Eurographics Symposium on Rendering (2025)
B. Wang and A. Wilkie (Editors)

Iterative Bayesian Nonparametric CP Decomposition for Hyperspectral Image Denoising

Wei Liu[1], Kaiwen Jiang[1], Jinzhi Lai[2] and Xuesong Zhang[†][1]

[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]Industrial Bank Co., Ltd, Shanghai, China

Abstract

Hyperspectral image (HSI) denoising relies on exploiting the multiway correlations hidden in the clean signals to discriminate between the randomness of measurement noise. This paper proposes a self-supervised model that has a three-layer algorithmic hierarchy to iteratively quest for the tensor decomposition based representation of the underlying HSI. The outer layer takes advantage of the non-local similarity of HSI via a simple but effective k-means++ algorithm to explore the patch-level correlation and yields clusters of patches with similar tensor ranks. The middle and inner layers consist of a Bayesian Nonparametric tensor decomposition framework. The middle one employs a multiplicative Gamma process prior for the low rank tensor decomposition, and a Gaussian-Wishart prior for a more flexible exploration of the correlations among the latent factor matrices. The inner layer is responsible for the finer regression of the residual multiway correlations leaked from the upper two layers. Our scheme also provides a principled and automatic solution to several practical HSI denoising factors, such as the noise level, the model complexity and the regularization weights. Extensive experiments validate that our method outperforms state-of-the-art methods on a series of HSI denoising metrics.

CCS Concepts

• ***Computing methodologies*** → ***Image processing;***

---

1. Introduction

Hyperspectral image (HSI) consisting of images of multiple wavelengths of the same scene, has wide applications, such as land cover classification [AOM14], remote sensing [OBP*23], agriculture [MLMLC*17], computer vision tasks related to HSI, including unmixing [FLW*22], super-resolution [WWB*23], object tracking and so on. During the simultaneous acquisition of different spectral images, the existing hyperspectral imaging methods often suffer from the problem of low signal-to-noise ratio, due to shorter exposure time and narrower bandwidth. Since hyperspectral images are often contaminated by serious noise, it is essential to develop software based denoising techniques to support HSI related applications.

There are mainly two categories of approaches for HSI denoising, including the 2D extended approach and the tensor-based approach. 2D image denoising has been researched for decades and a large number of researches have been proposed on this problem, such as K-SVD [AEB06] and BM3D [DFKE07]. When dealing with HSIs, traditional denoising methods usually tile the spectral channels in the spatial dimensions, which neglect the intrinsic 3D spectral-spatial structure, and thus lead to performance degradations. Based on BM3D, BM4D [MKEF13] takes the small local patches of the image into consideration. By exploiting the 3D nonlocal similarity of HSIs to remove noise in similar HSI 3D cubes collaboratively, BM4D achieves the state-of-the-art performance in 2D image denoising approaches.

3D HSI denoising approaches are based on two well known tensor decomposition methods, CP (CANDECOMP/PARAFAC) decomposition and Tucker decomposition. Renard [RBBT08] presented a low-rank tensor approximation (LRTA) method by employing the Tucker decomposition to obtain the low-rank approximation of the input HSI. Liu [LBF12] designed the PARAFAC method by utilizing the CP decomposition. The advantage of the both methods is that they take the correlation over different bands in HSI into consideration, and tried to eliminate the spectral redundancy of HSI. However, they have not utilized the nonlocal similarity property of HSI, and their performance may be sensitive to noise variance and types. Yi Peng [PMX*14] proposed the method of Nonlocal Tensor Dictionary Learning by Tucker decomposition. Furthermore, Gong [GCC20], Dian [DLF*20], Li [LCG*24],Wang [WNZ*22] and Xu [XWCW19] proposed solutions based on nonlocal similarity and tensor decompostion, however, complex op-

---

[†] Corresponding author: xuesong_zhang@bupt.edu.cn. This work was supported by the National Natural Science Foundation of China (61871055).

DOI: 10.2312/sr.20251193

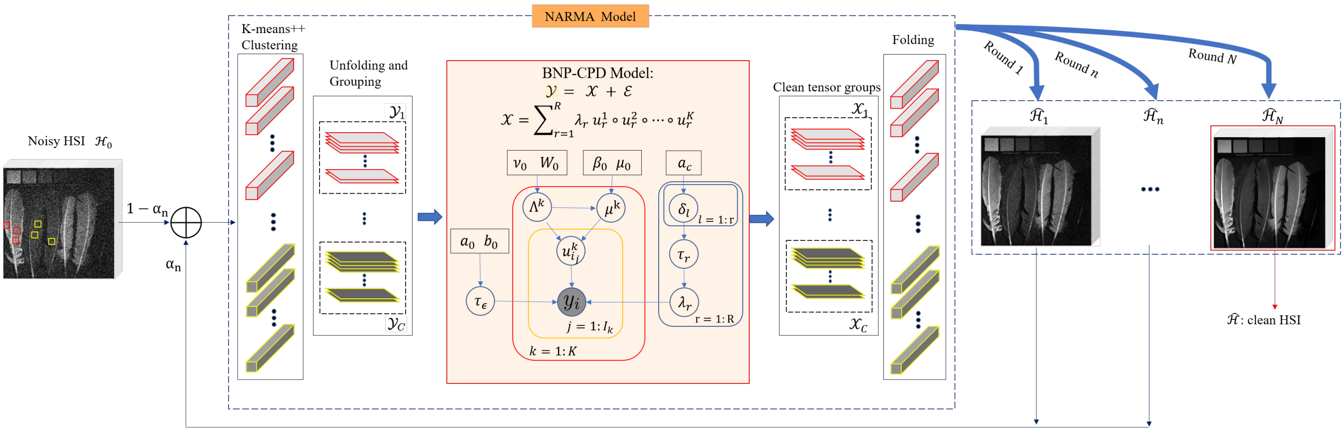


Figure 1: *The proposed IBNP-CPD framework consists of an outer NARMA model and an inner BNP-CPD model. NARMA performs HSI patch classification to exploit the non-local similarity and recursively refines the reconstruction. BNP-CPD employs MGP and Gaussian-Wishart hyperpriors to automatically and flexibly quest for the multiway correlation hidden in the tensors.*

timization algorithms are required, and some parameters must be predetermined manually.

Proper parameter selection is critical for HSI denoising algorithms to achieve good performance. Gong [GCC20] assumed that the noise variance was known a priori while the noise variance may be unknown in many applications; Yi Peng [PMX*14] briefly calculated the rank parameters by applying the *Akaike Information Criterion/Minimum Description Length* (AIC/MDL) method which were important to the quality of dictionary learning but not easy to set them to appropriate values. Zhou [ZCP*12] proposed a Bayesian nonparametric dictionary learning algorithm based on truncated beta Bernoulli process, which solved the problem of parameter selection in dictionary learning effectively. Ju [JSG*16] utilized it to tensor dimensions and performed well in video reconstruction. Chen [CCZ*18] proposed a Bayesian tensor decomposition approach with Gaussian-Wishart prior for spatiotemporal traffic data imputation. A further step towards Bayesian models with automatic parameter selection was taken by Ye [YWJ*23], developing a model that can infer the latent CP rank of HSI tensor adaptively without tuning parameters.

Witnessing the success of deep learning (DL) in a wide range of applications, some DL-based denoising methods for HSIs have been proposed with DL. Particularly, deep residual network was presented for image classification and showed significant performance [MAU*24]. Residual network (ResNet) is an extension of the convolutional neural network. It has jump connections, which facilitates the propagation of gradients, and has strong robustness and deep architecture [LMF19]. The biggest difference of the ResNet is that it uses a residual block and adds short connections in the convolutional layer, which can obtain more sufficient features equivalently [ZLM*17]. Inspired by ResNet, we attempt to capture more structural details of HSI by residual processing.

