



ARTICLE

Computationally Efficient Gradient-Aware Hyperspectral Image Denoising Using Center-Difference Convolutional Networks

Mahmood Ashraf^{1,2}, Nuha Zamzami³, Shtwai Alsubai⁴, Raed Alharthi⁵, Muhammad Umer^{6,*},
Yunyoung Nam⁷ and Yongwon Cho^{7,*}

¹Department of Computer Science and Information Technology, University of Kamalia, Kamalia, Pakistan

²Department of Communication and Cyber Security, Bahuddin Zakariya University, Multan, Pakistan

³Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia

⁴Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia

⁵Department of Computer Science and Engineering, University of Hafr Al-Batin, Hafar Al-Batin, Riyadh, Saudi Arabia

⁶Department of Computer Science & Information Technology, The Islamia University of Bahawalpur, Bahawalpur, Pakistan

⁷Department of Computer Science and Engineering, Soonchunhyang University, Asan, Republic of Korea

*Corresponding Authors: Muhammad Umer. Email: umersabir1996@gmail.com; Yongwon Cho. Email: dragon1won@sch.ac.kr

Received: 07 January 2026; Accepted: 27 April 2026; Published: 30 June 2026

ABSTRACT: Hyperspectral image (HSI) denoising is a crucial preprocessing step that significantly enhances the performance of downstream applications, such as object detection and classification. Whereas deep neural networks have achieved remarkable performance in HSI denoising, many existing models rely mostly on vanilla convolutions, which often fail to capture fine-grained noise patterns and structural details in real-time HSIs. To address these limitations, we propose a novel Center-Difference Convolutional Network (CDCN) designed to effectively suppress various noise types while preserving the inherent structure of HSIs. By leveraging center-difference convolution (CDC), our model captures both gradient and intensity information in the spatial domain, enabling better discrimination of subtle noise characteristics. The CDCN architecture processes 3D HSI cubes through separable 3D convolutions, efficiently extracting spatial-spectral features with minimal computational overhead. Additionally, a spatial-spectral attention mechanism is integrated to further refine feature representation. We evaluate the proposed method on one simulated dataset (Kennedy Space Center) and two real-world datasets (Pavia Center and Houston-2018). Experimental results demonstrate that CDCN consistently outperforms existing state-of-the-art approaches, achieving superior denoising performance while maintaining spectral-spatial information. Ablation studies also validate the effectiveness of CDC and attention mechanisms in enhancing denoising capability over standard convolutional baselines.

KEYWORDS: Attention mechanism; center difference network; image denoising; real-time image processing; hyperspectral imaging; remote sensing; edge-preserving filtering

1 Introduction

Hyperspectral images (HSIs) contain high spectral resolution features, widely employed in various remote sensing applications, including object recognition, unmixing, and classification [1,2]. Due to photon effects, sensor malfunctions, and atmospheric impacts, HSIs often exhibit noise, including Gaussian noise, stripe noise, random noise, and dead pixels [3–5]. Such noises can dramatically affect the interpretation of information. Hence, it is crucial to implement a pre-processing denoising step before analyzing HSI.

In recent decades, various techniques have been implemented to reduce noise from HSI. Most methods were initially designed for red-green-blue (RGB) or grayscale images, ignoring the spectral dimension, relying upon band-wise modeling, such as nonlocal self-similarity (NLSS) methods, block matching 3D filtering [6], and weighted nuclear norm [7] (WNNM). These techniques treat each band like a 2D image, thus distorting the spectral information. Although many researchers have recently utilized combined spatial-spectral information to remove noise from HSI, these optimization-based methods require tuning parameters for each HSI [8], which makes the denoising operation time-consuming and limits its applicability in real-world operating environments. Besides, these strategies could produce spectral distortion in complex scenes [9,10]. Consequently, research is focusing on models that can handle different kinds of HS data and perform well in complex scenarios. HSI-DeNet [11] demonstrated the effectiveness of convolutional networks for learning spatial-spectral features, while residual CNN-based methods further improved representational capability [12]. A trainable sparse coding model was introduced in [13], and deep spatial-spectral representation learning was explored in [14]. Global reasoning-based networks [15] and hybrid noise modeling approaches [16] enhanced performance under complex noise conditions. In addition, quasi-recurrent architectures have shown strong capability in modeling spectral dependencies [17]. Most of these methods process the spatial-spectral information of HSIs using different technologies, such as 2D convolutions [11], 3D convolutions (i.e., convolutions performed simultaneously on both spatial and spectral dimensions), and hybrid models with both 2D and 3D filters [16]. However, the HSI datasets are ecologically sensitive, resulting in fine-grained and complex noise. Although vanilla convolutional neural network (CNN)-based approaches can provide deep semantic features, the related features cannot generally provide fine-grained information, and therefore, they are less sensitive to HSI noise.

To overcome the limitation of vanilla convolution for HSI denoising, we propose a CDC-based architecture. The CDC exploits gradient features through a center difference strategy, enabling a sensitive awareness of noise, which mitigates the limitation of vanilla convolution (which is unable to extract fine-grain noise). More specifically, a novel CDC network (CDCN) is proposed to exploit spatial-spectral information for HSI denoising, which covers the fine-grain information and extracts intrinsic information (e.g., noise) more effectively than a vanilla CNN in complex environments.

The contribution of this paper is three-fold:

1. To accurately capture noise patterns, the CDCN is proposed, which completely learns the mapping between the noisy image and the clean image using the CDC strategy. CDCN can achieve better noise removal than vanilla convolution by simultaneously extracting the gradient and intensity features in the CDC. Focusing more attention on the center of the image, the CDC also contributes to denoising performance because the center contains more information.
2. To improve the feature extraction capabilities of the CDCN, separable convolutions are utilized to extract spatial and spectral features so that it can leverage on both spectral and spatial features for denoising. Based on separable convolutions, a feature learning module is designed that concurrently learns spatial and spectral features.
3. To explore the most relevant features for denoising, a spatial-spectral attention module is adopted in the CDCN.

The remainder of the paper is organized as follows. [Section 2](#) describes the related works for HSI denoising. [Section 3](#) outlines the proposed methodology. [Section 4](#) presents a preliminary evaluation of the proposed methodology, while the complete experimental results of the proposed and baseline models are discussed in [Section 5](#). In [Section 6](#), a complete discussion is being made why the proposed method performs better than the rest of the models, and finally conclusions with future work directions are made in [Section 7](#).

2 Related Works

To solve the HSI denoising problem, various methods have been proposed, which can be broadly categorized into two classes: classical and deep learning methods.

2.1 Classical Methods

Classic methods can be categorized into spatial and transform domain methods.

2.1.1 Spatial-Domain Methods

These approaches remove noise in the spatial domain, better preserving spatial and spectral information. These methods generally work on reasonable priors or assumptions. For spatial-domain approaches, the noisy HSI is mapped to a clean one. For instance, non-local [18–20] total variation [21], low-rank models [22–25], and sparse representation [26,27] techniques belong to spatial-domain methods. Yuan et al. [21] presented a spatial-spectral adaptive total variation method to minimize the noise from HS images. Chen et al. [9] proposed block-matching 4D filtering (BM4D) for HSI denoising. Similarly, a sparse-based representation technique has been proposed by Li et al. [27] to reduce noise in HSIs, utilizing both inter-band and intra-band structures in a spatial-spectral distributed sparse representation strategy. The low-rank tensor approximation model (LRTAM) proposed by Renard et al. [28] is also a typical spatial-domain method. LRTAM performs low-rank approximation on the spatial dimension after spectral dimensionality reduction. Similarly, Zhang et al. [23] proposed a low-rank matrix recovery (LRMR) algorithm for HSI denoising. The spatial-spectral total-variation regularized low-rank tensor factorization (SSTV-LRTF) has been introduced by Fan et al. [29] to minimize the mixture of noises from HSI. Xie and Li [10] introduced a non-convex low-ranked regularizer known as weighted-Schatten p -norm. Zhao and Yang [30] introduced a sparse spectral domain coding to capture local and global redundancy and correlation (RAC). Latexier and Bourennane [31] exploited the multidimensional Wiener filter (MWF) for HSI denoising.

2.1.2 Transform-Domain Methods

Transform-domain methods separate the clean data from the noisy ones in the transformed domain. Different transformation techniques, such as principal component analysis (PCA), the wavelet transform, and the Fourier transform, are used to transform the data into another space, where clean data are separated from noise. For example, a hybrid denoising model for spatial and spectral domains (HSSNR) has been proposed by Othman and Qian [32] utilizing a wavelet shrinkage technique for noise reduction. Atkinson et al. [33] introduced an estimation-based strategy to get the HSI in the noisy observation using the discrete Fourier transform (DFT) and wavelets to overcome the shortcomings of the Wiener filter. A method based on the first-order roughness penalty (FORP) has been presented by Rasti et al. [34]. These techniques have several primary functions, including feature representation of the noisy HSI and its mapping to the clean one. However, parameters in these methods have to be tuned for each HSI to achieve good efficiency of the underlying models. Besides, these approaches show sensitivity and instability for several bands of HSIs. Ashraf et al. [35] proposed MAOTformer to remove the noise from HSI.

However, both spatial and transform-domain methods have some limitations. Spatial domain methods may struggle to consider variations in the geometrical structure and relationships, causing a performance reduction for HSI denoising. Instead, transform-domain methods often require a deep tuning of parameters, even exhibiting sensitivity to some specific HS bands. Moreover, classical methods are limited by the engineering design in the feature extraction phase and usually work thanks to the design of some priors and/or assumptions that may be unsuitable in certain cases.

2.2 Deep Learning-Based Methods

Deep learning-based methods have recently demonstrated their effectiveness for various tasks, such as HSI classification [36,37], caption generation [38], and sharpening [39,40]. For image denoising, early works [41] exploited CNNs for feature extraction. The performance of these methods is considered sufficient for natural images. However, there is still room for improvement in the case of HSI (thanks to the proper consideration of the crucial spectral dimension of HSIs).

Recent research has exploited spatial and spectral information. The well-known 3D-DnCNN [41] has been utilized for HSI denoising, which enables consideration of three adjacent bands simultaneously with a 3D CNN. Although this method produces good results by retaining the spatial-spectral correlation among adjacent bands, the high number of HS spectral bands represents a significant limitation for this approach. Maffei et al. [42] claimed that traditional frameworks could not handle the high correlations among adjacent bands in a high spectral dimensionality. As a result, low-quality denoising is performed for HSIs. Leveraging on rich information in the spectral dimension, Yuan et al. [12] proposed a residual technique for HSIs to remove noise, where spatial and spectral information is simultaneously fed to the network. Moreover, multilevel representation and multiscale features are used for the final restoration. Dong et al. [14] introduced a spatial-spectral strategy to minimize the noise from HSIs. They introduced a modified 3D U-Net with separable filters, making the method computationally efficient. Zhang et al. [16] presented a gradient network to remove the hybrid noise from spatial-spectral pixels of HSIs. Furthermore, Cao et al. [15] introduced a deep spatial-spectral global reasoning framework for HSI denoising.

To explore long-range dependency for HSI denoising, many attention mechanisms have been integrated into HSI denoising networks. Kan et al. [43] addressed the high-frequency features of HSIs. They claimed that such features have more noise and introduced an attention-based octave network to reduce the noise from such features of HSIs. Wang et al. [44] presented a cross-attention-based network to minimize the noise from HSIs. This network used the attention module with a group of convolutions to extract the features by focusing on the most relevant bands. A CNN and a transformer-based strategy have been introduced by Gong et al. [45] to remove the noise in HSIs. Murugesan et al. [46] introduced an attention-based U-Net to extract the noise features. A mixed attention network for HSI denoising is proposed by Lai and Fu [47], who utilized a multi-head recurrent spectral attention mechanism to integrate inter-spectral features across all spectral bands. Although different technologies, such as spectral-spatial feature extraction and attention, have been adopted for HSI denoising, almost all the proposed networks are based on vanilla convolution. However, vanilla convolution cannot exploit high-frequency features, thus limiting its capability to extract fine-grained features and complex noise. On the other hand, the proposed network based on CDC is sensitive to high-frequency features by the gradient information, enabling the ability to extract fine-grained features and complex noise. Duan et al. [48] proposed a linear attention Mamba (LaMamba) to remove the noise from the HSI 3D selective scan mechanism, which was designed to obtain the spatial spectral continuous sequences with the help of six bidirectional scan orders. Similarly, Nachimuthu et al. [49] introduced a novel SqueezeNet-based denoising framework that leverages Fire modules for efficient feature extraction with fewer parameters.

3 Methodology

The proposed model is designed to reduce noise from HSIs by utilizing spatial and spectral information. We introduce a CDCN inspired by a center difference strategy [50] applied to the vanilla convolutional network for better generalization and representation. CDC is used to provide better results in extracting local structural properties. In contrast to the conventional convolution, CDC adds another term of center-of-differentiate, which directly represents the difference in intensity between the central pixel and the pixel

differences. Such an operation improves the description of local gradient information, which is especially useful in differentiating between structural features and noise. Noise in HSI can be difficult to notice since it exists as subtle variations within spatial regions and spectral bands, and even fine spatial structures can easily be destroyed in the process of denoising. CDC focuses on the local intensity differences to enable the network to capture spatial discontinuities like edges and textures and reduce noise elements. Unlike the initial structure of the CDCN, which uses the CDC as a part of a traditional 2D convolutional network, the proposed architecture utilizes the CDC as a redesigned feature extractor together with separable convolution layers, skip connections, and spatial-spectral attention to utilize the accurate spatial-spectral properties of HSI.

3.1 Problem Formulation

In Eq. (1), given a 3D cube, \tilde{X} , related to an HSI having a size of $W \times H \times B$, where H , W , and B represent the height, width, and number of spectral bands, respectively. The corresponding noisy HSI is represented as \tilde{X} and is modeled as

$$\tilde{X} = X + A, \quad (1)$$

where X is the clean image and A is an additive noise as $A = [A_1, A_2, \dots, A_B]$. The objective of HSI denoising is to reduce the additive noise in the noisy data, \tilde{X} , and extract the clean image, X , from the noisy observation. It is better to train a network with the different additive noises usually present in real-world HS images. For this purpose, an environment with different additive noises (EDAN) is established to obtain the noisy \tilde{X} data using $A_n = A_n(0, \sigma)$, which follows a normal distribution with variance σ , and zero-mean, where n is the band index. This allows for a noise distribution that is independent and identically distributed, introducing various controlled intensities of noise. Moreover, the effects of many random and uncontrolled processes are also simulated, which are present in real remote sensing images.

3.2 Proposed CDCN

As shown in Fig. 1, our CDCN consists of four parts: 1) spatial-spectral feature extraction module; 2) attention module; 3) CDC-based denoising block module; and 4) fully connection layer. First, the proposed CDCN takes as input the simulated noisy bands which is represented by n -th (for spatial information) and adjacent bands (for spectral information) to obtain spatial and spectral features. Adjacent bands help to enhance the quality of the image and increase the correlation among the different HS bands for subsequent analysis. Afterwards, spatial and spectral features are concatenated and passed to the attention mechanism to enhance the extracted features and to exploit spectral-spatial dependency for HSI denoising. Thus, the attention-enhanced spatial-spectral features are ready for the denoising module, where nine blocks based on CDC are employed to perform the denoising operation. Furthermore, the outputs of all the blocks are concatenated and enhanced again by the attention module. Finally, these enhanced features are passed to the fully connected layer for predicting noise, φ . The output X is obtained from the network by subtracting the noise from the input noisy image, \tilde{X} . The model is represented in Eq. (1).

3.3 Spatial-Spectral Feature Extraction Module

HS images contain abundant spectral information, which can be used to improve the denoising process. More specifically, HS images have high similarity and correlation in the textural characteristics and surface features. Accordingly, to properly utilize the high correlation, the proposed method takes a band with its adjacent bands as input. The details of spectral bands formulation are shared from Eqs. (2) to (6).

