Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 33(9): 5794–5828.
DOI: 10.3934/era.2025258
Received: 03 August 2025
Revised: 14 September 2025
Accepted: 19 September 2025
Published: 26 September 2025

Research article

Multi-branch network for brain image denoising based on U-shaped dynamic convolution and multi-scale feature extraction

Huimin Qu, Haiyan Xie* and Qianying Wang

School of Science, Dalian Maritime University, Dalian 116026, China

* **Correspondence:** Email: xiehy@dlmu.edu.cn.

Abstract: Brain images are often disturbed by noise during acquisition, compromising subsequent medical analysis and reducing diagnostic accuracy. Effective denoising that preserves the structural details is therefore essential. This essay proposes a multi-branch convolutional neural network for brain image denoising, integrating a U-shaped network, dynamic convolution, and multi-scale feature extraction. The model includes an attention-enhanced encoder–decoder to improve feature representation, dynamic convolution with a sparse mechanism for adaptive global modeling, multi-scale dense residual blocks with depth-separable convolution to capture local details efficiently, and a multi-branch fusion strategy to process features at different scales in parallel and refine them. Experiments on brain image datasets demonstrate that the proposed method achieves superior denoising performance, effectively suppressing noise while retaining fine structural details. In conclusion, the network significantly enhances the quality of brain images and shows potential for improving the accuracy of subsequent medical image analysis and diagnosis.

Keywords: brain image denoising; U-Net; dynamic convolution kernel; multi-scale feature extraction; multi-branch network

1. Introduction

As a core component of the body, the brain is enclosed in the skull due to its complex structure, which increases the difficulty of studying its function and complicates the diagnosis of brain diseases [1]. Medical imaging techniques provide noninvasive means for disease diagnosis and scientific research, but random noise degrades image quality and affects diagnostic accuracy [2]. To enhance the images' usability, effective noise reduction methods must be used to reduce interference, enhance key details, clarify the region of interest, and improve image analysis and diagnostic accuracy [3]. Positron emission tomography (PET) is a molecular imaging technique that reflects biological tissues' properties by detecting the three-dimensional distribution of a radiotracer [4]. PET images are more susceptible to

noise than imaging techniques such as computed tomography (CT), which reduces quantitative accuracy and compromises lesion detection. Therefore, developing effective noise reduction techniques has become important in PET imaging research [5]. In this paper, we focus on the noise characteristics of PET image datasets and investigate efficient denoising methods to reduce noise while retaining key metabolic and structural information to improve imaging quality and quantitative accuracy.

The main task of image denoising is to filter out the noise when using various filtering techniques while protecting critical information, such as image details [6]. Filtering can be used in the transform domain, e.g., curvilinear threshold filtering [7] and wavelet threshold filtering [8]. It can also be performed in the time domain, for example, the nonlocal means (NLM) [9], the Gaussian filter [10], the bilateral filter [11], the Wiener filter [12], and the total variation (TV) filter [13]. However, traditional PET image denoising tends to result in blurred edges, loss of details, or high computational complexity, challenging the balance between noise reduction and information retention.

With the continuous increase in research data and the improvement in computational power, deep learning-based denoising methods have gradually become a hot research topic [14]. Deep learning methods effectively overcome the limitations of traditional techniques with superior representation capabilities. The development of deep learning has driven the evolution of neural networks from shallow to deep architectures, enhanced noise reduction in PET images, and experimentally verified their superior performance [15]. Zhang et al. [16] proposed feedforward denoising convolutional neural networks (DnCNNs), which improve the image denoising performance and accelerate the training process by employing a deep network architecture, residual learning, and batch normalization in combination with new learning algorithms and regularization methods. Tian et al. [17] proposed an enhanced convolutional neural denoising network (ECNDNet) to speed up training and improve network convergence. Using a deep learning model to optimize the quality of low-dose PET images and comparing them with standard images, the results indicated that this method allowed reduced-dose fluorodeoxyglucose positron emission tomography (FDG PET) images to meet quality of care standards [18].

In recent years, the U-shaped network (U-Net) structure has been effective in medical image denoising and segmentation, and its encoder–decoder structure can effectively capture an image's global and local features [19]. Hong et al. [20] developed a data-driven single-image super-resolution sinogram method based on a deep residual convolutional neural network for improving the image resolution and noise characteristics of a large-pixelized crystal PET scanner. DeepPET, proposed by Häggström et al. [21] as a novel end-to-end PET image reconstruction technique based on a deep convolutional encoder–decoder network, enables direct and fast output of high-quality quantitative PET images by the PET sinusoidal map data input. A typical U-Net consists of contraction and symmetric expansion paths for extracting contextual information and upsampling reconstruction features. Its structure introduces jump connections to pass features from the contraction path to the expansion path and incorporates residual learning to optimize feature representation [22]. Based on the assumption that extracting potential noise components is easier to achieve than preserving the complex visual features of PET images in the hidden layer, this method aims to guide the network to output the noise information contained in the image [23]. Although these articles introduced a U-shaped architecture, they did not incorporate adaptive tuning of convolution kernels based on image content, which might limit their capability to extract and represent complex image features. Adapting the convolutional kernel may improve the model's adaptability and denoising effect.

In convolutional neural networks (CNNs), convolution is crucial in information extraction, where

features are extracted by stacking multiple convolutional layers and using different convolutional kernels. Standard convolution performs well in extracting local features, but two major defects exist. One is that the convolution kernel is fixed and independent of the input data, resulting in suboptimal feature extraction under complex data relationships. The second is that although local features can be extracted, it is challenging to capture global information, and stacking multiple layers of convolution increases the computational demand and optimization difficulty [24]. Lei et al. [25] proposed an adversarial self-ensembling network using dynamic convolution (ASE-Net). This dynamic convolution-based semi-supervised medical image segmentation network efficiently utilizes labeled and unlabeled data through coherent learning and dual discriminator networks while introducing a bidirectional attention component to avoid overfitting. Zheng et al. [24] designed a global and local attention unit (GLAU) module in which the weighted fusion of the global channel attention kernels and the local spatial attention kernels generated dynamic convolutional kernels. Soloviev et al. [26] proposed a CNN architecture based on dynamic convolution for the image matching problem by developing a dual-branch network in which one branch generates a convolutional kernel. Experiments show that the model outperforms the standard convolutional baseline model regarding learning speed and performance. Yun et al. [27] proposed a dynamic residual convolution (DRConv), which can effectively extract the local features related to inputs and solve the problems of limited representation capability and the difficulty of joint optimization of the kernel attention module. DRConv obtains globally salient features by labeling the attention as an input to kernel attention, improving the model's representation capability and selection ability. Although the articles above considered the feature of adaptively adjusting the convolutional weights according to the inputs and the mechanism of reducing redundant computation to enhance the generalization ability, they did not sufficiently consider the capacity for comprehensive extraction of features at different scales.

The traditional convolutional layer extracts features at a fixed scale, making it challenging to handle image information at different scales effectively. In recent years, using a multi-scale feature extraction module has become an effective method to improve a model's performance. The model can capture image details more comprehensively by fusing multi-scale features. Lee et al. [28] implemented multi-scale feature learning using CNNs with different input sizes to capture local audio features. In addition, features were extracted and aggregated from each layer of the pre-trained network to integrate multi-scale information for label prediction. Liu et al. [29] proposed a deep network for the fusion of infrared and visible images, which learned multi-scale features of multimodal images by cascading feature learning modules to discover common structures, guide the fusion process, and enhance the fusion effect through an edge-directed attention mechanism that attenuated noise and recovered details. Huo et al. [30] proposed a hierarchical multi-scale feature fusion network (HiFuse), a multi-scale three-branch hierarchical feature fusion network for improving medical image classification accuracy. Its key features are global and local feature blocks with a parallel hierarchical structure, an adaptive hierarchical feature fusion block (HFF block), and an inverted residual multi-layer perceptron (IRMLP). Despite the advantages of these articles in multi-scale feature extraction, there is still room for improvement in terms of adaptive convolutional weight adjustment and information flow optimization.

In general, different network architectures can extract complementary features, which helps to improve the image denoising effect [31]. Increasing the network's width is an effective strategy to improve the denoising performance. Tian et al. [32] proposed a novel dual-branching network architecture consisting of two different sub-networks, the upper network and the lower network, to use the diverse features to optimize the denoising effect fully. The dual denoising network (DudeNet) [31]

effectively improves the denoising performance and detail reconstruction of images by introducing a dual-channel sparse mechanism and fusing global and local features in the network. Yang et al. [33] proposed a dual-channel residual convolutional neural network (UDRN) for underwater image denoising, which introduced residual blocks into the dual-channel image denoising structure, extracted shallow spatial features, and improved dual-channel denoising performance. The brain dual-channel attentional residual network (BDARN) [34] is a dual-channel convolutional neural network model designed for brain image denoising. Wang et al. [35] proposed a novel dual-branch network (DuINet) specifically designed to capture complementary aspects of image information, which incorporates information exchange and perceptual loss. Inspired by dual-branch networks, this paper explores multi-branch networks whose parallel structure can efficiently extract multi-scale features and optimize feature fusion. Qu et al. [36] proposed a multi-CNN (MFDRAN) that incorporates feature distillation learning and dense residual attention for brain image denoising. Multi-branch networks utilize information from different scales and semantic levels to improve the model's expressive ability and denoising performance compared with dual-branch networks.

In summary, significant progress has been made in PET image denoising research within academic journals. Traditional CNN methods can effectively suppress Gaussian noise, but often lead to edge blur and loss of detail [17]. To address this issue, some studies have employed deeper network architectures to enhance the feature extraction capacity. However, these approaches substantially increase the computational complexity, which limits their efficiency in practical medical applications [21]. Dynamic convolutional networks have been explored to improve adaptability across different noise distributions, yet they still struggle to capture global semantic information and preserve fine-grained local structures simultaneously. Attention-based U-shaped networks have demonstrated strong global feature modeling capabilities [27]. Still, an over-reliance on attention tends to impair the capture of local details, thereby hindering the detection of minute lesions. Conversely, excessive dependence on local representations may also overlook broader contextual relationships.

Inspired by the abovementioned studies and multi-branch architectures, this paper proposes a novel multi-branch convolutional neural network incorporating key techniques such as a U-shaped [37] structure based on an attention mechanism [38], a dynamic convolutional kernel, a sparse mechanism, multi-scale feature extraction, and depth-separable convolution to enhance the denoising effect of brain images. The structure based on the attention mechanism can effectively fuse low-level and high-level features to enhance the denoising effect. The dynamic convolution kernel can adaptively adjust the filter according to the features of the input image to improve the model's ability to adapt to different noise patterns. Multi-scale feature extraction helps to capture noise features and detailed information of the image at various scales. Deep separable convolution reduces the computational complexity and improves the model's efficiency. Through the synergy of these strategies, the proposed model improves the denoising performance and provides an effective solution for brain image denoising tasks.

For this paper, the main contributions are as follows.

(1) This paper proposes a multi-branch convolutional neural network with an attention-based encoder–decoder, multi-scale feature extraction, and parallel dynamic convolutional kernels. These branches are adaptively fused to model global and local features, capturing fine-grained details while overcoming the limitations of single-attention and purely dynamic convolutional methods.

(2) The first branch incorporates an attention mechanism within the U-shaped structure to enhance encoder–decoder information transfer and cross-layer feature fusion, bridging the gap between high-

level semantics and low-level details. Adaptively weighting features on the basis of regional importance improves the stability and robustness of denoising under complex noise conditions.

(3) The second branch combines dynamic convolution with a sparsity mechanism to enhance high-frequency feature extraction in brain images. Dynamic convolution adapts the kernels to varying noise distributions. At the same time, the sparsity mechanism suppresses redundancy and emphasizes edges, textures, and subtle lesions, improving robustness and preserving anatomical details under complex imaging conditions.

(4) The third branch employs a multi-scale feature extraction module with dense residual blocks and parallel convolutional kernels to capture noise patterns and structural details in different receptive fields. Using depth-separable convolution reduces the computational complexity while preserving fine details and enhancing denoising performance.

(5) The proposed multi-branch network effectively denoises brain images, demonstrating superior noise suppression and structure preservation compared with existing methods, as confirmed by quantitative metrics and visual evaluations.

The rest of the paper is organized as follows. Section 2 briefly describes the related work. In Section 3, the proposed multi-branch CNN based on U-shaped dynamic convolutional multi-scale feature extraction is described in detail. Section 4 shows the experimental results and analysis, and compares the performance of the proposed network with other methods. Finally, the conclusions are given in Section 5.

2. Basic theory

Brain denoising aims to recover the original brain image y from the corresponding noisy image x. The general idea is to remove or reduce the noise level of the noisy image x to obtain a high-quality estimate of y [39].

2.1. Encoder–decoder

U-Net consists of a contraction path (encoder) and an expansion path (decoder). The contraction path is similar to a regular convolutional network, which gradually reduces the spatial resolution of the feature map through multi-layer convolution and pooling operations while increasing the number of channels to extract multi-scale, high-level feature representations. U-Net skip-connects the up-sampled feature map with the features of the corresponding layer of the encoder in the expansion path and adds a convolutional layer and a nonlinear activation function after each upsampling step to gradually recover the spatial information and improve the segmentation accuracy [40]. Due to the symmetry of the architecture, the U-net presents a typical U-shaped structure, which significantly enhances the efficiency and accuracy of image segmentation [41].

One of the core features of U-Net is skip connection, which fuses the feature maps of the corresponding layers of the encoder with the feature maps in the decoder at the decoding stage. This design preserves the low-level semantic information and combines the high-level semantic information, which improves the segmentation accuracy and enhances detail recovery. In the encoder–decoder subnetwork, the transform convolution unit extracts shallow features from the image and generates a 64-feature map. Using a four-scale U-net structure, the network learned the contextual information of the image in the encoder subnetwork. It recovered the multi-level features of the image through the decoder

subnetwork [35].

