data. To explicitly illuminate the irregularity type and data structure of different applications, we provide an intuitive summarization of them in Table II. The experiments are implemented on MATLAB R2023a on the Windows 10 platform on a desktop with Intel Core i9-10900K @2.30GHz and 32.00 GB RAM.

[2]For each run, both methods start from the same initialization with i.i.d. entries drawn from $U(0, 5)$. The CP-WOPT-style baseline follows the CP-WOPT [31] principle of optimizing all factor variables jointly via a nonlinear conjugate-gradient (NCG) method for the weighted least-squares objective. We use Polak–Ribière$^+$ updates and an Armijo backtracking line search with initial step $\alpha_0 = 1$ (shrinkage factor 0.5 and sufficient decrease parameter 10^{-4}). Its stopping criterion is set to a gradient norm tolerance of 10^{-6} or a maximum number of iterations.

TABLE II
Overview of Dataset Structure and Irregularity Type in Considered Applications.

Application	Type of Irregularity Challenge
Synthetic Data	
Case 1	Mild Irregularity: Varying sizes in a single mode.
Case 2	Severe Irregularity: Unaligned dimensions and highly varied entry counts.
Real-World Data	
Segment Completion	Spatial Irregularity: Varying sizes in spatial modes.
HSI Superpixel Denoising	Spatial Irregularity: Varying sizes in spatial modes.
Spatial Transcriptomics Recovery	Biologically-driven irregularity: Spatial sampling spots form a non-grid, staggered spatial pattern.

A. Synthetic Data

We first conduct experiments on synthetic ragged third-order tensors, which share the same dimension along two modes. Thus, we can directly compare our method with irregular tensor decomposition methods (TMac [61], TCTF [62], PARAFAC2 [29], intNMF [27] and CP-WOPT [31]). Then, we consider synthetic ragged tensors that no longer maintain the "good" structure of sharing the same dimension along two modes. In this case, we compare our method with several methods designed for regular tensors. Meanwhile, to launch those methods, we pad the ragged tensors into regular tensors by filling in zeros.

In the first case, after generating the ground-truth ragged tensors, we add the Gaussian noise with different signal-to-noise ratios (SNR). In the second case, the ground-truth ragged tensors are partially sampled from a low-rank tensor. It is noteworthy that this is different from tensor completion as the ground-truth tensor contains only those sampled entries, being fragmentary itself.

1) Case 1: In this part, we first evaluate our model's performance and noise robustness in a mildly irregular setting. These initial experiments use third-order synthetic data where the tensor size varies along only one mode. Specifically, we simulate third-order ragged tensors by the following steps.

(i) Three factor matrices $\mathbf{A}_j$s ($j = 1, 2, 3$) are generated with entries uniform-randomly distributed in $[0, 1]$, i.e., $\mathbf{A}_j = \texttt{rand}(N_j \times Z)$ with $Z = 10$, and $[N_1, N_2, N_3] = [100, 100, 40]$.
(ii) Construct a regular tensor $\tilde{\mathcal{X}} = [\![\mathbf{1}; \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\!]$.
(iii) Construct the synthetic third-order ragged tensor $\mathcal{X}$ with its elements obtained by

$$x_{n_1,n_2,n_3} = \tilde{x}_{n_1,n_2,n_3}, \text{ if } 1 \le n_2 \le N_2'(n_3),$$

TABLE III
THE RELATIVE APPROXIMATION ERRORS (IN PERCENTAGES, %) BY DIFFERENT METHODS IN TEST 1. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

SNR (dB)	Noisy	TMac [61]	TCTF [62]	CPD-PAM	PARAFAC2 [29]	intNMF [27]	CP-WOPT [31]	RTD-PAM
20	10.0425	5.9969	5.1888	3.5029	3.8328	3.4441	0.9496	**0.9031**
30	3.1799	3.1683	1.3730	5.4367	1.6083	1.7128	0.3364	**0.2858**
40	1.0077	2.6479	0.4239	1.9183	0.4397	0.5100	0.1435	**0.0906**
Time (s)	—	0.54	1.37	67.61	156.11	9.87	147.31	23.07

where $N_2'(n_3)$ is a function of n_3. For a fixed n_3, $N_2'(n_3)$ is prescribed by an integer value randomly selected in the interval of $[50, 100]$.

(iv) The Gaussian noise is element-wisely added to each x_{n_1,n_2,n_3} with signal-to-noise ratios (SNR) = 20, 30, and 40, to obtain a noisy ragged tensor $\mathcal{X}_{\text{noisy}}$.

Through the above steps, we can get a third-order ragged tensor $\mathcal{X}$ and its noisy version $\mathcal{X}_{\text{noisy}}$. The n_3-th mode-1-by-mode-2 slices of $\mathcal{X}$ (or $\mathcal{X}_{\text{noisy}}$) are rectangular and of the size $N_1 \times N_2'(n_3)$. Then, we apply six compared methods to factorize $\mathcal{X}_{\text{noisy}}$, including: i) TMac [61], a Tucker decomposition based method; ii) TCTF [62], a tensor singular value decomposition based method; iii) CPD-PAM, a CP decomposition method that utilizes the PAM algorithm, which serves as a crucial baseline to our method, demonstrating how the optimization process is adversely affected by padded zeros; iv) CP-WOPT [31], a method representing approaches that handle missing data in regular tensors via weighting, which employs a first-order optimization algorithm that incorporates a binary weighting scheme to nullify the gradient contributions from missing elements; v) PARAFAC2 [29], which can handle irregular tensors with size variations along only a single mode; and vi) intNMF [27], which is similarly restricted to tensors with size variations in one mode. As for TMac and TCTF, which are designed for regular tensors, we pad $\mathcal{X}_{\text{noisy}}$ with zeros to make it the same size as $\tilde{\mathcal{X}}$. PARAFAC2 and intNMF are able to directly handle the synthetic ragged tensor in this case. Then, we compute the relative approximation error, which is computed by

$$\frac{\left(\sum_{n_1=1,n_3=1}^{n_1=100,n_3=40}\sum_{n_2=1}^{n_2=N_2'(n_3)}|x_{n_1,n_2,n_3}^{\text{estimated}}-x_{n_1,n_2,n_3}|^2\right)^{\frac{1}{2}}}{\left(\sum_{n_1=1,n_3=1}^{n_1=100,n_3=40}\sum_{n_2=1}^{n_2=N_2'(n_3)}|x_{n_1,n_2,n_3}|^2\right)^{\frac{1}{2}}}$$

where, $x_{n_1,n_2,n_3}^{\text{estimated}}$ is the (n_1, n_2, n_3)-th element of the approximated tensor after factorization. The rank of the proposed RTD-PAM is set to 10. Meanwhile, to ensure a fair comparison, the parameters of all competing methods were carefully tuned to optimize their performance. The relative approximation errors (in percentages) of the results by different methods in Case 1 are reported in Table III.

