HSI Denoising: Global Non-Location Low-Rank Factorization and Sparse Factorization Based on Iteratively Boost Method

Hao Yan¹*

¹School of Computer Science, Chongqing University, Chongqing, 400044, China
*Corresponding e-mail: yanhao@cqu.edu.cn

Abstract. This study proposes global and non-location low-rank factorization and sparse factorization based on iteratively boost (GNS-ITboost) scheme from the perspective of the iteration method to improve Hyperspectral image (HSI) denoising results. First, this study establishes a novel iterative boost method by jointing global low-rank factorization, non-location low-rank factorization and sparse factorization to HSI denoising. Second, we propose the non-convex method, weight schatten $p$-norm method to more effectively approximate the rank function to obtain better non-location low-rank factorization. Third, this study provides a new noise estimation method by prior knowledge to accelerate obtaining a better denoising effect. Finally, proximal alternating minimization and weight sparse ideas are used to obtain the solution of denoising optimization problems. Furmore, evaluation results indicate that the proposed method can outperform the state-of-the-art methods in terms of FastHyDe, GLF, LRMR, SNLRSF, SSTV.

1. Introduction

In the past decade, hyperspectral images have received more and more attention. Hyperspectral image (HSI) can capture more information than traditional gray images and RGB images. Different materials have different reflectivities in different infrared spectral bands [1]. HSIs [2] have been applied in environment monitoring [2], classification [4][5] and target detection [6]. But the noise inevitably exists in HSIs, which is always mixed noise that can be obtained from sensor and sign transmission.

For the hyperspectral image, there exists numerous frames. The previous method is based on the Plug-and-Play(PnP) frame [7]. The assumption regularizer is decoupled that makes the hyperspectral image denoising problem transform into gray image denoising problem. In subsequent work, the research considers the HSI denoising from spectral and space. In [8], Yuan and Zhang used Bregman split iteration to solve the HSI denoising problem. In [9], Chen et. al. adopted the PCA method to deal with spectral. Then, they utilize BM4D to denoise reduce image. In [10], Chen et.al. hold the first band of HSI which is processed by PCA, has more information than other bands. In [1], Bioucas-Dia et.al. point out hyperspectral image that has high correlation between global spectrals. The correlation between global spectrals can be utilized to reduce the calculation. Bioucas-Dia provided an assumption, spectral vectors live in a subspace whose dimension is much smaller than the original HSI. Zhuang et. al. proposed FastHyDe[11] and GLF[12], which convert hyperspectral image denoising into RCI denoising by subspace method. FastHyDe algorithm [11] used Plug-and-Play(PnP) frame in subspace to accelerate denoising. They also introduce BM3D. Base on low-rank tensor idea, Zhuang propose a GLF algorithm [12].

In reality, the mixed noise often exists in HSI, such as Gaussian noise, impulse noise, stripe noise, etc. If sparse noise exists in HSI, it needs to consider the low-rank property of spatial and sparse property.
in the denoising method. One of the famous methods is robust principal component analysis (Robust-PCA) which is designed by low-rank and sparse property. In [13], Zhang et al. propose LRMR which converts HSI to 2-D matrix, then using Robust-PCA to recover HSI. In [14], Xie et al. propose weighted schatten p-norm minimization (WSNM) into the low-rank part of Robust-PCA. Through non-local low-rank ideas, we can construct similar tensor groups. However, it does not consider the low-rank property global spectral in this method. This lead to denoising time unacceptable.

The non-local method is a wide denoising method, which enhances non-local similarity to denoising HSI. The non-local method is proposed in the gray images, such as BM3D, BM4D. Xie et al. propose the non-local method into HSI. In recently, researchers consider two low-rank properties, global spectral low-rank property and non-local low-rank property, such as FastHyDe [11], GLF[12]. Cao et al. [15] introduce sparse factorization to design the denoising model. In [16], He et al. introduce weighted nuclear norm minimization (WNNM) to non-local low-rank method. High correlation exists in the band of the hyperspectral image, and some algorithms [14] don’t consider the correlation. In [14], Xie et al. denoised the original noise HSI by non-local low-rank method. Ref works on [11][12][15] that do not consider the iterative regularization method which can dig more information from the noise HSI.

To solve these problems and improve the denoising effect, this study proposes a novel GNS-ITboost method for HSI denoising. The contributions of this study can be summarized as follows:

- This study provides a new iterative boost framework. In each iterative step, the solution parameters will be adjusted according to the last iteration solution. At the same time, through this iterative method, the HSI denoising effect is promoted in every iteration.
- This study introduces the non-convex method, weight schatten p-norm, to enhance the similarity of the non-local tensor of the subspace of HSI. In every iteration step, according to prior knowledge of the above iteration, this study provides a new noise estimation method for every non-local tensor group to accelerate obtaining a better denoising effect.
- An proximal alternating minimization (PAM)-based on algorithm [18] is introduced to solve denoising problem, and a weighted sparse idea is introduced. Compared with several excellent methods, our method can provide a better denoising effect in simulation noise and real noise of HSI.