Let the n -th band be denoted as $X_n \in \mathbb{R}^{H \times W}$. Its adjacent spectral bands are defined as

$$B_n = \{X_{n-k}, \dots, X_n, \dots, X_{n+k}\}, \quad (2)$$

where $K = 2k + 1$ represents the total number of adjacent bands used. The stacked spectral cube is therefore written as

$$X_n \in \mathbb{R}^{H \times W \times K}. \quad (3)$$

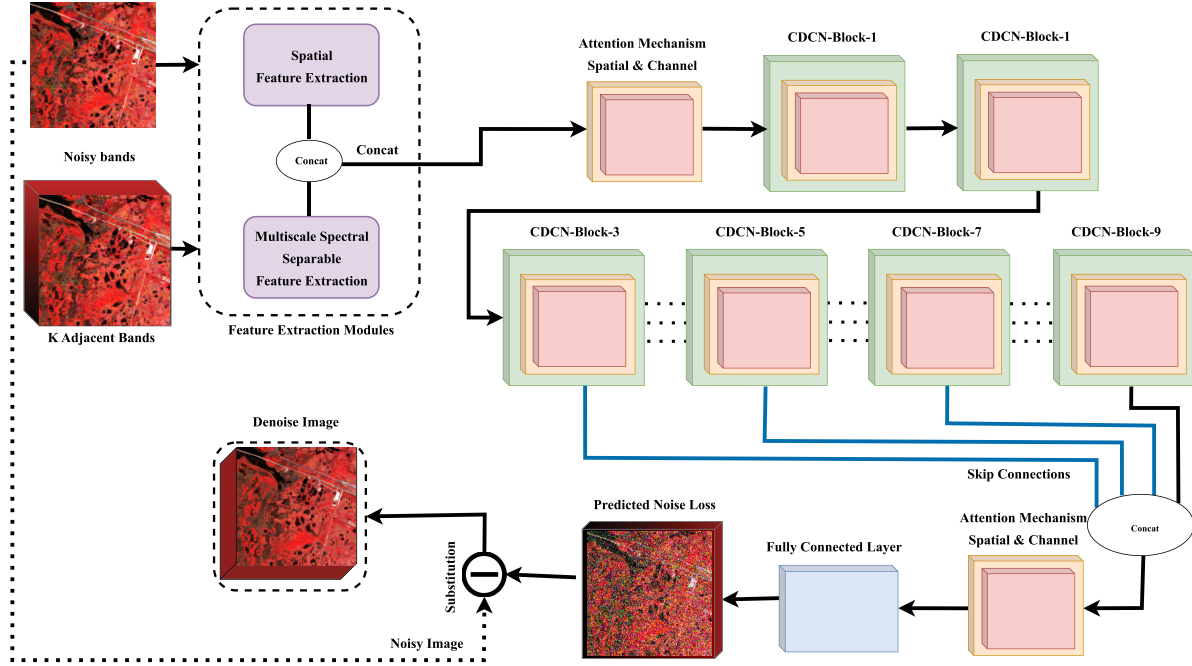


Figure 1: Flow diagram of the CDCN architecture for noise removal from HSIs. Concat is the concatenation operator.

More specifically, as shown in Fig. 2, this module consists of two branches to extract the spectral and spatial features using different convolution kernels. The upper branch of the module uses vanilla 2D convolutions to extract the spatial information of the current band by using the 2D convolution, which is defined as The spatial branch applies 2D convolution defined as

$$F_{i,j,c}^{2D} = \sum_{u=-r}^r \sum_{v=-r}^r W_{u,v,c}^{2D} \cdot X_n(i+u, j+v), \quad (4)$$

where W^{2D} denotes the convolution kernel, r is determined by kernel size, and c is the output channel index.

Similarly, the lower branch extracts the spatial-spectral information from a 3D cube. To reduce the computational complexity of 3D convolution, a separable convolution strategy is utilized to achieve 3D convolution. Besides, separable 3D convolution can better deal with the structural dissimilarity between spectral and inter-spatial features than vanilla 3D convolutions. That is because the vanilla 3D kernels cannot manage different structural information in a proper way. Moreover, the separable 3D convolution can reduce the number of parameters to stabilize the training of the network. In separable 3D convolution, the 1-D kernel extracts the spectral information, while the 2D kernel focuses on the spatial dimension of the HSI. So, for spectral-spatial extraction, separable 3D convolution is adopted. First, spectral convolution is applied as

$$F_{i,j}^{spec} = \sum_{t=-k}^k W_t^{spec} \cdot \mathcal{X}_n(i, j, n + t), \quad (5)$$

followed by spatial convolution

$$F_{i,j,c}^{3D} = \sum_{u=-r}^r \sum_{v=-r}^r W_{u,v,c}^{spat} \cdot F^{spec}(i + u, j + v), \quad (6)$$

where W^{spec} and W^{spat} denote spectral and spatial kernels, respectively.

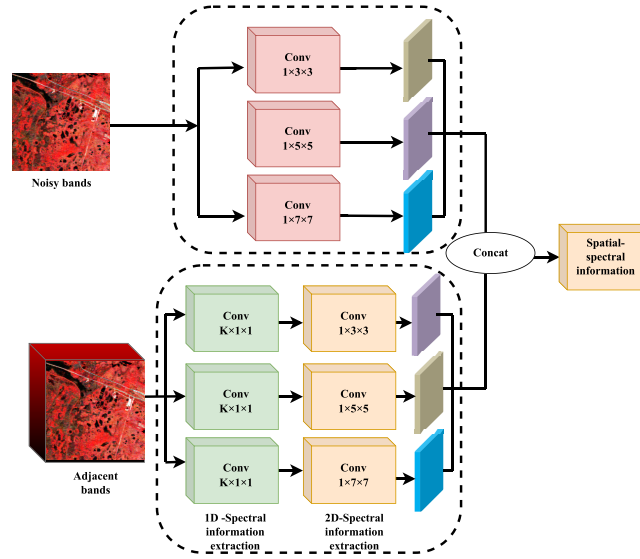


Figure 2: Spatial-spectral extraction module. Conv indicates the convolution operation, Concat is the concatenation operator, and K represents the number of adjacent bands.

After that, the extracted spectral and spatial information is then concatenated and fed into the attention mechanism to acquire enhanced features.

3.4 Attention Module

An attention module is proposed that contains channel attention and spatial attention to improve the feature representation. Specifically, the channel attention module helps the network emphasize informative spectral channels while suppressing less relevant ones, which is particularly important in hyperspectral imagery where different spectral bands contribute differently to the denoising process. Meanwhile, the spatial attention module focuses on significant spatial regions and improves the representation of important structural details such as edges and textures. These mechanisms enable the network to effectively capture both spectral dependencies and spatial structures, thereby improving the overall denoising performance, which is presented in Fig. 3. The first part learns the channel attention maps, C_M , in the channel dimension. In the same way, the spatial attention part learns the feature maps, S_M , in the spatial dimension and refines them. This part of attention is shared in Eq. (7).

$$\begin{aligned} F_1 &= C_M(F) \otimes F, \\ F_2 &= S_M(F_1) \otimes F_1, \end{aligned} \quad (7)$$

where \otimes is the element-wise multiplication, the input features are presented by F , F_1 is the tuned features by the attention channel, and the output features are presented by F_2 . The attention module has been widely used to enhance representations [51] and its ability has also been proven in increasing spectral and spatial feature representations.

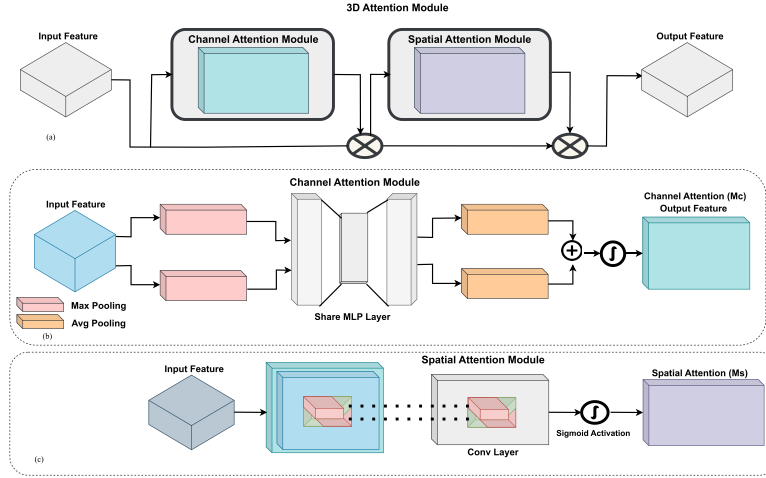


Figure 3: (a) Attention mechanism, (b) illustration of the channel attention module, and (c) illustration of the spatial attention module.

3.4.1 Channel Attention

This module combines feature maps concerning the spatial domain by utilizing the average and max-pooling layers. Features are forwarded by the attention module via a shared multi-layer perceptron (MLP), which has only a hidden layer, with 16 input channels. Channel attention maps, C_M , are obtained by merging the outputs of two branches, as follows in Eq. (8):

$$C_M(F) = \sigma(P_1(P_0(F_{\text{avg}}^C)) + P_1(P_0(F_{\text{max}}^C))), \quad (8)$$

where σ is used as an activation function, P_1 and P_0 are the parameters of the MLP, F_{avg}^C and F_{max}^C are the features generated in the spatial domain by the operations of the average and max-pooling layers, and C represents the size of the input channel.

3.4.2 Spatial Attention

In this module, the attention mechanism focuses on the spatial domain. Features maps F_{avg}^S and F_{max}^S are generated by the average and max-pooling layers. These features are concatenated and then sent to a CNN. The process of the spatial attention module is described in Eq. (9):

$$S_M(F) = \sigma(\text{Conv}[F_{\text{avg}}^S; F_{\text{max}}^S]), \quad (9)$$

where σ is used as an activation function and Conv is the 2D convolution with a kernel size of 7×7 . This module refines the spatial and spectral features, whereas the spectral feature module only extracts the related features. We applied the attention mechanism after the concatenation operation. In the proposed network, we used attention modules at two stages, before and after the architecture, to refine the representation and enhance the network's overall efficiency. The first attention helps to enhance the discriminative and contextually meaningful information. In contrast, the attention used after the CDCN refines the feature

representations retrieved from the model and selectively enhances important features by suppressing the less significant ones. This contributes to improving the performance of the overall network.

3.5 Denoising Module

The center of the image contains important information as claimed by [50]. This phenomenon is particularly remarkable for HSIs. Some bands at the beginning and at the end of the HSI are corrupted by noise, as mentioned in [12,42,52]. Therefore, we adopted a center difference strategy in our denoising module to reduce the noise from the HSI. CDC (center-difference convolution) modifies the standard convolution by incorporating the intensity difference between the central pixel and its neighboring pixels and makes the denoising operation efficient by giving more attention to the important features of the image. Our model removes the noise from HSIs better than vanilla convolution as discussed in Section 4. As far as we know, the proposed architecture is the first one trying to apply the center difference convolution as a denoising model for HSIs.

For an input feature map F , the center-difference convolution at spatial location p_0 is defined as in Eq. (10).

$$F_{CDC}(p_0) = \sum_{p_n \in \mathcal{R}} W(p_n) F(p_0 + p_n) - \theta F(p_0) \sum_{p_n \in \mathcal{R}} W(p_n), \quad (10)$$

where \mathcal{R} denotes the local receptive field, $W(p_n)$ represents the convolution kernel weight at position p_n , and $\theta \in [0, 1]$ controls the trade-off between gradient and intensity features. A higher value of θ gives more importance to the gradient features.

Convolution operations are performed on the center of the image by setting the kernel size to $k/2$. Fig. 4 shows each block is composed of average pooling, CDC, max pooling, leaky ReLU, two kinds of attention layers, and a vanilla convolution layer. The center difference is calculated by (14) and is shown in Fig. 5. Our denoising module is designed in a block-wise pattern as presented in

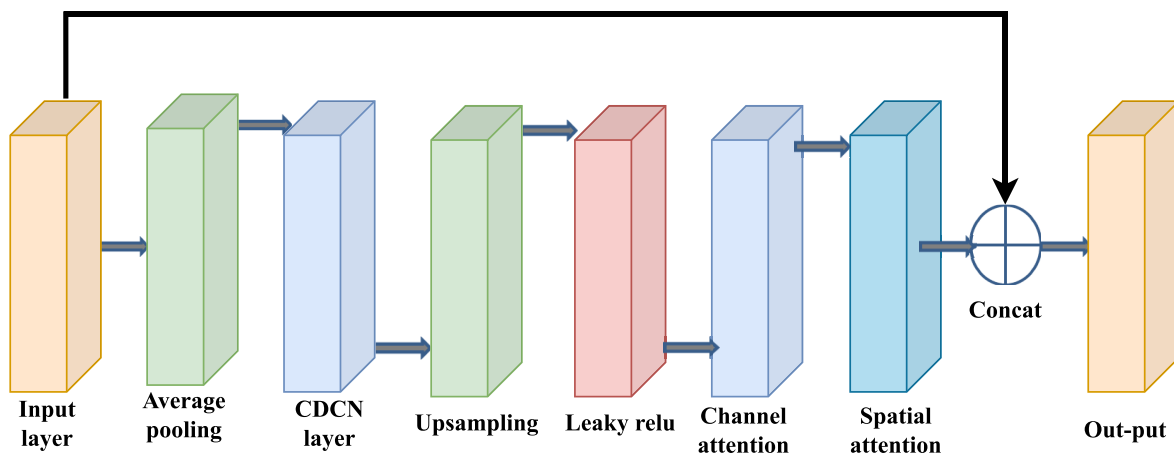


Figure 4: Composition of the CDCN block. Concat is the concatenation operator.

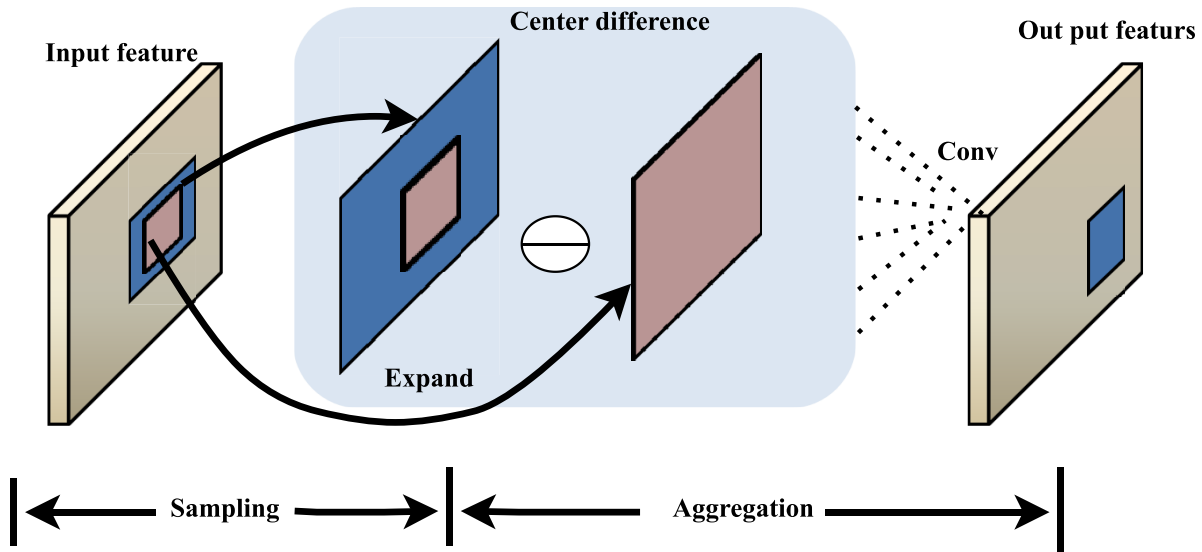


Figure 5: Center difference convolution operation. Conv indicates the convolution operation.

