sensors

MDPI

Article

A Hierarchical Low-Rank Denoising Model for Remote Sensing Images Based on Deep Unfolding

Fanqi Shao [1], Xiaolin Feng [2], Sirui Tian [2] and Tianyi Zhang [2,*]

[1] China Electronic Technology Group Corporation, Beijing 100846, China; sfanq@163.com
[2] School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China; fengxiaolin@njust.edu.cn (X.F.); tiansirui@njust.edu.cn (S.T.)
* Correspondence: 122104010563@njust.edu.cn; Tel.: +86-159-0438-8640

Abstract: Recently, the low-rank representation (LRR) model has been widely used in the field of remote sensing image denoising due to its excellent noise suppression capability. However, those low-rank-based methods always discard important edge details as residuals, leading to a common issue of blurred edges in denoised results. To address this problem, we take a new look at low-rank residuals and try to extract edge information from them. Therefore, a hierarchical denoising framework was combined with a low-rank model to extract edge information from low-rank residuals within the edge subspace. A prior knowledge matrix was designed to enable the model to learn necessary structural information rather than noise. Also, such traditional model-driven approaches require multiple iterations, and the solutions may be very complex and computationally intensive. To further enhance the noise suppression performance and computing efficiency, a hierarchical low-rank denoising model based on deep unrolling (HLR-DUR) was proposed, integrating deep neural networks into the hierarchical low-rank denoising framework to expand the information capture and representation capabilities of the proposed shallow model. Sufficient experiments on optical images, hyperspectral images (HSI), and synthetic aperture radar (SAR) images showed that HLR-DUR achieved state-of-the-art (SOTA) denoising results.

Keywords: low-rank; hierarchical model; remote sensing image denoising; deep unfolding; edge preservation

check for updates

Citation: Shao, F.; Feng, X.; Tian, S.; Zhang, T. A Hierarchical Low-Rank Denoising Model for Remote Sensing Images Based on Deep Unfolding. *Sensors* **2024**, *24*, 4574. https://doi.org/10.3390/s24144574

Academic Editor: Guangcai Sun

Received: 13 June 2024
Revised: 30 June 2024
Accepted: 13 July 2024
Published: 15 July 2024

1. Introduction

Remote sensing data acquisition often relies on sensors mounted on satellites or aircraft, which can capture electromagnetic spectra in different bands, including visible light, infrared, microwaves, etc. Due to its capability to cover vast geographical areas and provide continuous observations, remote sensing technology has important applications in numerous fields such as agriculture, forestry, geology, meteorology, environmental protection, and urban planning [1]. However, remote sensing images are inevitably affected by various factors during acquisition and transmission, leading to the presence of noise in the images [2].

In recent years, the rapid development of deep learning has injected new vitality into various fields [3,4], particularly in the domain of image processing [5–7]. Utilizing the powerful learning and representational capabilities of deep neural networks can more effectively solve many challenges and problems in image processing [8], which also brings inspiration to the issue of remote sensing image denoising. Compared to traditional image processing methods, deep learning, through an end-to-end training process, can automatically learn high-level, semantically rich feature representations from a large amount of data. These learned features not only have better robustness and generalization capability for image-denoising tasks but also provide richer image descriptions and understanding.

Deep learning methods have been proven to have significant effects in the task of denoising remote sensing images. For optical remote sensing images, a method of remote sensing image denoising based on a Generative Adversarial Network (GAN) named "Restoration Generative Adversarial Network with ResNet and DenseNet" (RRDGAN) was proposed [9]. This method acquires better-quality images by incorporating denoising and SR into a unified framework. Huang et al. integrated deep convolutional neural network (DCNN) denoiser prior into a unidirectional variation (UV) model, named UV-DCNN, to simultaneously destripe and denoise optical remote sensing images [10]. For hyperspectral images (HSI) images, Xie et al. introduced the Deep Spatio-Spectral Bayesian Posterior Network (DSSBPNet) for hyperspectral images [11]. DSSBPNet combines Bayesian variational posterior with deep neural networks, where a deep spatial-spectral network divides the input image into three parts, generating spatial-spectral gradients for each. Different convolutions are used in different parts of the DSS network. Meanwhile, noise estimation of the original data, noise distribution, and sparse noise gradients constitute the Bayesian posterior approach. Finally, the forward-backward propagation method is used to connect DSS with the Bayesian posterior. A novel, deep-learning framework for 3-D HSI denoising [12] was proposed, which decomposes 3-D filtering into 2-D spatial filtering and 1-D spectral filtering. This method can achieve substantial savings on the number of network parameters to keep the computational complexity low. For synthetic aperture radar (SAR) images, a novel two-component deep learning (DL) network was proposed for SAR despeckling [13]. First, the texture estimation subnetwork is constructed to produce the texture level map (TLM), which evaluates the randomness and scale of the texture distribution. Then, the noise removal subnetwork learns a spatially variable mapping between the noise and clean images with the help of TLM.

However, we cannot ignore the important role of traditional models in the field of denoising. Low-rank representation (LRR) models have been widely explored and improved in the field of remote sensing image denoising [14,15]. A method called NAILRMA [16] proposed an iterative regularization framework aiming to achieve denoising of signal subspaces. In NAILRMA, an adaptive iteration factor selection method based on noise variance was applied to each band of the hyperspectral image (HSI). Additionally, some tensor-based low-rank denoising methods [17–19] have also demonstrated satisfactory results. The Fisher–Tippett distribution-based WNNM (FT-WNNM) [20] attempts to recover underlying low-rank components from patch group matrices. The Composite Regularization Method for Spot Noise Reduction (CRM-SR) [21] introduces regularization terms, including TV regularization and NLLR regularization. However, determining the regularization parameters involved in the CRM-SR model is challenging. The NLLR framework [22] decomposes speckle noise into low-rank components with proposed truncation and weighted nuclear norm regularization. Considering the effectiveness of the LRR model, more and more denoising algorithms have combined low-rank models with deep learning denoising methods.

Zhang et al. proposed a Deep Low-Rank Prior (DLRP) [23] for remote sensing image denoising, firstly using the low-rank characteristics of nearby non-local self-similar blocks arranged in dictionary order to model the Global Objective Function (GOF). Secondly, with the help of an alternating iterative strategy, GOF can be easily decomposed into two independent sub-problems. Among them, the low-rank minimization denoising problem can be solved through learning with a deep convolutional neural network. Nguyen et al. proposed an HSI denoising method using deep image prior with sparse and low-rank prior [24]. However, hyperparameter selection in this method is still an open question. Sun et al. proposed a HSI denoising via LRR and CNN denoiser [25]. The sparse-based low-rank representation can explore the global correlations in both the spatial and spectral domains, and the CNN-based denoiser can represent the deep prior, which cannot be designed by traditional restoration models. However, such denoising methods often have a common problem, that is, they will filter out significant edge signals as noise, which will degrade the quality of images. Zhang et al. developed a self-supervised HSI denoising method via integrating a model-driven strategy with a data-driven strategy [26]. The proposed framework simultaneously cooperates with the spectral low-rankness prior and deep spatial prior (SLRP-DSP) for HSI self-supervised denoising. Regarding the characteristics of SAR images, more novel deep networks were proposed. Xiong et al. decomposed the SAR image in the SAR imaging-despeckling observation model into a sparse matrix and a low-rank matrix, and then established an optimization problem with the corresponding sparse and low-rank priors [27]. A despeckling model was proposed that uses deeper convolutional neural networks [28], which was never used before, as far as authors are concerned, for diminishing speckles in noisy SAR images. Multiple skip connections from the ResNet model are also employed in the authors' proposed architecture. However, these denoising methods often do not take into account the issue of edge preservation in the denoising results, leading to the loss of some important structural information in the denoised images.

In this paper, we combine the shallow low-rank denoising model with an autoencoder, following the method in [29], to construct a two-level deep encoder. By applying deep unfolding, the shallow LRR denoising model was introduced into the realm of deep learning. Thus, a hierarchical low-rank denoising model based on deep unrolling (HLR-DUR) was proposed. Integrating the characteristics of autoencoders, a two-level deep autoencoder was established to optimize the iterative solution process for the low-rank and edge components of remote sensing images. Utilizing the advantages of deep neural networks, the model captures more complex patterns in the data, thereby enhancing the effectiveness of image denoising. Finally, experiments conducted on optical remote sensing images, hyperspectral images, and SAR images demonstrated that HLR-DUR significantly outperforms shallow models across various metrics.

The main contributions can be summarized as follows:

1. We think the residuals of traditional low-rank models contain not only noise but also high-frequency edge information that has been filtered out. Therefore, we tried to re-extract edge information from the residuals to construct a hierarchical low-rank model for denoising remote sensing images. To better achieve edge extraction, we introduced manifold learning into the model, ensuring that useful structural information, rather than noise, is extracted in the edge subspace. Additionally, a new prior knowledge regulation was designed to distinguish between clean pixels and noisy pixels, guiding the model to learn useful information during the denoising process;
2. Due to the number of iterations required, traditional LRR models usually take a long time to process large-sized remote sensing images. To tackle this problem, we designed a hierarchical denoising model based on deep unfolding by combining shallow models with deep autoencoders. By mapping traditional iterative optimization algorithms onto the structure of neural networks, each iterative step was transformed into a layer of the network. This technique enhances the interpretability of neural networks, making each layer no longer a black box model but rather corresponding to a specific optimization step;
3. Experiments conducted on three types of images—optical remote sensing images, hyperspectral images, and SAR images—demonstrated the effectiveness of our proposed HLR-DUR model. HLR-DUR not only enhances denoising performance but also better preserves sufficient edge detail compared to existing methods, achieving better edge-preserving denoising.

2. Related Work

Wang et al. proposed that a low-rank prior and a fully convolutional Auto Encoder (AE) can be incorporated through modeling an energy minimization problem [30], which is similar to the Robust Principal Component Analysis (RPCA) model.

Given a data matrix $\mathbf{X}$, it can be decomposed as [31]:

$$\mathbf{X} = \mathbf{Z} + \mathbf{E} \tag{1}$$

where **Z** represents the low-rank part and E is a sparse matrix, which represents residual. **Z** and **E** can then be optimized by solving the following energy minimization problem:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_1 \\ &s.t.\mathbf{X} = \mathbf{Z} + \mathbf{E} \end{aligned} \tag{2}$$

where $\|\cdot\|_*$ denotes nuclear norm, $\|\cdot\|_1$ denotes l_1 norm, and λ is the regularization parameter.

Based on this model, the low-rank prior and a fully convolutional AE are incorporated by modeling an energy minimization problem, which can be mathematically depicted as follows:

$$\begin{aligned} &\min_{\mathbf{Z},\theta} \tfrac{1}{2}\|(f_\theta(\mathbf{Y}) - \mathbf{X})\mathbf{W}\|^2 + \lambda\|\mathbf{Z}\|_* \\ &s.t.\mathbf{Z} = f_\theta(\mathbf{Y}) \end{aligned} \tag{3}$$

where **Y** has the same dimensionality as the hyperspectral image, $f_\theta(\mathbf{Y})$ is the network which uses the uniform distribution as a prior to generate a reconstruction of the background, and **W** is the adaptive-weighted map.

Then, the problem can be solved by the Alternating Direction Method of Multipliers (ADMM). The model can be described as

$$\min_{\mathbf{Z},\theta} \frac{1}{2}\|(f_\theta(\mathbf{Y}) - \mathbf{X})\mathbf{W}\|^2 + \lambda\|\mathbf{Z}\|_* + \frac{\mu}{2}\|\mathbf{Z} - f_\theta(\mathbf{Y})\|^2 + \mathbf{S}^T(\mathbf{Z} - f_\theta(\mathbf{Y})) \tag{4}$$

where **S** stands for the Lagrange multiplier, and μ is a penalty parameter.