The remainder of this paper is organized as follows. The relevant research is discussed in section 2. Section 3 describes our proposed method. Experiment results validating the effectiveness of our method are shown in Section 4. The conclusion is providing at the final of this paper.

2. Problem Formulation

Hyperspectral images will cause various noises during the acquisition and transmission process, including Gaussian noise, impulse noise, stripe, deadline et al. The observed HSI is denoted as $Y \in \mathbb{R}^{M \times N \times B}$. Generally speaking, the degradation of HSI can be described as:

$$Y = X + N + S,$$

where $X \in \mathbb{R}^{M \times N \times B}$ represents clean HSI, and $N, S \in \mathbb{R}^{M \times N \times B}$ are Gaussian noise, sparse noise, respectively.

The method of HSI denoising can be mathematically described as

$$\text{arg min}_{X,S} \frac{1}{2} \| Y - X - S \|^2_F + \mu_1 R_1(X) + \mu_2 R_2(S),$$

where $R_1(\cdot)$ denotes the regularization term which is used to explore the prior information of the observed HSIs. $R_2(\cdot)$ denotes the regularization of sparse noise.

2.1. Nonlocal Low-Rank Approximation

$R_1(\cdot)$ is set as nonlocal low-rank approximation $\| \cdot \|_{NL}$. The model (2) can be formulated as

$$\text{arg min}_{X,S} \frac{1}{2} \| Y - X - S \|^2_F + \mu_1 \| X \|_{NL} + \mu_2 R_2(S),$$

(3)
The nonlocal low-rank approximation can be divided into three steps:

- According to the step size, the noisy HSI can be divided into many overlapping 3-D tensors. Each tensor has a non-local region. In the non-local region, we can obtain similarity tensor group of reference tensor by K-NN. \( R_i \) denotes the index of similarity tensors group. This study adopts the straighten method which is provided in [16] to construct corresponding matrix \( Y_i \) for similarity tensors group.
- The study adopts low-rank approximation method to obtain \( X_i \) for corresponding matrix \( Y_i \).
- We utilize the inverse method of straightening method to obtain the tensor group. Then, each tensor is placed in the corresponding position in HSI. We can obtain denoised HSI through dividing the reconstructed image by the count HSI.

2.2. Iteration Regularization
For hyperspectral images denoising, iterative methods are not considered. The main reason is that hyperspectral images have a property, a large number of spectral. This feature leads to the calculation times will be unacceptable. Xie [14] et. al. provide WSN-LRMA method that uses iteration method to denoise HSIs. In recently, some works adopt global spectral low-rank method to convert the HSI denoising problem into RCI denoising so as to reduce calculation, such as FastHyDe[11], GLF[12]. Based on the global spectral low-rank method, He et.al. provide a new iteration method, NGmeet[16].

3. PROPOSED GNS-ITBOOST METHOD
This method consists of 6 parts, including noise normalization, global spectral low-rank factorization, acquisition of non-local low-rank factors by WSNM and mapping relationship of tensor groups, PAM-based hyperspectral image denoising algorithm, generalization inverse, and iterative regularization. The structure of GNS-ITboost method is presented in Figure. 1. This study summarizes GNS-ITboost in Algorithm 2.

Figure 1. The flowchart of the proposed GNS-ITboost.

3.1. Noise Normalization
In [16], researcher considers additive noise which satisfies independent and identically distributed (i.i.d.) situations. In order to deal with more general situation, this study adopts normalization strategy to convert non-i.i.d. problem into i.i.d. problem. This study makes the covariance matrix denoted as \( C_{(k)} = E(n_{(k)}^2) \), where \( n_{(k)} \) is the matrix which straightened the band of hyperspectral image and stacking. The noise normalization HSI can be described as:

\[
\hat{Y}_{(k)} = Y_{(k)} \times \sqrt{C^{-1}}
\]

The non-i.i.d. problem transform to i.i.d. problem, and the denoising model can be rewritten as

\[
\arg \min_{\hat{Y},s} \frac{1}{2} \| \hat{Y}_{(k)} - \tilde{X}_{(k)} \|_F^2 + \mu_1 \| \tilde{X}_{(k)} \|_{NL} + \mu_2 R_2(\tilde{S}_{(k)}).
\]
In each iteration, the generalization inverse can be written as follow
\[ \mathcal{X}_{(k)} = \mathcal{X}_{(k)} \times 3 \sqrt{C_{(k)}} \]  
(6)

