



## Article

# Reconstruction of Compressed Hyperspectral Image Using SqueezeNet Coupled Dense Attentional Net

Divya Mohan <sup>1,\*</sup>, J. Aravinth <sup>1,\*</sup> and Sankaran Rajendran <sup>2</sup>

<sup>1</sup> Department of Electronics and Communication Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore 641112, India; m\_divya1@cb.students.amrita.edu

<sup>2</sup> Environmental Science Center, Qatar University, P.B. No. 2713, Doha 2713, Qatar; srajendran@qu.edu.qa

\* Correspondence: j\_aravinth@cb.amrita.edu

**Abstract:** This study addresses image denoising alongside the compression and reconstruction of hyperspectral images (HSIs) using deep learning techniques, since the research community is striving to produce effective results to utilize hyperspectral data. Here, the SqueezeNet architecture is trained with a Gaussian noise model to predict and discriminate noisy pixels of HSI to obtain a clean image as output. The denoised image is further processed by the tunable spectral filter (TSF), which is a dual-level prediction filter to produce a compressed image. Subsequently, the compressed image is analyzed through a dense attentional net (DAN) model for reconstruction by reverse dual-level prediction operation. All the proposed mechanisms are employed in Python and evaluated using a Ben-Gurion University-Interdisciplinary Computational Vision Laboratory (BGU-ICVL) dataset. The results of SqueezeNet architecture applied to the dataset produced the denoised output with a Peak Signal to Noise Ratio (PSNR) value of 45.43 dB. The TSF implemented to the denoised images provided compression with a Mean Square Error (MSE) value of 8.334. Subsequently, the DAN model executed and produced reconstructed images with a Structural Similarity Index Measure (SSIM) value of 0.9964 dB. The study proved that each stage of the proposed approach resulted in a quality output, and the developed model is more effective to further utilize the HSI. This model can be well utilized using HSI data for mineral exploration.

**Keywords:** hyperspectral image compression; denoising; reconstruction; deep learning; tunable spectral filter; SqueezeNet; dense attentional net; dense blocks; dual-level prediction



**Citation:** Mohan, D.; Aravinth, J.; Rajendran, S. Reconstruction of Compressed Hyperspectral Image Using SqueezeNet Coupled Dense Attentional Net. *Remote Sens.* **2023**, *15*, 2734. <https://doi.org/10.3390/rs15112734>

Academic Editor: Chiman Kwan

Received: 17 April 2023

Revised: 20 May 2023

Accepted: 22 May 2023

Published: 24 May 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Recently, the emergence of imaging information at high spatial and spectral resolutions has been drastically increased in remote sensing techniques [1–3]. Hyperspectral cameras and sensors collect information over the electromagnetic spectrum and produce hyperspectral images (HSIs) at a high spectral resolution of less than 10 nm [4]. For example, the airborne hyperspectral sensors, namely the Airborne Visible InfraRed Imaging Spectrometer (AVIRIS), store 224 bands between 400 and 2500 nm, and the Hyperspectral Digital Imagery Collection Experiment (HYDICE) dataset collects 210 spectral bands in between 400 and 2500 nm at a 10 nm bandwidth [5,6]. In addition, the HYPERION spaceborne hyperspectral sensor acquires 220 bands from 400 to 2500 nm with 30 m spatial resolution. All these data are highly correlatable and have huge volumes. These data can be recorded and maintained only by using effective compression techniques [7]. The high spectral information in HSI represent the deterministic details about the material and lighting [8,9]. Due to this characteristic, HSI has been well utilized in several fields for environment monitoring, anomaly detection [10], object recognition and classification, etc. [11–16]. However, studies show that the HSIs are affected by noise due to the cause of dark current and thermal electronics. The noises in HSI include impulse noise, Gaussian noise and sparse noise [17]. Huang et al. [18] introduced a Spatial–Spectral Weighted Nuclear Norm Minimization

(SSWNNM) model for HSI denoising. In this model, non-local similar cubic patches were identified and stacked into a low-rank (LR) matrix which is composed of spatial texture information. They used the multiple channels Weighted Nuclear Norm Minimization (WNNM) for recovering the spatial LR matrix. The experimentation was carried out on the HYDICE datasets for Urban and Indian Pines. They stated that the proposed SSWNNM achieved a better Mean Peak Signal to Noise Ratio (MPSNR) of 43.58, Mean Structural Similarity Index Measure (MSSIM) of 0.973, and Mean Feature Similarity Index Measure (MFSIM) of 0.989, respectively, on the Indian Pines dataset. Recently, Zeng et al. [19] presented a model for Global Spatial Spectral Total Variation (GS-STV) for HSI denoising. Here, GS-STV was applied into the LR tensor approach. LR was utilized for separating the HSI from the sparse noise, and GS-STV was used simultaneously for removing Gaussian noise and considered both spatial and spectral correlation. The quantitative measures of Picture Quality Indices (PQIs) were evaluated on the HYDICE and Indian Pines datasets. The performance was taken for varying the noise level, and the proposed GS-STV achieved a better MPSNR of 38.78 and MSSIM of 0.956 on the Indian Pines dataset in case 2. This model was not suitable for denoising heavy Gaussian noise, as it can only boost the local details and overall structural information in HSIs.

Thus, as stated above, the noise affects the HSI visual quality and minimizes the accuracy during image classification. It is essential to carry out HSI denoising as a pre-processing process for improving the quality of images before the interpretation and classification process [20]. The HSI data have a high amount of unessential information, and it is challenging for the transmission and storage of images. It is important to design an efficient model for HSI compression [21], which is the process of compressing the HSI without any retardation in its quality. Moreover, the reconstruction or restoration of HSI, the process of obtaining the original images from the distorted images, is also essential [22]. The reconstruction of HSI aims to recover the 3D spatial–spectral image from 2D measurement. In the last two decades, due to the advancement of the deep learning (DL) model, the convolutional neural network (CNN) has attained a lot of achievements in the applications of pattern recognition and computer vision. Especially, the DL models are highly used for the reconstruction of HSI [23]. However, the conventional DL models suffer from large model sizes, which leads to high training time, less flexibility and requires large memory [24]. Chong et al. [25] studied HSI compression and reconstruction using Block-Sparse Dictionary (B-SD) Learning model. In this study, training was completed using a set of signals. Then, the measurement matrix was used for compressing the HSI cube for reducing the volume of data. The performance of B-SD was compared with the University of Pavia and center datasets. The experimentation was carried out by varying the compressive sampling ratio (CSR) values and achieved better PSNR values of 19.99 and 25.91 for the CSR values of 0.05 and 0.10, respectively. Li et al. [26] introduced HSI compression by using a Correlation–Tucker decomposition (C-TD) model. C-TD was used for constructing the factor matrix, and hence, the dimensionality of the core tensor was determined. The proposed C-TD can be used in any TD model of order tensor. However, this model takes more time for processing and achieved a better PSNR value of 52.12 for the bit rate of 0.2.

Zikiou et al. [27] introduced HSI compression using 3D-discrete wavelet transform (3D-DWT) and support vector machine (SVM). In this work, both air and spaceborne sensors were considered, and performances were compared on the basis of spectral fidelity and rate distortion. This model achieved a better PSNR of 46.6 dB. The classification accuracy of 75.8 percent was found for all decoded HSI images. Wang et al. [28] introduced the CNN model for reconstructing 3D-compressed HSI images using a back-strapping process. The experimentation proved that the back-strapping network compressed the HSI correctly and quickly. This model also achieved a better MPSNR of 31.43 and MSSIM of 0.935 for scene 1 on the BGU iCVL dataset. This study proposes an enhanced deep learning-based model for effective denoising, compressing and reconstructing of HSI.

The major contributions of the work are as follows:

- (1) To build an effective framework that can denoise, compress and reconstruct an HSI in order to attain quality outputs compared with the other reconstruction mechanisms.
- (2) To introduce a dual-level prediction-based tunable spectral filter (TSF) for compressing the HSI input. The model predicts the pixel values and adopts the thresholding strategy to compare the pixel with a reference value.
- (3) To present a new and effective dense attentional net (DAN) model to reconstruct the compressed image. The model learns the reverse operation of dual-level prediction to obtain the reconstructed HSI.
- (4) Extending evaluations of the model in terms of different metrics to prove the performance improvement of the proposed model compared to the other existing state-of-the-arts models.

## 2. Materials and Methods

### 2.1. Simulation Scenario and Hyperspectral Image Data Sets

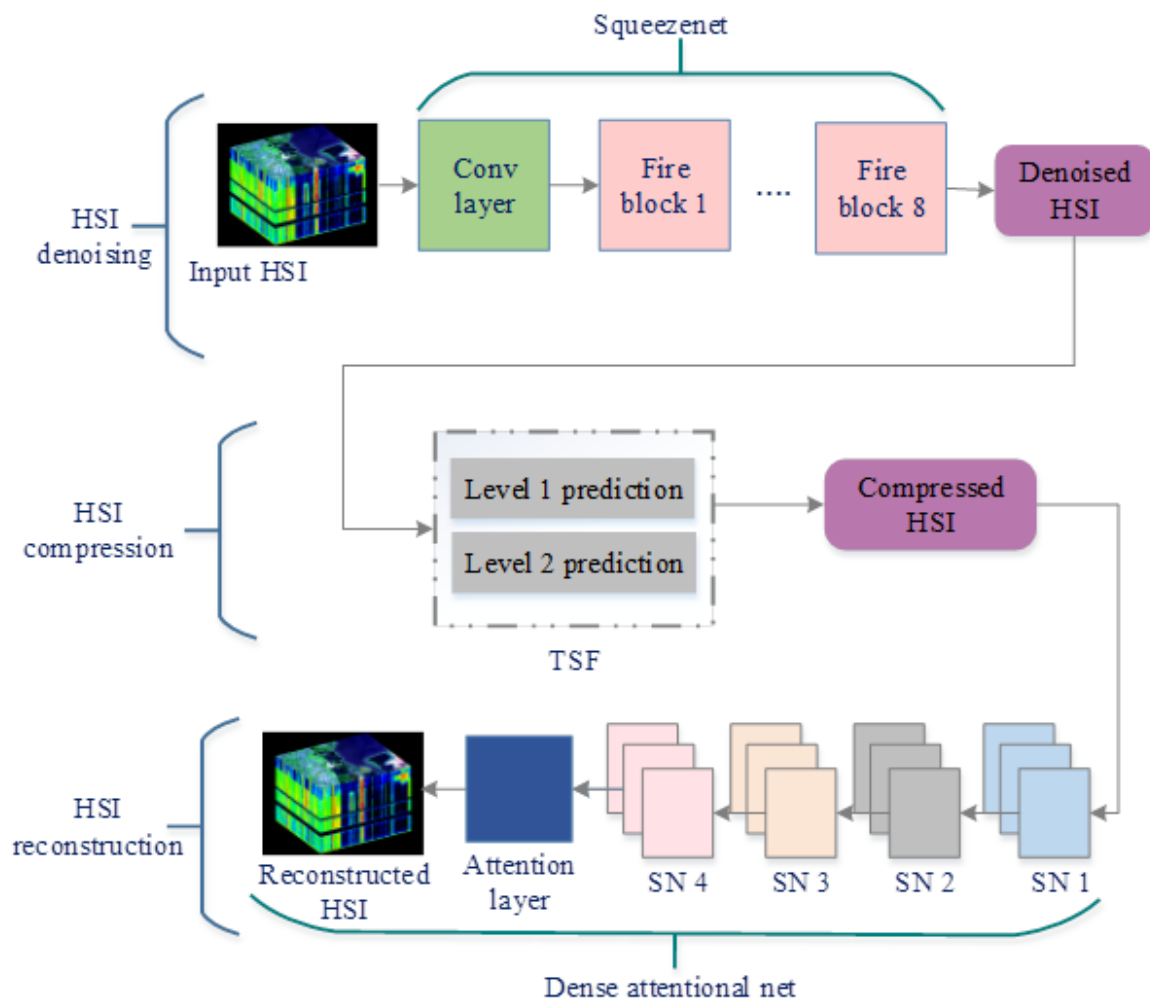
For evaluations of the proposed approach, the Ben-Gurion University-Interdisciplinary Computational Vision Laboratory (BGU-ICVL) hyperspectral image dataset [29] has been utilized. This dataset is acquired by a Specim PS Kappa DX4 hyperspectral camera and publicly available. It consists of 519 spectral bands ranging between 400 and 1000 nm in the electromagnetic spectrum. The spectral interval between the images is about 1.25 nm. The image has 12-bit radiometric resolution. The BGU-ICVL stored the data in .rgb and .mat files formats with a header file information at the .hdr file format. In this study, sample images that consist of different features including the building glass, grass with buildings, constructed buildings, agriculture field and plantation were downloaded from the BGU-ICVL (Figure 1) so that all the different features can be brought under the study of compression and reconstruction and evaluated by the following proposed methods. The files downloaded consisted of clean HSI, and additional Gaussian noise are added manually to perform denoising and train the proposed model. The entire implementation of the work is carried out using the Python simulation environment.