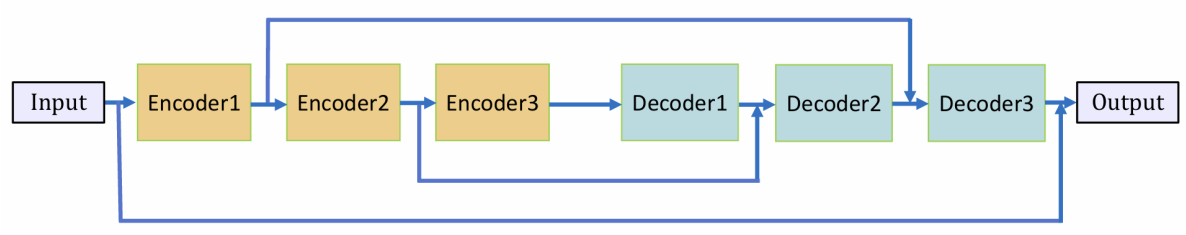

Figure 1. Flow of the encoder–decoder structure.

Figure 1 presents the detailed flow of the Encoder–decoder architecture. Three encoders process the input (x_1) to extract the feature maps x_2, x_3, and x_4. These features are progressively upsampled by three decoders with skip connections (Concat) from the corresponding encoder outputs, yielding the final output (y). The specific computational process of the encoders and decoders is detailed as follows.

$$x_2 = \text{Encoder1}(x_1), \quad x_3 = \text{Encoder2}(x_2), \quad x_4 = \text{Encoder3}(x_3), \tag{2.1}$$

$$\text{y} = \text{Concat}\Big(\text{Decoder3}(\text{Concat}(\text{Decoder2}(\text{Concat}(\text{Decoder1}(x_4), x_3), x_2)), x_1)\Big). \tag{2.2}$$

2.2. *Dynamic convolution kernel*

Most existing denoising convolutional neural networks are trained by sharing the parameters of convolutional layers. Even so, the failure to adaptively adjust the parameters according to different images leads to a lack of robustness in classification performance. To solve this problem, fusing parallel convolution with an attention mechanism and introducing dynamic convolution to optimize the convolution operation are proposed. Unlike the traditional fixed convolution kernel, dynamic convolution can adaptively adjust the weights of the convolution kernel according to the features of the input image, thus improving the adaptability of the model under different tasks or noise patterns and effectively reducing the computational cost to train more robust classifiers [42]. Tian et al. [43] proposed a multi-stage image denoising CNN with the wavelet transform through three stages: A dynamic convolutional block (DCB), two cascaded wavelet transform and enhancement blocks (WEBs), and a residual block (RB). The DCB uses dynamic convolution to dynamically tune the parameters of multiple convolutions to make a trade-off between noise reduction performance and computational cost. Cheng et al. [44] proposed a model called the encoder–decoder dynamic graph convolutional network (ED-DGCN) for multi-label classification of electrocardiographic (ECG) signals, which extracts features using an encoder, generates content-aware classification representations through a decoder, and models the labels' relationships with an adaptive dynamic graph convolution network to enhance inter-label correlations.

In traditional convolution operations, the convolution kernel parameters are fixed and do not vary with the input. Dynamic convolution introduces an adaptively tunable convolution kernel that adjusts the computation according to the input features. Figure 2 illustrates the complete process of dynamic convolution, including the input features, weight generation, dynamic convolution kernel generation, and the output features. Specifically, instead of using a single fixed convolutional kernel for each convolutional layer, a predefined set of N convolutional kernels $conv_i(i = 1 \cdots n)$ with the same scale and number of channels is used as the candidate set. On the basis of these weights, convolutional kernels

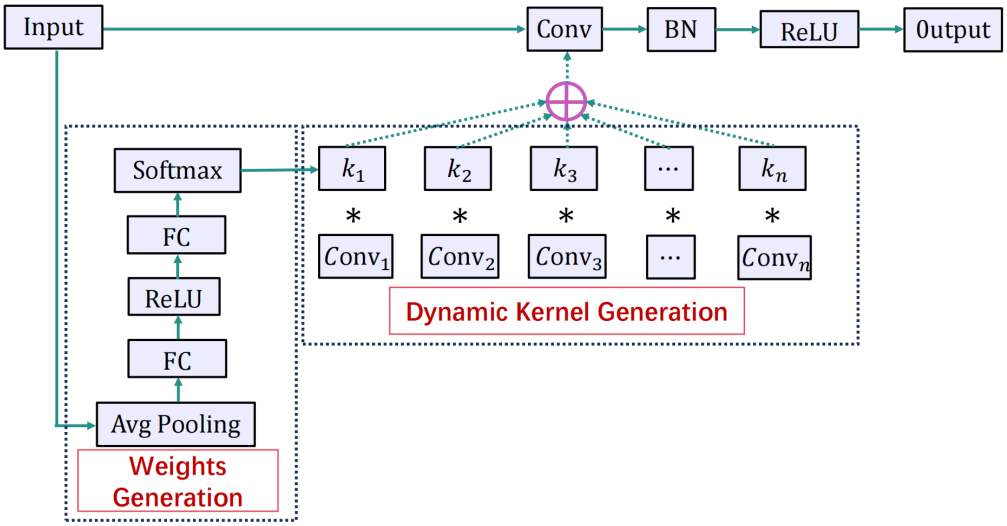

Figure 2. Dynamic convolution flowchart.

are weighted and summed to obtain the final convolutional kernel corresponding to the current input. The final convolutional kernel $W_{dyn}(x)$ representation takes the form,

$$W_{dyn}(x) = \sum_{i=1}^{n} conv_i k_i(x), \tag{2.3}$$

where the weight k_i $(i = 1, \cdots, n)$ is computed as follows: first, the global spatial features are extracted by global average pooling (GAP) on the input feature x. The input features are then mapped to the K-dimension by two fully connected mapping layers and normalized by softmax to ensure that the weighted sum is 1.

$$s = \sigma(W_2\delta(W_1 GAP(x))), \tag{2.4}$$

$$k_i(x) = \frac{e^{s_k}}{\sum_{j=1}^{n} e^{s_j}}, \tag{2.5}$$

where $GAP(x)$ denotes the global average pooling of the input feature x for extracting global spatial information; W_1 and W_2 are the weight matrices of the two fully connected layers, which are used for nonlinear mapping and dimensional transformation of the pooled features; δ denotes the rectified linear unit (ReLU) activation function, which increases the nonlinear expressive ability of the model; σ denotes the softmax function, which is used to normalize the output vector to ensure that the sum of all attention weights is 1; and S_k denotes the unnormalized response score of the kth convolutional kernel. Ultimately, the output feature y is computed by the dynamically generated convolution kernel for convolution.

$$y = W_{dyn}(x) * x = \sum_{i=1}^{n} k_i(x)(conv_i * x). \tag{2.6}$$

2.3. Multi-scale feature extraction

Multi-scale network structures are widely used in computer vision, especially image denoising tasks. By combining multi-scale feature extraction with the skip connections of CNNs, the model

can simultaneously process image features at different scales, improving its robustness and denoising performance [45]. Compared with traditional CNNs that use only fixed-scale convolutional kernels, multi-scale methods can more effectively cope with images with complex structures or diverse scales. Multi-scale feature extraction methods combine convolutional kernels at multiple scales to extract features at different scales. These different scales of convolution kernels can focus on the global structure or local details of the image, enabling the network to capture the multi-level information of the image comprehensively.

For example, larger convolutional kernels model the global structure, while smaller convolutional kernels help to extract local textures and details. A multi-branch structure is usually used in multi-scale feature extraction, with different branches using convolution kernels of various scales for feature extraction. Subsequently, the feature maps generated from each branch are fused. Ultimately, the information from different scales forms a richer and more comprehensive image representation.

Xu et al. [14] proposed a feature-enhanced multi-scale residual network (FEMRNet), which increases the receptive field by adding an enhanced feature extraction block (EFEB), fuses global and local features via the multi-scale residual backbone (MSRB), finely extracts the image information via the detail information recovery block (DIRB), and restores the image details via the merge reconstruction block (MRB). Multi-scale feature extraction performs well in image denoising, so we designed a new multi-scale feature extraction structure to enhance the denoising effect.

3. Network construction

This paper proposes a multi-branch convolutional neural network based on the U-shaped dynamic convolution and multi-scale feature extraction called U-shaped dynamic convolutional multi-scale multi-branch network (UDCMMN) for brain image denoising. The network introduces an encoder–decoder architecture based on an attention mechanism and combines dynamic convolution with a sparse mechanism to extract high-frequency information from images efficiently. In addition, the network replaces the ordinary convolutional kernel in the residual block with different sizes of convolutional kernels extracted from multi-scale features, which enhances the model's ability to adapt to varying scales of noise. Experimental results show that the proposed model, UDCMMN, performs better in the brain image denoising task and can effectively remove noise while maintaining detailed information with high denoising performance and robustness. The overall network architecture of the UDCMMN model is schematically shown in Figure 3.

The UDCMMN brain image denoising model adopts a hierarchical structure, which consists of four key parts: U-Net with a fusion attention mechanism (U-netAM), a dynamic sparse high-frequency network (DSHFN), a multi-scale depth separable residual network (MSDSRN), and a feature processing block (FPB). In Figure 3, the brain noisy images are first used as network inputs fed into three parallel networks. The brain's noisy images are input to U-netAM, which extracts multi-scale features by adaptively focusing on key regions, enhances the recognition of essential structures in brain images, and optimizes the information flow and feature fusion at multiple levels. The DSHFN and MSDSRN extract the overall and local features of the brain image. The FPB fuses the different features, and the first extracted features are further processed to get the final brain image noise feature map. Then, a clearer brain image is obtained by subtracting the input image from the second output noise feature map. Since the FPB partially adopts the structure in the UDRN [33] model, the FPB will not be described in

the network model in this paper. Next, other network components, including U-netAM, DSHFN, and MSDSRN, are described in detail.

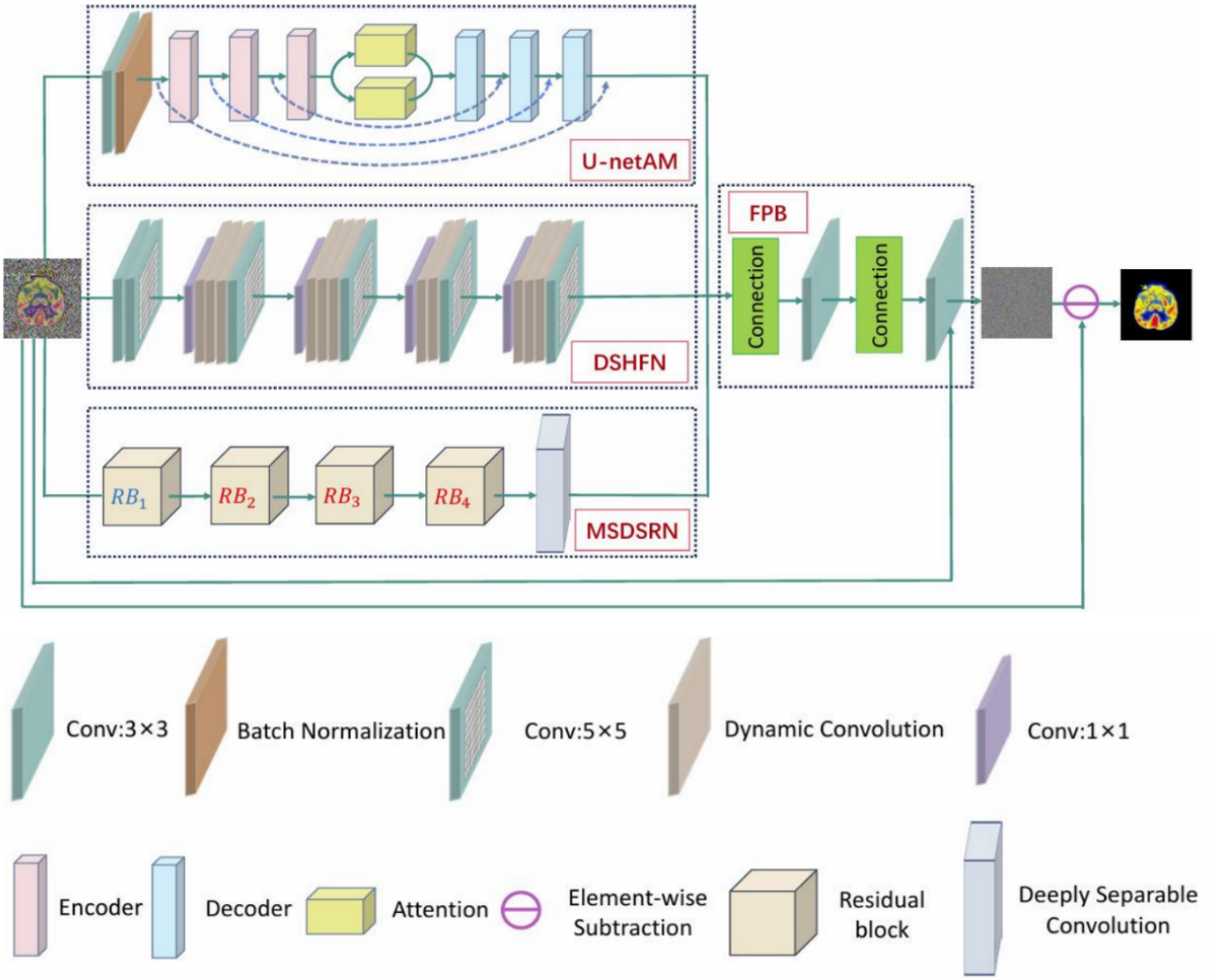

Figure 3. Diagram of the UDCMMN network model.

3.1. U-Net with fusion attention mechanism (U-netAM)

The U-netAM is based on the traditional U-Net and enhances the feature extraction and reconstruction by introducing the channel and spatial attention mechanisms. The network's encoder part gradually extracts the input image's multi-level features. In the process, the channel dimensions of the features are weighted and adjusted by the channel attention mechanism, which highlights the critical channel information. Meanwhile, the spatial attention mechanism focuses on the spatial information of the key regions in the image and adjusts the feature map to be weighted in the spatial dimension. A learnable weighting coefficient fuses the outputs of the dual attention mechanisms to achieve accurate extraction and optimization of multidimensional feature information. The decoder part restores the spatial resolution of the image by gradual upsampling. It utilizes a skip connection to splice the encoder's low-level features with the decoder's high-level features. This preserves the detailed information and effectively integrates the global features. Finally, the number of channels is reduced by a convolutional

layer, and the output image is restored to the original resolution of the input image using a bilinear interpolation method.