3.2. Global Spectral Low-Rank Factorization
Since hyperspectral images are highly correlated, this study uses this property to reduce calculation. In [1], Joes et.al. provide an assumption that the spectral vectors live in a \( K \)-dimensional subspace \( \mathcal{S}_K \) with \( B \gg K \). \( \mathcal{X} \) can be written as:
\[ \mathcal{X}_{(k)} = \mathcal{M}_{(k)} \times 3 \mathcal{A}_{(k)}, \]  
(7)
where \( \mathcal{M}_{(k)} \in \mathbb{R}^{M \times N \times B} \) is the reduce image, \( \mathcal{A}_{(k)} \in \mathbb{R}^{K \times B} \) is orthogonal basis matrix. According to Eq. (7), the HSI denoising model (5) can be reformulated as
\[ \begin{align*}
\arg \min_{\mathcal{M}_{(k)}, \mathcal{A}_{(k)}} & \frac{1}{2} \| \mathcal{Y}_{(k)} - \mathcal{M}_{(k)} \times 3 \mathcal{A}_{(k)} - \mathcal{S}_{(k)} \|_F^2 + \mu_1 \| \mathcal{M}_{(k)} \|_{NL} + \mu_2 R_2(\mathcal{S}_{(k)}), \\
\text{s.t.} & \quad \mathcal{A}_{(k)}^T \mathcal{A}_{(k)} = I_K.
\end{align*} \]  
(8)
This study adopts Hysime algorithm [2] to learn \( \mathcal{A}_{(k)} \) in every iteration.

3.3. Obtain Non-Local Low-Rank Approximation Factorization
This study applies \( G_{(i(k))} \) to represent the corresponding matrix of \( i \)-th similarity tensor groups which is obtained by non-local method [16]. \( \mathcal{L} \) is \( \mathcal{Y}_{(k)} \times A_{(k)} \), \( \mathcal{R}_{(i(k))} \) is the mapping from tensors to \( i \)-th tensor group. In this stage, we need to obtain \( \mathcal{L}_{(i(k))} , \mathcal{R}_{(i(k))} , \mathcal{L}_{(k)} \) for next stage. We utilize non-local regularization to obtain low-rank approximation tensor \( \mathcal{L}_{(k)} \). The problem of obtaining \( \mathcal{L} \) can be described as
\[ \mathcal{L}_{(k)} = \arg \min_{\mathcal{L}} \frac{1}{2} \| \mathcal{L}_{(k)} - \mathcal{L} \|_F^2 + \mu \| \mathcal{L} \|_{NL}, \]  
(9)
where \( \| \cdot \|_{NL} \) is a non-local approximation regularizer. According to non-local low-rank regularization, the problem can be transformed to low-rank matrix approximation problem to solution \( G_{(i(k))} \) through solving the following problem:
\[ \mathcal{G}_{(i(k))} = \arg \min_{\mathcal{G}} \frac{1}{\sigma_{(i(k))}^2} \| \mathcal{G}_{(i(k))} - \mathcal{G}_{(i(k))} \|_F^2 + \kappa(\mathcal{G}_{(i(k))}), \]  
(10)
where \( \kappa(\cdot) \) is a low-rank regularizer. This study adopts the weighted Schatten \( p \)-norm method.
To utilize spatial low-rank approximation on each non-local similarity group \( G_{(i(k))} \), traditional method usually sets \( \sigma_{(i(k))} \) to be 1 in each non-local group \( G_{(i(k))} \). This study provides two situations of corresponding noise estimation for the non-local similarity group as
- First iteration:
\[ \sigma_{(i(k))} = 1; \]  
(11)
- Otherwise:
\[ \sigma_{(i(k))} = \gamma \sqrt{\text{mean}(\| \mathcal{Y}_{(k)} - \mathcal{Y}_{(k)} \|_F^2)}. \]  
(12)
where \( \mathcal{Y}_{(k)} = \mathcal{Y} \times A \sqrt{C_{(k)}} \), \( \mathcal{Y} \) is the original noise image, \( \gamma \) is the scaling factor for re-estimation of noise variance, and \( \text{mean}(\cdot) \) represents the average of the tensor elements.
This study utilizes the weighted schatten \( p \)-norm of \( Z \in \mathbb{R}^{m \times n} \), which can be defined as
\[ \|Z\|_{w, z} = \left( \sum_{i=1}^{\min(n, m)} w_i z_i \right), \] (13)

where \(0 < p < 1\), \(w = [w_1, \ldots, w_{\min(m, n)}]\) is a non-negative weight vector, \(z_i\) is the \(i\)th singular value of \(Z\). This study uses the WSNM method \([17]\) to estimate \(\hat{G}_{i(k)}\). Then, the optimization problem can be defined as

\[
\hat{G}_{i(k)} = \arg \min_G \frac{1}{\sigma_{i(k)}^2} \| G_{i(k)} - \tilde{G}_{i(k)} \|^p + \| G_{i(k)} \|^p_{w, z_i},
\] (14)

where \(\sigma_{i(k)}^2\) denotes the noise variance, and the second term plays the role of low-rank regularization. \(g_{\alpha, i_{(k)}}\), the \(j\)-th singular value of \(G_{i(k)}\), with a larger value is more important than small ones since it represents the energy of the \(j\)-th component of \(G_{i(k)}\). Similarly, \(\hat{g}_{\alpha, i_{(k)}}\), the \(i\)-th singular value of the optimal solution of model (14), it should be shrunk less. The WSNM algorithm is summarized in Algorithm 1.