The denoising module shown in Fig. 1 consists of several blocks. Each block extracts the noisy features using a U-Net-like structure, as shown in Fig. 4. The downsampling operation is followed by the CDC scheme. The downsampling is performed by the average pooling layer, which reshapes the HSI, $\tilde{X}_n \in \mathbb{R}^{H \times W \times B}$, without dropping the original information. Thus, we obtain a larger receptive field while reducing the need for memory and computation. Then, CDC is conducted on the downsampled features to extract the noisy features using the center-based property defined in (14). Afterwards, upsampling is applied to these features to inverse the downsampling and retain the spatial dimension of the input images. Before the concatenation operation, features are passed to the attention mechanism. This mechanism improves the discriminative ability to remove the noisy and irrelevant features from HSIs, and all the outputs are concatenated with the input. Hence, the denoising operation of one block is completed, and the output of the current block is then sent to the next block for further processing.

3.6 Multi-Stage Feature Representation for Reconstruction

Different levels of features are indirectly connected to the layers at various depths layers. To effectively exploit these hierarchical features without making the direct attenuation, it is better to combine these several feature maps with the final denoising [53]. Consequently, as given in Fig. 1, features of multiple CDC blocks are concatenated to obtain a multi-stage feature representation in our model.

These multi-stage representations can be considered as multilevel skip connections [54], effectively overcoming the vanishing gradient problem [41]. The concatenation operation is shared in Eq. (11).

$$f_{\text{con}} = \text{Concat}\{f_{b3}, f_{b5}, f_{b7}, f_{b9}\}, \quad (11)$$

where $f_{b3}, f_{b5}, f_{b7}, f_{b9}$ are the multilevel feature representation from different CDC blocks, Concat is the concatenation operator, and f_{con} is the concatenation of the features from all the CDC blocks, then passed to a fully connected layer (FCL). As a result, the noise is predicted.

3.7 Architecture Details

For reproducibility, we provide explicit architectural specifications of the proposed network. The spatial feature extraction module employs three parallel convolution layers with kernel sizes of 3×3 , 5×5 , and 7×7 , respectively. Each branch produces 20 feature channels. The spectral feature extraction module utilizes 3D convolution with kernel size $(K, 1, 1)$, where $K = 24$ represents the number of adjacent spectral bands, followed by a $(1, 3, 3)$ convolution to capture joint spectral-spatial dependencies. The CDC-based denoising blocks use a 3×3 convolution kernel with padding 1 and stride 1. The number of output feature channels in each CDC block is 80. The hyperparameter θ controlling the center-difference contribution is set to 0.8.

Downsampling is performed using 2×2 average pooling with a stride of 2, while upsampling uses nearest-neighbor interpolation with a scale factor of 2. Each CDC layer is followed by a Leaky ReLU activation with a negative slope of 0.2. The channel attention module applies a reduction ratio of 16, and spatial attention uses a 7×7 convolution kernel.

Multi-level skip connections are introduced between the encoder and decoder stages to preserve spatial information and alleviate the vanishing gradient problem.

4 Preliminary

Vanilla Convolution: The basic operation in CNN-based architectures is the 2D spatial feature extraction. Here we present a short review of vanilla convolution. It contains two steps for feature extraction. In the first stage, the sampling is performed from the input features map, x . In the second stage, aggregation is performed on the sampled values through a weighted sum ($\sum w$). As a result of these two operations, the feature map, y , is obtained as output. We can formulate the equation of the above operations in Eq. (12).

$$y(l_c) = \sum_{l_s \in G} w(l_s) x(l_c + l_s), \quad (12)$$

where l_c is the current location of feature maps on input and output, and l_s is the position in the local receptive field, G .

Center Difference Convolution (CDC): A central difference technique is integrated into the convolution to improve the generalization and representation of the vanilla convolution. The CDC also contains two stages similar to that in the vanilla convolution, which are sampling and aggregation, as shown in Fig. 5. The sampling stage is the same as for the vanilla convolution, whereas the aggregation stage is different. The CDC network gives more attention to the center of the image and aggregates the center-oriented values by a difference strategy. Hence, (12) becomes:

$$y(l_c) = \sum_{l_s \in G} w(l_s) (x(l_c + l_s) - x(l_c)), \quad (13)$$

where if $l_c = (0, 0)$ (center location), the value of the gradient is always zero. Detailed information at gradient level is essential to differentiate the original and the noisy images. This shows that applying the center difference strategy to vanilla convolution is beneficial to noise removal. Combining the different strategy and the vanilla convolution, we can formulate the equation for CDC. Hence, we have:

$$y(l_c) = \theta \sum_{l_s \in G} w(l_s) x(l_c + l_s) - x(l_c) + (1 - \theta) \sum_{l_s \in G} w(l_s) x(l_c + l_s), \quad (14)$$

where θ is a hyper-parameter representing the trade-off between the gradient and the intensity information and gradient-based center difference features. In our case, the value of θ is empirically set to 0.7 for all the experiments, which offers a balanced contribution of spatial intensity and gradient information and causes stable denoising results. A higher θ value gives more importance to the center gradient information. Borrowing it from [50], we make this model feasible for HSI denoising.

4.1 Network Computational Complexity Analysis

Let the input feature map be of size $H \times W$ with C_{in} input channels and C_{out} output channels, and the convolution kernel size be $k \times k$.

For a vanilla convolution layer, the computational complexity is

$$\mathcal{O}(HWC_{in}C_{out}k^2). \quad (15)$$

The proposed center-difference convolution (CDC) introduces an additional subtraction operation inside the aggregation stage. However, this operation is performed within the same receptive field and does not introduce extra nested loops over spatial or channel dimensions.

Therefore, the computational complexity of CDC remains

$$\mathcal{O}(HWC_{in}C_{out}k^2), \quad (16)$$

which is the same order as vanilla convolution. The difference lies only in a constant factor due to the additional center-difference computation.

Since our denoising module stacks L CDC-based blocks, the overall complexity of the denoising module becomes

$$\mathcal{O}(LHWC_{in}C_{out}k^2). \quad (17)$$

Thus, the proposed model maintains the same asymptotic complexity as standard CNN-based architectures while providing improved denoising capability.

4.2 Implementation Details

4.2.1 Data Preparation

Data are normalized before feeding them into the network to make the information more appropriate. HS images have spatial and spectral dimensions, containing redundant spectral information, which cannot be ignored for denoising. Therefore, we train our model in an end-to-end fashion by employing a 3D spatial-spectral cube to solve the noise problem, thus benefiting from the high number of spectral bands. More specifically, we made a spectral-based data cube, $X_n \in R^{W \times H \times B}$, corresponding to the n -th band from the original image, where W denotes the width, H represents the height, and B is the number of spectral bands of the cube.

4.2.2 Training Details

After building the framework for HSI denoising, a loss function has been defined for the whole network. The ℓ_2 loss function is widely applied to train networks, but the generated results are usually over-smoothed. Hence, we used the ℓ_1 loss function to overcome the over-smoothing problem:

$$L_1 = \frac{1}{P} \sum_{i=1}^P \|X - X^{\text{gt}}\|_1, \quad (18)$$

where P represents the number of patches for training, X is the output image, $\|\cdot\|_1$ is the ℓ_1 norm, and X^{gt} refers to the noise-free (ground-truth) image. Each kernel size for CDC is represented in Fig. 2. We set the patch size to 20×20 , and the value of K has been set to 24. The learning rate has been fixed to 0.0001 using the Adam optimizer. 100 epochs have been used to train the network (exploiting the same setting for all the networks). were run in the Google Colab, using an NVIDIA Tesla T4 GPU with 16 GB GPU memory. PyTorch has been considered as the framework and all the experiments. This configuration was used for both training and testing of all the methods.

5 Experimental Results

To test the effectiveness of our CDCN model in removing noise from HSIs, we performed experiments on both simulated and real data. The related images are shown in Fig. 6. The performance of the proposed network is compared with existing solutions, i.e., LRMR [23], DnCNN [41], MemNet [55], DeNet [11], GradNet [56], ENCAM [57], HSIDwRD [58], and MAN [47], SST [59] SERT [60] and SSIT [61]. Qualitative and quantitative experiments have been provided using different quality metrics for performance assessment.

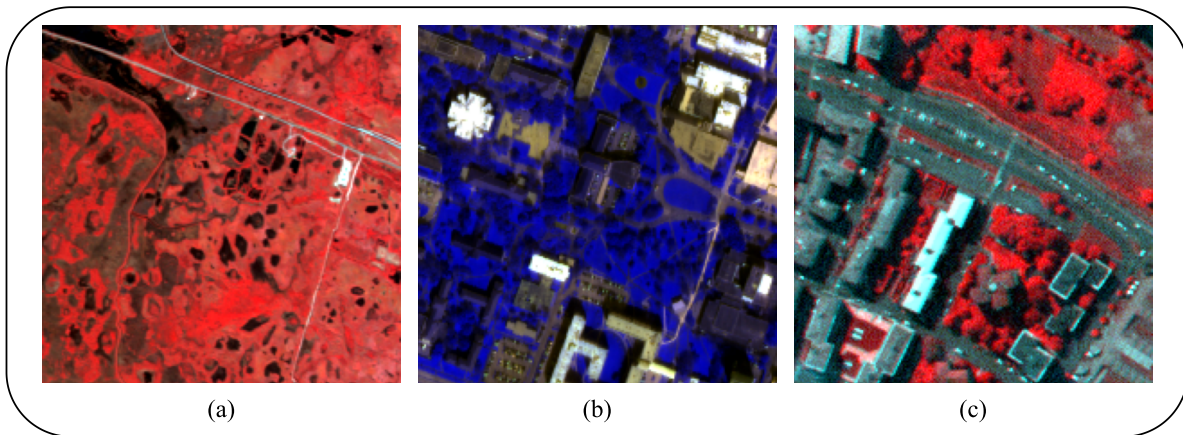


Figure 6: HSIs used in our experiments with pseudo color representation: (a) Kennedy space center shown in pseudo color with (43, 21, 11) bands, (b) Houston-2018 shown in pseudo color with (2, 3, 35) bands, and (c) Pavia center shown in pseudo with (97, 3, 2) bands.

5.1 Evaluation Metrics

To quantitatively assess the performance, three metrics have been used, i.e., the peak signal-to-noise ratio (PSNR), the structural similarity index measurement (SSIM), and the spectral angle mapper (SAM). These indicators are widely exploited to measure the performance of HSI denoising.

The PSNR metric can be defined as in Eq. (19):

$$\text{PSNR} = 20 \log \frac{\max(X)}{\sqrt{\text{MSE}}}, \quad (19)$$

where $\max(\cdot)$ indicates the maximum value of the pixels in the image X , while MSE is the mean square error between the reference and restored images.

Instead, the SSIM index is defined as:

$$\text{SSIM} = \frac{(2\mu_X\mu_Y + s_1)(2\sigma_{XY} + s_2)}{(\mu_X^2 + \mu_Y^2 + s_1)(\sigma_X^2 + \sigma_Y^2 + s_2)}, \quad (20)$$

where X is the reference (ground-truth) image, Y is the restored (predicted) one, μ is the mean operator, σ^2 is the variance operator, while s_1 and s_2 are constants.

Finally, the SAM quality metric is as follows:

$$\text{SAM} = \arccos\left(\frac{\langle X, Y \rangle}{\|X\|_2 \|Y\|_2}\right), \quad (21)$$

where $\langle \cdot, \cdot \rangle$ is the inner product, $\arccos(\cdot)$ refers to the arccosine function, and $\|\cdot\|_2$ indicates the ℓ_2 norm. The value of the SAM index is calculated per pixel (usually in degrees), and then averaged to get the overall quality index. For PSNR and SSIM, higher values indicate better performance. Instead, the lower, the better for the SAM metric. Ideal values are 0 for SAM, 1 for SSIM, and $+\infty$ for PSNR.

5.2 Simulated Data

For the simulated experiments, the Kennedy Space Center (KSC) dataset (that is publicly available) has been used, having a size of $512 \times 614 \times 176$. The image has been divided into two parts to train and test the model. A region of $200 \times 200 \times 176$ has been cropped for testing purposes and the rest has been used to train the model. The training part has been cropped into patches of size of 20×20 with a stride of 20. The noisy data have been produced by adding the noise on the different bands of the HSI. Data augmentation techniques (multi-angle image rotation and multi-scale resizing) have been applied, and noisy images have been generated to train the network to address different kinds of noises. To perform the experiments, we added the noises into the clean image using three different cases:

Case 1: In the first case, we added noise of equal intensity for the different spectral bands. For instance, we set σ_n from 5 to 100, and the effects of these intensities are reported in [Tables 1 and 2](#).

Case 2: We added noise with random intensities $\sigma_n = \text{rand}(25)$ to the different spectral bands. The intensity of the noise varies and follows the random probability distribution (setting the range to 25 for the uniform distribution), all the results are noted in [Table 2](#).

Case 3: In this case, we utilized the Gaussian noise to generate the noisy image for the simulated experiments. Gaussian noise distributions for the different bands have been used in this case. $\sigma = \text{Gau}(200, 30)$ has been used along the spectral dimension and the noise level varies following the Gaussian curve [62]. The Gaussian distribution has two parameters, i.e., mean and standard deviation, in our case, they are set to 200 and 30, respectively. The results related to the Gaussian noise are reported in [Table 2](#).

For simulated data, one model is trained for all the test cases. More specifically, there are three training stages. In the first stage, we added the noise levels $\sigma = 5, 25, 50, 75, 100$ to the image to generate the training dataset. In the second stage, random noise levels have been added to the image to obtain the training samples. In the last stage, Gaussian noise has been included to generate the noisy data to train the network. Finally, the model effectiveness is measured with noises generated as in the above-mentioned cases (cases from 1 to 3).

Table 1: Quantitative results for different methods at various noise levels ($\sigma = 5, 25, 50, 75$). The best results are in boldface.

Models	$\sigma = 5$				$\sigma = 25$				$\sigma = 50$				$\sigma = 75$			
	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM	
LRMR	28.30 ± 3.562	0.928 ± 0.226	0.061 ± 0.059	27.59 ± 3.340	0.815 ± 0.209	0.096 ± 0.052	23.81 ± 2.779	0.668 ± 0.169	0.142 ± 0.059	22.17 ± 2.525	0.581 ± 0.139	0.179 ± 0.067				
DnCNN	26.29 ± 2.709	0.884 ± 0.129	0.097 ± 0.057	25.09 ± 2.498	0.778 ± 0.112	0.116 ± 0.056	23.44 ± 2.095	0.672 ± 0.092	0.135 ± 0.061	21.93 ± 1.830	0.603 ± 0.077	0.155 ± 0.051				
MemNet	27.55 ± 4.508	0.848 ± 0.079	0.121 ± 0.080	25.63 ± 3.238	0.768 ± 0.060	0.121 ± 0.153	23.40 ± 3.111	0.661 ± 0.053	0.163 ± 0.100	21.46 ± 3.250	0.570 ± 0.057	0.204 ± 0.106				
DeNet	41.57 ± 2.025	0.980 ± 0.021	0.032 ± 0.034	29.36 ± 3.328	0.770 ± 0.028	0.140 ± 0.074	22.63 ± 4.169	0.546 ± 0.015	0.278 ± 0.028	21.11 ± 2.094	0.521 ± 0.037	0.241 ± 0.054				
GradNet	41.08 ± 3.806	0.980 ± 0.015	0.031 ± 0.014	29.17 ± 4.013	0.854 ± 0.043	0.091 ± 0.100	25.51 ± 2.017	0.718 ± 0.047	0.127 ± 0.032	23.07 ± 3.047	0.623 ± 0.054	0.165 ± 0.106				
ENCAM	41.85 ± 3.102	0.981 ± 0.038	0.030 ± 0.019	33.38 ± 4.364	0.924 ± 0.067	0.063 ± 0.034	28.90 ± 2.876	0.842 ± 0.070	0.105 ± 0.024	26.80 ± 2.802	0.792 ± 0.091	0.126 ± 0.040				
HSIDwRD	36.85 ± 2.056	0.975 ± 0.254	0.032 ± 0.217	29.41 ± 4.423	0.848 ± 0.226	0.090 ± 0.120	25.00 ± 3.341	0.709 ± 0.192	0.135 ± 0.122	23.20 ± 4.261	0.623 ± 0.182	0.159 ± 0.214				
MAN	20.11 ± 2.247	0.776 ± 0.163	0.247 ± 0.130	19.14 ± 2.063	0.622 ± 0.119	0.305 ± 0.122	17.14 ± 1.670	0.455 ± 0.082	0.418 ± 0.110	15.31 ± 1.334	0.346 ± 0.058	0.519 ± 0.101				
SST	35.78 ± 2.055	0.972 ± 0.041	0.042 ± 0.121	28.83 ± 2.321	0.852 ± 0.221	0.092 ± 0.120	25.08 ± 1.689	0.720 ± 0.081	0.122 ± 0.0126	23.43 ± 2.791	0.629 ± 0.119	0.150 ± 0.050				
SERT	41.99 ± 3.988	0.986 ± 0.389	0.028 ± 0.115	34.25 ± 2.260	0.931 ± 0.213	0.066 ± 0.032	28.96 ± 2.971	0.854 ± 0.182	0.104 ± 0.063	26.90 ± 2.343	0.796 ± 0.143	0.127 ± 0.321				
SSIT	42.16 ± 1.043	0.987 ± 0.013	0.026 ± 0.124	34.30 ± 1.932	0.933 ± 0.214	0.057 ± 0.123	28.98 ± 3.563	0.858 ± 0.215	0.103 ± 0.167	27.02 ± 2.410	0.798 ± 0.216	0.128 ± 0.158				
CDCN	42.89 ± 0.672	0.987 ± 0.003	0.024 ± 0.002	34.86 ± 1.682	0.939 ± 0.047	0.056 ± 0.022	29.64 ± 1.654	0.863 ± 0.071	0.102 ± 0.035	27.35 ± 1.827	0.809 ± 0.084	0.126 ± 0.046				

Table 2: Quantitative results for different methods at various noise levels ($\sigma = 100$, $rand(25)$ and $Gau(200, 30)$). The best results are in boldface.