Algorithm 1 outlines the training steps of the proposed U-netAM for reference.

Algorithm 1: The forward propagation procedure of U-netAM

Input: Input image $x_0 \in \mathbb{R}^{B\times 3\times H\times W}$

Output: Reconstructed image m_3

Stage 1: Preliminary convolution

$x_1 \leftarrow \text{ReLU}(\text{BN}(\text{Conv}_{3\times 3}(x_0)))$

Stage 2: Encoder path

$x_2 \leftarrow ConvBlock(x_1), \quad x_2' \leftarrow MaxPool2d(x_2)$

$x_3 \leftarrow ConvBlock(x_2'), \quad x_3' \leftarrow MaxPool2d(x_3)$

$x_4 \leftarrow ConvBlock(x_3'), \quad x_4' \leftarrow MaxPool2d(x_4)$

$Ca_{out} \leftarrow ChannelAttention(x_4')$

$Sa_{out} \leftarrow SpatialAttention(x_4')$

$x_5 \leftarrow \alpha \cdot Ca_{out} + (1-\alpha) \cdot Sa_{out}$

Stage 3: Decoder path

$y_1 \leftarrow ConvTranspose2d(\text{ReLU}(\text{BN}(\text{Conv}_{3\times 3}(x_5))))$

$m_1 \leftarrow Concat(y_1, x_3)$

$y_2 \leftarrow ConvTranspose2d(\text{ReLU}(\text{BN}(m_1)))$

$m_2 \leftarrow Concat(y_2, x_2)$

$y_3 \leftarrow ConvTranspose2d(\text{ReLU}(\text{BN}(m_2)))$

$m_3 \leftarrow Concat(y_3, x_1)$

return m_3

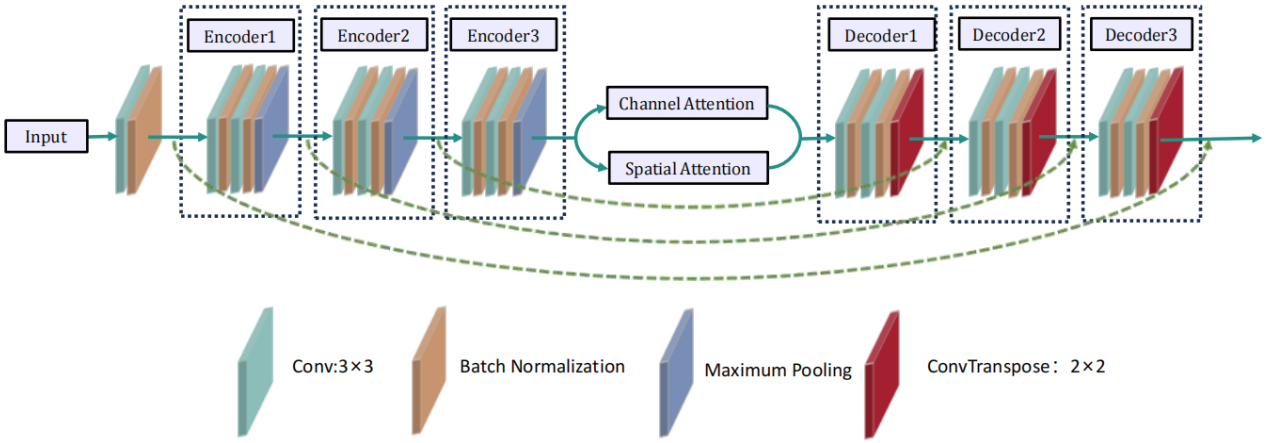

Figure 4. Diagram of the U-netAM of the fusion attention mechanism.

The flowchart of U-netAM is shown in Figure 4. U-netAM is divided into three stages: The preliminary convolutional processing, the encoder, and the decoder. Suppose the input image is $x_0 \in \mathbb{R}^{B\times 3\times H\times W}$, where B is the batch size, 3 is the number of channels in the image, and H and W are the height and width of the image. In the first stage, we introduce a convolutional layer of size 3×3 as a bottleneck layer to adaptively extract the useful information from these localized features to obtain a higher quality first input feature map x_1. The expression for x_1 is as follows:

$$x_1 = RELU(BN(Conv^{3\times3}(x_0))), \tag{3.1}$$

where $Conv^{3\times3}$ indicates that the size of the convolution kernel is 3×3; BN is the batch normalization, which is designed to improve the stability of the training and accelerate the convergence; and $RELU$ is the activation function, which is used to introduce the nonlinear relationship to prevent the model from becoming a linear model.

In the second stage, a convolution operation is performed on x_1 through three identical encoders, each consisting of a convolution block ($ConvBlock$) and a pooling layer, and each convolution block contains two convolution layers and two normalization layers. In the first encoder, x_1 is subjected to a convolution operation, and the output channel is increased from 64 to 128, followed by batch normalization and $RELU$ activation, to obtain the first encoder feature map x_2, which is expressed as follows:

$$x_2 = ConvBlock(x_1). \tag{3.2}$$

Next, x_2 goes through the maximum pooling layer to get the second pooled feature map x_2' with the feature map size halved. The expression for x_2' is as follows:

$$x_2' = MaxPool2d(x_2), \tag{3.3}$$

where $MaxPool2d$ is a 2×2 pooling operation that halves the feature map's height and width. Continuing with a similar approach, the second encoder feature map x_3 and the third pooled feature map x_3' are obtained after the convolution block and the pooling layer process; x_3' is found as follows:

$$x_3 = ConvBlock(x_2'), \tag{3.4}$$

$$x_3' = MaxPool2d(x_3). \tag{3.5}$$

This step increases the number of channels of the feature map from 128 to 256 and reduces the image size by pooling. The convolution block and the pooling layer's last layer obtain the third encoder feature map x_4 and the fourth pooled feature map x_4'.

$$x_4 = ConvBlock(x_3'), \tag{3.6}$$

$$x_4' = MaxPool2d(x_4). \tag{3.7}$$

This step increases the number of channels in the feature map to 512. Next, the channel and spatial attention mechanisms are processed in parallel. The channel attention module learns the weights of each channel and focuses on the critical channel information to strengthen the essential features and get Ca_{out}. The spatial attention module enhances the features in the critical region by learning the importance of the features over the spatial domain to get Sa_{out}.

$$Ca_{out} = \text{ChannelAttention}(x_4'), \tag{3.8}$$

$$Sa_{out} = \text{SpatialAttention}(x_4'). \tag{3.9}$$

Finally, we obtain the final encoder output x_5 by weighting the channel and spatial attention's combined outputs; thus x_5 is found as follows :

$$x_5 = \alpha \cdot Ca_{out} + (1 - \alpha) \cdot Sa_{out}, \tag{3.10}$$

where α is a learnable parameter indicating the proportion of the importance of channel attention and spatial attention.

In the third stage, the compressed feature maps are gradually upsampled to the original image size. The first convolutional block of the decoder upsampling process performs feature extraction through the convolutional block to obtain the first decoder feature map y_1; y_1 is found as follows:

$$y_1 = ConvTranspose2d\left(ReLU(BN(Conv^{3\times3}(x_5)))\right), \tag{3.11}$$

where $ConvTranspose2d$ is a transposed convolution operation, often called "deconvolution", which implements the image upsampling process. After upsampling, it is spliced with the second encoder feature map x_3 to get the first fused feature map m_1; m_1 is found as follows :

$$m_1 = Concat(y_1, x_3). \tag{3.12}$$

The second convolution block of the decoder upsamples m_1 and performs a convolution operation to obtain the second decoder feature map y_2; y_2 is found as follows:

$$y_2 = ConvTranspose2d\left(ReLU(BN(m_1))\right). \tag{3.13}$$

After upsampling, *Concat* is spliced with the first encoder feature map x_2 to obtain the second fused feature map m_2; m_2 is obtained as follows :

$$m_2 = Concat(y_2, x_2). \tag{3.14}$$

The third convolution block of the decoder upsamples m_2 and performs a convolution operation to obtain the third decoder feature map y_3; y_3 is found as follows :

$$y_3 = ConvTranspose2d\left(ReLU(BN(m_2))\right). \tag{3.15}$$

After upsampling, *Concat* is spliced with the first input feature map x_1 to obtain the third fused feature map m_3; m_3 is found as follows:

$$m_3 = Concat(y_3, x_1). \tag{3.16}$$

These steps demonstrate the implementation of the image denoising task by extracting and recovering the image features through an encoder–decoder structure using convolution, pooling, upsampling, and an attention mechanism.

3.2. Dynamic sparse high-frequency network (DSHFN)

The DSHFN can effectively extract and recover high-frequency details in brain image denoising, preserving edge and structural information, which is crucial for diagnostic analysis. The network combines a dynamic convolution kernel with a sparse mechanism based on the image's features and noise patterns. It adaptively generates convolution kernels that selectively decompose the feature map by frequency and convolve with the original features to recover critical information accurately. The process is dynamic, and the generation of the convolution kernel is adjusted in realtime based on the image's content so that critical high-frequency details are effectively retained while noise is removed. Specifically, the DSHFN decomposes the image into low-frequency and high-frequency parts

for processing. The low-frequency part mainly contains the overall structural information of the image, while the high-frequency part focuses on the details and edge information in the image. A high-pass filter enhances high-frequency information. By incorporating a weighted fusion strategy, the DSHFN can recover fine high-frequency details during denoising, effectively suppressing noise while preventing over-smoothing. This leads to an improvement in the quality of brain images.

Algorithm 2 outlines the training steps of the proposed DSHFN for reference.

Algorithm 2: The forward propagation procedure of the DSHFN

Input: Input feature map $x \in \mathbb{R}^{N\times C\times H\times W}$
Output: Output feature map $Output$
Step 1: Kernel generation
$z \leftarrow \text{AdaptiveAvgPool2d}(x)$
$z \leftarrow \text{Conv}^{1\times1}(z)$
$z \leftarrow \text{BN}(z)$
$W \leftarrow \text{Softmax}(z)$
Step 2: Unfold input
$X_u \leftarrow \text{Unfold}(\text{Pad}(x), K)$
Step 3: Low-frequency feature extraction
$low \leftarrow \sum_{i=1}^{K^2} X_u(i) \cdot W(i)$
Step 4: High-frequency feature extraction
$high \leftarrow x - low$
Step 5: Fusion
$Output \leftarrow \alpha \cdot low + \beta \cdot high, \alpha = 0.3, \beta = 0.7$
return $Output$

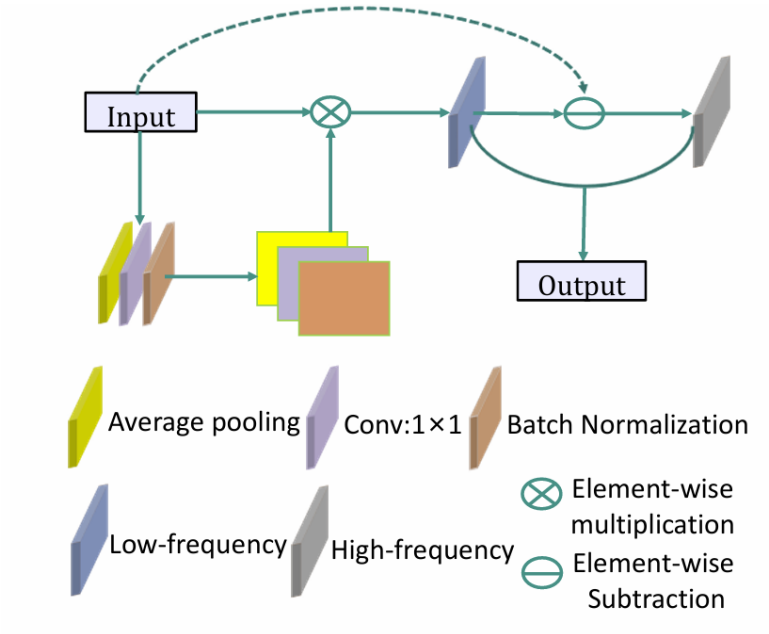

Figure 5. Diagram of dynamic convolution.

The implementation process of dynamic convolution is shown in Figure 5. The specific process is

that the input feature map x is subjected to adaptively average pooling ($AdaptiveAvgPool2d$), 1×1 convolution, batch normalization (BN), and softmax to compute the dynamic convolution kernel W

$$W = Softmax(BN(Conv^{1\times1}(AdaptiveAvgPool2d(x))), \tag{3.17}$$

where W is a dynamic convolution kernel of shape $(N, C \times K^2, 1, 1)$, and the convolution kernel weights are assumed to sum to 1 by softmax normalization. The $Unfold$ operation transforms the input feature map x into a collection of local patches centered around each pixel, resulting in X_u

$$X_u = Unfold(Pad(x), K), \tag{3.18}$$

where K defines the spatial size of the local receptive field or patch for dynamic convolution. $Pad(x)$ ensures that the input edges are properly handled by adding extra pixels around the border. The shape of the local window is $\left(N, kernelNumber, C/\text{kernelNumber}, K^2, H \times W\right)$. The dynamic convolution kernel W is multiplied and summed element by element with the unfolded input features x to obtain the low-frequency features low; low is obtained as follows:

$$low = \sum_{i=1}^{K^2} X_u(i) \cdot W(i). \tag{3.19}$$

The high-frequency information $high$ is obtained by subtracting the low-frequency feature low from the feature map x.

$$high = x - low. \tag{3.20}$$

Finally, the low-frequency and high-frequency features are fused by weighting, setting the weight of the low-frequency features as α and the weight of the high-frequency features as β. The final output is out,

$$out = \alpha \cdot low + \beta \cdot high. \tag{3.21}$$

After several experiments, the hyperparameters α and β were set to $\alpha = 0.3$ and $\beta = 0.7$.