Notably, regular tensor decomposition methods cannot handle the synthetic tensor $\mathcal{X}$, even though it possesses a simple structure with variation confined to the second dimension. The inferior performance of these methods stems primarily from approximation errors introduced by zero-padding. Our method delivers the highest accuracy across all SNR levels and consistently outperforms the second-best CP-WOPT baseline in both performance and efficiency. In contrast, although PARAFAC2 and intNMF can factorize the synthetic tensor well, they only utilize a shared matrix in their factorizations to characterize the third mode, limiting their performance.

2) Case 2: In this case, to evaluate the effectiveness and robustness to noise of our method on a ragged tensor whose size varies along all modes, we devise a more challenging scenario, where the ragged tensor is a subset of a regular tensor of the size $100 \times 100 \times 100$ by random sampling from $\tilde{\mathcal{X}}$. Specifically, the synthetic ragged tensor is generated by following steps.

(i) Generate three factor matrices $\mathbf{A}_j$s ($j = 1, 2, 3$) with entries uniform-randomly distributed in $[0, 1]$, i.e., $\mathbf{A}_j = \texttt{rand}(100 \times 10)$,

(ii) Obtain a regular tensor $\tilde{\mathcal{X}} = [\![\mathbf{1}; \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\!]$,

(iii) Construct the index set Ω by randomly selected 30% entries from

$$\left\{(n_1, n_2, n_3) | \, n_j \in \mathbb{N}^+ \text{ and } 1 \leq n_j \leq 100 \text{ for } j = 1, 2, 3\right\},$$

(iv) The synthetic third-order ragged tensor $\mathcal{X}$ is obtained by

$$x_{n_1,n_2,n_3} = \tilde{x}_{n_1,n_2,n_3}, \text{ if } (n_1, n_2, n_3) \in \Omega.$$

(v) The Gaussian noise is element-wisely added to each x_{n_1,n_2,n_3} with signal-to-noise ratios (SNR) = 20, 30, and 40, to obtain a noisy ragged tensor $\mathcal{X}_{\text{noisy}}$

In this case, $\mathcal{X}_{\text{noisy}}$ is highly irregular so that PARAFAC2 [29] and intNMF [27] can no longer handle it. Therefore, we only compare our method with TMac [61], TCTF [62], CPD-PAM, and CP-WOPT [31] by filling zero into Ω^C. The relative approximation errors, which are computed by

$$\frac{\sum_{(n_1,n_2,n_3)\in\Omega}\left(|x_{n_1,n_2,n_3}^{\text{estimated}}-x_{n_1,n_2,n_3}|^2\right)^{\frac{1}{2}}}{\left(\sum_{(n_1,n_2,n_3)\in\Omega}|x_{n_1,n_2,n_3}|^2\right)^{\frac{1}{2}}},$$

of the approximations after factorization by different methods are shown in Table IV. Not surprisingly, the results show that our method can well factorize $\mathcal{X}_{\text{noisy}}$ with robustness to the noise and relatively high efficiency.

B. Real-World Data

In this section, we apply our method to real-world data of ragged tensors. Three tasks are considered, including image segment completion, hyperspectral image denoising, and spatial transcriptomics recovery.

TABLE IV
THE RELATIVE APPROXIMATION ERRORS (IN PERCENTAGES, %) BY DIFFERENT METHODS IN TEST 2. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

SNR (dB)	Noisy	TMac	TCTF	CPD-PAM	CP-WOPT	RTD-PAM
20	10.0012	67.8216	63.3759	69.9258	0.9920	**0.9909**
30	3.1612	67.7506	63.3858	69.8352	0.3138	**0.3077**
40	0.9978	67.8559	63.4032	69.7203	0.1106	**0.1019**
Time (s)	—	1.18	4.26	7.86	121.02	17.20

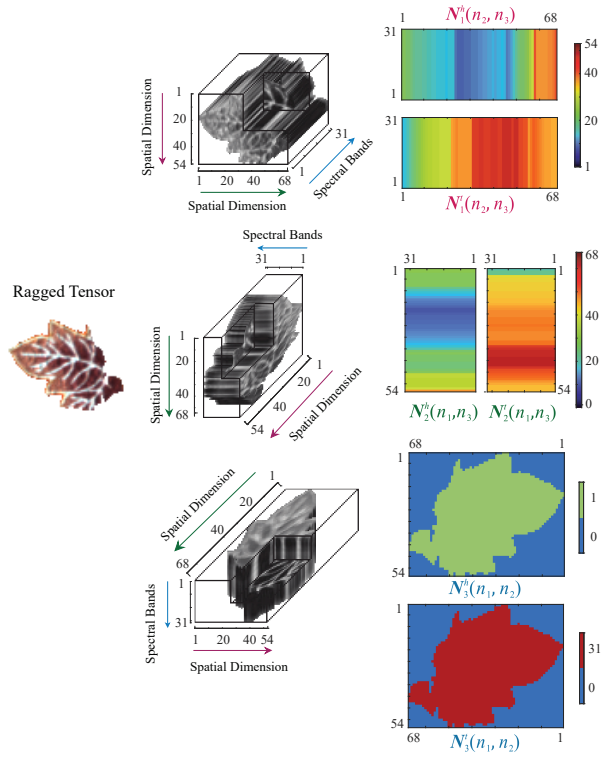


Fig. 5. An illustration of the head ($\boldsymbol{N}_j^h$) and tail ($\boldsymbol{N}_j^t$) index tables (Eq. (2)) for a spatially irregular segment of the MSI *Cloth*. Three cross-sectional views of the ragged tensor are shown, with red, green, and blue arrows indicating the first, second, and third dimensions, respectively. Additionally, colormaps for the index tables are provided, where different colors represent different index values. Due to the uniform spectral length in the third dimension, the tables $\boldsymbol{N}_3^h$ and $\boldsymbol{N}_3^t$ contain only two unique values, leading to simple two-color maps.

1) Image Segment Completion: In this part, we employ ragged tensors derived from real-world images via segmentation to assess our method's capability for segment-level reconstruction. The segments[3] are obtained using segmentation techniques from multispectral images (MSIs) and color images. First, two MSIs, *Cloth* and *Pompoms*, from the CAVE data set[4] are considered and corresponding segments are extracted by the superpixel segmentation [63]. Figure 5 illustrates one such segment, showing its part-sectioned views alongside colormaps of its head and tail index tables for each dimension. We can see that this segment maintains explicit semantic information.

In this situation, although the spectral mode is in a fixed dimension, PARAFAC2 [29], and intNMF [27] can not handle them. Therefore, we compare our method with four aforementioned methods, i.e., TMac [61], TCTF [62], CPD-PAM, as well as CP-WOPT [31], and one matrix nuclear norm minimization low-rank matrix recovery method [64], denoted as "LRMR". To facilitate the implementation of these methods, their input is set as a regular tensor, which is the smallest cube cropped from the multispectral image that contains the segment. This can be viewed as padding the ragged tensor with contextual information. LRMR can directly cope with the matrix obtained by stacking spectral vectors (mode-3 fibers) from the ragged tensor or the mode-3 unfolding matrix of the standard cube containing the segment. Thus, we test both settings and denote them as "LRMR-RAG" for the former one and "LRMR-STD" for the latter one, respectively.