Among self-supervised techniques, one notable category consists of methods based on untrained neural networks [SH19]. These methods, which hold significant potential for image restoration, benefit from the powerful representational abilities of neural networks while eliminating the need for additional training data. Ulyanov et al. [UVL18]were the first to extend untrained neural networks from RGB images to hyperspectral images (HSIs), proposing a self-supervised HSI restoration framework. Later, Luo et al. [LZJ*21] introduced a spatio-spectral constrained untrained neural network. A key limitation of these methods is their susceptibility to error accumulation, a common issue in iterative procedures, making them especially sensitive to heavily degraded HSIs with substantial noise. Although Deep Learning-Based DDS2M [MZZT23] is also a multi-step generation process, it does not suffer from such accumulated errors. Similarly, our method also avoids this accumulation of errors.

In summary, we propose the Bayesian Nonparametric tensor decomposition for the task of HSI denoising and the main contributions are as follows:

- We propose a Bayesian nonparametric CP decomposition (BNP-CPD) method that can jointly infer the tensor rank, noise variance and latent factor matrices in a principled way, which competes with current deep learning based self-supervised HSI denoising methods.
- We introduce Gaussian-Wishart priors over the tensor decomposition (TD) factor matrices, which allows for more flexible exploration of the correlations among factors. Extensive numerical experiments validate that oblique factors do benefit the spectral fidelity.
- We further propose the Iterative Bayesian Nonparametric CP Decomposition (IBNP-CPD) framework to exploit the nonlocal self-similarity across space and the global correlation across spectrum, which gradually captures the intrinsic multiway correlations hidden in the HSI in a recursive restoration manner.

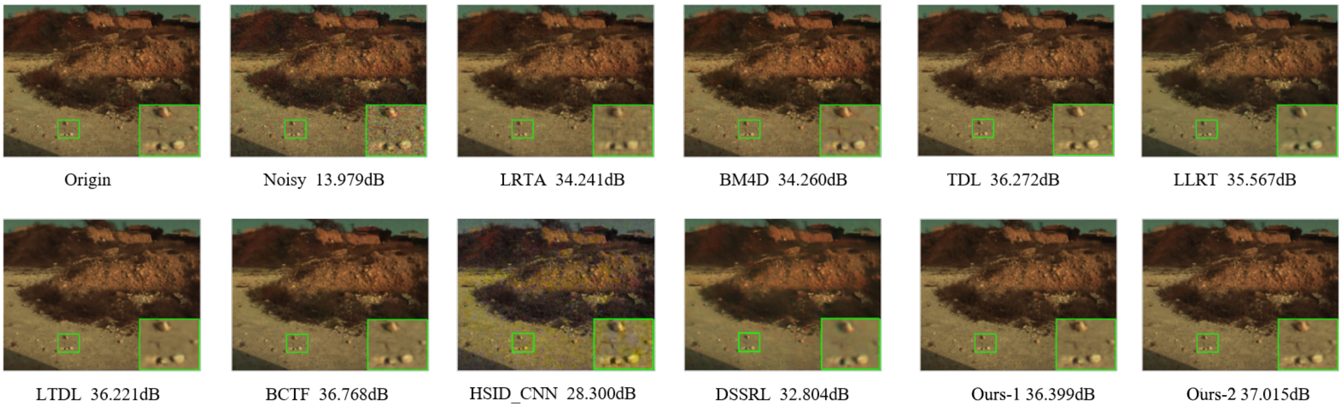

Origin
Noisy 13.979dB
LRTA 34.241dB
BM4D 34.260dB
TDL 36.272dB
LLRT 35.567dB
LTDL 36.221dB
BCTF 36.768dB
HSID_CNN 28.300dB
DSSRL 32.804dB
Ours-1 36.399dB
Ours-2 37.015dB

Figure 2: *The denoising results of all methods above on eve_0331-1647 in ICVL with* $\sigma = 0.2$.

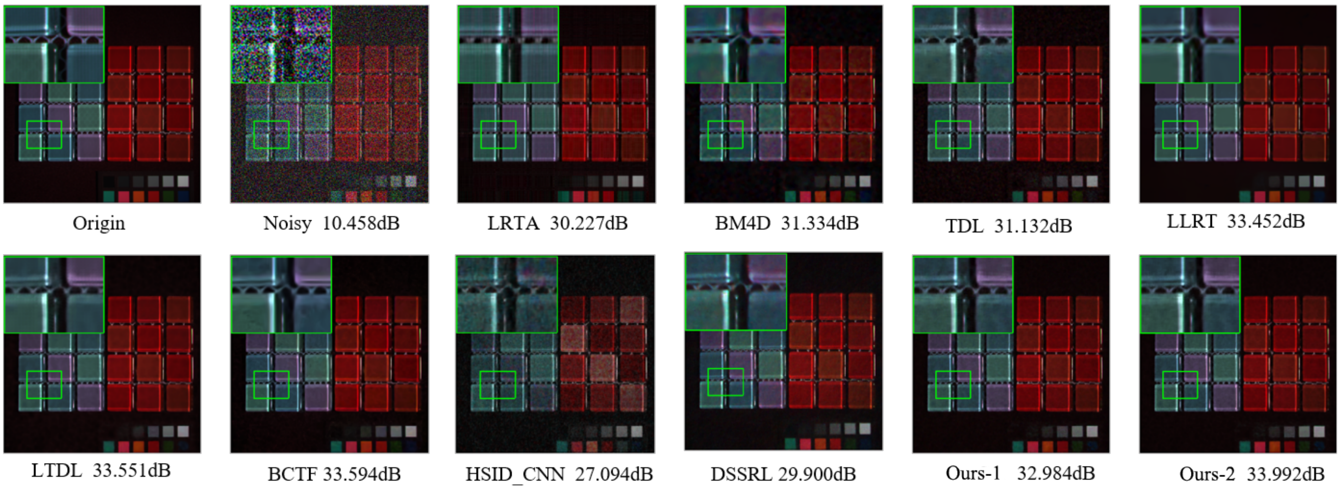

Origin
Noisy 10.458dB
LRTA 30.227dB
BM4D 31.334dB
TDL 31.132dB
LLRT 33.452dB
LTDL 33.551dB
BCTF 33.594dB
HSID_CNN 27.094dB
DSSRL 29.900dB
Ours-1 32.984dB
Ours-2 33.992dB

Figure 3: *The denoising results of all methods on glass_tiles in CAVE with* $\sigma = 0.3$.

2. Notations and Preliminaries

Throughout the paper, scalars, vectors, matrices and tensors are denoted by italic letters, bold lower case letters, upper case letters and calligraphic upper case letters, respectively.

2.1. Tensor Notations

A tensor of order K can be denoted as $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots I_K}$, whose element $(i_1, i_2, \ldots, i_K)$ of tensor $\mathcal{A}$ is denoted as $\mathcal{A}_{i_1 i_2 \ldots i_K}$. The tensor $\mathcal{A}$ can be unfolded into a matrix $A^k \in \mathbb{R}^{I_k \times (I_1 \times \ldots I_{k-1} \times I_{k+1} \cdots \times I_K)}$ along the k-th mode, which is called the mode-k $(k = 1, \ldots, K)$ flattening matrix of the tensor $\mathcal{A}$.

2.2. CP Decomposition

The CP [KB09] decomposition decomposes a tensor into a sum of rank-one tensors. An order K tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_K}$ can be represented by the CP decomposition as

$$\hat{\mathcal{Y}} = \sum_{r=1}^{R} \lambda_r \boldsymbol{u}_r^1 \circ \boldsymbol{u}_r^2 \circ \cdots \circ \boldsymbol{u}_r^K \tag{1}$$

where $\boldsymbol{u}_r^k \in \mathbb{R}^{I_k}, k \in \{1, 2, \ldots, K\}$ and $\circ$ denotes the vector outer product. λ_R is recognized as the tensor rank which controls the model complexity.