**Algorithm 1 WSNM algorithm**

Input: \(\tilde{G}_{i(k)}\)
Output: \(\hat{G}_{i(k)}\)

1. for \(j = 1 : r\) do
2. \(t_p^{\text{OPT}} = (2 \lambda w_s (1 - p))^{\frac{1}{p-1}} + \lambda w_s p (2 \lambda w_s (1 - p))^{\frac{p-1}{p}}\)
3. if \(|g_{\alpha, j, i_{(k)}}| \leq t_p^{\text{OPT}}\) then
4. \(\hat{g}_{\alpha, j, i_{(k)}} = 0\)
5. else
6. \(\hat{g}_{\alpha, j, i_{(k)}} = g_{\alpha, j, i_{(k)}}\)
7. for \(t = 0, 1, \ldots, T\) do
8. \(\hat{g}_{\alpha, j, i_{(k)}}^{t+1} = |g_{\alpha, j, i_{(k)}}| - w_s p (\hat{g}_{\alpha, j, i_{(k)}}^t)^{p-1}\)
9. \(t = t + 1\)
10. end for
11. \(\hat{g}_{\alpha, j, i_{(k)}}^{t+1} = \text{sgn} (g_{\alpha, j, i_{(k)}}) \hat{g}_{\alpha, j, i_{(k)}}^{t+1}\)
12. end if
13. end for

### 3.4. PAM-Based HSI Denosing

In each iteration process, based on the low-rank tensor \(G_{i(k)}\) and the mapping relationship \(R_{i(k)}\), we convert the tensor \(Y_{i(k)}, M_{i(k)}, S_{i(k)}\) into matrix \(Y \in \mathbb{R}^{B \times MN}, M \in \mathbb{R}^{K \times MN}, S \in \mathbb{R}^{B \times MN}\). For convenience to describe, we use \(A\) to represent \(A_{i(k)}\). The HSI denoising problem (3) is converted as follows

\[
f(A, M, S) = \arg \min_{A, M, S} \frac{1}{2} ||Y - AM - S||_p^2
\]

\[
+ \mu_1 \sum_i \frac{1}{\sigma_{i(k)}^2} ||R_{i(k)} - G_{i(k)}||_p^2 + \mu_2 ||W \odot S||_1 + \xi_i (A^T A).
\] (15)
This study uses a weighting factor here to promote this sparsity. \( f(A, M, S) \) is the objective function of (15). According to the PAM-based framework, we convert the problem (15) as follow:

\[
A_{i+1} = \arg \min_{A} f(A, M_i, S_i) + \frac{\tau}{2} ||A - A_i||^2_F, \tag{16}
\]

\[
M_{i+1} = \arg \min_{M} f(A_{i+1}, M, S_i) + \frac{\tau}{2} ||M - M_i||^2_F, \tag{17}
\]

\[
S_{i+1} = \arg \min_{S} f(A_{i+1}, M_{i+1}, S) + \frac{\tau}{2} ||S - S_i||^2_F, \tag{18}
\]

where \( \tau > 0 \) denotes the proximal parameter. We make \( L(\cdot) \) and \( R(\cdot) \) be the left and right singular, respectively. Then, this problem can be solved as follow.

- **Step 1:** Update \( A \)
  \[
  A_{i+1} = \arg \min_{A, A'} f(A, M, S) + \frac{1}{2} ||Y - AM_i - S_i||^2_F + \frac{\tau}{2} ||A - A_i||^2_F = L(\rho) R(\rho), \tag{19}
  \]
  where \( \rho = (Y - S_i)M_i - \tau A_i \).

- **Step 2:** Update \( M \)
  \[
  M_{i+1} = \arg \min_{M} \frac{1}{2} ||Y - A_{i+1}M - S_i||^2_F \tag{20}
  \]
  \[+ \mu_i \sum_t \frac{1}{\sigma_{i(t)}^2} \|R_{i(t)}M - G_{i(t)}\|^2_F + \frac{\tau}{2} ||M - M_i||^2_F. \]

According to quadratic optimization problem, we can obtain

\[
M_{i+1} = \left( I + \mu_i \sum_t \frac{2}{\sigma_{i(t)}^2} R_{i(t)}^T R_{i(t)} + \tau I \right)^{-1} \times \left( A_{i+1}^T (Y - S_i) + \mu_i \sum_t \frac{2}{\sigma_{i(t)}^2} R_{i(t)}^T G_{i(t)} + \tau M_i \right). \tag{21}
\]

- **Step 3:** Update \( S \)
  \[
  S_{i+1} = \arg \min_{S} \frac{1}{2} ||Y - A_{i+1}M_{i+1} - S||^2_F + \mu_i ||W_i \odot S||_1 + \frac{\tau}{2} ||S - S_i||^2_F. \tag{22}
  \]
  We have:
  \[
  S_{i+1} = \text{shrink}_{\mu_i} \left( \hat{S}, W_i \odot \frac{\mu_i}{1 + \tau} \right), \tag{23}
  \]
  where \( \hat{S} = (Y - A_{i+1}M_{i+1} + \tau S_i)/(1 + \tau) \), \( W_i = 1/(||\hat{S}|| + \epsilon) \).