Figure 1. HSI images of BGU-ICVL dataset.

## 2.2. Deep Learning Methods

In general, deep learning-based methodologies are found to be more effective in dealing with various applications such as image detection [30], image classification and especially image reconstruction processes. In this study, the denoising, compression and reconstruction of HSI is carried out in the vision to build an effective model and to attain higher quality outputs than the other existing models. The SqueezeNet model is introduced initially to denoise the HSI to produce quality images. Then, the denoised images are passed through the tunable spectral filter (TSF) to produce compressed images. The proposed filtering technique is prediction based to obtain results in effective compression without the reduction of its quality. Finally, the compressed image is provided as the input to the dense attentional net (DAN) model to reconstruct the HSI. The proposed framework is named SqueezeNet coupled DAN (SDANet), and the architecture of the proposed framework is displayed in Figure 2. The detailed descriptions of the three main stages of the framework are given in the following sections. The entire implementations of the work are carried out using the Python simulation environment.



**Figure 2.** Architecture of the proposed SDANet framework.

### 2.2.1. HSI Denoising

The first step of the proposed work is HSI denoising, where the input images from the dataset are denoised to remove the noise present in the image. This step enhances the quality of the image and results in improved performance. To perform denoising, the SqueezeNet [31] model is utilized in the proposed work. This model is an extension of the traditional convolutional neural network (CNN) with enhanced extraction capability.

The SqueezeNet architecture comprises fire blocks that enhance the extraction capability of the network. Here, the first convolution layer receives the input and extracts the major features, and then, the maxpooling operation is carried out. Then, the features are provided to the fire blocks where the discrimination of the features takes place to identify the noisy pixels. One of the main advantages of the SqueezeNet model compared to the traditional CNN is that it uses fewer parameters and maintains competitive accuracy. This reduces the training time of the network and improves the overall efficiency. The SqueezeNet-based HSI denoising model [32] of this study is given in Figure 3a. The SqueezeNet architecture comprises 2 convolutional layers with different numbers of convolutional kernels, 8 fire blocks, 3 max pooling layers and 1 global average pooling layer at the end (Figure 3a). The three main strategies followed in the SqueezeNet are used to reduce the number of parameters and to maintain better accuracy. In this model, the  $3 \times 3$  filters are replaced by  $1 \times 1$  filters, and the number of input channels are reduced to  $3 \times 3$  filters. Here, the activation maps of the convolutional layers are maintained large by performing late downsampling. This step is developed to improve overall accuracy. The fire block in the SqueezeNet model is considered as a fundamental unit and is composed of squeeze and expand layers with convolutional filters. The model of fire block is shown in Figure 3b. The SqueezeNet denoising model is compared with the U-Net+GSM+GCM model [32] to prove the effectiveness of the proposed method. The U-Net+GSM+GCM model contains two feature extraction layers, four encoding units, three decoding units and one reconstruct layer. Although this model achieves excellent performance, the design of the network still lacks interpretation. So, we have focussed on designing an interpretable network deduced from the traditional HSI denoising method, and thus, each layer of the proposed network has its own physical interpretation. The design and implementation is also more effective in case of the proposed model.

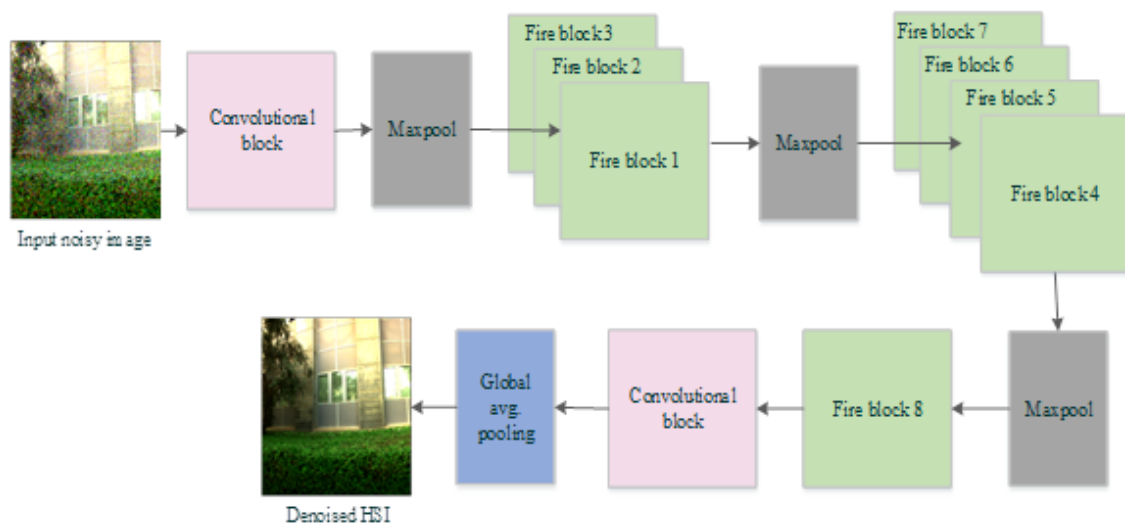
Throughout the model, the rectified linear unit (ReLU) activation is carried out. While reaching the first fire block, the squeeze layer applies the  $1 \times 1$  filter over the input and extracts the crucial features. Then, the expand layer applies the  $1 \times 1$  and  $3 \times 3$  filters, resulting in expanded output with more depth. The squeeze operation performs compression, whereas the expand operation enhances the depth without any changes in the feature size. Then, the outputs of the expand layer are concatenated and then passed onto the successive blocks. The squeeze and expand operations are repeated for every fire block, and the overall output is sent to the global average pooling layer to obtain the final denoised image. The mathematics of squeeze operation carried out in the fire block is provided in Equation (1).

$$S_q(y) = \sum_{\mu=1}^{FM} \sum_{\iota=1}^C w_{\iota}^S x_{\iota}^{\mu} \quad (1)$$

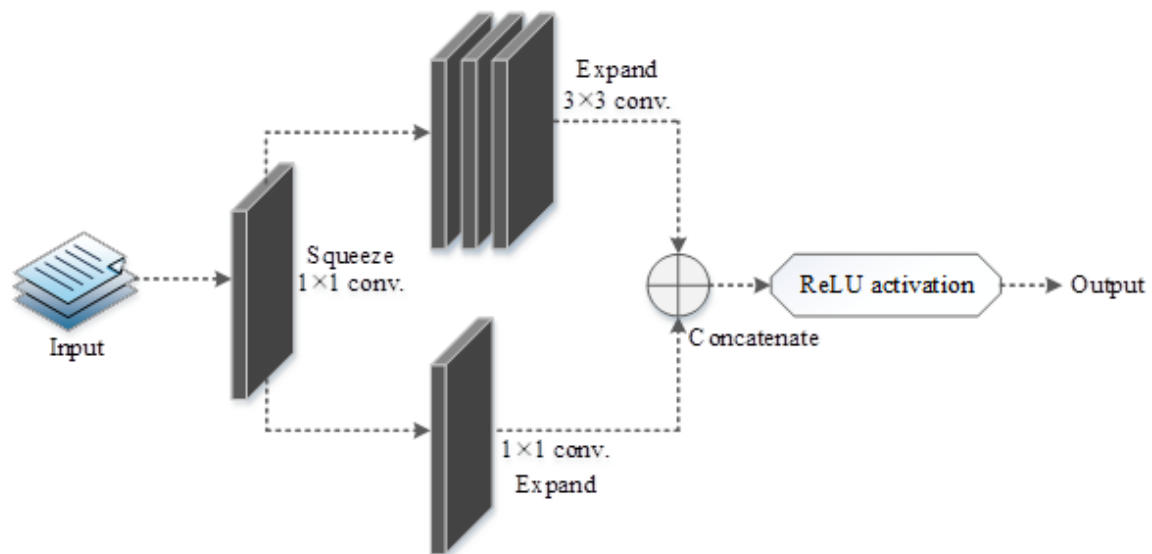
where  $FM$  indicates the feature maps,  $C$  indicates the channels,  $w_{\iota}^S$  is the weight associated with the  $\iota$ th channel, and  $x_{\iota}^{\mu}$  is the input associated with the  $\mu$ th feature map. The outputs of the squeeze operations are the weighted combinations of the feature maps. The ReLU activation carried out in the model can be mathematically expressed as shown in Equation (2).

$$R_l(x) = \begin{cases} x; & x \geq 0 \\ 0; & x < 0 \end{cases} \quad (2)$$

where  $R_l(x)$  indicates the output of the ReLU activation function corresponding to the input  $x$ . After extraction of the features from the input HSI, the SqueezeNet model evaluates and removes the noisy pixels. Both the spatial and spectral information of the HSI images are extracted and learned by the model to perform denoising. The concatenation operation in the fire blocks is responsible for concatenating both the spatial and spectral features into one to provide the multi-level feature representation. The denoised output received from the SqueezeNet model is sent to the HSI compression stage.



a)



b)

**Figure 3.** (a) The SqueezeNet-based HSI denoising model. (b) The model of fire block.

### 2.2.2. HSI Compression

For compressing [33,34] the input HSI, this model introduces the predicted-based TSF method. In this method, an extension of the prediction-based recursive least square (RLS) filter [35] is employed over the denoised image that are obtained from the SqueezeNet model. The high computation time of RLS filter makes it less effective. However, to avoid this, a dual-level traversal scheme is embedded within the method to obtain the prediction value. The prediction is carried out two times (as dual-level traversal scheme), where the RLS is initially applied, and then, a backward traversal takes place to provide the final prediction value as described below.

(i) Level 1 prediction: In level 1, the RLS filtering technique is applied to the first input of HSI to obtain the corresponding prediction value. Here, the RLS filter predicts the intra-band in the HSI by spatial decorrelation, which removes incoherent spatial information from the single band of an HSI. In detail, the steps that are carried out in the removal of spatial correlations follow: Step 1: Compute the arithmetic mean of the pixels of the image in its local neighborhood based on Equation (3)

$$M(p, q, r) = [v(p, q - 1, r) + v(p - 1, q, r) + v(p - 1, q + 1, r) + v(p - 1, q - 1, r)]/4 \quad (3)$$

where  $(p, q, r)$  indicates the pixel's current value on the  $r^{th}$  band at  $(p, q)$ .

