# FieldFormer: Self-supervised Reconstruction of Physical Fields via Tensor Attention Prior

Panqi Chen, Siyuan Li, Lei Cheng, *Member, IEEE*, Xiao Fu, *Senior Member, IEEE*, Yik-Chung Wu, *Senior Member, IEEE*, and Sergios Theodoridis, *Life Fellow, IEEE*



***Abstract*—Reconstructing physical field tensors from *in situ* observations, such as radio maps and ocean sound speed fields, is crucial for enabling environment-aware decision making in various applications, e.g., wireless communications and underwater acoustics. Field data reconstruction is often challenging, due to the limited and noisy nature of the observations, necessitating the incorporation of prior information to aid the reconstruction process. Deep neural network-based data-driven structural constraints (e.g., "deeply learned priors") have showed promising performance. However, this family of techniques faces challenges such as model mismatches between training and testing phases. This work introduces FieldFormer, a self-supervised neural prior learned solely from the limited *in situ* observations without the need of offline training. Specifically, the proposed framework starts with modeling the fields of interest using the tensor Tucker model of a high multilinear rank, which ensures a universal approximation property for all fields. In the sequel, an attention mechanism is incorporated to learn the sparsity pattern that underlies the core tensor in order to reduce the solution space. In this way, a "complexity-adaptive" neural representation, grounded in the Tucker decomposition, is obtained that can flexibly represent various types of fields. A theoretical analysis is provided to support the recoverability of the proposed design. Moreover, extensive experiments, using various physical field tensors, demonstrate the superiority of the proposed approach compared to state-of-the-art baselines. The code is available at** https://github.com/OceanSTARLab/FieldFormer.

***Index Terms*—3D physical field reconstruction, tensor attention prior, tensor completion.**

## I. Introduction

THE accurate characterization of signal propagation in complex environments, such as underwater acoustics in the sea or electromagnetic waves in urban areas, is the stepping stone towards *environment-aware* wireless communications, target detection and recognition [1]–[6]. To accomplish this, several types of three-dimensional (3D) physical fields have been developed to provide valuable information across a given geographical region. Examples include the ocean sound speed field [7], which governs sound transmission in a spatially 3D ocean environment, and the radio map [2], which reveals information about the propagation of radio power across two spatial domains and one frequency domain.

Despite the vital role of the aforementioned 3D physical fields, crafting a finely detailed field that precisely captures the rapid variations of physical quantities (such as sound speeds or radio powers) across multiple domains (such as spaces or frequencies) presents a highly challenging task. Due to the high cost of in-situ measurements, sensors are often sparsely deployed across the geographical region, leaving a substantial portion of the physical fields unobserved [8], [9]. Using such limited and potentially noisy samples to reconstruct the complete 3D physical field is a typical ill-posed inverse problem, which has undergone extensive studies in recent years [1]–[4], [10]–[16].

Within the vast literature, the primary idea is to supplement the ill-posed reconstruction process with various informative priors of the associated physical fields. Early studies utilized *hand-crafted priors* rooted in basic assumptions about these fields, such as local smoothness [3], [12], [13], [17], [18] and global coherence [19], [20]. These assumptions could be readily translated into analytical forms such as total variations [18] and low-rank modeling [19], [20]. Despite their simplicity and interpretability, methods based on hand-crafted priors encounter challenges when the underlying structure of physical fields becomes complex. For instance, in the deep ocean, the presence of internal waves and eddies causes significant fluctuations in sound speed across large spatial scales [7]. Similarly, in urban areas, the proliferation of obstacles exacerbates the shadowing effect [21].

To excel in complex environments, there has been a notable focus on *data-driven priors.* These priors can be broadly classified into two categories: supervised priors and unsupervised priors (also called trained priors and untrained priors, respectively). Supervised priors rely on training data from historical measurements or simulators. Despite their promising performance in applications like ocean sound speed field recovery [22], [23] and radio map estimation [1], [15], [16], [24], [25], these methods face a number of challenges: 1) the performance deteriorates substantially when the fields targeted for reconstruction follow different data distributions compared to the training data; 2) the learned priors need to be retrained if the scenarios change; and 3) obtaining high-

The work of Lei Cheng was supported in part by the National Natural Science Foundation of China under Grant 62371418. The work of Xiao Fu was supported in part by the National Science Foundation (NSF) under Projects NSF ECCS-2024058 and NSF CCF-2210004. (Corresponding author: Lei Cheng)

Panqi Chen, Siyuan Li, Lei Cheng are with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. (e-mails: panq_chen, 3180100878, lei_cheng}@zju.edu.cn); Xiao Fu is with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331 USA (e-mail: xiao.fu@oregonstate.edu); Yik-Chung Wu is with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (email: ycwu@eee.hku.hk); Sergios Theodoridis is with the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Greece (email: stheodor@di.uoa.gr).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author.

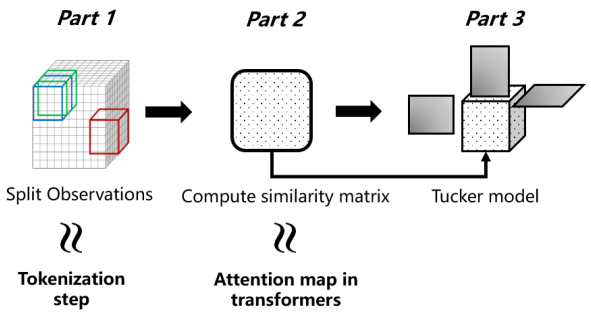



Fig. 1: The schematic figure illustrating the rationale of the proposed method.

quality training data is not always feasible, especially for physical fields situated in deep-sea or rural areas [8], [9]. To address the aforementioned challenges, there has been a surging interest in unsupervised data-driven priors [26]. These neural network-based priors only leverage inherent inductive-biases [27] that match the structure of the data, requiring no additional training data. This idea was used in various fields, e.g., image restoration [28] and sound speed field recovery [10]. Nonetheless, their effectiveness is often hampered by the limited number of observations and the predetermined model architecture/complexity. This prompts an intriguing question: *Can a self-supervised learning approach be devised to further distill knowledge from a limited amount of samples, which will lead to a "complexity-adaptive" data-driven prior to enhance physical field reconstruction?*

**Contributions.** To address this question, we propose *Field-Formers*. These comprise neural representations for field data that build upon a) a tensor Tucker model, b) an attention mechanism and c) self-supervised learning. Central to our idea is to leverage the notion of attention mechanism to automatically adjust the complexity of the adopted Tucker model, so that the representation strikes a reasonable balance between universality and parsimony. The main rationale of the proposed method lies in the following concept. We split the observation tensor in smaller cubes (see the first part of Fig. 1). One could view this process as the equivalent of the tokenization step in natural language processing (NLP). Then, a similarity matrix among the various tokens/cubes is computed, similar to the attention map in transformers (see the second part of Fig. 1). The similarity (attention) matrix implicitly implies a sparsity structure, since less similar parts/tokens lead to low-value attention weights. It is exactly this information that will be exploited by imposing it on the adopted Tucker model (see the third part of Fig. 1). Following this rationale, we devise (multi-head) tensor attention priors ((MH)TAP) to enable learning the sparse patterns of the core tensor of an over-complete Tucker model, which will be elaborated in Sec. III. The tensor attention mechanism is critical in capturing both short- and long-range dependencies among different areas of the field, and mapping such dependencies into the core tensor.

In addition to model design, the paper also studies various aspects that are of theoretical interest. We analyze the expected number of the non-zero elements in the core tensor to represent field data. Furthermore, we provide recoverability guarantees under the proposed model, which reveal the trade-off between sample and model complexities. Extensive experimental results, using ocean sound speed fields and radio maps, are presented that demonstrate the excellent performance of the proposed approaches.

**Notations:** Lower- and upper-case bold letters (e.g., $\mathbf{x}$ and $\mathbf{X}$) are used to denote vectors and matrices, respectively. Upper-case bold calligraphic letters and upper-case calligraphic letters (e.g., $\boldsymbol{\mathcal{X}}$ and $\mathcal{X}$) are used to denote tensors and sets. Operations $\otimes, \odot, *, \circ$ denote Kronecker product, Khatri-Rao product, Hadamard product and outer product respectively. $\|\cdot\|_{\mathrm{F}}, \|\cdot\|_0$, $\|\cdot\|_2$ and $\|\cdot\|_*$ represent Frobenius norm, $L_0$ norm, $L_2$ norm and nuclear norm, respectively. $|\mathcal{X}|$ represents the cardinality of set $\mathcal{X}$. $\|$ and $\mod$ are exact division and modulus operators.

## II. Problem Statement and Prior Art

In this section, we present the problem setup of 3D physical fields reconstruction and introduce the prior art.

### A. Problem Setup

The objective is to reconstruct the ground-truth 3D physical field, denoted as $\boldsymbol{\mathcal{X}}_\natural \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, from a limited number of noisy observations, denoted as $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. The typical observation or sensing model is defined as follows:

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{O}} * (\boldsymbol{\mathcal{X}}_\natural + \boldsymbol{\mathcal{N}}), \tag{1}$$

where $\boldsymbol{\mathcal{N}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ represents the noise tensor, and the binary tensor $\boldsymbol{\mathcal{O}}$ indicates the observed entries, where $\boldsymbol{\mathcal{O}}(i_1, i_2, i_3) = 1$ if the $(i_1, i_2, i_3)$-$th$ point is observed, and $\boldsymbol{\mathcal{O}}(i_1, i_2, i_3) = 0$ otherwise.

Based on the sensing model in (3), the reconstruction problem can be formulated in the following conceptual form:

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{X}}} \ \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * \boldsymbol{\mathcal{X}}\|_{\mathrm{F}}^2, \\ &\text{s.t.} \ \ \boldsymbol{\mathcal{X}} \in \mathcal{F}, \end{aligned} \tag{2}$$

where $\mathcal{F}$ denotes a set of structural constraints of the 3D field $\boldsymbol{\mathcal{X}}$. The recovery problem is ill-posed, as the number of observations is often much smaller than the signal dimension $I_1 I_2 I_3$. The key to tackling such a challenging inverse task lies in selecting a proper $\mathcal{F}$ that reflects prior information of $\boldsymbol{\mathcal{X}}$ and incorporating the related information to recover $\boldsymbol{\mathcal{X}}$. We also denote full observation $\tilde{\boldsymbol{\mathcal{Y}}}$ as:

$$\tilde{\boldsymbol{\mathcal{Y}}} = \boldsymbol{\mathcal{X}}_\natural + \boldsymbol{\mathcal{N}}. \tag{3}$$

In the following subsections, we briefly review the prior art on designing $\mathcal{F}$ and the remaining challenges.

### B. Prior Art and Challenges Ahead

**Handcrafted prior:** Many early methods in this domain use a relatively simple constraint set $\mathcal{F}$, e.g., low (matrix/tensor) rank [19], [20], and total variation [3], [17], [18]. The respective implementation is also relatively straightforward: one can often approximate these constraints using convex regularization terms, e.g., using the tensor nuclear norm [29]

$\frac{1}{3}\sum_{l=1}^{3}\|\mathbf{X}_{(l)}\|_*$ to approximate the low-rank constraint on $\boldsymbol{\mathcal{X}}$, where $\mathbf{X}_{(l)}$ is the mode-$l$ folding of $\boldsymbol{\mathcal{X}}, \forall l$. Although these constraints/regularization terms are simple to incorporate and easy to interpret, they often have limited capabilities in handling complex scenarios, requiring careful design for specific tasks.

**Supervised and unsupervised data-driven priors:** Another idea is to learn a generative model of $\boldsymbol{\mathcal{X}}$ from historical or simulated data $\{\boldsymbol{\mathcal{X}}_n^{\text{train}}\}_{n=1}^N$. This can be done via using popularized neural generative models such as autoencoders (AEs) [1] or generative adversarial networks (GANs) [25]; e.g., conceptually, AE-based generative model learning can be formulated as

$$\min_{\boldsymbol{\theta}} \ \frac{1}{N}\sum_{n=1}^{N}\|D_{\boldsymbol{\theta}}(Q_{\boldsymbol{\beta}}(\boldsymbol{\mathcal{X}}_n^{\text{train}})) - \boldsymbol{\mathcal{X}}_n^{\text{train}})\|_{\text{F}}^2. \tag{4}$$

Here, the representation model $\boldsymbol{\mathcal{X}} \approx D_{\boldsymbol{\theta}}(\boldsymbol{z})$, where $D_{\boldsymbol{\theta}}(\cdot)$ denotes a neural generative model and $\boldsymbol{z}$ is the associated latent representation of $\boldsymbol{\mathcal{X}}$; $Q_{\boldsymbol{\beta}}(\boldsymbol{\mathcal{X}})$ is the so-called "encoder" parameterized by $\boldsymbol{\beta}$ such that $Q_{\boldsymbol{\beta}}(\boldsymbol{\mathcal{X}}) \approx \boldsymbol{z}$. Once $D_{\boldsymbol{\theta}^\star}(\cdot)$ is learned with $\boldsymbol{\theta}^\star$ denoting the learned parameters, problem (2) can be simplified as

$$\min_{\boldsymbol{z}} \ \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * D_{\boldsymbol{\theta}^\star}(\boldsymbol{z})\|_{\text{F}}^2. \tag{5}$$

This type of data-driven prior partially mitigates the challenges related to handcrafted priors, particularly excelling in capturing the intricate details of physical fields [1], [24]. Nonetheless, their performance heavily depends on the quality and quantity of the training datasets $\{\boldsymbol{\mathcal{X}}_n^{\text{train}}\}_{n=1}^N$. Consequently, they face difficulties in adapting to real-time environmental shifts, potentially requiring re-training.

A workaround is to employ the so-called unsupervised priors, i.e., representing $\boldsymbol{\mathcal{X}}$ as $\boldsymbol{\mathcal{X}} = D(\boldsymbol{\theta})$, i.e., a neural network with untrained parameters $\boldsymbol{\theta}$, and solve the following:

$$\min_{\boldsymbol{\theta}} \ \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * D(\boldsymbol{\theta})\|_{\text{F}}^2. \tag{6}$$

In this way, neural architectures introduce useful inductive bias to model complex $\boldsymbol{\mathcal{X}}$, but training data is not needed. The input of the corresponding neural network is excited by random noise [28], [30]. Such untrained models attracted much attention from the vision community [28], [30] and were also used in sound speed field recovery [10]. However, designing the neural architecture is often nontrivial, and implementing such complex models often requires many heuristics that are hard to interpret, e.g., early stopping [28].

**Self-supervised approach:** To address the limitations of both supervised and unsupervised methods mentioned earlier, this paper follows a different path aiming to reconstruct physical fields using a self-supervised approach. The essence of self-supervised learning is to learn the underlying structure of the data by generating features at the output of an encoder, which is trained via targets that are constructed from the available unlabeled data, e.g., [31]–[33]. In our specific context, the reconstruction problem can be formulated as follows:

$$\begin{aligned} &\min_{\boldsymbol{\theta}} \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * \boldsymbol{\mathcal{X}}\|_{\text{F}}^2, \\ &\text{s.t.} \quad \boldsymbol{\mathcal{X}} = D_{\boldsymbol{\theta}}(\boldsymbol{\mathcal{Y}}). \end{aligned} \tag{7}$$

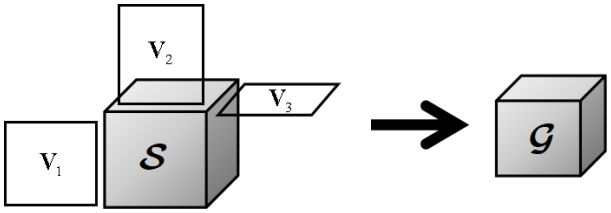



Fig. 2: Illustration of a 3-$rd$ order tensor Tucker model.

Here, $D_{\boldsymbol{\theta}}(\boldsymbol{\mathcal{Y}})$ defines the feasible set $\mathcal{F}$ (the feasible set could be considered as the encoder branch in a self-supervised task, starting from the input $\boldsymbol{\mathcal{Y}}$) and recovers the entire field from limited and noisy observations $\boldsymbol{\mathcal{Y}}$, which act as the respective targets for the training. The parameters $\boldsymbol{\theta}$ are optimized through self-supervised learning, contrasting with supervised approaches with pre-trained parameters. On the other hand, the proposed approach leverages input observations $\boldsymbol{\mathcal{Y}}$ to dynamically learn the model architecture/complexity, while earlier unsupervised methods are reliant on a fixed neural network architecture/complexity. Self-supervised prior learning was recently seen in vision [34] and hyperspectral imaging [35], showing appealing characteristics. However, [34] and [35] are not well suited for handling complex physical fields, as the former fails to capture multidimensional interactions within 3D fields, while the latter imposes restrictive low-rank constraints. In this work, our interest lies in designing self-supervised prior learning mechanisms tailored for 3D physical fields.

## III. Proposed Approach

In this section, we first introduce the tensor Tucker model and the attention mechanism as preliminaries, and then we elaborate how we propose our framework based on them.