### 3.5. Iteration Regularization

In this stage, we utilize its iterative regularization, and the mathematical expression can be described as

\[
\mathcal{Y}_{(k+1)} = \eta \mathcal{X}_{(k)} + (1 - \eta) \mathcal{Y}, \tag{24}
\]

where \( \eta \) is trade-off denoised image \( \mathcal{X}_{(k)} \) and original noisy image \( \mathcal{Y} \). Following the work in [16], in the iterative process we also increase \( K \) by

\[
K = K + \lambda k, \tag{25}
\]

where \( \lambda \) is a constant value. The GNS-ITboost method is summarized in Algorithm 2.
Algorithm 2  GNS-ITboost algorithm

Input: HSI $\mathcal{Y}$

Output: Denoised HSI $\tilde{\mathcal{X}}_{(k)}$

1 for $k = 1, 2, \ldots, \text{iter}$ do
  1) Noise normalization:
  2) Obtain low-rank factor of global spectral via Hysime algorithm [2];
  3) Obtain non-location factor
  3-1) Obtain $\tilde{G}_{(k)}$, mapping relationship $\mathcal{R}_{(k)}$, noise estimation $\sigma_{(k)}$;
  3-2) Obtain $\hat{G}_{(k)}$ by Algorithm 1 (WSNM);
  4) PAM-based on HSI denoising:
     While convergence
     update $A_{(k)}$ using (16);
     update $\mathcal{M}_{(k)}$ using (17);
     update $S_{(k)}$ using (18);
   End while
  5) Generalization inverse:
  6) iterative regularization:
     if $k! = \text{iter}$ then
     $\mathcal{Y}_{(k+1)} = \eta \mathcal{X}_{(k)} + (1 - \eta) \mathcal{Y}_i$ $K = K + \lambda k$.
     end if
  End for

4. Experimental Results

In this section, this study provides examples to compare in simulation data and real data. The algorithm in comparison includes: FastHyDe[11], GLF[12], LRMR[20], SNLRSF[15], SSTV[8].

4.1. Simulated Data Experiments

We use the mean peak signal-to-noise ratio (MPSNR), the mean structural similarity index (MSSIM), and the mean of spectral angle mapping (MSAM) for comparison. During the experiment, we adopt two commonly used datasets to obtain the results, including Indian Pines dataset (of size $145 \times 145 \times 220$) and Pavia city center dataset (of size $256 \times 256 \times 80$). In the first step of processing data, each band is normalized to $[0, 1]$. Six situations of noise are as follows:

- Case 1: The white Gaussian is added in each band of dataset. The variance of zero-mean Gaussian noise is set as 0.12, 0.24, 0.36.
- Case 2: The white Gaussian is added in each band of dataset. The variance of zero-mean Gaussian noise is sampled from $[0.12, 0.24]$.
- Case 3: Based on Case 2, the salt&pepper noise is added in 30% band of HSI. The noise proportion of each band is randomly sampled from $[0.1, 0.2]$.
- Case 4: The Gaussian noise and Salt&pepper noise which are add in HSI are the same as Case 2 and Case 3, respectively. The stripe noise is added in 20% band of HSI. The number of stripe is randomly selected from set $[6, 7, \ldots, 15]$. The value of stripe is randomly sampled from $[0.6, 0.8]$. 
Case 5: The Gaussian noise and Salt&pepper noise which are added in HSI are the same as Case 2 and Case 3, respectively. The deadline noise is added in 20% of HSI. The weight of deadline is randomly selected from [1,2,3]. The number of deadline is randomly adopted from [6,7,...,10]. The value of deadline is set to 0.

Case 6: We adopt same situation with Case 2, Case 3, Case 4 and Case 5 about the Gaussian noise and salt and pepper noise, stripe, deadline, respectively.