To evaluate the effectiveness of different methods, we focus on the completion task (filling in missing values within the valid boundary) for these segments. Our method and the others need to recover the segments from their partial entries, which are sampled[5] uniformly at random with the sampling rate (SR) of 0.1. To ensure fairness, partial sampling is also applied to the complementary areas in the standard cube surrounding the segment, maintaining the same SR. This setup can be viewed as a low-rank tensor completion task for ragged tensors. We use the relative error (RE) and peak signal-to-noise ratio (PSNR) to measure the recovery ability of approximation results after factorization. The RE is computed in the same way as the relative approximation error in Section IV-A2, with Ω defined as the index set of the segment. Given the ground-truth (GT) data, the PSNR is calculated as follows

$$10 \cdot \log_{10} \frac{\left(\max_{(n_1,n_2,n_3)\in\Omega} x^{\text{estimated}}_{n_1,n_2,n_3}\right)^2}{\frac{1}{|\Omega|}\sum_{(n_1,n_2,n_3)\in\Omega}\left(x^{\text{estimated}}_{n_1,n_2,n_3} - x^{\text{gt}}_{n_1,n_2,n_3}\right)^2}.$$

Table V presents the numerical results. Our method consistently outperforms competing methods in both RE and PSNR values. The visual results, as shown in Fig. 6, further illustrate and confirm the superior performance of our method. Meanwhile, CP-WOPT achieves second-best performance but with the highest time cost. The matrix-based methods, LRMR-STD and LRMR-RAG, outperform the remaining methods, likely because their minimization of the matrix nuclear norm provides greater robustness to the low sampling rate (0.1 in Table V) than factorization-based methods. Additionally, the performance of LRMR-RAG, which focuses solely on the

[3]We remark here that, when processing those regular pictures (or multispectral images), our attention focuses on the objects of interest. Previously, people could only deal with those regular pictures (containing those objects of interest) and construct many sophisticated techniques, e.g., deep neural networks, to force their algorithms to concentrate on specific areas. From this perspective, it would be better to deal directly with those interested parts if possible. However, previous methods cannot accomplish this. Therefore, we test our method on those segments to validate that it can directly handle them well.

[4]https://cave.cs.columbia.edu/repository/Multispectral

[5]Notably, the sampling process here differs from that in Section IV-A2, where the sampled entries are used to construct a highly irregular tensor. In this section, the ragged tensor (or segment) has already been formed, and sampling is used to create partial observations for testing the recovery capabilities of various methods on the ragged tensor.

TABLE V
THE QUANTITATIVE RESULTS BY DIFFERENT METHODS ON THE SEGMENTS OF MSIS (*Cloth* AND *Pompoms*) FOR COMPLETION WITH THE SAMPLING RATE = 10%. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

	Cloth						*Pompoms*						
Segments													
Metrics	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	Time (s)
Observed	0.949	8.96	0.950	8.97	0.948	10.89	0.950	7.90	0.949	16.66	0.949	11.13	—
TMac [61]	0.887	9.54	0.897	9.47	0.886	11.47	0.864	8.73	0.848	17.64	0.930	11.31	0.64
TCTF [62]	0.882	9.59	0.887	9.56	0.880	11.54	0.878	8.59	0.898	17.14	0.896	11.63	0.73
CPD-PAM	0.874	9.67	0.876	9.68	0.866	11.67	0.883	8.54	0.865	17.47	0.866	11.93	39.94
LRMR-STD [64]	0.307	18.77	0.310	18.69	0.312	20.55	0.481	13.82	0.527	21.77	0.368	19.37	3.86
LRMR-RAG [64]	0.281	19.52	0.277	19.69	0.275	21.64	0.430	14.78	0.379	24.63	0.320	20.59	1.20
CP-WOPT [31]	0.159	24.45	0.157	24.59	0.214	23.82	0.046	34.17	0.100	36.20	**0.036**	**39.51**	55.28
RTD-PAM	**0.141**	**25.52**	**0.144**	**25.35**	**0.155**	**26.61**	**0.036**	**36.27**	**0.072**	**39.10**	**0.036**	**39.51**	10.72

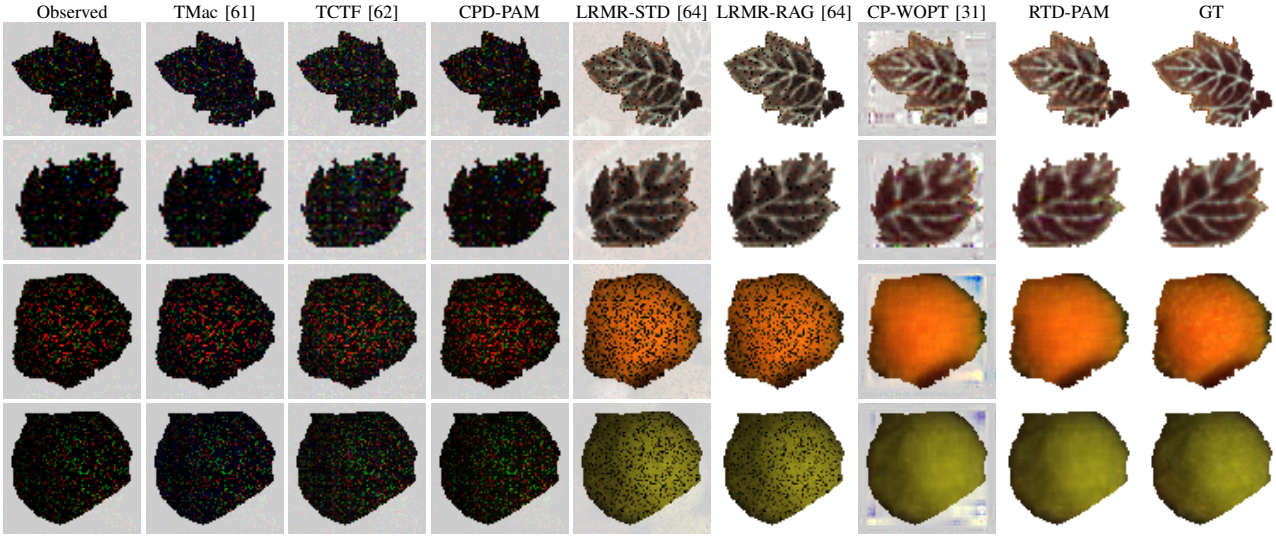


Fig. 6. Pseudo-color images (R-30 G-20 B-10) of restored results by various methods on segments of the MSIs *Cloth* (the first two rows) and *Pompoms* (the last two rows) with SR = 10%. To vividly illustrate the segments, the complementary areas that make the ragged tensors regular are covered by the gray color.

segment, surpasses that of LRMR-STD, indicating that the contextual information in the surrounding cube areas does not aid recovery in this case due to the homogeneity within the segment entries. To further investigate this, we introduce an extreme case. Here, the spectral vectors (mode-3 fibers) of the first segment in MSI *Cloth* are replaced by identical vectors, generated by evenly spacing values between 0 and 1. Meanwhile, the supplementary areas in the regular cube remain unchanged. The first column in Table VI displays the pseudo-color image of the regular cube, with the orange region indicating the ragged tensor. We then perform low-rank tensor completion on this extremely homogeneous ragged tensor with SRs of 0.05 and 0.1, with the quantitative results presented in Table VI. As shown in Table VI, our method almost perfectly reconstructs the simple-structured ragged tensor while the additional padding pixels introduced to convert the ragged tensor into a regular tensor severely disrupted the performance of the other methods in comparison.