Finally, the above-shown optimization problem can be split into three subproblems, and each subproblem can be solved alternately in an iterative procedure.

3. Proposed Methodology

As previously mentioned, the presence of noisy data can destroy the low-rank structure of the data. So, the LRR model splits the data into clean parts and noisy parts, separating the noisy data while maintaining the low-rank structure of the data. However, the LRR model has a drawback that cannot be ignored. Specifically, it strives to maintain a low-rank structure by filtering out useful high-frequency information, which then causes important image edges to be discarded as part of residuals. Motivated by this phenomenon, we first try to construct a hierarchical low-rank model to extract edge details from residuals. Then, we try to combine the proposed hierarchical model with deep neural networks to further improve the denoising effect.

3.1. Hierarchical Low-Rank Model

3.1.1. Model Formulation

To address the issue of traditional low-rank models failing to preserve structural edge information, we modify the traditional low-rank model using a prior knowledge matrix. This modification allows the model to better distinguish between clean and noisy pixels. The model can be represented as

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{Z}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\ &s.t.\ \mathbf{H} \circ \mathbf{X} = \mathbf{H} \circ \mathbf{Z} + \mathbf{E} \end{aligned} \tag{5}$$

where **X** represents the original noisy image, **Z** represents the low-rank component, and **E** represents the low-rank residual. **H** is the prior knowledge matrix used in this model, "$\circ$" denotes the Hadamard product, and λ is an adjustable parameter used to balance the residual term.

It is crucial to extract the edge information that has been filtered out from the residuals. Building on (5), we innovatively introduce a constraint term $\mathbf{H} \circ \mathbf{E} = \mathbf{H} \circ \mathbf{E_1} + \mathbf{H} \circ \mathbf{E_2}$, where $\mathbf{E_1}$ represents the edge part extracted from the residual, and $\mathbf{E_2}$ represents the noise component in the residual. By further decomposing the residual, the filtered edge information can be obtained, thereby enhancing the quality of the denoising results. However, considering the different requirements for extracting residuals and low-rank components, we further set up hierarchical prior knowledge information, implemented through different coefficients, to meet different requirements. Specifically, when extracting the low-rank part, the goal is to achieve as clean denoising results as possible, that is, the less noise, the better, even if it means that the low-rank part of the image may have blurred edges and lose some detail. When extracting edge structures, the focus is more on effectively extracting rich edge information, even if some noise is introduced in the process of extracting edges. To achieve this goal, we further propose a hierarchical prior knowledge to better extract the edge details from the residuals, with the model presented as follows:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E},\mathbf{E_1},\mathbf{E_2}} \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1} \\ &s.t.\ \mathbf{H_1} \circ \mathbf{X} = \mathbf{H_1} \circ \mathbf{Z} + \mathbf{E} \\ &\quad \mathbf{H_2} \circ \mathbf{E} = \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{E_2} \end{aligned} \tag{6}$$

It is noteworthy that in the constraints of (6), $\mathbf{H}$ matrices with different subscripts appear, as a special design of the hierarchical model. Here, $\mathbf{H}_1$ is used to obtain the clean low-rank part from the noisy image, while $\mathbf{H}_2$ is used to find more edge details from the residuals. As mentioned above, $\mathbf{H}_1$ should be more stringent to ensure that as much noise as possible is eliminated. In contrast, the constraints on $\mathbf{H}_2$ are relatively lenient because $\mathbf{H}_2$ needs to extract more useful information from the residuals.

However, extracting sufficient edge information from residuals containing a large amount of noise is not an easy task. To ensure that our model learns the required edge information rather than noise, and to ensure that good data geometric structures are captured within the model, a manifold learning framework was also introduced to the proposed model [32]. Manifold learning is a machine learning technique that explores and exploits the underlying low-dimensional structures of high-dimensional data, based on the idea that while data may be presented in high-dimensional spaces, they actually distribute along some low-dimensional manifolds. The purpose of manifold learning is to reveal these low-dimensional structures to better understand the essential characteristics of the data. By introducing manifold learning, robust subspace projections $\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r)$ can be learned from the residuals $\mathbf{E_1}$, allowing for the extraction of edge subspaces. By introducing dynamic affinity graph regularization, manifold learning can be incorporated into the original model, with the formula represented as follows:

$$\begin{aligned} &\min_{\mathbf{V},\mathbf{Z},\mathbf{W},\mathbf{E},\mathbf{E_1},\mathbf{E_2}} \tfrac{1}{2}\sum_{i,j} \|\mathbf{V^T}\mathbf{e}_{1,i} - \mathbf{V^T}\mathbf{e}_{1,j}\|_2^2 \mathbf{W}_{ij} + \lambda \|\mathbf{W}\|_F^2 \\ &s.t.\ \mathbf{H_1} \circ \mathbf{X} = \mathbf{H_1} \circ \mathbf{Z} + \mathbf{E} \\ &\quad \mathbf{H_2} \circ \mathbf{E} = \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{E_2} \\ &\quad \mathbf{V^T}\mathbf{V} = \mathbf{I} \\ &\quad \mathbf{W}1 = 1, \mathbf{W}_{ii} = 0, \mathbf{W} \geq 0 \end{aligned} \tag{7}$$

where $\mathbf{e}_{1,i}$ and $\mathbf{e}_{1,j}$ represent the values of corresponding samples in the residual $\mathbf{E_1}$, and $1/2\sum_{i,j} \|\mathbf{V^T}\mathbf{e}_{1,i} - \mathbf{V^T}\mathbf{e}_{1,j}\|_2^2$ is an increased coupling term. It is a part of our model because we do not know which part of the data is affected by noise.$\mathbf{W}$ is the dynamic affinity graph matrix, and λ is an adjustable parameter used to balance the dynamic affinity graph.

For two similar samples $\mathbf{x}_i$ and $\mathbf{x}_j$, if the values of corresponding $\mathbf{e}_{1,i}$ and $\mathbf{e}_{1,j}$ are very small, this also suggests that the distance between them will be very small. Therefore, it can be considered that they primarily contain edge information, which are the edge details to be extracted from the residual $\mathbf{E}$. Conversely, if they are part of the noise, the distance between them will be increased. Based on this assumption, a weighting coefficient $\mathbf{W}_{ij}$ is introduced to adaptively learn projections $\mathbf{V}$ with similar residuals. When two residual samples are similar, $\mathbf{W}_{ij}$ is set to non-zero. Otherwise, $\mathbf{W}_{ij}$ is set to zero to mitigate the impact of noise. At the same time, three non-negative constraints $\mathbf{W}_{ii} = 0$, $\mathbf{W} \geq 0$, and $\mathbf{W}1 = 1$ are also added to the model, and these three constraints are used to ensure that the affinity graph will be normalized. Finally, the complete model is as follows:

$$\begin{aligned} &\min_{\mathbf{V},\mathbf{Z},\mathbf{W},\mathbf{E},\mathbf{E_1},\mathbf{E_2}} \frac{1}{2}\sum_{i,j} \|\mathbf{V^T}\mathbf{e}_{1,i} - \mathbf{V^T}\mathbf{e}_{1,j}\|_2^2 \mathbf{W}_{ij} + \lambda_1\|\mathbf{Z}\|_* + \lambda_2\|\mathbf{E}\|_{2,1} + \lambda_3\|\mathbf{W}\|_F^2 \\ &s.t.\ \mathbf{H_1} \circ \mathbf{X} = \mathbf{H_1} \circ \mathbf{Z} + \mathbf{E} \\ &\quad \mathbf{H_2} \circ \mathbf{E} = \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{E_2} \\ &\quad \mathbf{V^T}\mathbf{V} = \mathbf{I} \\ &\quad \mathbf{W}1 = 1, \mathbf{W}_{ii} = 0, \mathbf{W} \geq 0 \end{aligned} \tag{8}$$

where λ_1, λ_2, λ_3 are adjustable parameters used to balance the different terms.

To solve (8), we employed the Augmented Lagrange Multiplier method (ALM). The coupling term $1/2\sum_{i,j} \|\mathbf{V^T}\mathbf{e}_{1,i} - \mathbf{V^T}\mathbf{e}_{1,j}\|_2^2$ can be rewritten as $tr(\mathbf{V^T}(\mathbf{FL\hat{E}^T})\mathbf{V})$, where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the graph Laplacian operator of $\mathbf{W}$, and $\mathbf{D} = diag\sum_i \mathbf{W}_{ij}$ is the diagonal matrix. Then, a relaxation variable $\mathbf{S}$ to denote $\mathbf{Z}$, a relaxation variable $\mathbf{E_1}$ to denote $\hat{\mathbf{E}}$ and a relaxation variable $\mathbf{R}$ to denote $\mathbf{E}$ are introduced to make (8) easier to solve. Additionally, another variable $\mathbf{F}$ is introduced to avoid the Sylvester equation during the solution process for $\mathbf{E}$. Therefore, (8) is reformulated as

$$\begin{aligned} &\min_{\mathbf{V},\mathbf{Z},\mathbf{W},\mathbf{E},\mathbf{E_1},\mathbf{E_2}} tr(\mathbf{V^T}(\mathbf{FL\hat{E}^T})\mathbf{V}) + \lambda_1\|\mathbf{Z}\|_* + \lambda_2\|\mathbf{E}\|_{2,1} + \lambda_3\|\mathbf{W}\|_F^2 \\ &s.t.\ \mathbf{H_1} \circ \mathbf{X} = \mathbf{H_1} \circ \mathbf{Z} + \mathbf{E} \\ &\quad \mathbf{H_2} \circ \mathbf{E} = \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{E_2} \\ &\quad \mathbf{V^T}\mathbf{V} = \mathbf{I} \\ &\quad \mathbf{W}1 = 1, \mathbf{W}_{ii} = 0, \mathbf{W} \geq 0 \\ &\quad \mathbf{S} = \mathbf{Z}, \mathbf{E_1} = \hat{\mathbf{E}}, \mathbf{F} = \hat{\mathbf{E}}, \mathbf{R} = \mathbf{E} \end{aligned} \tag{9}$$

3.1.2. Prior Knowledge Regulation Construction

Firstly, we constructed the basic prior knowledge matrix based on the work of ROLD [33]. ROLD is a rank-ordered logarithmic difference, which can identify more noisy pixels with fewer false hits. With such statistics as cornerstones, we tried to construct our prior knowledge matrix. The uncorrupted probability h_i of each pixel k_i using ROLD is as follows:

$$h_i = \exp(-\alpha ROLD(k_i)) \tag{10}$$

where α is the coefficient used to obtain image features under different requirements. When α is large, the difference between uncontaminated pixels and contaminated pixels will be widened, and the final result will filter most speckle noise, but more details such as edges will also be filtered out together. On the contrary, when we set α small, it is more likely to retain the edge details of images even though some speckle noise may also be left on images.

Furthermore, we adapted the abovementioned prior knowledge matrix to make it more compatible with SAR images. The backscattered signatures from point targets are dominated by a small number of strong scatterers, which will make them classified as noisy pixels. Therefore, we adopted the method of the Enhanecd Lee filter (EnLee) [34] to divide all pixels into three categories, and combined it with (10) to make the prior knowledge matrix more suitable for SAR images. Through using the hierarchical prior knowledge matrices, our method can better learn the intrinsics of the low-rank part and edge part, and obtain clearer results.