Models	$\sigma = 100$			$\sigma = rand(25)$			$\sigma = Gau(200, 30)$		
	PSNR	SSIM	SAM	PSNR	SSIM	SAM	PSNR	SSIM	SAM
LRMR	21.24 ± 2.201	0.520 ± 0.113	0.195 ± 0.078	20.04 ± 2.230	0.805 ± 0.215	0.108 ± 0.077	26.95 ± 3.342	0.811 ± 0.201	0.097 ± 0.055
DnCNN	21.08 ± 1.427	0.537 ± 0.047	0.166 ± 0.036	24.81 ± 2.551	0.852 ± 0.133	0.120 ± 0.058	25.50 ± 2.570	0.785 ± 0.114	0.107 ± 0.058
MemNet	20.70 ± 2.070	0.523 ± 0.038	0.223 ± 0.057	25.76 ± 2.562	0.838 ± 0.063	0.133 ± 0.056	25.56 ± 2.935	0.767 ± 0.061	0.125 ± 0.047
DeNet	20.34 ± 2.850	0.474 ± 0.043	0.275 ± 0.092	34.01 ± 4.935	0.976 ± 0.034	0.079 ± 0.075	29.88 ± 6.597	0.779 ± 0.014	0.132 ± 0.025
GradNet	22.24 ± 1.427	0.568 ± 0.052	0.185 ± 0.052	33.54 ± 4.748	0.977 ± 0.039	0.079 ± 0.123	30.61 ± 2.763	0.844 ± 0.030	0.093 ± 0.037
ENCAM	25.84 ± 2.713	0.752 ± 0.090	0.148 ± 0.040	34.06 ± 4.927	0.985 ± 0.113	0.074 ± 0.079	34.10 ± 4.642	0.979 ± 0.057	0.671 ± 0.026
HSIDwRD	21.90 ± 2.682	0.563 ± 0.138	0.178 ± 0.148	32.74 ± 5.840	0.977 ± 0.282	0.080 ± 0.146	28.92 ± 4.205	0.829 ± 0.221	0.095 ± 0.102
MAN	14.33 ± 1.168	0.303 ± 0.049	0.572 ± 0.096	27.02 ± 3.461	0.897 ± 0.150	0.167 ± 0.110	19.35 ± 2.103	0.624 ± 0.121	0.296 ± 0.123
SST	21.97 ± 5.445	0.577 ± 0.208	0.174 ± 0.220	27.36 ± 2.371	0.950 ± 0.153	0.091 ± 0.0092	29.52 ± 5.518	0.837 ± 0.344	0.088 ± 0.110
SERT	25.90 ± 2.121	0.766 ± 0.084	0.147 ± 0.045	34.20 ± 3.923	0.980 ± 0.114	0.079 ± 0.078	34.22 ± 3.523	0.910 ± 0.056	0.076 ± 0.027
SSIT	26.09 ± 2.025	0.768 ± 0.086	0.143 ± 0.072	34.23 ± 0.823	0.975 ± 0.016	0.076 ± 0.0086	34.24 ± 1.87	0.911 ± 0.054	0.067 ± 0.033
CDCN	26.51 ± 2.050	0.776 ± 0.094	0.134 ± 0.061	34.33 ± 0.936	0.981 ± 0.006	0.073 ± 0.007	34.28 ± 1.921	0.940 ± 0.048	0.054 ± 0.022

Fig. 7 show the spectral response curves at three different spatial locations, i.e., pixel (30, 30), (50, 50), and (70, 70), under the high noise intensity of $\sigma = 100$. The blue curves show the noise, where as red curves are used to show the denoised outputs produced by the CDCN model. Fig. 7a shows high noise corruption in pixel (30, 30) as the noisy signal in the spectral bands has a high variation. The denoised curve however, is much smoother and follows the underlying spectral trend much better showing good noise suppression and the presence of important spectral features. Also, at pixel (50, 50) in Fig. 7b, where the signal intensity is relatively higher and the spectral variations are more pronounced, the proposed algorithm effectively removes noise without any structural damage being done to peaks and transitions. This shows the capability of the model to process the complicated spectral patterns without over-smoothing. The signal in Fig. 7c at the pixel (70, 70) is relatively weak and is more susceptible to noise. Nevertheless, the denoised result is consistent and predictable, actually hiding the noisy details, but preserving the overall spectral pattern.

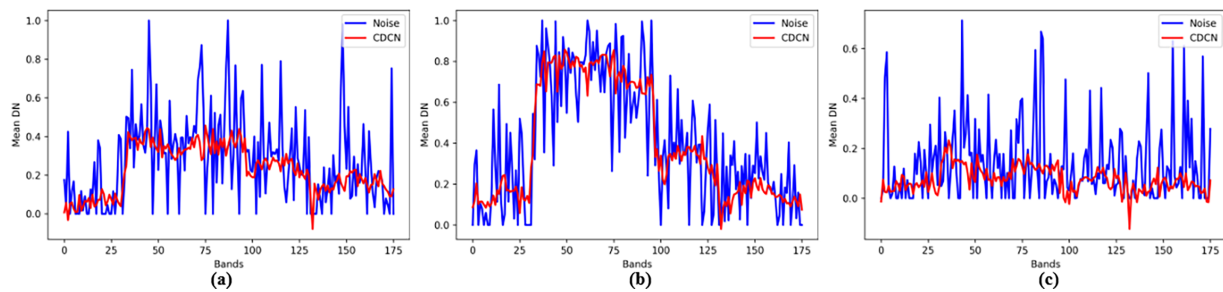


Figure 7: (a) Pixel (30, 30), (b) (50, 50), and (c) (70, 70) denoising curves achieved at $\sigma = 100$ of the proposed model.

Overall, the outputs achieved from different spatial locations confirm that the proposed method achieves good and consistent denoising performance. It efficiently removes high-intensity noise while preserving spectral information, which is crucial for reliable HSI analysis.

5.3 Results on Simulated Data

First, we measured the performance of our model according to the first case, where we added the noise with the same intensity to the different bands. The experimental results of the first case of $\sigma = \{5, 25, 50, 75\}$ are reported in Table 1, while the $\sigma = 100$ results of this case are listed in Table 2 using the three evaluation metrics (PSNR, SSIM, and SAM). Our method removed the noise from the HS images and predicted high-quality images. More specifically, we added different intensities of noises using $\sigma = \{5, 25, 50, 75, 100\}$. The results are reported in Tables 1 and 2 and visual representations of the denoised images for $\sigma = 100$ are shown in Fig. 8. We can see that the results of our approach are superior with respect to the other solutions. We compared the proposed network with the LRMR algorithm. The results of LRMR show a clear partial effect because the noise in the original data has not been removed in a proper way. DnCNN only considers the spatial noise and does not take the spectral information into account. MemNet shows residual noise. DeNet can only remove a reduced amount of noise. The denoised image produced by the GradNet has some artifacts, and ENCAM still has some artifacts, whereas ENCAM produced a better-denoised image. However, ENCAM requires a higher computational burden with lower performance with respect to our proposal. We also compared our method with HSIDwRD and MAN. HSIDwRD produces a better outcome but low quantitative results can be noted using standard settings, while MAN produces poor results with $\sigma = 100$. Additionally, a comparison was conducted with SST and SERT algorithms. SST achieved 21.97 ± 5.445 PSNR, 0.577 ± 0.208 SSIM and 0.174 ± 0.220 SAM, whereas SERT achieved 25.90 ± 2.121 PSNR, 0.766 ± 0.084 SSIM and produced 0.147 ± 0.045 SAM. On $\sigma = 100$ SERT predicted the good best quantitative results

and produced the good denoised image which can be seen in Fig. 8k. Similarly, SSIT is the other transformer-based method that produced the second-best result in this series. This model also predicted the clearer map which is presented in Fig. 8l, and achieved 26.09 ± 2.025 , 0.768 ± 0.086 and 0.143 ± 0.072 PSNR, SSIM and SAM, respectively. However, the images obtained by the proposed CDCN are much better with high accuracy than all the other approaches in the benchmark. As the quantitative results are shown in Tables 1 and 2. It is noticed that the proposed approach obtains significant performance on PSNR when compared with existing techniques. Moreover, the presented method attains the lowest SAM metric, which shows that the proposed method is better at maintaining spectral consistency. Two zoomed regions of the KSC image are presented in Fig. 9, where we can notice that the visual results produced by LRMN, DnCNN, MemNet, DeNet, and MAN still contain noises. GradNet, ENCAM, and HSIDwRD show better results in minimizing the noise on the KSC image. MAN generally shows poor outcomes. Although the visual results in Fig. 8g,h are clear, the PSNR values of GradNet and ENCAM are 22.24 ± 1.427 and 25.84 ± 2.713 . However the transformer based SST, SERT and SSIT produced the good results where as the proposed model gets 26.51 ± 2.050 . The denoising results by our approach obtain the minimum noise. MemNet, GradNet, ENCAM and HSIDwRD produced good results, but the structure information is poorly preserved. In the same way, LRMN and DnCNN are not the best solutions for denoising, as shown in Fig. 8b,c, respectively. The images still have noise. Our method shows good qualitative and quantitative results as depicted in Fig. 8l and in the close-ups in Fig. 9l. For a fair comparison, we run all the models using the same parameters. Fig. 10 depict the training curves considering PSNR, SSIM, and SAM as quality metrics, which are further discussed in the next section i.e., convergence analysis.

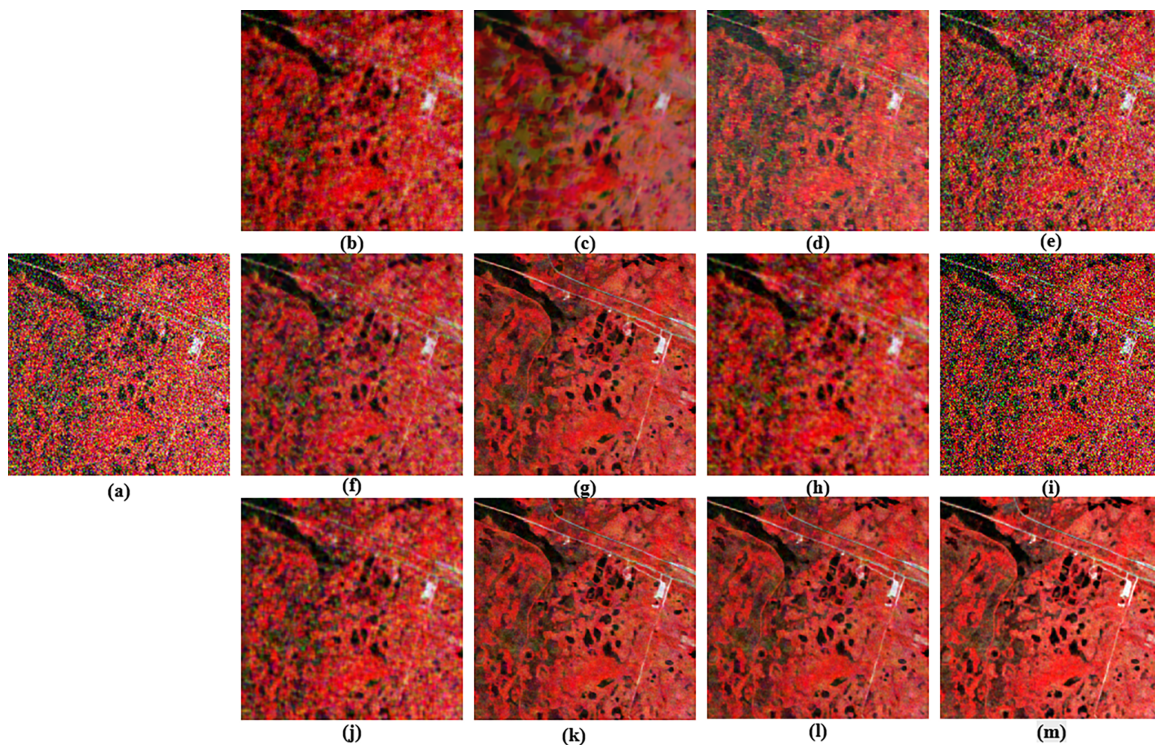


Figure 8: Denoised images for the KSC dataset under noise level $\sigma = 100$ for Case 1: (a) noisy image with pseudo-color by (43, 21, 11) bands, (b) LRMN, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN.

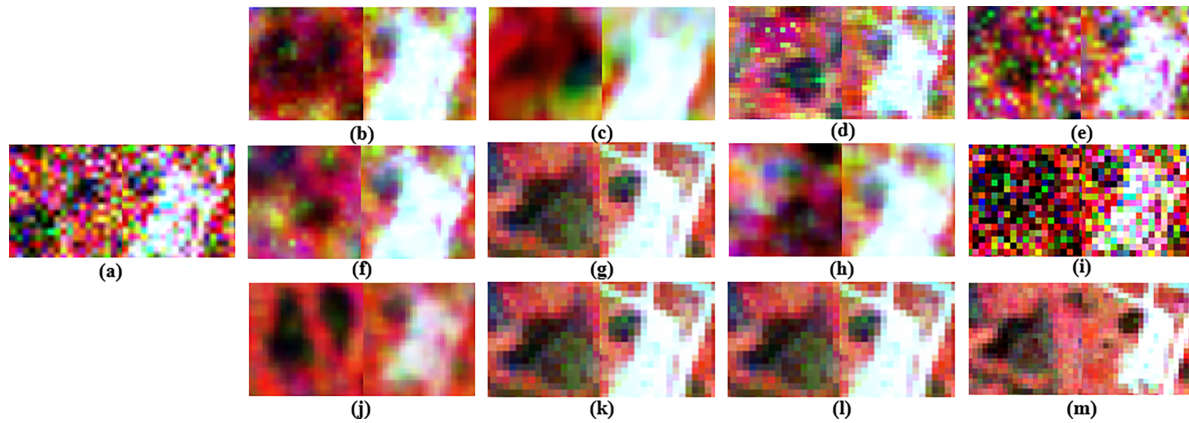


Figure 9: Close-ups for the denoised images for the KSC dataset under noise level $\sigma = 100$ for Case 1: (a) noisy image with pseudo-color by (43, 21, 11) bands, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) CDCN.