Table 1. MPSNR of the Indian Pines dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRM | GLF | SNLRSF | SSTV | OURS |
|-------------|----------|-----|-----|--------|------|------|
| Case1-1     | 18.4109  | 35.6938 | 34.4413 | 35.7520  | 33.5534 | 31.6637 | 37.8962 |
| Case1-2     | 12.3952  | 34.2139 | 29.0327 | 34.2648  | 32.6633 | 26.6538 | 35.5969 |
| Case1-3     | 8.8769   | 32.7529 | 25.8692 | 32.7586  | 31.7690 | 23.5959 | 33.5480 |
| Case2       | 14.9784  | 35.0969 | 31.2359 | 35.1347  | 33.1315 | 28.9176 | 36.6135 |
| Case3       | 13.4632  | 32.5976 | 30.6768 | 32.6228  | 32.2734 | 28.5182 | 35.5154 |
| Case4       | 13.3995  | 31.8234 | 30.2277 | 30.2277  | 31.6413 | 28.2575 | 34.4367 |
| Case5       | 13.0825  | 30.6724 | 28.9565 | 30.8395  | 30.8974 | 27.5075 | 33.3629 |
| Case6       | 12.9965  | 29.8506 | 28.4853 | 30.0922  | 30.2570 | 27.3053 | 32.4198 |

Table 2. MSSIM of the Indian Pines dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRM | GLF | SNLRSF | SSTV | OURS |
|-------------|----------|-----|-----|--------|------|------|
| Case1-1     | 0.3185   | 0.9839 | 0.9033 | 0.9838  | 0.9737 | 0.8140 | 0.9913 |
| Case1-2     | 0.1627   | 0.9723 | 0.7594 | 0.9717  | 0.9606 | 0.6262 | 0.9846 |
| Case1-3     | 0.0965   | 0.9578 | 0.6335 | 0.9569  | 0.9401 | 0.5038 | 0.9754 |
| Case2       | 0.2259   | 0.9791 | 0.8312 | 0.9788  | 0.9687 | 0.7182 | 0.9879 |
| Case3       | 0.1886   | 0.9719 | 0.8155 | 0.9710  | 0.9652 | 0.7034 | 0.9858 |
| Case4       | 0.1862   | 0.9676 | 0.8126 | 0.9680  | 0.9634 | 0.7006 | 0.9838 |
| Case5       | 0.1828   | 0.9607 | 0.7956 | 0.9651  | 0.9620 | 0.6918 | 0.9793 |
| Case6       | 0.1804   | 0.9501 | 0.7846 | 0.9610  | 0.9597 | 0.6879 | 0.6879 |

Table 3. MSAM of the Indian Pines dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRM | GLF | SNLRSF | SSTV | OURS |
|-------------|----------|-----|-----|--------|------|------|
| Case1-1     | 13.5345  | 1.6371 | 1.8957 | 1.6133  | 2.1068 | 2.7996 | 1.2263 |
| Case1-2     | 25.4148  | 1.9122 | 3.4138 | 1.8601  | 2.2525 | 4.8372 | 1.5860 |
| Case1-3     | 35.1524  | 2.2444 | 4.8205 | 2.1621  | 2.4580 | 6.7916 | 2.0501 |
| Case2       | 20.2694  | 1.7586 | 2.7364 | 1.7217  | 2.1891 | 3.7873 | 1.4475 |
| Case3       | 24.3335  | 2.9419 | 2.9192 | 2.9309  | 2.3743 | 3.9464 | 1.6352 |
| Case4       | 24.3432  | 3.3063 | 3.2139 | 3.2890  | 2.7628 | 4.1125 | 2.1029 |
| Case5       | 25.6567  | 4.1113 | 4.5891 | 4.0733  | 2.9556 | 4.7359 | 2.2477 |
| Case6       | 25.8432  | 4.5758 | 4.8248 | 4.5238  | 3.3070 | 4.8166 | 2.6189 |

Figure 2. PSNR values of each band in terms of Indian Pines dataset.
4.2. Quantitative Comparison

In this paper, we utilize common noise evaluation indicators including MPSNR, MSSIM and MSAM. Table 1 - 6 show the indicators comparison between the art-of-state algorithms and the proposed GNS-ITboost algorithm. At the same time, we bolded the best value of the denoising effect to compare various algorithms. Through observation, it can be found that in the Indian dataset, the proposed GNS-ITboost algorithm performs the best. In the Pavia city center dataset, except for the MSA of Case1-1 and Case 1-3, the SNLRSF algorithm is slightly better than GNS-ITboost, and our algorithm is better than other algorithms. Figure 2 and Figure 3 show the comparison of denoising indexes for each spectral band. It can be found that in the Indian dataset, GNS-ITBoost is significantly better than other algorithms. In Figure 4 and Figure 5, the proposed algorithm is significantly better than the comparison algorithm by comparing FastHyDe[11], GLF[12], LRMR[20], SNLRSF[15], SSTV[8].