Step 2: Subtract the computed mean in Equation (3) from the original pixel value at position  $(p, q, r)$  using Equation (4).

$$v'(p, q, r) = v(p, q, r) - M(p, q, r) \quad (4)$$

Step 3: The process is repeated for all the bands in HSI leading to the formation of a 3D matrix of size  $(\mathfrak{R} \times E \times \wp)$  where  $\mathfrak{R}$  indicates the total number of rows,  $E$  indicates the total columns in a band and  $\wp$  indicates the total number of bands in the considered input HSI. This matrix is then provided as the input to the next step.

(ii) Level 2 prediction: In level 2, the final prediction is obtained based on a traversal scheme. This prediction is carried out by traversing the casual pixels in the current band in the backward direction to obtain the best prediction value. Based on this activity, the prediction values of all the bands in the HSI are obtained, thereby resulting in a more accurate and effective compressed image output. Prediction based on the pixels in the current band provides more information regarding the HSI, and the spectral correlations in the image result in accurate compression. The pseudocode for level 2 prediction is provided in Table 1.

A starting point  $I_p$  is initially set, and the pixel values are traversed by keeping track of the prediction reference  $\Psi_{rf}$ . The outer loop is dedicated to the  $Y$  pixel lines considered in the traversal. The inner loop  $\Phi$  indicates the traversal within the pixel line. The traversed pixel value that is close in range to the prediction reference is tracked. A threshold value  $Th$  is set prior to the traversal of the pixel values to identify the prediction pixel value. Then, an error value  $\Psi_{err}$  between the two pixels is computed, and if the error value is close to the pre-computed threshold value, then the particular pixel is chosen as the predicted value. The setting up of threshold in the TSF helps to reduce the time taken for compression and results in early termination. In addition, the nearest neighbor pixels of the current pixel can be accurately obtained by the proposed technique, and hence, the prediction can produce better outcomes. Another significant advancement of the proposed TSF-based compression scheme is that the quality of the image even after compression remains the same due to the prediction of local neighbor pixels in the image.

The compression technique can be further utilised in the application of mineral mapping, as this method of dual prediction effectively compresses the HSI data compared to existing models, which is proved in the performance analysis conducted and explained in the later sections.

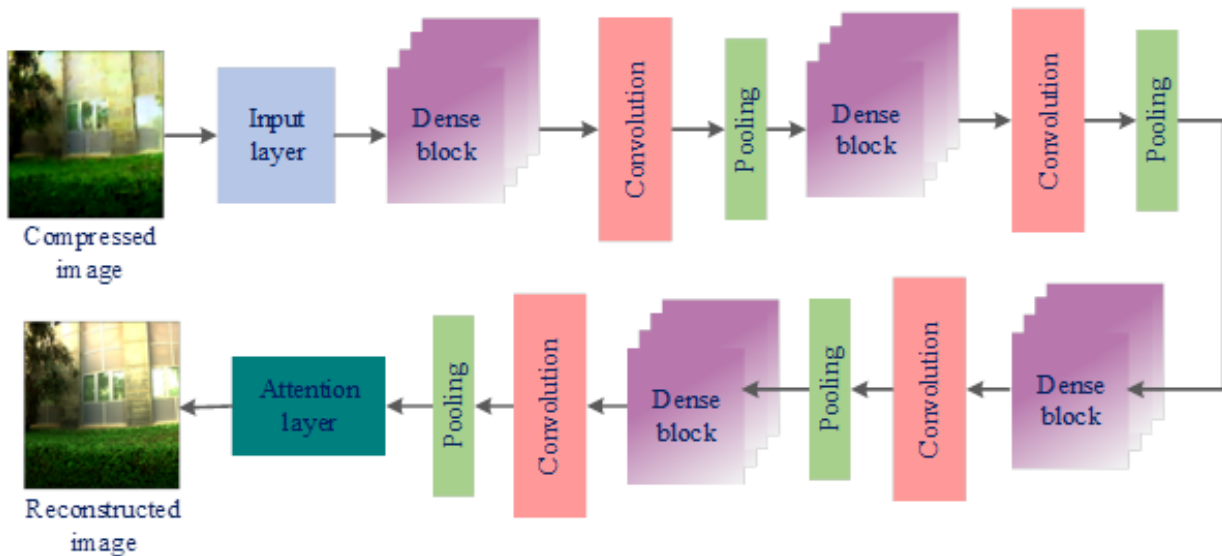
### 2.2.3. HSI Reconstruction

The third and most significant stage of the proposed approach is HSI reconstruction, where a new deep learning model called DAN is introduced and employed to achieve the desired results. The input to the reconstruction phase is the compressed image obtained from the last stage. The aim of the reconstruction model is to learn the reverse process of compression where the traversal process is initially reversed to obtain the spectral decorrelation. Here, the intra-band prediction process is reversed with the administration of initialization and advancement modules, and the fine details of the image are focused to maintain the spectral and spatial information present in it. Thus, the output of the reverse process is the reconstructed HSI with enriched spectral and spatial information. In this

study, the proposed DAN model is introduced for HSI reconstruction to obtain accurate and effective output. The model is composed of four sub-networks to perform the reverse process and to reconstruct the HSI. The four sub-networks are spatial initialization (SI), spatial advancement (SA), spectra initialization (SrI) and spatial–spectral advancement (SSrA). Each network is composed of dense blocks that are capable of learning the deep features of the image. The architecture of DAN used in reconstructing the HSI is shown in Figure 4.

**Table 1.** Pseudocode of level 2 prediction.

<p>Initialize: index of the current band <math>\beta</math>, current pixel coordinates <math>(p, q)</math>, image width <math>I_\omega</math>, prediction reference <math>\Psi_{rf}</math>, count of lines within the traversal boundary <math>Y</math></p> <p>Set the threshold value for traversal <math>m_{err} = 65536</math>;</p> <p>For <math>l = p</math> to <math>\max\{(p - Y), 0\}</math></p> <p style="padding-left: 20px;">{ if <math>(l == p) I_\rho = q - 1</math>;</p> <p style="padding-left: 40px;">else <math>I_\rho = I_\omega - 1</math> ;</p> <p style="padding-left: 20px;">For <math>\Phi = I_\rho</math> to 1</p> <p style="padding-left: 40px;"><math>\{\Psi_{err} = \text{abs}(\text{ImageBuffer}(\beta, l, \Phi) - \Psi_{rf})</math></p> <p style="padding-left: 60px;">if <math>(\Psi_{err} &lt; m_{err})</math></p> <p style="padding-left: 80px;"><math>m_{err} = \Psi_{err}</math> ;</p> <p style="padding-left: 40px;"><math>\varphi = \text{ImageBuffer}(\beta, l, \Phi)</math>;</p> <p style="padding-left: 20px;">if <math>(m_{err} \leq Th[\beta])</math></p> <p style="padding-left: 40px;">return <math>(\varphi)</math> ; }</p> <p style="padding-left: 20px;">return <math>(\varphi)</math> ; }</p>
---



**Figure 4.** DAN architecture for HSI reconstruction.

During this HSI reconstruction, the SI network receives the compressed image as the input, and the dense blocks within the network extracts and learns the features of the compressed image. The reverse process of compression starts from the SI sub-network, and the features learned by this network are then provided to the SA network. The spatial correlations along both the horizontal and vertical directions are considered with the application of the same filter for every pixel in the same row. In addition, every filter convolves with the pixels present in the same column of each layer. The features are transmitted to successive sub-networks, and then, the final set of features from the SSrA is provided to the attention layer to retard the least relevant features. The output from the attention layer is the final reconstructed HSI image. The brief descriptions about the spatial



initialization (SI), spatial advancement (SA), spectra initialization (SrI) and spatial–spectral advancement (SSrA) are given below.

(a) Spatial initialization (SI) network: This network comprises convolutional layers with dense blocks for deep feature learning. The main aim of this network is to reverse the level 2 prediction process and to obtain the 3D matrix. This is attained by applying to the network the compressive patch  $v^i$ , and  $v_b^i$  indicates the pixel values in the  $b^{th}$  row of  $v^i$ . Thus, the output of the  $b^{th}$  row can be expressed as Equation (5).

$$o_1(v_b^i) = \max(w_{1,b} \times v^i + \vartheta_{1,b}, 0) \quad (5)$$

where  $w_{1,b}$  indicates the weight value and  $\vartheta_{1,b}$  indicates the biases for the  $b^{th}$  row. It is noteworthy that the number of feature maps generated is consistent with that of the spectral band. The ReLU activation is applied after every layer in the network model to carry out the nonlinear operation. For every row of the compressed image patch, the learning process is carried out, which results in the desired 3D matrix.

(b) Spatial advancement (SA) network: This network obtains the predicted 3D matrix as input and tries to predict the traversal between the pixels to attain the prediction value. Moreover, the spatial information present in the image is used as a reference by the network model, and this information is extracted and learned by the dense blocks within this network. The reverse operation of traversal is carried out in this network model to attain the desired reconstruction results. Each denseblock takes the output of the previous denseblock as input and adds the learned mapping to the input of the current block as output. The desired mapping is performed by a few stacked underlying layers in CNN with the help of a mapping function. If the output is similar to the input, then the mapping is identical. The stacked layers are expected to approximate the difference between the input and output. As a result, the convolutional layers initialized with zero means can be trained to estimate the difference value.

(c) Spectra initialization (SrI) network: In this network, the feature maps attained from the SA network are learned, and the level 1 prediction process is reversed. The value of  $v'(p, q, r)$  is obtained as the output of this network model based on the training with the feature maps with different spectral bands. The output of the  $s^{th}$  band obtained from the first layer of this network can be mathematically expressed as shown in Equation (6).

$$\begin{aligned} f_1(c_1^i) &= \max(g_{1,1} * \text{concat}(c_1^i, c_2^i) + \beta_{1,1}, 0) \\ f_1(c_s^i) &= \max(g_{1,s} * \text{concat}(c_{s-1}^i, c_s^i, c_{s+1}^i) + \beta_{1,s}, 0) \\ f_1(c_S^i) &= \max(g_{1,S} * \text{concat}(c_{S-1}^i, c_S^i) + \beta_{1,S}, 0) \end{aligned} \quad (6)$$

where  $c_s^i$  indicates the output of the SA network,  $g_{1,s}$  indicates the filters of the layers and  $\text{concat}()$  is the concatenation operation. The convolution, pooling and reversal operations are performed on the given input to predict the  $v'(p, q, r)$  value.

(d) Spatial–spectral advancement (SSrA) network: This layer finally predicts the mean value and results in the reconstructed HSI image. This is completed with the help of convolution operation along with global average pooling at the end. An additional attention layer is attached at the end of this network model to obtain high-quality outputs. The overall output obtained from this layer can be expressed as shown in Equation (7).

$$F* = \text{ReLU}(w * f_1) \quad (7)$$

where  $\text{ReLU}$  indicates the ReLU activation,  $w$  indicates the weight value for training and  $f_1$  indicates the output of the SrI network. The hyperparameter settings of the proposed approach are presented briefly in the following Table 2.

**Table 2.** Hyperparameter settings for the proposed approach.

Hyperparameters	Values
<b>SqueezeNet</b>	
No. of convolution layers	2
No. of fire blocks	4
No. of hidden units	10
No. of hidden neurons	450,000
Initial learning rate	0.001
Dropout rate	0.1–0.25
Mini batch size	16
Max epochs	5000
<b>DenseNet</b>	
No. of dense blocks	16
No. of convolution layers	3
No. of hidden units	16
No. of hidden neurons	450,000
Initial learning rate	0.001
Dropout rate	0.1–0.25
Mini batch size	16
Max epochs	5000

### 2.3. Performance Analysis

The performance analysis of the proposed approach in terms of different performance metrics are detailed under this section. The analysis of the proposed approach has been given for the denoising, compression and reconstruction stages. The comparison of the proposed approach is carried out with the existing mechanisms such as HSI deep CNN (HSID-CNN), CNN-based HSI denoising (HSID-DeNet), U-Net with global spatial module and global channel module (U-Net+GSM+GCM), two-step iterative shrinkage/thresholding (TwIST), gradient projection for sparse reconstruction (GPSR), generalized alternating projection based total variation (GAP-TV) and backtracking reconstruction network (BTR-Net). The result values for comparison are taken from U-Net+GSM+GCM [32] and BTR-Net [28].