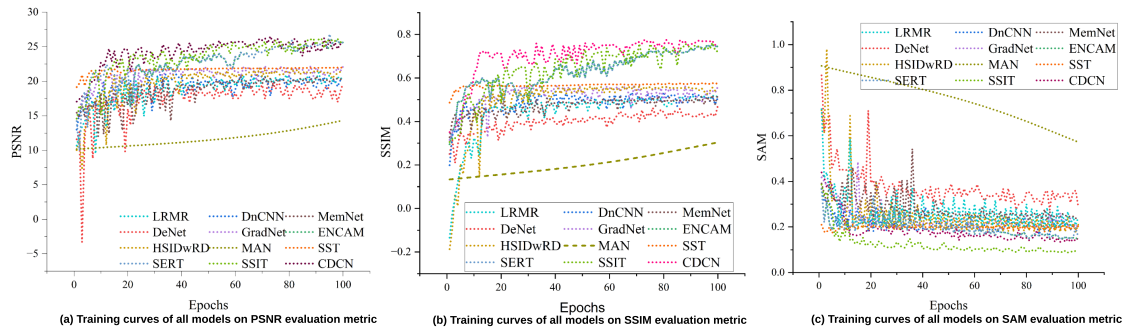


Figure 10: (a) PSNR, (b) SSIM, and (c) SAM values at noise level $\sigma = 100$ for all the models using simulated data.

Table 2 and Fig. 11 present the results about the second case, where we performed the experiments using $\sigma = rand(25)$. Analyzing the quantitative results, ENCAM achieved a better result on SSIM. LRMR, DnCNN, MemNet, MAN and SST achieved the lowest numeric results, Similarly DeNet, GradNet SERT and SSIT produced the better numeric results on random (25) noise. Whereas the CDCN model has superior results on the SAM and PSNR metrics. About the images produced for Case 2, see Fig. 11, all the methods predict similar images except for LRMR, DnCNN, and MemNet, which got poorer outcomes. The close-ups for Case 2 are depicted in Fig. 12. Again, Fig. 13 depicts the training curves considering PSNR, SSIM, and SAM as quality metrics.

Table 2 and Fig. 14 report the outcomes of the Gaussian noise (Case 3), where our model shows again superior performance with respect to the benchmark. MAN obtained the lowest quantitative performance, Whereas LRMR, DnCNN, MemNet could not perform well on Gaussian noise. On the other hand DeNet, GradNet, HSIDwRD and SST produced the better results, Similarly ENCAM generated highest SSIM i.e., 0.979 ± 0.057 but produced the poor SAM results, i.e., 0.671 ± 0.026 . So in contrast SERT and SSIT provided the stable results, but this algorithm still behind CDCN. The DC magnified case-3 of all (SSIM, LRMR, DnCNN, MemNet, MAN, and SST) with noise inclusion is shared in Fig. 15.

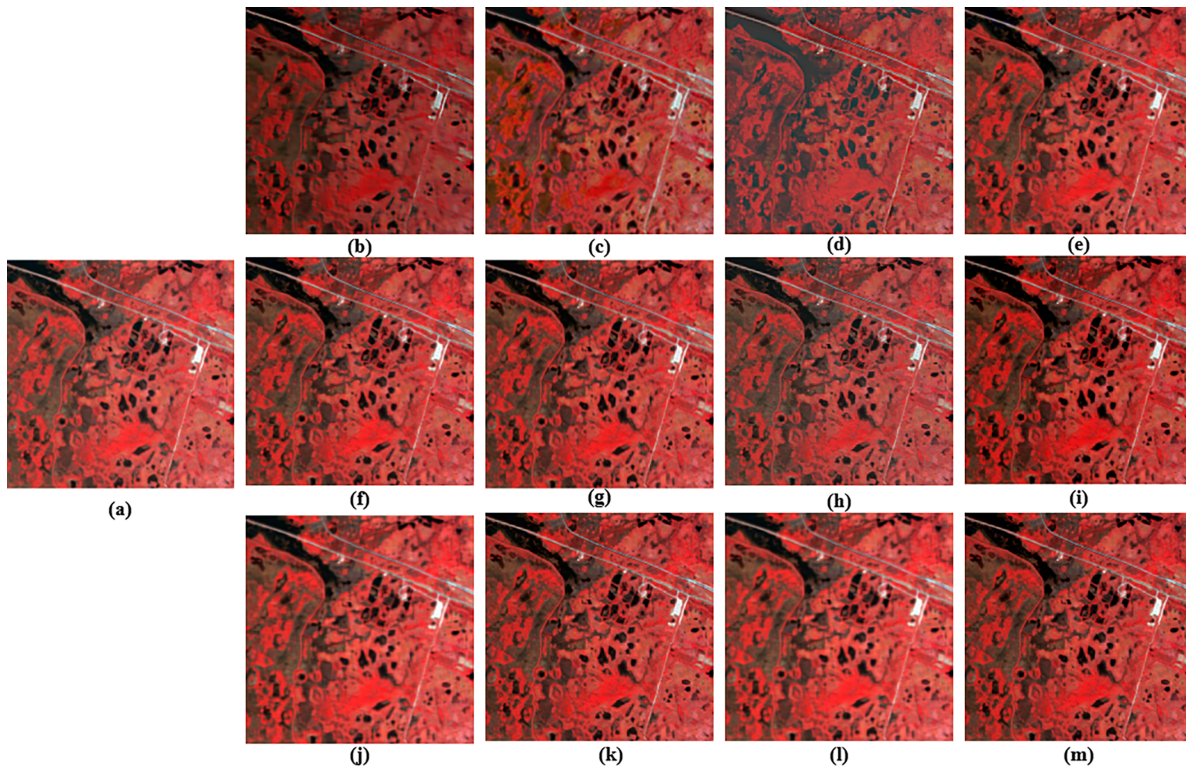


Figure 11: Denoised images for the KSC dataset for Case 2: (a) Pseudo color of noisy image with (43, 21, 11) bands, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN.

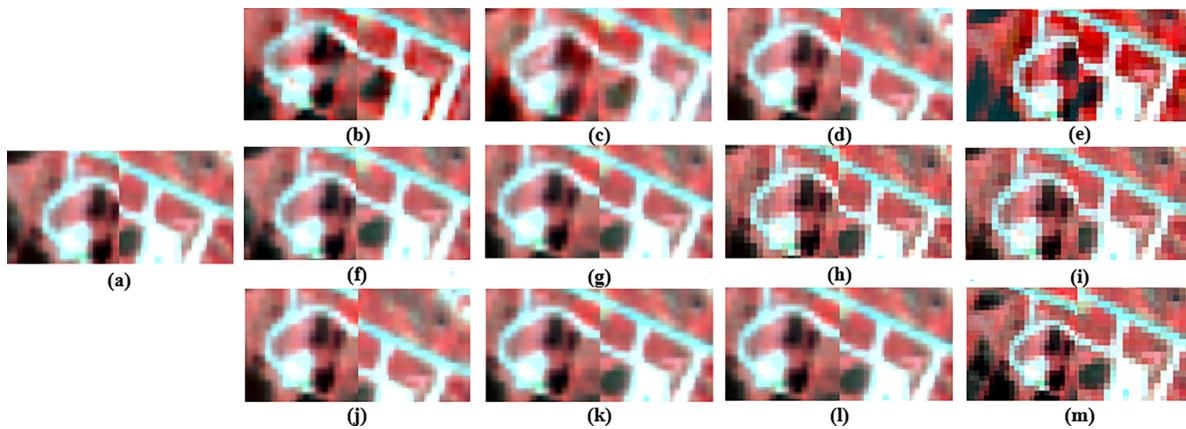


Figure 12: Close-ups for the denoised images for the KSC dataset under noise level $\sigma = \text{rand}(25)$ for Case 2: (a) noisy image, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT, and (m) CDCN. All the predicted images are composed of pseudo-color with (43, 21, 11) bands.

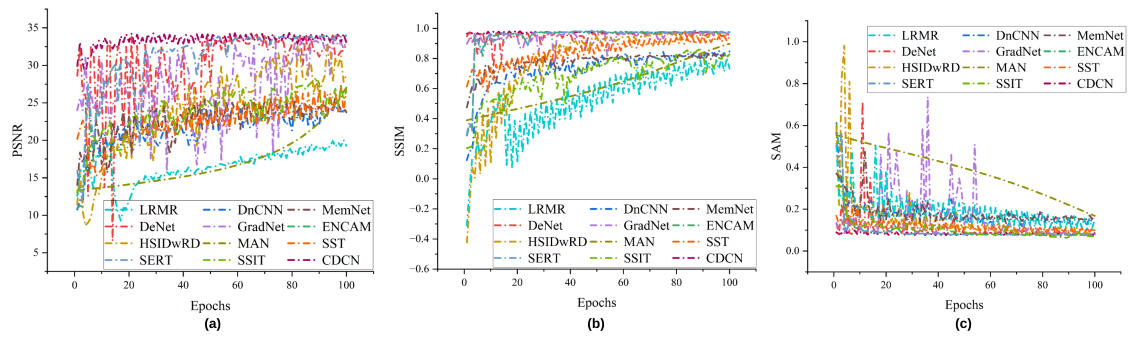


Figure 13: (a) PSNR, (b) SSIM, and (c) SAM values at noise level $\sigma = rand(25)$ for all the models using simulated data.

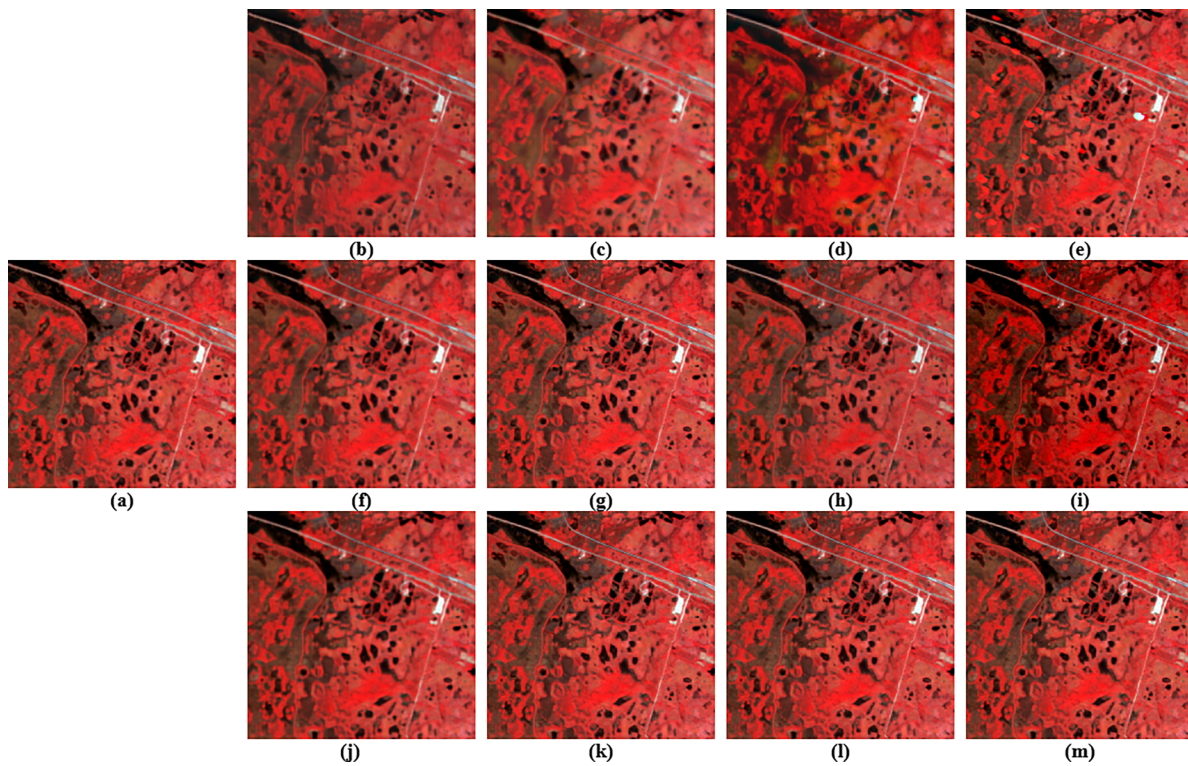


Figure 14: Denoised images for the KSC dataset for Case 3: (a) Pseudo-color noisy image with (43, 21, 11) bands, (b) LRM, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) Proposed CDCN.

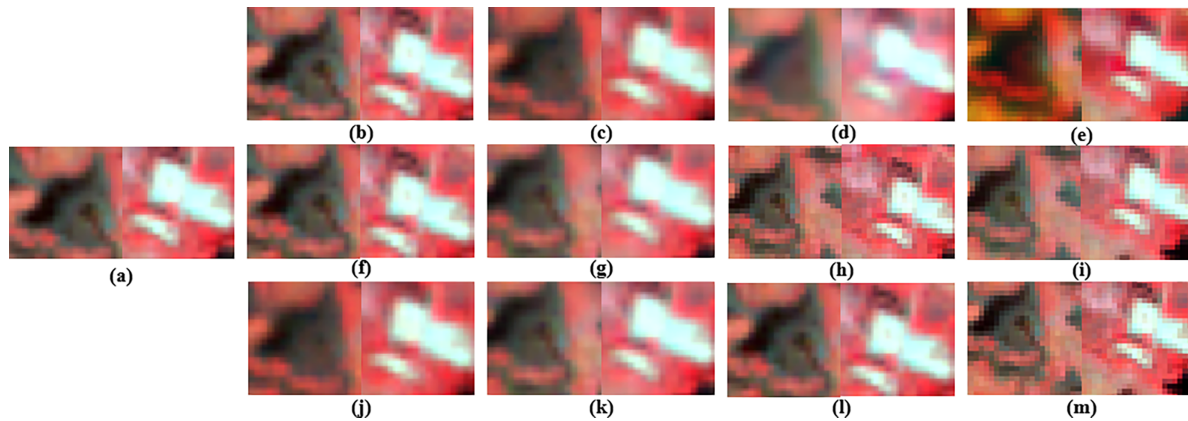


Figure 15: Close-ups for the denoised images for the KSC dataset under noise level $\sigma = Gau(200, 30)$ for Case 3: (a) noisy image of pseudo-color with (43, 21, 11) bands, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN.

For HSI interpretation, the quality of the spectral signatures is essential because they enable us to distinguish the ground objects based on their physical properties. To check the effectiveness of the experimental results, each experiment was repeated three times, and the final results are reported as mean \pm standard deviation. We concluded that LRMR, DnCNN, DeNet, GradNet, ENCAM, SST, SERT, and SSIT provide good results when working with a high signal-to-noise ratio, e.g., $\sigma = 5$. The main drawback of most of the compared approaches is the poor performance, these methods do not focus on the fine-grained information, and they do not fully utilize intrinsic information with respect to the proposed method, achieving high quantitative and qualitative performance.

5.4 Training Convergence Analysis

Fig. 10 shows the training development of various HSI denoising models over epochs with respect to the metrics of PSNR, SSIM and SAM at $\sigma = 100$. As Fig. 10a demonstrates, PSNR values of all the methods steadily rise as the training number of epochs increases, which means that the models gradually learn how to eliminate noise and rebuild high-quality hyperspectral images. On the same note, Fig. 10b indicates that the values of the SSIM grow consistently throughout the training period and this indicates the better maintenance of spatial structural information in the restored images. The values of the SAMs, on the other hand, in Fig. 10c show a downward trend as training continues, and so spectral distortion between the reconstructed and ground-truth hyperspectral data fades.

It is also possible to note that most of the methods improve their performance at a rapid rate during the initial training epochs and then gradually stabilize as the models approach convergence. It is worth noting that, the proposed CDCN model is always characterized by better PSNR and SSIM values with lower SAM values than the rival approaches. Furthermore, it is observed that the performance curves of CDCN stabilize after a few epochs, which shows that the network is optimized and converges stably. Those observations indicate that the CDCN framework can learn discriminative spectral spatial representations and produce the best denoising results in the course of training.