$U^k = [\boldsymbol{u}_1^k, \boldsymbol{u}_2^k, \ldots, \boldsymbol{u}_R^k]$the $I_k \times R$ factor matrix of the k-th mode of the tensor $\mathcal{Y}$, where $u_{i_k r}^k$ is the element at (i_k, r) in the factor matrix U^k. Denote $\boldsymbol{i} = (i_1, i_2, \ldots, i_K)$, then for each element $y_{\boldsymbol{i}}$, Eq. (1) can be rewritten as

$$\hat{y}_{\boldsymbol{i}} = \sum_{r=1}^{R} \lambda_r \prod_{k=1}^{K} \boldsymbol{u}_{i_k r}^k \tag{2}$$

Eq. (2) tells the fact that the element $y_{\boldsymbol{i}}$ is determined by the set of row vectors $u_{i_k}^k$, which is the i_k-th row of the k-th factor matrix U_k, for $k = 1, \ldots, K$.

Proceedings published by Eurographics - The European Association for Computer Graphics.

```
Algorithm 1.HSI Denoising
Input: Noisy HSI $\mathcal{H}$
For n=0 : $N$
  update $\mathcal{H}_n = \alpha_n\hat{\mathcal{H}}_{n-1} + (1-\alpha_n)\mathcal{H}_0, \alpha_0 = 0, \alpha_n = 0.8$,
  for $n = 1, \dots, N$.
  Cluster: Clustering full-band patches into $C$ clusters
  by K-means++, clusters are denoted by $\mathcal{Y} = \{\mathcal{Y}_c\}_{c=1}^{C}$.
  For cluster=1 : $C$
    Initialization: Set $\boldsymbol{\mu}_0 = 0$ ,$\beta_0$, $W_0$ (identity matrix),
    $v_0 = R$ (CP rank), initialize $\mathrm{U}^k$ with random values,
    $a_0 = 1, b_0 = 1, a_c = 3$.
    For t=1 : maxiter
      For k=1:$K$
        Update $\tau_r$ and $\mu_r$ via Eqs.(19) and (20).
        Sample $\lambda_r \sim N(\mu_r, {\tau_r}^{-1})$ via Eq. (18).
        Update $W_0^k$ and $v_0^k$ via Eqs.(23) and (24).
        Sample $\Lambda^k \sim Wishart(W_0^k, v_0^k)$ via Eq.(22).
        Sample $\boldsymbol{\mu}^k \sim N(\boldsymbol{\mu}^k, (\Lambda^k)^{-1})$ via Eq. (22).
        For $i_k$ =1 : $I_k$
          Sample $\boldsymbol{u}_{i_k}^k \sim N(\boldsymbol{\mu}_{i_k}^k, (\Lambda_{i_k}^k)^{-1})$ via Eq. (31).
        End for
      End for
      Update $a_0$ and $b_0$.
      Sample $\tau_\epsilon \sim Gamma(a_0, b_0)$ via Eq. (35).
    End for
    Obtain clean FBP cluster $\hat{\mathcal{Y}}$ .
  End for
End for
Output: Denoised HSI $\hat{\mathcal{H}}$.
```

2.3. Basics of Variational Bayesian Inference

Variational Bayesian inference (VBI) is an approximation technique widely used in Bayesian estimation and machine learning. Given a probabilistic model $p(\mathcal{Y},\Theta)$ in which $\mathcal{Y}$ and Θ denote the observed data and the hidden variables, respectively. The task is to search for the posterior distribution $p(\Theta|\mathcal{Y})$ and then infer the model parameter Θ. To circumvent the computational burden of the involved multiple integration, the wisdom of VBI is to search for a probability density function $q(\Theta)$:

$$\begin{aligned}\min_{q(\Theta)} KL(q(\Theta))\,||\,p(\Theta|\mathcal{Y}) \\ \triangleq -E_q\left\{\ln\frac{p(\Theta|\mathcal{Y})}{q\Theta}\right\}\end{aligned} \tag{3}$$

$$KL(q||p) = -\int q(\Theta)\ln\frac{p(\Theta|\mathcal{Y})}{q(\Theta)}d\Theta \tag{4}$$

where $KL(q(\Theta))\,||\,p(\Theta|\mathcal{Y})$ is the Kullback-Leibler divergence between $p(\Theta|\mathcal{Y})$ and $q(\Theta)$.

Assuming $q(\Theta)$ is factorizable, $q(\Theta) = \prod_{s=1:S} q(\Theta_s)$ over the mutually disjoint non-empty subsets Θ_s(i.e., $\cup_{s=1}^{S}\Theta_s = \Theta$ and $\cap_{s=1}^{S}\Theta_s = \varnothing$). Then for each Θ_s, the optimal approximation becomes

$$q_s(\Theta_s) = \frac{e^{\langle \ln p(\mathcal{Y},\Theta)\rangle_{\{i\neq s\}}}}{\int e^{\langle \ln p(\mathcal{Y},\Theta)\rangle_{\{i\neq s\}}}d\Theta_s} \tag{5}$$

where $\langle\cdot\rangle_{i\neq s}$ denotes the expectation with respect to the distributions $q_i(\Theta_i)$ for all $s \neq i$.

3. Proposed Method

The HSI can be viewed as a 3D tensor. Given the additive Gaussian noise, the noisy observation $\mathcal{H}$ can be described as

$$\mathcal{H} = \mathcal{X} + \mathcal{E} \tag{6}$$

where $\mathcal{X}$ is the unknown clean HSI, and $\mathcal{E} \sim N\left(0, \tau_\epsilon^{-1}\right)$ is the additive noise with unknown precision τ_ϵ. $\mathcal{H} \in \mathbb{R}^{H\times W\times B}$ where H, W, B represent the two spatial and one spectral resolutions of the HSI.

Our framework, as shown in Fig. 1, can be regarded as a nonlinear autoregressive moving average (NARMA) model of order 1. The recursive restoration manner can be formulated as follows,

$$\hat{\mathcal{H}}_n = f(\alpha_n\hat{\mathcal{H}}_{n-1} + (1-\alpha_n)\mathcal{H}_0) \tag{7}$$

where $f: \mathbb{R}^{I_1\times I_2\times\cdots\times I_K} \mapsto \mathbb{R}^{I_1\times I_2\times\cdots\times I_K}$ is a nonlinear mapping that is implemented via the embedded BNP-CPD model and the *k-means++* [AV07] classification and grouping manipulations. $\hat{\mathcal{H}}_n$ is the denoised output of the NARMA model at time n, and $\mathcal{H}_0$ is the observed noisy input at time 0. $\alpha_0 = 0, \alpha_n \in (0,1)$, for $n = 1, \dots, N$, are the weights. Since the noise in $\hat{\mathcal{H}}_{n-1}$ is supposed suppressed compared with $\mathcal{H}_0$, the updated input in Eq. (7) will have higher SNR such that BNP-CPD is capable of the incremental recovery of some subtle structures which could have been overwhelmed by heavier noise in the previous iteration.

At each iteration, the updated input HSI is cut into over-lapped 3D patches and classified into C groups using K-means++. Since CP decomposition assumes a common rank for all factor matrices, we collapse the two spatial dimensions into one dimension and assemble each group of HSI patches into a third order tensor $\mathcal{Y}_c, 1 \leq c \leq C$ with modes (spatial, spectral, patch). Such a treatment can effectively exploit the non-local similarity and at the meanwhile prepare tensors with approximately balanced dimensions as well as ranks. Then the new tensors at round n will be sequentially fed into the inner BNP-CPD model.