### 2.4. Evaluation of TSF Model

The images after compression using TSF are then evaluated using metrics such as the Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Normalized Cross-Correlation (NCC), Structural Content (SC), Maximum Difference (MD), Normalized Absolute Error (NAE) and Compression Ratio (CR). Mean Square Error measures the error with respect to the center of the image values. It is calculated as the average of the cumulative squared value of the error difference between the original and the compressed image as shown in Equation (8).

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (I(i,j) - C(i,j))^2. \quad (8)$$

where  $I(i,j)$  and  $C(i,j)$  are two images of size  $M \times N$ ,  $I$  represents the original image and  $C$  represents the compressed image. A lower MSE value shows less error in the compressed image. A higher PSNR value implies a better image after compression. Normalized Cross-Correlation (NCC) gives the structural content (SC) of the image as shown in Equation (9).

$$NCC = \frac{\sum_{i=1}^M \sum_{j=1}^N I(i,j)C(i,j)}{\sum_{i=1}^M \sum_{j=1}^N I(i,j)^2}. \quad (9)$$

SC is used to measure the image similarity. Consider the image represented as the  $M \times N$  matrix; then, the structural content variation factor can be evaluated using Equation (10).

$$SC = \frac{\sum_{i=1}^M \sum_{j=1}^N I(i, j)}{\sum_{i=1}^M \sum_{j=1}^N C(i, j)}. \quad (10)$$

where SC is the structural content factor which is the ratio between the sum of the pixel values of the original image  $I(i, j)$  before compression and the sum of the pixel values of the compressed image  $C(i, j)$ . Maximum Difference (MD) is a measure to find the difference in content between the original and compressed image. The higher the difference, the lesser the image quality will be, as shown in Equation (11).

$$MD = (|I(i, j) - C(i, j)|). \quad (11)$$

Normalized Absolute Error (NAE) is also used to find the difference between the original and reconstructed image. The lower the error, the higher the quality will be, as shown in Equation (12).

$$NAE = \frac{\sum_{i=1}^M \sum_{j=1}^N |I(i, j) - C(i, j)|}{\sum_{i=1}^M \sum_{j=1}^N |C(i, j)|}. \quad (12)$$

Compression Ratio (CR) [36] is defined as the ratio of the memory size of the original image to the memory size of the compressed image, as given in Equation (13).

$$CR = \frac{\text{Size of Original Image}}{\text{Size of Compressed Image}}. \quad (13)$$

### 2.5. Evaluation of SDANet Model

The performance of the proposed approach is evaluated in terms of Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Relative Absolute Error (RAE), Spectral Angle Mapper (SAM), PSNR Human Visual System (PSNR-HVS) and Multi-Scale SSIM (MSSIM), which helps in the comprehensive evaluation of denoising, compression and reconstruction results. The mathematical formulas of the performance measures are given as shown in Equations (14)–(17).

$$PSNR = \frac{1}{\gamma} \sum_{i=1}^{\gamma} 10 \times \log_{10} \left( \frac{\max_i^2}{MSE_i} \right). \quad (14)$$

$$SSIM = \frac{1}{\gamma} \sum_{i=1}^{\gamma} \frac{(2\mu_{p_i}\mu_{q_i} + a_1)(2\sigma_{p_i q_i} + a_2)}{(\mu_{p_i}^2 + \mu_{q_i}^2 + a_1)(\sigma_{p_i}^2 + \sigma_{q_i}^2 + a_2)} \quad (15)$$

$$SAM = \arccos \left( \frac{\langle \Omega, \Omega' \rangle}{\|\Omega\|_2 \|\Omega'\|_2} \right) \quad (16)$$

$$RAE = \frac{1}{\eta_s \eta_p \eta_q} \sum_{i=1}^{\eta_s} \sum_{j=1}^{\eta_p} \sum_{k=1}^{\eta_q} \frac{|F * (i, j, k) - f_1(i, j, k)|}{f_1(i, j, k)} \quad (17)$$

where  $\gamma$  indicates the number of spectral bands,  $\max_i^2$  indicates the maximum pixel value of the  $i$ th band,  $MSE_i$  indicates the Mean Square Error (MSE) between the processed and original image of the  $i$ th band,  $\mu_{p_i}$  and  $\mu_{q_i}$  are the mean values of the images  $p$  and  $q$ ,  $\sigma_{p_i}$  and  $\sigma_{q_i}$  are the variances of  $p$  and  $q$ ,  $a_1$  and  $a_2$  are the constants set to 0.0001 and 0.0009,  $\langle \Omega, \Omega' \rangle$  indicates the dot product between the original and denoised spectrums  $\Omega$  and  $\Omega'$ ,  $\|\bullet\|_2$  indicates the binary norm,  $\eta_s$  indicates the count of spectral bands,  $\eta_p$  and  $\eta_q$  indicate

the spatial resolution of HSI, and  $F^*(i, j, k)$  and  $f_1(i, j, k)$  are the points at the  $i$ th spectral band with coordinates  $(j, k)$ .

PSNR-HVS [37,38] is the modified form of PSNR after taking contrast sensitivity function into account. The contrast sensitivity function has the potential of adding more information about the functioning of the visual system by assessing sensitivity over a wide range of spatial frequencies. It is calculated in the DCT domain in  $8 \times 8$  blocks.

The multi-scale SSIM (MSSIM) [38,39] metric is calculated by taking the reference and distorted image signals as the input. A low-pass filter is applied and downsamples the filtered image by a factor of 2. The original image is indexed as scale 1, and the highest scale is given as  $M$ , which is obtained after  $M - 1$  iterations. At each scale, the contrast comparison and the structure comparison are calculated. The luminance comparison is computed only at the highest scale. The MSSIM is obtained by combining the measurements at different scales.

### 3. Results

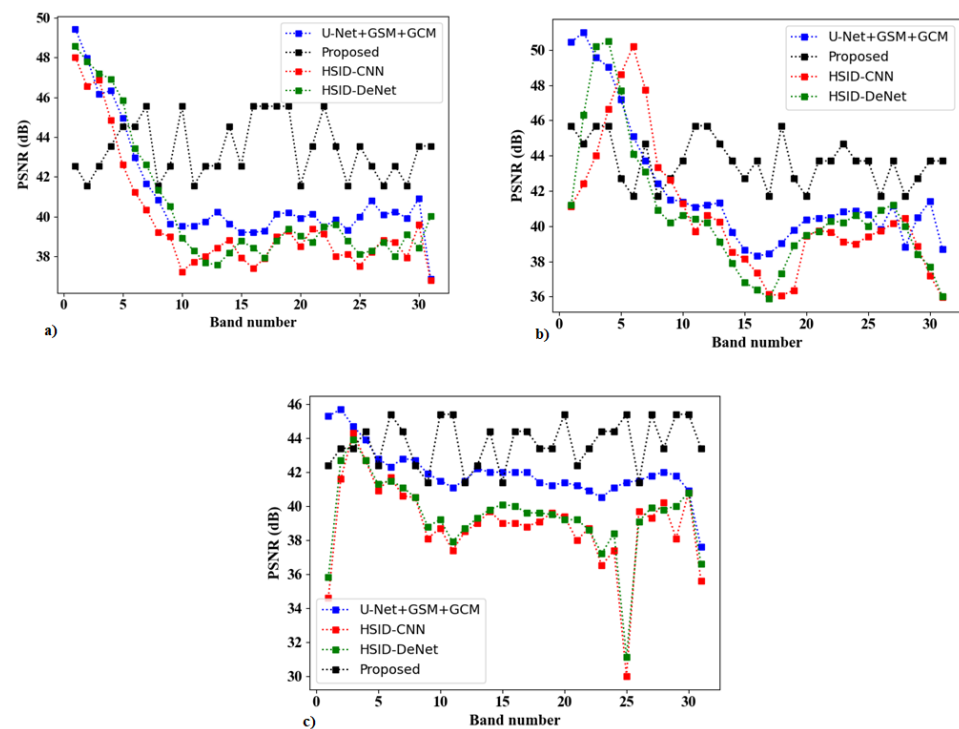
This section presents the results of the different stages of the proposed approach. The performance evaluation has been completed using the metrics explained in Sections 2.4 and 2.5, and the results are shown below. The results of the proposed approach have also been compared with the discussed existing methodologies and demonstrated under this section as shown below.

#### 3.1. Denoising of HSI

The denoising of HSI is carried out as described in the Methodology section (Section 2.2.1). The performance of the SDANet has been evaluated in terms of Peak Signal to Noise Ratio (PSNR) at noise levels (NLs) = 30, 50 and 70 to the spectral bands 1 to 30. The obtained results are compared with the results of existing denoising approaches. Table 3 shows the values of all approaches extracted to the spectral bands 5, 10, 15, 20, 25 and 30, and the performances can be interpreted from Figure 5. The interpretation results obtained to the noise level = 30 show a relatively high performance of SDANet when compared with the other models (Table 3). The PSNR value attained by the proposed model for band number 30 is 43.43 dB, whereas the PSNR values attained by the existing models such as HSID-DeNet, HSID-CNN and U-Net+GSM+GCM are 40.02 dB, 36.78 dB and 36.84 dB, respectively. The PSNR values of SDANet are higher in the bands and more optimal, and the results may be due to the effectiveness of the deep learning model that is utilized in this study. The method of denoising showed an excellent extraction capability in the removal of noise from the spectral bands. The denoising can also be depicted from the graphical representation (Figure 5a). In addition, the values obtained for noise level = 50 show that the performance of the SDANet is more optimal than those of the other models. The overall PSNR value for band number 30 at NL = 50 is 43.23 dB, whereas the compared models such as HSID-DeNet, HSID-CNN and U-Net+GSM+GCM are 36.0 dB, 36.0 dB and 38.69 dB, respectively (Table 3). The graphical representation of the PSNR for NL = 50 PSNR of the SDANet is low at the beginning and increases gradually with the increase of band numbers (Figure 5b). In addition, the interpretation of PSNR values of noise level = 70 for SDANet with other models shows an optimal PSNR. The overall PSNR value attained by the SDANet for the band number 30 is 43.34 dB, whereas the other models show 36.6 dB, 35.6 dB and 37.6 dB for HSID-DeNet, HSID-CNN, and U-Net+GSM+GCM, respectively (Table 3). The plot drawn for the PSNR of noise level = 70 further confirms the existing optimal performance (Figure 5c). Overall, the study of denoising in the spectral bands 1 to 30 of HSI using different noise levels (levels = 30, 50 and 70) provided high values of PSNR and showed a high performance of SDANet when compared with the other models.

**Table 3.** Results of comparative analysis of PSNR of different noise levels for the spectral bands 5, 10, 15, 20, 25 and 30.