### A. Preliminaries

**Tensor Tucker model:** The tensor Tucker model serves as the cornerstone of the proposed framework. We choose Tucker model because it appears to fit to the context of field estimation naturally. In particular, the core tensor resembles the attention map that admits interesting interpretation. Other tensor decomposition models may not offer such an immediate connection to attention. Specifically, we employ a third-order Tucker model in this work, as illustrated in Fig. 2. Its mathematical expression is as follows:

$$\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{S}} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \times_3 \mathbf{V}_3, \tag{8}$$

where $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ represents the core tensor and $\{\mathbf{V}_l \in \mathbb{R}^{I_l \times R_l}\}_{l=1}^3$ denote three sets of factor matrices. The output is denoted by $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. Note that $\times_l$ stands for mode-$l$ product. Specifically, the mode-$l$ product of a third order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{I_1 \times \cdots \times I_l \times \cdots \times I_L}$ and a matrix $\mathbf{B} \in \mathbb{R}^{J_l \times I_l}$, denoted as $\boldsymbol{\mathcal{A}} \times_l \mathbf{B}$, produces a $L$-$th$ order tensor $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{I_1 \times \cdots \times J_l \times \cdots \times I_L}$. And it can be expressed as

$$\boldsymbol{\mathcal{C}}_{i_1, i_2, \cdots, j_l, \cdots i_L} = \sum_{k=1}^{I_l} \boldsymbol{\mathcal{A}}_{i_1, i_2, \cdots, k, \cdots i_L} \mathbf{B}_{j_l, k}. \tag{9}$$

The Tucker model was introduced as *higher-order singular value decomposition* (SVD) [36] as it can retain orthogonality of $\boldsymbol{V}_l, \forall l$ (yet other decomposition such as the canonical polyadic decomposition (CPD) [37] cannot). The Tucker model is a *universal representer*—that is, when $R_1, R_2$ and $R_3$ are large enough (up to $R_l = I_l$), any tensor can be expressed by a Tucker decomposition model [37]. This is analogous the matrix case—i.e., any real-valued matrix admits an SVD.

**Attention mechanism:** The scaled dot-product attention mechanism [38] has been widely adopted in NLP and computer vision [39], [40] for its powerful feature extraction capabilities. Given $N$ input vectors of dimension $K$ (e.g., word embeddings), they can be organized into the input matrix $\mathbf{P} \in \mathbb{R}^{N\times K}$. The matrix $\mathbf{P}$ is then projected into different "embedding spaces":

$$\mathbf{Q} = \mathbf{P}\mathbf{W}_Q, \mathbf{K} = \mathbf{P}\mathbf{W}_K, \mathbf{V} = \mathbf{P}\mathbf{W}_V, \tag{10}$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{K\times M}$ are the feature embedding matrices and $M$ represents the latent embedding dimension. The matrices $\mathbf{Q}, \mathbf{K}$, and $\mathbf{V}$ are the query, key, and value matrices, respectively, all sharing the size of $N \times M$. Consequently, the formulation of attention is as follows:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{SoftMax}(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{M}})\mathbf{V}, \tag{11}$$

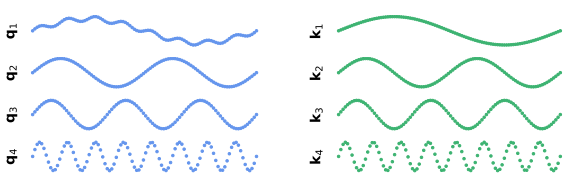
where SoftMax function does row normalization of $\mathbf{Q}\mathbf{K}^{\mathrm{T}}$ after scaled by $\sqrt{M}$. Note that $\mathbf{Q}\mathbf{K}^{\mathrm{T}}$ results in an attention map (i.e., row-by-row correlation matrix) of size $N\times N$, with the $n$-$th$ row representing the similarities between $n$-$th$ input vector and the rest. In a nutshell, attention is basically an expansion of the input matrix $\mathbf{P}$ in terms of value vectors (i.e., rows in $\mathbf{V}$) with weighting coefficients expressing mutual similarities. Less similar vectors are weighted with small weights. Thus this mechanism can also be used to impose sparsity in the model.

A toy example is provided in Fig. 3 to illustrate the process of computing an attention map, *which typically reflects some sparse patterns due to the predominantly incoherent nature of the latent features.*

### B. Proposed Sparse Tensor Attention Module

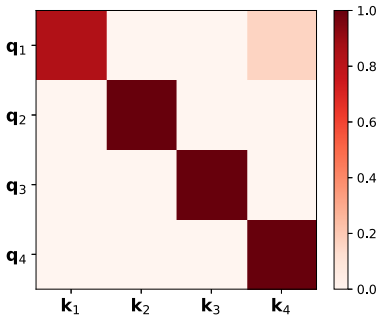
The expressiveness of the Tucker model depends on the sizes or dimensions of the core tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{R_1\times R_2\times R_3}$ in comparison to the output tensor $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{I_1\times I_2\times I_3}$. Specifically, when the core tensor is large, i.e., $R_l \geq I_l, \forall l$, it becomes expressive enough to represent an arbitrary tensor of size $(I_1, I_2, I_3)$. However, in the absence of suitable regularization, directly fitting such a Tucker model to limited observations of physical fields can lead to overfitting. To address this issue, previous methods have proposed sparsifying the core tensor $\boldsymbol{\mathcal{S}}$ by incorporating various sparsity-aware regularizers [23], [41], [42]. Nevertheless, these handcrafted regularizers do not adapt to *in-situ* data, and thus often exhibit model mismatches in field estimation problems.

Towards data-adaptive sparse coding for the core tensor $\boldsymbol{\mathcal{S}}$, we propose leveraging the attention mechanism. Our idea is inspired by the underlying similarities between the expansions




(a) Toy example of query matrix $\mathbf{Q}$. The first row of $\mathbf{Q}$ is a sine wave in combination with $\mathbf{k}_1$ and $\mathbf{k}_4$ (as shown in (b)). The rest rows of $\mathbf{Q}$ are similar to that of $\mathbf{K}$.

(b) Toy example of key matrix $\mathbf{K}$. Each row of $\mathbf{K}$ (i.e., $\mathbf{k}_l, l = 1, 2, 3, 4$) is a sine wave with different frequencies. Therefore, they are orthogonal to each other.




(c) Toy example of $\mathbf{Q}\mathbf{K}^{\mathrm{T}}$.

Fig. 3: Toy example of query, key matrices and their similarities represented by $\mathbf{Q}\mathbf{K}^{\mathrm{T}}$. Obviously, only $\mathbf{q}_1$ shows some similarities with $\mathbf{k}_1$ and $\mathbf{k}_4$ while others show no similarities but themselves.

in Eq. (8) and Eq. (11). By comparing the two expressions, one can interpret the Tucker model as a tensorized version of the attention matrix. Specifically, the core tensor $\boldsymbol{\mathcal{S}}$ acts as the weight coefficient tensor, and the three factor matrices $\{\mathbf{V}_l\}_{l=1}^3$ serve as the value matrices. This observation motivates us to propose a way for automatically learning the sparsity pattern of the core tensor $\boldsymbol{\mathcal{S}}$.

*1) Local Region Representation:* To achieve this goal, as shown in the first part of Fig. 4, we begin by extracting cubes from the observed 3D physical field $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1\times I_2\times I_3}$ using 3D windows of sizes $(K_1, K_2, K_3)$ and strides of sizes $(S_1, S_2, S_3)$ in the three modes[1]. This process generates a total of $N$ cubes, where $N = \prod_{l=1}^3 J_l$ and $J_l = (\frac{I_l - K_l}{S_l} + 1)$. Specifically, we define $\boldsymbol{\mathcal{C}}_n \in \mathbb{R}^{K_1\times K_2\times K_3}$ to represent the $n$-$th$ cube extracted from $\boldsymbol{\mathcal{Y}}$:

$$\boldsymbol{\mathcal{C}}_n = \boldsymbol{\mathcal{Y}}(\mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3), \tag{12}$$

where the index set $\{\mathbf{i}_l = 1 + (m_l - 1)S_l : (m_l - 1)S_l + K_l, \forall l\}$ and $m_1 = (n-1) \| J_2J_3 + 1, m_2 = [(n-1) \mod J_2J_3] \| J_3 + 1, m_3 = (n-1) \mod J_3 + 1$. A clearer illustration of patch extraction can be found in the cube extraction part of Fig. 4, where the first cube $\boldsymbol{\mathcal{C}}_1$ is represented in blue, the second cube $\boldsymbol{\mathcal{C}}_2$ is in green, and so forth. These $N$ cubes are then vectorized, resulting in a data matrix $\mathbf{P} = [\mathbf{c}_1, \ldots, \mathbf{c}_N]^{\mathrm{T}} \in \mathbb{R}^{N\times K_1K_2K_3}$ where $\mathbf{c}_n = \text{vec}(\boldsymbol{\mathcal{C}}_n) \in \mathbb{R}^{K_1K_2K_3}$.

[1] Guidelines for the selection of window sizes and stride sizes are presented in Appendix L.

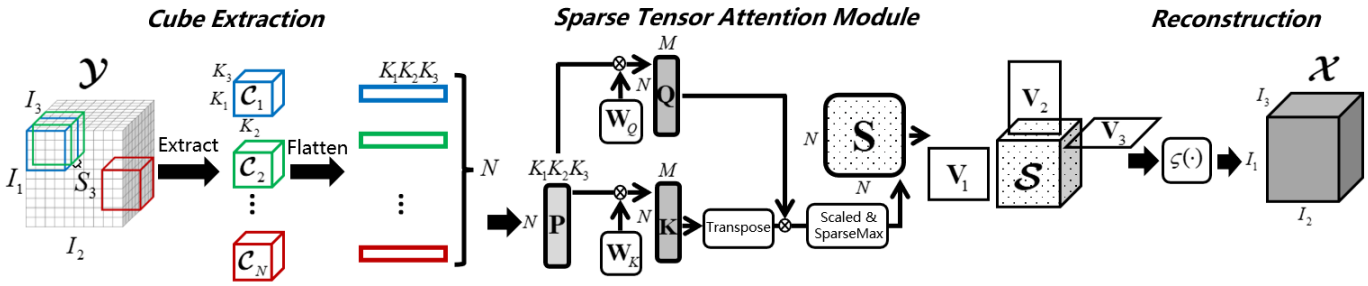



Fig. 4: The detailed architecture of the proposed tensor attention prior (TAP) model for reconstructing 3D physical fields with limited observations.

*2) Sparse Tensor Attention Construction:* Next, we introduce the proposed sparse tensor attention (STA) module. As shown in the second part of Fig. 4, we project the data matrix $\mathbf{P}$ into two latent spaces using the embedding matrices $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{K_1K_2K_3\times M}$. This projection yields the query matrix $\mathbf{Q} = \mathbf{P}\mathbf{W}_Q \in \mathbb{R}^{N\times M}$ and the key matrix $\mathbf{K} = \mathbf{P}\mathbf{W}_K \in \mathbb{R}^{N\times M}$. The dot products between the query and key matrices are then computed, resulting in a $N \times N$ correlation matrix $\mathbf{Q}\mathbf{K}^{\mathrm{T}}$.

Unlike the traditional scaled dot-product attention mechanism in (11), which assumes the full observation and scales the correlation matrix with a constant $\sqrt{M}$, we apply element-wise division with a scaling matrix $\mathbf{M}$. The operation is represented as $\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M}$, where $\oslash$ is the element-wise division operator and $\mathbf{M} \in \mathbb{R}^{N\times N}$ is the matrix containing the norms of embedded features, with each element being:

$$\mathbf{M}(n_1, n_2) = \|\mathbf{q}_{n_1}\|_2 \|\mathbf{k}_{n_2}\|_2. \tag{13}$$

Here, $\mathbf{q}_{n_1}$ denotes the $n_1$-$th$ row of $\mathbf{Q}$ and $\mathbf{k}_{n_2}$ denotes the $n_2$-$th$ row of $\mathbf{K}$. The scaling matrix $\mathbf{M}$ is used to normalize the dot products, addressing the issue of energy imbalance caused by different missing patterns across extracted cubes. Then, the results are passed through the SparseMax function [43]. This process generates the *sparse attention map*, which is similar to a correlation matrix and is denoted as

$$\mathbf{S} = \mathrm{SparseMax}(\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M}) \in \mathbb{R}^{N\times N}. \tag{14}$$

The SparseMax function, like the SoftMax function, aims to produce a normalized score vector, but its output is much sparser (see illustrations in Fig. 11 and brief implementation details in Appendix C). Concretely, it selects a certain number of leading entries of the input while setting the rest to zero, and finally normalizes these support entries to sum up to 1. Further details can be found in [43].

One might concern whether the missing values in $\boldsymbol{\mathcal{Y}}$ could significantly affect the assessment of tensor attention. However, our observation is that they do not. The reason is that when calculating the correlation between two high-dimensional signals, a few missing values in each signal will not substantially degrade the correlation estimation.

*3) Decoder Design:* Finally, the sparse attention map is used to reconstruct the entire tensor. This corresponds to what we call "decoder design" in neural representation learning.

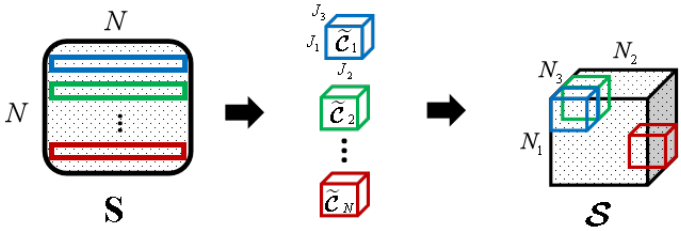


Fig. 5: Illustration of attention map tensorization.

The goal now is to tensorize the sparse attention map $\mathbf{S}$ into a sparse tensor $\boldsymbol{\mathcal{S}}$ of appropriate dimensions, which will act as the core tensor that interacts with the three factor matrices to produce the output. Note that $\boldsymbol{\mathcal{S}}$ is the tensorized version of $\mathbf{S}$, containing the same elements as $\mathbf{S}$. To this end, the following procedure is adopted.

The sparse attention map consists of $N$ row vectors (i.e, $\mathbf{S} = [\tilde{\mathbf{c}}_1, \cdots, \tilde{\mathbf{c}}_N]^{\mathrm{T}}$), with the $n$-$th$ row vector $\tilde{\mathbf{c}}_n^{\mathrm{T}} \in \mathbb{R}^{1\times N}$ encoding the correlations between $n$-$th$ cube with other $N$ cubes. We first tensorize each row of $\mathbf{S} \in \mathbb{R}^{N\times N}$ into a sub-tensor (with size $J_1 \times J_2 \times J_3$). Specifically, the $n$-$th$ sub-tensor $\tilde{\boldsymbol{\mathcal{C}}}_n$ can be represented as

$$\tilde{\boldsymbol{\mathcal{C}}}_n(j_1, j_2, j_3) = \tilde{\mathbf{c}}_n((j_1-1)J_2J_3 + (j_2-1)J_3 + j_3). \tag{15}$$

This process is done orderly to ensure that the relative positions of the entries within each sub-tensor align with the positions of the corresponding cubes in $\boldsymbol{\mathcal{Y}}$. Next, we stack these sub-tensors in the same order to construct the sparse core tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N_1\times N_2\times N_3}$, where $N_l = (J_l)^2, \forall l$. That is,

$$\boldsymbol{\mathcal{S}}(\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3) = \tilde{\boldsymbol{\mathcal{C}}}_n, \tag{16}$$

where the index set $\{\mathbf{n}_l = 1 + (m_l - 1)J_l : m_l J_l, \forall l\}$ and $m_1 = (n-1)\|J_2J_3 + 1, m_2 = [(n-1) \mod J_2J_3]\|J_3 + 1, m_3 = (n-1) \mod J_3 + 1$. The relative positions within each sub-tensor are also consistent with the arrangement of the extracted cubes in $\boldsymbol{\mathcal{Y}}$. As a result, the physical meaning of of an entry, $\boldsymbol{\mathcal{S}}(n_1, n_2, n_3)$, is that it quantifies the correlation between the embedding feature of the $x$-$th$ and the $y$-$th$ cubes, where $x = (n_1\|J_1)J_2J_3 + (n_2\|J_2)J_3 + (n_3\|J_3) + 1$ and $y = ((n_1-1) \mod J_1)J_2J_3 + ((n_2-1) \mod J_2)J_3 + ((n_3-1) \mod J_3 + 1)$. The tensorization process is illustrated in Fig. 5, which can be simply understood as the reorganization of the elements of $\mathbf{S}$ into the tensor $\boldsymbol{\mathcal{S}}$, thereby rendering $\boldsymbol{\mathcal{S}}$

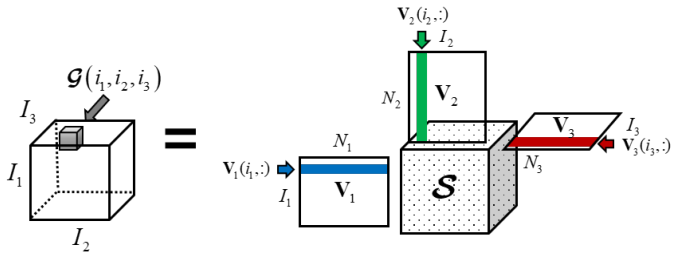



Fig. 6: Element-wise view of the sparse tensor attention (STA) module.

inherently sparse. Its primary goal is to preserve the spatial relationships of each entry, mirroring the relative positions of the cubes that are extracted from $\mathcal{Y}$, see Fig. 4. This ensures the construction of an informative core tensor that leverages the priors, which come from the observations, so that to effectively capture multidimensional interactions. The formulation of the tensorization process can be conveniently implemented with PyTorch, as it presented in Appendix A.

The size of $\mathcal{S}$ determines the respective size of each one of the three learnable factor matrices (a.k.a value matrices) $\{\mathbf{V}_1 \in \mathbb{R}^{I_1 \times N_1}, \mathbf{V}_2 \in \mathbb{R}^{I_2 \times N_2}, \mathbf{V}_3 \in \mathbb{R}^{I_3 \times N_3}\}$. The proposed sparse tensor attention module can be expressed as follows:

$$\begin{aligned} &\mathcal{G} = \text{STA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3) = \\ &\mathcal{S} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \times_3 \mathbf{V}_3 \in \mathbb{R}^{I_1 \times I_2 \times I_3}, \\ &\text{s.t.} \quad \mathcal{S} = \text{Tensorize}(\text{SparseMax}(\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M})). \end{aligned} \tag{17}$$

Essentially, the proposed sparse tensor attention (STA) module can be interpreted as the summation of all multiplicative interactions of the value matrices weighted by the sparse core tensor $\mathcal{S}$. More specifically, an element-wise interpretation of Eq. (17) can be formulated as:

$$\mathcal{G}(i_1, i_2, i_3) = \mathcal{S} \times_1 \mathbf{V}_1(i_1, :) \times_2 \mathbf{V}_2(i_2, :) \times_3 \mathbf{V}_3(i_3, :). \tag{18}$$

The illustration of this formulation can be seen in Fig. 6. Note that $\mathbf{V}_l(i_l, :)$ can be interpreted as the $i_l$-$th$ feature embedding of mode-$l$ (see the blue, green and red feature vectors in Fig. 6). The sparse core tensor $\mathcal{S}$, which contains spatial correlation weights, appropriately combines all the feature embeddings (i.e., $\mathbf{V}_l(i_l, :), \forall l$) and produces the output $\mathcal{G}(i_1, i_2, i_3)$.