4. BNP-CPD Model

4.1. Low-Rank and Oblique Factor Hyperpriors

4.1.1. The MGP hyperprior

As tensor rank estimation is an NP-hard and ill-posed problem, probabilistic TD methods usually deal with this difficulty by placing sparsity-inducing priors directly over the factor matrices [ZZC15, CCS*20]. However, sparsity does not necessarily lead to low-rankness of factor matrices, at least not until the algorithm iterations arrive at the highly peaked prior distributions. The CP decomposition in Eq. (1) implies that driving the weights $\{\lambda_r\}_{r=1}^{R}$ towards zero can effectively result in low-rank approximation. Therefore, we employ the MGP hyperprior [RWG*14] over $\{\lambda\}_{r=1}^{R}$ to constrain the number of effective factors in each U^k. The MGP prior is described as:

$$\lambda_r \sim N\left(0, \tau_r^{-1}\right), \quad 1 \leq r \leq R \tag{8}$$

$$\tau_r = \prod_{l=1}^{r} \delta_l, \quad \delta_l \sim Gamma\left(a_c, 1\right) \quad a_c > 1 \tag{9}$$

In practice, one can set a large truncation level R (up to $\max\{I_k\}_{k=1}^{K}$), and the MGP will shrink λ_r towards zero as r increases. Consequently, the leading columns in each U^k will correspond to factors that have high contribution to the latent tensor. In this way, we implement the *nonparametric* characteristic of our BNP-CPD model.

4.1.2. The Gaussian-Wishart Hyperprior

Eq. (2) tells the fact that the element y_i is determined by the row vector dependence, which inspires us that the covariance of the row vector $\boldsymbol{u}_{i_k}^{k}$ should be taken into consideration for better exploration of the multiway correlations. Different from [RWG*14] which assumes the columns of factor matrices are independent to each other, we take the row vectors as the basic inference elements and employ the following multivariate Gaussian priors for them:

$$\boldsymbol{u}_{i_k}^{k} \sim N\left(\boldsymbol{\mu}^k, \left(\Lambda^k\right)^{-1}\right) \tag{10}$$

where $\boldsymbol{\mu}^k \in \mathbb{R}^R$ is the mean vector and $\Lambda^k \in \mathbb{R}^{R\times R}$ is the non-diagonal covariance matrix.

To further enhance the flexibility of our model, we utilize conjugate Gaussian-Wishart priors for the parameters $\boldsymbol{\mu}^k$ and Λ^k. Wishart distribution is a generalization of the multivariate Chi-Square distribution. Therefore, it can be used to describe the covariance matrix of multivariate normal distributions. This means we can capture the correlations among the factors in U^k in a computationally tractable row-by-row manner, which will improve the robustness of the model under complex noises. The hyper-prior on $\boldsymbol{\mu}^k$ and Λ^k are defined as follows

$$\left(\boldsymbol{\mu}^k, \Lambda^k\right) \sim Gaussian-Wishart\left(\boldsymbol{\mu}_0, \beta_0, W_0, \nu_0\right) \tag{11}$$

$$\begin{aligned} p\left(\boldsymbol{\mu}^k, \Lambda^k \mid -\right) &\sim N\left(\boldsymbol{\mu}^k \mid \boldsymbol{\mu}_0, \left(\beta_0\Lambda^k\right)^{-1}\right) \\ &\times Wishart\left(\Lambda^k \mid \mathrm{W}_0, \nu_0\right) \end{aligned} \tag{12}$$

The Wishart distribution with ν_0 degrees of freedom and a $r\times r$ scale matrix W_0 which is usually called the scaling matrix reads

$$\begin{aligned} Wishart\left(\Lambda^k \mid W_0, \nu_0\right) &= \frac{1}{C}\left|\Lambda^k\right|^{\frac{\nu_0-R-1}{2}} \\ &\exp\left\{-\frac{1}{2}\operatorname{tr}\left(W_0^{-1}\Lambda^k\right)\right\} \end{aligned} \tag{13}$$

where $C = 2^{\nu_0 R/2}\Gamma_R\left(\nu_0/2\right)\left|\Lambda\right|^{\nu_0/2}$, $\operatorname{tr}(\cdot)$ is the trace of the matrix. In addition to automatic inference of the tensor rank, employing a Bayesian model allows for convenient estimation of the noise variance from data. According to Eq. (6), we place a non-informative conjugate Gamma prior over the noise precision τ_ϵ to improve the robustness of the model.

$$\tau_\epsilon \sim Gamma\left(a_0, b_0\right) \tag{14}$$

where a_0 is called the shape parameter and b_0 is called the inverse scale parameter in a Gamma distribution.

4.2. Likelihood of CP Decomposition

Thereafter, for convenience we omit the tensor index c in the subscript of $\mathcal{Y}_c, 1 \leq c \leq C$. Accordingto Eq. (1), the likelihood fucntion of $\mathcal{Y}$ in the probabilistic CP decomposition model is given by:

$$\begin{aligned} &p(\mathcal{Y} | \left\{U^k\right\}_{k=1}^{K}, \{\lambda_r\}_{r=1}^{R}, \tau_\epsilon) \\ &\propto \exp(-\frac{\tau_\epsilon}{2}\left\|\mathcal{Y} - \textstyle\sum_{r=1}^{R}\lambda_r \boldsymbol{u}_r^1 \circ \boldsymbol{u}_r^2 \circ \cdots \circ \boldsymbol{u}_r^K\right\|_F^2) \end{aligned} \tag{15}$$

We assume that $\mathcal{Y}$ is corrupted by $\mathcal{E}$ with each element having power $\frac{1}{\sqrt{\tau_\epsilon}}$. According to Eq. (2), then for $\boldsymbol{i} = (i_1, i_2, \ldots, i_k)$, $y_{\boldsymbol{i}} \sim N(\hat{y}_{\boldsymbol{i}}, \tau_\epsilon^{-1})$, the Eq. (15) can be rewritten as:

$$p(y_{\boldsymbol{i}} | \left\{U^k\right\}_{k=1}^{K}, \{\lambda_r\}_{r=1}^{R}, \tau_\epsilon) \propto \exp(-\frac{\tau_\epsilon}{2}(y_{\boldsymbol{i}} - \hat{y}_{\boldsymbol{i}})^2) \tag{16}$$

where $\hat{y}_i$ is specified in Eq. (2).

5. Gibbs Sampling

Gibbs sampling is a widely used MCMC algorithm for Bayesian posterior inference [SM08].Since we employ conjugate priors for parameters and hyper-parameters, the posterior distributions for all parameters can be derived analytically. Based on the VBI principle, Gibbs sampling can be used to solve the BNP-CPD model. In every iteration, each parameter is sequentially updated via sampling from its posterior distribution keeping all other parameters fixed.