3.3. Multi-scale depth separable residual network (MSDSRN)

The MSDSRN is an efficient network structure that integrates multi-scale feature extraction and depth-separable convolution, aiming to optimize the image feature extraction process and solve the gradient vanishing problem in deep networks by introducing a residual structure.

First, the network employs a multi-scale convolution mechanism that utilizes convolution kernels of different sizes (e.g., 1×1, 3×3, and 5×5) for feature extraction from the input image. Multi-scale convolution can capture different scales of feature information in the image, from local to global. Specifically, 1×1 convolution mainly works on feature compression in the channel dimension, and 3×3 convolution can effectively extract detailed features in the local region. In comparison, 5×5 convolution can capture a larger range of spatial information. By splicing the output features from different scales of convolution, the network can provide a comprehensive feature representation of the image from multiple dimensions, which enhances the perception of diverse information.

Second, the network employs deeply separable convolution to improve the computational efficiency and reduce the number of parameters. Depth-separable convolution works by decomposing the standard

convolution operation into two steps. The first step is deep convolution, which performs the convolution operation independently for each input channel. The second step is point-by-point convolution, which maps the output channels of the depth convolution using 1×1 convolution. This reduces the amount of computation and the number of parameters while preserving the vital information in the image, improving the efficiency and scalability of the model.

In addition, the residual structure in the network effectively alleviates the gradient vanishing problem in training the deep network by directly summing the input features with the output of the convolutional layer. The introduction of residual blocks allows information to propagate more smoothly through the network, improving the training stability of the network and the trainability of the deep network. Each residual block contains a multi-scale convolution block, which effectively promotes the fusion of feature information and enhances the expressive ability of the network. The flowchart of the multi-scale expansion convolution residual block is shown in Figure 6.

Algorithm 3 outlines the training steps of the proposed MSDSRN for reference.

Algorithm 3: The forward propagation procedure of the MSDSRN

Input: Input feature map $x \in \mathbb{R}^{C\times H\times W}$
Output: Output feature map $Output$
Step 1: First multi-scale convolution
$Y_1 \leftarrow \text{ReLU}(BN_1(\text{MultiscaleConv}_1(x)))$
Step 2: Second multi-scale convolution
$Y_2 \leftarrow BN_1(\text{MultiscaleConv}_2(Y_1))$
Step 3: Projection (if needed)
if *channel mismatch or stride* $\neq 1$ **then**
  $X_{proj} \leftarrow \text{MultiscaleConv}_1(x)$
else
  $X_{proj} \leftarrow x$
Step 4: Residual connection
$Output \leftarrow \text{ReLU}(Y_2 + X_{proj})$
return $Output$

As can be seen from Figure 6, in this multi-scale residual block, the input feature map is denoted as $x \in R^{C\times H\times W}$, where C denotes the number of input channels, and H and W are the height and width of the feature map. After the first multi-scale convolution module ($MultiscaleConv_1$), the 1×1, 3×3, and 5×5 convolution results are spliced by the channel dimensions, and the outputs are processed by the batch normalization (BN_1) and the activation function (ReLU) to obtain the intermediate feature Y_1,

$$Y_1 = ReLU(BN_1(MultiscaleConv_1(x))). \tag{3.22}$$

Subsequently, Y_1 enters into the second multi-scale convolution module ($MultiscaleConv_2$) and batch normalizes (BN_1) again to get the output feature Y_2,

$$Y_2 = BN_1(MultiscaleConv_2(Y_1)). \tag{3.23}$$

If the input $MultiscaleConv_1$ is not consistent with the output channel or the step size is not 1, the input is adjusted by the first multi-scale convolution ($MultiscaleConv_1$) to get the projection mapping (X_{proj});

otherwise the original input x is used directly.

$$X_{proj} = \begin{cases} MultiscaleConv_1(x), & \text{if } use\ MultiscaleConv_1 = True. \\ X, & otherwise. \end{cases} \tag{3.24}$$

Finally, the feature map Y_2 is summed with X_{proj} to realize the residual connection, and the final output is generated by the activation function (ReLU).

$$Output = \text{ReLU}(Y_2 + X_{proj}). \tag{3.25}$$

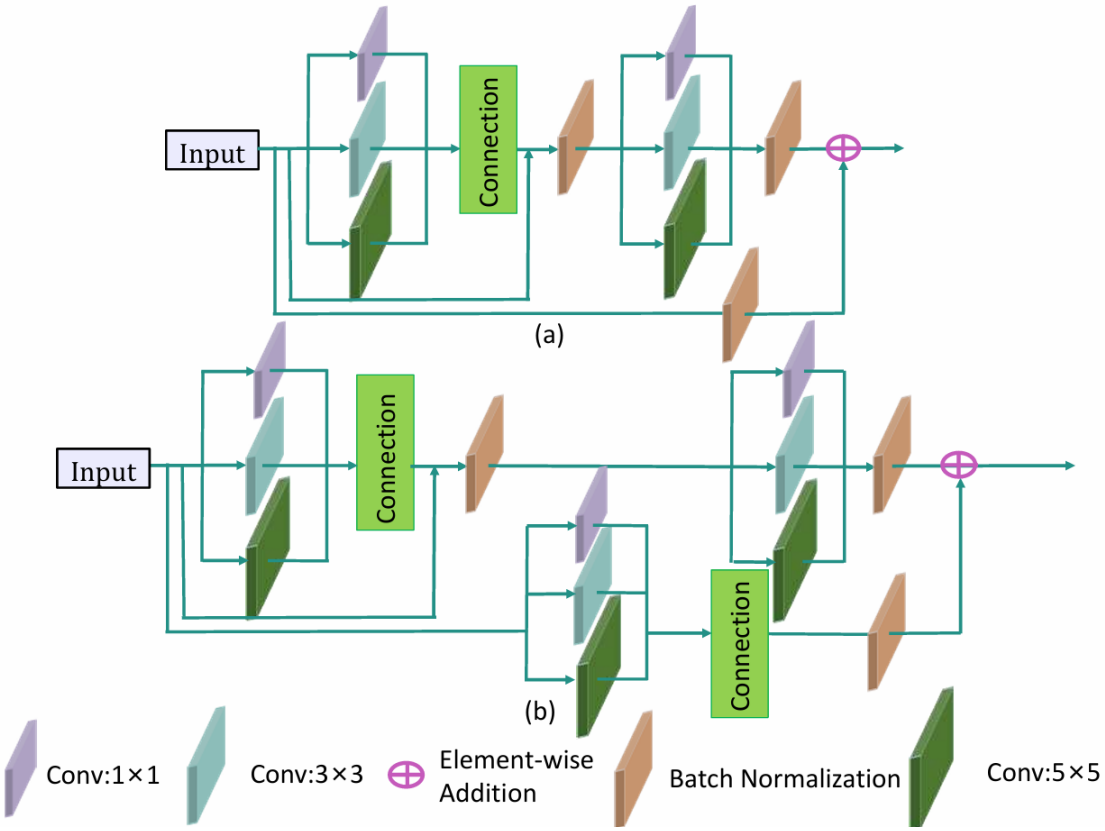

Figure 6. Diagram of the multi-scale dilated convolutional residual block.

4. Numerical experiment

In this section, we describe the dataset used for the experiments and illustrate the specific setup for network training. The contributions of the components of the modeling framework are analyzed, and experiments are carried out on the dataset, where the experimental setup and results are evaluated and discussed quantitatively and qualitatively.

4.1. Experimental preparation

4.1.1. Experimental setup

Compared with stochastic gradient descent (SGD) , the Adam algorithm offers superior convergence speed and enhanced adaptability to complex noise, rendering it more effective for the brain image denoising task in this study [34]. Therefore, experiments are conducted using the Adam algorithm [46] to train and optimize the network, based on the Pytorch framework to train the network. After several experiments, the hyperparameters β_1 and β_2 controlling the decay rate of the first-order and second-order momentum estimation are set to β_1=0.9 and β_2=0.9. The loss function adopts a weighted combination of mean absolute error (MAE) and mean squared error (MSE)

$$\mathcal{L} = \alpha \cdot \frac{1}{N}\sum_{i=1}^{N}|x_i - \hat{x}_i| + \beta \cdot \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2, \tag{4.1}$$

where x_i and $\hat{x}_i$ denote the ground truth and predicted pixel values, respectively, and N is the total number of pixels. MAE emphasizes pixel-level accuracy and is robust to outliers, while MSE penalizes larger deviations and preserves global structure. Their combination achieves a balance between local detail and overall fidelity.

Before training and testing, all images are converted to a tensor and normalized to [0, 1]. The learning rate determines the magnitude of the model's parameter updates. A high learning rate accelerates convergence but may lead to oscillations, so gradually decreasing the learning rate is often used to balance training speed and stability.

This paper dynamically adjusts the learning rate according to different stages during the training process. Specifically, when the number of epochs is 60, the learning rate is set to 1×10^{-5}. Between 40 and 60 epochs, the learning rate is set to 1×10^{-4}. When the number of epochs is 40, the learning rate is set to 1×10^{-3}.

4.1.2. Experimental data

In this study, the AANLIB medical image dataset [47] is selected to evaluate the denoising performance of the UDCMMN model on brain noisy images. The CBSD68 [48] and Set15 [49] datasets are introduced to verify their generalization ability on other color images. To test the model's performance in real noise environments, the SIDD [50] and RNI15 [51] real noise datasets are used. Multiple datasets are synthesized to ensure the scientific and broad applicability of the experiments and to provide a reliable assessment of the denoising performance under different noise conditions.

In this study, 40 brain images are filtered from the dataset, of which 20 brain images are used for pre-training and 20 brain images are used for testing. The pre-training images are cropped by a sliding window of size 41×41 with a step size of 10 and normalized to [0,1]. To expand the diversity of the training dataset, the following eight transformations are applied to each cropped image: (1) Horizontal flipping ; (2) 270° counterclockwise rotation; (3) 180° counterclockwise rotation; (4) 90° counterclockwise rotation; (5) original cropped image; (6) 90° counterclockwise rotation after horizontal flipping ; (7) 180° counterclockwise rotation after horizontal flipping ; (8) 270° counterclockwise rotation after horizontal flipping. This process results in 3872 cropped 41×41 brain image patches used to construct the training dataset. All methods are performed on the same training and test sets with consistent training times to ensure the fairness of the experiments.

4.1.3. Evaluation indicator

In this paper, the peak signal-to-noise ratio (PSNR) [36], structural similarity (SSIM) [52], normalized mean square error (NMSE) [53], multi-scale structural similarity (MS-SSIM) [34], and learned perceptual image patch similarity (LPIPS) [54] are used to compare the evaluation values of different methods quantitatively. Table 1 details each objective evaluation index's value ranges and assessment effects.

Table 1. Objective evaluation metrics for image denoising quality.

Evaluation	Scope	Effect
PSNR	[10 dB, 50 dB]	A higher PSNR implies less distortion and better denoising performance.
SSIM	[0, 1]	A higher SSIM reflects greater structural similarity and improved visual quality.
NMSE	[0, 1]	Values closer to 0 indicate better restoration with less error and distortion.
MS-SSIM	[0, 1]	Higher values reflect better structural similarity across multiple scales.
LPIPS	[0, 1]	Lower values indicate higher perceptual similarity based on deep features.

4.2. Ablation experiment

To validate the effectiveness of each module, three ablation experiments are designed in this paper to evaluate the contribution of the U-netAM, the DSHFN, and the MSDSRN to the model's performance. The AANLIB brain image dataset is used and trained the same number of times under a noise level of 25 to ensure the fairness and reliability of the evaluation results.

4.2.1. U-netAM ablation experiment

To validate the effectiveness of the proposed U-net with the fused attention mechanism in the brain image denoising task, we designed ablation experiments to compare the performance difference between the UDRN and the model after adding the fused attention U-net on this basis. The model adopts a U-shaped encoder–decoder structure and introduces a channel attention mechanism and a spatial attention mechanism for parallel processing in the encoder, which weights and fuses the channel dimension and the spatial dimension of the feature map, respectively, to enhance the network's ability to perceive the key features and to realize the accurate extraction and optimization of multidimensional feature information. In the decoding process, the low-level features in the fusion encoder are fused through skip connections to enhance the detail restoration ability of the reconstructed image effectively. The experimental results are shown in Table 2, where the bold data indicate the best performance.

Table 2. Average test results with and without U-NetAM.

Network	PSNR	SSIM	NMSE	MS-SSIM	LPIPS
UDRN	40.3819	0.9851	0.0013	0.9964	0.0057
U-netAM	**40.9078**	**0.9878**	**0.0012**	**0.9971**	**0.0050**

From Table 2, it can be seen that U-netAM improves PSNR by about 0.53 dB (1.30%) and SSIM by 0.0027 (0.27%) compared with UDRN in the brain image denoising task, indicating that it outperforms UDRN in terms of both image reconstruction quality and structure preservation, and also outperforms

UDRN in the NMSE, MS-SSIM, and LPIPS metrics, which verifies the effectiveness of the proposed U-netAM model in enhancing the performance of brain image denoising.

4.2.2. DSHFN ablation experiment

To verify the effectiveness of the DSHFN proposed in this paper in the task of brain image denoising, we designed an ablation experiment to compare the difference in performance difference between the UDRN and the model after adding dynamic convolution and a sparse mechanism. The experiment replaces the traditional convolutional kernel based on the UDRN with a convolutional kernel that can be dynamically adjusted according to the input features. It combines the sparse mechanism to enhance the extraction of high-frequency information. The network performs multi-level processing on the input image through the dynamic convolution module, forming two branches, low-frequency and high-frequency, which deal with the structural and detailed information of the image, respectively, and fuse the two through weighting to preserve the overall structure and suppress noise interference, improving the ability to restore edges and textures. The experimental results are presented in Table 3, where values in bold indicate the best performance.

Table 3. Average test results with and without DSHFN.