Next, we employ two color images (CIs) from the Visual Object Classes Challenge 2012 (VOC2012)[6] and two additional MSIs *Flower* and *Toy* from the CAVE data set. We create four segments from these test images using semantic segmentation techniques [65]. These segments are illustrated in the last column of Fig. 7 and are respectively named *chair*, *sofa*, *flower*, and *toy* based on their semantic information. Sampling rates are set to 0.4 and 0.5 for the color image segments *chair* and *sofa*, and 0.05 and 0.1 for the MSI segments *flower* and *toy*. Corresponding regular cubes are generated by cropping from the original images. The methods used for comparison remain the same as in previous sections. Table VII exhibits the RE and PSNR values and Fig. 7 displays

[6]http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

TABLE VI
THE RE VALUES OF RESULTS BY DIFFERENT METHODS ON THE SPECIAL CASE FOR SEGMENT COMPLETION WITH DIFFERENT SRS. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

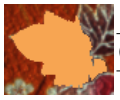
Data	SR	Observed	TMac	TCTF	CPD-PAM	LRMR-STD	LRMR-RAG	RTD-PAM
	0.05	0.975	0.969	0.824	0.936	0.578	0.523	**0.018**
	0.1	0.949	0.928	0.900	0.868	0.337	0.234	**0.005**

TABLE VII
THE QUANTITATIVE RESULTS ON FOUR SEGMENTS OF COLOR IMAGES (CIS) AND MSIS. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

Segment		Observed			TMac [61]		TCTF [62]		CPD-PAM		LRMR-STD [64]		LRMR-RAG [64]		CP-WOPT [31]		RTD-PAM	
		SR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR	RE	PSNR
CI	*chair*	0.4	0.775	8.38	0.551	11.345	0.552	11.32	0.504	12.11	0.530	11.67	0.501	12.16	0.098	26.30	**0.090**	**27.06**
		0.5	0.705	9.20	0.456	12.980	0.460	12.91	0.392	14.30	0.426	13.57	0.392	14.29	0.087	27.38	**0.079**	**28.19**
	sofa	0.4	0.776	7.71	0.520	11.19	0.522	11.14	0.488	11.73	0.469	12.08	0.468	12.10	0.036	34.36	**0.032**	**35.32**
		0.5	0.705	8.53	0.406	13.34	0.405	13.35	0.373	14.06	0.353	14.55	0.352	14.58	0.034	35.00	**0.029**	**36.27**
MSI	*flower*	0.05	0.975	11.66	0.955	11.85	0.922	12.15	0.941	11.98	0.619	15.61	0.569	16.35	0.139	28.61	**0.125**	**29.52**
		0.1	0.949	11.90	0.900	12.36	0.903	12.33	0.868	12.67	0.399	19.42	0.329	21.11	0.103	31.21	**0.101**	**31.33**
	toy	0.05	0.975	10.05	0.953	10.25	0.934	10.43	0.934	10.42	0.589	14.44	0.591	14.41	0.125	27.87	**0.107**	**29.24**
		0.1	0.948	10.30	0.918	10.58	0.914	10.62	0.863	11.11	0.360	18.72	0.355	18.84	**0.089**	**30.80**	0.090	30.76
Time (s)			—		0.65		2.64		77.22		29.37		9.65		298.60		67.62	

Fig. 7. Visual results by various methods on *semantic units* of the color images *Chair*, *Sofa* (the first two rows, SR = 50%), and MSIs *Flower* and *Toy* (the last two rows, SR = 5%). To vividly illustrate the segments, the supplementary areas for composing the regular tensors are covered by the gray color.

the visual results. For the MSI segments *flower* and *toy*, although a different segmentation technique is applied, the results remain largely consistent with previous observations. For the color image segments *chair* and *sofa*, the performance of LRMR-STD and LRMR-RAG is comparable, showing no clear advantage over the first three methods. This reduced advantage for color image segments is due to the third dimension being limited to 3, which restricts the preservation of inner structure in the mode-3 unfolding matrix (or the matrix formed by stacking mode-3 fibers). While CP-WOPT outperforms the other methods except for ours, its computational cost remains extremely high. In most cases, our method achieves the best performance in both numerical metrics and visual quality with a lower time cost.

2) HSI Denoising: To further evaluate the noise robustness of the proposed method on real-world data, we apply it to Gaussian noise removal for hyperspectral images (HSIs) used in remote sensing. HSI denoising is a typical inverse problem where the goal is to recover a clean HSI from a noisy observation. A common approach is low-rank tensor decomposition (or minimizing the nuclear norm of the unfolded matrix), which exploits the low-rank structure of the HSI to effectively mitigate noise. From the experimental results in the previous part, it is evident that when the spectral image is pre-segmented into superpixels, grouping similar local spectral vectors together, the internal homogeneity within these superpixels is higher than in the image as a whole, corresponding to lower-rankness. Our CP-based ragged tensor decomposition method effectively capitalizes on this property. Therefore, we first apply the superpixel segmentation method [63] to the noisy HSI, clustering similar spectral vectors together. We then perform ragged tensor decomposition on each superpixel

TABLE VIII
THE QUANTITATIVE RESULTS BY DIFFERENT METHODS ON HSIS (*Pavia City Center* AND *Washington DC Mall*) FOR HSI DENOISING WITH THE NOISE LEVEL $\sigma = 20$. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

HSI		Observed	TMac	TCTF	CPD-PAM	LRMR	RTD-PAM
Pavia	RE	0.348	0.242	0.227	0.197	0.296	**0.134**
	PSNR	22.11	25.42	25.90	27.14	23.73	**30.42**
DC Mall	RE	0.270	0.221	0.267	0.197	0.220	**0.104**
	PSNR	22.11	24.64	22.87	25.77	24.71	**30.44**
Time (s)		—	3.4	21.7	490.1	107.3	98.5

individually, obtaining a low-rank approximation for each segment, and finally combine these approximations to produce the denoised result. In contrast, the methods considered in the previous section can directly process the entire HSI as a regular tensor. These methods do not perform as well on irregular segments, so for a fair comparison, they are applied directly to the noisy HSI as a whole.

We select two HSIs: a $200 \times 200 \times 80$ (spatial height $\times$ spatial width $\times$ spectrum) subimage from the *Pavia City Center* dataset[7] and a $256 \times 256 \times 160$ subimage from the *Washington DC Mall* dataset[8]. The noisy HSI is generated by adding zero-mean Gaussian noise with a standard deviation (noise level) of $\sigma = 20$ (provided that the entry values in HSI are normalized into the interval of $[0, 255]$). We also adopt the RE and PSNR to measure the quantitative quality of results, both computed over the entire HSI.