The module operates intuitively: when the extracted cubes exhibit significant similarities, indicating a simpler representation model for the considered 3D physical field, the resulting sparse attention map highlights these similarities, and it consequently leads to a reduced number of the non-zero elements in the core tensor for reconstructing the 3D physical field, and vice versa. *Therefore, the proposed module can adaptively adjust the model's complexity based on information that is extracted from the the observations.*

### C. Proposed Tensor Attention Prior

In the previous two sections, we have discussed how to utilize a limited number of observations to construct a sparse attention map and then generate the core tensor for the Tucker model. The nonlinear activation function $\varsigma(\cdot)$ is then introduced to produce the final output, aiming to further enhance the expressive power of the proposed model while maintaining stable gradients during training [44]. We recommend using Tanh or Sigmoid to regulate the output range. The overall model, illustrated in Fig. 4, is referred to as the *tensor attention prior* (TAP) representation model.

Given the TAP model, the problem of reconstructing the physical field can be formulated as follows:

$$\begin{aligned} \min_{\mathbf{W}_Q, \mathbf{W}_K, \{\mathbf{V}_l\}_{l=1}^3} \; & \|\mathcal{Y} - \mathcal{O} * \varsigma(\mathcal{S} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \times_3 \mathbf{V}_3)\|_{\mathrm{F}}^2, \\ \text{s.t.} \quad & \mathcal{S} = \text{Tensorize}(\text{SparseMax}(\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M})), \\ & \mathbf{Q} = \mathbf{P}\mathbf{W}_Q, \mathbf{K} = \mathbf{P}\mathbf{W}_K. \end{aligned} \tag{19}$$

Once the parameters $\mathbf{W}^*{}_Q, \mathbf{W}^*{}_K, \{\mathbf{V}^*{}_l\}_{l=1}^3$ are learned, the reconstructed field is given by:

$$\begin{aligned} \mathcal{X} = & \varsigma(\text{Tensorize}(\text{SparseMax}([\mathbf{P}\mathbf{W}^*{}_Q]\,[\mathbf{P}\mathbf{W}^*{}_K]^{\mathrm{T}} \oslash \mathbf{M}) \\ & \times_1 \mathbf{V}^*{}_1 \times_2 \mathbf{V}^*{}_2 \times_3 \mathbf{V}^*{}_3)). \end{aligned} \tag{20}$$

The resulting 3D FieldFormer algorithm for 3D physical field reconstruction is presented in **Algorithm 1**.

**Algorithm 1** 3D FieldFormer based on TAP.

**Input:** Observations $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, binary tensor $\mathcal{O} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, window size in three modes $(K_1, K_2, K_3)$, stride size in three modes $(S_1, S_2, S_3)$;
**Initialization:** Initialize the query and key matrices $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{M \times M}$ as well as the value matrices $\mathbf{V}_1 \in \mathbb{R}^{I_1 \times N_1}, \mathbf{V}_2 \in \mathbb{R}^{I_2 \times N_2}, \mathbf{V}_3 \in \mathbb{R}^{I_3 \times N_3}$.

1: Extract cubes from observations $\mathcal{Y}$ to get $\mathbf{P}$.
2: **while** not converge **do**
3: Compute the query and key through $\mathbf{Q} = \mathbf{P}\mathbf{W}_Q, \mathbf{K} = \mathbf{P}\mathbf{W}_K$.
4: Compute the output of sparse attention module through Eq. (17) and obtain reconstructed 3D physical field $\mathcal{X}$ via Eq. (20).
5: Compute the loss $\|\mathcal{Y} - \mathcal{O} * \mathcal{X}\|_{\mathrm{F}}^2$
6: Update $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$ according to the loss using the Adam optimizer.
7: **end while**

**Output:** The reconstructed 3D physical field $\mathcal{X}$.

*Remark 1 (Initializations and Convergence):* All the parameters are initialized using samples drawn from uniform distributions following the Kaiming initialization scheme [45]. We employ the Adam optimizer to minimize the objective function. Given that the objective is differentiable with respect to all parameters, the convergence of Adam to a stationary point over long-run iterations with a constant stepsize has been established in [46].

### D. Multi-Head Tensor Attention Prior

To further enhance the extracted information, the Multi-Head Sparse Tensor Attention (MHSTA) generalization is also

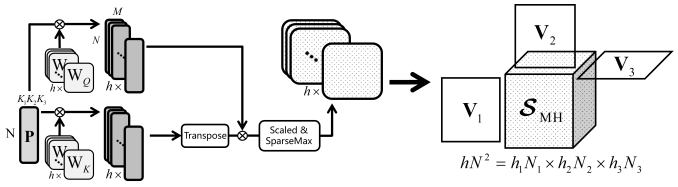



Fig. 7: Illustration of Multi-head Sparse Tensor Attention Module.

proposed, where one can utilize multiple embedding subspaces for quantifying similarity and computing the corresponding attention maps. This is illustrated in Fig. 7. First, $h$ query and key matrices are employed to perform, in parallel, inner products followed by the corresponding SparseMax operations. Thus, $h$ sparse attention maps are obtained. These maps are then reshaped[2] to obtain $\boldsymbol{\mathcal{S}}_{\text{MH}} \in \mathbb{R}^{h_1N_1 \times h_2N_2 \times h_3N_3}$. The Tucker model is applied next, where $\boldsymbol{\mathcal{S}}_{\text{MH}}$ operates with three sets of value matrices $\{\mathbf{V}_1 \in \mathbb{R}^{I_1 \times h_1N_1}, \mathbf{V}_2 \in \mathbb{R}^{I_2 \times h_2N_2}, \mathbf{V}_3 \in \mathbb{R}^{I_3 \times h_3N_3}\}$. Note that equation $hN^2 = h_1N_1 \times h_2N_2 \times h_3N_3$ should be guaranteed. This produces the output:

$$\begin{aligned}&\text{MHSTA}(\{\mathbf{Q}_i\}_{i=1}^h, \{\mathbf{K}_i\}_{i=1}^h, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3) = \\ &\quad \boldsymbol{\mathcal{S}}_{\text{MH}} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \times_3 \mathbf{V}_3 \in \mathbb{R}^{I_1 \times I_2 \times I_3} \\ &\quad \text{s.t.} \quad \boldsymbol{\mathcal{S}}_{\text{MH}} = \text{Tensorize}(\text{SparseMax}( \\ &\quad \text{Concat}(\mathbf{Q}_1\mathbf{K}_1^{\text{T}} \oslash \mathbf{M}_1, \cdots, \mathbf{Q}_h\mathbf{K}_h^{\text{T}} \oslash \mathbf{M}_h))),\end{aligned} \tag{21}$$

where $\mathbf{Q}_i = \mathbf{P}\mathbf{W}_Q^i$, $\mathbf{K}_i = \mathbf{P}\mathbf{W}_K^i$ and $\mathbf{M}_i$ is computed from the row-wise norms of $\mathbf{Q}_i$ and $\mathbf{K}_i$.

Then, we propose the *multi-head tensor attention prior* (MHTAP) architecture by simply replacing the STA of TAP with multi-head STA (MHSTA). The formulation of MHTAP for reconstructing 3D physical fields is as follows:

$$\begin{aligned}&\min_{\{\mathbf{W}_Q^i, \mathbf{W}_K^i\}_{i=1}^h, \{\mathbf{V}_l\}_{l=1}^3} \\ &\quad \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * \varsigma(\boldsymbol{\mathcal{S}}_{\text{MH}} \times_1 \mathbf{V}_1 \times_2 \mathbf{V}_2 \times_3 \mathbf{V}_3)\|_{\text{F}}^2.\end{aligned} \tag{22}$$

The associated FieldFormer algorithm is similar to **Algorithm 1** and summarized in Appendix B.

*Remark 2 (Extension to Higher-order Field Tensors):* The proposed FieldFormer naturally extends to higher-order tensors by extracting multidimensional cubes, computing their similarities to generate a sparse attention map, and tensorizing it into a multidimensional core as described in Sec. III-B. Finally, reconstruction is carried out using a multidimensional Tucker model.

## IV. Recoverability Analysis

In this section, we investigate the recoverability of the ground-truth 3D field tensor $\boldsymbol{\mathcal{X}}_\natural$ based on the problem formulated in (19). Our analysis adapts the framework established in [1], [47], [48] to the proposed nonlinear Tucker model that includes a sparse core tensor with its sparsity pattern determined by attention schemes, non-orthogonal factor matrices, and a nonlinear activation function. Notably, we derive new theoretical results on the covering number and generalization errors associated with the proposed model.

To proceed, we denote $\Omega$ as the set of observed indices, i.e., $\Omega = \{(i_1, i_2, i_3) | \boldsymbol{\mathcal{O}}(i_1, i_2, i_3) = 1\}$ and introduce the following definitions.

*Definition 1 (Solution Set):* Let $\mathcal{X}_{\text{TAP}} \subset \mathbb{R}^{I_1 \times I_2 \times I_3}$ be the *solution set* that contains all solutions $\tilde{\boldsymbol{\mathcal{X}}}$ of (19) satisfying $\tilde{\boldsymbol{\mathcal{X}}} = \varsigma(\tilde{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3)$, where $\|\tilde{\boldsymbol{\mathcal{S}}}\|_{\text{F}} \leq \alpha$ and $\|\tilde{\mathbf{V}}_l\|_{\text{F}} \leq \beta$ for $i = 1, 2, 3$.

*Definition 2:* Define $\text{Gap}(\tilde{\boldsymbol{\mathcal{X}}}, \Omega) = \sqrt{\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})}$, where

$$\begin{aligned}\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}}) &= \frac{1}{|\Omega|} \sum_{(i_1,i_2,i_3) \in \Omega} \|\tilde{\boldsymbol{\mathcal{Y}}}(i_1, i_2, i_3) - \tilde{\boldsymbol{\mathcal{X}}}(i_1, i_2, i_3)\|_2^2, \\ \text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}}) &= \frac{1}{I_1 I_2 I_3} \sum_{i_1,i_2,i_3} \|\tilde{\boldsymbol{\mathcal{Y}}}(i_1, i_2, i_3) - \tilde{\boldsymbol{\mathcal{X}}}(i_1, i_2, i_3)\|_2^2.\end{aligned} \tag{23}$$

Note that $\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})$ represents the loss measured on the observed part of the data, while $\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})$ represents the loss measured over the entire data. Thus, $\text{Gap}(\tilde{\boldsymbol{\mathcal{X}}}, \Omega)$ can be interpreted as the *generalization error*.

We can then show the following lemmas based on the definitions provided above.

*Lemma 1:* The expected number of the non-zero elements in the sparse core tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N_1 \times N_2 \times N_3}$ is $\mathbb{E}(\|\boldsymbol{\mathcal{S}}\|_0) = N\mathbb{E}(n(\mathbf{z}))$, where $n(\mathbf{z})$ is a random variable representing the number of the non-zero elements in each row of the sparse attention map $\mathbf{S}$. The expectation of the non-zero elements $\mathbb{E}(n(\mathbf{z})) = \sum_{i=1}^N iP(n(\mathbf{z}) = i)$, where $P(n(\mathbf{z}))$ follows the distribution specified in (24) under the assumption that all entries of $\mathbf{Q}\mathbf{K}^{\text{T}} \oslash \mathbf{M}$ are independent random variables drawn from a normal distribution with a mean of 0 and a variance of 1 [38].

$$\begin{aligned}P(n(\mathbf{z}) = 1) &= 2 - 2\Phi(\frac{1}{\sqrt{2}}), \\ P(n(\mathbf{z}) = n) &= (2 - 2\Phi(\frac{1}{\sqrt{2n}})) \times \prod_{i=2}^{n} (2\Phi(\frac{1}{2i-2}) - 1), \\ &\qquad\qquad n = 2, \cdots, N-1, \\ P(n(\mathbf{z}) = N) &= \prod_{i=2}^{N} (2\Phi(\frac{1}{\sqrt{2i-2}}) - 1),\end{aligned} \tag{24}$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of Gaussian distribution with mean 0 and variance 1:

$$\Phi(X) = \int_{-\infty}^{X} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx. \tag{25}$$

*Proof*: See Appendix D. ■

Lemma 1 characterizes the expected number of the non-zero elements within the sparse tensor attention map, a metric

[2]We first tensorize $h$ attention maps individually as before and obtain $\{\boldsymbol{\mathcal{S}}_i \in \mathbb{R}^{N_1 \times N_2 \times N_3}\}_{i=1}^h$. Then, to make the sparse core tensor has the appropriate dimensionality (each dimension is relatively balanced), we choose $h_1, h_2, h_3$ such that $h = h_1h_2h_3$. $\{\boldsymbol{\mathcal{S}}_i \in \mathbb{R}^{N_1 \times N_2 \times N_3}\}_{i=1}^h$ are then stacked along the corresponding modes to form $\boldsymbol{\mathcal{S}}_{\text{MH}} \in \mathbb{R}^{h_1N_1 \times h_2N_2 \times h_3N_3}$ with multidimensional correlations preserved.

useful for assessing the complexity of the proposed model. Building on the similar assumption made in the attention analysis using the SoftMax operator [38], Lemma 1 establishes the sparsity level result for the SparseMax operator. This lemma is useful for future endeavors that employ the SparseMax operator to derive attention scores.

*Lemma 2:* The covering number [49] of the $\varepsilon$-net of the solution set $\mathcal{X}_{\text{TAP}}$ is given by

$$\mathsf{N}(\mathcal{X}_{\text{TAP}}, \varepsilon) \leq \left[\frac{3T(\beta^3 + 3\alpha\beta^2)}{\varepsilon}\right]^{\|\boldsymbol{\mathcal{S}}\|_0 + \sum_{l=1}^{3} N_l I_l} \alpha^{\|\boldsymbol{\mathcal{S}}\|_0} \beta^{\sum_{l=1}^{3} N_l I_l}, \tag{26}$$

where $T$ is the Lipschitz constant of the activation function and $\mathbb{E}(\|\boldsymbol{\mathcal{S}}\|_0)$ is given by *Lemma 1*.

*Proof*: See Appendix E. ∎

*Lemma 3:* Let $\text{Gap}^*(\Omega) = \sup_{\tilde{\boldsymbol{\mathcal{X}}} \in \mathcal{X}_{\text{TAP}}} \left|\text{Gap}(\tilde{\boldsymbol{\mathcal{X}}}, \Omega)\right|$ be the supremum of $\text{Gap}(\tilde{\boldsymbol{\mathcal{X}}}, \Omega)$. Based on the sensing model in (3), the following inequality holds with a probability of at least $1 - \delta$:

$$\text{Gap}^*(\Omega) \leq \frac{2\varepsilon}{\sqrt{|\Omega|}} + \left(\frac{\xi^2 \omega}{2} \log(\frac{2\mathsf{N}(\mathcal{X}_{\text{TAP}}, \varepsilon)}{\delta})\right)^{\frac{1}{4}}, \tag{27}$$

where $\varepsilon > 0$ is a positive scalar and $\omega = (\frac{1}{|\Omega|} + \frac{1}{|\Omega| I_1 I_2 I_3} - \frac{1}{I_1 I_2 I_3})$. $\mathsf{N}(\mathcal{X}_{\text{TAP}}, \varepsilon)$ is the covering number of $\mathcal{X}_{\text{TAP}}$ (see *Lemma 2*), and $\xi = \left(\nu + v + \alpha\beta^3\right)^2$ with $\nu = \max_{i_1,i_2,i_3} |\boldsymbol{\mathcal{X}}_\natural(i_1, i_2, i_3)|$ and $v = \max_{i_1,i_2,i_3} |\boldsymbol{\mathcal{N}}(i_1, i_2, i_3)|$.

*Proof*: See Appendix F. ∎

Lemma 2 and Lemma 3 characterize the generalization error for the proposed TAP model. They extend the results developed in [1], [47], [48] to account for the sparse tensor core, non-orthogonal factors, and the non-linear activation function. As a result, in addition to paving the way for establishing the recoverability of the proposed model (refer to Theorem 1 below), these two lemmas themselves are essential for quantifying the generalization error in any future nonlinear tensor Tucker model.

Finally, we can derive the main recoverability theorem based on the aforementioned lemmas and definitions.

*Theorem 1(Recoverability):* Under the sensing model described in (3), assume that $\boldsymbol{\mathcal{X}}^* = \varsigma(\boldsymbol{\mathcal{S}}^* \times_1 \mathbf{V}_1^* \times_2 \mathbf{V}_2^* \times_3 \mathbf{V}_3^*)$, where $\boldsymbol{\mathcal{S}}^*$ and $\{\mathbf{V}_l^*\}_{l=1}^3$ are obtained from any optimal solution of (19). Here, $\boldsymbol{\mathcal{S}}^*$ represents a sparse core tensor and the activation function $\varsigma(\cdot)$ is $T$-Lipschitz continuous. Also assume that $\boldsymbol{\mathcal{S}}^*$ and $\{\mathbf{V}_l^*\}_{l=1}^3$ satisfy the specifications in the solution set in *Definition 1*. Then, with a probability of at least $1 - \delta$, the following statement holds:

$$\frac{\|\boldsymbol{\mathcal{X}}^* - \boldsymbol{\mathcal{X}}_\natural\|_{\text{F}}}{\sqrt{I_1 I_2 I_3}} \leq \text{Gap}^*(\Omega) + (\frac{1}{\sqrt{I_1 I_2 I_3}} \|\boldsymbol{\mathcal{N}}\|_{\text{F}} + \frac{1}{\sqrt{|\Omega|}} \|\boldsymbol{\mathcal{O}} * \boldsymbol{\mathcal{N}}\|_{\text{F}}) + \frac{1}{\sqrt{|\Omega|}} \left\|\tilde{\boldsymbol{\mathcal{X}}}^* - \boldsymbol{\mathcal{X}}_\natural\right\|_{\text{F}}. \tag{28}$$

where $\tilde{\boldsymbol{\mathcal{X}}}^* = \arg\min_{\tilde{\boldsymbol{\mathcal{X}}} \in \mathcal{X}_{\text{TAP}}} \left\|\tilde{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}_\natural\right\|_{\text{F}}^2$ and $\text{Gap}^*(\Omega)$ is upper bounded by (27) with the covering number in (26).