The training curves on the random noise of different methods are shown in Fig. 13, it can be noted that the deep learning based methods achieve the incremental improvement in the denoising of the results as the number of training runs. The SST, SERT SSIT and CDCN models demonstrate consistent stability of convergence in all evaluation measures. specifically, the PSNR curves increase steadily trend and attains

a competitive progress when we compare with other techniques. In the same way, the SSIM training curves also increases steadily throughout training which means that the suggested model is effective in terms of preserving structural information and eliminating the noise. The spectral distortion during the training process is relatively low with regard to the proposed method in terms of the SAM metric. These findings indicate that the proposed architecture is capable of effectively learning strong spatial-spectral representations even when there is a noise randomly distributed.

Fig. 16 presents the convergence behavior of different comparing methods under the Gaussian noise setting, as the number of epochs increases all the methods improve their denoising efficiency, however some models could not improve their learning such as DnCNN and MemNet on the different metrics such as PSNR, SSIM and SAM. SST learning performance is not satisfactory as in the $\sigma = 100$ and random noise setting, however the curves of SERT, SSIT and CDCN demonstrate the stable learning and achieved the constantly improvement in PSNR and SSIM through out the training epochs. Meanwhile, the SAM value declines over time, meaning that it preserves spectral properties better. As compared to comparing methods, the convergence behavior of the CDCN model is relatively stable, which implies that the combination of the feature extraction unit, separable convolutions, skip connections, and spatial-spectral attention mechanisms make hyperspectral learning both the spatial and spectral dependencies in hyperspectral data.

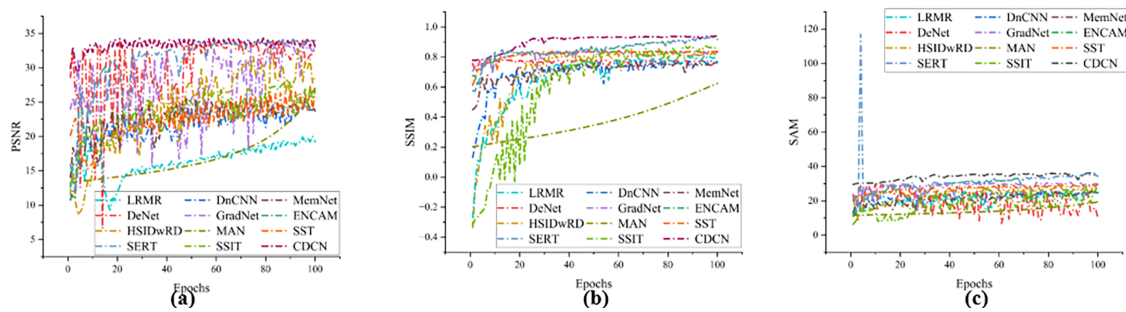


Figure 16: (a) PSNR, (b) SSIM, and (c) SAM values at noise level $\sigma = Gau(200, 30)$ for all the models using simulated data.

5.5 Experiments on Real Data

In real cases, HSI datasets contain different types of noises. Therefore, two real-world datasets have been selected to assess the effectiveness of our proposed network, i.e., Houston-2018 (HT) and Pavia Center (PC). Different methods have been used to minimize the noise from these two real datasets, at the end to checkout the effectiveness of the denoising, additionally the classification experiments were performed using the SVM classifier on the denoised HS Images, obtained from the denoisers. For this purpose two well known classification metrics were used i.e., Over all accuracy (OA), refers to the proportion of correctly classified samples out of total samples and the other was Kappa coefficient which measure the agreement between predicted values and the actual classified values, while accounting for the possibility of agreement occurring by chance. Finally, the quantitative and the qualitative results of the all the methods were listed in Tables 3 and 4 for the comparison.

Table 3: Classification results (%) for the HT dataset using an SVM classifier. Best results are in boldface. OI represents the original image.

Index	OI	LRMR	DnCNN	MemNet	DeNet	GradNet	ENCAM	HSIDwRD	MAN	DSTrans	SERT	SSIT	CDCN
OA	94.70	93.69	93.05	94.08	94.96	95.36	95.84	94.80	94.22	95.87	96.00	96.04	96.16
Kappa	81.29	77.54	73.29	78.29	82.48	83.40	81.62	81.64	84.02	84.24	83.04	86.00	86.24

Table 4: Classification results (%) for the PC dataset using an SVM classifier. The best results are in boldface. OI, representing the original image.

Index	OI	LRMR	DnCNN	MemNet	DeNet	Gradnet	ENCAM	HSIDwRD	MAN	SST	SERT	SSIT	CDCN
OA	74.59	94.19	91.8	85.39	91.4	96.2	94.19	95.80	90.60	95.25	96.15	96.20	96.80
Kappa	70.92	92.51	89.32	81.01	88.84	95.09	92.49	94.57	87.90	94.19	95.26	95.30	95.86

5.5.1 HT

The HT dataset has been collected by combining the efforts of the IEEE Geoscience and Remote Sensing Society (GRSS) and Houston University. This dataset was introduced first in January 2018 for a data fusion contest. HT [63] was acquired by the ITRES CASI 1500 instrument. We took the image of size 210×954 pixels with 48 bands. The noise distorts some bands of the HT datasets, and we removed the noise from the 48 bands of the HT image. For the experiments, we cropped a region of the image of size of 200×200 removing Gaussian and impulse noises from the HT dataset. Denoising results of various compared methods are depicted in Figs. 17 and 18. Removing the noise from real HS image datasets is a big challenge. Pseudo-color images using (2, 3, 35) bands of the 48 denoised ones are shown in Fig. 17 reports the visual outcomes. The results show that GradNet and ENCAM efficiently reduce the noise, but the denoising results still have some stripes and residual noise. The ENCAM method has a good ability to reduce the noise from the HSI. Similarly, MemNet is considered a good denoising algorithm but not good under high levels of noise. HSIDwRD removes the noise well, but it appears powerless against heavy stripping. Although MAN is a deep learning HSI denoising algorithm that employs spatial-spectral information and considers an attention module to remove mix noise, it only shows better results on $rand(25)$ in our simulated data experiments. SST, SERT and SSIT algorithms showed the better performance on the real HS images to remove the noise. Where as the proposed model shows overall better results in terms of quantitative results and visual effects by retaining the link between spatial and spectral information. To assess the performance of denoising methods on real data, we tested the classification accuracy of the denoised results obtained by the benchmark. The curves generated for (a) PSNR, (b) SSIM, and (c) SAM during the noise removal from real data (HT) is shared in Fig. 19. Table 3 reports the HT classification results and Fig. 20 illustrates the classified results which obtained after the denoising step. Table 3 shows that the OA of the original image is 94.70% and the kappa coefficient is 81.29%. After denoising, the classification results have been improved, see Table 3, whereas the different denoising methods presented visual effects in Fig. 20. The proposed model achieved good accuracy with respect to the other methods getting an OA of 96.16% and a kappa coefficient of 86.24%. The maps in Fig. 20 show that traditional methods still fail to classify many pixels, thus showing that these methods cannot effectively manage real cases. On other hand, CDCN efficiently handles real datasets overcoming the existing issues.

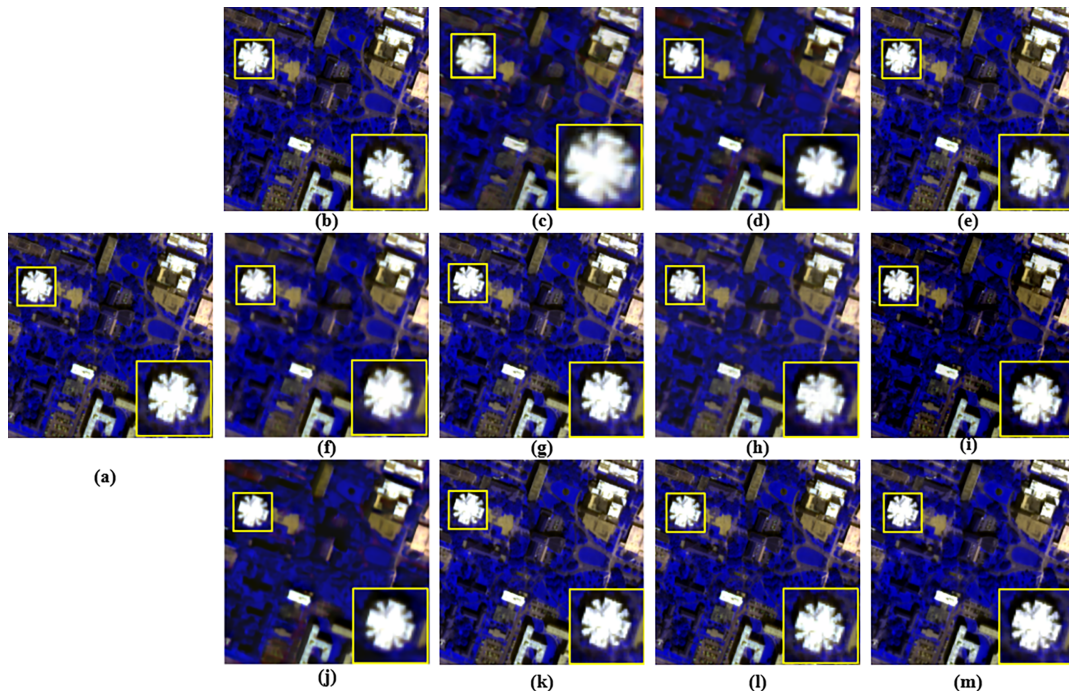


Figure 17: Outcomes for the HT dataset: (a) pseudo-color noisy image (bands 2, 3, and 35), (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN. Close-ups are indicated with yellow rectangles.



Figure 18: Outcomes for the HT dataset: (a) noisy image (band 2), (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN. Close-ups are indicated with red rectangles.

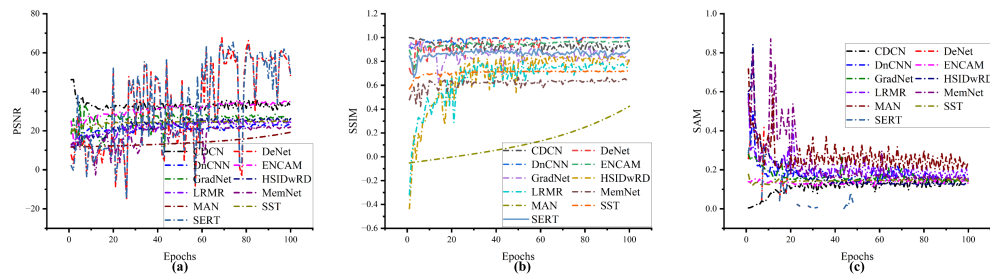


Figure 19: (a) PSNR, (b) SSIM, and (c) SAM curves generated during the noise removal from real data (HT).

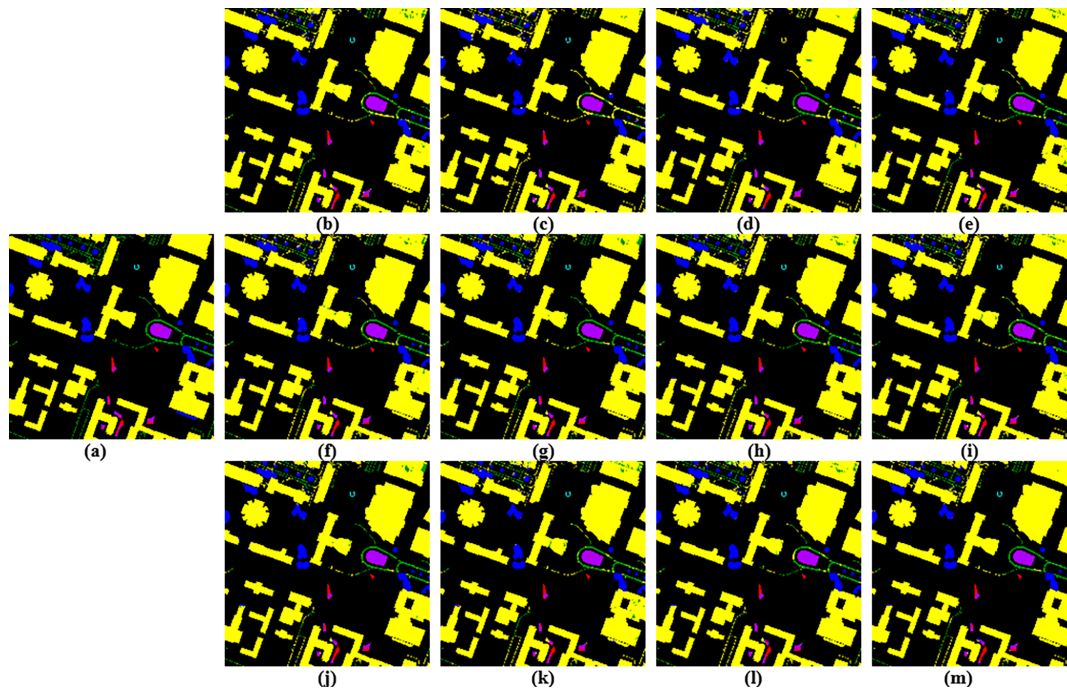


Figure 20: Classification maps for the HT dataset using an SVM classifier before and after noise removal: (a) ground-truth, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN.

5.5.2 Pavia Center (PC)

The ROSIS sensor collected the PC dataset over Pavia, a city in northern Italy. It has a size of 1096×715 with 102 spectral bands. For the experiments, we used a sub-image of size $200 \times 200 \times 102$. The ground truth consists of nine classes. Some bands (in the first positions) of the PC dataset are corrupted by noise. Figs. 21 and 22 depict the denoised image for the PC dataset obtained using LRMR, DnCNN, MemNet, DeNet, GradNet, ENCAM, HSIDwRD, MAN, and the proposed technique. Furthermore, some denoised images from the PC dataset are provided in Figs. 21 and 22. The visual results show that GradNet, ENCAM, HSIDwRD, MAN, SST, SERT and SSIT can remove the noise, but uneven noise still exists in the denoised results. ENCAM and LRMR denoising results are instead better, but they are overly smooth. Although DeNet also performs well on the PC dataset, SST, SERT and SSIT showed the good classified results where as our proposed model again shows the best performance by exploiting the relationship between the spatial and spectral dimensions of the image.

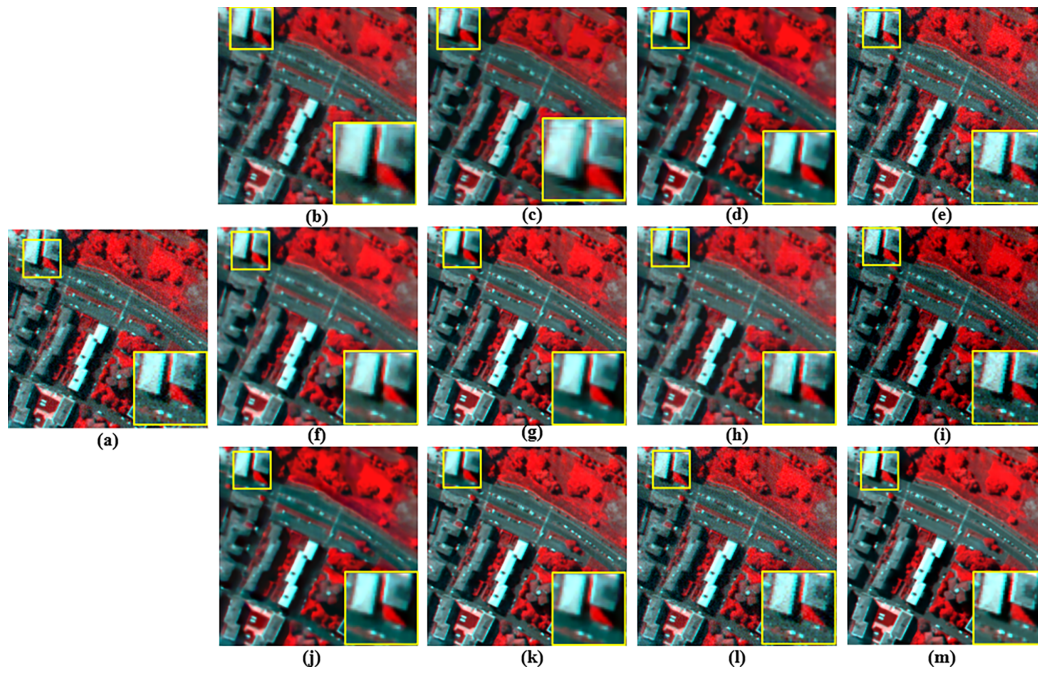


Figure 21: Outcomes for the PC dataset: (a) pseudo-color noisy image (bands 97, 3, and 2), (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT and (m) proposed CDCN. Close-ups are indicated with yellow rectangles.