Table VIII presents the quantitative denoising results obtained by all methods, while Fig. 8 displays the pseudo-color images created by stacking the 65-th, 45-th, and 15-th spectral bands. The results clearly demonstrate that our method accurately reconstructs the fine details of the HSIs, outperforming other methods that process the HSI in an integral (or holistic) manner. Although LRMR performs well in preserving details, it shows noticeable color deviations compared to the clean HSI.

3) Spatial Transcriptomics Recovery: In this part, we demonstrate our method's practical utility and its ability to handle complex, non-meshgrid data using datasets from spatial transcriptomics (ST) [24], [25]. ST is a cutting-edge technique that quantitatively maps the spatial distribution of polyadenylated transcripts within tissue sections. This method combines barcoded oligo-dT arrays with standard histological brightfield imaging to generate high-resolution spatial maps of gene expression, providing valuable insights into the spatial organization of transcriptomics across diverse biological contexts. As illustrated in Fig. 1, the spatial spots are arranged in a staggered pattern across rows, resembling an "orange crate packing" configuration to increase the packing density. Meanwhile, the spatial region of the tissue exhibits an irregular shape, making this a particularly challenging scenario. In this

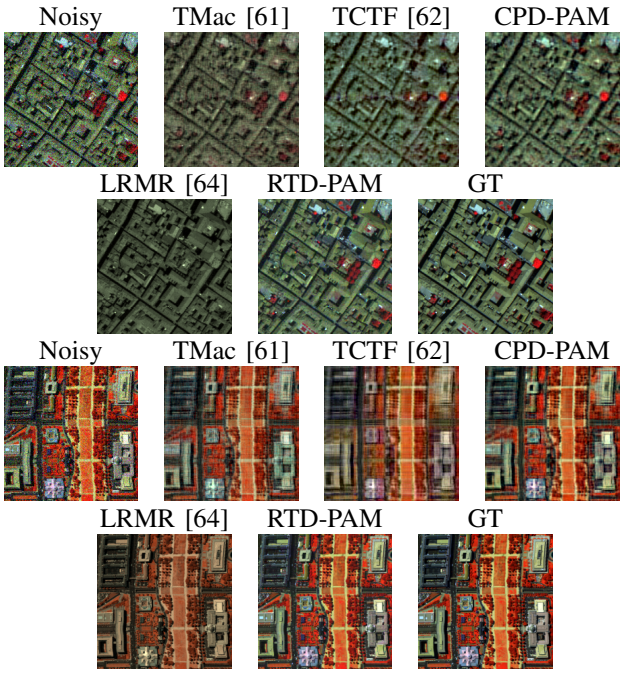


Fig. 8. Pseudo-color images (R-65 G-45 B-15) by different methods on HSIs Pavia City Center (top two rows) and Washington DC Mall (bottom two rows) for HSI denoising with noise level $\sigma = 20$.

part, we select the *FFPE Mouse Brain*[9] and *FFPE Mouse Kidney*[10] datasets from 10x Genomics as the testing data. The datasets are preprocessed using the Python package Scanpy[11], which is recommended on the website where the datasets are hosted. After preprocessing, we obtain two highly irregular third-order tensors, each with two spatial modes and one gene mode.

During the capture of spatial transcriptomics data, certain primers for specific genes may lose efficacy, leading to missing data. Therefore, we first consider the recovery of spatial transcriptomics data with missing entries, which correspond to these inactive primers. The two irregular third-order tensors obtained from the spatial transcriptomics data are set as the ground-truth data. Random missing entries are introduced with a missing ratio of 50%, producing the observed incomplete data. Due to the "orange crate packing" configuration, which complicates the implementation of both our CP-based ragged tensor decomposition method and regular tensor decomposition methods, each irregular tensor is split into two parts by selecting the odd and even rows separately. After this splitting, the tensors remain spatially irregular and resemble the image segments discussed in the previous section. For regular tensor decomposition methods, zeros are padded to make the spatial areas rectangular. Once the recovery results are obtained, the two splits are combined to form the final reconstructed tensors. We evaluate the recovery performance using RE and R-Square

[7]https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

[8]https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html

[9]Spatial Gene Expression from C57BL/6 mouse (B1), Adult Mouse Brain (FFPE) dataset by spaceranger, v1.3.0, 10x Genomics, (2021, August 16). https://www.10xgenomics.com/datasets/adult-mouse-brain-ffpe-1-standard-1-3-0

[10]Spatial Gene Expression from C57BL/6 mouse (C1), Adult Mouse Kidney (FFPE) dataset by spaceranger, v1.3.0, 10x Genomics, (2021, August 16). https://www.10xgenomics.com/datasets/adult-mouse-kidney-ffpe-1-standard-1-3-0

[11]https://scanpy.readthedocs.io/en/stable/

TABLE IX
THE QUANTITATIVE RESULTS BY DIFFERENT METHODS ON VARIOUS GENES OF TWO SPATIAL GENE EXPRESSION DATASETS (*FFPE Mouse Brain* AND *FFPE Mouse Kidney*) WITH THE MISSING RATIO = 50%. THE **BEST** VALUES ARE HIGHLIGHTED IN BOLD.

Data	*FFPE Mouse Brain*						*FFPE Mouse Kidney*						
Genes	Cplx1		Cnp		Trf		Gatm		Nccrp1		Calb1		Time (s)
Methods	RE	RS	RE	RS	RE	RS	RE	RS	RE	RS	RE	RS	
Observed	0.230	—	0.444	—	0.577	—	0.466	—	0.674	—	0.552	—	—
TMac [61]	0.203	-0.024	0.359	0.140	0.530	-0.115	0.365	0.161	0.528	0.119	0.479	-0.131	0.88
TCTF [62]	0.214	-0.137	0.373	0.073	0.520	-0.075	0.377	0.110	0.538	0.085	0.463	-0.056	12.38
CPD-PAM	0.178	0.216	0.301	0.396	0.376	0.172	0.318	0.366	0.416	0.455	0.366	0.341	34.12
LRMR [64]	0.180	0.199	0.196	0.743	0.256	0.740	0.206	0.735	0.252	0.799	0.223	0.755	31.20
CP-WOPT [31]	**0.098**	0.759	0.148	0.855	0.189	0.858	0.124	0.903	**0.160**	**0.919**	0.166	0.864	10515.20
RTD-PAM	**0.098**	**0.764**	**0.145**	**0.859**	**0.185**	**0.864**	**0.123**	**0.905**	**0.160**	**0.919**	**0.164**	**0.867**	104.25