Table 4. MPSNR of the Pavia dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRMR | GLF | SNLRSF | SSTV | OURS  |
|-------------|----------|------|-----|--------|------|-------|
| Case1-1     | 18.4170  | 35.8581 | 31.4059 | 36.4556 | 36.5926 | 32.0236 | 36.8520 |
| Case1-2     | 12.3941  | 32.3903 | 26.6216 | 32.6339 | 32.8619 | 26.5258 | 33.1359 |
| Case1-3     | 8.8794   | 30.3859 | 23.3651 | 30.4100 | 30.8864 | 23.2396 | 30.7696 |
| Case2       | 14.5434  | 33.8127 | 28.2294 | 34.1741 | 34.3308 | 28.6533 | 34.7502 |
| Case3       | 13.6732  | 31.7485 | 27.9324 | 32.1734 | 32.9935 | 28.4043 | 33.0214 |
| Case4       | 13.4867  | 30.4459 | 27.5765 | 31.0117 | 32.0834 | 28.1034 | 32.4437 |
| Case5       | 13.4814  | 30.1451 | 27.3082 | 30.7595 | 31.9312 | 27.8482 | 32.0412 |
| Case6       | 13.3079  | 28.0263 | 26.3708 | 29.2893 | 30.4289 | 27.0998 | 30.8914 |

Table 5. MSSIM of the Pavia dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRMR | GLF | SNLRSF | SSTV | OURS  |
|-------------|----------|------|-----|--------|------|-------|
| Case1-1     | 0.3598   | 0.9615 | 0.8987 | 0.9667 | 0.9676 | 0.9039 | 0.9702 |
| Case1-2     | 0.1355   | 0.9224 | 0.7592 | 0.9258 | 0.9281 | 0.7471 | 0.9338 |
| Case1-3     | 0.0647   | 0.8840 | 0.6115 | 0.8823 | 0.8926 | 0.6034 | 0.8904 |
| Case2       | 0.2063   | 0.9413 | 0.8153 | 0.9463 | 0.9470 | 0.8207 | 0.9530 |
| Case3       | 0.1820   | 0.9226 | 0.8062 | 0.9295 | 0.9333 | 0.8137 | 0.9386 |
| Case4       | 0.1779   | 0.8991 | 0.7994 | 0.9151 | 0.9247 | 0.8085 | 0.9339 |
| Case5       | 0.1751   | 0.8968 | 0.7958 | 0.9138 | 0.9235 | 0.8040 | 0.9272 |
| Case6       | 0.1698   | 0.8216 | 0.7759 | 0.8869 | 0.9045 | 0.7898 | 0.9107 |

Table 6. MSAM of the Pavia dataset in terms of different algorithm.

| Noisy image | FastHyDe | LRMR | GLF | SNLRSF | SSTV | OURS  |
|-------------|----------|------|-----|--------|------|-------|
| Case1-1     | 35.3154  | 5.1323 | 10.5192 | 4.3072 | 4.0995 | 9.5071 | 4.2391 |
| Case1-2     | 51.9093  | 6.8278 | 16.1236 | 5.7398 | 5.3732 | 15.4882 | 5.2346 |
Case1-3  61.4750  8.1450  21.4497  6.8421  **6.0330**  20.4806  6.0865  
Case2  46.6548  6.0954  14.2758  5.1191  4.8377  12.8836  4.7859  
Case3  48.2487  10.0356  14.0524  8.5886  6.5505  12.7604  5.5897  
Case4  48.1143  11.1838  14.1975  9.4362  7.4448  13.7605  5.7081  
Case5  48.6405  10.0356  14.0524  8.5886  6.5505  12.7604  5.5897  
Case6  48.9051  15.9935  16.4428  12.1994  10.0665  15.2268  **8.7325**  

Figure 4. PSNR values of each band in terms of Pavia dataset.

Figure 5. SSIM values of each band in terms of Pavia dataset.

4.3. Visual Quality Comparison

Figure 6 and Figure 7 intuitively reflect the recovery of various algorithms under 8 noise situations. In the Indian dataset, we can observe that there are residual images by FastHyDe, LRMR, and SSTV. In the Pavia city center dataset, GLF and SNLRSF also have residual images in Case 4 and Case 6, but the proposed algorithm still does not have residual images. Our proposed algorithm can best approach to the original HSI.
Figure 6. Denoised results of different algorithms in terms of Indian Pines dataset.
4.4. Real Data Experiments
This study considers two commonly real datasets for denoising, namely Indian pine $145 \times 145 \times 220$ and EO-1 $1000 \times 400 \times 242$. We use FastHyDe[11], GLF[12], LRMR[20], SNLRSF[15], SSTV[8] to compare.

4.4.1. EO-1 Hyperion Dataset.
The size of EO-1 datasets is $1000 \times 400 \times 242$. We adopt sub-image ($400 \times 200 \times 166$) for experience. Figure 8 shows the original HSI and denoising HSI of the 132 of EO-1 which are destoried by Gaussian noise and stripe noise. All algorithms can provide effective results, but different algorithm provide different denoising level. Our algorithm can provide more details.
4.4.2. AVIRIS Indian Pines Dataset.
The size of Indian Pines is $145 \times 145 \times 220$. Some bands of dataset contain Gaussian noise, impulse noise. In Figure 9 provide false color image (R: 1, G: 80, B: 200). At the same time, we provide the denoising results by different algorithms. Due to non-convex low-rank method that can obtain better non-local low-rank factor and iterative regularization, we can obtain more information from noise image, and our algorithm can provide better visual quality and keep more details.