*Proof*: See Appendix G. ∎

Theorem 1 presents a bound on the estimation error for recovering the ground-truth 3D physical fields, based on the criterion described in (19). Upon closer inspection of (28), it is readily seen that the recovery error stems from three primary sources: the generalization error $\text{Gap}^*(\Omega)$, the sensing noise $\boldsymbol{\mathcal{N}}$, and the representation error of the TAP model. Consequenlty, exact recovery occurs only when these three error sources simultaneously approach zero. This necessitates the noise to diminish, and the representation model to strike an optimal balance between conciseness (resulting in near-zero generalization error) and expressiveness (resulting in near-zero representation error).

Note that the number of the non-zero elements of $\boldsymbol{\mathcal{S}}$ represents a trade-off between the conciseness and expressiveness of TAP model, thereby influencing the generalization error $\text{Gap}^*(\Omega)$. An increase in the number of the non-zero elements in the attention map amplifies the generalization error $\text{Gap}^*(\Omega)$, but at the same time it enhances the model's capacity to represent a more intricate 3D field, thereby reducing the representation error $\left\|\tilde{\boldsymbol{\mathcal{X}}}^* - \boldsymbol{\mathcal{X}}_\natural\right\|_{\text{F}}$, and vise versa.

Further discussions, concerning comparisons of the recoverability analysis with the existing model [1], can be found in Appendix H. The impact of the sparsity level and observation patterns on the recoverability analysis is discussed in Appendix M. The main insight is that the proposed model has the potential to achieve lower reconstruction error than [1], particularly under distribution shifts. This advantage stems from the self-supervised learning of $||\boldsymbol{\mathcal{S}}||_0$, which allows the model's covering number (see Lemma 2) to adapt to the observed data, thereby reducing the generalization error. Moreover, since our model does not rely on any pretrained models, its representation error remains unaffected by distribution shifts. Our recoverability analysis above can also be easily extended to the MHTAP model.

## V. Experiments and Discussions

In this section, we present experimental results to demonstrate the effectiveness and versatility of the proposed methods using two 3D physical field datasets. We first compare our methods against state-of-the-art (SOTA) techniques and then carry out the ablation studies to draw insights concerning the proposed approach. The corresponding algorithms are implemented using PyTorch 1.13.1 and all experiments are performed on a RTX 4070 GPU with 8 GB of GPU memory.

### A. Ocean Sound Speed Field (SSF) Reconstruction

*3D SSF data:* In the following experiments, the South China Sea SSF dataset denoted as $\boldsymbol{\mathcal{X}}_\natural \in \mathbb{R}^{20\times20\times20}$ is utilized [10], [23]. The dataset covers a spatial area of 152km × 152km × 190m and has a horizontal resolution of 8 km and a vertical resolution of 10 m.

*Performance metric:* The reconstruction performance is evaluated by the root mean square error (RMSE) [10]:

$$\text{RMSE} = \sqrt{\frac{1}{I}\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}}_\natural\|_{\text{F}}^2}, \tag{29}$$

where $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{X}}_\natural$ represent the reconstructed SSF and the ground truth, respectively. The total number of SSF entries $I$

TABLE I: Average reconstruction errors (RMSEs) of the proposed methods and the benchmarks for different $\rho$ values. The bold and underlined numbers represent the lowest and second lowest RMSEs in the comparisons, respectively.

| Methods | $\rho = 5\%$ | $\rho = 10\%$ | $\rho = 20\%$ | $\rho = 30\%$ |
|---|---|---|---|---|
| Tucker-ALS | 2.666 | 1.527 | 0.663 | 0.411 |
| LRTC | 2.723 | 2.228 | 1.181 | 0.773 |
| TNN | 2.803 | 1.562 | 0.405 | 0.312 |
| TAP | **0.954** | <u>0.560</u> | <u>0.346</u> | <u>0.245</u> |
| MHTAP | <u>1.225</u> | **0.535** | **0.317** | **0.218** |

equals $I_1 \times I_2 \times I_3$. The reported RMSEs are averaged over 10 Monte-Carlo trials.

*Baseline:* Three unsupervised reconstruction methods, namely Tucker-ALS [37], LRTC [19], and TNN [10] are selected. The Tucker-ALS and LRTC methods are model-based approaches that utilize handcrafted priors, specifically the low-rankness property. The TNN method, on the other hand, can be regarded as an untrained deep-learning version of the Tucker-ALS method.

*Implementation Details:* See Appendix I.

*Results:* We first test the proposed methods using the noise-free SSF data under various observation rates $\rho = \frac{\|\boldsymbol{\mathcal{O}}\|_0}{I_1 \times I_2 \times I_3}$.

Table I presents the RMSEs of different algorithms across different $\rho$ values. As the observation rate increases, the reconstruction errors decrease for all algorithms. Notably, when $\rho \geq 20\%$, TNN demonstrates significant advantages over the model-based methods (Tucker-ALS and LRTC). Furthermore, the proposed methods consistently outperform the SOTA baselines in all scenarios. MHTAP exhibits superior performance compared to TAP when $\rho$ is higher ($\rho \geq 10\%$). This is attributed to the MHTAP's ability to capture more information from different subspaces of the input as more measurements of the SSF are observed. However, when $\rho$ is relatively low, MHTAP tends to overfit the limited observations due to its increased number of parameters. Fig. 8 illustrates the reconstructed SSF data of the five methods and their corresponding RMSEs at $\rho = 10\%$. TAP and MHTAP provide more accurate fits to the missing entries. Overall, the proposed methods achieve significantly superior results in both quantitative and visual evaluations compared to the very recent TNN model.

Next, we evaluate the performance of the proposed methods using noisy SSF data, where Gaussian noise with a mean of 0 and variance $\sigma^2$ is added according to the sensing model (3). Table II presents the RMSEs of different algorithms at various sampling ratios and noise powers. It is observed that the TNN algorithm proves ineffective at a low observation rate ($\rho = 10\%$). In contrast, the proposed methods outperform all the baseline algorithms across all $\rho$ values. Particularly, when the observation rate is very low (e.g., $\rho = 10\%$), TAP exhibits superior noise robustness compared to MHTAP, primarily due to its reduced number of parameters. Conversely, when the observation rate is high ($\rho \geq 20\%$), MHTAP achieves better performances than TAP.

TABLE II: Average reconstruction errors (RMSEs) of the proposed methods and the benchmarks for different $\rho$ values and Gaussian noise powers. The bold and underlined numbers represent the lowest and second lowest RMSEs in the comparisons, respectively.

| | Methods | $\rho = 10\%$ | $\rho = 20\%$ | $\rho = 30\%$ |
|---|---|---|---|---|
| $\sigma = 0.1$ | Tucker-ALS | 1.548 | 0.655 | 0.412 |
| | LRTC | 2.232 | 1.195 | 0.789 |
| | TNN | 1.683 | 0.471 | 0.329 |
| | TAP | **0.571** | <u>0.382</u> | <u>0.293</u> |
| | MHTAP | <u>0.609</u> | **0.346** | **0.261** |
| $\sigma = 0.2$ | Tucker-ALS | 1.546 | 0.663 | 0.425 |
| | LRTC | 2.247 | 1.226 | 0.826 |
| | TNN | 1.784 | 0.516 | 0.356 |
| | TAP | **0.600** | <u>0.425</u> | <u>0.351</u> |
| | MHTAP | <u>0.610</u> | **0.406** | **0.318** |
| $\sigma = 0.3$ | Tucker-ALS | 1.549 | 0.639 | 0.451 |
| | LRTC | 2.266 | 1.270 | 0.875 |
| | TNN | 2.170 | 0.561 | 0.423 |
| | TAP | **0.685** | <u>0.482</u> | <u>0.419</u> |
| | MHTAP | <u>0.714</u> | **0.471** | **0.388** |

TABLE III: Average reconstruction errors (RMSEs) of the proposed methods for different $\rho$ values and Laplacian noise powers.

| | Methods | $\rho = 10\%$ | $\rho = 20\%$ | $\rho = 30\%$ |
|---|---|---|---|---|
| $\sigma = 0.1$ | TAP (Laplacian) | 0.617 | 0.385 | 0.302 |
| | MHTAP (Laplacian) | 0.633 | 0.356 | 0.283 |
| $\sigma = 0.2$ | TAP (Laplacian) | 0.622 | 0.424 | 0.351 |
| | MHTAP (Laplacian) | 0.662 | 0.410 | 0.336 |
| $\sigma = 0.3$ | TAP (Laplacian) | 0.717 | 0.487 | 0.426 |
| | MHTAP (Laplacian) | 0.745 | 0.491 | 0.420 |

To evaluate the robustness of our method, we add to the observations Laplacian noise with a mean of 0 and variance $\sigma^2$ and assess the reconstruction performances. As shown in Table III, the reconstruction error increases only slightly in the presence of Laplacian noise, compared to Gaussian noise of the same power, indicating the robustness of our method to non-Gaussian noise sources.

### *B. Radio Map Tensor Reconstruction*

In this section, we conduct experiments on radio map tensor reconstruction.

*Data Description:* The radio map tensors used in the following experiments are generated from the joint path loss and log-normal shadowing model [1], [50], which can be expressed as:

$$\boldsymbol{\mathcal{X}}_{\natural} = \sum_{r=1}^{R} \mathbf{S}_r \circ \mathbf{c}_r \in \mathbb{R}^{I_1 \times I_2 \times I_3}. \tag{30}$$

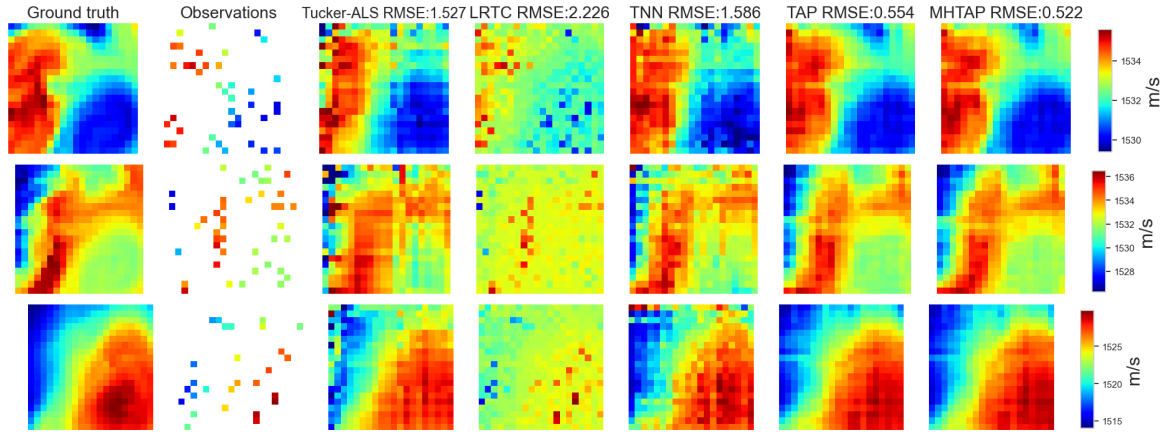



Fig. 8: Visualizations of ground-truth, observation and reconstructed SSF data of various methods at depth 0m (top), 100m (middle), 200m (bottom) under $\rho = 10\%$. The corresponding RMSEs are provided at the top of the figure.

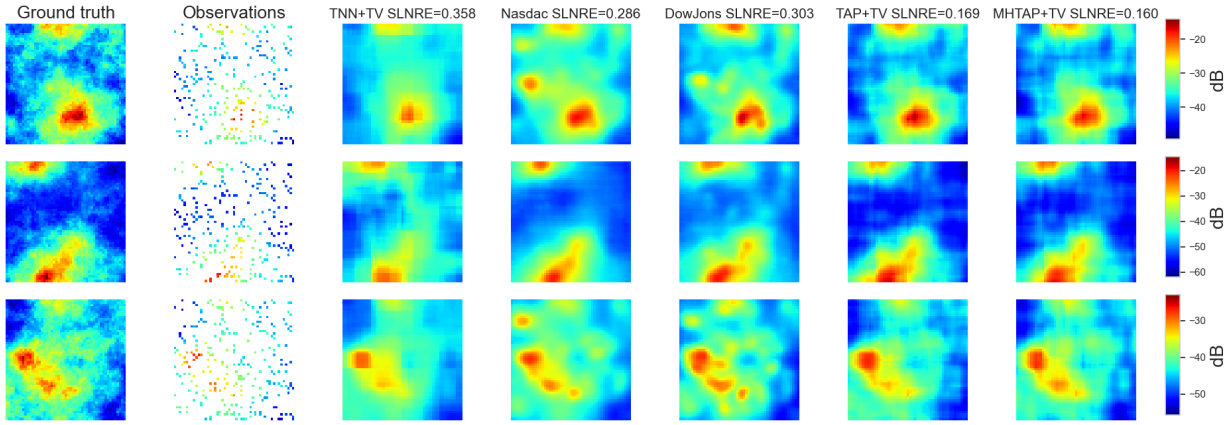



Fig. 9: Visualizations of ground-truth, observation and reconstructed radio maps of various methods at the 7-$th$ (top), 18-$th$ (middle) and 29-$th$ (bottom) frequency bins; $\rho = 10\%$, $R = 7$, $d_{\text{corr}} = 50$, $\eta = 10$. The corresponding SLNREs are provided at the top of the figure.

Here, $R$ is the number of emitters and $\mathbf{S}_r \in \mathbb{R}^{I_1 \times I_2}, \mathbf{c}_r \in \mathbb{R}^{I_3}$ is the spatial loss function (SLF) and power spectrum density (PSD) of $r$-$th$ emitter. Specifically, the $r$-$th$ SLF at location $\mathbf{m} = [i, j]^{\mathrm{T}}$ is generated by

$$\mathbf{S}_r(i,j) = \|\mathbf{m} - \mathbf{m}_r\|_2^{-\gamma_r} 10^{\frac{v_r(\mathbf{m})}{10}}, \tag{31}$$

where $\mathbf{m}_r$ is the coordinate of $r$-$th$ emitter, $\gamma_r$ is the $r$-$th$ path loss coefficient, and $v_r(\mathbf{m})$ is the correlated shadowing component that is generated from Gaussian distribution with zero mean and variance $\eta_r$. The auto-correlation function between $\mathbf{m}$ and $\mathbf{m}^{'}$ is $\mathbb{E}(v_r(\mathbf{m}), v_r(\mathbf{m}^{'})) = \eta_r \exp(-\|\mathbf{m} - \mathbf{m}^{'}\|_2 / d_{\text{corr}})$, in which $d_{\text{corr}}$ represents the decorrelation distance. The PSD of the emitter $r$ is given by $\mathbf{c}_r(k) = \sum_{i=1}^{M} a_{(i,r)} \text{sinc}^2(k - f_{(i,r)}/w_{(i,r)})$, where $a_{(i,r)}$ is the scaling factor of $r$-$th$ emitter at $i$-$th$ subband drawing from a uniform distribution over the interval $(0.5, 2.5)$; $f_{(1,r)}, \cdots, f_{(M,r)}$ are the central frequencies of $M$ subbands available to $r$-$th$ emitter and $M$ is set to 10. $w_{(i,r)}$ controls the width of sidelobe of $r$-$th$ emitter at $i$-$th$ subband drawing from an uniform distribution over the interval $(2, 4)$. In the experiments, $I_1 = I_2 = 51$, $I_3 = 64$.

*Performance metric:* The performance metric is the scaled log-domain normalized reconstruction error (SLNRE) [25]:

$$\text{SLNRE} = 100 \times \frac{\|\boldsymbol{\mathcal{X}}_{\log} - \boldsymbol{\mathcal{X}}_{\natural\log}\|_{\text{F}}^2}{\|\boldsymbol{\mathcal{X}}_{\natural\log}\|_{\text{F}}^2}, \tag{32}$$

where $\boldsymbol{\mathcal{X}}_{\log}$ is the log-transformed reconstructed radio map while $\boldsymbol{\mathcal{X}}_{\natural\log}$ being the log-transformed ground truth radio map. The SLNRE is an appropriate metric for skewed data like radio map [25]. All the SLNREs are averaged over 10 Monte-Carlo trials.

*Baseline:* We use SOTA untrained method TNN [10] and trained methods Nasdac and DowJons [1], to benchmark the proposed methods. Nasdac learns a generative deep network to reconstruct individual SLFs of the emitters from sensor measurements based on block term decomposition model. And DowJons leverages the generative deep network as structural constraints and reconstruct radio map tensors based on optimization criterion.

*Implementation Details:* See Appendix J.