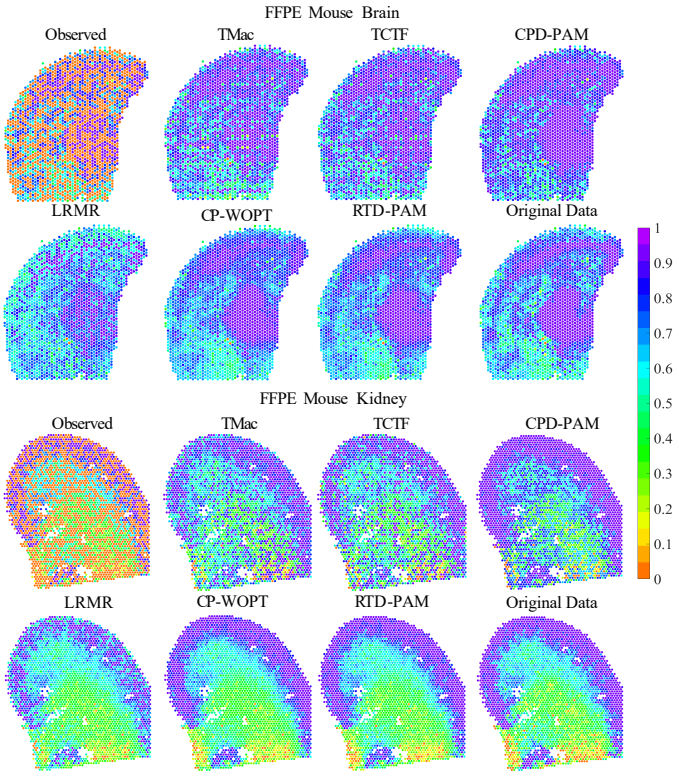


Fig. 9. Visual results by different methods on gene *Cplx1* of *FFPE Mouse Brain* and gene *Gatm* of *FFPE Mouse Kidney* for spatial gene expression recovery (missing ratio = 50%).

(RS) values. The RS value is computed by

$$1 - \frac{\sum_{i\in\Omega}(x_i^{\text{estimated}} - x_i^{\text{gt}})^2}{\sum_{i\in\Omega}(x_i^{\text{gt}} - \bar{x}^{\text{gt}})^2},$$

where $\bar{x}^{\text{gt}}$ is the average of all x_i^{gt}s. These metrics are computed for the expression slices of six genes (three genes from each dataset) as shown in Table IX. Specifically, the LRMR employed in this section is the irregular version, i.e., the LRMR-RAG as previously mentioned.

The quantitative results in Table IX show that RTD-PAM consistently achieves the best performance across all genes in both datasets, demonstrating its superior ability to recover data accurately. Although CP-WOPT attains competitive accuracy, it is computationally inefficient, with a runtime $100.8\times$ slower than ours, which limits its real-world applications. While

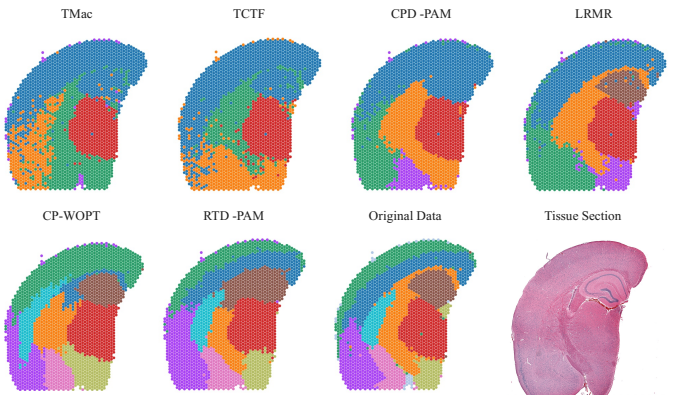


Fig. 10. Visual results of Cluster analysis by different methods on *FFPE Mouse Brain* for spatial gene expression data recovery (missing ratio = 50%).

LRMR shows relatively better performance than the remaining methods, it still falls short of CP-WOPT and ours. These results highlight the effectiveness of RTD-PAM in addressing the challenges of spatially irregular tensor data recovery. Fig. 9 visually compares the recovery results of various methods on the genes Cplx1 (from the *FFPE Mouse Brain* dataset) and Gatm (from the *FFPE Mouse Kidney* dataset). When the missing ratio is 50%, the observed data show significant missing patterns. Among the methods, RTD-PAM and CP-WOPT produce the most visually similar results to the original data, preserving fine-grained spatial patterns and expression levels. In contrast, other methods fail to capture the subtle spatial details, e.g., slight distortions introduced by LRMR on the gene Cplx1.

Then, we apply the Louvain algorithm [66] to perform clustering on the FFPE Mouse Brain dataset. Clustering is a crucial step in spatial transcriptomics data analysis as it helps to identify distinct spatial regions within a tissue based on gene expression profiles. This allows researchers to uncover functional and structural heterogeneity within tissues, such as identifying specific cell types, functional domains, or pathological regions. Fig. 10 showcases the clustering outcomes based on results recovered by different methods, alongside the original data and a reference tissue section for visual comparison. From Fig. 10, it is evident that the clustering result from the result recovered by our method (RTD-PAM) is the most accurate and coherent. Although CP-WOPT provides

a more reasonable result than the other methods, it still fails to correctly separate adjacent regions (e.g., the light-green, purple, and blue clusters) compared to the original data. Notably, the brown region in the RTD-PAM result aligns more closely with the actual tissue section, demonstrating a higher fidelity to the underlying biological structure. This is a significant improvement over the clustering results derived from the results of other methods, where inconsistencies and mismatches are more pronounced. For example, results by TMac and TCTF display irregular boundaries and fail to capture the fine-grained distinctions between different tissue regions, leading to less reliable clustering results.

A key observation is that the clustering result using the original data shows noticeable deviations, particularly in the brown region, compared to the tissue section. This suggests that even the original data has limitations in capturing the true structure of the spatial transcriptomics data, whereas RTD-PAM can recover this structure more effectively, potentially even enhancing the interpretability of the spatial relationships.

Across datasets, experimental results illustrate a clear hierarchy of performance rooted in algorithmic design. First, methods requiring zero-padding (TMac, TCTF, and especially CPD-PAM) consistently underperform. By treating meaningless padded entries as real data, their factorizations are fundamentally distorted. Second, while methods like PARAFAC2 can handle some irregularity, they are restricted to single-mode variations and are thus not applicable to the more general ragged structures studied here. The mask-aware CP-WOPT framework avoids distortion and achieves more competitive accuracy, but its holistic design reveals a profound algorithmic inefficiency. Its matrix-level updates are geometry-agnostic and ill-suited for the irregular optimization landscape, resulting in prohibitive runtimes. In contrast, our method's success stems from a design that is natively tailored to the ragged domain. The separable, row-wise PAM scheme avoids padding-induced distortion, naturally handles multidimensional irregularity, and decouples the optimization into a series of small, stable, and efficient subproblems. This geometry-aware approach consistently yields superior accuracy with substantially lower computational cost.

4) Ablation Study: In this subsection, we conducted an ablation study on the third segment of the MSI Cloth, the pattern of which is shown in Table V to analyze the influence of different ranks. As illustrated in Figure 11, the performance of the proposed method improves as the rank increases from 4 to 14, after which the relative error begins to slowly increase. Concurrently, the time cost steadily increases with the rank.

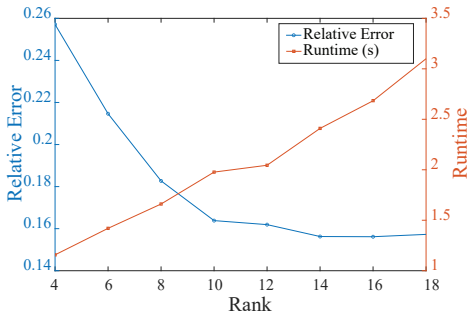


Fig. 11. The relative error values and runtime by our method with different rank values on the third segment of MSI *Cloth* in Table V (sampling rate = 10%).