Network	PSNR	SSIM	NMSE	MS-SSIM	LPIPS
UDRN	40.3819	0.9851	0.0013	0.9964	0.0057
DSHFN	**40.9165**	**0.9877**	**0.0012**	**0.9969**	**0.0052**

From Table 3, it can be seen that the introduction of DSHFN shows advantages in five image quality evaluation metrics. Specifically, compared with UDRN, DSHFN improves PSNR by about 1.32%, SSIM by about 0.26%, and MS-SSIM by 0.05%, indicating that it is superior in terms of reconstruction quality and structure preservation. As for the error metrics, the NMSE and LPIPS of DSHFN are reduced by 7.69% and 8.77%, respectively, indicating that it is more advantageous in noise suppression and perceptual quality. In summary, this ablation experiment verified the critical role of DSHFN in enhancing the denoising performance of brain images.

4.2.3. MSDSRN ablation experiment

To validate the effectiveness of the MSDSRN proposed in this paper in the task of brain image denoising, we designed ablation experiments to analyze the effect of the multi-scale convolutional residual block and depth-separable convolution on the improvement of the model's performance. The experiments compare the UDRN and the improved model (MSDSRN). While UDRN employs conventional convolutional kernels for feature extraction, MSDSRN introduces a multi-scale convolutional mechanism that combines convolutional kernels at different scales of 1×1, 3×3, and 5×5 to obtain rich contextual information, together with depth-separable convolution to reduce the number of parameters and the computational complexity. In addition, the convolution module is embedded in the residual structure to enhance the capacity for feature transfer and gradient flow. With this design, the model improves the adaptability to different scale noise patterns while maintaining a light weight. The experimental results are shown in Table 4, where the bold data indicate the best performance.

Table 4. Average test results with and without MSDSRN.

Network	PSNR	SSIM	NMSE	MS-SSIM	LPIPS
UDRN	40.3819	0.9851	0.0013	0.9964	0.0057
MSDSRN	**40.9455**	**0.9882**	**0.0012**	**0.9968**	**0.0053**

Table 4 shows that the proposed MSDSRN achieves an improvement in performance in all five evaluation metrics compared with UDRN. Specifically, PSNR improves by 0.5636 dB (1.40%), SSIM by about 0.31%, and MS-SSIM by about 0.04%, indicating the enhanced quality and structural similarity of the reconstructed images. Meanwhile, the NMSE and LPIPS are reduced by 7.69% and 7.02%, respectively, indicating that the model also performs better in terms of reconstruction error and perceptual quality. These results validate the advantages of MSDSRN in terms of image fidelity and perceptual consistency. In summary, this ablation experiment verifies the critical role of the combination of multi-scale dilated convolutional kernels and depth-separable convolution in enhancing the denoising performance of brain images.

These three branches exhibit complementary and synergistic design principles. The first branch optimizes cross-layer information exchange through an attention-enhanced encoder–decoder architecture, enhancing the fusion of global and local features. The second branch combines dynamic convolutions with sparsity mechanisms to strengthen the modeling capabilities for high-frequency edges and textures. At the same time, the third branch employs multi-scale convolutional kernels to capture noise patterns and structural features across different scales. Their synergistic interaction enables the model to balance global modeling, detail preservation, and multi-scale feature representation. Consequently, it simultaneously ensures denoising performance and structural fidelity in complex noise environments.

4.3. Comparison experiment

The comparison experiment in this study comprises two parts, one based on brain images and the other on open ground based data-sets, to evaluate the generalizability of the proposed model.

4.3.1. Brain images comparison experiment

To verify the effectiveness of the UDCMMN proposed in this paper in the task of brain image denoising, the training set images and test set images are selected from the Harvard Medical dataset, and quantitative experiments and qualitative experiments are conducted on the brain images under noise levels of 15, 25, and 35.

The UDCMMN employs a three-branch parallel structure, and the noisy images are input into the U-NetAM, DSHFN, and MSDSRN sub-networks simultaneously. U-NetAM can focus on key regions adaptively and extract multi-scale structural features by fusing the channel and spatial attention mechanisms within the U-Net network architecture. DSHFN introduces dynamic convolution and sparse mechanisms to enhance the extraction and representation of high-frequency details. MSDSRN combines multi-scale dilation and depth-separable convolution to effectively extract local and global features and improve the model's ability to model complex noise. In the testing process, multiple evaluation metrics, including PSNR, SSIM, NMSE, MS-SSIM, and LPIPS, are used to evaluate the denoising performance comprehensively.

In the experimental setup, we evaluate each model's brain image denoising performance with PSNR

as the evaluation metric at a noise level of 15. Each model is trained on the training set for 70 periods, and then the performance is tested on the test set. The experimental results are shown in Table 5, and the bold font indicates the best result among all the compared methods. As shown in Table 5, the proposed UDCMMN model outperforms other methods in terms of the average PSNR. Compared with ECNDNet, it improves 15.73 dB, which is 55.71%. Compared with DnCNN, it improves by 9.82 dB or 28.73%. Compared with DudeNet, the improvement is 2.45 dB or 5.89%, alongside a 1.26 dB or 2.95% improvement over UDRN. The UDCMMN model demonstrates superior denoising performance on brain images compared with BDARN and MFDRAN. Compared with the former two, the average PSNR improved by approximately 1.11% and 1.50%, respectively. This shows that UDCMMN has a performance advantage. Statistical analysis shows that this enhancement has a significant difference in most comparisons ($p < 0.05$), indicating that UDCMMN has stronger expressive ability and stability in noise suppression, edge preservation, and detail recovery. In the task of denoising low-noise brain images, UDCMMN can effectively remove complex noise interference while maintaining the structural information of the image.

Table 5. PSNR of different methods for denoising brain images at a noise level of $\sigma = 15$.

Image	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
Image 1	28.4641	36.7754	42.8468	44.8510	45.7513	45.4573	**45.9241**
Image 2	28.4608	36.5052	42.7507	44.4998	45.4538	45.0702	**45.6717**
Image 3	28.4407	36.1027	42.5177	44.2240	45.0996	44.7413	**45.2402**
Image 4	28.4105	35.8521	42.3344	44.0132	44.7646	44.5291	**44.9427**
Image 5	28.3796	35.5895	42.1590	43.6382	44.4521	44.2112	**44.7892**
Image 6	28.3510	35.3740	41.9391	43.3480	44.0825	43.8723	**44.3792**
Image 7	28.3397	35.0834	41.8996	43.1651	43.9515	43.6937	**44.2932**
Image 8	28.3440	34.9632	41.9413	43.1350	43.9850	43.6914	**44.3624**
Image 9	28.3479	34.6409	41.9274	43.0917	43.9127	43.6647	**44.3189**
Image 10	28.3383	34.3814	41.8624	43.0721	43.9178	43.6787	**44.3217**
Image 11	28.3039	34.0516	41.7378	42.9175	43.7178	43.4614	**44.2003**
Image 12	28.2399	33.0453	40.7179	41.6658	42.2036	41.8962	**42.7288**
Image 13	28.1921	33.0216	41.0571	41.8494	42.5717	42.5897	**43.2963**
Image 14	28.1380	32.9514	41.0563	42.1193	42.8489	42.8798	**43.4988**
Image 15	28.0882	32.9173	40.8885	41.9185	42.6735	42.7183	**43.3978**
Image 16	28.0496	32.7365	40.6749	41.4084	42.1131	42.3066	**42.9836**
Image 17	28.0403	32.8701	40.8092	41.7762	42.5248	42.4580	**43.1719**
Image 18	28.0375	32.5721	40.7413	41.6556	42.3630	42.2543	**43.0265**
Image 19	28.0437	32.2054	40.4815	41.0596	41.7831	41.6788	**42.5097**
Image 20	28.0294	31.8451	40.5480	41.2084	41.9966	41.9555	**42.7677**
Average	28.2520	34.1742	41.5445	42.7309	43.5084	43.3404	**43.9912**

In the experimental setup, we evaluate each model's brain image denoising performance using PSNR as the evaluation metric at a noise level of 25. Each model is trained on the training set for 70 periods, and then its performance is tested on the test set. The experimental results are shown in Table 6, and the bold font indicates the best result among all the compared methods. As shown in

Table 6, the proposed UDCMMN model outperforms other methods in terms of the average PSNR. Compared with ECNDNet, the PSNR is improved by 18.38 dB, which is 81.15%. Compared with DnCNN, the improvement is 1.72 dB, which is 4.38%. Compared with DudeNet, the improvement is 0.78 dB or 1.93%. Compared with UDRN, the improvement is 0.67 dB, which is 1.65%. UDCMMN demonstrates superior denoising performance on brain images compared with BDARN and MFDRAN. Compared with the former two, the average PSNR improves by approximately 1.03% and 0.22%, respectively. This fully demonstrates the advantage of UDCMMN in image reconstruction quality. Statistical tests show that these enhancements are significantly different in most comparisons ($p < 0.05$). This indicates that UDCMMN has stronger stability and robustness in noise suppression and structural information retention. In denoising moderately noisy brain images, UDCMMN can effectively remove noise interference while maintaining the structural information from the image.

Table 6. PSNR of different methods for denoising brain images at a noise level of $\sigma = 25$.

Image	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
Image 1	22.9376	41.3676	41.6487	42.1468	42.3982	42.5885	**42.6371**
Image 2	22.9287	41.5018	41.6142	41.9843	42.2923	42.4778	**42.6010**
Image 3	22.9010	41.2898	41.3201	41.7332	41.9820	**42.2449**	42.2365
Image 4	22.8602	40.8700	41.2038	41.4941	41.6450	**41.9415**	41.8757
Image 5	22.8243	40.5338	40.9140	41.1757	41.3957	41.6874	**41.7071**
Image 6	22.7923	40.2966	40.6730	40.8990	41.0850	**41.4066**	41.3507
Image 7	22.7726	40.0882	40.5312	40.7417	40.8958	41.1833	**41.2949**
Image 8	22.7658	39.9893	40.6445	40.8250	41.0132	41.2892	**41.3972**
Image 9	22.7703	39.7859	40.5822	40.6830	40.9089	41.2264	**41.2858**
Image 10	22.7700	39.8075	40.5324	40.7228	40.9803	41.3203	**41.3714**
Image 11	22.7279	39.5453	40.4306	40.5869	40.8837	41.2129	**41.2614**
Image 12	22.6578	37.4394	39.0277	39.0619	39.2187	39.5927	**39.7060**
Image 13	22.5887	38.5461	39.7078	39.5001	39.8647	40.3790	**40.4234**
Image 14	22.5117	38.7659	39.9242	39.9429	40.2676	40.5648	**40.6909**
Image 15	22.4485	38.5646	39.7725	39.7896	40.1199	40.4129	**40.6058**
Image 16	22.4038	37.9773	39.5142	39.2908	39.5484	40.0495	**40.1589**
Image 17	22.3863	38.0750	39.7174	39.6761	39.8910	40.2174	**40.4234**
Image 18	22.3865	37.8021	39.5144	39.4951	39.7450	40.0625	**40.2662**
Image 19	22.3913	37.1777	39.0273	38.8715	39.1909	39.5087	**39.7021**
Image 20	22.3719	37.0677	39.1398	39.0170	39.2687	39.7137	**39.9646**
Average	22.6599	39.3246	40.2720	40.3819	40.6297	40.9540	**41.0480**

In the experimental setup, we evaluate each model's brain image denoising performance under a noise level of 35, and the evaluation metric used is PSNR. Each model is trained on the training set for 70 periods, and then the performance is tested on the test set. The experimental results are shown in Table 7, and the bold font indicates the optimal results achieved among all the compared methods. As shown in Table 7, the proposed UDCMMN model outperforms other methods in terms of the average PSNR. Compared with ECNDNet, the PSNR is improved by 17.62 dB, which is 82.57%. Compared with DnCNN, the improvement is 3.21 dB, which is 8.97%. Compared with DudeNet, the

improvement is 0.27 dB or 0.69%. Compared with UDRN, the improvement is 0.25 dB, which is 0.64%. The UDCMMN model demonstrates superior denoising performance on brain images compared with BDARN and MFDRAN. Compared with the former two, the average PSNR improved by approximately 0.55% and 0.43%, respectively. This result further validates the effectiveness and improvement of UDCMMN in image reconstruction quality. These enhancements show significant differences in the statistical tests ($p < 0.05$), further validating the advantages of UDCMMN in noise suppression, image detail retention, and robustness. Especially at high noise levels, UDCMMN can effectively remove noise and better preserve the image's structure and details.

Table 7. PSNR of different methods for denoising brain images at a noise level of $\sigma = 35$.

Image	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
Image 1	21.5883	39.4936	40.4237	40.2731	40.4249	40.3583	**40.5022**
Image 2	21.5857	39.4007	40.3312	40.2481	**40.4359**	40.3111	40.4248
Image 3	21.5613	39.0510	39.9121	40.0118	40.1038	40.0566	**40.1304**
Image 4	21.5301	38.2241	39.8032	39.6742	39.7800	39.7166	**39.8567**
Image 5	21.4965	37.8699	39.3969	39.4052	39.4524	39.3481	**39.5763**
Image 6	21.4685	38.3380	39.1945	39.1782	39.1115	39.1521	**39.2884**
Image 7	21.4527	37.0960	39.0124	38.9465	38.9330	39.0087	**39.1383**
Image 8	21.4485	36.7240	39.1624	39.0521	39.0776	39.0421	**39.2231**
Image 9	21.4545	38.0336	39.1158	38.9940	39.0135	39.0550	**39.1892**
Image 10	21.4502	36.6233	39.0518	39.0971	39.1174	39.1407	**39.3013**
Image 11	21.4118	35.1597	38.8826	38.9561	38.8067	38.9425	**39.0430**
Image 12	21.3424	33.2355	37.1475	37.2345	37.3462	37.4823	**37.5456**
Image 13	21.2832	32.8824	37.9772	38.1361	38.1654	38.2513	**38.4191**
Image 14	21.2210	29.4202	38.1779	38.3204	38.4373	38.4934	**38.7143**
Image 15	21.1624	30.9225	38.0231	38.1840	38.2077	38.3429	**38.5808**
Image 16	21.1152	30.3132	37.7971	37.7674	37.8313	37.9846	**38.1277**
Image 17	21.1036	35.2424	38.0409	38.0425	38.1096	38.1349	**38.4133**
Image 18	21.1040	35.2248	37.7989	37.9100	37.9340	38.0001	**38.3212**
Image 19	21.1058	36.0785	37.3693	37.4680	37.4356	37.4936	**37.7487**
Image 20	21.0883	35.9614	37.3870	37.5240	37.4287	37.7292	**37.8373**
Average	21.3487	35.7648	38.7003	38.7212	38.7576	38.8022	**38.9691**

The experimental results on the test set of brain images are shown in Table 8, and the bold font indicates the optimal results achieved among all the compared methods. As shown in Table 8, on the test set, our proposed UDCMMN model outperforms most other methods in brain image denoising under noise levels of 15, 25, and 35. Specifically, at a noise level of 15, the PSNR value of the UDCMMN improves by about 1.26 dB (+2.95%) compared with the UDRN. At a noise level of 25, the improvement is about 0.67 dB (+1.65%). At a noise level of 35, the improvement is about 0.25 dB (+0.64%). UDCMMN outperforms ECNDNet across all three noise levels, achieving PSNR improvements ranging from 15.74 to 18.39 dB, equivalent to performance gains of 55.71% to 82.56%. Compared with BDARN, UDCMMN demonstrates PSNR enhancements of approximately 0.55% to 1.11%. Compared with MFDRAN, UDCMMN achieved PSNR improvements ranging from approximately 0.23% to 1.50%.