5.1. Sampling λ_r

For the update of $\lambda_r, 1 \leq r \leq R$,

$$\begin{aligned} \hat{y}_{\boldsymbol{i}} &= (\prod_{k=1}^{K} \boldsymbol{u}_{i_k r}^{k})\lambda_r + (\sum_{r' \neq r} \lambda_{r'} \prod_{k=1}^{K} \boldsymbol{u}_{i_k r'}^{k}) \\ &= a_{\boldsymbol{i}}^{r}\lambda_r + b_{\boldsymbol{i}}^{r} \end{aligned} \tag{17}$$

$$p(\lambda_r | \mathcal{Y}, U^k, \tau_\epsilon) \propto N(\lambda_r | \hat{\mu}_r, \hat{\tau}_r^{-1}), 1 \leq k \leq K \tag{18}$$

where

$$\hat{\tau}_r = \tau_r + \tau_\epsilon \sum_{\boldsymbol{i}} (a_{\boldsymbol{i}}^{r})^2 \tag{19}$$

and

$$\hat{\mu}_r = \hat{\tau}_r^{-1}\tau_\epsilon \sum_{\boldsymbol{i}} a_{\boldsymbol{i}}^{r}(y_{\boldsymbol{i}} - b_{\boldsymbol{i}}^{r}). \tag{20}$$

5.2. Sampling $\boldsymbol{\mu}^k$ and Λ^k for $1 \leq k \leq K$

The likelihood of the factor matrix U^k can be factorized into the product of conditional distributions of I_k individual vectors:

$$\begin{aligned}&\mathcal{L}(U^k|\boldsymbol{\mu}^k,\Lambda^k)\\&\propto \left|\Lambda^k\right|^{\frac{I_k}{2}} \prod_{i_k=1}^{I_k} \exp\left\{-\frac{1}{2}(\boldsymbol{u}_{i_k}^k-\boldsymbol{\mu}^k)^T\Lambda^k(\boldsymbol{u}_{i_k}^k-\boldsymbol{\mu}^k)\right\}\end{aligned} \tag{21}$$

Given the likelihood term above and the Gaussian-Wishart hyper-prior, we can factorize the joint posterior distribution for hyper-parameters $\boldsymbol{\mu}^k$ and Λ^k as follows:

$$\begin{aligned}&p(\boldsymbol{\mu}^k,\Lambda^k|U^k,\boldsymbol{\mu}_0,W_0,\nu_0,\beta_0)\\&\propto \mathcal{L}(U^k|\boldsymbol{\mu}^k,\Lambda^k)\times N(\boldsymbol{\mu}^k|\boldsymbol{\mu}_0,(\beta_0\Lambda^k)^{-1})\\&\times Wishart(\Lambda^k|W_0,\nu_0)\\&\propto N(\boldsymbol{\mu}^k|\hat{\boldsymbol{\mu}}^k,(\hat{\Lambda}^k)^{-1})\times Wishart(\Lambda^k|\hat{W}_0^k,\hat{\nu}_0^k)\end{aligned} \tag{22}$$

where the parameters in these two distributions can be computed by:

$$\hat{W}_0^k = \left(I_k S^k + \frac{I_k\beta_0}{I_k+\beta_0}(\bar{\boldsymbol{u}}^k-\boldsymbol{\mu}_0)(\bar{\boldsymbol{u}}^k-\boldsymbol{\mu}_0)^T + W_0^{-1}\right)^{-1} \tag{23}$$

$$\hat{\nu}_0^k = I_k + \nu_0 \tag{24}$$

$$\hat{\boldsymbol{\mu}}^k = \frac{1}{I_k+\beta_0}(I_k\bar{\boldsymbol{u}}^k + \beta_0\boldsymbol{\mu}_0) \tag{25}$$

$$\hat{\Lambda}^k = (I_k+\beta_0)\Lambda^k \tag{26}$$

where $\bar{u}^k$ and S^k are two statistics defined as below:

$$\bar{\boldsymbol{u}}^k = \frac{1}{I_k}\sum_{i_k=1}^{I_k}\boldsymbol{u}_{i_k}^k, \tag{27}$$

$$S^k = \frac{1}{I_k}\sum_{i_k=1}^{I_k}(\boldsymbol{u}_{i_k}^k-\bar{\boldsymbol{u}}^k)(\boldsymbol{u}_{i_k}^k-\bar{\boldsymbol{u}}^k)^T \tag{28}$$

5.3. Sampling factor matrices U^k for $1 \leq k, j \leq K$

The factor matrix U^k is updated by sampling all $\boldsymbol{u}_{i_k}^k \in \mathbb{R}^r (1 \leq i_k \leq I_k)$ one by one. The likelihood can be written as

$$\begin{aligned}&\mathcal{L}(\mathcal{Y}|\boldsymbol{u}_{i_k}^k,U^{\{j\neq k\}},\tau_\epsilon)\\&\propto \prod_{i_{\{j\neq k\}}=1}^{I_{\{j\neq k\}}} \exp\left\{-\frac{\tau_\epsilon}{2}(y_{\boldsymbol{i}}-\hat{y}_{\boldsymbol{i}})^2\right\}\\&= \exp\left\{-\frac{\tau_\epsilon}{2}\sum_{i_{\{j\neq k\}}=1}^{I_{\{j\neq k\}}}(y_{\boldsymbol{i}}-(\boldsymbol{u}_{i_j}^j)^T Ha(\boldsymbol{u}_{i_k}^k))^2\right\}\end{aligned} \tag{29}$$

$$Ha(\boldsymbol{u}_{i_k}^k) = (\boldsymbol{u}_{i_1}^1 \circledast \cdots \circledast \boldsymbol{u}_{i_{j-1}}^{j-1} \circledast \boldsymbol{u}_{i_{j+1}}^{j+1} \circledast \cdots \circledast \boldsymbol{u}_{i_K}^K) \tag{30}$$

where the symbol $\circledast$ represents the Hadamard product. Combining Eqs.(12) and (29), the posterior distribution can be written as:

$$\begin{aligned}&p(\boldsymbol{u}_{i_k}^k|\mathcal{Y},U^{\{j\neq k\}},\tau_\epsilon,\boldsymbol{\mu}^k,\Lambda^k)\\&\propto \mathcal{L}(\mathcal{Y}|\boldsymbol{u}_{i_k}^k,U^{\{j\neq k\}},\tau_\epsilon)\times N(\boldsymbol{u}_{i_k}^k|\boldsymbol{\mu}^k,(\Lambda^k)^{-1})\\&\propto N(\boldsymbol{u}_{i_k}^k|\hat{\boldsymbol{\mu}}_{i_k}^k,(\hat{\Lambda}_{i_k}^k)^{-1})\end{aligned} \tag{31}$$

where

$$\hat{\Lambda}_{i_k}^k = \tau_\epsilon \sum_{i_{\{j\neq k\}}=1}^{I_{\{j\neq k\}}} Ha(\boldsymbol{u}_{i_k}^k)Ha(\boldsymbol{u}_{i_k}^k)^T + \Lambda^k \tag{32}$$

$$\hat{\boldsymbol{\mu}}_{i_k}^k = (\hat{\Lambda}_{i_k}^k)^{-1}(\tau_\epsilon \sum_{i_{\{j\neq k\}}=1}^{I_{\{j\neq k\}}} y_i Ha(\boldsymbol{u}_{i_k}^k) + \Lambda^k\boldsymbol{\mu}^k). \tag{33}$$

5.4. Sampling precision τ_ϵ for $1 \leq k \leq K$

The likelihood of all observations is given by

$$\mathcal{L}\left(\mathcal{Y}|\left\{U^k\right\}_{k=1}^K,\tau_\epsilon\right) \propto \prod_{\boldsymbol{i}}\tau_\epsilon^{1/2}\exp\left\{-\tfrac{\tau_\epsilon}{2}(y_{\boldsymbol{i}}-\hat{y}_{\boldsymbol{i}})^2\right\} \tag{34}$$

Combining the above likelihood term and the prior term in Eq. (14) will give the posterior of τ_ϵ, which is also a Gamma distribution parameterized by $\hat{a}_0$ and $\hat{b}_0$:

$$\begin{aligned}&p(\tau_\epsilon|\mathcal{Y},\left\{U^k\right\}_{k=1}^K,a_0,b_0)\\&\propto \mathcal{L}\left(\mathcal{Y}|\left\{U^k\right\}_{k=1}^K,\tau_\epsilon\right)\times Gamma(\tau_\epsilon|a_0,b_0)\\&\propto Gamma(\tau_\epsilon|\hat{a}_0,\hat{b}_0)\end{aligned} \tag{35}$$

where $\hat{a}_0 = \frac{1}{2}\prod_{k=1}^K I_k + a_0$ and $\hat{b}_0 = \frac{1}{2}\sum_{\boldsymbol{i}}(y_{\boldsymbol{i}}-\hat{y}_{\boldsymbol{i}})^2 + b_0$.