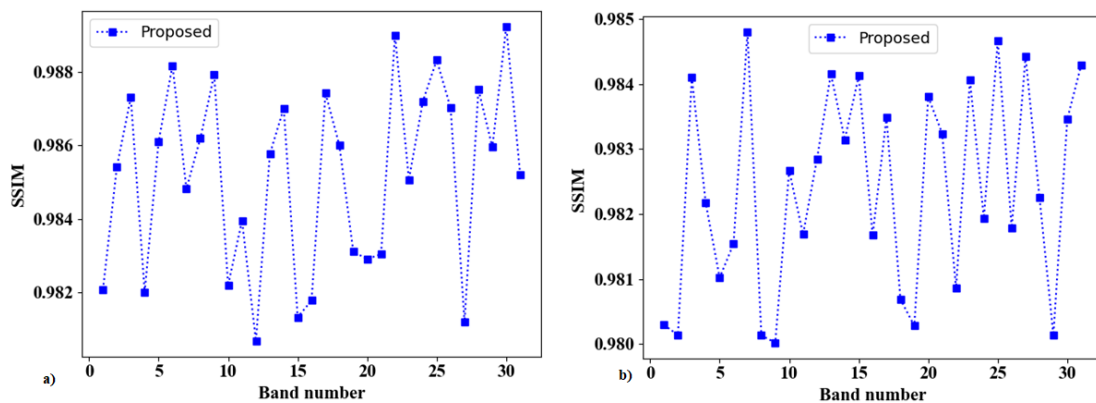
Noise Levels	Spectral Bands	PSNR (dB)			
		SDANet	HSID-DeNet	HSID-CNN	U-Net+GSM+GCM
30	5	44.42	43.42	41.22	42.98
	10	41.34	38.26	37.71	39.52
	15	45.41	38.43	37.39	39.19
	20	43.4	38.7	39.36	40.13
	25	42.41	38.26	38.21	40.78
	30	43.43	40.02	36.78	36.84
50	5	41.52	44.1	47.74	45.08
	10	44.92	40.4	40.61	41.07
	15	43.22	36.4	36.11	38.29
	20	43.16	39.7	39.66	40.44
	25	41.59	40.9	40.15	39.81
	30	43.23	36	36	38.69
70	5	45.43	41.5	41.7	42.3
	10	45.39	37.9	37.4	41.1
	15	44.45	40	39	42
	20	42.42	39.2	38	41.2
	25	41.45	39.1	39.7	41.5
	30	43.34	36.6	35.6	37.6

**Figure 5.** Plots of PSNR for different noise levels viz. (a) 30, (b) 50 and (c) 70 for all models (SDANet (proposed), U-Net+GSM+GCM, HSID-CNN and HSID-DeNet) in the spectral bands 1 to 30.

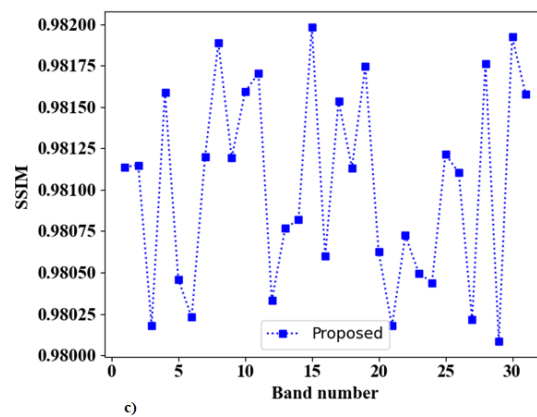
The performance of the SDANet has been evaluated in terms of SSIM and Peak Signal to Noise Ratio-HVS (PSNR) at noise levels (NLs) = 30, 50 and 70 to the spectral bands 1 to 30. Table 4 shows the values extracted to the spectral bands 5, 10, 15, 20, 25 and 30, and the performances can be interpreted from Figures 6 and 7. The interpretation results obtained to the noise level = 30 show a relatively high performance of SDANet. The SSIM value attained by the proposed model for band number 30 is 0.9852 at 30 dB, 0.9842 at 50 dB and 0.9815 at 70 dB, respectively. The PSNR-HVS values of SDANet are also optimal. The overall PSNR-HVS values for band number 30 at NL = 30 is 0.9941 dB, at NL = 50 is 0.9855 and at NL = 70 is 0.9833, respectively. Figure 6a–c show the variation of SSIM with band numbers for different noise levels of 30, 50 and 70, respectively. The PSNR-HVS variation with respect to the band numbers is plotted in Figure 7a, Figure 7b and Figure 7c, respectively.

**Table 4.** Results of analysis of SSIM and PSNR-HVS of different noise levels for the spectral bands 5, 10, 15, 20, 25 and 30.

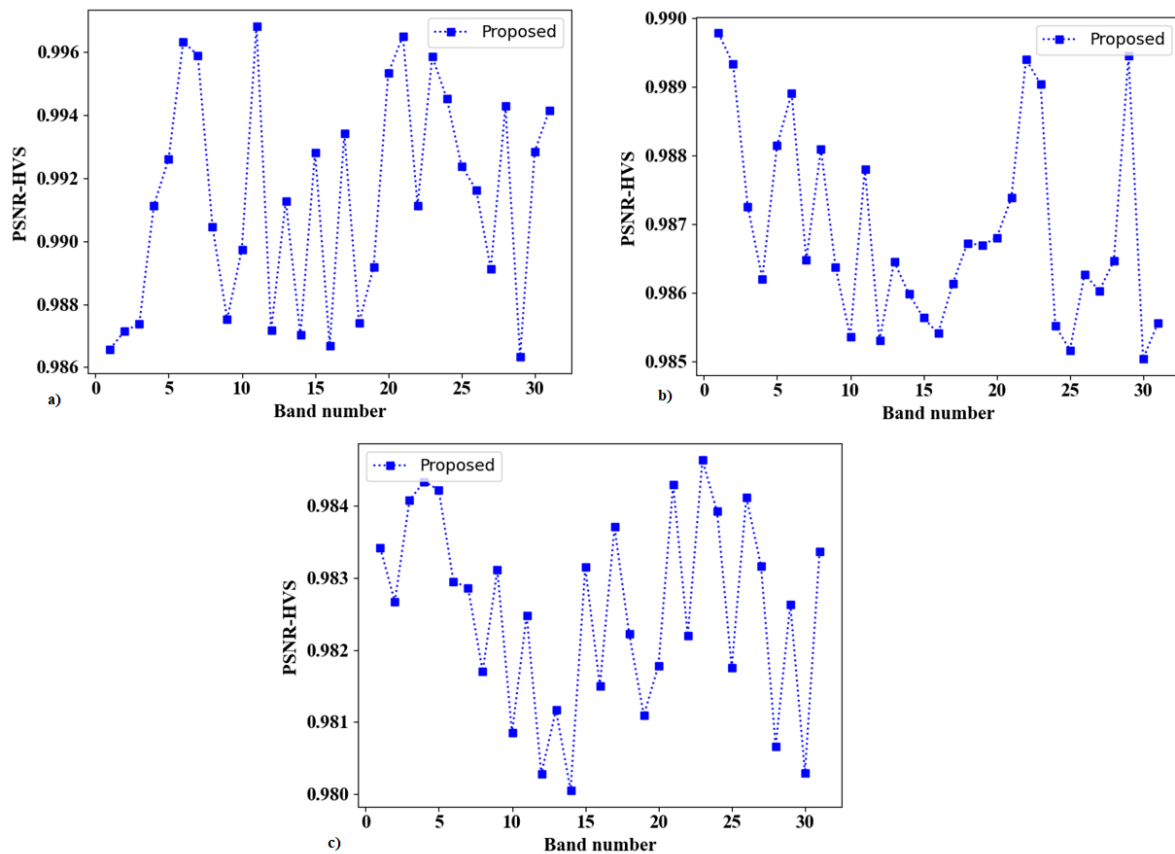
Noise Levels	Spectral Bands	SSIM (dB)	PSNR-HVS (dB)
30	5	0.9881	0.9963
	10	0.9839	0.9968
	15	0.9817	0.9866
	20	0.9830	0.9965
	25	0.9870	0.9916
	30	0.9852	0.9941
50	5	0.9815	0.9889
	10	0.9816	0.9878
	15	0.9834	0.9854
	20	0.9832	0.9873
	25	0.9817	0.9862
	30	0.9842	0.9855
70	5	0.9802	0.9829
	10	0.9817	0.9824
	15	0.9805	0.9814
	20	0.9801	0.9842
	25	0.9811	0.9841
	30	0.9815	0.9833



**Figure 6.** Cont.



**Figure 6.** Plots of SSIM for different noise levels viz. (a) 30, (b) 50 and (c) 70 for the proposed model in the spectral bands 1 to 30.



**Figure 7.** Plots of PSNR-HVS for different noise levels viz. (a) 30, (b) 50 and (c) 70 for the proposed model in the spectral bands 1 to 30.

### 3.2. Compression of HSI

The compression of each image is carried out after the denoising stage using TSF, and the metrics explained in Section 2.4 have been evaluated for analyzing the performance of compression. Five different images (as shown in Figure 8) are randomly selected from the dataset and analysed using the metrics such as Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR), Normalized Cross-Correlation (NCC), Structural Content (SC), Maximum Difference (MD), Normalized Absolute Error (NAE). Table 4 shows the quantitative analysis values.



**Figure 8.** Randomly selected images from the dataset for evaluating the performance of compression.

Table 5 shows that the compression model used here gives better quality images and also preserved the minute structural details of the image.

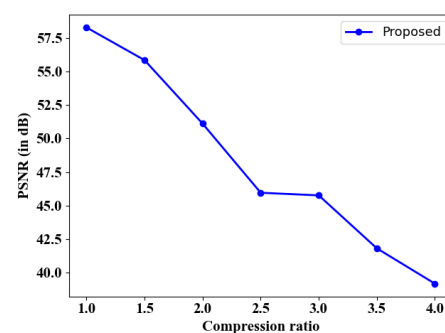
**Table 5.** Results of performance evaluation of compression.

Metrics	Image 1	Image 2	Image 3	Image 4	Image 5
PSNR	40.811	40.254	40.216	40.240	39.507
MSE	8.334	8.722	8.845	8.998	8.602
NCC	0.990	0.994	0.991	0.991	0.991
SC	1	1	1	1	1
MD	27.657	26.906	26.452	26.281	26.586
NAE	0.010	0.016	0.018	0.012	0.019

Table 6 shows the details of PSNR calculated at different values of Compression Ratio (CR). The CR values of 1, 1.5, 2, 2.5, 3, 3.5 and 4 are considered here, and the plot between these is shown in Figure 9. Although the PSNR value decreases with the increase in CR, the proposed model manages to maintain a better PSNR value even at a Compression Ratio of 4.