Usually, two classes are considered in traditional filters: homogeneous and heterogeneous. Enhanced Lee filter presented that isolated point targets need special treatment as a third class. For an NLOOK intensity image, it is filtered as follows:

$$\hat{R}(t) = \begin{cases} \bar{I}, C_i < C_{\min} \\ \bar{I}(t) + \omega(t)(I(t) - \bar{I}(t)) \quad , C_{\min} \leq C_i \leq C_{\max} \\ I, C_i > C_{\max} \end{cases} \tag{11}$$

where $C_i = \frac{std}{\bar{I}}$, std is the standard deviation within the filter window. $C_{\min} = \frac{1}{\sqrt{NLOOK}}$, $C_{\max} = \sqrt{1 + \frac{2}{NLOOK}}$.

The weight function can be written as

$$\omega(t) = \exp(\frac{-(C_i - C_{\min})}{(C_{\max} - C_i)}) \tag{12}$$

Learning from EnLee, we divide all pixels into three categories to obtain the weight matrix ε:

(1) $C_i > C_{\max}$, which means that this pixel is a point target and needs to be preserved. So, we set the corresponding value in ε to 0.9 (a very big value).
(2) $C_{\min} < C_i < C_{\max}$, update ε as $1 - \omega$ according to (12).
(3) $C_i < C_{\min}$, which represents that this pixel is a noisy pixel and needs to be smoothed. So, we set the corresponding value in ω to 0.1 (a respectively small value).

Then, the new prior knowledge matrix is constructed as follows:

$$\mathbf{H_i} = h_i \circ \varepsilon \tag{13}$$

Finally, $\mathbf{H_i}$ needs to be normalized to represent the probability of 0–1.

3.2. Deep Unfolding Model HLR-DUR Based on Autoencoder

In recent years, deep low-rank representation models have received increasing attention as a potential solution for designing interpretable neural networks. By mapping traditional iterative optimization algorithms into the structure of neural networks, each iterative step is transformed into a layer of the network, enhancing the interpretability of neural networks. Each layer of the network is no longer a black-box model but corresponds to a specific optimization step. In the unfolded network, models based on traditional physical meanings are used as the basic architecture, and the parameters that need manual adjustment are transformed into trainable parameters of the network. The core idea of the deep unfolding technique is to utilize deep neural networks to simulate and optimize the iterative process of low-rank representation. Therefore, based on (9), the new model can be described as

$$\begin{aligned} &\min_{\mathbf{V},\mathbf{Z},\mathbf{S},\mathbf{W},\mathbf{E},\mathbf{E_1},\mathbf{E_2},\hat{\mathbf{E}},\mathbf{F},\mathbf{R}} tr(\mathbf{V^T}(\mathbf{F}\mathbf{L}\hat{\mathbf{E}}^{\mathbf{T}})\mathbf{V}) + \lambda_1\|\mathbf{S}\|_* + \gamma_1\|\mathbf{Z} - f_{de1}(\mathbf{X};\theta_1)\|_F^2 \\ &+\lambda_2\|\mathbf{E}\|_{2,1} + \gamma_2\|\mathbf{E_1} - f_{de2}(\mathbf{E};\theta_2)\|_F^2 + \lambda_3\|\mathbf{W}\|_F^2 \\ &s.t.\ \mathbf{H_1} \circ \mathbf{X} = \mathbf{H_1} \circ \mathbf{Z} + \mathbf{E}, \mathbf{H_2} \circ \mathbf{E} = \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{E_2} \\ &\quad \mathbf{E_1} = \hat{\mathbf{E}}, \mathbf{S} = \mathbf{Z}, \mathbf{R} = \mathbf{E} \\ &\quad \mathbf{V^T}\mathbf{V} = \mathbf{I}, \mathbf{W}1 = 1, \mathbf{W}_{ii} = 0, \mathbf{W} \geq 0 \end{aligned} \tag{14}$$

where $f_{de1}(\mathbf{X};\theta_1) = g(W^L \ldots g(W^i \ldots g(W^2\mathbf{X})))$ denotes a multi-layer deep autoencoder, $\theta_1 = \{W^2, \ldots, W^L\}$ is a set of learning parameters used for learning the low-rank component. Additionally, $f_{de2}(\mathbf{E};\theta_2)$ is used to extract edge part, and θ_2 is also a set of learning parameters.

Before solving for the optimal solution of each variable in (14), it is necessary to first obtain its Augmented Lagrangian Multiplier (ALM) function; the complete equation is as follows:

$$\begin{aligned}
\min_{\mathbf{V},\mathbf{Z},\mathbf{S},\mathbf{W},\mathbf{E},\hat{\mathbf{E}},\mathbf{E_1},\mathbf{E_2},\mathbf{F},\mathbf{R}} \quad & tr(\mathbf{V^T}(\mathbf{FL\hat{E}^T})\mathbf{V}) + \lambda_1\|\mathbf{S}\|_* + \lambda_2\|\mathbf{E}\|_{2,1} + \lambda_3\|\mathbf{W}\|_F^2 \\
& +\gamma_1\|\mathbf{Z} - f_{de1}(\mathbf{X};\theta_1)\|_F^2 + \gamma_2\|\mathbf{E_1} - f_{de2}(\mathbf{E};\theta_2)\|_F^2 \\
& +tr(Y_1^T(\mathbf{H_1} \circ \mathbf{X} - \mathbf{H_1} \circ \mathbf{Z} - \mathbf{E})) + tr(Y_2^T(\mathbf{H_2} \circ \mathbf{R} - \mathbf{H_2} \circ \mathbf{E_1} - \mathbf{E_2})) \\
& +tr(Y_3^T(\mathbf{S} - \mathbf{Z})) + tr(Y_4^T(\hat{\mathbf{E}} - \mathbf{E_1})) + tr(Y_5^T(\mathbf{F} - \hat{\mathbf{E}})) + tr(Y_6^T(\mathbf{R} - \mathbf{E})) \\
& +\frac{\mu}{2}(\|\mathbf{H_1} \circ \mathbf{X} - \mathbf{H_1} \circ \mathbf{Z} - \mathbf{E}\|_F^2 + \|\mathbf{H_2} \circ \mathbf{R} - \mathbf{H_2} \circ \mathbf{E_1} - \mathbf{E_2}\|_F^2 \\
& +\|\mathbf{S} - \mathbf{Z}\|_F^2 + \|\hat{\mathbf{E}} - \mathbf{E_1}\|_F^2 + \|\mathbf{F} - \hat{\mathbf{E}}\|_F^2 + \|\mathbf{R} - \mathbf{E}\|_F^2) \\
s.t.\ & \mathbf{V^T V} = \mathbf{I} \\
& \mathbf{W}1 = 1, \mathbf{W}_{ii} = 0, \mathbf{W} \geq 0
\end{aligned} \tag{15}$$

where Y_1, Y_2, Y_3, Y_4, Y_5 and Y_6 are the introduced Lagrange multiplier.

By establishing a two-level deep encoder, the traditional hierarchical low-rank model is extended to a deep learning network. In traditional iterative algorithms, the denoising task is typically addressed through multiple iterative steps, with each step being computed based on the output of the previous one. In deep unfolding, these iterative steps are "unfurled" into a sequence of layers, each layer corresponding to an iterative step. Within these unfolded layers, the iterative process of low-rank representation is simulated and optimized through a deep autoencoder network.

The flowchart of the proposed model is depicted in Figure 1. Initially, the noisy image undergoes preprocessing with a prior knowledge matrix. Subsequently, the preprocessed noisy image is fed into HLR-DUR to obtain the low-rank and edge components. Finally, the denoised edge and low-rank components are merged to produce a clean image.

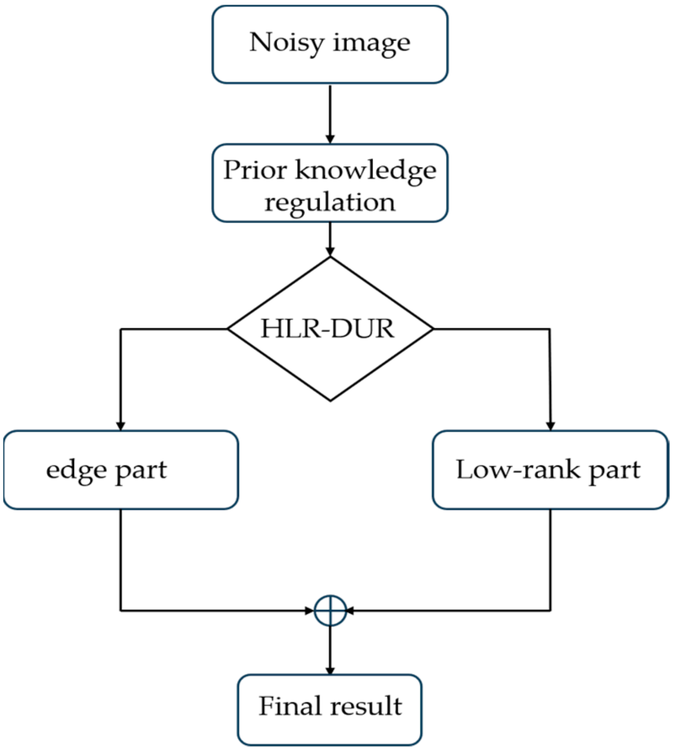


Figure 1. The flowchart of the proposed method.

The above iterative process is unfolded into a neural network, whose general architecture and components are shown in Figure 2. It consists of four main modules: the low-rank module, the edge module, the update module for the other parameters, and the Lagrangian multiplier module. Each module corresponds to the subproblem.

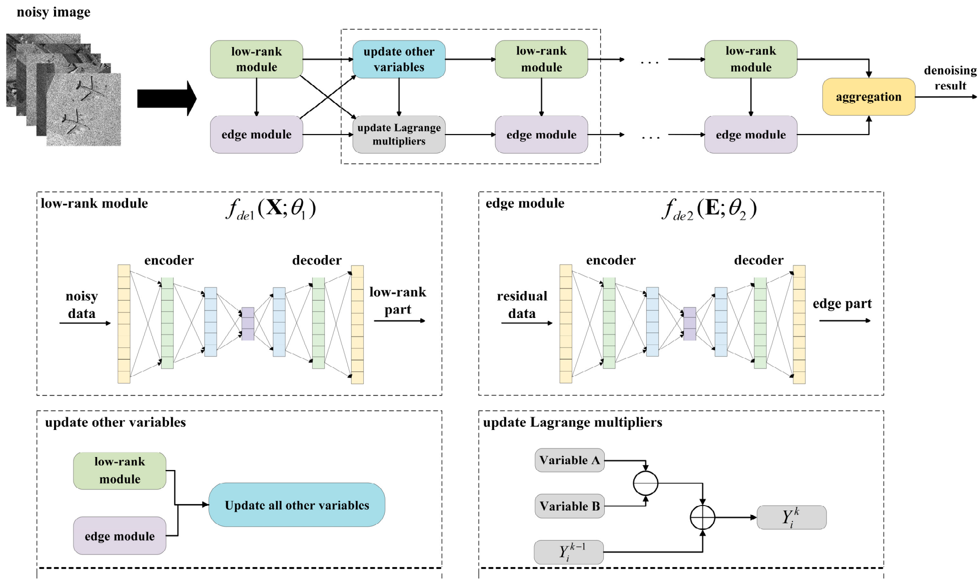


Figure 2. The structure of the HLR-DUR model.

Next, we introduce each module.

- low-rank module

The low-rank module is used to compute the clean low-rank component extracted from noisy data. θ_1 is computed by minimizing $L_{\theta 1}$ with respect to the weight parameters θ_1 while keeping other parameters fixed. The optimal solution is obtained by solving the following subproblem:

$$L_{\theta 1} = \|\mathbf{Z} - f_{de1}(\mathbf{X};\theta_1)\|_F^2 \tag{16}$$

The optimization is performed using the Gradient Descent (GD) algorithm, and the formula can be written as follows:

$$\theta_1 = \theta_1 - \zeta \partial L_{\theta 1}/\partial \theta_1 \tag{17}$$

where ζ is the learning rate and $\partial L_{\theta 1}/\partial \theta_1$ represents the gradient of $L_{\theta 1}$.