6. Experiments and analysis

6.1. HSI datasets

We utilize two HSI datasets, the CAVE dataset and the ICVL dataset to evaluate the denoising performance of the proposed algorithm. The CAVE dataset contains 32 scenes, each with spatial resolution 512×512 and spectral resolution 31 which includes full spectral resolution reflectance data from 400nm to 700nm. The ICVL dataset includes 201 scenes and each has spatial resolution 1300×1392 and the spectral resolution has been downsampled to 31 spectral channels from 400nm to 700nm. Each of these HSIs is scaled into the interval [0,1] in our experiments. Due to computational constraints, HSIs in ICVL dataset are further spatially downsampled to 696×650. With these clean HSIs, we generate additive noise to evaluate the denoising performance of the proposed method.

6.2. Comparison methods

We compare our method with the following denoising methods: BM3D [DFKE07], LRTA [RBBT08], BM4D [MKEF13], TDL [PMX*14], LLRT [CYZ17], BCTF [WF19], LTDL [GCC20],

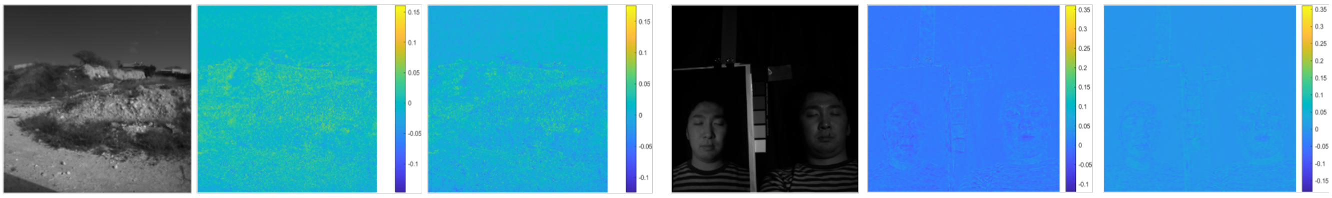

ICVL eve_0331-1647, channel 31 residual of round 1 residual of round 2 CAVE photo_and_face, channel 20 residual of round 1 residual of round 2

Figure 4: *Illustrations of NARMA residuals. The residuals after the first round still contain subtle structures, while the second round refinement removes much of these textures.*

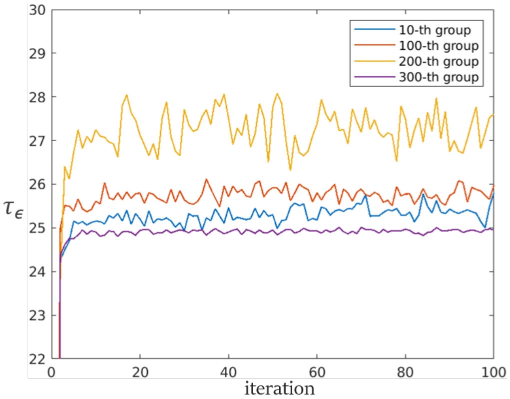
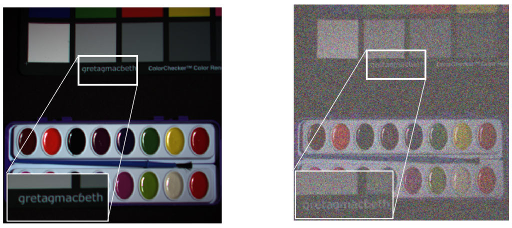


Figure 5: *The estimates of the noise precision τ_ϵ during 100 iterations. We draw the results from four HSI patch groups (10-th,100-th, 200-th, 300-th) of eve_0331-1647 with $\sigma = 0.2$ after clustering. Since $\tau_\epsilon = \frac{1}{\sigma^2}$, the estimated noise variance by BNP-CPD is close to the true value.*


Origin Noisy

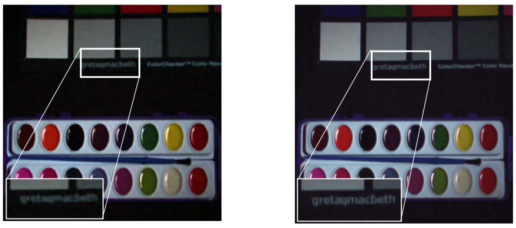

DDS2M Ours-2

Figure 6: *The denoising results of DDS2M and Ours on paints in CAVE with σ=0.2*

HSID-CNN [YZL*19], DSSRL [DWW*19] and the diffusion-based DDS2M [MZZT23]. Most parameters in the compared methods are selected as suggested in the reference or defaults in their published codes.

As for our parameters setting, the HSI patch size is $6 \times 6 \times 31$, with a spatial step size 4. And the number of K-means++ clusters is one hundredth of the total patches. As for the CP rank, we only need to provide an estimation of the upper bound of the tensor rank, and here we set $R = 18$. Other initializations of parameters are given in Algorithm 1. We found that the NARMA iterations beyond 3 will bring minute marginal improvement so we report two rounds results of NARMA, dubbed Ours-1 and Ours-2 in this section.

6.3. Evaluation metrics

We evaluate performance of all methods by five quantitative picture quality indices (PQI), including peak signal-to-noise ratio (PSNR), structure similarity (SSIM), feature similarity (FSIM), dimensionless global relative error of synthesis (ERGAS) and spectral angle mapper (SAM). The better denoising performance of a method can be indicated by high PSNR, SSIM or FSIM and low SAM or ERGAS. PSNR and SSIM are usually used in image processing and computer vision. They evaluate the spatial similarity and structure similarity between the target image and the reference image, but as for HSI, the indices SAM and ERGAS which can evaluate spectral similarity between two images is as important as above three indices.

6.4. Performance evaluation

We generate Gaussian noise with $\sigma = \{0.1, 0.2, 0.3\}$, and generate Gaussian noise with $\sigma = 0.1$ and salt-and-pepper noise with an intensity of 0.05. For each noise setting, all of the five PQI average values for each method on the CAVE dataset and ICVL dataset are recorded in Table 1 and Table 2. It can be observed that the proposed method almost outperforms all the competing methods for the five different PQI. It is worth mentioning that Ours-1 is performed without making use of the residual refinement although the performance is not so outstanding, it still does better than the first three methods and the deep learning methods in Table1. Based on Ours-1, Ours-2 further extracts the features in the residuals and achieves the average PSNR improvement about 0.5dB $\sim$ 1.3dB.

Proceedings published by Eurographics - The European Association for Computer Graphics.