*Results:* The first dataset (radio map 1) is generated using the typical setting described in [1], where the number of emitters $R = 7$, the decorrelation distance $d_{\text{corr}} = 50$ and the

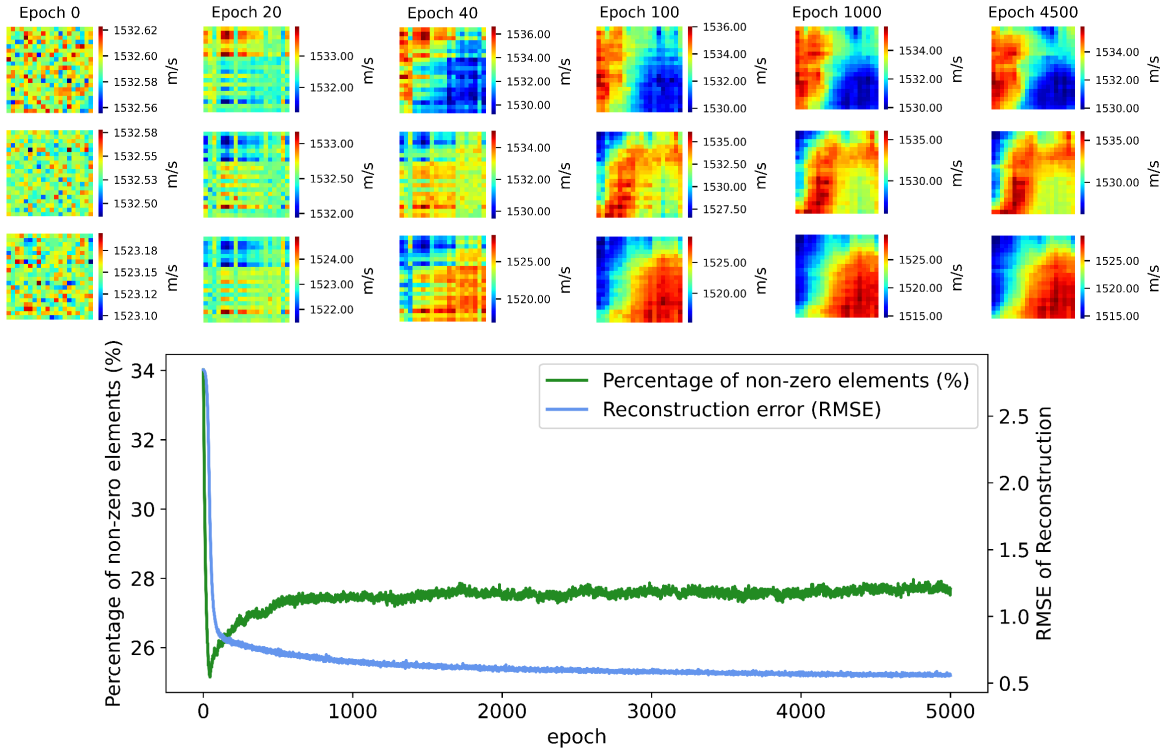



Fig. 10: Reconstruction results of the proposed FieldFormer on SSF data with $10\%$ observations over epochs. **Top**: Visualizations of reconstruction results over various epochs on depth 0m (top), 100m (middle), 200m (bottom). **Bottom**: Illustration of the percentage of the non-zero elements in the Tucker core tensor and the reconstruction error over epochs.

shadowing coefficient $\eta = 6$ for all $r$. In the subsequent experiments, we use a fiber-wise sampling methodology to obtain observed radio map tensors, with sensors randomly distributed across a 2D spatial domain to capture measurements over the entire frequency spectrum.

Table IV shows the SLNREs for the trained methods (Nasdac, DowJons), the untrained methods (TNN, TAP, MHTAP) and these untrained methods enhanced by total variation (TV) regularization [18] under various $\rho$ values on reconstructing radio map 1. Total variation regularization provides complementary structual regularization on the smoothness of the 3D field and serves as a widely used, easily implemented handcrafted prior in reconstruction tasks. One can see that the proposed TAP and MHTAP consistently perform better than TNN, with MHTAP consistently outperforming TAP. However, these untrained methods (TNN, TAP, MHTAP) are still inferior to Nasdac and DowJons, especially when $\rho$ is small, due to the lack of supervised prior knowledge. The performance gap can be simply filled in via incorporating a simple TV regularization to enforce the spatial smoothness of the reconstructed maps (details can be found in Appendix K). Specifically, the SLNRE of TNN decreases rapidly as $\rho$ increases and surpasses Nasdac and DowJons when $\rho \geq 15\%$, indicating that TNN becomes effective given enough observations. the proposed methods outperform all the baselines.

Fig. 9 provides the illustrative example given $\rho = 10\%, R = 7, d_{\text{corr}} = 50$ and $\eta = 10$, bridging the SLNREs and the visual quality of the reconstructions. First, both Nasdac and Dowjons accurately reconstruct the radio map. Then, these untrained methods, further aided by TV regularization which enforces multidimensional smoothness of the outputs, achieve good visual results given few observations. Finally, the proposed methods capture more fine-grained features of the radio map compared to the Nasdac and the DowJons.

Next, we consider radio map tensors with more complex shadowing environments. We create the radio map tensors with $R = 10$, $d_{\text{corr}} = 50$ under various $\eta_s$ (i.e., 9, 10, 12) for all $r$,

TABLE IV: Average reconstruction error (SLNREs) of different methods in reconstructing radio map 1 versus various $\rho$ values. Bold number and underlining number indicate the lowest and the second lowest SLNREs in the comparisons, respectively.

| Methods | $\rho = 5\%$ | $\rho = 10\%$ | $\rho$=15% | $\rho = 20\%$ |
|---|---|---|---|---|
| Nasdac | 0.246 | 0.192 | 0.179 | 0.161 |
| DowJons | 0.216 | 0.173 | 0.167 | 0.149 |
| TNN | 83.65 | 25.72 | 1.574 | 0.337 |
| TAP | 7.132 | 0.914 | 0.489 | 0.223 |
| MHTAP | 5.517 | 0.480 | 0.326 | 0.168 |
| TNN+TV | 0.539 | 0.242 | 0.158 | 0.116 |
| TAP+TV | <u>0.189</u> | <u>0.100</u> | <u>0.073</u> | <u>0.067</u> |
| MHTAP+TV | **0.180** | **0.091** | **0.071** | **0.065** |

TABLE V: Average reconstruction error (SLNREs) of different methods in radio map reconstruction task under $\rho = 10\%$ versus various $\eta_s$. Bold number and underlining number indicate the lowest and the second lowest SLNREs in the comparisons, respectively.

| Methods | $\eta = 9$ | $\eta = 10$ | $\eta = 12$ |
|---|---|---|---|
| Nasdac | 0.238 | 0.286 | 0.667 |
| DowJons | 0.232 | 0.303 | 0.631 |
| TNN+TV | 0.304 | 0.369 | 0.411 |
| TAP+TV | <u>0.130</u> | <u>0.170</u> | <u>0.232</u> |
| MHTAP+TV | **0.128** | **0.162** | **0.222** |

which are the out-of-training distribution data for the Nasdac and the DowJons. In the subsequent experiments, the untrained methods are all enhanced by TV regularization.

Table. V displays the SLNREs of different methods given $\rho = 10\%$. One can see that Nasdac and the DowJons deteriorate in reconstructing the radio maps that they have never seen before, being surpassed by TNN when $\eta = 12$. With TV regularization, the TAP and MHTAP consistently outperform all the baselines and MHTAP still remains superior to TAP. These experiments verify that the proposed methods generalize better in handling 3D physical fields of varying complexity levels without the cost of extensive training and the risk of inaccurately estimating the number of emitters.

We then provide detailed analysis of why the proposed approach outperforms the baseline methods in certain scenarios (e.g., at low observation rates), particularly by examining the specific properties of the proposed method. Both Tucker-ALS and TNN, like the proposed FieldFormer, are based on the Tucker model. However, they share a key limitation: the need to predefine the size of the Tucker core, which directly determines the model's complexity. Although we carefully tune hyperparameters (e.g., the multilinear rank) over a large search space to achieve their best overall performance, these methods – due to their fixed, non-adaptive model complexity – struggle to adapt to different observation scenarios, particularly at low observation rates. As shown in Fig. 8, both methods tend to overfit the noise in sparse observations.

LRTC, which is based on the t-SVD model, seeks to minimize the tensor tubal rank while reconstructing the entire field from sparse observations. However, this objective function prevents LRTC from capturing the fine details of the physical field, leading to underfitting – particularly when reconstructing complex structures such as the SSF, as it is illustrated in Fig. 8.

Nasdac and DowJons, which are trained on pre-collected data, rely on the assumption that the training and test data share similar distributions. However, in real-world scenarios, where data distribution shifts occur, these pre-trained models fail to generalize effectively, leading to significant performance degradation.

In contrast, the proposed FieldFormer dynamically adjusts its model complexity to fit the observations without requiring training data. This adaptability helps mitigate *overfitting, underfitting, and sensitivity to data distribution shifts*, leading to superior performance compared to the baseline methods.

### C. Complexity-adaptive Neural Representation

In this subsection, we provide detailed discussions and experimental results to explain how the attention mechanism adjusts model complexity. The proposed FieldFormer, based on the Tucker model, leverages an attention mechanism to dynamically control the sparsity of the core tensor during the reconstruction process. Fig. 10 provides detailed visualizations of the intermediate steps on reconstructing the SSF data. The top of Fig. 10 illustrates reconstruction results at different epochs, while the bottom shows the percentage of the non-zero elements in the Tucker core tensor and the corresponding reconstruction error over epochs. It is readily observed that the reconstruction results progress *from coarse to fine as the number of epochs increases*. The percentage of the non-zero elements drops sharply before gradually rising to a plateau, while the reconstruction error consistently decreases. This phenomenon suggests that the model complexity is *dynamically adjusted* throughout the reconstruction process.

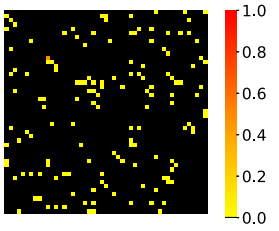
At the beginning of reconstruction, the parameters of the attention mechanism (i.e., $\mathbf{W}_Q, \mathbf{W}_K$) are randomly initialized, causing similarity scores between different cubes to be randomly distributed. Consequently, the percentage of the non-zero elements in the core tensor is relatively high. As the model optimizes the loss function, the attention mechanism becomes more effective at capturing meaningful similarities between different cubes. This leads to a reduction in the percentage of the non-zero elements, as similarity scores concentrate on the most relevant pairs. More specifically, at the early epochs, since the parameters are updated in the direction of *the steepest descent* of the loss function, the model primarily fits the lower-frequency components (e.g., the mean value) of the field, as shown in Fig. 10. This is reflected in the rapid decrease in reconstruction error and the relatively coarse visual appearance of the reconstruction results. During this phase, model complexity remains low, as indicated by a low percentage of the non-zero elements in the Tucker core tensor. As training progresses, in order to further decrease the reconstruction loss, the model gradually captures higher-frequency components of the field, leading to an increase in model complexity. As shown in Fig. 10, the reconstruction error declines more slowly in later stages, while the reconstruction results become increasingly fine-grained. This explains the subsequent rise in the percentage of the non-zero elements in the core tensor. Eventually, the model reaches an *optimal level of complexity* for representing the field, after which the percentage of the non-zero elements stabilizes. We further provide discussions on the trade-off between computational cost and reconstruction accuracy in Appendix N.

### D. Ablation Study

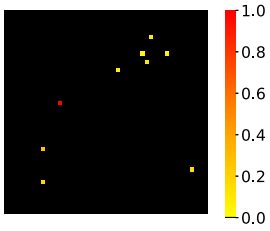
*1) Impact of Prior Information from Partial Observations Incorporated by Attention Mechanism:* In this paper, we propose using dot-product between query and key (generated from extracted cubes $\mathbf{P}$) to gain informative priors from the limited

TABLE VI: Ablation study on reconstructing radio map 1 with SLNRE metric. All the methods are enhanced by TV regularization.

| Methods (+TV) | $\rho = 5\%$ | $\rho = 10\%$ | $\rho$=15% | $\rho = 20\%$ |
|---|---|---|---|---|
| TAP | **0.189** | **0.100** | **0.073** | **0.067** |
| TAP wo OI | 13.75 | 0.911 | 0.574 | 0.632 |
| MHTAP | **0.180** | **0.091** | **0.071** | **0.065** |
| MHTAP wo OI | 0.385 | 0.263 | 0.284 | 0.261 |




(a) Attention map (SoftMax).




(b) Attention map (SparseMax).

Fig. 11: Partial visulization of attention maps learned by TAP under $\rho = 20\%$ using SoftMax and SparseMax activation function respectively.

observations. This approach allows for adaptively tuning the sparsity of the core tensor in a self-supervised manner.

Therein, we conduct the ablation study on TAP/MHTAP in reconstructing radio map 1 to prove its efficacy. For comparison, we replace $\mathbf{P}$ containing observation information with noise matrix $\mathbf{N}_\varepsilon$ sharing the same shape with $\mathbf{P}$. Therefore, the original sparse core tensor of TAP $\boldsymbol{\mathcal{S}}$ in (19) becomes $\text{Tensorize}(\text{SparseMax}([\mathbf{N}_\varepsilon\mathbf{W}_Q][\mathbf{N}_\varepsilon\mathbf{W}_K]^{\mathrm{T}} \oslash \mathbf{M})$. Similarly, the ablated sparse core tensor of MHTAP is $\text{Tensorize}(\text{SparseMax}(\text{Concat}([\mathbf{N}_\varepsilon\mathbf{W}_Q^1][\mathbf{N}_\varepsilon\mathbf{W}_K^1]^{\mathrm{T}} \oslash \mathbf{M}_1, \cdots, [\mathbf{N}_\varepsilon\mathbf{W}_Q^h][\mathbf{N}_\varepsilon\mathbf{W}_K^h]^{\mathrm{T}} \oslash \mathbf{M}_h)))$. *In this way, the information prior from partial observations is discarded.*

Table VI shows the average results of TAP/MHTAP compared to TAP/MHTAP without observation information (OI). The performances deteriorate significantly in the absence of the observation information, indicating that the self-supervised prior extracted from the limited observations is useful in the proposed methods in the 3D physical field reconstruction task.

*2) Impact of SparseMax Function:* We introduce the SparseMax function into the TAP/MHTAP model to boost the sparsity of the attention map. In this subsection, we compare its performance with the commonly used SoftMax function to demonstrate its effectiveness. The following experiments are conducted on SSF data. Tab. VII shows the average reconstruction results of TAP/MHTAP using SoftMax function or using SparseMax function under various $\rho$ values. It indicates that the utilization of SparseMax activation enhances performance, particularly when $\rho$ is small. For enhanced visual clarity, we depict $50 \times 50$ sub-matrices extracted from attention maps learned by TAP using SoftMax and SparseMax activation in Fig. 11. The attention map learned using the SoftMax function

TABLE VII: Average reconstruction errors (RMSEs) of TAP and MHTAP using SparseMax activation or SoftMax activation versus various $\rho$ values.

| Methods | $\rho = 5\%$ | $\rho = 10\%$ | $\rho = 20\%$ | $\rho$=30% |
|---|---|---|---|---|
| TAP (SparseMax) | **1.085** | **0.564** | **0.351** | **0.236** |
| TAP (SoftMax) | 1.391 | 0.745 | 0.363 | 0.246 |
| MHTAP (SparseMax) | **1.349** | **0.543** | **0.319** | **0.215** |
| MHTAP (SoftMax) | 1.626 | 0.714 | 0.347 | 0.231 |

typically consists of many uniformly small scores, whereas the attention map learned using the SparseMax function yields fewer but larger scores. These fewer but larger scores effectively capture the main features of the signals and alleviate overfitting in the reconstruction task.

## VI. Conclusion and Future Directions

In this paper, we introduced FieldFormers based on TAP and MHTAP models for self-supervised reconstruction of 3D physical fields from limited observations. By bridging the Tucker model with the attention scheme, the proposed approach effectively captured both short- and long-range correlations within the limited observations. This learned information was then utilized to adaptively adjust the complexity of the tensor Tucker model, specifically the sparsity of the core tensor. As a result, our method demonstrated promising generalization performance across various types of 3D physical fields. Theoretical analysis was provided to characterize the recovery of ground-truth 3D physical fields using the proposed methods. Additionally, the analysis shed light on the sources of reconstruction errors. Furthermore, we conducted numerical experiments on diverse datasets, including ocean sound speed fields and radio maps, to validate the superiority of the proposed methods in 3D physical field reconstruction compared to SOTA baselines. While the proposed method has demonstrated strong performance in reconstructing 3D physical fields from limited observations, several directions remain open for future research. In particular, reducing the computational complexity of the tensor attention mechanism would enhance scalability. Also, developing a generative pretrained version of the proposed approach, which is robust to distribution shifts, could further enhance the reconstruction performance, even under extremely limited observations.

## References

[1] S. Shrestha, X. Fu, and M. Hong, "Deep spectrum cartography: Completing radio map tensors using learned neural models," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1170–1184, 2022.

[2] D. Romero and S.-J. Kim, "Radio map estimation: A data-driven approach to spectrum cartography," *IEEE Signal Processing Magazine*, vol. 39, no. 6, pp. 53–72, 2022.

[3] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Group-lasso on splines for spectrum cartography," *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4648–4663, 2011.

[4] L. Cheng, X. Ji, H. Zhao, J. Li, and W. Xu, "Tensor-based basis function learning for three-dimensional sound speed fields," *The Journal of the Acoustical Society of America*, vol. 151, no. 1, pp. 269–285, 01 2022. [Online]. Available: https://doi.org/10.1121/10.0009280

[5] Y. Yue, H. Zheng, Z. Shi, and G. Liao, "Two-stage reconstruction for co-array 2d doa estimation of mixed circular and noncircular signals," *IEEE Transactions on Vehicular Technology*, pp. 1–15, 2025.

[6] Y. Yue, Z. Zhang, and Z. Shi, "Generalized widely linear robust adaptive beamforming: A sparse reconstruction perspective," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 5, pp. 5663–5673, 2024.

[7] R. Dashen, W. H. Munk, and K. M. Watson, *Sound transmission through a fluctuating ocean*. Cambridge University Press, 2010.

[8] Z.-Q. Luo, X. Zheng, D. López-Pérez, Q. Yan, X. Chen, N. Wang, Q. Shi, T.-H. Chang, and A. Garcia-Rodriguez, "Srcon: A data-driven network performance simulator for real-world wireless networks," *IEEE Communications Magazine*, vol. 61, no. 6, pp. 96–102, 2023.