- edge module

The edge module is used to extract useful structural edge information from the residual part. Its solution process is similar to that of the low-rank module, which can be described as

$$L_{\theta 2} = \|\mathbf{E_1} - f_{de2}(\mathbf{E};\theta_2)\|_F^2 \tag{18}$$

$$\theta_2 = \theta_2 - \zeta \partial L_{\theta 2}/\partial \theta_2 \tag{19}$$

- other variables

This module is used to update other variables apart from the low-rank part and the edge part. We have eight variables that need to be updated: $\mathbf{V}, \mathbf{S}, \mathbf{W}, \mathbf{E}, \hat{\mathbf{E}}, \mathbf{E_2}, \mathbf{F}, \mathbf{R}$. We adopt the alternating direction method of multipliers (ADMM) [35] strategy to accelerate convergence.

V subproblem:

Firstly, update **V** by fixing the other variables, and this subproblem can be expressed as

$$\begin{aligned}&\min_{\mathbf{V}} tr(\mathbf{V^T}(\mathbf{FL\hat{E}^T})\mathbf{V})\\&s.t.\ \mathbf{V^T V}=\mathbf{I}\end{aligned} \tag{20}$$

Denote $\mathbf{FL\hat{E}^T}=\mathbf{Q}$, then we can obtain

$$\begin{aligned}&\min_{\mathbf{V}} tr(\mathbf{V^T Q V})\\&s.t.\ \mathbf{V^T V}=\mathbf{I}\end{aligned} \tag{21}$$

V is obtained using the above-shown eigenvalue decomposition problem whose optimal solution involves the set of k eigenvectors of the topmost k smallest eigenvalues of **Q**.

S subproblem:

$$\min_{\mathbf{S}} \lambda_1 \|\mathbf{S}\|_* + \frac{\mu}{2}\|\mathbf{S}-\mathbf{Z}+\frac{Y_3}{\mu}\|_F^2 \tag{22}$$

Using the singular value thresholding (SVT), the optimal **S** is obtained as

$$\mathbf{S}=\Phi_{\frac{\lambda_1}{\mu}}(\mathbf{Z}-\frac{Y_3}{\mu}) \tag{23}$$

where Φ represents the soft-thresholding operator.

W subproblem:

$$\begin{aligned}&\min_{\mathbf{W}} \frac{1}{2}\sum_{i,j}\|\mathbf{V^T e}_{1,i}-\mathbf{V^T e}_{1,j}\|_2^2 \mathbf{W}_{ij}+\lambda_3\|\mathbf{W}\|_F^2\\&s.t.\ \mathbf{W}1=1, \mathbf{W}_{ii}=0, \mathbf{W}\geq 0\end{aligned} \tag{24}$$

We can find that (24) is independent for each i, so $\mathbf{W}_i$ can be solved separately as follows:

$$\begin{aligned}&\min_{\mathbf{W}_i} \frac{1}{2}\sum_{i,j}\|\mathbf{V^T e}_{1,i}-\mathbf{V^T e}_{1,j}\|_2^2 \mathbf{W}_{ij}+\lambda_3\|\mathbf{W}_i\|_F^2\\&s.t.\ \mathbf{W}1=1, \mathbf{W}_{ii}=0, \mathbf{W}\geq 0\end{aligned} \tag{25}$$

E subproblem:

$$\min_{\mathbf{E}} \lambda_2\|\mathbf{E}\|_{2,1}+\frac{\mu}{2}\|\mathbf{H_1}\circ\mathbf{X}-\mathbf{H_1}\circ\mathbf{Z}-\mathbf{E}+\frac{Y_1}{\mu}\|_F^2+\frac{\mu}{2}\|\mathbf{R}-\mathbf{E}+\frac{Y_6}{\mu}\|_F^2 \tag{26}$$

Denote $\mathbf{T}=\frac{1}{2}(\mathbf{H_1}\circ\mathbf{X}-\mathbf{H_1}\circ\mathbf{Z}+\mathbf{R}+\frac{Y_1}{\mu}+\frac{Y_6}{\mu})$, $\tau=\frac{\lambda_2}{\mu}$.

We can obtain the i-th column of the variable **E** as

$$\mathbf{E}(:,i)=\begin{cases}\frac{\|t_i\|-\tau}{\|t_i\|}t_i\ ,\ if\ \tau<\|t_i\|\\ 0\ ,\ otherwise\end{cases} \tag{27}$$

Ê subproblem:

$$\min_{\mathbf{\hat{E}}} tr(\mathbf{V^T}(\mathbf{FL\hat{E}^T})\mathbf{V})+\mu\|\mathbf{\hat{E}}-\mathbf{E_1}+\frac{Y_4}{\mu}\|_F^2+\mu\|\mathbf{F}-\mathbf{\hat{E}}+\frac{Y_5}{\mu}\|_F^2 \tag{28}$$

Set $\partial/\partial\mathbf{\hat{E}}=0$, the optimized result can be obtained as

$$\hat{\mathbf{E}}=\frac{2\mu\mathbf{E_1}-\mathbf{VV^T FL}-2Y_4+2\mu\mathbf{F}+2Y_5}{4\mu} \tag{29}$$

$\mathbf{E_2}$ subproblem:

$$\min_{\mathbf{E_2}} \|\mathbf{H_2} \circ \mathbf{R} - \mathbf{H_2} \circ \mathbf{E_1} - \mathbf{E_2} + \frac{Y_2}{\mu}\|_F^2 \tag{30}$$

$$\mathbf{E_2} = (\mathbf{H_2} \circ \mathbf{R}) - (\mathbf{H_2} \circ \mathbf{E_1}) + \frac{Y_2}{\mu} \tag{31}$$

$\mathbf{F}$ subproblem:

$$\min_{\mathbf{F}} tr(\mathbf{V}^{\mathbf{T}}(\mathbf{F}\mathbf{L}\hat{\mathbf{E}}^{\mathbf{T}})\mathbf{V}) + \mu\|\mathbf{F} - \hat{\mathbf{E}} + \frac{Y_5}{\mu}\|_F^2 \tag{32}$$

$$\mathbf{F} = \frac{2\mu\mathbf{E} - \mathbf{V}\mathbf{V}^{\mathbf{T}}\hat{\mathbf{E}}\mathbf{L}^{\mathbf{T}} - 2Y_5}{2\mu} \tag{33}$$

$\mathbf{R}$ subproblem:

$$\min_{\mathbf{R}} \|\mathbf{H_2} \circ \mathbf{R} - \mathbf{H_2} \circ \mathbf{E_1} - \mathbf{E_2} + \frac{Y_2}{\mu}\|_F^2 + \|\mathbf{R} - \mathbf{E} + \frac{Y_6}{\mu}\|_F^2 \tag{34}$$

$$\mathbf{R} = (\mathbf{H_2} \circ \mathbf{H_2} \circ \mathbf{E_1} + \mathbf{H_2} \circ \mathbf{E_2} - \mathbf{H_2} \circ \frac{Y_2}{\mu} + \mathbf{E} - \frac{Y_6}{\mu}) \div (1 + \mathbf{H_2} \circ \mathbf{H_2}) \tag{35}$$

where ÷ represents element-wise division.

- Lagrange multipliers

  In this section, we compute and update the Lagrange multipliers.

$$Y_1 = Y_1 + \mu(\mathbf{H_1} \circ \mathbf{X} - \mathbf{H_1} \circ \mathbf{Z} - \mathbf{E}) \tag{36}$$

$$Y_2 = Y_2 + \mu(\mathbf{H_2} \circ \mathbf{R} - \mathbf{H_2} \circ \mathbf{E_1} - \mathbf{E_2}) \tag{37}$$

$$Y_3 = Y_3 + \mu(\mathbf{S} - \mathbf{Z}) \tag{38}$$

$$Y_4 = Y_4 + \mu(\hat{\mathbf{E}} - \mathbf{E_1}) \tag{39}$$

$$Y_5 = Y_5 + \mu(\mathbf{F} - \hat{\mathbf{E}}) \tag{40}$$

$$Y_6 = Y_6 + \mu(\mathbf{R} - \mathbf{E}) \tag{41}$$

4. Results

To better compare the denoising results of deep learning methods on remote sensing images, we selected the following classical deep learning algorithms as comparative methods: DnCNN [36], FFDNet [37], HSID-CNN [38], GRN [39], DD-CNN [40], and PDSNet [41].

In Section 4.1, we use the Mean Peak Signal-to-Noise Ratio (MPSNR) and Mean Structural Similarity Index (MSSIM) as evaluation metrics. In Section 4.3, we employ the Equivalent Number of Looks (ENL) and Edge Preservation Index (EPI) for evaluation. Our specific parameter settings are as follows: batch size of 150, ADAM optimizer, and learning rate of 5×10^{-3}. For selecting parameters such as rank threshold, hierarchical levels, and regularization terms, we integrated our proposed HLR-DUR model into an AutoML tool. Across different datasets, we employed cross-validation to automatically search for optimal parameters.

4.1. Experiments on Optical Images

4.1.1. Dataset Description

In this section, we selected the NWPU-RESISC45 dataset as the experimental subject. The NWPU-RESISC45 dataset is an optical remote sensing image dataset, which contains 45 different types of natural and man-made scenes, such as airports, bridges, forests, lakes, ports, etc., with approximately 700 different images for each type. These images are known for their high resolution and diversity, making them highly suitable for training and testing computer vision algorithms for image recognition, classification, and denoising. In order to conduct our experiments, the dataset was converted into single-channel grayscale images, and Gaussian noise with different variances of 0.01, 0.02, 0.04, 0.06, 0.08, and 0.1 was added to conduct the denoising experiments.

4.1.2. Experiments and Analysis

Noise with a variance of 0.04 is added to the first column of images in Figures 3 and 4 show the denoising results for each comparison algorithm In Figure 4f, it is evident that DD-CNN exhibits the poorest visual performance, characterized by noticeable noise artifacts in the image and significant blur and deformation along the edges of the airplane. While DnCNN managed to recover the overall structure of the noise-free image, it suffered severe blurring in edge details. The lines in the top-left corner of the image were nearly entirely filtered out, and the airplane's edges in the top-right corner appeared notably blurred. GRN retained some crucial edge information compared to the others but still exhibited noticeable noise residues, resulting in subpar visual quality. In contrast, FFDNet, HSID-CNN, PDSNet, and HLR-DUR achieved comparatively better denoising results, preserving more structural information in the image.