	Noisy HSI	LRTA	BM4D	TDL	LLRT	LTDL	BCTF	HSID-CNN	DSSRL	DDS2M	Ours-1	Ours-2
$\sigma = 0.1$												
PSNR↑	20.000	37.249	38.819	39.506	**41.777**	41.607	41.659	34.108	39.257	41.521	41.228	41.734
SSIM↑	0.141	0.901	0.935	0.945	0.965	0.964	0.964	0.850	0.940	0.967	0.961	**0.968**
FSIM↑	0.750	0.965	0.969	0.979	0.984	0.986	0.986	0.936	0.971	0.949	0.987	**0.989**
ERGAS↓	575.046	73.061	61.286	55.803	44.391	44.288	43.653	126.901	56.792	**36.102**	45.127	43.040
SAM↓	0.836	0.122	0.149	0.113	0.066	0.068	0.074	0.190	0.089	0.098	0.102	**0.066**
$\sigma = 0.2$												
PSNR↑	13.980	33.924	35.134	35.644	38.074	38.007	38.332	30.801	34.400	35.172	37.318	**38.44**
SSIM↑	0.045	0.836	0.867	0.877	0.933	0.931	0.929	0.741	0.849	0.853	0.912	**0.937**
FSIM↑	0.561	0.928	0.935	0.949	0.967	0.970	0.972	0.876	0.934	0.841	0.970	**0.975**
ERGAS↓	1150	107.620	95.317	87.171	73.152	66.808	63.894	226.219	100.287	76.40	71.122	**62.886**
SAM↓	1.064	0.145	0.228	0.187	0.096	0.090	0.104	0.2879	0.150	0.199	0.165	**0.090**
$\sigma = 0.3$												
PSNR↑	10.457	32.048	32.956	33.086	36.199	35.992	36.326	28.566	30.830	32.951	34.884	**36.421**
SSIM↑	0.021	0.791	0.802	0.798	0.908	0.905	0.895	0.690	0.715	0.766	0.856	**0.909**
FSIM↑	0.450	0.899	0.904	0.914	0.951	0.956	0.959	0.862	0.895	0.776	0.953	**0.961**
ERGAS↓	1725	133.853	124.300	117.829	93.871	84.208	80.438	331.928	154.808	105.95	94.309	**79.212**
SAM↓	1.185	0.160	0.289	0.252	0.112	**0.106**	0.131	0.347	0.221	0.246	0.218	0.110

Table 1: *The average denoising performance on CAVE and ICVL.*

	Noisy HSI	LTDL	LLRT	BCTF	Ours-2
PSNR↑	15.765	28.274	19.302	28.635	**28.673**
SSIM↑	0.090	0.721	0.182	0.735	**0.737**
FSIM↑	0.480	0.961	0.757	0.950	**0.964**
ERGAS↓	544.322	265.327	644.484	**255.946**	260.154
SAM↓	0.572	0.213	0.595	**0.219**	0.221

Table 2: *The average denoising performance under a combination of Gaussian noise with* $\sigma = 0.1$ *and salt-and-pepper noise with an intensity of 0.05.*

Fig. 4 provides a visual inspection of the residuals in the first round of NARMA, illustrating the effectiveness of this recursive manner of restoration.

To visualize the denoising performance, we display two results in Fig. 2 and Fig. 3, respectively. The color images are composed of bands at 700 nm, 500 nm, and 450 nm of corresponding denoised HSIs. Additionally, Fig. 6 shows the denoising results of diffusion-based method DDS2M and our method. It can be observed that the proposed methods introduce less blurry effect and recover more texture details compared with other methods. Fig. 5 shows the noise precision estimates in the Gibbs sampling process, validating that our method can automatically estimate the noise variance from the input data and thus meets real application requirements better.

Finally, we present some comments on the results of the two deep learning based methods, HSID-CNN [YZL*19] and DSSRL [DWW*19]. Both of the two methods rely on end-to-end training. We used their published modules to test our samples with necessary dimension adjustment. Not surprisingly, their performances are not so competitive as in the original publications. But as the training set of DSSRL includes the CAVE dataset, the drop of performance is not so severe as HSID-CNN. We test SURE-CNN [VHUS20], which is an unsupervised CNN trained based on the Stein's unbiased risk estimate. However, there are about 42% outputs of SURE-CNN with PSNR less than 20dB, which is maybe due to the error of estimates in the noise variance. Therefore, we did not include its performance in Table 1. As shown in Fig. 6, we also test the diffusion-based method DDS2M [MZZT23]. These trails suggest that for the HSI denoising problem tensor-based methods are still more capable of exploiting the latent multiway correlations in HSIs in a direct and principled way.

6.5. Ablation experiments

Table 3 presents the results of the IBNP-CPD network with different module configurations. Excluding the recursive restoration module significantly impacts HSI denoising performance. Specifically, on the CAVE dataset with Gaussian noise of $\sigma = 0.1$, the removal of the module results in a decline in all evaluation metrics. PSNR drops from 41.31 to 41.25, SSIM from 0.9787 to 0.9779, FSIM from 0.9792 to 0.9785, ERGAS from 36.90 to 37.14 (the lower, the better), and SAM from 0.0758 to 0.0776 (the lower, the better), highlighting the crucial role of local components in handling highly noisy inputs.

7. Conclusion

The proposed IBNP-CPD denoising framework is characterized by a recursive and hierarchical methodology that is capable of effec-

Model	PSNR↑	SSIM↑	FSIM↑	ERGAS↓	SAM↓
IBNP-CPD net	**41.31**	**0.9787**	**0.9792**	**36.90**	**0.0758**
w/o recursive restoration	41.25	0.9779	0.9785	37.14	0.0776

Table 3: *Ablation study results on the important design elements of IBNP-CPD net, The "w/o recursive restoration" implies IBNP-CPD excluding the recursive restoration module.*

tively seizing the multiway correlations in HSI, and by a principled Bayesian nonparametric tensor decomposition model that employs the multiplicative Gamma process and the Gaussian-Wishart hyperpriors over the latent tensor factors. Experimental results show that our performance significantly outperforms SOTA traditional methods and self-supervised deep learning.

In denoising tasks, Bayesian nonparametrics (BNP) offers clear advantages over deep learning. It naturally quantifies uncertainty. In contrast, deep learning denoising results are single-valued. Moreover, BNP can assess the reliability of the denoising process. Additionally, BNP performs effectively even with limited data and prior knowledge, demonstrating strong adaptability-unlike deep learning, which heavily relies on large datasets. In the future, BNP and deep learning can be integrated into hybrid models, with deep learning extracting features and BNP handling uncertainty and incorporating prior knowledge. Furthermore, BNP principles can be embedded into deep learning architectures to enhance interpretability and uncertainty quantification capabilities.

References

[AEB06] Aharon M., Elad M., Bruckstein A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process. 54*, 11 (2006), 4311–4322. 1

[AOM14] Abe B. T., Olugbara O. O., Marwala T.: Comparison of svm and rf for hyperspectral image classification. *J. Earth Syst. Sci. 123* (2014), 779–790. 1

[AV07] Arthur D., Vassilvitskii S.: K-means++: The advantages of careful seeding. vol. 8, pp. 1027–1035. `doi:10.1145/1283383.1283494`. 4

[CCS*20] Cheng L., Chen Z., Shi Q., Wu Y.-C., Theodoridis S.: Towards probabilistic tensor canonical polyadic decomposition 2.0: Automatic tensor rank learning using generalized hyperbolic prior. *arXiv: 1406.2407* (2020). 4

[CCZ*18] Chen Y., Cao X., Zhao Q., Meng D., Xu Z.: Denoising hyperspectral image with non-i.i.d. noise structure. *IEEE Trans. Cybernetics 48*, 3 (2018), 1054–1066. `doi:10.1109/TCYB.2017.2677944`. 2

[CYZ17] Chang Y., Yan L., Zhong S.: Hyper-laplacian regularized unidirectional low-rank tensor recovery for multispectral image denoising. In *2017 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (2017), pp. 5901–5909. `doi:10.1109/CVPR.2017.625`. 6

[DFKE07] Dabov K., Foi A., Katkovnik V., Egiazarian K.: Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. Image Process. 16*, 8 (2007), 2080–2095. `doi:10.1109/TIP.2007.901238`. 1, 6

[DLF*20] Dian R., Li S., Fang L., Lu T., Bioucas-Dias J. M.: Nonlocal sparse tensor factorization for semiblind hyperspectral and multispectral image fusion. *IEEE Trans. Cybern. 50*, 10 (2020), 4469–4480. `doi:10.1109/TCYB.2019.2951572`. 1