[9] D. Behringer, T. Birdsall, M. Brown, B. Cornuelle, R. Heinmiller, R. Knox, K. Metzger, W. Munk, J. Spiesberger, R. Spindel *et al.*, "A demonstration of ocean acoustic tomography," *Nature*, vol. 299, no. 5879, pp. 121–125, 1982.

[10] S. Li, L. Cheng, T. Zhang, H. Zhao, and J. Li, "Striking the right balance: Three-dimensional ocean sound speed field reconstruction using tensor neural networks," *The Journal of the Acoustical Society of America*, vol. 154, no. 2, pp. 1106–1123, 2023.

[11] G. Zhang, X. Fu, J. Wang, X.-L. Zhao, and M. Hong, "Spectrum cartography via coupled block-term tensor decomposition," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3660–3675, 2020.

[12] W. C. Van Beers and J. P. Kleijnen, "Kriging interpolation in simulation: a survey," in *Proceedings of the 2004 Winter Simulation Conference, 2004.*, vol. 1. IEEE, 2004.

[13] G. Boccolini, G. Hernandez-Penaloza, and B. Beferull-Lozano, "Wireless sensor network for spectrum cartography based on kriging interpolation," in *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC)*. IEEE, 2012, pp. 1565–1570.

[14] S.-J. Kim and G. B. Giannakis, "Cognitive radio spectrum prediction using dictionary learning," in *2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 3206–3211.

[15] Y. Teganya and D. Romero, "Deep completion autoencoders for radio map estimation," *IEEE Transactions on Wireless Communications*, vol. 21, no. 3, pp. 1710–1724, 2021.

[16] X. Han, L. Xue, F. Shao, and Y. Xu, "A power spectrum maps estimation algorithm based on generative adversarial networks for underlay cognitive radio networks," *Sensors*, vol. 20, no. 1, p. 311, 2020.

[17] J. Li and A. D. Heap, "Spatial interpolation methods applied in the environmental sciences: A review," *Environmental Modelling & Software*, vol. 53, pp. 173–189, 2014.

[18] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[19] Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, "Novel methods for multilinear data completion and de-noising based on tensor-svd," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3842–3849.

[20] S. Li, L. Cheng, T. Zhang, H. Zhao, and J. Li, "Graph-guided bayesian matrix completion for ocean sound speed field reconstruction," *The Journal of the Acoustical Society of America*, vol. 153, no. 1, pp. 689–710, 2023.

[21] S. Bi, J. Lyu, Z. Ding, and R. Zhang, "Engineering radio maps for wireless resource management," *IEEE Wireless Communications*, vol. 26, no. 2, pp. 133–141, 2019.

[22] M. Bianco and P. Gerstoft, "Dictionary learning of sound speed profiles," *The Journal of the Acoustical Society of America*, vol. 141, no. 3, pp. 1749–1758, 03 2017. [Online]. Available: https://doi.org/10.1121/1.4977926

[23] P. Chen, L. Cheng, T. Zhang, H. Zhao, and J. Li, "Tensor dictionary learning for representing three-dimensional sound speed fields," *The Journal of the Acoustical Society of America*, vol. 152, no. 5, pp. 2601–2616, 11 2022. [Online]. Available: https://doi.org/10.1121/10.0015056

[24] Y. Teganya and D. Romero, "Data-driven spectrum cartography via deep completion autoencoders," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–7.

[25] S. Timilsina, S. Shrestha, and X. Fu, "Quantized radio map estimation using tensor and deep generative models," *IEEE Transactions on Signal Processing*, 2023.

[26] A. Qayyum, I. Ilahi, F. Shamshad, F. Boussaid, M. Bennamoun, and J. Qadir, "Untrained neural network priors for inverse imaging problems: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, pp. 6511–6536, 2023.

[27] J. Baxter, "A model of inductive bias learning," *Journal of artificial intelligence research*, vol. 12, pp. 149–198, 2000.

[28] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.

[29] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, "Tensor-based formulation and nuclear norm regularization for multienergy computed tomography," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1678–1693, 2014.

[30] Y. Zheng, J. Li, Y. Li, J. Guo, X. Wu, and J. Chanussot, "Hyperspectral pansharpening using deep prior and dual attention residual network," *IEEE transactions on geoscience and remote sensing*, vol. 58, no. 11, pp. 8059–8076, 2020.

[31] S. Theodoridis, *Machine Learning: From the Classics to Deep Networks, Transformers, and Diffusion Models*. Academic Press, 2025.

[32] Y. Bengio, Y. Lecun, and G. Hinton, "Deep learning for ai," *Communications of the ACM*, vol. 64, no. 7, pp. 58–65, 2021.

[33] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE transactions on knowledge and data engineering*, vol. 35, no. 1, pp. 857–876, 2021.

[34] W. He, T. Uezato, and N. Yokoya, "Interpretable deep attention prior for image restoration and enhancement," *IEEE Transactions on Computational Imaging*, vol. 9, pp. 185–196, 2023.

[35] Y.-S. Luo, X.-L. Zhao, T.-X. Jiang, Y. Chang, M. K. Ng, and C. Li, "Self-supervised nonlinear transform-based tensor nuclear norm for multi-dimensional image recovery," *IEEE Transactions on Image Processing*, vol. 31, pp. 3793–3808, 2022.

[36] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A singular value decomposition for higher-order tensors," in *Second ATHOS workshop, Sophia-Antipolis, France*, 1993.

[37] 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.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[39] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[40] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.

[41] X. Tong, L. Cheng, and Y.-C. Wu, "Bayesian tensor tucker completion with a flexible core," *IEEE Transactions on Signal Processing*, 2023.

[42] F. Roemer, G. Del Galdo, and M. Haardt, "Tensor-based algorithms for learning multidimensional separable dictionaries," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 3963–3967.

[43] A. Martins and R. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *International conference on machine learning*. PMLR, 2016, pp. 1614–1623.

[44] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231222008426

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1026–1034.

[46] A. Barakat and P. Bianchi, "Convergence and dynamical behavior of the adam algorithm for nonconvex stochastic optimization," *SIAM Journal on Optimization*, vol. 31, no. 1, pp. 244–274, 2021. [Online]. Available: https://doi.org/10.1137/19M1263443

[47] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*. Cambridge university press, 2012.

[48] H. Rauhut, R. Schneider, and Ž. Stojanac, "Low rank tensor recovery via iterative hard thresholding," *Linear Algebra and its Applications*, vol. 523, pp. 220–262, 2017.

[49] D.-X. Zhou, "The covering number in learning theory," *Journal of Complexity*, vol. 18, no. 3, pp. 739–767, 2002.

[50] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[51] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are rnns: Fast autoregressive transformers with linear attention," in

*International conference on machine learning*. PMLR, 2020, pp. 5156–5165.

## APPENDIX

### A. Pseudo code of tensorization

```
def tensorize_to_core(s, flag):
if flag == 'TAP': #s:sparse attention map(N*N)
s = s.view(J1, J2, J3, J1, J2, J3)
s = s.permute(0,3,1,4,2,5)
core_tensor = s.reshape(N1, N2, N3)
elif flag == 'MHTAP': #s:sparse attention map(hN*N)
s = s.view(h1, h2, h3, J1, J2, J3, J1, J2, J3)
s = s.permute(0,3,6,1,4,7,2,5,8)
core_tensor = s.reshape(h1*N1, h2*N2, h3*N3)
return core_tensor
```

### B. 3D FieldFormer based on MHTAP.

The algorithm is summarized in **Algorithm 2**.

**Algorithm 2** 3D FieldFormer based on MHTAP.

**Input:** Observations $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, indicating tensor $\boldsymbol{\mathcal{O}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, window size in three mode $(K_1, K_2, K_3)$, stride size in three mode $(S_1, S_2, S_3)$;

**Initialization:** Initialize the query and key matrices $\{\mathbf{W}_Q^i, \mathbf{W}_K^i\}_{i=1}^h$ as well as the value matrices $\mathbf{V}_1 \in \mathbb{R}^{I_1 \times N_1}, \mathbf{V}_2 \in \mathbb{R}^{I_2 \times N_2}, \mathbf{V}_3 \in \mathbb{R}^{I_3 \times N_3}$.

1: Extract cubes from observations $\boldsymbol{\mathcal{Y}}$ to get $\mathbf{P}$.
2: **while** not converge **do**
3: Compute the query-key pairs through $\{\mathbf{Q}_i = \mathbf{P}\mathbf{W}_Q^i, \mathbf{K}_i = \mathbf{P}\mathbf{W}_K^i\}_{i=1}^h$.
4: Compute the output of sparse attention module through Eq. (17) and obtain reconstruction with $\varsigma(\text{MHSTA}(\{\mathbf{Q}_i\}_{i=1}^h, \{\mathbf{K}_i\}_{i=1}^h, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3))$.
5: Compute the loss $\|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * \boldsymbol{\mathcal{X}}\|_{\mathrm{F}}^2$
6: Update $\{\mathbf{W}_Q^i, \mathbf{W}_K^i\}_{i=1}^h, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$ according to the loss using the Adam optimizer.
7: **end while**

**Output:** The reconstructed 3D physical field $\boldsymbol{\mathcal{X}}$.

### C. Implementation details for the SparseMax function

The algorithm is summarized in **Algorithm 3**.

### D. Proof of Lemma 1

Without loss of generality, assume all entries in $\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M}$ are independent random variables with mean 0 and variance 1. SparseMax applies row-wise sparse normalization and we treat all rows equivalently, so we take a specific row of $\mathbf{Q}\mathbf{K}^{\mathrm{T}} \oslash \mathbf{M}$, denoted by $\mathbf{z} = \left[z_{(1)}, z_{(2)}, \cdots, z_{(N)}\right] \in \mathbb{R}^N$ for example. Firstly, SparseMax sorts $\mathbf{z}$ as $z_1 \leq z_2 \leq \cdots \leq z_N$. Secondly, it compute support number $n(\mathbf{z}) = \max\{n \in [N] | 1 + nz_n > \sum_{j \leq n} z_j\}$. Thirdly, it computes normalization coefficient $\tau(\mathbf{z}) = \frac{\sum_{j \leq n(\mathbf{z})} z_j - 1}{n(\mathbf{z})}$ with $n(\mathbf{z})$ leading scores. Finally, $n(\mathbf{z})$ leading scores are normalized with summation to 1 while others are set to zeroes through threshold function $[z_i - \tau(\mathbf{z}))]_+$. It is worthy to note that given a specific number $k$, if $1 + kz_k \leq \sum_{j \leq k} z_j$, then $1 + (k+1)z_{k+1} \leq 1 + kz_k + z_{k+1} \leq \sum_{j \leq k} z_j + z_{k+1} = \sum_{j \leq k+1} z_j$ holds, which means that if $n = k$ does not satisfy the inequality

**Algorithm 3** SparseMax Function.

**Input:** Vector $\mathbf{z} = \left[z_{(1)}, z_{(2)}, \cdots, z_{(N)}\right] \in \mathbb{R}^N$.

1: Sort the elements of $\mathbf{z}$ in descending order:

$$z_1 \geq z_2 \geq \cdots \geq z_N.$$

2: Determine the support number $n(\mathbf{z})$ satisfying:

$$n(\mathbf{z}) = \max\left\{ n \in \{1, \ldots, N\} \mid 1 + nz_n > \sum_{j=1}^{n} z_j \right\}.$$

3: Compute the threshold $\tau(\mathbf{z})$:

$$\tau(\mathbf{z}) = \frac{\sum_{j \leq n(\mathbf{z})} z_j - 1}{n(\mathbf{z})}.$$

4: Compute the final normalized output:

$$p_{(i)} = \max(z_{(i)} - \tau, 0), \quad \forall i \in \{1, \ldots, N\}.$$

**Output:** Normalized output $\mathbf{p} = \left[p_{(1)}, p_{(2)}, \cdots, p_{(N)}\right] \in \mathbb{R}^N$.

$(1 + nz_n > \sum_{j \leq n} z_j)$, then any $n > k$ does not satisfy the inequality as well. Therefore, if $n = k-1$ satisfy the inequality but $n = k$ do not, then $n(\mathbf{z}) = k - 1$ holds.

Before computing $\mathbb{E}(n(\mathbf{z}))$, we need to acquire the probability distribution of discrete random variable $n(\mathbf{z})$. Given the above property, we have:

$$\begin{aligned} &P(n(\mathbf{z}) = 1) = \\ &P(1 + z_1 > z_1) P(1 + 2z_2 \leq z_1 + z_2 | z_2 < z_1) = \\ &1 \times \frac{P(1 + z_2 \leq z_1, z_2 < z_1)}{P(z_2 < z_1)} = \frac{P(z_1 - z_2 \geq 1)}{P(z_2 - z_1 < 0)} \\ &= \frac{P(\frac{z_1 - z_2}{\sqrt{2}} \geq \frac{1}{\sqrt{2}})}{P(\frac{z_2 - z_1}{\sqrt{2}} < 0)} = \frac{1 - \Phi(\frac{1}{\sqrt{2}})}{\Phi(0)}, \end{aligned} \tag{33}$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of Gaussian distribution with mean 0 and variance 1:

$$\Phi(X) = \int_{-\infty}^{X} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx. \tag{34}$$

Similarly, we can get $P(n(\mathbf{z}) = 2)$:

$$\begin{aligned} &P(n(\mathbf{z}) = 2) = \\ &P(1 + z_1 > z_1) P(1 + 2z_2 \geq z_1 + z_2 | z_2 < z_1) \\ &P(1 + 3z_3 \leq z_1 + z_2 + z_3 | z_3 < z_2) = \\ &\frac{P(1 + 2z_2 \geq z_1 + z_2, z_2 < z_1)}{P(z_2 < z_1)} \times \\ &\frac{P(1 + 3z_3 \leq z_1 + z_2 + z_3, z_3 < z_2)}{P(z_3 < z_2)} = \\ &\frac{P(0 < z_1 - z_2 < 1)}{P(z_2 < z_1)} \times \frac{P(z_1 + z_2 - 2z_3 \geq 1)}{P(z_3 < z_2)} = \\ &\frac{\Phi(\frac{1}{\sqrt{2}}) - \Phi(0)}{\Phi(0)} \times \frac{1 - \Phi(\frac{1}{2})}{\Phi(0)}. \end{aligned} \tag{35}$$

To conclude, for $N > 3$, we have:

$$\begin{aligned} &P(n(\mathbf{z}) = 1) = 2 - 2\Phi(\frac{1}{\sqrt{2}}), \\ &P(n(\mathbf{z}) = n) = (2 - 2\Phi(\frac{1}{\sqrt{2n}})) \times \prod_{i=2}^{n} (2\Phi(\frac{1}{2i-2}) - 1), \\ &\qquad\qquad n = 2, \cdots, N-1, \\ &P(n(\mathbf{z}) = N) = \prod_{i=2}^{N} (2\Phi(\frac{1}{\sqrt{2i-2}}) - 1). \end{aligned} \tag{36}$$

We can infer from Eq. (36) that $P(n(\mathbf{z}))$ is a constant independent of $N$ and that most of the probability density falls into small values. And the expectation of $n(\mathbf{z})$ can be computed through $\mathbb{E}(n(\mathbf{z})) = \sum_i^N iP(n(\mathbf{z}) = i)$. Therefore, the expectation of the number of the non-zero elements in attention map (i.e., $\|\boldsymbol{\mathcal{S}}\|_0$) equals $N\mathbb{E}(n(\mathbf{z}))$.

### *E. Proof of Lemma 2*

*Lemma 4* [47] *:* Let $\mathcal{Z}$ be a subset of a vector space of real dimension $R$ with unit norm ($\|\mathcal{Z}\|_{\mathrm{F}} = 1$) and let $0 < \varepsilon < 1$. The covering number of $\mathcal{Z}$ scaled by $\varepsilon$ is upper bounded by:

$$\mathsf{N}(\mathcal{Z}, \varepsilon) \leq \left(\frac{3}{\varepsilon}\right)^R. \tag{37}$$

■

Consider the set $\mathcal{S} = \{\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N_1 \times N_2 \times N_3}, \|\boldsymbol{\mathcal{S}}\|_{\mathrm{F}} \leq \alpha\}$ containing the core tensor and $\mathcal{V}_1 = \{\mathbf{V}_1 \in \mathbb{R}^{I_1 \times N_1}, \|\mathbf{V}_1\|_{\mathrm{F}} \leq \beta\}, \mathcal{V}_2 = \{\mathbf{V}_2 \in \mathbb{R}^{I_2 \times N_2}, \|\mathbf{V}_2\|_{\mathrm{F}} \leq \beta\}, \mathcal{V}_3 = \{\mathbf{V}_3 \in \mathbb{R}^{I_3 \times N_3}, \|\mathbf{V}_3\|_{\mathrm{F}} \leq \beta\}$ containing the factor matrices. Since $\mathcal{S}$ is an Euclidean ball that has a radius of $\alpha$ in the $\|\boldsymbol{\mathcal{S}}\|_0$-dimensional space. Then, according to *Lemma 4*, the covering number of $\mathcal{S}$ is upper bounded by :

$$\mathsf{N}(\mathcal{S}, \varepsilon) \leq \left(\frac{3\alpha}{\varepsilon}\right)^{\|\boldsymbol{\mathcal{S}}\|_0}. \tag{38}$$

The above leads to the following upper bound:

$$\mathsf{N}(\mathcal{S}, \frac{\varepsilon}{\beta^3 + 3\alpha\beta^2}) \leq \left(\frac{3\alpha(\beta^3 + 3\alpha\beta^2)}{\varepsilon}\right)^{\|\boldsymbol{\mathcal{S}}\|_0}. \tag{39}$$

Also, $\mathcal{V}_l$ is an Euclidean ball that has a radius of $\beta$ in the $I_l N_l$-dimensional space. Similarly, the covering number of $\mathcal{V}_l$ is upper bounded by :

$$\mathsf{N}(\mathcal{V}_l, \varepsilon) \leq \left(\frac{3\beta}{\varepsilon}\right)^{I_l N_l}, \tag{40}$$

for $i = 1, 2, 3$. This also holds that:

$$\mathsf{N}(\mathcal{V}_l, \frac{\varepsilon}{\beta^3 + 3\alpha\beta^2}) \leq \left(\frac{3\beta(\beta^3 + 3\alpha\beta^2)}{\varepsilon}\right)^{I_l N_l}, \tag{41}$$

for $i = 1, 2, 3$. Here we define the set $\mathcal{X} = \{\tilde{\boldsymbol{\mathcal{X}}} = \tilde{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3 | \tilde{\boldsymbol{\mathcal{S}}} \in \mathcal{S}, \{\tilde{\mathbf{V}}_l \in \mathcal{V}_l\}_{i=1}^3\}$.