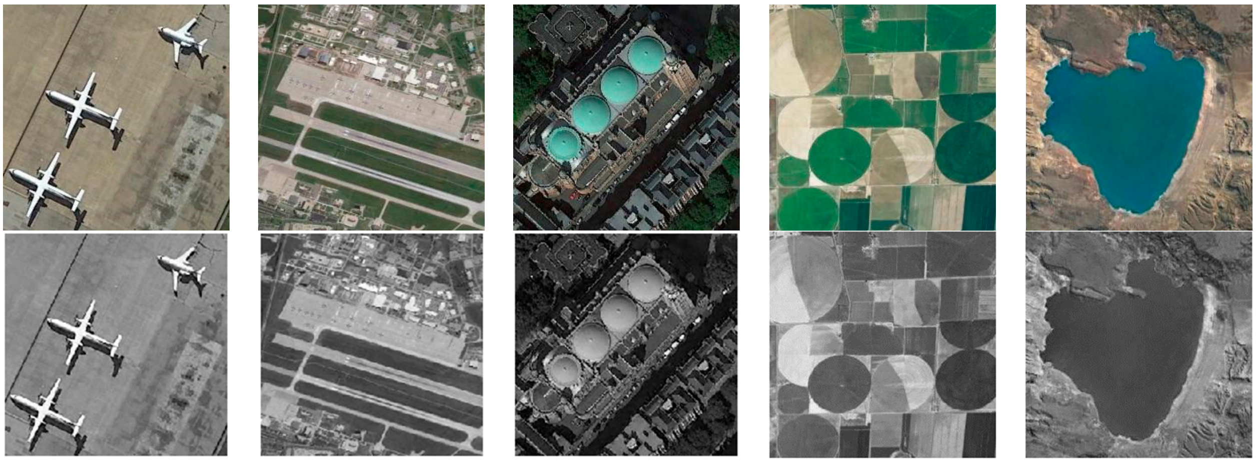
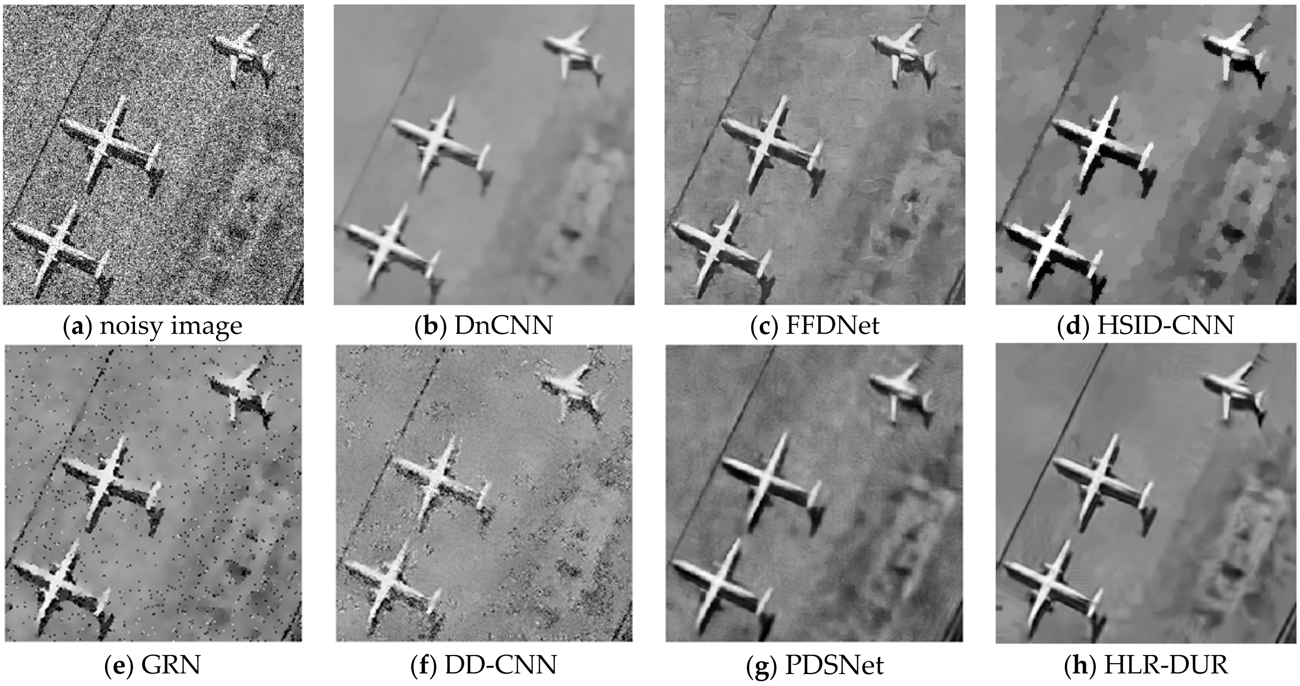


Figure 3. NWPU-RESISC45 dataset.

However, it can be seen that among all compared algorithms, HLR-DUR was the only method that clearly and completely preserved the lines in the upper left corner, indicating a significant advantage of HLR-DUR in edge preservation. Additionally, the shape and structure of the three airplanes in the denoising results of the HLR-DUR were closest to the original image. Although some texture information was missing in the denoised result of HLR-DUR, the overall edge structure was the most completely preserved. It is evident that introducing the hierarchical denoising model into the deep learning domain significantly improved the model's edge preservation ability and denoising effect.

In this section, we evaluate the denoising results of the dataset under different noise levels using MPSNR and MSSIM as evaluation metrics. These metrics were chosen because they are commonly used to assess denoising performance across entire datasets, providing a comprehensive and reliable evaluation of algorithm performance. From the data in Table 1, it is evident that DnCNN, despite its poor visual effect in Figure 4, performed well on both MPSNR and MSSIM metrics, with its MPSNR results being second only to the HLR-DUR. GRN and DD-CNN achieved competitive denoising results at noise variances of 0.01, 0.02, and 0.04, but their MPSNR and MSSIM values dropped faster than other comparative algorithms when the noise pollution level became severe, indicating that these two algorithms may not effectively remove noise under high noise conditions. Meanwhile, FFDNet and HSID-CNN showed more stable performance but did not achieve the best results in numerical evaluation. HLR-DUR achieved the highest metric values in most cases, especially in PSNR. Even with a noise variance of 0.1, the MPSNR value obtained by HLR-DUR remained higher than 30, showing a clear advantage over other algorithms. As for the MSSIM metric, HLR-DUR achieved the best values under different levels of noise, indicating HLR-DUR's superiority in structure preservation.



(**a**) noisy image (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 4. The denoising results of each comparative algorithm in Figure 3.

Table 1. Evaluation of denoising results on the NWPU-RESISC45 dataset.

Noise Variance	Evaluation Metrics	DnCNN	FFDNet	HSID-CNN	GRN	DD-CNN	PDSNet	HLR-DUR
0.01	MPSNR	33.8562	29.1558	25.1696	23.4655	24.7268	28.4567	36.8927
	MSSIM	0.7627	0.7924	0.7248	0.7085	0.7376	0.7073	0.7706
0.02	MPSNR	31.9557	27.5778	25.0029	22.9981	22.9527	25.4539	36.4328
	MSSIM	0.7250	0.7176	0.7222	0.6749	0.6778	0.7018	0.7625
0.04	MPSNR	29.7556	26.0261	24.4276	21.1774	19.5923	23.5846	35.6956
	MSSIM	0.6772	0.6445	0.7106	0.4552	0.3993	0.6936	0.7433
0.06	MPSNR	28.3637	25.1928	23.8439	17.4169	16.8556	20.3977	34.9296
	MSSIM	0.6562	0.5838	0.6651	0.2329	0.1986	0.5725	0.6706
0.08	MPSNR	27.7164	24.1979	23.1341	17.3257	14.8928	18.2633	32.9710
	MSSIM	0.6366	0.5186	0.5973	0.2312	0.1250	0.5591	0.6290
0.1	MPSNR	27.0592	23.4049	22.3332	14.5862	14.2963	18.0361	30.8986
	MSSIM	0.5575	0.4681	0.5214	0.1491	0.1099	0.4826	0.5928

4.2. Experiments on Hyperspectral Images

4.2.1. Dataset Description

The AVIRIS Indian Pines dataset was acquired in 1992 by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor over the Indian Pines test site in northwestern Indiana. The data size is 145 × 145 × 220, and the pseudo-color image is shown in Figure 5. Several bands in this dataset contain mixed noise, including impulse noise and Gaussian noise.

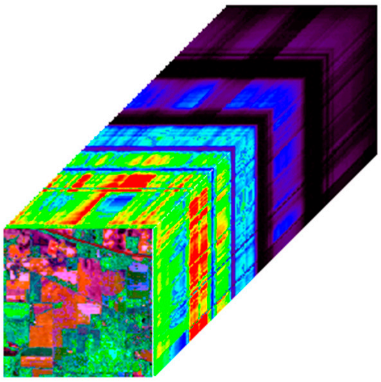
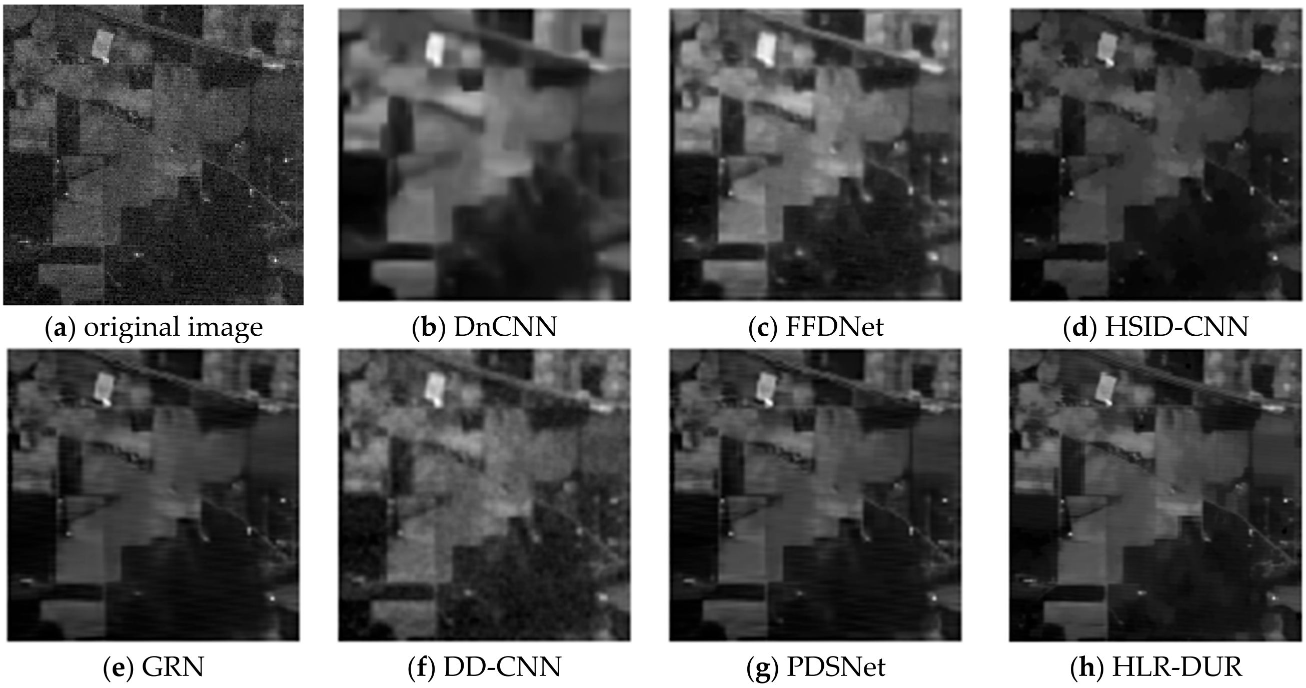


Figure 5. AVIRIS Indian Pines dataset (R: 29, G: 42, B: 89).

4.2.2. Experiments and Analysis

Firstly, Figure 6 shows the denoising results of all algorithms on the Pavia city center dataset. FFDNet, HSID-CNN, and GRN all achieved good denoising results for hyperspectral images, while DnCNN caused obvious edge blurring and DD-CNN failed to effectively remove noise. Furthermore, both PDSNet and our proposed HLR-DUR model excelled in removing noise from homogeneous regions while preserving edge information, achieving superior visual outcomes.