[DWW*19] Dong W., Wang H., Wu F., Shi G., Li X.: Deep spatial–spectral representation learning for hyperspectral image denoising. *IEEE Transactions on Computational Imaging 5*, 4 (2019), 635–648. `doi:10.1109/TCI.2019.2911881`. 7, 8

[FLW*22] Feng X.-R., Li H.-C., Wang R., Du Q., Jia X., Plaza A.: Hyperspectral unmixing via nonnegative matrix factorization: A review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 15* (2022), 4414–4436. 1

[GCC20] Gong X., Chen W., Chen J.: A low-rank tensor dictionary learning method for hyperspectral image denoising. *IEEE Trans. Signal Process. 68* (2020), 1168–1180. `doi:10.1109/TSP.2020.2971441`. 1, 2, 6

[JSG*16] Ju F., Sun Y., Gao J., Hu Y., Yin B.: Nonparametric tensor dictionary learning with beta process priors. *Neurocomputing 218* (2016), 120–130. URL: `https://www.sciencedirect.com/science/article/pii/S0925231216309602`, `doi:10.1016/j.neucom.2016.08.064`. 2

[KB09] Kolda T. G., Bader B. W.: Tensor decompositions and applications. *SIAM Rev.* (2009). 3

[LBF12] Liu X., Bourennane S., Fossati C.: Denoising of hyperspectral images using the parafac model and statistical performance analysis. *IEEE Trans. Geosci. Remote Sens. 50*, 10 (2012), 3717–3724. `doi:10.1109/TGRS.2012.2187063`. 1

[LCG*24] Li D., Chu D., Guan X., He W., Shen H.: Adaptive regularized low-rank tensor decomposition for hyperspectral image denoising and destriping. *IEEE Trans. Geosci. Remote Sens. 62* (2024), 1–17. `doi:10.1109/TGRS.2024.3385536`. 1

[LMF19] Liu X., Meng Y., Fu M.: Classification research based on residual network for hyperspectral image. In *2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP)* (2019), pp. 911–915. 2

[LZJ*21] Luo Y.-S., Zhao X.-L., Jiang T.-X., Zheng Y.-B., Chang Y.: Hyperspectral mixed noise removal via spatial-spectral constrained unsupervised deep image prior. In *IEEE AESORS* (2021), IEEE J.Sel., pp. 9435–9449. 2

[MAU*24] Mardieva S., Ahmad S., Umirzakova S., Rasool M. J. A., Whangbo T. K.: Lightweight image super-resolution for iot devices using deep residual feature distillation network. *Knowledge-Based Systems 285* (2024), 111343. URL: `https://www.sciencedirect.com/science/article/pii/S0950705123010912`, `doi:10.1016/j.knosys.2023.111343`. 2

[MKEF13] Maggioni M., Katkovnik V., Egiazarian K., Foi A.: Nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE Trans. Image Process. 22*, 1 (2013), 119–133. `doi:10.1109/TIP.2012.2210725`. 1, 6

[MLMLC*17] Montesinos-López A., Montesinos-López O. A., Cuevas J., Mata-López W. A., Burgueño J., Mondal S., Huerta J., Singh R., Autrique E., González-Pérez L., et al.: Genomic bayesian regression models for predicting wheat yield using hyperspectral data. *Plant Methods 13* (2017), 1–29. 1

[MZZT23] Miao Y., Zhang L., Zhang L., Tao D.: Dds2m: Self-supervised denoising diffusion spatio-spectral model for hyperspectral image restoration. In *ICCV* (2023), pp. 12086–12096. 2, 7, 8

[OBP*23] Omia E., Bae H., Park E., Kim M. S., Baek I., Kabenge I., Cho B.-K.: Remote sensing in field crop monitoring: A review of sensor systems and data analyses. *Remote Sens. 15*, 2 (2023), 354. 1

[PMX*14] Peng Y., Meng D., Xu Z., Gao C., Yang Y., Zhang B.: Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* (2014), pp. 2949–2956. `doi:10.1109/CVPR.2014.377`. 1, 2, 6

[RBBT08] Renard N., Bourennane S., Blanc-Talon J.: Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geosci. Remote Sens. Lett. 5*, 2 (2008), 138–142. `doi:10.1109/LGRS.2008.915736`. 1, 6

[RWG*14] RAI P., WANG Y., GUO S., CHEN G., DUNSON D., CARIN L.: Scalable bayesian low-rank decomposition of incomplete multiway tensors. In *ICML* (2014), PMLR, pp. 1800–1808. 5

[SH19] SIDOROV O., HARDEBERG J. Y.: Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution. In *ICCV Workshops* (2019). 2

[SM08] SALAKHUTDINOV R., MNIH A.: Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proc. 25th Int. Conf. Mach. Learn.* (2008), pp. 880–887. 5

[UVL18] ULYANOV D., VEDALDI A., LEMPITSKY V.: Deep image prior. In *CVPR* (2018), pp. 9446–9454. 2

[VHUS20] VAN HAN N., ULFARSSON M., SVEINSSON J.: Hyperspectral image denoising using sure-based unsupervised convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing PP* (07 2020). `doi:10.1109/TGRS.2020.3008844`. 8

[WF19] WEI K., FU Y.: Low-rank bayesian tensor factorization for hyperspectral image denoising. *Neurocomputing 331* (2019), 412–423. URL: `https://www.sciencedirect.com/science/article/pii/S0925231218312116`, `doi:https://doi.org/10.1016/j.neucom.2018.10.023`. 6

[WNZ*22] WANG Z., NG M. K., ZHUANG L., GAO L., ZHANG B.: Nonlocal self-similarity-based hyperspectral remote sensing image denoising with 3-d convolutional neural network. *IEEE Trans. Geosci. Remote Sens. 60* (2022), 1–17. 1

[WWB*23] WU C., WANG D., BAI Y., MAO H., LI Y., SHEN Q.: Hsr-diff: Hyperspectral image super-resolution via conditional diffusion models. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.* (2023), pp. 7083–7093. 1

[XWCW19] XU Y., WU Z., CHANUSSOT J., WEI Z.: Nonlocal patch tensor sparse representation for hyperspectral image super-resolution. *IEEE Trans. Image Process. 28*, 6 (2019), 3034–3047. 1

[YWJ*23] YE F., WU Z., JIA X., CHANUSSOT J., XU Y., WEI Z.: Bayesian nonlocal patch tensor factorization for hyperspectral image super-resolution. *IEEE Trans. Image Process. 32* (2023), 5877–5892. `doi:10.1109/TIP.2023.3326687`. 2

[YZL*19] YUAN Q., ZHANG Q., LI J., SHEN H., ZHANG L.: Hyperspectral image denoising employing a spatial–spectral deep residual convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing 57*, 2 (2019), 1205–1218. `doi:10.1109/tgrs.2018.2865197`. 7, 8

[ZCP*12] ZHOU M., CHEN H., PAISLEY J., REN L., LI L., XING Z., DUNSON D., SAPIRO G., CARIN L.: Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans. Image Process. 21*, 1 (2012), 130–144. `doi:10.1109/TIP.2011.2160072`. 2

[ZLM*17] ZHONG Z., LI J., MA L., JIANG H., ZHAO H.: Deep residual networks for hyperspectral image classification. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (2017), pp. 1824–1827. 2

[ZZC15] ZHAO Q., ZHANG L., CICHOCKI A.: Bayesian cp factorization of incomplete tensors with automatic rank determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence 37*, 9 (2015), 1751–1763. URL: `https://ieeexplore.ieee.org/document/7010937/`, `doi:10.1109/TPAMI.2015.2392756`. 4