Let $\bar{\mathcal{S}}$ be an $\frac{\varepsilon}{\beta^3 + 3\alpha\beta^2}$-net of $\mathcal{S}$ and $\bar{\mathcal{V}}_l$ be an $\frac{\varepsilon}{\beta^3 + 3\alpha\beta^2}$-net of $\mathcal{V}_l$ for $i = 1, 2, 3$, respectively. And we define the set $\bar{\mathcal{X}} =$

$\{\bar{\boldsymbol{\mathcal{X}}} = \bar{\boldsymbol{\mathcal{S}}} \times_1 \bar{\mathbf{V}}_1 \times_2 \bar{\mathbf{V}}_2 \times_3 \bar{\mathbf{V}}_3 | \bar{\boldsymbol{\mathcal{S}}} \in \bar{\mathcal{S}}, \{\bar{\mathbf{V}}_l \in \bar{\mathcal{V}}_l\}_{l=1}^3\}$. Hence, the cardinality of $\bar{\mathcal{X}}$ is upper bounded by

$$\begin{aligned} &|\bar{\mathcal{X}}| \leq \\ &\left(\frac{3\alpha(\beta^3+3\alpha\beta^2)}{\varepsilon}\right)^{\|\boldsymbol{\mathcal{S}}\|_0} \prod_{i=1}^{3}\left(\frac{3\beta(\beta^3+3\alpha\beta^2)}{\varepsilon}\right)^{I_l N_l} \\ &= \left[\frac{3(\beta^3+3\alpha\beta^2)}{\varepsilon}\right]^{\|\boldsymbol{\mathcal{S}}\|_0+\sum_{l=1}^3 N_l I_l} \alpha^{\|\boldsymbol{\mathcal{S}}\|_0}\beta^{\sum_{l=1}^3 N_l I_l} \end{aligned} \tag{42}$$

Consider $\tilde{\boldsymbol{\mathcal{X}}} \in \mathcal{X}$ and $\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}$, which can be represented as $\tilde{\boldsymbol{\mathcal{X}}} = \tilde{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3, \tilde{\boldsymbol{\mathcal{S}}} \in \mathcal{S}, \{\tilde{\mathbf{V}}_l \in \mathcal{V}_l\}_{l=1}^3$ and $\bar{\boldsymbol{\mathcal{X}}} = \bar{\boldsymbol{\mathcal{S}}} \times_1 \bar{\mathbf{V}}_1 \times_2 \bar{\mathbf{V}}_2 \times_3 \bar{\mathbf{V}}_3, \bar{\boldsymbol{\mathcal{S}}} \in \bar{\mathcal{S}}, \{\bar{\mathbf{V}}_i \in \bar{\mathcal{V}}_l\}_{l=1}^3$ respectively. Hence, for any $\tilde{\boldsymbol{\mathcal{X}}}$, there exist $\bar{\boldsymbol{\mathcal{X}}}$ such that the following holds:

$$\begin{aligned} \|\tilde{\boldsymbol{\mathcal{X}}} - \bar{\boldsymbol{\mathcal{X}}}\|_{\mathrm{F}} &= \left\|\tilde{\boldsymbol{\mathcal{S}}} \times \tilde{\mathbf{V}}_1 \times \tilde{\mathbf{V}}_2 \times \tilde{\mathbf{V}}_3 - \bar{\boldsymbol{\mathcal{S}}} \times \bar{\mathbf{V}}_1 \times \bar{\mathbf{V}}_2 \times \bar{\mathbf{V}}_3\right\|_{\mathrm{F}} \\ &= \|\tilde{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3 \pm \bar{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3 \pm \\ &\quad \bar{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \bar{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3 \pm \bar{\boldsymbol{\mathcal{S}}} \times_1 \bar{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3 \\ &\quad - \bar{\boldsymbol{\mathcal{S}}} \times_1 \bar{\mathbf{V}}_1 \times_2 \bar{\mathbf{V}}_2 \times_3 \bar{\mathbf{V}}_3\|_{\mathrm{F}} \\ &\leq \|\tilde{\boldsymbol{\mathcal{S}}} \times \tilde{\mathbf{V}}_1 \times \tilde{\mathbf{V}}_2 \times (\tilde{\mathbf{V}}_3 - \bar{\mathbf{V}}_3)\|_{\mathrm{F}} + \\ &\quad \|\tilde{\boldsymbol{\mathcal{S}}} \times \tilde{\mathbf{V}}_1 \times (\tilde{\mathbf{V}}_2 - \bar{\mathbf{V}}_2) \times \bar{\mathbf{V}}_3\|_{\mathrm{F}} + \\ &\quad \|\tilde{\boldsymbol{\mathcal{S}}} \times (\tilde{\mathbf{V}}_1 - \bar{\mathbf{V}}_1) \times \bar{\mathbf{V}}_2 \times \bar{\mathbf{V}}_3\|_{\mathrm{F}} + \\ &\quad \|(\tilde{\boldsymbol{\mathcal{S}}} - \bar{\boldsymbol{\mathcal{S}}}) \times \bar{\mathbf{V}}_1 \times \bar{\mathbf{V}}_2 \times \bar{\mathbf{V}}_3\|_{\mathrm{F}} \\ &\leq 3\alpha\beta^2 \frac{\varepsilon}{\beta^3+3\alpha\beta^2} + \beta^3 \frac{\varepsilon}{\beta^3+3\alpha\beta^2} = \varepsilon. \end{aligned} \tag{43}$$

Therefore, $\bar{\mathcal{X}}$ is an $\varepsilon$-net of $\mathcal{X}$ with a covering number is upper bounded by

$$\begin{aligned} &\mathsf{N}(\mathcal{X}, \varepsilon) \leq \\ &\left[\frac{3(\beta^3+3\alpha\beta^2)}{\varepsilon}\right]^{\|\boldsymbol{\mathcal{S}}\|_0+\sum_{l=1}^3 N_l I_l} \alpha^{\|\boldsymbol{\mathcal{S}}\|_0}\beta^{\sum_{l=1}^3 N_l I_l} \end{aligned} \tag{44}$$

Next, consider the $T$-Lipschitz continuous activation function $\varsigma(\cdot)$ such that the following holds for any $\tilde{\boldsymbol{\mathcal{X}}}_1, \tilde{\boldsymbol{\mathcal{X}}}_2 \in \mathcal{X}$:

$$\|\varsigma(\tilde{\boldsymbol{\mathcal{X}}}_1) - \varsigma(\tilde{\boldsymbol{\mathcal{X}}}_2)\|_{\mathrm{F}} \leq T\|\tilde{\boldsymbol{\mathcal{X}}}_1 - \tilde{\boldsymbol{\mathcal{X}}}_2\|_{\mathrm{F}} \tag{45}$$

Consider the set $\mathcal{X}_{\text{TAP}} = \{\varsigma(\tilde{\boldsymbol{\mathcal{X}}}) = \varsigma(\tilde{\boldsymbol{\mathcal{S}}} \times_1 \tilde{\mathbf{V}}_1 \times_2 \tilde{\mathbf{V}}_2 \times_3 \tilde{\mathbf{V}}_3) | \tilde{\boldsymbol{\mathcal{S}}} \in \mathcal{S}, \{\tilde{\mathbf{V}}_l \in \mathcal{V}_l\}_{i=1}^3\}$, we find that an $\frac{\varepsilon}{T}$-net of $\mathcal{X}$ is an $\varepsilon$-net of $\mathcal{X}_{\text{TAP}}$. Hence, we have

$$\begin{aligned} &\mathsf{N}(\mathcal{X}_{\text{TAP}}, \varepsilon) \leq \\ &\left[\frac{3T(\beta^3+3\alpha\beta^2)}{\varepsilon}\right]^{\|\boldsymbol{\mathcal{S}}\|_0+\sum_{l=1}^3 N_l I_l} \alpha^{\|\boldsymbol{\mathcal{S}}\|_0}\beta^{\sum_{l=1}^3 N_l I_l}. \end{aligned} \tag{46}$$

This completes the proof.

### *F. Proof of Lemma 3*

*Lemma 5:* [21] Let $\mathcal{Y}_1, \cdots, \mathcal{Y}_w$ be a set of samples taken without replacement from $\{y_1, \cdots, y_n\}$ of mean $\mu$. Denote $a = \min_i y_i$ and $b = \max_i y_i$, then

$$\Pr\left[|\frac{1}{w}\sum_{i=1}^{w}\mathcal{Y}_i - \mu| \geq t\right] \leq 2\exp\big(-\frac{2wt^2}{(1-(w-1)/n)(b-a)^2}\big). \tag{47}$$

■

Consider a set of variables $\mathcal{W}(i_1,i_2,i_3) = (\tilde{\boldsymbol{\mathcal{Y}}}(i_1,i_2,i_3) - \tilde{\boldsymbol{\mathcal{X}}}(i_1,i_2,i_3))^2, \forall (i_1,i_2,i_3) \in \Omega$. Then, the sample mean of $\mathcal{W}(i_1,i_2,i_3)$ is the empirical loss $\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})$ and the actual mean is $\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})$. One can see that $\mathcal{W}(i_1,i_2,i_3) = (\tilde{\boldsymbol{\mathcal{Y}}}(i_1,i_2,i_3) - \tilde{\boldsymbol{\mathcal{X}}}(i_1,i_2,i_3))^2 \leq (|\tilde{\boldsymbol{\mathcal{Y}}}(i_1,i_2,i_3)| + |\tilde{\boldsymbol{\mathcal{X}}}(i_1,i_2,i_3)|)^2 \leq (\nu + v + \alpha\beta^3)^2 = \xi$ with $\nu = \max_{i_1,i_2,i_3} |\boldsymbol{\mathcal{X}}_\natural(i_1,i_2,i_3)|$ and $v = \max_{i_1,i_2,i_3} |\boldsymbol{\mathcal{N}}(i_1,i_2,i_3)|$. Using *Lemma 5*, we have $\Pr\Big[|\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}}) - \text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})| \geq t\Big] \leq 2\exp\big(-\frac{2|\Omega|t^2}{(1-(|\Omega|-1)/I_1I_2I_3)\xi^2}\big)$. Denote an $\varepsilon$-net of $\mathcal{X}_{\text{TAP}}$ as $\bar{\mathcal{X}}_{\text{TAP}}$. Using the union bound on all $\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}_{\text{TAP}}$ yields $\Pr\Big[\sup_{\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}_{\text{TAP}}} |\text{loss}_1(\bar{\boldsymbol{\mathcal{X}}}) - \text{loss}_2(\bar{\boldsymbol{\mathcal{X}}})| \geq t\Big] \leq 2|\bar{\mathcal{X}}_{\text{TAP}}| \exp\big(-\frac{2|\Omega|t^2}{(1-(|\Omega|-1)/I_1I_2I_3)\xi^2}\big)$. Equivalently, with probability at least $1-\delta$, the following holds:

$$\begin{aligned} &\sup_{\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}_{\text{TAP}}} |\text{loss}_1(\bar{\boldsymbol{\mathcal{X}}}) - \text{loss}_2(\bar{\boldsymbol{\mathcal{X}}})| \leq \\ &\sqrt{\frac{\xi^2 \log(2|\bar{\mathcal{X}}_{\text{TAP}}|/\delta)}{2}\big(\frac{1}{|\Omega|} + \frac{1}{|\Omega|I_1I_2I_3} - \frac{1}{I_1I_2I_3}\big)}. \end{aligned} \tag{48}$$

Let $u(\Omega) \triangleq \sup_{\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}_{\text{TAP}}} |\text{loss}_1(\bar{\boldsymbol{\mathcal{X}}}) - \text{loss}_2(\bar{\boldsymbol{\mathcal{X}}})|$. We have $\sup_{\bar{\boldsymbol{\mathcal{X}}} \in \bar{\mathcal{X}}_{\text{TAP}}} |\sqrt{\text{loss}_1(\bar{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_2(\bar{\boldsymbol{\mathcal{X}}})}| \leq \sqrt{u(\Omega)}$.

Note that

$$\begin{aligned} &|\sqrt{\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_2(\bar{\boldsymbol{\mathcal{X}}})}| = \\ &\frac{1}{\sqrt{I_1I_2I_3}}|\|\tilde{\boldsymbol{\mathcal{Y}}} - \tilde{\boldsymbol{\mathcal{X}}}\|_{\mathrm{F}} - \|\tilde{\boldsymbol{\mathcal{Y}}} - \bar{\boldsymbol{\mathcal{X}}}\|_{\mathrm{F}}| \\ &\leq \frac{1}{\sqrt{I_1I_2I_3}}\|\tilde{\boldsymbol{\mathcal{Y}}} - \tilde{\boldsymbol{\mathcal{X}}} - \tilde{\boldsymbol{\mathcal{Y}}} + \bar{\boldsymbol{\mathcal{X}}}\|_{\mathrm{F}} = \frac{1}{\sqrt{I_1I_2I_3}}\|\bar{\boldsymbol{\mathcal{X}}} - \tilde{\boldsymbol{\mathcal{X}}}\|_{\mathrm{F}} \\ &\leq \frac{\varepsilon}{\sqrt{I_1I_2I_3}}. \end{aligned} \tag{49}$$

Similarly, we have $|\sqrt{\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_1(\bar{\boldsymbol{\mathcal{X}}})}| \leq \frac{\varepsilon}{\sqrt{|\Omega|}}$. Then we can show that the following holds:

$$\begin{aligned} &\sup_{\tilde{\boldsymbol{\mathcal{X}}} \in \mathcal{X}_{\text{TAP}}} |\sqrt{\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})}| \leq \\ &\frac{\varepsilon}{\sqrt{I_1I_2I_3}} + \sqrt{u(\Omega)} + \frac{\varepsilon}{\sqrt{|\Omega|}}. \end{aligned} \tag{50}$$

Therefore, the following holds using the definition of $u(\Omega)$ with probability at least $1-\delta$:

$$\begin{aligned} &\sup_{\tilde{\boldsymbol{\mathcal{X}}} \in \mathcal{X}_{\text{TAP}}} |\sqrt{\text{loss}_1(\tilde{\boldsymbol{\mathcal{X}}})} - \sqrt{\text{loss}_2(\tilde{\boldsymbol{\mathcal{X}}})}| \leq \\ &\frac{2\varepsilon}{\sqrt{|\Omega|}} + \big(\frac{\xi^2 w}{2}\log(\frac{2}{\delta}|\bar{\mathcal{X}}_{\text{TAP}}|)\big)^{\frac{1}{4}}. \end{aligned} \tag{51}$$

This completes the proof.

### *G. Proof of Theorem 1*

Denote the empirical loss and actual loss associated with an optimal solution $\boldsymbol{\mathcal{X}}^*$ as:

$$\begin{aligned} \sqrt{\text{loss}_1(\boldsymbol{\mathcal{X}}^*)} &= \frac{1}{\sqrt{|\Omega|}}\|\boldsymbol{\mathcal{O}} * (\tilde{\boldsymbol{\mathcal{Y}}} - \boldsymbol{\mathcal{X}}^*)\|_{\mathrm{F}}, \\ \sqrt{\text{loss}_2(\boldsymbol{\mathcal{X}}^*)} &= \frac{1}{\sqrt{I_1I_2I_3}}\|\tilde{\boldsymbol{\mathcal{Y}}} - \boldsymbol{\mathcal{X}}^*\|_{\mathrm{F}}. \end{aligned} \tag{52}$$

Then the following chain of inequality holds:

$$\begin{aligned}
&\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{X}}^*-\boldsymbol{\mathcal{X}}_\natural\|_{\mathrm{F}}=\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{X}}^*-\tilde{\boldsymbol{\mathcal{Y}}}+\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&\leq \frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{X}}^*-\tilde{\boldsymbol{\mathcal{Y}}}\|_{\mathrm{F}}+\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&\leq \frac{1}{\sqrt{|\Omega|}}\|\boldsymbol{\mathcal{O}}*(\tilde{\boldsymbol{\mathcal{Y}}}-\boldsymbol{\mathcal{X}}^*)\|_{\mathrm{F}}+\mathrm{Gap}^*(\Omega)+\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&\leq \frac{1}{\sqrt{|\Omega|}}\|\boldsymbol{\mathcal{O}}*(\tilde{\boldsymbol{\mathcal{Y}}}-\tilde{\boldsymbol{\mathcal{X}}}^*)\|_{\mathrm{F}}+\mathrm{Gap}^*(\Omega)+\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&\leq \frac{\|\boldsymbol{\mathcal{O}}*(\tilde{\boldsymbol{\mathcal{X}}}^*-\boldsymbol{\mathcal{X}}_\natural)\|_{\mathrm{F}}+\|\boldsymbol{\mathcal{O}}*(\boldsymbol{\mathcal{X}}_\natural-\tilde{\boldsymbol{\mathcal{Y}}})\|_{\mathrm{F}}}{\sqrt{|\Omega|}}\\
&+\mathrm{Gap}^*(\Omega)+\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&\leq \frac{1}{\sqrt{|\Omega|}}\left\|\tilde{\boldsymbol{\mathcal{X}}}^*-\boldsymbol{\mathcal{X}}_\natural\right\|_{\mathrm{F}}+\frac{1}{\sqrt{|\Omega|}}\|\boldsymbol{\mathcal{O}}*\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\\
&+\mathrm{Gap}^*(\Omega)+\frac{1}{\sqrt{I_1I_2I_3}}\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}\,.
\end{aligned}\tag{53}$$

This completes the proof.