V. Conclusions

This study introduces a separable tensor decomposition framework specifically designed for handling ragged tensors, a class of data structures that challenge traditional tensor decomposition methods. Unlike conventional approaches, which often rely on zero-padding or other preprocessing techniques to adapt irregular data into regular tensor formats, our method directly factorizes ragged tensors into a series of factors, preserving their intrinsic structure without introducing distortions. This innovation not only expands the applicability of tensor decomposition methods but also addresses a significant gap in analyzing irregular, real-world datasets. To operationalize this framework, we developed a weighted tensor decomposition model supported by a proximal alternating minimization (PAM) algorithm. Rigorous theoretical analysis confirmed the global convergence of the algorithm, ensuring robust and efficient computation. By restricting computations to valid entries, the proposed approach effectively manages the irregular nature of ragged tensors, avoiding unnecessary overhead. Extensive experiments on diverse datasets—including color images, multispectral and hyperspectral imaging data, and spatial transcriptomics data—validate the effectiveness of the proposed method. Across all scenarios, our approach consistently outperforms competing methods, demonstrating its capability to accurately capture the inherent structure of ragged tensors and underscoring its versatility in addressing a wide range of practical applications. We note that while our method is flexible for discrete, indexed data, it is not designed for off-the-grid data, such as point clouds. Furthermore, the model currently relies on a manually specified rank, and developing an adaptive rank-selection strategy is an important direction for future work.

References

[1] K. Li, L. Van Gool, and D. Dai, "Test-time training for hyperspectral image super-resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[2] A. Dziembowski, D. Mieloch, J. Y. Jeong, and G. Lee, "Immersive video postprocessing for efficient video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 8, pp. 4349–4361, 2023.

[3] T. Zhou, F. Porikli, D. J. Crandall, L. Van Gool, and W. Wang, "A survey on deep learning technique for video segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 6, pp. 7099–7122, 2022.

[4] A. Aravkin, R. Kumar, H. Mansour, B. Recht, and F. J. Herrmann, "Fast methods for denoising matrix completion formulations, with applications to robust seismic data interpolation," *SIAM Journal on Scientific Computing*, vol. 36, no. 5, pp. S237–S266, 2014.

[5] L. Omberg, G. H. Golub, and O. Alter, "A tensor higher-order singular value decomposition for integrative analysis of dna microarray data from different studies," *Proceedings of the National Academy of Sciences*, vol. 104, no. 47, pp. 18 371–18 376, 2007.

[6] F. L. Hitchcock, "Multiple invariants and generalized rank of a p-way matrix or tensor," *Journal of Mathematics and Physics*, vol. 7, no. 1-4, pp. 39–79, 1928.

[7] ——, "The expression of a tensor or a polyadic as a sum of products," *Journal of Mathematics and Physics*, vol. 6, no. 1-4, pp. 164–189, 1927.

[8] R. B. Cattell, ""Parallel proportional profiles" and other principles for determining the choice of factors by rotation," *Psychometrika*, vol. 9, no. 4, pp. 267–283, 1944.

[9] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 145–163, 2015.

[10] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, 2017.

[11] Y. Liu and M. K. Ng, "Deep neural network compression by Tucker decomposition with nonlinear response," *Knowledge-Based Systems*, vol. 241, p. 108171, 2022.

[12] F. Cong, Q.-H. Lin, L.-D. Kuang, X.-F. Gong, P. Astikainen, and T. Ristaniemi, "Tensor decomposition of EEG signals: A brief review," *Journal of Neuroscience Methods*, vol. 248, pp. 59–69, 2015.

[13] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[14] I. V. Oseledets, D. Savostianov, and E. E. Tyrtyshnikov, "Tucker dimensionality reduction of three-dimensional arrays in linear time," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 939–956, 2008.

[15] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[16] R. A. Harshman *et al.*, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, no. 1, p. 84, 1970.

[17] M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra and its Applications*, vol. 435, no. 3, pp. 641–658, 2011.

[18] 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*, vol. 34, no. 1, pp. 148–172, 2013.

[19] D. E. Deutsch, "Quantum computational networks," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 425, no. 1868, pp. 73–90, 1989.

[20] I. V. Oseledets, "Tensor-train decomposition," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.

[21] H. Chen and T. Barthel, "Machine learning with tree tensor networks, cp rank constraints, and tensor dropout," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 7825–7832, 2024.

[22] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[23] Y. Ren, J. Lou, L. Xiong, and J. C. Ho, "Robust irregular tensor factorization and completion for temporal health data analysis," in *ACM International Conference on Information & Knowledge Management*, 2020, pp. 1295–1304.

[24] P. L. Ståhl, F. Salmén, S. Vickovic, A. Lundmark, J. F. Navarro, J. Magnusson, S. Giacomello, M. Asp, J. O. Westholm, M. Huss *et al.*, "Visualization and analysis of gene expression in tissue sections by spatial transcriptomics," *Science*, vol. 353, no. 6294, pp. 78–82, 2016.

[25] A. Rao, D. Barkley, G. S. França, and I. Yanai, "Exploring tissue architecture using spatial transcriptomics," *Nature*, vol. 596, no. 7871, pp. 211–220, 2021.

[26] S. P. Ponnapalli, M. A. Saunders, C. F. Van Loan, and O. Alter, "A higher-order generalized singular value decomposition for comparison of global mrna expression from multiple organisms," *PloS one*, vol. 6, no. 12, p. e28072, 2011.

[27] P. Chalise and B. L. Fridley, "Integrative clustering of multi-level 'omic data based on non-negative matrix factorization algorithm," *PloS one*, vol. 12, no. 5, p. e0176278, 2017.

[28] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, 2001, pp. 556–562.

[29] R. A. Harshman, "PARAFAC2: Mathematical and technical notes," *UCLA working papers in phonetics*, vol. 22, pp. 30–47, 1972.

[30] J.-G. Jang, J. Lee, J. Park, and U. Kang, "Accurate PARAFAC2 decomposition for temporal irregular tensors with missing values," in *IEEE International Conference on Big Data*. IEEE, 2022, pp. 982–991.

[31] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41–56, 2011.

[32] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 46, no. 1, pp. 459–494, 2014.

[33] 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*, vol. 137, no. 1, pp. 91–129, 2013.

[34] A. Smilde, R. Bro, and P. Geladi, *Multi-way analysis: applications in the chemical sciences*. John Wiley & Sons, 2005.

[35] T. G. Kolda, "Multilinear operators for higher-order decompositions." Sandia National Laboratories, Tech. Rep., 2006.

[36] J. B. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra and Its Applications*, vol. 18, no. 2, pp. 95–138, 1977.

[37] M. E. Kilmer, L. Horesh, H. Avron, and E. Newman, "Tensor-tensor algebra for optimal representation and compression of multiway data," *Proceedings of the National Academy of Sciences*, vol. 118, no. 28, p. e2015851118, 2021.