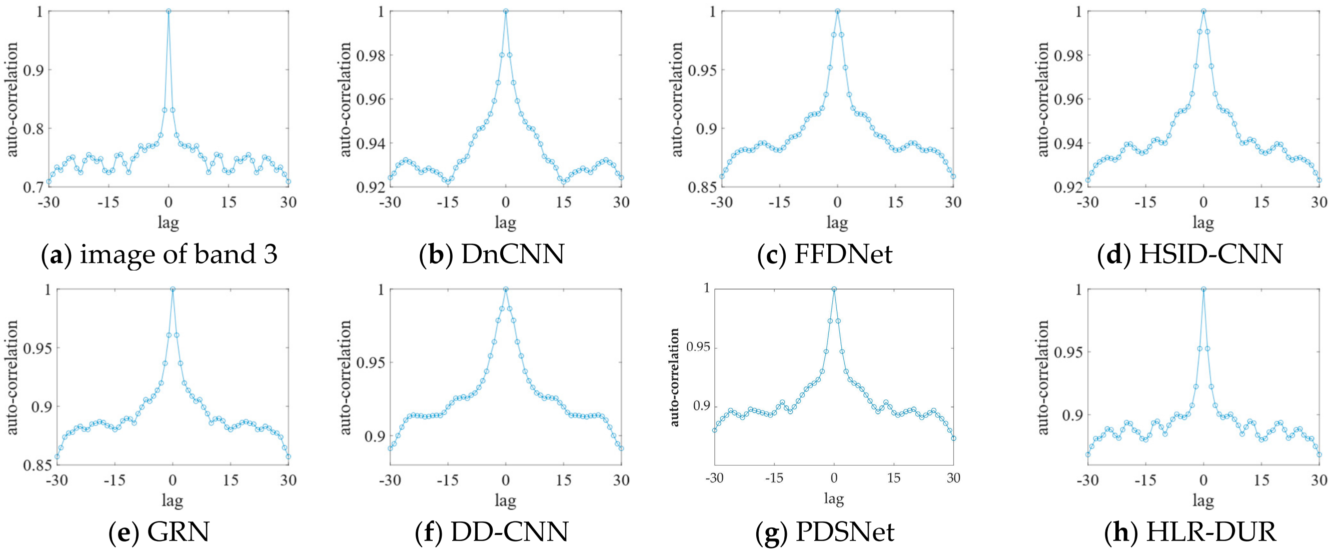


(**a**) original image (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 6. The denoising results of all algorithms on the Pavia city center dataset.

Figure 7 shows the auto-correlation curves of the denoised images. The auto-correlation curve depicts the correlation of the image at different lags, reflecting the consistency and repetitive pattern of the image texture. The closer the autocorrelation curve is to that of the original image, the better the denoising algorithm performs in preserving the image texture and structure. This proximity indicates that the algorithm can effectively remove noise while preserving edge details and texture features, which is crucial for maintaining image quality. Although the HLR-DUR model and the PDSNet model showed comparable visual effects, the results in Figure 6 clearly demonstrate that our proposed HLR-DUR model better preserves the details of the original image, as evidenced by its autocorrelation curve closely resembling that of the original image. Other algorithms showed a certain degree of broadening, which is typically the result of over-smoothing. Over-smoothing leads to the loss of unique texture details in the image, creating an unnatural visual effect, which is undesirable in image processing. The broadening observed in the comparative algorithms indicates that while noise was removed, a significant amount of texture information was also discarded. In summary, the analysis of the auto-correlation curves revealed that HLR-DUR exhibited superior ability in preserving the original texture and structure of the image.



(**a**) image of band 3 (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 7. The auto-correlation curves of the denoised images.

4.3. Experiments on SAR Images

4.3.1. Dataset Description

In the experiments of this section, we used the Virtual SAR dataset as the experimental subject. The Virtual SAR dataset is a simulated SAR image dataset, which randomly adds different levels of speckle noise based on the NWPU-RESISC45 dataset. The Virtual SAR dataset consists of 31,500 images, each with a size of 256 × 256. Each image has a corresponding noise-free version and a simulated noise-added version, as shown in Figure 8.

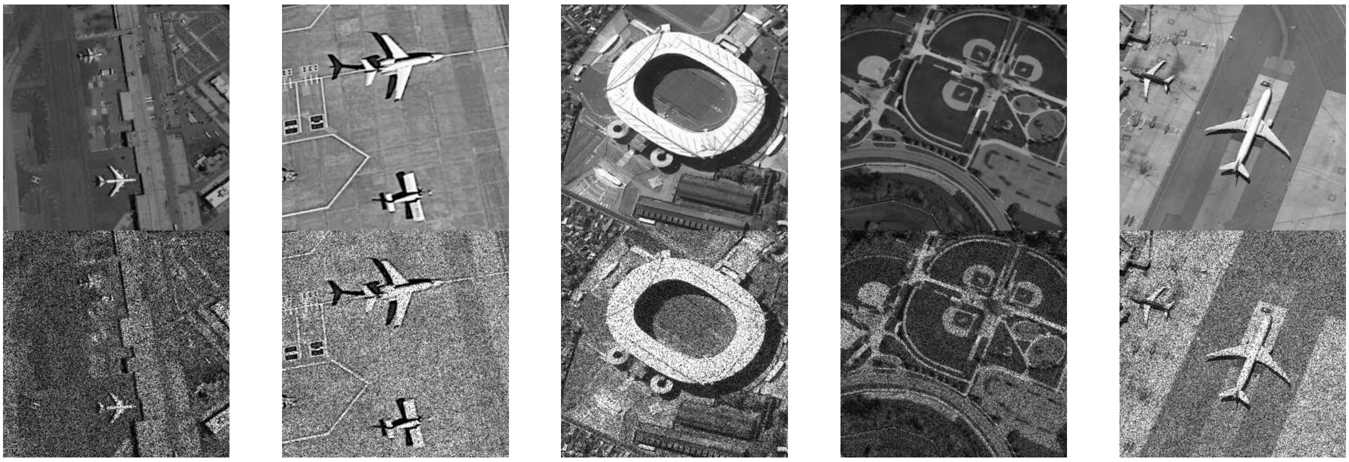
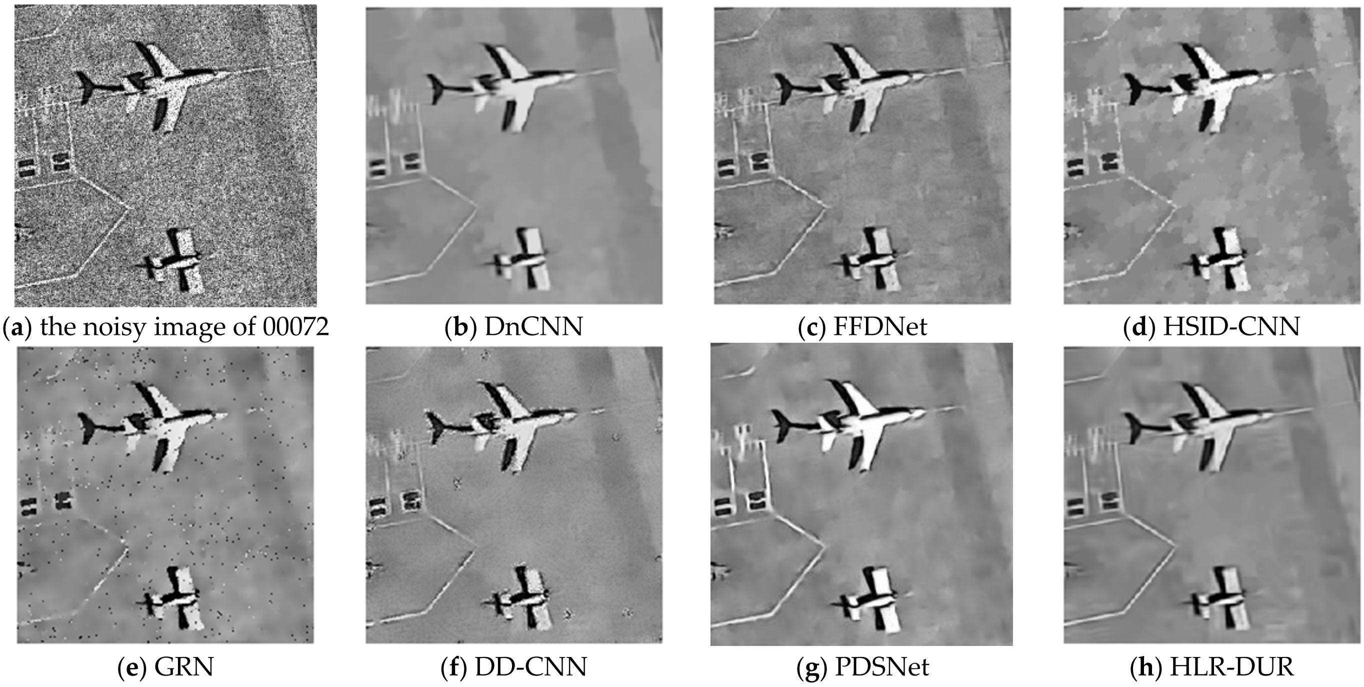


Figure 8. The samples of the Virtual SAR dataset.

4.3.2. Experiments and Analysis

As shown in Figure 9, taking data number 00072 from the Virtual SAR dataset as an example, the denoising results of various comparative algorithms are presented. Among all the methods, the denoising result obtained by the GRN algorithm still had obvious noise affecting the image quality, while the FFDNet and HSID-CNN algorithms failed to smoothly denoise the homogeneous areas, resulting in some noisy spots in the image. DnCNN, DD-CNN, and PDSNet, which are better at smoothing noise, did not adequately preserve edge information, with the runway lines in the upper left corner of these two images almost completely filtered out. In other algorithms, these lines were partially or completely preserved. Visually, HLR-DUR still achieved the best effect. It is observed that the structural outline of the airplane was clearer in the result of HLR-DUR, and the denoising effect in the homogeneous areas was also better.

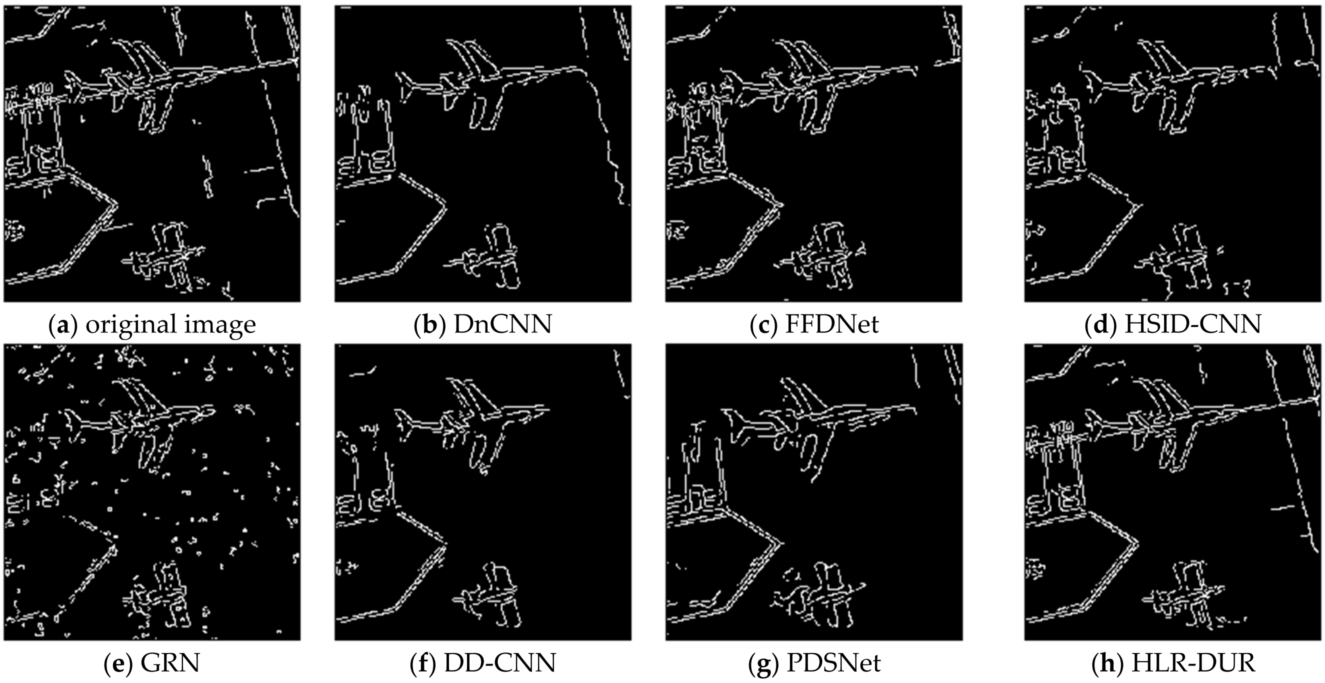

(**a**) the noisy image of 00072 (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 9. Denoised results of the Virtual SAR dataset.