Figure 9. Denoised results of different algorithms in terms of Indian Pines dataset.

5. Conclusion
In this paper, we provide a method, GNS-ITboost that eliminates hyperspectral image noise. This study uses the global spectral low-rank method to reduce the amount of calculation and use the WSNM method to obtain the low-rank factor in the low-rank part. On the basis of obtaining the low-rank factor, in order to solve the optimization model, we adopt the PAM-based framework to optimize and strengthen its sparsity through the weighted sparsity method. In addition, we provide experiments on the corresponding comparison algorithms, including FastHyDe, GLF, LRMR, SNLRSF, SSTV to compare the denoising results.

References
[1] Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N., & Chanussot, J. (2013). Hyperspectral remote sensing data analysis and future challenges. IEEE Geoscience and Remote Sensing Magazine, 1(2), 6-36.
[2] Bioucas-Dias, J. M., & Nascimento, J. M. (2008). Hyperspectral subspace identification. IEEE Transactions on Geoscience and Remote Sensing, 46(8), 2435-2445.
[3] Stuart, M. B., McGonigle, A. J., & Willmott, J. R. (2019). Hyperspectral imaging in environmental monitoring: a review of recent developments and technological advances in compact field deployable systems. Sensors, 19(14), 3071.
[4] Ghamisi, P., Yokoya, N., Li, J., Liao, W., Liu, S., Plaza, J., ... & Plaza, A. (2017). Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. IEEE Geoscience and Remote Sensing Magazine, 5(4), 37-78.
[5] Zeng, S., Wang, Z., Gao, C., Kang, Z., & Feng, D. (2018). Hyperspectral image classification with global–local discriminant analysis and spatial–spectral context. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(12), 5005-5018.
[6] Manolakis, D., & Shaw, G. (2002). Detection algorithms for hyperspectral imaging applications. IEEE signal processing magazine, 19(1), 29-43.
[7] Venkatakrishnan, S. V., Bouman, C. A., & Wohlberg, B. (2013, December). Plug-and-play priors for model based reconstruction. In 2013 IEEE Global Conference on Signal and Information Processing (pp. 945-948). IEEE.
[8] Yuan, Q., Zhang, L., & Shen, H. (2012). Hyperspectral image denoising employing a spectral–spatial adaptive total variation model. IEEE Transactions on Geoscience and Remote Sensing, 50(10), 3660-3677.
[9] Chen G, Bui T D, Quach K G, et al. Denoising hyperspectral imagery using principal component analysis and block-matching 4D filtering[J]. Canadian Journal of Remote Sensing, 2014, 40(1): 60-66.
[10] Chen, G., & Qian, S. E. (2010). Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage. IEEE Transactions on Geoscience and remote sensing, 49(3), 973-980.

[11] Zhuang, L., & Bioucas-Dias, J. M. (2018). Fast hyperspectral image denoising and inpainting based on low-rank and sparse representations. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11(3), 730-742.

[12] Zhuang, L., & Bioucas-Dias, J. M. (2017, September). Hyperspectral image denoising based on global and non-local low-rank factorizations. In 2017 IEEE International Conference on Image Processing (ICIP) (pp. 1900-1904). IEEE.

[13] Zhang, H., He, W., Zhang, L., Shen, H., & Yuan, Q. (2013). Hyperspectral image restoration using low-rank matrix recovery. IEEE transactions on geoscience and remote sensing, 52(8), 4729-4743.

[14] Xie, Y., Qu, Y., Tao, D., Wu, W., Yuan, Q., & Zhang, W. (2016). Hyperspectral image restoration via iteratively regularized weighted schatten p-norm minimization. IEEE Transactions on Geoscience and Remote Sensing, 54(8), 4642-4659.

[15] Cao, C., Yu, J., Zhou, C., Hu, K., Xiao, F., & Gao, X. (2019). Hyperspectral image denoising via subspace-based nonlocal low-rank and sparse factorization. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12(3), 973-988.

[16] He, W., Yao, Q., Li, C., Yokoya, N., & Zhao, Q. (2019, June). Non-local meets global: An integrated paradigm for hyperspectral denoising. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 6861-6870). IEEE.

[17] Xie, Y., Gu, S., Liu, Y., Zuo, W., Zhang, W., & Zhang, L. (2016). Weighted Schatten $p$-norm minimization for image denoising and background subtraction. IEEE transactions on image processing, 25(10), 4842-4857.

[18] Attouch, H., Bolte, J., & Svaiter, B. F. (2013). Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods. Mathematical Programming, 137(1-2), 91-129.

[19] Oh, T. H., Matsushita, Y., Tai, Y. W., & So Kweon, I. (2015). Fast randomized singular value thresholding for nuclear norm minimization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4484-4493).

[20] Zhang, H., He, W., Zhang, L., Shen, H., & Yuan, Q. (2013). Hyperspectral image restoration using low-rank matrix recovery. IEEE transactions on geoscience and remote sensing, 52(8), 4729-4743.