**Table 6.** Results of performance evaluation of PSNR with Compression Ratio.

Compression Ratio	PSNR (dB)
1	58.31
1.5	55.86
2	51.11
2.5	45.94
3	45.75
3.5	41.79
4	39.14



**Figure 9.** PSNR versus Compression Ratio.

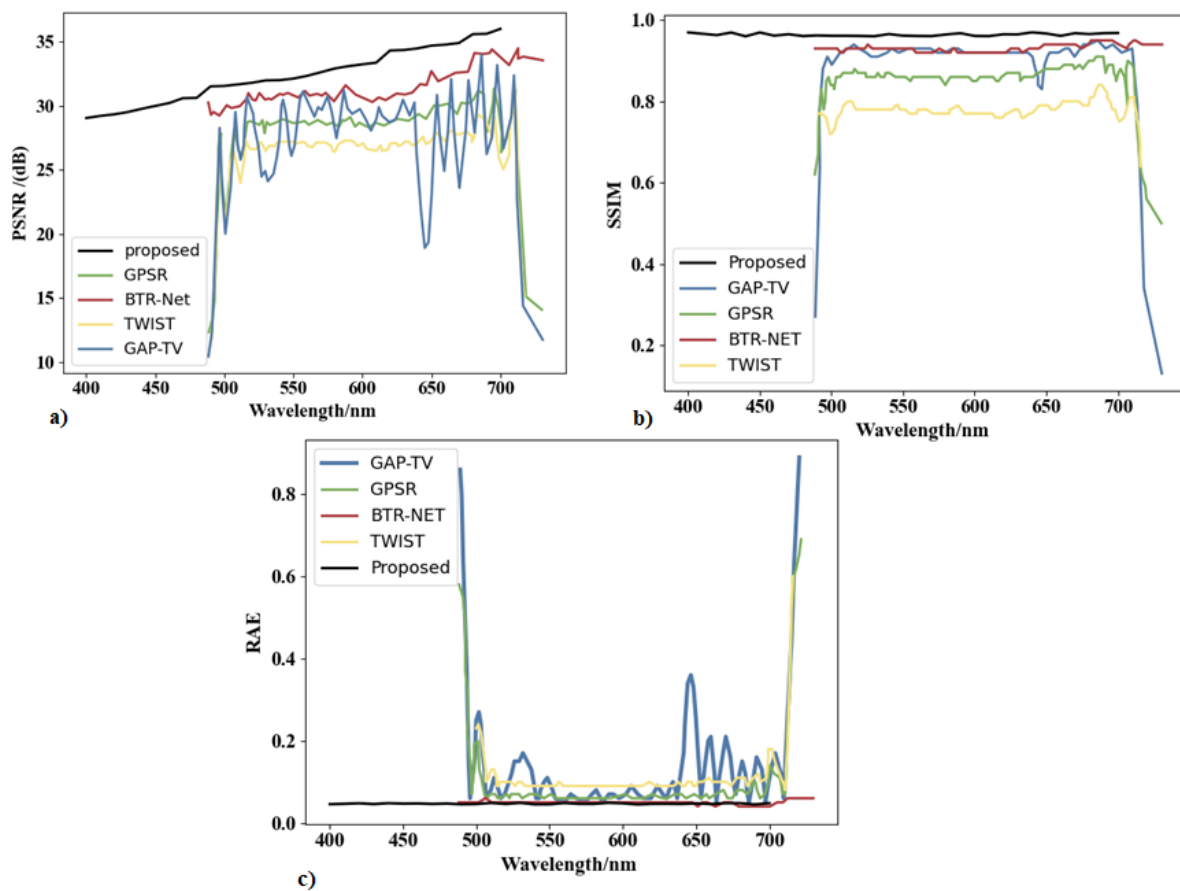


### 3.3. Reconstruction of HSI

The reconstruction of HSI is carried out, and the results of SDANet are provided with the results of Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) and Relative Absolute Error (RAE). Table 7 shows the overall Peak Signal to Noise Ratio (PSNR) attained by the SDANet model, which is about 35.98 dB for 700 nm at a Compression Ratio of 6. The other existing models such as TwIST, GPSR, GAP-TV and BTR-Net provided the PSNR values of 25.89 dB, 27.57 dB, 26.81 dB and 31.55 dB, respectively. The graph in Figure 10a compares the values of PSNR with the results of existing models. It is obvious that the SDANet model performed well in the reconstruction when compared with the other models. In this study, a significant improvement is observed in the performance of the proposed SDANet model in terms of PSNR, which is due to the proposed DAN model as well as the utilization of an additional attention layer to concentrate more on the selected features.

**Table 7.** Results of comparative analysis of PSNR, SSIM and RAE by the TwIST, GPSR, GAP-TV, BTR-Net and SDANet (proposed in this study) models.

Models	PSNR (dB)	SSIM (dB)	RAE
TwIST	25.89	0.752	0.153
GPSR	27.57	0.842	0.116
GAP-TV	26.81	0.884	0.160
BTR-Net	31.55	0.929	0.058
SDANet	35.98	0.9964	0.043



**Figure 10.** Graphs comparing the results of PSNR, SSIM and RAE with the TwIST, GPSR, GAP-TV, BTR-Net and SDANet (proposed in this study) models. (a) PSNR versus Wavelength/nm. (b) SSIM versus Wavelength/nm. (c) RAE versus Wavelength/nm.

The interpretation of the overall Structural Similarity Index Measure (SSIM) values shows that the SDANet model achieved a high value of 0.9964 dB when compared to the other existing models such as TwIST, GPSR, GAP-TV and BTR-Net, which provided the values of 0.752 dB, 0.842 dB, 0.884 dB and 0.929 dB, respectively (Table 7). The comparison of SSIM values of the SDANet model with the other models is graphically shown in Figure 10b. From the figure, it is clear that the SDANet model assured an optimal SSIM value compared to the other models. However, the BTR-Net model showed optimal SSIM values compared to the other models, and the least value is attained by the TwIST model. Furthermore, the interpretation of the overall RAE values of the SDANet model showed a minimum value of 0.043 when compared with the RAE values of the other models such as TwIST, GPSR, GAP-TV and BTR-Net, which provided 0.153, 0.116, 0.160, and 0.058, respectively (Table 7). The comparison of the percentage of the SDANet model with the percentages of other models is provided in Figure 10c. From the figure, it is clear that the performance of the SDANet model is more optimal than those of the compared models. The RAE attained by the SDANet model is very low compared to the other models, proving that the system is highly accurate in reconstruction. Among the compared models, BTR-Net resulted in a maximum optimal value and the least performance is obtained by the GAP-TV model. Overall, the interpretation of values from the table shows that the SDANet model (proposed in this study) is more optimal in reconstructing the HSI from the compressed image when compared with the results of other models.

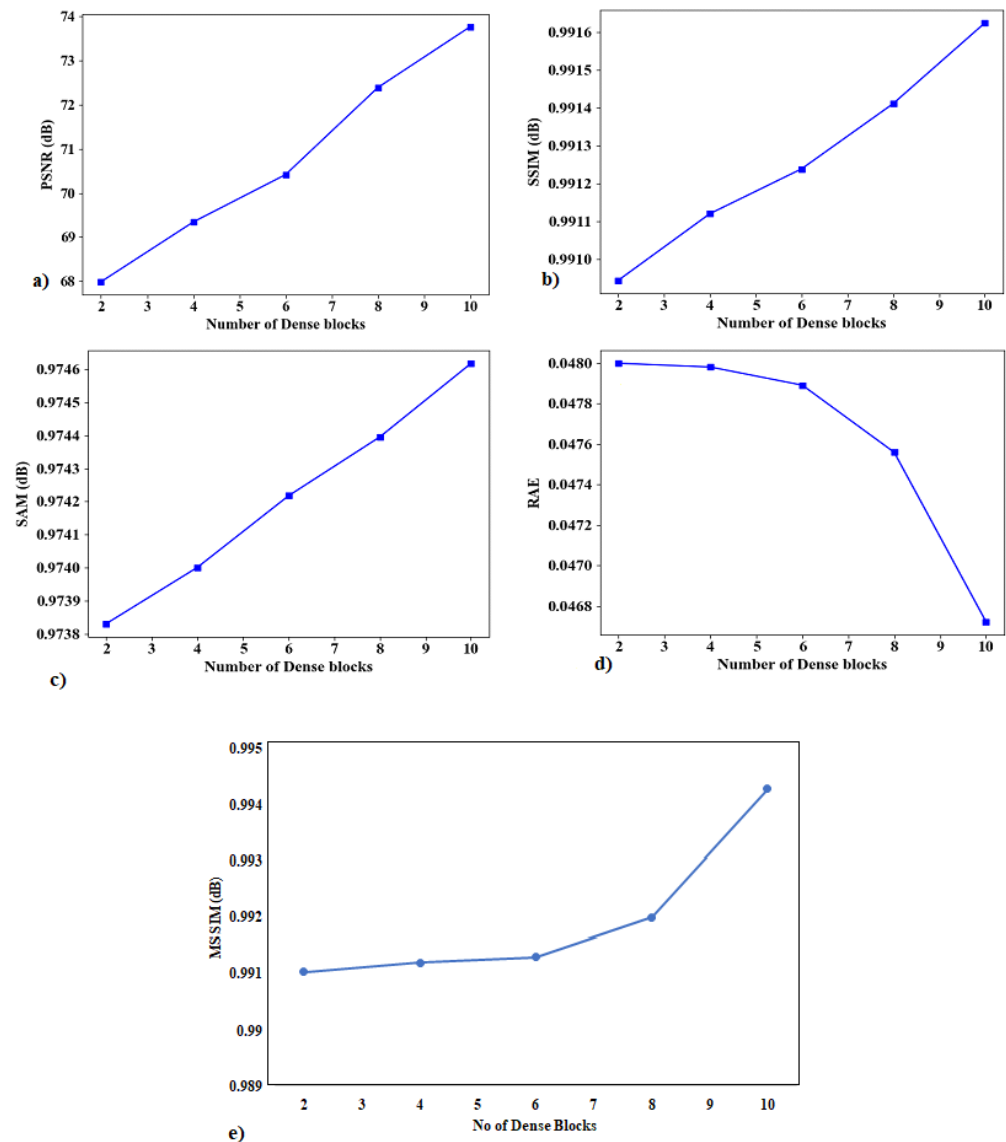
### 3.4. Performance of SDANet Model

#### 3.4.1. Dense Block Analysis

Moreover, the PSNR, SSIM, MSSIM, RAE and SAM values of the SDANet model are further studied with the values of varying numbers of dense blocks that are carried out to understand more about the capability of the model. The results analysis is given in Table 8. The interpretation of the values against the increase of the number of dense blocks shows the occurrence of high values and a significant gradual increase of values among the PSNR, SSIM, MSSIM, RAE and SAM. From the values, it is clear that the performance of the SDANet model is high and increases with the increase of the dense blocks, since these blocks are responsible for extracting the deep features present in the image. The variation plots drawn for PSNR, SSIM, MSSIM, RAE and SAM values against the number of dense blocks to understand the capability of the SDANet model are given in Figure 11. From the figures, it is evident that the presence of dense blocks in the proposed model makes more sense. Figure 11a–c,e confirmed further the increased performance of the model by increasing the PSNR, SSIM, SAM and MSSIM values when the number of dense blocks is increased. The PSNR value of the proposed approach shows an increasing trend when the dense blocks are increased. The same case is encountered in the SSIM plot, and a decreasing trend is seen in the performance of the proposed approach in terms of error in Figure 11d. The study of performance variations of the SDANet model in terms of different metrics by varying the dense blocks between 2 and 10 indicates the effectiveness of using dense blocks for learning.

**Table 8.** Results of the performance analyses of the SDANet model with respect to number of dense blocks.

No. of Dense Blocks	PSNR (dB)	SSIM (dB)	MSSIM (dB)	SAM (dB)	RAE
2	68.31	0.9909	0.99104	0.9738	0.0479
4	69.56	0.9911	0.9912	0.9740	0.0479
6	70.58	0.9912	0.9913	0.9742	0.0478
8	72.42	0.9914	0.992	0.9744	0.0475
10	73.73	0.9916	0.9943	0.9746	0.0467



**Figure 11.** Plots of different performance metrics (PSNR, SSIM, MSSIM, RAE and SAM) by varying the dense blocks. (a) PSNR versus No of Dense Blocks. (b) SSIM versus No of Dense Blocks. (c) SAM versus No of Dense Blocks. (d) RAE versus No of Dense Blocks. (e) MSSIM versus No of Dense Blocks.

### 3.4.2. Kernel Size Analysis

Furthermore, the performance results (shown in Table 9) are analyzed by using varying kernel sizes of the convolutional layers within dense blocks. In this study, there are three convolutional layers within the dense blocks, and each of the layers includes kernels considered for striding over the input image. Here, each convolutional layer consists of different numbers of kernels with varying kernel sizes, and these are taken to identify the percentage of influence and to attain the overall results. The kernel sizes of the three convolutional layers are varied accordingly: (1) 11-1-7 where the first layer includes 11 kernels, the second layer includes 1 kernel and the final layer includes 7 kernels; (2) 9-1-5 where there are 9 kernels in the first layer, 1 kernel in the second layer and 5 kernels in the final layer; and (3) 7-1-3 where there are 7 kernels in the first layer, 1 kernel in the second layer and 3 kernels in the final layer. The results obtained by the study are given in Table 7. The interpretation of results of the SDANet model against the different kernel size 11-1-7 shows a maximum PSNR value about 65.77 when compared to the results of other kernel sizes.