To further evaluate the edge preservation capabilities of different denoising algorithms on SAR images, we used the Canny edge detection operator to extract edges from the denoised results of each comparative algorithm, as shown in Figure 10. The Canny edge detection operator is widely used in various image processing and computer vision tasks, including image segmentation, feature extraction, and pattern recognition. Although it is not the most advanced algorithm, due to its reliability, it remains one of the most popular edge detection methods in image processing. In the extracted edge results, it can be seen that due to incomplete noise removal in GRN, its edge extraction was also affected. While other comparative algorithms generally extracted clear and complete target edges, specifically the edge extracted by HLR-DUR was the most complete and closest to the original image edges.

Table 2 shows the numerical evaluation of the denoising performance of various comparative algorithms on five images from the Virtual SAR dataset. Because we were evaluating the denoising performance of SAR images, we employed ENL and EPI as evaluation metrics to assess the despeckling and edge preservation capabilities. It can be seen that on the SAR dataset, the HLR-DUR model performed well on both ENL and EPI metrics, meaning it effectively maintains the image's edges and details while reducing noise. Other algorithms such as DnCNN and DD-CNN exhibited poorer performance in edge preservation compared to FFDNet, HSID-CNN, GRN, and PSDNet. While FFDNet, HSID-CNN, GRN, and PSDNet also achieved respectable results in terms of ENL and EPI, overall, there remains a gap compared to HLR-DUR.



(**a**) original image (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

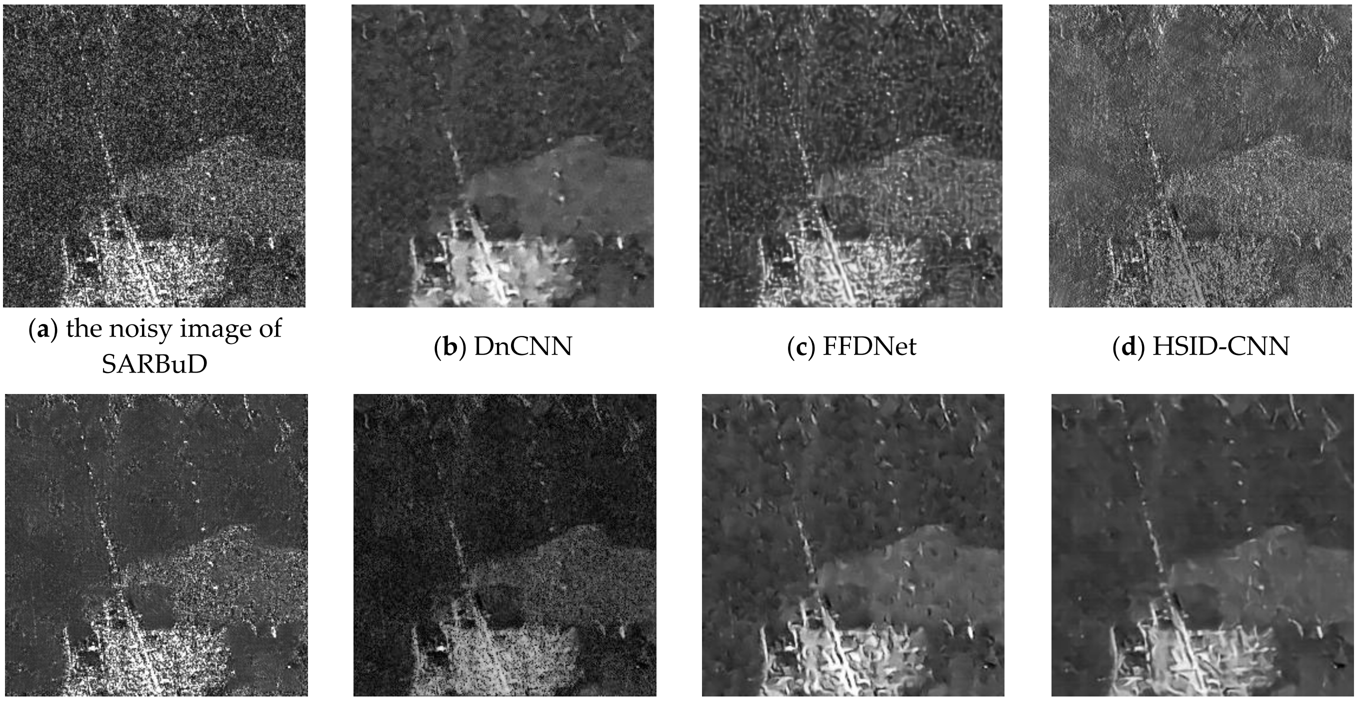
(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 10. Comparison of the edge extraction results after denoising by comparative algorithms.

Table 2. Evaluation of denoising results on the Virtual SAR dataset.

		Noisy Image	DnCNN	FFDNet	HSID-CNN	GRN	DD-CNN	PDSNet	HLR-DUR
Image 1	ENL	5.5820	103.5885	118.0177	114.4488	125.6164	98.0918	116.3475	125.9335
	EPI	-	0.5839	0.6674	0.6937	0.7022	0.5803	0.6309	0.7356
Image 2	ENL	9.3213	96.2285	95.7900	101.6122	104.6173	84.0133	107.3329	115.3813
	EPI	-	0.5236	0.5937	0.6796	0.6725	0.6116	0.5682	0.6823
Image 3	ENL	7.6195	107.8969	116.3891	113.5986	116.4556	100.3012	114.2744	124.5229
	EPI	-	0.6365	0.8779	0.8727	0.8991	0.6013	0.8917	0.9164
Image 4	ENL	7.7927	76.6310	73.9612	80.8328	80.9381	60.4646	83.6534	81.9254
	EPI	-	0.5929	0.7398	0.8634	0.8929	0.7013	0.7972	0.9147
Image 5	ENL	8.8592	103.7392	109.3430	116.1278	110.9981	101.4649	117.5947	125.6757
	EPI	-	0.7030	0.8562	0.8519	0.8244	0.7778	0.8139	0.8662

In Figure 11, we present the denoising results obtained from a real SAR dataset, SARBuD. The SARBuD dataset is a synthetic aperture radar dataset designed for benchmarking despeckling algorithms. It includes a variety of simulated SAR images with different noise levels, providing a comprehensive resource for evaluating the performance of image-denoising techniques. It can be observed that FFDNet, HSID-CNN, and GRN algorithms exhibit suboptimal performance with residual noise remaining in uniform regions. While DnCNN, DD-CNN, and PDSNet demonstrate relatively effective denoising, they tend to blur or even filter out image edges. HLR-DUR achieved the best visual results among the methods evaluated.



(**a**) the noisy image of SARBuD (**b**) DnCNN (**c**) FFDNet (**d**) HSID-CNN

(**e**) GRN (**f**) DD-CNN (**g**) PDSNet (**h**) HLR-DUR

Figure 11. Denoised results of the SARBuD dataset.

As shown in Table 3, it is evident that the HLR-DUR model consistently outperformed other algorithms when faced with real SAR image datasets. This indicates that it achieves superior evaluation metrics in most scenarios, suggesting its capability to effectively balance denoising and edge preservation in real SAR images.

Table 3. Evaluation of denoising results on the SARBuD dataset.

		Noisy Image	**DnCNN**	**FFDNet**	**HSID-CNN**	**GRN**	**DD-CNN**	**PDSNet**	**HLR-DUR**
Image 1	ENL	10.7727	67.4415	55.9390	54.1302	57.7159	58.7127	65.9892	72.2878
	EPI	-	0.4201	0.5413	0.5265	0.5159	0.5337	0.5712	0.6175
Image 2	ENL	12.6964	62.5110	67.8803	63. 4276	63.5245	69.7125	77. 0636	92.3050
	EPI	-	0.6521	0.7019	0.7060	0.8226	0.6715	0.8064	0.8626
Image 3	ENL	10.6408	56.4970	63.1859	70.1377	66.9889	54.0735	82.4010	78.4357
	EPI	-	0.6361	0.71845	0.6423	0.7223	0.6531	0.7312	0.7907

The HLR-DUR model showed significant advantages through many evaluation metrics. By conducting a comprehensive analysis of the denoising results for virtual SAR images and real SAR images, it can be seen that compared to other commonly used denoising algorithms such as DnCNN, FFDNet, HSID-CNN, GRN, DD-CNN, and PDSNet, HLR-DUR more effectively preserves the image's edge parts and detail information while reducing noise levels. Especially in dealing with complex and diverse image denoising problems, HLR-DUR demonstrated its comprehensive advantages, achieving the goal of edge preservation and denoising, and holds great potential in the field of image denoising.

4.4. Processing Speed and Ablation Experiments

4.4.1. Ablation Experiment and Accuracy and Loss Curves

Figure 12 illustrates the training and validation loss results of HLR-DUR, demonstrating its rapid convergence and minimal loss values, indicating effective denoising performance.

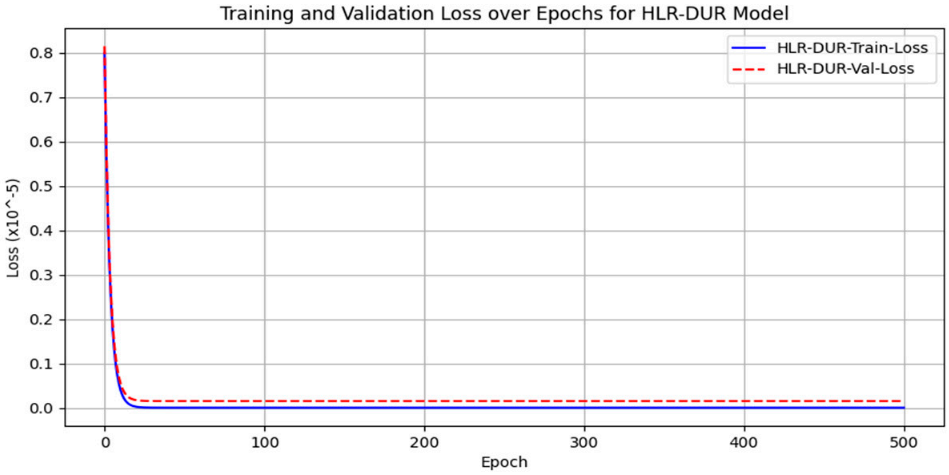


Figure 12. Training accuracy and loss curves.

We conducted ablation experiments on the deep model. The primary objective of these experiments was to assess the impact of the edge modules incorporated within the model. The results presented in Table 4 unequivocally demonstrate that integrating the edge module significantly improved both the denoising efficacy and the model's ability to preserve edges. This outcome robustly validates the effectiveness of the edge module in enhancing overall model performance.