The MS-SSIM of UDCMMN is improved by approximately 7.5%, 22.1%, and 27.6% compared with ECNDNet at noise levels of σ = 15, 25, and 35, respectively. Comparing the results of the other methods, as well as the other four evaluation metrics (SSIM, NMSE, and LPIPS), it can be concluded that the UDCMMN model exhibits good denoising results at noise levels of 15, 25, and 35.

Table 8. Average test values of various denoising methods on the test set.

Evaluation	Noise	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
PSNR	σ = 15	28.2519	34.4296	41.5445	42.7309	43.5084	43.3404	**43.9912**
	σ = 25	22.6599	39.2132	40.2720	40.3819	40.6297	40.9540	**41.0480**
	σ = 35	21.3487	35.5568	38.7003	38.7212	38.7576	38.8022	**38.9691**
SSIM	σ = 15	0.3416	0.9673	0.6767	0.9921	**0.9930**	0.9922	0.9926
	σ = 25	0.2508	0.9872	0.7090	0.9870	0.9871	**0.9883**	0.9873
	σ = 35	0.2319	0.9801	0.7332	0.9818	0.9821	0.9815	**0.9831**
NMSE	σ = 15	0.0215	0.0052	0.0609	0.0008	0.0007	0.0007	**0.0006**
	σ = 25	0.2508	0.0018	0.0612	**0.0012**	**0.0012**	**0.0012**	**0.0012**
	σ = 35	0.1081	0.0045	0.0624	0.0020	**0.0019**	**0.0019**	**0.0019**
MS-SSIM	σ = 15	0.9288	0.9932	0.3719	0.9981	0.9983	0.9983	**0.9985**
	σ = 25	0.8167	0.9963	0.3724	0.9964	0.9967	0.9970	**0.9971**
	σ = 35	0.7802	0.9924	0.3719	0.9947	0.9948	0.9950	**0.9953**
LPIPS	σ = 15	0.0833	0.0165	0.1756	0.0026	0.0023	0.0024	**0.0022**
	σ = 25	0.3533	0.0063	0.1770	0.0057	0.0054	**0.0051**	0.0052
	σ = 35	0.2616	0.0143	0.1796	0.0102	0.0098	**0.0088**	0.0090

Table 9. Average test values of various denoising methods on the training set.

Evaluation	Noise	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
PSNR	σ = 15	28.2634	34.4714	42.5156	42.8301	43.8415	43.5136	**44.1799**
	σ = 25	22.6752	39.5657	40.4391	40.7007	40.8856	41.1243	**41.1947**
	σ = 35	21.3617	35.7628	37.9619	38.9037	38.9340	39.0226	**39.1356**
SSIM	σ = 15	0.3351	0.9598	0.7100	0.9918	**0.9933**	0.9924	0.9931
	σ = 25	0.2441	0.9873	0.7151	0.9872	0.9847	**0.9884**	0.9876
	σ = 35	0.2253	0.9799	0.6958	0.9817	0.9820	0.9817	**0.9833**
NMSE	σ = 15	0.0225	0.0046	0.0603	0.0008	**0.0006**	0.0007	**0.0006**
	σ = 25	0.0835	0.0017	0.0611	0.0014	**0.0012**	**0.0012**	**0.0012**
	σ = 35	0.1132	0.0047	0.0634	0.0020	**0.0019**	**0.0019**	**0.0019**
MS-SSIM	σ = 15	0.9280	0.9924	0.3809	0.9982	0.9984	0.9983	**0.9986**
	σ = 25	0.8145	0.9963	0.3806	0.9964	0.9966	0.9970	**0.9971**
	σ = 35	0.7775	0.9922	0.3772	0.9947	0.9948	0.9951	**0.9953**
LPIPS	σ = 15	0.0853	0.0214	0.1723	0.0027	0.0024	0.0024	**0.0022**
	σ = 25	0.3548	0.0066	0.1737	0.0060	0.0057	**0.0052**	**0.0052**
	σ = 35	0.2654	0.0150	0.1768	0.0098	0.0095	0.0088	**0.0087**

The experimental results on the brain image training set are shown in Table 9, and the bold font

indicates the optimal results achieved among all the compared methods. As shown in Table 9, the UDCMMN model demonstrates superior overall denoising performance compared with the majority of the other methods on the brain image training set at a noise levels of 15, 25, and 35. Under different noise levels, UDCMMN consistently exhibits higher PSNR than UDRN, with an improvement of about 1.35 dB (3.15%) at a noise of level 15, 0.49 dB (1.21%) at a noise of level 25, and 0.23 dB (0.60%) at a noise of level 35, which reflects its robust performance advantage under different noise intensities. UDCMMN maintains consistent advantages across three noise levels ($\sigma = 15, 25, 35$), achieving PSNR values approximately 0.39% to 0.77% higher than those of BDARN and 0.17% to 1.53% higher than those of MFDRAN, demonstrating the best overall performance. The proposed UDCMMN model achieves MS-SSIM improvements of approximately 7.61%, 22.42%, and 28.03% over ECNDNet at noise levels of $\sigma = 15$, $\sigma = 25$, and $\sigma = 35$, respectively. This demonstrates its robust performance advantage under different noise levels. Comparing the other methods under various noise levels and combining performance of the SSIM, NMSE, MS-SSIM, and LPIPS metrics, it is shown that UDCMMN has a stable and excellent denoising ability under different noise conditions.

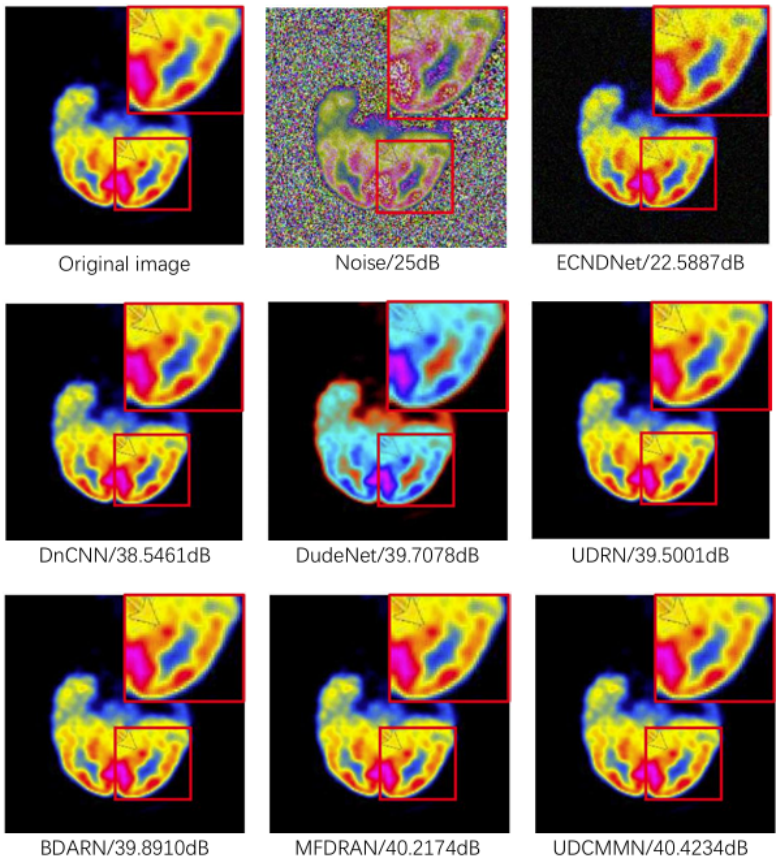

Figure 7. Brain image denoising results by different methods ($\sigma = 25$).

Figure 7 demonstrates the performance of different methods in brain image denoising, and the local zoom in the upper right corner highlights the difference in the effectiveness of each technique in terms of detail recovery. As seen in Figure 7, ECNDNet still has evident residual noise after denoising, DudeNet performs poorly in color reproduction, and the denoised image is off-color. Although the differences between UDCMMN and BDARN/MFDRAN are relatively subtle to the naked eye,

UDCMMN demonstrates superior performance in balancing the details' structural fidelity and noise suppression. In contrast, the proposed UDCMMN model achieves the highest PSNR value in brain image denoising, and the restoration of edges and textures is closer to the original clear image. UDCMMN achieves better denoising performance while maintaining the images' structure and details.

4.3.2. Comparison experiments on ground-based public datasets

To verify the generality and robustness of the denoising model UDCMMN proposed in this paper, two widely used ground-based public datasets, CBSD68 and Set15, are selected for denoising experiments under noise levels of 15, 25, and 35. Meanwhile, the image datasets RNI15 and SIDD, which contain real noise, are also selected further to evaluate the model's actual performance in real scenes.

Table 10. Comparison of the denoising effects of different denoising models on the CBSD68 dataset.

Evaluation	σ	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
PSNR	15	25.8171	27.8904	33.0656	33.2195	33.1039	33.4806	**33.5053**
SSIM	15	0.7687	0.8984	0.5342	0.9572	0.9559	0.9601	**0.9611**
NMSE	15	0.0137	0.0128	0.1342	0.0028	0.0030	0.0026	**0.0025**
MS-SSIM	15	0.9106	0.9520	0.8332	0.9786	0.9780	0.9794	**0.9810**
LPIPS	15	0.0982	0.0910	0.2161	0.0574	**0.0426**	0.0520	0.0459
PSNR	25	21.6414	19.6373	29.8383	30.1969	29.4373	30.5240	**30.8634**
SSIM	25	0.6011	0.7361	0.5272	0.9246	0.9057	0.9315	**0.9361**
NMSE	25	0.0358	0.4405	0.1312	0.0072	0.0092	0.0052	**0.0049**
MS-SSIM	25	0.8209	0.7825	0.8227	0.9568	0.9527	0.9595	**0.9645**
LPIPS	25	0.2319	0.2511	0.2506	0.1161	0.1091	**0.0773**	0.0849
PSNR	35	17.7544	26.9670	28.7866	28.8690	28.8480	28.9186	**28.9603**
SSIM	35	0.4461	0.8575	0.4406	0.9050	0.9039	0.9058	**0.9068**
NMSE	35	0.0876	0.0151	0.1406	0.0081	0.0081	**0.0079**	0.0080
MS-SSIM	35	0.7331	0.9168	0.8066	0.9414	**0.9443**	0.9421	**0.9443**
LPIPS	35	0.5629	0.1971	0.3000	0.1582	0.0929	0.1291	**0.1218**

The experimental results of denoising the CBSD68 dataset with different models are shown in Table 10, and the bold font indicates the optimal results achieved among all the compared methods. As shown in Table 10, the UDCMMN model achieves better denoising performance than the majority of the compared models on the CBSD68 dataset at a noise level of 25. Specifically, the PSNR of UDCMMN is 30.8634 dB, which is improved by 9.22 dB compared with ECNDNet, with an improvement of about 42.61%. The improvement is 11.23 dB compared with DnCNN, which is about 57.16%. Compared with DudeNet, PSNR is improved by 1.03 dB, corresponding to an improvement of about 3.44%. Compared with UDRN, the improvement is 0.67 dB, which is about 2.21%. The proposed UDCMMN model achieves improvements over UDRN of approximately 1.26% in SSIM, a 31.94% reduction in NMSE, an improvement of 0.81% in MS-SSIM, and a 26.91% reduction in LPIPS. Compared with the BDARN and MFDRAN models, the UDCMMN model achieved PSNR improvements of 4.84% and 1.11%, respectively. These results show that UDCMMN has a stronger denoising ability to maintain the overall structural details and image quality. In summary, when denoising the CBSD68 dataset, the UDCMMN

model outperforms the other methods in terms of PSNR, SSIM, NMSE, MS-SSIM, and LPIPS under multiple noise levels, demonstrating more substantial image restoration and higher perceptual quality.

Table 11. Comparison of the denoising effects of the different denoising models on the Set15 dataset.