The SSIM values and MSSIM values are identified to be constant on different kernel sizes, whereas the results of SAM and RAE showed minor differences. Overall, the study of the SDANet model against the different kernel sizes showed no significant major variations by analyzing the different kernel sizes.

**Table 9.** Results of performance analysis of SDANet model with different kernel sizes.

Kernel Sizes	PSNR	SSIM	MSSIM	SAM	RAE
11-1-7	65.77	0.996	0.9967	0.019	0.045
9-1-5	64.10	0.996	0.9972	0.019	0.045
7-1-3	65.07	0.996	0.9974	0.020	0.048

### 3.5. Visual Interpretation and Validation

A visualized analysis of the results of the different stages of the SDANet model is carried out to understand more about the outcome of the model. Figure 12 presents the images of different noise levels (levels of 30, 50 and 70) and the respective denoised images. The visual interpretations of images show that the images are stable and have no variation, even when the noise level is varied, the denoised output for all the cases looks the same. This proves the stability of the SDANet model for different noisy inputs. The resulted images prove the efficacy of the SDANet model in denoising the HSI. In addition, the interpretation of input, denoised, compressed and reconstructed images further confirms that the SDANet model generated quality output images (Figure 13).



**Figure 12.** Results of noisy and denoised images of different noise levels.

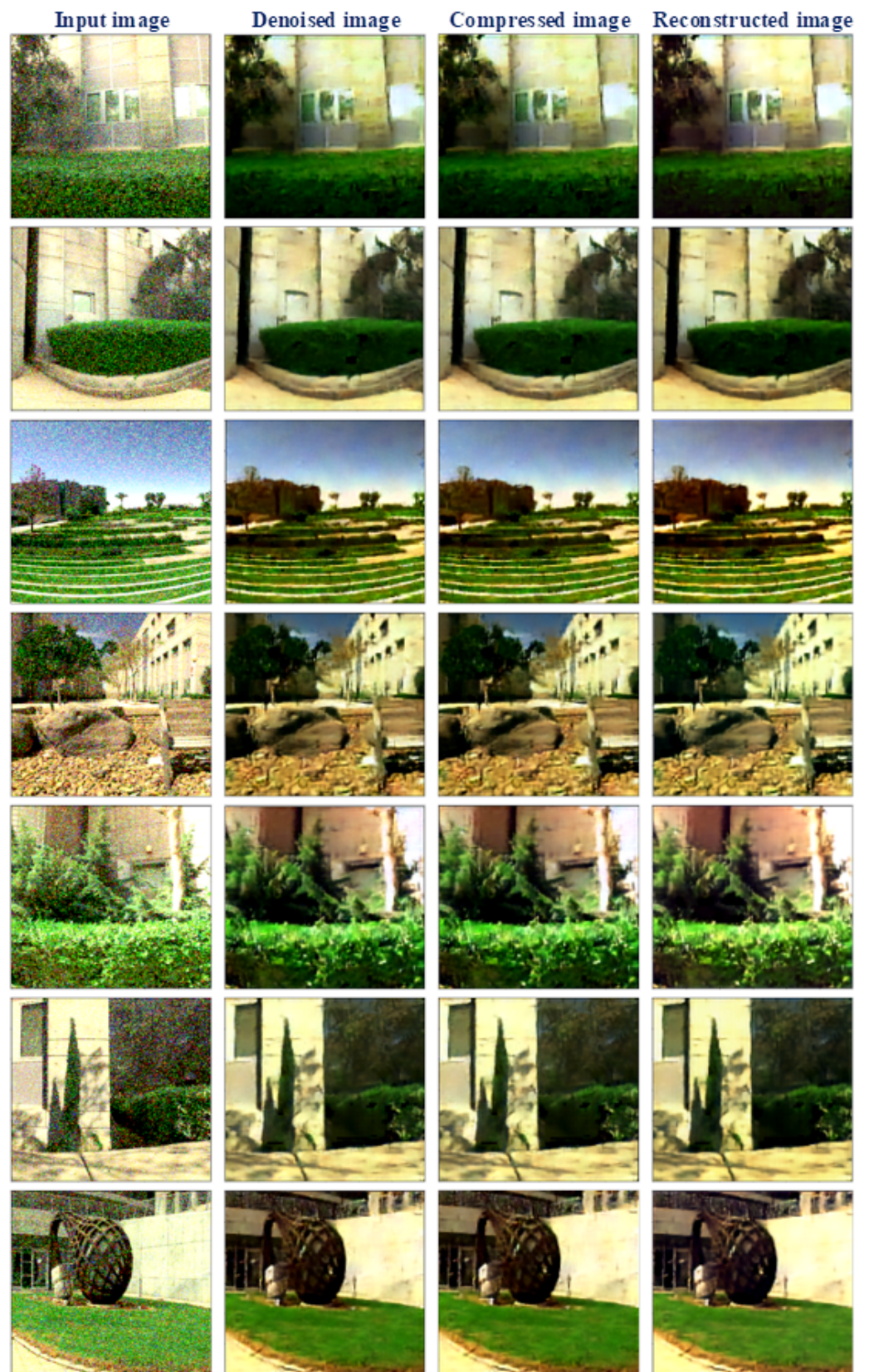


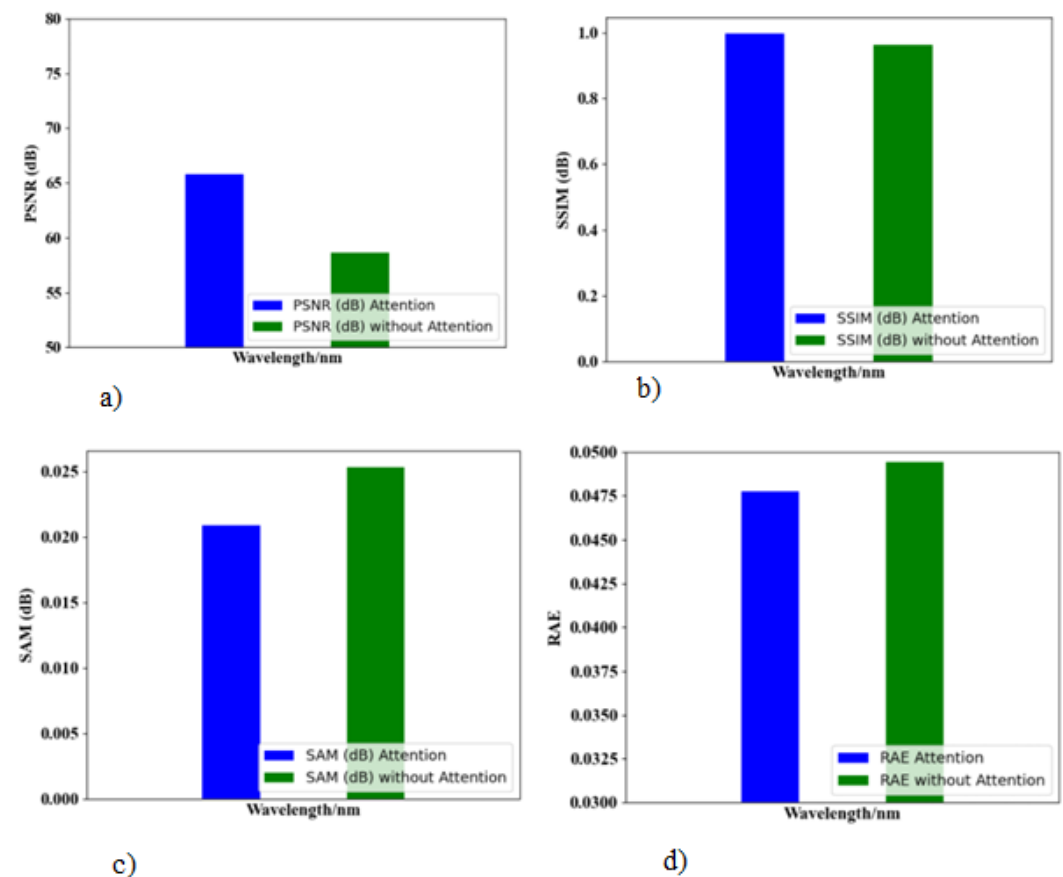
Figure 13. The input, denoised, compressed and reconstructed images of SDANet model.

### 3.6. Ablation Experiments

Ablation experiments are carried out for the SDANet model to provide a deeper analysis in terms of performance. The results of the performance comparison of the SDANet model with and without an attention layer are presented in Table 10. The interpretation of results shows that the SDANet model requires an attention at the end of the model to generate better results due to better feature extraction. However, the values are optimal when the model uses an attention layer at the end compared to the model without an attention layer. The bar chart drawn for the performance of the SDANet model with and without an attention layer is given in Figure 14. From the plots, it is clear that the attention layer at the end is crucial to see performance improvement in the proposed model in terms of reconstructing the HSI. The PSNR plot shows an increase in its overall value with the use of an attention layer compared to that without an attention layer (Figure 14a). The same is observed in the performance comparison in terms of SSIM and SAM (Figure 14b,c). Figure 14d shows that the RAE value is reduced, proving that the system is resistant to loss while using an attention layer at the end.

**Table 10.** Performance comparison with and without an attention layer in SDANet.

Metrics	With Attention	Without Attention
PSNR (dB)	63.49	58.47
SSIM (dB)	0.99	0.97
SAM	0.024	0.027
RAE	0.0421	0.045



**Figure 14.** Performance of the proposed approach with and without an attention layer. (a) PSNR versus Wavelength/nm. (b) SSIM versus Wavelength/nm. (c) SAM versus Wavelength/nm. (d) RAE versus Wavelength/nm.

### 3.7. K-Fold Cross Validation

The k-fold cross-validation is carried out to prove the effectiveness of the model over different amounts of training sets. The main goal of k-fold cross-validation is to reduce the dependence of the proposed model over a single partition of the dataset. In our experiments, a total of five folds are considered to evaluate the proposed model. The downloaded HSI are divided into five folds in a random manner, and for each run, one of the folds is kept for testing, and the remaining are considered for training the model. The validation experiments are repeated five times to produce the results. The experimental results of k-fold cross-validation are presented in Table 11. The interpretation of results shows that fold 1 achieved 64.86 dB of PSNR, 0.9966 dB of SSIM, 0.997 dB of MSSIM, 0.020 of SAM and 0.047 of RAE. Fold 2 resulted in 60.99 dB of PSNR, 0.9965 dB of SSIM, 0.9969 dB of MSSIM, 0.020 of SAM and 0.047 of RAE. For fold 3, the model produced 61.53 dB of PSNR, 0.9964 dB of SSIM, 0.9971 dB of MSSIM, 0.019 of SAM and 0.045 of RAE. Fold 4 produced 63.47 dB of PSNR, 0.9964 dB of SSIM, 0.9970 dB of MSSIM, 0.020 of SAM and 0.047 of RAE. Finally, fold 5 produced 63.87 dB of PSNR, 0.9963 dB of SSIM, 0.9969 dB of MSSIM, 0.019 of SAM and 0.048 of RAE. From the values obtained, it is clear that the proposed approach performed well in every fold of the dataset. It is obvious from the experiments that the proposed model is more effective on different HSIs and produced better results on different testing folds. Moreover, the conducted validation ensured that each sample of the dataset is exactly used once for the validation. The model proves to be independent of a single fold of the dataset and provides better results on different samples of the dataset. The overall results suggest that the proposed model is more accurate and produces reliable results on different folds of the dataset.