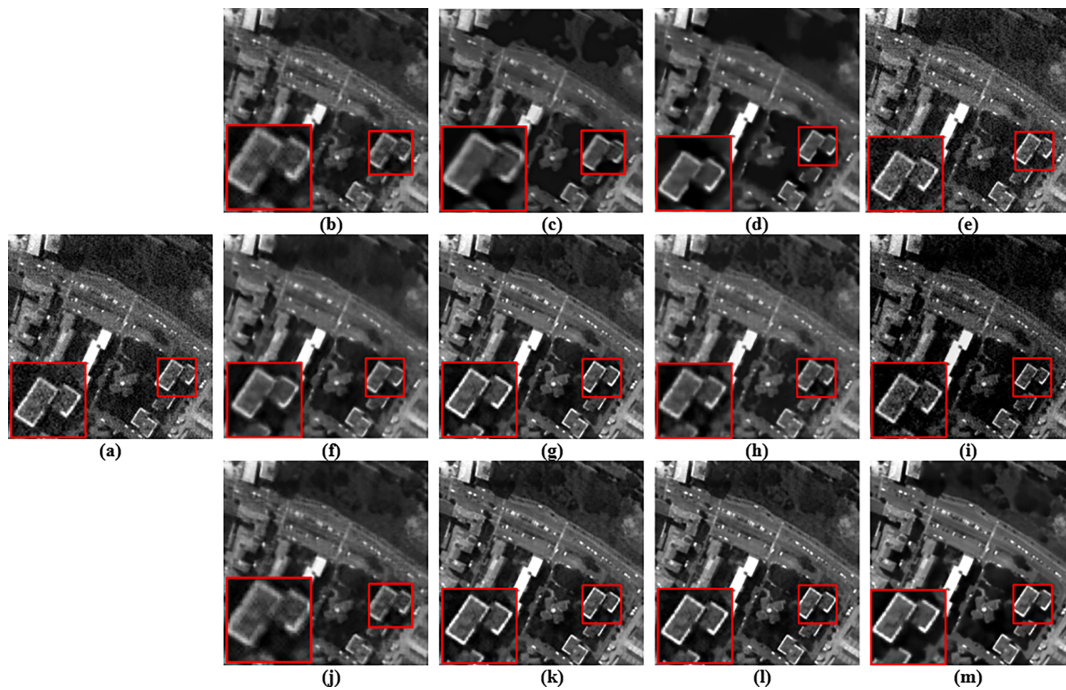


Figure 22: Outcomes for the PC dataset: (a) noisy image (band 2), (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT, and (m) proposed CDCN. Close-ups are indicated with red rectangles.

We considered 102 spectral bands for our experiments removing the noise from the HSI. The classification results are reported in Table 4. The proposed method got the highest OA and kappa coefficient values. The classification maps for the compared approaches are shown in Fig. 23.

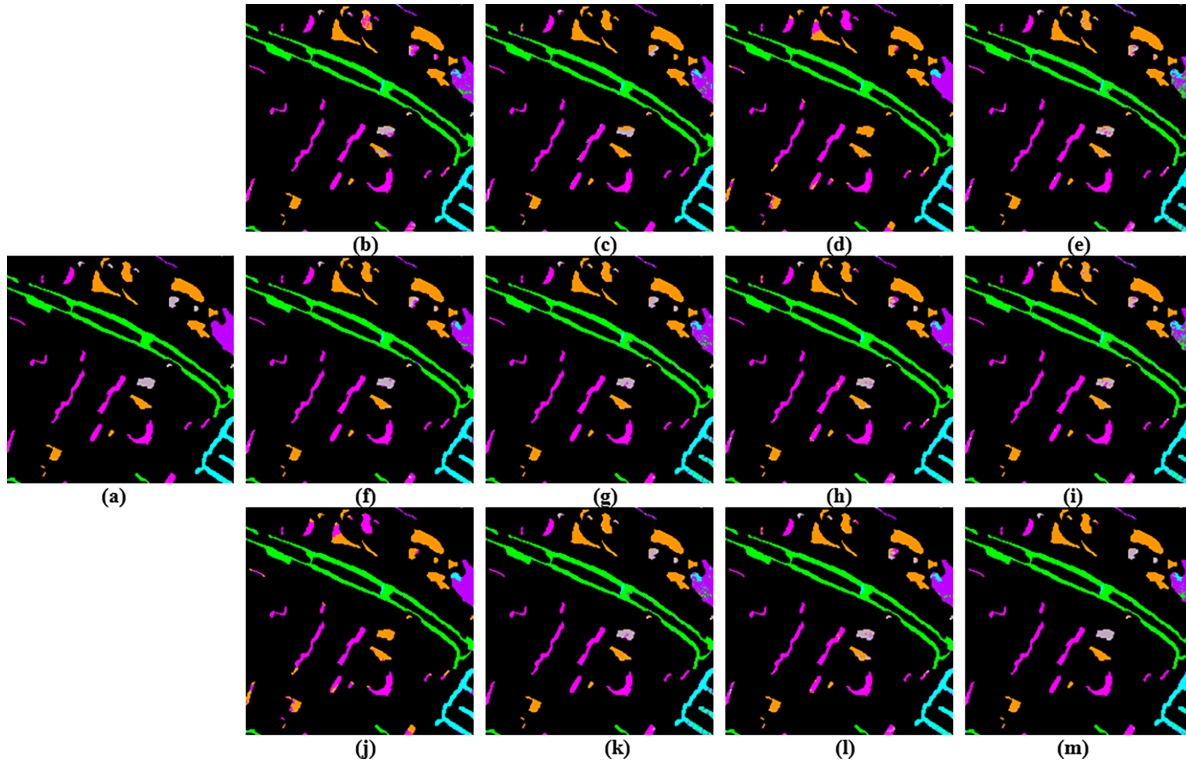


Figure 23: Classification maps for the PC dataset using an SVM classifier before and after denoising: (a) ground-truth, (b) LRMR, (c) DnCNN, (d) MemNet, (e) DeNet, (f) GradNet, (g) ENCAM, (h) HSIDwRD, (i) MAN, (j) SST, (k) SERT, (l) SSIT, and (m) proposed CDCN.

5.6 Statistical Significance Analysis

To evaluate the statistical significance of the proposed CDCN framework, a two-sample t -test is performed by comparing its overall accuracy (OA) with the mean OA of competing methods across both datasets. For statistical analysis, the results are taken from Tables 3 and 4. The statistical analysis is conducted using aggregated performance metrics, treating different methods as independent samples.

The computed p -values for both datasets are shared in the Table 5 with a significance threshold of 0.05, indicating that the performance improvements achieved by the proposed CDCN framework are statistically significant and not due to random variation.

Table 5: Statistical significance analysis of the proposed CDCN framework using a two-sample t -test based on overall accuracy (OA).

Dataset	Mean OA (Others)	CDCN OA	t -Value	p -Value
HT	94.89	96.16	3.12	0.005
PC	92.96	96.80	4.27	0.0004

Note: Bold values shows superior better results of proposed model than other models.

5.7 Ablation Study

The effectiveness of the CDCN structure is discussed in this subsection. We introduced the CDC-based architecture with an attention module in the HSI denoising domain. For the ablation study, we used the Houston dataset to measure the effectiveness of each module of our proposed architecture. The results of the ablation study are reported in Table 6:

1. CDCN w/o attention refers to the proposed CDCN without the attention module.
2. CDCN w/o CDC means that only vanilla convolutions are adopted, whereas the CDC has been removed. This model can also be viewed as the proposed CDCN with $\theta = 0$ in Eq. (14).
3. CDCN is the proposed network.

Table 6: Ablation study, the best results are in boldface.

Index	CDCN w/o Attention	CDCN w/o CDC	CDCN
PSNR	29.35	29.54	29.63
SSIM	0.860	0.858	0.863
SAM	0.099	0.103	0.102

The comparison with CDCN w/o CDC and the proposed CDCN shows that using the CDC property produced better results on two metrics (i.e., SSIM and SAM) but decreased the PSNR value. Comparing CDCN w/o CDC and the proposed CDCN, we have vanilla convolutions replace the CDC in the proposed network to get the CDCN w/o CDC approach. Experimental results show that CNCD w/o CDC produced worse results than the proposed CDCN, considering both SSIM and SAM indexes, thus proving the effectiveness of the proposed CDC module. Hence, the outcomes show that our method (which applies attention and CDC) is superior to traditional CNN-based solutions.

5.8 Computational Efficiency

The computational efficiency of the proposed framework is analyzed by measuring the average running time of different methods on simulated data. The results are summarized in Table 7. All experiments are conducted using the hardware configuration described in the experimental setup, which includes an NVIDIA Tesla T4 GPU with 16 GB GPU memory, and the models were implemented using the PyTorch deep learning framework.

Table 7: Average running times (in seconds), parameters (in million), and GFLOPs using simulated data. The best result is in boldface.

Methods	LRMR	DnCNN	MemNet	DeNet	GradNet	ENCAM	HSIDwRD	MAN	SST	SERT	SSIT	CDCN
Time (s)	57.51	6.12	12.18	9.69	2.24	42.12	2.03	4.87	3.70	6.36	3.50	5.11
Parameters (M)	2.78	0.76	2.90	1.23	0.32	2.56	0.25	1.20	1.51	0.82	0.75	0.66
GFLOPs	2.56	1.80	5.84	1.20	1.14	6.12	0.87	2.37	3.60	2.42	1.53	0.10

Deep learning-based approaches generally demonstrate faster processing compared with traditional optimization-based methods. As shown in Table 7, HSIDwRD achieves the shortest running time of 2.03 s, while the proposed CDCN model requires approximately 5.11 s per image during inference. Based on this inference time, the corresponding processing speed is approximately 0.20 frames per second (FPS).

Although the proposed model does not strictly satisfy real-time processing requirements, it demonstrates competitive computational efficiency compared with several existing deep learning models while achieving superior denoising performance. The proposed CDCN method ranks among the top-performing approaches in terms of efficiency while delivering improved qualitative and quantitative results due to the integration of attention mechanisms and structural feature extraction. This indicates that the proposed framework achieves a favorable balance between computational cost and reconstruction quality.

[Table 7](#) also provides the model complexity of all the comparing methods and the proposed model in terms of parameter count and GFLOPs. It is proved that there is a trade-off between representational capacity and computational efficiency. Models such as MemNet, LRMR, and ENCAM have higher parameter counts, i.e., 2.90M, 2.78M, and 2.56M, respectively, which may enhance feature extraction capability but at the cost of increased computational burden. This is further reflected in their GFLOPs, where ENCAM and MemNet attain 6.12 and 5.84 GFLOPs, respectively.

In contrast, lightweight models such as HSI_{DWRD} and CDCN achieve remarkable efficiency. HSI_{DWRD} has the smallest number of parameters (0.25M), while CDCN requires the lowest computational cost (0.10 GFLOPs). Notably, these models maintain a favorable balance between complexity and efficiency, making them promising algorithms for HSI denoising. These observations show the importance of designing architectures that balance performance and efficiency, especially for practical applications where both memory and computational constraints are critical considerations.

6 Discussion

We performed the experiments on simulated and real datasets using the CDCN-based network instead of employing the simple vanilla convolutional layers. Seven different noise levels were used to train the model, and the performance was compared with eleven well-known classic, deep learning and transformer based algorithms using the same quality indexes, i.e., PSNR, SSIM, and SAM. We can conclude that CDCN performed better than the compared approaches, effectively capturing the details of the fine noise and retaining the structure of the HS image. The numerical results show that CDCN removes most of the noise during the noise-removing process. However, with random noise, CDCN showed lower SSIM values than the ENCAM algorithm. On the other side, CDCN got a clear superiority in terms of numeric results on all the comparing methods. Another point to discuss is related to the ablation study. Indeed, the proposed CDCN denoised the HSI better with the attention mechanism than the vanilla CNN with the attention module. The last point is instead about the model's efficiency in terms of inference efficiency, number of parameters, and computational cost (GFLOPs), computed in [Table 7](#). It can be noted that CDCN requires 5.11 s to run, the 4th best efficient denoising model among the compared ones. Nevertheless, It can be observed that the proposed CDCN demonstrates strong efficiency. In terms of parameter size, the model contains only 0.66M parameters, making it the third most lightweight model among all compared methods. Furthermore, the proposed approach achieves the lowest computational cost, requiring only 0.10 GFLOPs. Considering both qualitative and quantitative results, it is proved that the proposed method offers an excellent balance between performance and efficiency, making it a more effective for HSI denoising. Meanwhile, we can also check that the model is stable during training, as the loss graph decreases smoothly. The loss curve is shared in [Fig. 24](#).

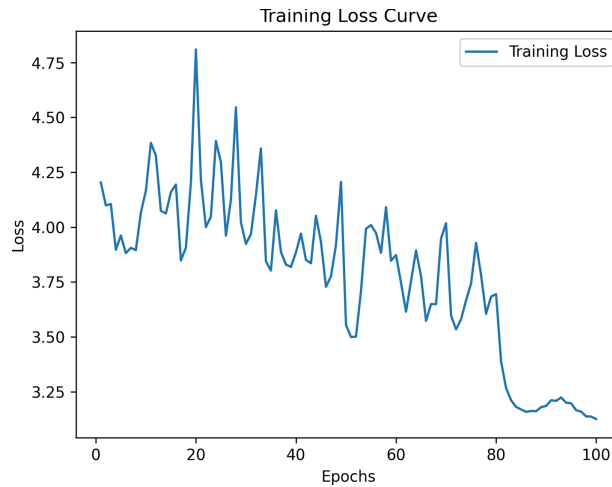


Figure 24: Proposed CDCN model training loss curve.

7 Conclusions

This research proposed a deep-learning-based architecture to reduce noise in HSIs. CNN-based methods provide deep semantic features, but these networks cannot extract fine-grained information in environmentally sensitive cases. HSIs are ecologically sensitive. We presented a novel center difference convolution network to address such issues, exploiting both spatial and spectral information of HSIs. The proposed method extracted features by giving a weightage to the center of the image because most of the information is present there. Therefore, the proposed model extracted features using center difference convolutions. Furthermore, this research utilizes an attention mechanism with CDCN to refine the HSI features in the spatial-spectral domain. The results are further validated using an ablation study and a computational analysis, pointing out that our solution is also cost-efficient.

One major limitation of the proposed framework is that it is evaluated primarily on publicly available datasets, which may not fully represent the diversity and variability encountered in real-world scenarios. The proposed framework achieves an inference speed of approximately 0.20 FPS with the current implementation, which is not suitable for real-world deployment. We will aim to make the proposed framework lightweight. Secondly, although the model demonstrates strong predictive capability, deep learning models may still suffer from limited interpretability, which can affect their practical adoption in certain application domains.

Future work will focus on addressing these limitations by extending the framework to larger and more diverse datasets collected from multiple sources to improve generalization capability. In addition, future studies may explore lightweight model architectures and model compression techniques to reduce computational complexity and facilitate real-time deployment. Lastly, incorporating advanced explainable artificial intelligence (XAI) techniques could improve the interpretability of the model.