[38] Y. Miao, L. Qi, and Y. Wei, "Generalized tensor function via the tensor singular value decomposition based on the t-product," *Linear Algebra and its Applications*, vol. 590, pp. 258–303, 2020.

[39] C. Wang, X.-L. Zhao, Y.-B. Zheng, B.-Z. Li, and M. K. Ng, "Functional tensor singular value decomposition," *SIAM Journal on Scientific Computing*, vol. 47, no. 4, pp. A2180–A2204, 2025.

[40] A. Wang, Y. Qiu, H. Huang, Z. Jin, G. Zhou, and Q. Zhao, "Towards a geometric understanding of tensor learning via the t-product," in *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. [Online]. Available: https://openreview.net/forum?id=0uo2GPd94f

[41] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, "Tensor ring decomposition," *arXiv preprint arXiv:1606.05535*, 2016.

[42] R. Penrose *et al.*, "Applications of negative dimensional tensors," *Combinatorial Mathematics and its Applications*, vol. 1, no. 221-244, p. 3, 1971.

[43] R. P. Feynman, "Quantum mechanical computers." *Foundations of Physics*, vol. 16, no. 6, pp. 507–532, 1986.

[44] J. Biamonte and V. Bergholm, "Tensor networks in a nutshell," *arXiv preprint arXiv:1708.00006*, 2017.

[45] B. Ermiş, E. Acar, and A. T. Cemgil, "Link prediction in heterogeneous data via generalized coupled tensor factorization," *Data Mining and Knowledge Discovery*, vol. 29, no. 1, pp. 203–236, 2015.

[46] J. Wang, Y. Hu, T.-X. Jiang, J. Tan, and Q. Li, "Essential tensor learning for multimodal information-driven stock movement prediction," *Knowledge-Based Systems*, vol. 262, p. 110262, 2023.

[47] T. G. Kolda, B. W. Bader, and J. P. Kenny, "Higher-order web link analysis using multilinear algebra," in *Fifth IEEE International Conference on Data Mining (ICDM'05)*. IEEE, 2005, pp. 8–pp.

[48] B. W. Bader and T. G. Kolda, "Efficient matlab computations with sparse and factored tensors," *SIAM Journal on Scientific Computing*, vol. 30, no. 1, pp. 205–231, 2008.

[49] J. Li, J. Choi, I. Perros, J. Sun, and R. Vuduc, "Model-driven sparse cp decomposition for higher-order tensors," in *2017 IEEE international parallel and distributed processing symposium (IPDPS)*. IEEE, 2017, pp. 1048–1057.

[50] S. Zhou, S. Erfani, and J. Bailey, "Online cp decomposition for sparse tensors," in *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 1458–1463.

[51] B. Hidasi and D. Tikk, "Fast als-based tensor factorization for context-aware recommendation from implicit feedback," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2012, pp. 67–82.

[52] J. Zhao, S. Zheng, H. Huo, M. Gong, T. Zhang, and L. Qu, "Fast weighted cp decomposition for context-aware recommendation with explicit and implicit feedback," *Expert Systems with Applications*, vol. 204, p. 117591, 2022.

[53] J. Jiang, F. Sanogo, and C. Navasca, "Low-cp-rank tensor completion via practical regularization," *Journal of Scientific Computing*, vol. 91, no. 1, p. 18, 2022.

[54] H. Che, B. Pan, M.-F. Leung, Y. Cao, and Z. Yan, "Tensor factorization with sparse and graph regularization for fake news detection on social networks," *IEEE Transactions on Computational Social Systems*, 2023.

[55] W. W. Sun, J. Lu, H. Liu, and G. Cheng, "Provable sparse tensor decomposition," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 79, no. 3, pp. 899–916, 2017.

[56] M. Cheng, L. Jing, and M. K. Ng, "A weighted tensor factorization method for low-rank tensor completion," in *2019 IEEE Fifth Interna-*

tional Conference on Multimedia Big Data (BigMM). IEEE, 2019, pp. 30–38.
[57] X. Zhang, J. Zheng, L. Zhao, Z. Zhou, and Z. Lin, "Tensor recovery with weighted tensor average rank," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 1, pp. 1142–1156, 2022.
[58] 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*, vol. 35, no. 2, pp. 438–457, 2010.
[59] A. Cichocki, R. Zdunek, and S.-i. Amari, "Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization," in *International Conference on Independent Component Analysis and Signal Separation*. Springer, 2007, pp. 169–176.
[60] 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*, vol. 6, no. 3, pp. 1758–1789, 2013.
[61] Y. Xu, R. Hao, W. Yin, and Z. Su, "Parallel matrix factorization for low-rank tensor completion," *Inverse Problems and Imaging*, vol. 9, no. 2, pp. 601–624, 2015.
[62] P. Zhou, C. Lu, Z. Lin, and C. Zhang, "Tensor factorization for low-rank tensor completion," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1152–1163, 2017.
[63] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
[64] E. Candes and B. Recht, "Exact matrix completion via convex optimization," *Communications of the ACM*, vol. 55, no. 6, pp. 111–119, 2012.
[65] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *European Conference on Computer Vision*, 2018, pp. 801–818.
[66] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

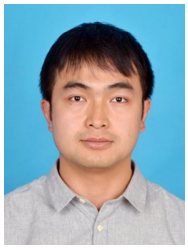
Michael K. Ng received the BSc and MPhil degrees from the University of Hong Kong, in 1990 and 1992, respectively, and the PhD degree from the Chinese University of Hong Kong, in 1995. He was a research fellow of Computer Sciences Laboratory with Australian National University, from 1995 to 1997, and an Assistant/Associate professor of the University of Hong Kong from 1997 to 2005. He was a professor/chair professor with the Department of Mathematics, Hong Kong Baptist University, from 2006 to 2019. He was a Chair Professor in Research Division of Mathematical and Statistical Science at The University of Hong Kong, from 2019 to 2023. He is currently a chair professor in mathematics and a chair professor in data science with Hong Kong Baptist University. His research interests include bioinformatics, image processing, scientific computing, and data mining. He is selected for the 2017 Class of Fellows of the Society for Industrial and Applied Mathematics. He was the recipient of the Feng Kang Prize for his significant contributions in scientific computing. He serves on the editorial board members of several international journals

Yexun Hu received the M.S. degree from the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics (SWUFE), Chengdu, China, in 2023. He is currently pursuing the Ph.D. degree with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu, China.

His research interests include tensor modeling and algorithm for high-order data recovery.

Xi-Le Zhao received the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2009 and 2012. He worked as a post-doc with Prof. M. Ng at Hong Kong Baptist University from 2013 to 2014. He worked as a visiting scholar with Prof. J. Bioucas Dias at University of Lisbon from 2016 to 2017. He is currently a Professor with the School of Mathematical Sciences, UESTC.

His research interests include image processing, machine learning, and scientific computing. More information can be found on his homepage https://zhaoxile.github.io/.



Tai-Xiang Jiang received the Ph.D. degrees in mathematics from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2019. He is currently a Professor with the School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics (SWUFE), China.

His research interests include sparse and low-rank modeling, tensor decomposition and multidimensional image processing. More information can be found on his homepage https://taixiangjiang.github.io/