Table 4. Results of the ablation experiments.

	Evaluation Metrics	**0.01**	**0.02**	**0.04**	**0.06**	**0.08**	**0.1**
Without Edge Module	MPSNR	35.3927	35.0233	34.6201	33.2930	31.9223	29.1146
	MSSIM	0.6111	0.6032	0.5725	0.5549	0.7022	0.5803
HLR-DUR	MPNSR	36.8927	36.4328	35.6956	34.9296	32.9710	30.8986
	MSSIM	0.7023	0.6882	0.6620	0.6321	0.6290	0.6116

4.4.2. Results of the Processing Speed and Parameters Experiment

We employed four NVIDIA® GeForce RTX 3090 GPUs which are manufactured by NVIDIA Corporation, a technology company based in Santa Clara, California, United States.to evaluate the processing speed and parameter count of training models using 256 × 256 SAR images as input. Table 5 presents the parameter sizes and processing details of seven models. Among them, the model with the largest parameter count was the enhanced DD-CNN, which totaled 7,955,552 parameters. The model requiring the longest computational time was PDSNet, with an approximate execution time of 76.26 ms. Our proposed HLR-DUR model had 28,311 parameters and took about 1.2 ms per epoch to compute.

Table 5. Different model training parameters and time consumption comparison.

Methods	**DnCNN**	**FFDNet**	**HSID-CNN**	**GRN**	**DD-CNN**	**PDSNet**	**HLR-DUR**
Time(ms)	22.51	50.13	47.13	67.78	63.16	76.26	1.2
Parameters	667,008	485,316	556,097	1,322,251	7,955,552	1,366,111	28,311

5. Conclusions

In this paper, a new deep low-rank model for remote sensing image denoising named HLR-DUR was proposed. Compared with existing denoising methods, HLR-DUR combines the advantages of the traditional LRR model and deep learning, enabling it to extract edge information using prior knowledge regulations while further enhancing the model's denoising performance and computational efficiency through the deep autoencoder. HLR-DUR combines the shallow low-rank denoising model with an autoencoder to construct a two-level deep encoder. It uses deep unfolding to extract both the low-rank and edge components. By integrating traditional iterative algorithms with the deep learning framework, the iterative steps are unfolded into multiple layers, which are then aggregated to obtain the final denoised result. Experiments conducted on three types of remote sensing datasets, optical remote sensing images, hyperspectral images, and SAR images, demonstrated that HLR-DUR achieves better edge-preserving denoising results than SOTA models and significantly improves the denoising effect.

Author Contributions: Conceptualization, F.S. and T.Z.; methodology, F.S. and S.T.; software, X.F. and T.Z.; Data collection, X.F.; validation, F.S. and X.F.; investigation, S.T.; writing—original draft preparation, T.Z.; writing—review and editing, F.S. and S.T.; supervision, F.S. and S.T.; project administration, F.S.; funding acquisition, F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available in [IEEEDataport and Puedue University Research Repository] at [https://ieee-dataport.org/open-access/virtual-sar-synthetic-dataset-deep-learning-based-speckle-noise-reduction-algorithms and https://purr.purdue.edu/publications/1947/1].

Acknowledgments: The authors would like to thank the Editor and the anonymous Referees for their instructive comments.

Conflicts of Interest: Author Fanqi Shao was employed by the company China Electronics Technology Group Corporation. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Sabins, F.F., Jr.; Ellis, J.M. *Remote Sensing: Principles, Interpretation, and Applications*; Waveland Press: Long Grove, IL, USA, 2020.
2. Rasti, B.; Chang, Y.; Dalsasso, E.; Denis, L.; Ghamisi, P. Image restoration for remote sensing: Overview and toolbox. *IEEE Geosci. Remote Sens. Mag.* **2021**, *10*, 201–230. [CrossRef]
3. Li, G.; Gu, X.; Chen, C.; Zhou, C.; Xiao, D.; Wan, W.; Cai, H. Low-Frequency Magnetotelluric Data Denoising Using Improved Denoising Convolutional Neural Network and Gated Recurrent Unit. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5909216. [CrossRef]
4. Li, G.; Wu, S.; Cai, H.; He, Z.; Liu, X.; Zhou, C.; Tang, J. IncepTCN: A new deep temporal convolutional network combined with dictionary learning for strong cultural noise elimination of controlled-source electromagnetic data. *Geophysics* **2023**, *88*, E107–E122. [CrossRef]
5. Han, L.; Zhao, Y.; Lv, H.; Zhang, Y.; Liu, H.; Bi, G. Remote sensing image denoising based on deep and shallow feature fusion and attention mechanism. *Remote Sens.* **2022**, *14*, 1243. [CrossRef]
6. Huang, Z.; Zhu, Z.; Wang, Z.; Shi, Y.; Fang, H.; Zhang, Y. DGDNet: Deep gradient descent network for remotely sensed image denoising. *IEEE Geosci. Remote Sens. Lett.* **2023**, *20*, 6002405. [CrossRef]
7. Tian, C.; Fei, L.; Zheng, W.; Xu, Y.; Zuo, W.; Lin, C.-W. Deep learning on image denoising: An overview. *Neural Netw.* **2020**, *131*, 251–275. [CrossRef] [PubMed]
8. Jiao, L.; Zhao, J. A survey on the new generation of deep learning in image processing. *IEEE Access* **2019**, *7*, 172231–172263. [CrossRef]
9. Feng, X.; Zhang, W.; Su, X.; Xu, Z. Optical remote sensing image denoising and super-resolution reconstructing using optimized generative network in wavelet transform domain. *Remote Sens.* **2021**, *13*, 1858. [CrossRef]
10. Huang, Z.; Zhang, Y.; Li, Q.; Li, Z.; Zhang, T.; Sang, N.; Xiong, S. Unidirectional variation and deep CNN denoiser priors for simultaneously destriping and denoising optical remote sensing images. *Int. J. Remote Sens.* **2019**, *40*, 5737–5748. [CrossRef]
11. Xie, Y.; Qu, Y.; Tao, D.; Wu, W.; Yuan, Q.; Zhang, W. Hyperspectral image restoration via iteratively regularized weighted schatten p-norm minimization. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 4642–4659. [CrossRef]

12. 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*, 635–648. [CrossRef]
13. Gu, F.; Zhang, H.; Wang, C. A two-component deep learning network for SAR image denoising. *IEEE Access* **2020**, *8*, 17792–17803. [CrossRef]
14. Feng, X.; Tian, S.; Abhadiomhen, S.E.; Xu, Z.; Shen, X.; Wang, J.; Zhang, X.; Gao, W.; Zhang, H.; Wang, C. Edge-Preserved Low-Rank Representation via Multi-Level Knowledge Incorporation for Remote Sensing Image Denoising. *Remote Sens.* **2023**, *15*, 2318. [CrossRef]
15. 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. [CrossRef]
16. He, W.; Zhang, H.; Zhang, L.; Shen, H. Hyperspectral image denoising via noise-adjusted iterative low-rank matrix approximation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 3050–3061. [CrossRef]
17. Ma, T.-H.; Xu, Z.; Meng, D. Remote sensing image denoising via low-rank tensor approximation and robust noise modeling. *Remote Sens.* **2020**, *12*, 1278. [CrossRef]
18. Zhang, H.; Liu, L.; He, W.; Zhang, L. Hyperspectral image denoising with total variation regularization and nonlocal low-rank tensor decomposition. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 3071–3084. [CrossRef]
19. Li, C.; Ma, Y.; Huang, J.; Mei, X.; Ma, J. Hyperspectral image denoising using the robust low-rank tensor recovery. *J. Opt. Soc. Am. A* **2015**, *32*, 1604–1612. [CrossRef] [PubMed]
20. Guan, D.; Xiang, D.; Tang, X.; Kuang, G. SAR image despeckling based on nonlocal low-rank regularization. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 3472–3489. [CrossRef]
21. Chen, G.; Li, G.; Liu, Y.; Zhang, X.-P.; Zhang, L. SAR image despeckling based on combination of fractional-order total variation and nonlocal low rank regularization. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 2056–2070. [CrossRef]
22. Geng, J.; Fan, J.; Ma, X.; Wang, H.; Cao, K. An iterative low-rank representation for SAR image despeckling. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 72–75.
23. Baier, G.; He, W.; Yokoya, N. Robust nonlocal low-rank SAR time series despeckling considering speckle correlation by total variation regularization. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 7942–7954. [CrossRef]
24. Zhang, H.; Chen, H.; Yang, G.; Zhang, L. LR-Net: Low-rank spatial-spectral network for hyperspectral image denoising. *IEEE Trans. Image Process.* **2021**, *30*, 8743–8758. [CrossRef]
25. Nguyen, H.V.; Ulfarsson, M.O.; Sigurdsson, J.; Sveinsson, J.R. Deep sparse and low-rank prior for hyperspectral image denoising. In Proceedings of the IGARSS 2022–2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 17–22 July 2022; pp. 1217–1220.
26. Sun, H.; Liu, M.; Zheng, K.; Yang, D.; Li, J.; Gao, L. Hyperspectral image denoising via low-rank representation and CNN denoiser. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *15*, 716–728. [CrossRef]
27. Zhang, Q.; Yuan, Q.; Song, M.; Yu, H.; Zhang, L. Cooperated spectral low-rankness prior and deep spatial prior for HSI unsupervised denoising. *IEEE Trans. Image Process.* **2022**, *31*, 6356–6368. [CrossRef]
28. Xiong, K.; Zhao, G.; Wang, Y.; Shi, G. SAR imaging and despeckling based on sparse, low-rank, and deep CNN priors. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 4501205. [CrossRef]
29. Monga, V.; Li, Y.; Eldar, Y.C. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Process. Mag.* **2021**, *38*, 18–44. [CrossRef]
30. Passah, A.; Amitab, K.; Kandar, D. SAR image despeckling using deep CNN. *IET Image Process.* **2021**, *15*, 1285–1297. [CrossRef]
31. Wang, S.; Wang, X.; Zhang, L.; Zhong, Y. Deep low-rank prior for hyperspectral anomaly detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5527017. [CrossRef]
32. Candès, E.J.; Li, X.; Ma, Y.; Wright, J. Robust principal component analysis? *J. ACM* **2011**, *58*, 11. [CrossRef]
33. Dong, Y.; Chan, R.H.; Xu, S. A detection statistic for random-valued impulse noise. *IEEE Trans. Image Process.* **2007**, *16*, 1112–1120. [CrossRef]
34. Lopes, A.; Touzi, R.; Nezry, E. Adaptive speckle filters and scene heterogeneity. *IEEE Trans. Geosci. Remote Sens.* **1990**, *28*, 992–1000. [CrossRef]
35. Hong, M.; Luo, Z.-Q. On the linear convergence of the alternating direction method of multipliers. *Math. Program.* **2017**, *162*, 165–199. [CrossRef]
36. Huang, Z.; Wang, Z.; Zhu, Z.; Zhang, Y.; Fang, H.; Shi, Y.; Zhang, T. DLRP: Learning deep low-rank prior for remotely sensed image denoising. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 6508905. [CrossRef]
37. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622. [CrossRef] [PubMed]
38. 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*, 1205–1218. [CrossRef]
39. 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*, 5504714. [CrossRef]

40. Shan, W.; Liu, P.; Mu, L.; Cao, C.; He, G. Hyperspectral image denoising with dual deep CNN. *IEEE Access* **2019**, *7*, 171297–171312. [CrossRef]
41. Lin, C.; Qiu, C.; Jiang, H.; Zou, L. A Deep Neural Network Based on Prior-Driven and Structural Preserving for SAR Image Despeckling. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 6372–6392. [CrossRef]

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.