Acknowledgement: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00218176) and the Soonchunhyang University Research Fund.

Funding Statement: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2023-00218176) and the Soonchunhyang University Research Fund.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Muhammad Umer; methodology, Mahmood Ashraf, Muhammad Umer and Raed Alharthi; software, Mahmood Ashraf, Nuha

Zamzami, Muhammad Umer, Shtwai Alsubai and Raed Alharthi; validation, Yunyoung Nam; formal analysis, Yunyoung Nam and Yongwon Cho; investigation, Nuha Zamzami; data curation, Muhammad Umer and Shtwai Alsubai; writing—original draft preparation, Mahmood Ashraf, Muhammad Umer, Nuha Zamzami and Shtwai Alsubai; writing—review and editing, Raed Alharthi, Yunyoung Nam and Yongwon Cho; visualization, Mahmood Ashraf; supervision, Yunyoung Nam and Yongwon Cho; project administration, Yunyoung Nam and Yongwon Cho; funding acquisition, Yunyoung Nam and Yongwon Cho. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets utilized in this research are publicly available and can be accessed via the URL:

1. KSC: [https://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Kennedy_Space_Center_\(KSC\)](https://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Kennedy_Space_Center_(KSC)).
2. PC: https://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Pavia_Centre_scene
3. HT: <https://github.com/YuxiangZhang-BIT/Data-CSHSI?tab=readme-ov-file>

The specific details regarding the datasets, including any necessary information on how to access or cite them, can be found at the provided URL(s). All data used in this study can be freely obtained from the aforementioned source.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Harsanyi JC, Chang C-I. Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *IEEE Trans Geosci Remote Sens.* 1994;32(4):779–85.
2. Su H, Cai Y, Du Q. Firefly-algorithm-inspired framework with band selection and extreme learning machine for hyperspectral image classification. *IEEE J Sel Top Appl Earth Obs Remote Sens.* 2016;10(1):309–20. doi:10.1109/jstars.2016.2591004.
3. Su H, Du Q, Chen G, Du P. Optimized hyperspectral band selection using particle swarm optimization. *IEEE J Sel Top Appl Earth Obs Remote Sens.* 2014;7(6):2659–70. doi:10.1109/jstars.2014.2312539.
4. Plaza A, Benediktsson JA, Boardman JW, Brazile J, Bruzzone L, Camps-Valls G, et al. Recent advances in techniques for hyperspectral image processing. *Remote Sens Environ.* 2009;113:S110–22. doi:10.1016/j.rse.2007.07.028.
5. Lu X, Wang Y, Yuan Y. Graph-regularized low-rank representation for destriping of hyperspectral images. *IEEE Trans Geosci Remote Sens.* 2013;51(7):4009–18. doi:10.1109/tgrs.2012.2226730.
6. Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans Image Process.* 2007;16(8):2080–95. doi:10.1109/tip.2007.901238.
7. Gu S, Zhang L, Zuo W, Feng X. Weighted nuclear norm minimization with application to image denoising. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* Piscataway, NJ, USA: IEEE; 2014. p. 2862–9.
8. Ye M, Qian Y, Zhou J. Multitask sparse nonnegative matrix factorization for joint spectral-spatial hyperspectral imagery denoising. *IEEE Trans Geosci Remote Sens.* 2014;53(5):2621–39. doi:10.1109/tgrs.2014.2363101.
9. Chen Y, Cao X, Zhao Q, Meng D, Xu Z. Denoising hyperspectral image with non-iid noise structure. *IEEE Trans Cybern.* 2017;48(3):1054–66. doi:10.1109/tcyb.2017.2677944.
10. Xie W, Li Y. Hyperspectral imagery denoising by deep learning with trainable nonlinearity function. *IEEE Geosci Remote Sens Lett.* 2017;14(11):1963–7. doi:10.1109/lgrs.2017.2743738.
11. Chang Y, Yan L, Fang H, Zhong S, Liao W. Hsi-denet: hyperspectral image restoration via convolutional neural network. *IEEE Trans Geosci Remote Sens.* 2018;57(2):667–82.
12. Yuan Q, Zhang Q, Li J, Shen H, Zhang L. Hyperspectral image denoising employing a spatial-spectral deep residual convolutional neural network. *IEEE Trans Geosci Remote Sens.* 2018;57(2):1205–18. doi:10.1109/tgrs.2018.2865197.
13. Bodrito T, Zouaoui A, Chanussot J, Mairal J. A trainable spectral-spatial sparse coding model for hyperspectral image restoration. *Adv Neural Inf Process Syst.* 2021;34:5430–42.

14. Dong W, Wang H, Wu F, Shi G, Li X. Deep spatial-spectral representation learning for hyperspectral image denoising. *IEEE Trans Comput Imaging*. 2019;5(4):635–48. doi:10.1109/tci.2019.2911881.
15. Cao X, Fu X, Xu C, Meng D. Deep spatial-spectral global reasoning network for hyperspectral image denoising. *IEEE Trans Geosci Remote Sens*. 2021;60:55047114. doi:10.1109/tgrs.2021.3069241.
16. Zhang Q, Yuan Q, Li J, Liu X, Shen H, Zhang L. Hybrid noise removal in hyperspectral imagery with a spatial-spectral gradient network. *IEEE Trans Geosci Remote Sens*. 2019;57(10):7317–29. doi:10.1109/tgrs.2019.2912909.
17. Wei K, Fu Y, Huang H. 3-D quasi-recurrent neural network for hyperspectral image denoising. *IEEE Trans Neural Netw Learn Syst*. 2020;32(1):363–75. doi:10.1109/TNNLS.2020.2978756.
18. Deng C, Li L, He Z, Li J, Zhu Y. Monte carlo non-local means method for hyperspectral image denoising. In: *Proceedings of the 2018 IEEE International Geoscience and Remote Sensing Symposium*. Piscataway, NJ, USA: IEEE; 2018. p. 4772–5.
19. Zhang R, Yang L, Liu G, Feng X. Weighted nuclear norms of transformed tensors for nonlocal hyperspectral image denoising. *IEEE Geosci Remote Sens Lett*. 2022;19:16011305. doi:10.1109/lgrs.2022.3186877.
20. Wang Z, Ng MK, Zhuang L, Gao L, Zhang B. Nonlocal self-similarity-based hyperspectral remote sensing image denoising with 3D convolutional neural network. *IEEE Trans Geosci Remote Sens*. 2022;60:5531617.
21. Yuan Q, Zhang L, Shen H. Hyperspectral image denoising employing a spectral-spatial adaptive total variation model. *IEEE Trans Geosci Remote Sens*. 2012;50(10):3660–77. doi:10.1109/tgrs.2012.2185054.
22. Bourennane S, Fossati C, Juan J. Improvement of classification based on noise and spectral dimensionality reduction for hyperspectral image. *Geosci Remote Sens*. 2018;1(1):9–17. doi:10.23977/geors.2018.11012.
23. Zhang H, He W, Zhang L, Shen H, Yuan Q. Hyperspectral image restoration using low-rank matrix recovery. *IEEE Trans Geosci Remote Sens*. 2013;52(8):4729–43. doi:10.1109/tgrs.2013.2284280.
24. Xie S, Wang S, Song C, Wang X. Hyperspectral image reconstruction based on spatial-spectral domains low-rank sparse representation. *Remote Sens*. 2022;14(17):4184. doi:10.3390/rs14174184.
25. Chen Y, Huang T-Z, He W, Zhao X-L, Zhang H, Zeng J. Hyperspectral image denoising using factor group sparsity-regularized nonconvex low-rank approximation. *IEEE Trans Geosci Remote Sens*. 2021;60:5515916.
26. Gao Q, Lim S, Jia X. Spectral-spatial hyperspectral image classification using a multiscale conservative smoothing scheme and adaptive sparse representation. *IEEE Trans Geosci Remote Sens*. 2019;57(10):7718–30. doi:10.1109/tgrs.2019.2915809.
27. Li J, Yuan Q, Shen H, Zhang L. Noise removal from hyperspectral image with joint spectral-spatial distributed sparse representation. *IEEE Trans Geosci Remote Sens*. 2016;54(9):5425–39. doi:10.1109/tgrs.2016.2564639.
28. Renard N, Bourennane S, Blanc-Talon J. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geosci Remote Sens Lett*. 2008;5(2):138–42. doi:10.1109/lgrs.2008.915736.
29. Fan H, Li C, Guo Y, Kuang G, Ma J. Spatial-spectral total variation regularized low-rank tensor decomposition for hyperspectral image denoising. *IEEE Trans Geosci Remote Sens*. 2018;56(10):6196–213.
30. Zhao Y-Q, Yang J. Hyperspectral image denoising via sparse representation and low-rank constraint. *IEEE Trans Geosci Remote Sens*. 2014;53(1):296–308. doi:10.1109/igarss.2013.6721354.
31. Letexier D, Bourennane S. Noise removal from hyperspectral images by multidimensional filtering. *IEEE Trans Geosci Remote Sens*. 2008;46(7):2061–9.
32. Othman H, Qian S-E. Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage. *IEEE Trans Geosci Remote Sens*. 2006;44(2):397–408. doi:10.1109/tgrs.2005.860982.
33. Atkinson I, Kamalabadi F, Jones DL. Wavelet-based hyperspectral image estimation. In: *Proceedings of the 2003 IEEE International Geoscience and Remote Sensing Symposium*. Piscataway, NJ, USA: IEEE; 2003. p. 743–5.
34. Rasti B, Sveinsson JR, Ulfarsson MO, Benediktsson JA. Hyperspectral image denoising using first order spectral roughness penalty in wavelet domain. *IEEE J Sel Top Appl Earth Obs Remote Sens*. 2013;7(6):2458–67. doi:10.1109/jstars.2013.2272879.
35. Ashraf M, Chen L, Zhou X, Rakha MA. A joint architecture of mixed-attention transformer and octave module for hyperspectral image denoising. *IEEE J Sel Top Appl Earth Obs Remote Sens*. 2024;17:4331–49.

36. Wambugu N, Chen Y, Xiao Z, Tan K, Wei M, Liu X, et al. Hyperspectral image classification on insufficient-sample and feature learning using deep neural networks: a review. *Int J Appl Earth Obs Geoinf*. 2021;105(12):102603. doi:10.1016/j.jag.2021.102603.
37. Sellami A, Tabbone S. Deep neural networks-based relevant latent representation learning for hyperspectral image classification. *Pattern Recognit*. 2022;121(1):108224. doi:10.1016/j.patcog.2021.108224.
38. Das R, Singh TD. Assamese news image caption generation using attention mechanism. *Multimed Tools Appl*. 2022;81(7):10051–69. doi:10.1007/s11042-022-12042-8.
39. Chen L, Lai Z, Vivone G, Jeon G, Chanussot J, Yang X. ArbRPN: a bidirectional recurrent pansharpening network for multispectral images with arbitrary numbers of bands. *IEEE Trans Geosci Remote Sens*. 2022;60:5406418.
40. Chen L, Vivone G, Nie Z, Chanussot J, Yang X. Spatial data augmentation: improving the generalization of neural networks for pansharpening. *IEEE Trans Geosci Remote Sens*. 2023;61:5401711.
41. Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a gaussian denoiser: residual learning of deep cnn for image denoising. *IEEE Trans Image Process*. 2017;26(7):3142–55.
42. Maffei A, Haut JM, Paoletti ME, Plaza J, Bruzzone L, Plaza A. A single model CNN for hyperspectral image denoising. *IEEE Trans Geosci Remote Sens*. 2019;58(4):2516–29.
43. Kan Z, Li S, Hou M, Fang L, Zhang Y. Attention-based octave network for hyperspectral image denoising. *IEEE J Sel Top Appl Earth Obs Remote Sens*. 2021;15:1089–1102.
44. Wang Z, Shao Z, Huang X, Wang J, Lu T. SSCAN: a spatial-spectral cross attention network for hyperspectral image denoising. *IEEE Geosci Remote Sens Lett*. 2021;19:5508805.
45. Gong Z, Gao F, Dong J, Qi L. Hyperspectral image denoising based on parallel cross-fusion network. In: *Proceedings of the 2022 IEEE International Geoscience and Remote Sensing Symposium*. Piscataway, NJ, USA: IEEE; 2022. p. 1528–31.
46. Murugesan R, Nachimuthu N, Prakash G. Attention based deep convolutional U-Net with CSA optimization for hyperspectral image denoising. *Infrared Phys Technol*. 2023;129(10):104531. doi:10.1016/j.infrared.2022.104531.
47. Lai Z, Fu Y. Mixed attention network for hyperspectral image denoising. arXiv:2301.11525. 2023.
48. Duan P, Luo Y, Kang X, Li S. Lamamba: linear attention mamba for hyperspectral image denoising. *IEEE Trans Geosci Remote Sens*. 2025;63:5527113.
49. Nachimuthu N, Murugesan R, Dharmalingam M, Prakash G. Revolutionizing hyper spectral image denoising: a squeeze-net paradigm. *Sci Rep*. 2026;16:7419.
50. Yu Z, Qin Y, Li X, Wang Z, Zhao C, Lei Z. Multi-modal face anti-spoofing based on central difference networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. Piscataway, NJ, USA: IEEE; 2020. p. 2766–74.
51. Woo S, Park J, Lee J-Y, Kweon IS. CBAM: convolutional block attention module. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Cham, Switzerland: Springer; 2018. p. 3–19.
52. Shi Q, Tang X, Yang T, Liu R, Zhang L. Hyperspectral image denoising using a 3-D attention denoising network. *IEEE Trans Geosci Remote Sens*. 2021;59(12):10348–63.
53. Jin KH, McCann MT, Froustey E, Unser M. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans Image Process*. 2017;26(9):4509–22. doi:10.1109/tip.2017.2713099.
54. Wei Y, Yuan Q, Shen H, Zhang L. Boosting the accuracy of multispectral image pansharpening by learning a deep residual network. *IEEE Geosci Remote Sens Lett*. 2017;14(10):1795–9. doi:10.1109/lgrs.2017.2736020.
55. Tai Y, Yang J, Liu X, Xu C. MemNet: a persistent memory network for image restoration. In: *Proceedings of the IEEE International Conference on Computer Vision*. Piscataway, NJ, USA: IEEE; 2017. p. 4549–57.
56. Liu Y, Anwar S, Zheng L, Tian Q. GradNet image denoising. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. Piscataway, NJ, USA: IEEE; 2020. p. 2140–9.
57. Ma H, Liu G, Yuan Y. Enhanced non-local cascading network with attention mechanism for hyperspectral image denoising. In: *Proceedings of the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Piscataway, NJ, USA: IEEE; 2020. p. 2448–52.
58. Zhang T, Fu Y, Li C. Hyperspectral image denoising with realistic data. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Piscataway, NJ, USA: IEEE; 2021. p. 2228–37.

59. Li M, Fu Y, Zhang Y. Spatial-spectral transformer for hyperspectral image denoising. *Proc AAAI Conf Artif Intell.* 2023;37(1):1368–76. doi:10.1609/aaai.v37i1.25221.
60. Li M, Liu J, Fu Y, Zhang Y, Dou D. Spectral enhanced rectangle transformer for hyperspectral image denoising. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* Piscataway, NJ, USA: IEEE; 2023. p. 5805–14.
61. Chen Z, Liu C, Zhou J. SSIT: a spatial-spectral interactive transformer for hyperspectral image denoising. *Sci Remote Sens.* 2025;12:100276.
62. Su H, Zhao B, Du Q, Du P, Xue Z. Multifeature dictionary learning for collaborative representation classification of hyperspectral imagery. *IEEE Trans Geosci Remote Sens.* 2018;56(4):2467–84. doi:10.1109/tgrs.2017.2781805.
63. Le Saux B, Yokoya N, Hansch R, Prasad S. 2018 IEEE GRSS data fusion contest: multimodal land use classification [technical committees]. *IEEE Geosci Remote Sens Mag.* 2018;6(1):52–4.