**Table 11.** K-fold cross-validation of the proposed approach.

Folds	PSNR (dB)	SSIM (dB)	MSSIM (dB)	SAM	RAE
1	64.86	0.9966	0.997	0.020	0.047
2	60.99	0.9965	0.9969	0.020	0.047
3	61.53	0.9964	0.9971	0.019	0.045
4	63.47	0.9964	0.9970	0.020	0.047
5	63.87	0.9963	0.9969	0.019	0.048
Average	62.94	0.9964	0.9969	0.019	0.046

## 4. Discussion

The simulations proved that the proposed model is more optimal than the other existing mechanisms in comparison. The proposed model is capable of generating more accurate reconstruction results than the other compared models. Moreover, the model enhanced the performance even in the denoising stage. The SqueezeNet model is introduced for denoising, and the fire blocks of the model learned the deep features and discriminated the noisy pixels from the original ones. This strategy helped to increase the performance of the proposed model. Various analyses under different noise levels proved that the model is more stable over different inputs with different noise levels. Moreover, the visualized analysis of the denoising stage proved that the model is stable under different noise levels. The visualized analysis provided for different stages of the proposed model also proved that the model is stable and effective in the compression and reconstruction of HSI.

## 5. Conclusions

In this work, a new and effective model for the denoising, compression and reconstruction of HSI has been introduced. The model is based on deep learning and is capable of providing more high-quality outputs than the other reconstruction models demonstrated in the literature. In this study, the SqueezeNet model is trained with the noisy images for denoising. The fire blocks in the model discriminated the noisy pixels from the original

pixels and resulted in effective denoising. The denoised image is then fed to the TSF model for compression. The dual-level prediction strategy is applied to attain higher performance in compression, and the result of this model is a compressed HSI. This output is then fed to the DAN model for reconstruction. The advantage of this model is that it uses four sub-networks where each network is responsible for reverse operation. The reconstruction is carried out by reversing the process of dual-level prediction. The final output of the proposed SDANet is a reconstructed HSI, and the analysis is carried out by comparing the obtained output with that of the original HSI. The proposed model is implemented in Python, and the evaluations are carried out using the BGU-ICVL dataset. The simulation analysis proved that the model is more accurate and reliable compared to the other models. In the future, we would like to extend the present work with additional features such as an inclusion of efficient optimization strategies to tune the hyperparameters so that the overall training time can be reduced. The accuracy in the performance of the proposed model shows that this technique can be utilized for mineral mapping and analysis and can be established as an effective model for mineral exploration with the reduced processing time.

**Author Contributions:** Conceptualization, J.A.; Methodology, J.A.; Software, D.M.; Validation, J.A. and R.S.; Formal analysis, D.M. and J.A.; Investigation, R.S.; Resources, D.M.; Data curation, R.S.; Writing—original draft, D.M.; Writing—review & editing, J.A.; Project administration, R.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Dian, R.; Li, S.; Sun, B.; Guo, A. Recent advances and new guidelines on hyperspectral and multispectral image fusion. *Inf. Fusion* **2021**, *69*, 40–51. [[CrossRef](#)]
2. Wang, Q.; Wu, Z.; Jin, J.; Wang, T.; Shen, Y. Low rank constraint and spatial spectral total variation for hyperspectral image mixed denoising. *Signal Process.* **2018**, *142*, 11–26. [[CrossRef](#)]
3. Nidhin Prabhakar, T.V.; Geetha, P. Two-dimensional empirical wavelet transform based supervised hyperspectral image classification. *ISPRS J. Photogramm. Remote Sens.* **2017**, *133*, 37–45. [[CrossRef](#)]
4. He, C.; Sun, L.; Huang, W.; Zhang, J.; Zheng, Y.; Jeon, B. TSLRLN: Tensor subspace low-rank learning with non-local prior for hyperspectral image mixed denoising. *Signal Process.* **2021**, *184*, 108060. [[CrossRef](#)]
5. Tao, L.; Mughees, A. *Deep Learning for Hyperspectral Image Analysis and Classification*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 5, pp. 1–207.
6. Sara, D.; Mandava, A.K.; Kumar, A.; Duela, S.; Jude, A. Hyperspectral and multispectral image fusion techniques for high resolution applications: A review. *Earth Sci. Inform.* **2021**, *14*, 1685–1705. [[CrossRef](#)]
7. Zhuang, L.; Ng, M.K.; Fu, X. Hyperspectral Image Mixed Noise Removal Using Subspace Representation and Deep CNN Image Prior. *Remote Sens.* **2021**, *13*, 4098. [[CrossRef](#)]
8. Bahraini, T.; Azimpour, P.; Yazdi, H.S. Modified-mean-shift-based noisy label detection for hyperspectral image classification. *Comput. Geosci.* **2021**, *155*, 104843. [[CrossRef](#)]
9. Fan, Y.R.; Huang, T.Z. Hyperspectral image restoration via superpixel segmentation of smooth band. *Neurocomputing* **2021**, *455*, 340–352. [[CrossRef](#)]
10. Zhu, Q.; Zhu, Y.; Sun, J.; Wu, Z. A Spark-Based Parallel Implementation of Compressed Hyperspectral Image Reconstruction and Anomaly Detection. In Proceedings of the 2022 Tenth International Conference on Advanced Cloud and Big Data (CBD), Guilin, China, 4–5 November 2022; pp. 54–59.
11. Anand, R.; Veni, S.; Aravinth, J. Big data challenges in airborne hyperspectral image for urban landuse classification. In Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017; pp. 1808–1814.
12. Bera, S.; Shrivastava, V.K. Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification. *Int. J. Remote Sens.* **2020**, *41*, 2664–2683. [[CrossRef](#)]
13. Ma, D.; Maki, H.; Neeno, S.; Zhang, L.; Wang, L.; Jin, J. Application of non-linear partial least squares analysis on prediction of biomass of maize plants using hyperspectral images. *Biosyst. Eng.* **2020**, *200*, 40–54. [[CrossRef](#)]
14. Su, H.; Wu, Z.; Zhu, A.X.; Du, Q. Low rank and collaborative representation for hyperspectral anomaly detection via robust dictionary construction. *ISPRS J. Photogramm. Remote Sens.* **2020**, *169*, 195–211. [[CrossRef](#)]



15. Wang, R.; Hu, H.; He, F.; Nie, F.; Cai, S.; Ming, Z. Self-weighted collaborative representation for hyperspectral anomaly detection. *Signal Process.* **2020**, *177*, 107718. [[CrossRef](#)]
16. Anand, R.; Veni, S.; Aravinth, J. Robust classification technique for hyperspectral images based on 3D-discrete wavelet transform. *Remote Sens.* **2021**, *13*, 1255. [[CrossRef](#)]
17. 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.* **2017**, *57*, 7317–7329. [[CrossRef](#)]
18. Huang, X.; Du, B.; Tao, D.; Zhang, L. Spatial-spectral weighted nuclear norm minimization for hyperspectral image denoising. *Neurocomputing* **2020**, *399*, 271–284. [[CrossRef](#)]
19. Zeng, H.; Xie, X.; Ning, J. Hyperspectral image denoising via global spatial-spectral total variation regularized nonconvex local low-rank tensor approximation. *Signal Process.* **2021**, *178*, 107805. [[CrossRef](#)]
20. Jacob, N.V.; Sowmya, V.; Soman, K.P. Effect of denoising on hyperspectral image classification using deep networks and kernel method. *J. Intell. Fuzzy Syst.* **2019**, *36*, 2067–2073. [[CrossRef](#)]
21. Kwan, C.; Larkin, J. New Results in Perceptually Lossless Compression of Hyperspectral Images. *J. Signal Inf. Process.* **2019**, *10*, 96–124. [[CrossRef](#)]
22. Fu, Y.; Lam, A.; Sato, I.; Sato, Y. Adaptive Spatial-Spectral dictionary learning for hyperspectral image restoration. *Int. J. Comput. Vis.* **2017**, *122*, 228–245. [[CrossRef](#)]
23. Zhang, L.; Wei, W.; Zhang, Y.; Shen, C.; Van Den Hengel, A.; Shi, Q. Cluster sparsity field: An internal hyperspectral imagery prior for reconstruction. *Int. J. Comput. Vis.* **2018**, *126*, 797–821. [[CrossRef](#)]
24. Paul, A.; Kundu, A.; Chaki, N.; Dutta, D.; Jha, C.S. Wavelet enabled convolutional autoencoder based deep neural network for hyperspectral image denoising. *Multimed. Tools Appl.* **2022**, *81*, 2529–2555. [[CrossRef](#)]
25. Chong, Y.; Zheng, W.; Li, H.; Qiao, Z.; Pan, S. Hyperspectral image compression and reconstruction based on block-sparse dictionary learning. *J. Indian Soc. Remote Sens.* **2018**, *46*, 1171–1186. [[CrossRef](#)]
26. Li, R.; Pan, Z.; Wang, Y.; Wang, P. The correlation-based Tucker decomposition for hyperspectral image compression. *Neurocomputing* **2021**, *419*, 357–370. [[CrossRef](#)]
27. Zikiou, N.; Lahdir, M.; Helbert, D. Support vector regression-based 3D-wavelet texture learning for hyperspectral image compression. *Vis. Comput.* **2020**, *36*, 1473–1490. [[CrossRef](#)]
28. Wang, X.; Xu, T.; Zhang, Y.; Fan, A.; Xu, C.; Li, J. Backtracking Reconstruction Network for Three-Dimensional Compressed Hyperspectral Imaging. *Remote Sens.* **2022**, *14*, 2406. [[CrossRef](#)]
29. Arad, B.; Ben-Shahar, O. Sparse recovery of hyperspectral signal from natural RGB images. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Proceedings, Part VII 14; Springer: Cham, Switzerland, 2016.
30. Tang, Y.; Huang, Z.; Chen, Z.; Chen, M.; Zhou, H.; Zhang, H.; Sun, J. Novel visual crack width measurement based on backbone double-scale features for improved detection automation. *Eng. Struct.* **2023**, *274*, 115158. [[CrossRef](#)]
31. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
32. 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*, 1–14. [[CrossRef](#)]
33. Kwan, C.; Larkin, J. Perceptually Lossless Compression for Mastcam Multispectral Images: A Comparative Study. *J. Signal Inf. Process.* **2019**, *10*, 139–166. [[CrossRef](#)]
34. Valsesia, D.; Magli, E. High-Throughput Onboard Hyperspectral Image Compression With Ground-Based CNN Reconstruction. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 9544–9553. [[CrossRef](#)]
35. Dua, Y.; Kumar, V.; Singh, R.S. Parallel lossless HSI compression based on RLS filter. *J. Parallel Distrib. Comput.* **2021**, *150*, 60–68. [[CrossRef](#)]
36. Mishra, K.; Singh, S.K.; Nagabhushan, P. An Improved SVD based Image Compression. In Proceedings of the 2018 Conference on Information and Communication Technology (ICT), Jabalpur, India, 26–28 October 2018.
37. Egiazarian, K.; Astola, J.; Ponomarenko, N.; Lukin, V.; Battisti, F.; Carli, M. A new full-reference quality metrics based on HVS. In Proceedings of the Second International Workshop on Video Processing and Quality Metrics, Scottsdale, AZ, USA, 22–24 January 2006; Volume 4.
38. Al-Najjar, Y.; Chen, D. Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI. *Int. J. Sci. Eng. Res.* **2012**, *3*, 1–5.
39. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multi Scale Structural Similarity for Image Quality Assessment. In Proceedings of the 37th IEEE Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 9–12 November 2003.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.