Evaluation	σ	ECNDNet	DnCNN	DudeNet	UDRN	BDARN	MFDRAN	UDCMMN
PSNR	15	26.1876	26.3118	33.0735	33.2172	33.4016	33.5720	**33.6555**
SSIM	15	0.8097	0.8485	0.3917	0.9593	0.9596	0.9611	**0.9629**
NMSE	15	0.0124	0.0148	0.2792	0.0028	0.0027	0.0026	**0.0025**
MS-SSIM	15	0.9279	0.9444	0.7676	0.9794	0.9793	0.9803	**0.9809**
LPIPS	15	0.0629	0.1290	0.2367	0.0518	0.0448	0.0433	**0.0388**
PSNR	25	22.0992	25.6649	30.8064	31.2390	30.6308	31.1029	**31.4100**
SSIM	25	0.6799	0.8462	0.3470	0.9429	0.9322	0.9441	**0.9465**
NMSE	25	0.0318	0.0153	0.2800	0.0042	0.0050	0.0043	**0.0041**
MS-SSIM	25	0.8583	0.9265	0.7626	0.9665	0.9627	0.9646	**0.9684**
LPIPS	25	0.1575	0.1671	0.2560	0.0802	0.0754	0.0631	**0.0618**
PSNR	35	19.4378	24.6612	28.5512	29.3785	29.0925	29.3000	**29.4341**
SSIM	35	0.5776	0.8585	0.2446	0.9223	0.9091	0.9187	**0.9270**
NMSE	35	0.0586	0.0186	0.2792	0.0064	0.0066	0.0071	**0.0063**
MS-SSIM	35	0.7946	0.8963	0.7559	0.9524	0.9462	**0.9533**	0.9516
LPIPS	35	0.2532	0.1606	0.2667	0.1178	0.1031	**0.0896**	0.1128

The experimental results of denoising the Set15 dataset with different models are shown in Table 11, and the bold font indicates the optimal results achieved among all the compared methods. According to the results in Table 11, it can be seen that the PSNR of UDCMMN reaches 33.6555 dB on the Set15 dataset under a noise of level 15, which is improved to different degrees compared with all the other models. UDCMMN improves by 7.47 dB (28.51%) compared with ECNDNet, UDCMMN improves 7.34 dB (27.92%) compared with DnCNN, UDCMMN improves by 0.58 dB (1.76%) compared with DudeNet, and UDCMMN improves by 0.44 dB (1.32%) compared with UDRN, which fully demonstrates its superior performance in image denoising tasks. Under a noise level of 25, compared with DudeNet, the proposed UDCMMN model achieves improvements of approximately 1.97% in PSNR and 172.66% in SSIM, a 98.54% reduction in NMSE, a 26.99% improvement in MS-SSIM, and a 75.86% reduction in LPIPS. UDCMMN achieves the best performance on the Set12 dataset, with PSNR being improved by 2.54% (0.78 dB) compared with BDARN and by 0.99% (0.31 dB) compared with MFDRAN. In summary, when denoising the Set15 dataset, the UDCMMN model performs well in all five evaluation metrics under noise levels of 15, 25, and 35, and the overall denoising performance is better than that of the other compared methods.

In this study, the denoising performance of the UDCMMN model in real noise environments is explored by conducting experiments on two datasets, RNI15 and SIDD. The RNI15 dataset contains authentic noise images, which can show the denoising advantages of the UDCMMN model under complex noise conditions, and a performance comparison is conducted with other denoising models to verify its superiority in real noise environments. In contrast, the SIDD dataset contains virtually noise-free, high-quality images suitable for in-depth quantitative and qualitative evaluations of UDCMMN

and other models, thus providing a more comprehensive measure of the denoising ability of each model.

Figure 8 shows the comparative results of denoising of different methods on the RNI15 dataset, and the bottom right corner is the local zoomed-in image of the red-boxed area. It can be observed that there is a certain degree of color shift in the image generated by DudeNet, indicating that DudeNet is insufficient in maintaining the color authenticity of the original image. In contrast, the UDCMMN model is closer to the original image regarding subjective visual effects and retains structural details and texture information more completely. In terms of quantitative metrics, according to the PSNR evaluation results, the UDCMMN model achieves the second-highest score after the current optimal model, which further verifies its effectiveness and robustness in image denoising tasks. On the BNI15 dataset, despite subtle differences in subjective visual comparisons, UDCMMN demonstrates superiority over both BDARN and MFDRAN in terms of the PSNR metric. This objective result indicates that our model achieves higher pixel-level reconstruction accuracy, striking a better balance between noise suppression and the fidelity of image detail.

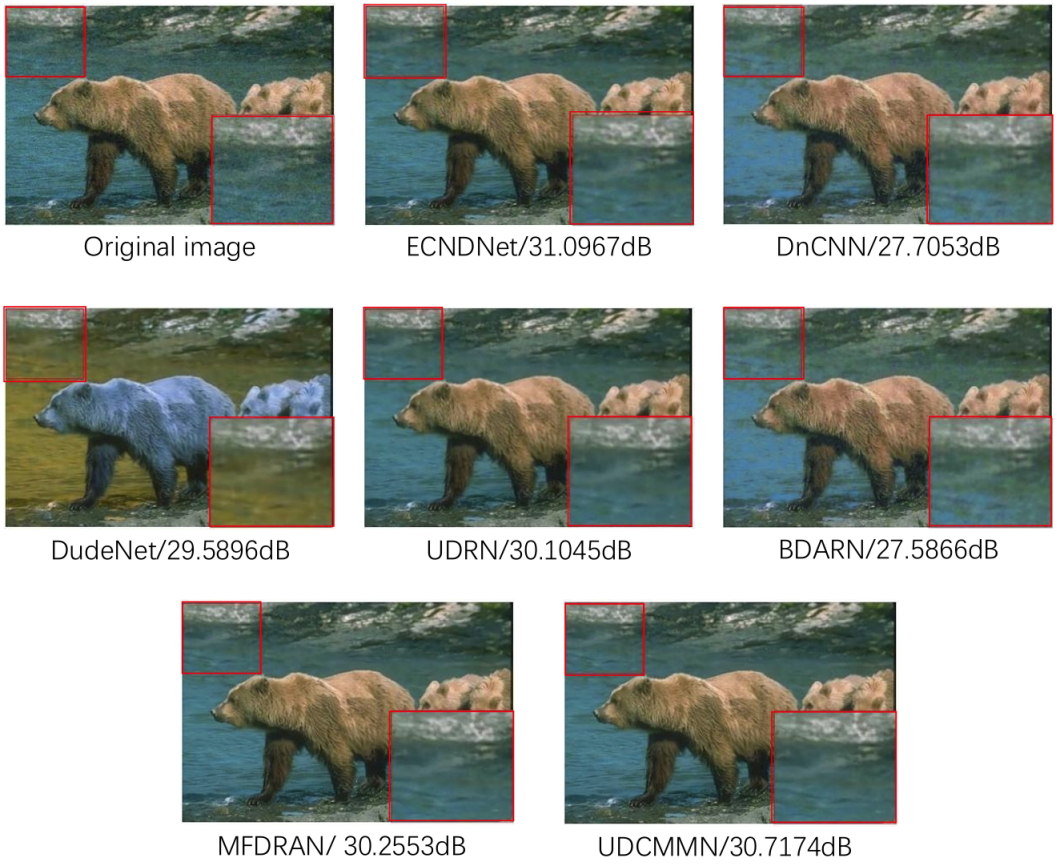

Figure 8. Comparison of the visualization of the results of different methods on the RNI15 dataset.

Figure 9 compares the results of the different methods on the SIDD dataset with the local zoomed-in image of the red-boxed area in the upper right corner for presenting the detailed differences more clearly. From the perspective of image visualization, the DudeNet model introduces a noticeable color shift in the denoising process, resulting in a significant difference between the hue of the output image and the original image. In contrast, the UDCMMN model performs better regarding color fidelity, and

its denoising results are visually closer to the original image. The overall brightness of the output image of the DnCNN model is improved, while the results of the ECNDNet model are on the dark side. According to the PSNR evaluation results, the UDCMMN model achieves the highest score among the compared denoising methods, indicating better performance in preserving the image structure and restoring detail. Combining the subjective visual effect and the objective evaluation indices, the proposed UDCMMN model outperforms other comparative methods in terms of noise suppression, high-frequency information recovery, and structure preservation, demonstrating better visual quality and stronger robustness. Although the visual differences in the denoising results among the different models on the SIDD dataset are limited under extremely high image quality conditions, the objective quantitative metrics (PSNR) indicate that UDCMMN still achieves significant improvements in pixel-level fidelity compared with the BDARN and MFDRAN models. This demonstrates that our model better balances noise suppression and detail preservation.

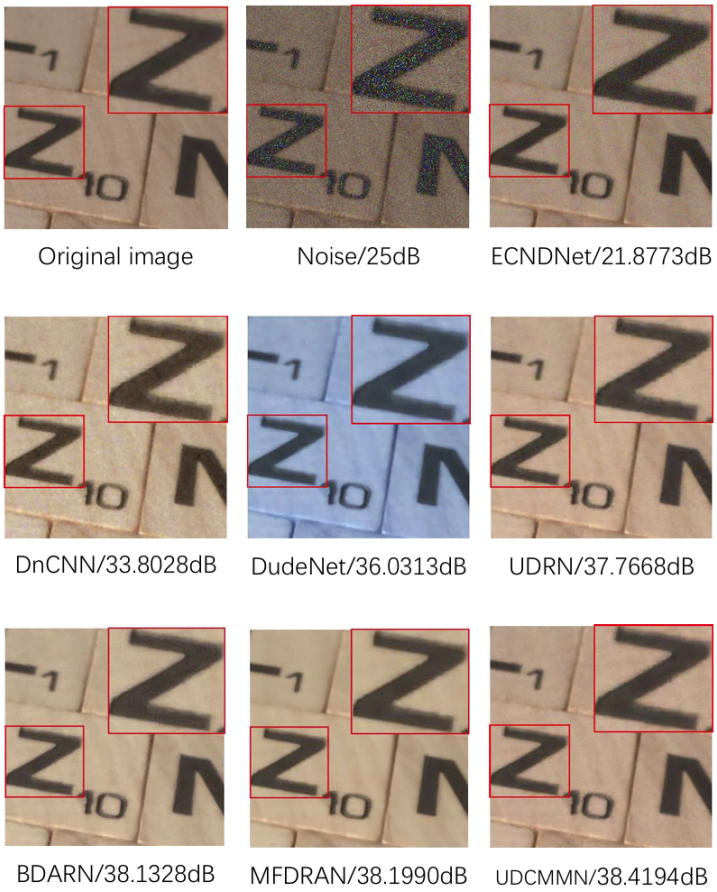

Figure 9. Comparison of the results of different methods on the SIDD dataset.

In the image denoising task, the single runtime of different models versus the number of parameters is shown in Table 12. ECNDNet and DnCNN are single-channel networks; DudeNet, UDRN, and BDARN adopt dual-channel architectures; and MFDRAN and the proposed UDCMMN model are based on multi-channel designs. Compared with the dual-channel model DudeNet, UDCMMN is more advantageous regarding the number of parameters and the runtime. Specifically, in terms of the models' running efficiency, the single running time of UDCMMN is 539.3681 seconds, compared with DudeNet's

832.2255 seconds, a reduction of about 292.86 seconds (35.18%), which improves the running efficiency while maintaining the denoising performance. Regarding model complexity, the number of parameters of UDCMMN is 0.853 M, which is reduced by about 0.227 M (21.02%), compared with DudeNet's 1.080 M, further reflecting the advantage of the model's light weight. Compared with MFDRAN, the UDCMMN model proposed in this study reduces the training time by approximately 29.0%, decreases the number of parameters by about 34.4%, and lowers the GPU memory consumption by roughly 20.8%. This demonstrates that UDCMMN reduces the model's complexity and resource consumption while maintaining the denoising performance. This result shows that the proposed UDCMMN model effectively reduces the model's size while maintaining good denoising performance, which helps to improve the adaptability of the model's deployment in resource-constrained environments. Given the experimental results of brain image denoising in Tables 8 and 9, UDCMMN also outperforms other methods and shows good overall performance in terms of both denoising effect and computational efficiency.

Table 12. Running time, parameters, and memory usage of different models in a single training session.

Models	Training time (s)	Parameters (M)	Memory (MB)
ECNDNet	225.6290	0.520M	131.97
DnCNN	197.7364	0.558M	136.58
DudeNet	832.2255	1.080M	218.79
UDRN	439.2483	0.820M	219.75
BDARN	358.0739	0.824M	262.11
MFDRAN	760.2237	1.301M	509.33
UDCMMN	539.3681	0.853M	403.38

5. Summary

In this paper, an improved brain image denoising model, UDCMMN, is proposed to improve the denoising effect of brain images. The model has four parts: The U-netAM, DSHFN, MSDSRN, and FPB. U-netAM effectively enhances the feature extraction and reconstruction capability and provides richer image information to improve brain images' clinical utility for denoising tasks through multi-level feature extraction. The DSHFN focuses on recovering high-frequency details in the image and preserving critical edge and structure information. The MSDSRN can recognize different features of an image from the local to the global level through multi-scale feature capture and overcomes the gradient vanishing problem in deep networks through the residual structure. The FPB further improves the denoising effect of the model by fusing the features extracted from multiple branches. According to the qualitative and quantitative experimental results, UDCMMN performs very well on various benchmark datasets, preserving high-frequency details and image structure, and improving the denoising effect.

Ablation experiments and comparative results demonstrate that UDCMMN outperforms six other methods in terms of most objective metrics (PSNR, SSIM, NMSE, MS-SSIM, and LPIPS) and subjective visual quality across brain images and multiple public datasets. This indicates the model effectively suppresses noise while maximally preserving tissue details and structural features, enhancing brain images' clinical utility. Furthermore, experiments on denoising other image types demonstrate that

UDCMMN also exhibits excellent performance, further validating its universality and robustness for image denoising tasks. The model's design philosophy includes the integration of the U-Net architecture with convolutional neural networks, the fusion of dynamic convolutional kernels with sparsity mechanisms, multi-scale feature extraction, the combination of dense residual learning with depth-separable convolutions, and a multi-branch feature fusion strategy.

Despite this, the study has limitations. The model was trained and evaluated solely under additive Gaussian noise conditions, making it challenging to handle complex or mixed noise encountered in clinical settings (e.g., Gaussian–Poisson superimposition, motion artifacts, and device interference), and its generalization capability remains to be validated. The experiments relied on a single modality and limited datasets, whereas actual clinical images exhibit variations in equipment, scanning parameters, and individual differences, potentially affecting cross-institutional and cross-population applicability. Furthermore, deep learning in medical imaging continues to face challenges, including insufficient interpretability, high computational resource consumption, and integration with clinical workflows.