### *H. Comparisons of recoverability analysis*

The key differences between the proposed recoverability analysis and prior work lie in:

- The use of a nonlinear Tucker model with an activation function.
- The presence of a core tensor with a sparsity pattern.
- The use of non-orthogonal factor matrices.

These differences primarily influence the generalization error (i.e., $\mathrm{Gap}^*(\Omega)$) in the reconstruction error bound. In [1], the generalization error is given by:

$$\mathrm{Gap}^*(\Omega)\leq\frac{2cR}{\sqrt{|\Omega|}}+\Big(\frac{\xi^2\omega}{2}\log\big(\frac{2\mathsf{N}(\mathcal{X}_{\mathbf{g}R,\boldsymbol{\theta}_d},cR)}{\delta}\big)\Big),\tag{54}$$

with the covering number

$$\mathsf{N}(\mathcal{X}_{\mathbf{g}R,\boldsymbol{\theta}_d},\varepsilon)\leq\big(\frac{3R(\alpha+\beta)}{\varepsilon}\big)^{R(K+D)}\alpha^{RK}(Pq)^{RD}.\tag{55}$$

In contrast, the proposed model yields:

$$\mathrm{Gap}^*(\Omega)\leq\frac{2\varepsilon}{\sqrt{|\Omega|}}+\Big(\frac{\xi^2\omega}{2}\log\big(\frac{2\mathsf{N}(\mathcal{X}_{\mathrm{TAP}},\varepsilon)}{\delta}\big)\Big),\tag{56}$$

$$\mathsf{N}(\mathcal{X}_{\mathrm{TAP}},\varepsilon)\leq\left[\frac{3T(\beta^3+3\alpha\beta^2)}{\varepsilon}\right]^{\|\boldsymbol{\mathcal{S}}\|_0+\sum_{l=1}^3N_lI_l}\alpha^{\|\boldsymbol{\mathcal{S}}\|_0}\beta^{\sum_{l=1}^3N_lI_l}.\tag{57}$$

Comparing (53)-(56), we observe that the key difference in the generalization error arises from the covering numbers $\mathsf{N}(\mathcal{X}_{\mathbf{g}R,\boldsymbol{\theta}_d},\varepsilon)$ and $\mathsf{N}(\mathcal{X}_{\mathrm{TAP}},\varepsilon)$. In the SOTA model [1], the covering number is determined by pre-selected hyperparameters, such as the assumed source number, $R$ and the latent space dimensionality, $D$. Choosing overly large values increases the bound, thereby raising the risk of higher generalization error. In contrast, the proposed model's generalization error depends on the sparsity of the core tensor, $\|\boldsymbol{\mathcal{S}}\|_0$, which represents the number of non-zero elements in the core tensor. Instead of being pre-fixed, $\|\boldsymbol{\mathcal{S}}\|_0$ is learned in a self-supervised manner during training. This highlights the importance of learning the core tensor sparsity and aligns with the experimental results.

Additionally, these differences also influence the representation error in the reconstruction error bound:

$$\|\tilde{\boldsymbol{\mathcal{X}}}^*-\boldsymbol{\mathcal{X}}_\natural\|_{\mathrm{F}},\tag{58}$$

where $\tilde{\boldsymbol{\mathcal{X}}}^*$ represents the best reconstruction from the solution set (see Definition 1) and $\boldsymbol{\mathcal{X}}_\natural$ denotes the ground truth field. In [1], the optimal reconstruction is given by:

$$\tilde{\boldsymbol{\mathcal{X}}}^*=\arg\min_{\tilde{\boldsymbol{\mathcal{X}}}\in\mathcal{X}_{\mathbf{g}R,\boldsymbol{\theta}_d}}\|\tilde{\boldsymbol{\mathcal{X}}}-\boldsymbol{\mathcal{X}}_\natural\|_{\mathrm{F}},\tag{59}$$

while in the proposed model:

$$\tilde{\boldsymbol{\mathcal{X}}}^*=\arg\min_{\tilde{\boldsymbol{\mathcal{X}}}\in\mathcal{X}_{\mathrm{TAP}}}\|\tilde{\boldsymbol{\mathcal{X}}}-\boldsymbol{\mathcal{X}}_\natural\|_{\mathrm{F}}.\tag{60}$$

For the model in [1], the representation error is determined by the pretrained deep model with parameters $\boldsymbol{\theta}_d$. If the test data distribution differs significantly from the training data, the prior information embedded in the pretrained model may misleadingly constrain the solution set, resulting in increased representation error. In contrast, the proposed model does not rely on pretrained models, and its high expressiveness – since the Tucker model is a universal approximator – enables a lower representation error.

In summary, by comparing the two primary sources that contribute to the reconstruction error – the generalization error in (53)-(56) and the representation error in (57)-(59) – we conclude that the proposed model has the potential to achieve a lower reconstruction error compared to [1], particularly under distribution shifts. This advantage stems from the proposed model's ability to adaptively adjust its complexity by learning the sparsity of the core tensor, unlike the fixed model complexity in [1], which depends on a pretrained deep model.

### *I. Implementation Details of Ocean Sound Speed Field Reconstruction*

a. Tucker-ALS: The core tensor size is set to (5, 5, 5), and the size of the three factor matrices is set to (20, 5)
b. LRTC: The hyperparameters $\alpha$ and $\beta$ are both set to 1.
c. TNN: We choose the TNN with layer dimensionalities of (5, 5, 5), (10, 10, 10), and (20, 20, 20), which have been shown to be the most competent in the SSF reconstruction task [10]. The learning rate is set to $4e^{-3}$. The number of parameters is 875.
d. TAP: The cube size is set to (4, 4, 4) and the stride is set to (2, 2, 2). Consequently, $N=(\frac{20-4}{2}+1)^3=729$ cubes can be extracted, and the reshape size $(N_1,N_2,N_3)$ is (81, 81, 81). The learning rate is set to $4e^{-3}$. The number of parameters is 12K.
e. MHTAP: The cube size is set to (5, 5, 5) and the stride is set to (3, 3, 3). This configuration allows us to extract a total of $N=(\frac{20-5}{3}+1)^3=216$ cubes, with $M=64$. We choose the head number $h$ to be 8 ($h_1=h_2=h_3=2$) and set the reshape size $(h_1N_1,h_2N_2,h_3N_3)$ to $(72,72,72)$, which approximates the reshape size used in TAP. The learning rate is set to $4e^{-3}$ and the number of parameters is 132K.

### J. Implementation Details of Radio Map Reconstruction

a. TNN: We choose the TNN with dimensionality of each layer being (10,10,10), (25,25,32), (51,51,64). The learning rate is set to $4e^{-3}$. The number of parameters is 6.5K.
b. TAP: The cube size is set to (6,6,8), the stride is set to (5,5,8). Therefore, $N = (\frac{51-6}{5}+1)^2(\frac{64-8}{8}+1) = 800$ cubes can be extracted and $M = 288$. The size of core tensor $(N_1, N_2, N_3)$ = (100,100,64). The learning rate is set to $4e^{-3}$. The number of parameters is 180K.
c. MHTAP: The cube size is set to (9,9,8), the stride is set to (7,7,8). Therefore, $N = (\frac{51-9}{7}+1)^2(\frac{64-8}{8}+1) = 392$ cubes can be extracted and $M = 288$. We choose the head number $h$ = 4 and we set the size of core tensor $(h_1N_1, h_2N_2, h_3N_3) = (98, 98, 64)$ approximating that of TAP. The learning rate is set to $4e^{-3}$. The number of parameters is 1.5M.

### K. Enhanced Loss Function with TV Regularization for Untrained Methods

To further enhance the competitiveness of those untrained models, we incorporate the commonly used total variation (TV) regularization [18] into the loss function to enhance the performance. The second order TV regularization for 3D physical fields is defined as:

$$\begin{aligned}\|\boldsymbol{\mathcal{X}}\|_{\text{TV}} =& \sum_k \sum_{i,j} \sqrt{D_x^2 + D_y^2 + 2D_{x,y}^2} \\ D_x =& (\boldsymbol{\mathcal{X}}(i_1+1, i_2, i_3) - 2\boldsymbol{\mathcal{X}}(i_1, i_2, i_3) + \boldsymbol{\mathcal{X}}(i_1-1, i_2, i_3))^2 \\ D_y =& (\boldsymbol{\mathcal{X}}(i_1, i_2+1, i_3) - 2\boldsymbol{\mathcal{X}}(i_1, i_2, i_3) + \boldsymbol{\mathcal{X}}(i_1, i_2-1, i_3))^2 \\ D_{x,y} =& \frac{1}{4}(\boldsymbol{\mathcal{X}}(i_1+1, i_2+1, i_3) - \boldsymbol{\mathcal{X}}(i_1+1, i_2-1, i_3) - \\ & \boldsymbol{\mathcal{X}}(i_1-1, i_2+1, i_3) + \boldsymbol{\mathcal{X}}(i_1-1, i_2-1, i_3))^2 \end{aligned} \tag{61}$$

Then the loss function of those untrained methods becomes:

$$\min_{\boldsymbol{\mathcal{X}}} \|\boldsymbol{\mathcal{Y}} - \boldsymbol{\mathcal{O}} * \boldsymbol{\mathcal{X}}\|_{\text{F}}^2 + \gamma \|\boldsymbol{\mathcal{X}}\|_{\text{TV}}, \tag{62}$$

where $\boldsymbol{\mathcal{X}}$ is the reconstructed 3D physical field and $\gamma$ is the trade-off parameter.

### L. Generic guidelines for the choice of hyperparameters

Here, we provide guidelines for selecting dimensions $(R_1, R_2, R_3)$, $(K_1, K_2, K_3)$ and $(S_1, S_2, S_3)$.

Our key idea is to first construct an over-complete tensor Tucker model with a Tucker core larger than the physical field to ensure expressiveness (i.e., $R_l > I_l, \forall l$). We then employ an attention mechanism to adaptively learn the sparsity pattern (model complexity). To achieve this, we recommend choosing $K_l \leq \lfloor \frac{I_l}{3} \rfloor, \forall l$, where $K_l$ and $I_l$ denote the window size and the original field size for mode $l$, respectively. The stride size controls the number of extracted cubes, and we suggest choosing $S_l \leq \lfloor \frac{K_l}{2} \rfloor, \forall l$.

Once $(K_1, K_2, K_3)$ and $(S_1, S_2, S_3)$ are selected, the size of the core tensor, $(R_1, R_2, R_3)$, can be determined by the proposed tensorization steps in Sec. III-B, ensuring that $R_l > I_l, \forall l$ holds. Additionally, the empirical results indicate that the corresponding final reconstruction remains insensitive to different hyperparameter choices, provided the specified conditions (i.e., $R_l > I_l, \forall l$) are met.

### M. Impact of the sparsity level and pattern of the observations

Since our goal is to develop a "complexity-adaptive" model for general physical field reconstructions, our derivation of the reconstruction error bound (Theorem 1) imposes minimal restrictions on the sparsity levels ($\rho$) and the patterns in the original observation set, $\boldsymbol{\mathcal{Y}}$. Specifically, the sparsity level $\rho$ primarily influences the error bound through the term $|\Omega|$ in Eq. (26) and Eq. (27). A higher $\rho$ (i.e., larger $|\Omega|$) reduces the estimation error bound, as shown in Eq. (27).

Although the sparsity pattern of $\boldsymbol{\mathcal{Y}}$ is not explicitly included in the recoverability analysis, we conduct experiments with different sampling patterns to evaluate the reconstruction performance under practical scenarios. In these experiments, we use random sampling for SSF and fiber-wise sampling for the radio map. Empirically, the proposed model performs well with both sampling strategies, demonstrating its robustness.

While our derived error bound does not explicitly quantify the impact of the sparsity pattern of $\boldsymbol{\mathcal{Y}}$, it does influence reconstruction results to some extent. This presents an interesting direction for future work – to refine the error bound by incorporating sparsity patterns. In particular, adversarial sampling scenarios (e.g., block-missing observations) can degrade the performance of the proposed model by creating large variations in the observation counts across different cubes, leading to misestimated similarity scores.

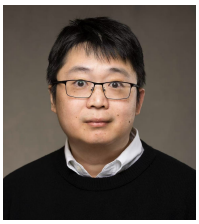
### N. Trade-off between computational cost and reconstruction accuracy

The proposed method exhibits a trade-off between computational cost and reconstruction accuracy when scaling to larger datasets or more complex fields. FieldFormer is particularly well-suited for handling spatial-temporal continuous fields, which naturally result in a sparse attention map. This allows the model to maintain a relatively low model complexity and computational cost, while achieving state-of-the-art reconstruction accuracy. However, when applied to more complex fields, the model adapts accordingly and the percentage of the non-zero elements in the Tucker core tensor could increase, activating more learnable parameters. As a result, computational costs rise to ensure accurate reconstruction. Similarly, when scaling to larger datasets, more cubes are extracted to compute similarity scores, expanding the sparse attention map. This requires additional parameters to characterize the entire model, inevitably increasing the computational cost to maintain reconstruction accuracy. Nevertheless, the concern of computational overhead can be mitigated by implementing linear attention mechanisms [51], which provide a more efficient alternative while preserving accuracy.

**Panqi Chen** received the BSc degree from Xidian University, Xi'an, China, in 2022. He is currently working toward the PhD degree with the College of Information Science and Electronic engineering, Zhejiang University, Hangzhou, China. His research interests include signal processing and machine learning.

**Siyuan Li** received the BSc degree in Electronic Science and Technology from Zhejiang University, Hangzhou, China. He is currently working toward the PhD degree with the College of Information Science and Electronic Engineering, Zhejiang University. His research interests include signal processing and machine learning.

**Lei Cheng** (Member, IEEE) is currently ZJU Young Professor with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. He received the B.Eng. degree from Zhejiang University in 2013, and the Ph.D. degree from The University of Hong Kong in 2018. He was a Research Scientist in Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen, from 2018 to 2021. He is the author of the book "Bayesian Tensor Decomposition for Signal Processing and Machine Learning", Springer, 2023. He was a Tutorial Speaker in IEEE ICASSP 2023, Invited Speaker in ASA 2024 and IEEE COA 2014. He is now Associate Editor for Elsevier Signal Processing and Young Editor for Acta Acustica. His research interests are in Bayesian machine learning for tensor data analytics and interpretable machine learning for information systems..



**Xiao Fu** (Senior Member, IEEE) received the the Ph.D. degree in Electronic Engineering from The Chinese University of Hong Kong (CUHK), Shatin, N.T., Hong Kong, in 2014. He was a Postdoctoral Associate with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, from 2014 to 2017. Since 2017, he has been with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA, where he is now an Associate Professor. His research interests include the broad area of signal processing and machine learning. Dr. Fu received the 2022 IEEE Signal Processing Society (SPS) Best Paper Award and the 2022 IEEE SPS Donald G. Fink Overview Paper Award. He is a recipient of the National Science Foundation (NSF) CAREER Award in 2022, the College of Engineering Engelbrecht Early Career Award in 2023, and the University Promising Scholar Award in 2024. He serves as a member of the IEEE SPS Sensor Array and Multichannel Technical Committee (SAM-TC) and the Signal Processing Theory and Methods Technical Committee (SPTM-TC). He is currently a Suject Editor of SIGNAL PROCESSING and an Associate Editor of IEEE TRANSACTIONS ON SIGNAL PROCESSING.

**Yik-Chung Wu** (Senior Member, IEEE) received the B.Eng. (EEE) and M.Phil. degrees from The University of Hong Kong (HKU) in 1998 and 2001, respectively, and the Ph.D. degree from Texas A&M University, College Station, in 2005. From 2005 to 2006, he was with Thomson Corporate Research, Princeton, NJ, USA, as a Member of Technical Staff. Since 2006, he has been with HKU, where he is currently as an Associate Professor. He was a Visiting Scholar at Princeton University in Summers of 2015 and 2017. His research interests include signal processing, machine learning, and communication systems. He served as an Editor for IEEE COMMUNICATIONS LETTERS and IEEE TRANSACTIONS ON COMMUNICATIONS. He is currently an Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING and Journal of Communications and Networks.

**Sergios Theodoridis** (Life Fellow, IEEE) is Professor Emeritus with the Department of Informatics and Telecommunications of the National and Kapodistrian University of Athens, Greece, and Director of Education of the HERON European center of excellence for AI and Robotics, Greece.

He is the author of the book "Machine Learning: From the Classics to Deep Networks, Transformers and Diffusion Models", Academic Press, 3rd Ed., 2025, the co-author of the best-selling book "Pattern Recognition", Academic Press, 4th Ed. 2009, the co-author of the book "Introduction to Pattern Recognition: A MATLAB Approach", Academic Press, 2010.

He is the co-author of seven papers that have received Best Paper Awards including the 2014 IEEE Signal Processing Magazine Best Paper Award and the 2009 IEEE Computational Intelligence Society Transactions on Neural Networks Outstanding Paper Award.

He has received an honorary doctorate degree (D.Sc) from the University of Edinburgh, UK, in 2023. He is the recipient of the 2021 IEEE Signal Processing Society (SPS) Norbert Wiener Award, the 2017 EURASIP Athanasios Papoulis Award, the 2014 IEEE SPS Carl Friedrich Gauss Education Award and the 2014 EURASIP Meritorious Service Award. He has served as a Distinguished Lecturer for the IEEE SP as well as the Circuits and Systems societies.

He has served as Vice President IEEE Signal Processing Society, as President of the European Association for Signal Processing (EURASIP), as a member of the Board of Governors for the IEEE Circuits and Systems (CAS) Society, and as the chair of the IEEE SPS awards board.

He is Fellow of IET, a Corresponding Fellow of the Royal Society of Edinburgh (RSE), a Fellow of EURASIP and a Life Fellow of IEEE.