Future work will introduce noise modeling methods that more closely approximate real-world scenarios and systematically validate multi-modal, multi-center clinical data to enhance the model's robustness under complex noise and cross-scenario conditions. Concurrently, transfer and self-supervised learning will be explored to reduce the reliance on large-scale labeled datasets. These efforts will be combined with interpretability modules and a lightweight design to enhance the method's universality, explainability, and clinical application value.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work is supported by the Basic Scientific Research Project of the Educational Department of Liaoning Province under grant number 984250013.

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. M. Rahimpour, J. Bertels, A. Radwan, H. Vandermeulen, S. Sunaert, D. Vandermeulen, et al., Cross-modal distillation to improve MRI-based brain tumor segmentation with missing MRI sequences, *IEEE Trans. Biomed. Eng.*, **69** (2022), 2153–2164. https://doi.org/10.1109/TBME.2021.3137561
2. V. Jain, A. Datar, Y. K. Jain, Medical image denoising using optimal attention block-based pyramid denoising network, *Biomed. Signal Process. Control*, **108** (2025), 107794. https://doi.org/10.1016/j.bspc.2025.107794
3. S. Jiang, L. Guo, G. Cheng, X. Chen, C. Zhang, Z. Chen, Brain extraction from brain MRI images based on Wasserstein GAN and O-Net, *IEEE Access*, **9** (2021), 136762–136774. https://doi.org/10.1109/ACCESS.2021.3113309

4. A. Sanaat, H. Arabi, I. Mainta, V. Garibotto, H. Zaidi, Projection space implementation of deep learning–guided low-dose brain PET imaging improves performance over implementation in image space, *J. Nucl. Med.*, **61** (2020), 1388–1396. https://doi.org/10.2967/jnumed.119.239327
5. F. Hashimoto, Y. Onishi, K. Ote, H. Tashima, A. J. Reader, T. Yamaya, Deep learning-based PET image denoising and reconstruction: a review, *Radiol. Phys. Technol.*, **17** (2024), 24–46. https://doi.org/10.1007/s12194-024-00780-3
6. A. Kaur, G. Dong, A complete review on image denoising techniques for medical images, *Neural Process. Lett.*, **55** (2023), 7807–7850. https://doi.org/10.1007/s11063-023-11286-1
7. A. Bal, M. Banerjee, P. Sharma, M. Maitra, An efficient wavelet and curvelet-based PET image denoising technique, *Med. Biol. Eng. Comput.*, **57** (2019), 2567–2598. https://doi.org/10.1007/s11517-019-02014-w
8. G. Elaiyaraja, N. Kumaratharan, T. C. S. Rao, Fast and efficient filter using wavelet threshold for removal of Gaussian noise from MRI/CT scanned medical images/color video sequence, *IETE J. Res.*, **68** (2022), 10–22. https://doi.org/10.1080/03772063.2019.1579679
9. H. V. Bhujle, B. H. Vadavadagi, NLM based magnetic resonance image denoising–A review, *Biomed. Signal Process. Control*, **47** (2019), 252–261. https://doi.org/10.1016/j.bspc.2018.08.031
10. A. K. Singh, Major development under Gaussian filtering since unscented Kalman filter, *IEEE/CAA J. Autom. Sin.*, **7** (2020), 1308–1325. https://doi.org/10.1109/JAS.2020.1003303
11. W. Liang, J. Long, K. C. Li, J. Xu, N. Ma, X. Lei, A fast defogging image recognition algorithm based on bilateral hybrid filtering, *ACM Trans. Multimedia Comput. Commun. Appl.*, **17** (2021), 42. https://doi.org/10.1145/3391297
12. R. Liu, Y. Li, H. Wang, J. Liu, A noisy multi-objective optimization algorithm based on mean and Wiener filters, *Knowl.-Based Syst.*, **228** (2021), 107215. https://doi.org/10.1016/j.knosys.2021.107215
13. B. Shi, M. Li, Y. Lou, Adaptively weighted difference model of anisotropic and isotropic total variation for image denoising, *J. Nonlinear Var. Anal.*, **7** (2023), 563–580. https://doi.org/10.23952/jnva.7.2023.4.07
14. X. Xu, Q. Wang, L. Guo, J. Zhang, S. Ding, FEMRNet: Feature-enhanced multi-scale residual network for image denoising, *Appl. Intell.*, **53** (2023), 26027–26049. https://doi.org/10.1007/s10489-023-04895-9
15. L. Xiang, Y. Qiao, D. Nie, L. An, W. Lin, Q. Wang, et al., Deep auto-context convolutional neural networks for standard-dose PET image estimation from low-dose PET/MRI, *Neurocomputing*, **267** (2017), 406–416. https://doi.org/10.1016/j.neucom.2017.06.048
16. K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising, *IEEE Trans. Image Process.*, **26** (2017), 3142–3155. https://doi.org/10.1109/TIP.2017.2662206
17. C. Tian, Y. Xu, L. Fei, J. Wang, J. Wen, N. Luo, Enhanced CNN for image denoising, *CAAI Trans. Intell. Technol.*, **4** (2019), 17–23. https://doi.org/10.1049/trit.2018.1054
18. B. Pan, D. Pu, L. Fu, F. Zhang, Z. Ping, H. Liu, et al., Deep learning enhanced FDG PET acquisition: a true low dose evaluation, *J. Nucl. Med.*, **64** (2023), 810.

19. K. Spuhler, M. Serrano-Sosa, R. Cattell, C. DeLorenzo, C. Huang, Full-count PET recovery from low-count image using a dilated convolutional neural network, *Med. Phys.*, **47** (2020), 4928–4938. https://doi.org/10.1002/mp.14402

20. X. Hong, Y. Zan, F. Weng, W. Tao, Q. Peng, Q. Huang, Enhancing the image quality via transferred deep residual learning of coarse PET sinograms, *IEEE Trans. Med. Imaging*, **37** (2018), 2322–2332. https://doi.org/10.1109/TMI.2018.2830381

21. I. Häggström, C. R. Schmidtlein, G. Campanella, T. J. Fuchs, DeepPET: A deep encoder–decoder network for directly solving the PET image reconstruction inverse problem, *Med. Image Anal.*, **54** (2019), 253–262. https://doi.org/10.1016/j.media.2019.03.013

22. A. Sano, T. Nishio, T. Masuda, K. Karasawa, Denoising PET images for proton therapy using a residual U-net, *Biomed. Phys. Eng. Express*, **7** (2021), 025014. https://doi.org/10.1088/2057-1976/abe33c

23. A. Mehranian, S. D. Wollenweber, M. D. Walker, K. M. Bradley, P. A. Fielding, K. H. Su, et al., Image enhancement of whole-body oncology [^{18}F]-FDG PET scans using deep neural networks to reduce noise, *Eur. J. Nucl. Med. Mol. Imaging*, **49** (2022), 539–549. https://doi.org/10.1007/s00259-021-05478-x

24. C. Zheng, Y. Li, J. Li, N. Li, P. Fan, J. Sun, et al., Dynamic convolution neural networks with both global and local attention for image classification, *Mathematics*, **12** (2024), 1856. https://doi.org/10.3390/math12121856

25. T. Lei, D. Zhang, X. Du, X. Wang, Y. Wan, A. K. Nandi, Semi-supervised medical image segmentation using adversarial consistency learning and dynamic convolution network, *IEEE Trans. Med. Imaging*, **42** (2023), 1265–1277. https://doi.org/10.1109/TMI.2022.3225687

26. I. Soloviev, A. Kovalchuk, V. Klinshov, Dynamic convolution for image matching, *Eur. Phys. J. Spec. Top.*, 2024. https://doi.org/10.1140/epjs/s11734-024-01373-2

27. S. Yun, Y. Ro, Strengthening dynamic convolution with attention and residual connection in kernel space, *IEEE Access*, **12** (2024), 13626–13633. https://doi.org/10.1109/ACCESS.2024.3356064

28. J. Lee, J. Nam, Multi-level and multi-scale feature aggregation using pretrained convolutional neural networks for music auto-tagging, *IEEE Signal Process. Lett.*, **24** (2017), 1208–1212. https://doi.org/10.1109/LSP.2017.2713830

29. J. Liu, X. Fan, J. Jiang, R. Liu, Z. Luo, Learning a deep multi-scale feature ensemble and an edge-attention guidance for image fusion, *IEEE Trans. Circuits Syst. Video Technol.*, **32** (2021), 105–119. https://doi.org/10.1109/TCSVT.2021.3056725

30. X. Huo, G. Sun, S. Tian, Y. Wang, L. Yu, J. Long, et al., HiFuse: Hierarchical multi-scale feature fusion network for medical image classification, *Biomed. Signal Process. Control*, **87** (2024), 105534. https://doi.org/10.1016/j.bspc.2023.105534

31. C. Tian, Y. Xu, W. Zuo, B. Du, C. W. Lin, D. Zhang, Designing and training of a dual CNN for image denoising, *Knowl.-Based Syst.*, **226** (2021), 106949. https://doi.org/10.1016/j.knosys.2021.106949

32. C. Tian, Y. Xu, W. Zuo, Image denoising using deep CNN with batch renormalization, *Neural Networks*, **121** (2020), 461–473. https://doi.org/10.1016/j.neunet.2019.08.022

33. J. J. Yang, H. Y. Xie, N. N. Xue, A. M. Zhang, Research on underwater image denoising based on dual-channels residual network, *Comput. Eng.*, **49** (2023), 188–198. https://doi.org/10.19678/j.issn.1000-3428.0064662

34. H. Qu, H. Xie, Q. Wang, Brain image denoising using dual-channel attentional residual network, *Digit. Signal Process.*, **165** (2025), 105309. https://doi.org/10.1016/j.dsp.2025.105309

35. X. Wang, Y. Tang, C. Yao, Y. Gao, Y. Chen, DuINet: A dual-branch network with information exchange and perceptual loss for enhanced image denoising, *Digit. Signal Process.*, **156** (2025), 104835. https://doi.org/10.1016/j.dsp.2024.104835

36. H. Qu, H. Xie, Q. Wang, Multi-convolutional neural network brain image denoising study based on feature distillation learning and dense residual attention, *Electron. Res. Arch.*, **33** (2025), 1231–1266. https://doi.org/10.3934/era.2025055

37. Z. Cai, L. Xu, J. Zhang, Y. Feng, L. Zhu, F. Liu, ViT-DualAtt: An efficient pornographic image classification method based on Vision Transformer with dual attention, *Electron. Res. Arch.*, **32** (2024), 6698–6716. https://doi.org/10.3934/era.2024313

38. B. Chen, W. Wu, Z. Li, T. Han, Z. Chen, W. Zhang, Attention-guided cross-modal multiple feature aggregation network for RGB-D salient object detection, *Electron. Res. Arch.*, **32** (2024), 643–669. https://doi.org/10.3934/era.2024031

39. J. Gurrola-Ramos, T. Alarcon, O. Dalmau, J. V. Manjón, MRI Rician noise reduction using recurrent convolutional neural networks, *IEEE Access*, **12** (2024), 128272–128284. https://doi.org/10.1109/ACCESS.2024.3446791

40. Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, J. Liang, UNet++: A nested U-Net architecture for medical image segmentation, *Lect. Notes Comput. Sci.*, **11045** (2018), 3–11. https://doi.org/10.1007/978-3-030-00889-5_1

41. N. Siddique, S. Paheding, C. P. Elkin, V. Devabhaktuni, U-Net and its variants for medical image segmentation: A review of theory and applications, *IEEE Access*, **9** (2021), 82031–82057. https://doi.org/10.1109/ACCESS.2021.3086020

42. J. Yang, D. S. Marcus, A. Sotiras, DMC-Net: Lightweight dynamic multi-scale and multi-resolution convolution network for pancreas segmentation in CT images, *Biomed. Signal Process. Control*, **109** (2025), 107896. https://doi.org/10.1016/j.bspc.2025.107896

43. C. Tian, M. Zheng, W. Zuo, B. Zhang, Y. Zhang, D. Zhang, Multi-stage image denoising with the wavelet transform, *Pattern Recognit.*, **134** (2023), 109050. https://doi.org/10.1016/j.patcog.2022.109050

44. Y. Cheng, W. Zhu, D. Li, L. Wang, Multi-label classification of arrhythmia using dynamic graph convolutional network based on encoder-decoder framework, *Biomed. Signal Process. Control*, **95** (2024), 106348. https://doi.org/10.1016/j.bspc.2024.106348

45. T. Wu, P. Li, J. Sun, B. P. Nguyen, Adaptive edge prior-based deep attention residual network for low-dose CT image denoising, *Biomed. Signal Process. Control*, **98** (2024), 106773. https://doi.org/10.1016/j.bspc.2024.106773

46. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, preprint, arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980

47. K. A. Johnson, J. A. Becker, *The Whole Brain Atlas*, Harvard Medical School, 2005. Available from: https://www.med.harvard.edu/aanlib/home.html.

48. S. Roth, M. J. Black, Fields of experts: A framework for learning image priors, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, **2** (2005), 860–867. https://doi.org/10.1109/CVPR.2005.160

49. M. Bevilacqua, A. Roumy, C. Guillemot, M. A. Morel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding, in *Proceedings British Machine Vision Conference*, (2012), 135. https://doi.org/10.5244/C.26.135

50. A. Abdelhamed, S. Lin, M. S. Brown, A high-quality denoising dataset for smartphone cameras, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, (2018), 1692–1700.

51. M. Lebrun, M. Colom, J. M. Morel, The noise clinic: a blind image denoising algorithm, *Image Process. OnLine*, **5** (2015), 1–54. https://doi.org/10.5201/ipol.2015.125

52. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.*, **13** (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861

53. P. Jakub, M. Grzegorz, Based on spectral analysis of voice controllable surveillance system using normalized mean square error, *Adv. Mater. Res.*, **340** (2012), 156–160. https://doi.org/10.4028/www.scientific.net/AMR.340.156

54. T. Li, H. Feng, L. Wang, L. Zhu, Z. Xiong, H. Huang, Stimulating diffusion model for image denoising via adaptive embedding and ensembling, *IEEE Trans. Pattern Anal. Mach. Intell.*, **46** (2024), 8240–8257. https://doi.org/10.1109/TPAMI.2024.3432812

AIMS Press