

## Superresolution Reconstruction of Magnetic Resonance Images Based on a Nonlocal Graph Network

Yuanhang Li, Xiang Li, Yidan Yan and Chaochao Hu\*

College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, P R China

### Abstract

**INTRODUCTION:** High-resolution (HR) medical images are very important for doctors when diagnosing the internal pathological structures of patients and formulating precise treatment plans.

**OBJECTIVES:** Other methods of superresolution cannot adequately capture nonlocal self-similarity information of images. To solve this problem, we proposed using graph convolution to capture non-local self-similar information.

**METHODS:** This paper proposed a nonlocal graph network (NLGN) to perform single magnetic resonance (MR) image SR. Specifically, the proposed network comprises a nonlocal graph module (NLGM) and a nonlocal graph attention block (NLGAB). The NLGM is designed with densely connected residual blocks, which can fully explore the features of input images and prevent the loss of information. The NLGAB is presented to efficiently capture the dependency relationships among the given data by merging a nonlocal operation (NL) and a graph attention layer (GAL). In addition, to enable the current node to aggregate more beneficial information, when information is aggregated, we aggregate the neighbor nodes that are closest to the current node.

**RESULTS:** For the scale  $r=2$ , the proposed NLGN achieves PSNR of 38.54 dB and SSIM of 0.9818 on the  $T(T1, BD)$  dataset, and yielding a 0.27 dB and 0.0008 improvement over the CSN method, respectively.

**CONCLUSION:** The experimental results obtained on the IXI dataset show that the proposed NLGN performs better than the state-of-the-art methods.

**Keywords:** Magnetic resonance imaging, Superresolution reconstruction, Nonlocal operation, Nonlocal self-similarity, Graph attention

Received on 30 March 2022, accepted on 02 May 2022, published on 04 May 2022

Copyright © 2022 Yuanhang Li *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [Creative Commons Attribution license](#), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eetiot.v8i29.769

\*Corresponding author. Email: [chaos-hu@home.hpu.edu.cn](mailto:chaos-hu@home.hpu.edu.cn)

### 1. Introduction

In recent years, medical imaging technologies, such as computed tomography (CT), magnetic resonance imaging (MRI), and positron emission tomography (PET), have played important roles in scientific research and clinical medicine. Notably, because magnetic resonance (MR) images have the advantage of producing clear images with high soft tissue contrast and distinct characteristics, they have gradually become the main data source for model training in medical auxiliary systems based on deep learning.

Resolution is one of the most important measures for MR images, and high-resolution (HR) MR images are especially helpful for clinicians when performing diagnoses. However, the acquisition of HR MR images may increase the cost of the system, increase the scanning time, and reduce the signal-to-noise ratio (SNR) due to the employed hardware device, body motion, and imaging time. To address these problems, superresolution (SR) reconstruction technology can be used to reconstruct low-resolution (LR) MR images, and higher-quality MR images can be obtained under the same imaging environment and hardware equipment. Image superresolution (SR) reconstruction refers to the process of recovering high-resolution (HR) images from low-resolution (LR) images. Currently, the SR reconstruction technique is

widely used for images in many fields, such as face images [1, 2], remote sensing images [4, 5], medical images [5-8].

Traditional SR methods are mainly based on interpolation [9], reconstruction [10, 11], and shallow learning [12, 13]. Although interpolation-based methods are computationally simple and efficient in terms of image details, they often lead to artifacts in reconstructed MR images. Reconstruction-based methods can overcome the unfavorable oversmoothing effects of interpolation-based methods. However, they depend on accurate image registration. Methods based on shallow learning can achieve high-quality reconstructed images. However, these methods have difficulty obtaining the optimal model parameters, and they are not suitable for most image reconstruction tasks.

With the successful application of deep learning technology in image classification [14-19], target detection [20-22], and image segmentation [23, 24], this approach has also been applied in SR reconstruction tasks. The superresolution convolutional neural network (SRCNN) [25] was first proposed based on a CNN, and it has achieved better results than those of the traditional methods. The SR approach for very deep convolutional networks (VDSR) [26] was presented to increase network depth and reduce the training difficulty based on the residual connections of a residual network (ResNet) [27]. Lim et al. [28] built a more profound and better-performing network, the enhanced deep super-resolution network (EDSR), by deleting the batch normalization (BN) layer of the residual block because they found that the BN layer was not adequate for reconstruction tasks. To improve information flow, the cascaded multiscale cross network (CMSCN) [29] was proposed and used to progressively cascade a series of subnetworks together to infer high-resolution features. The channel splitting network (CSN) [6] was presented and used to divide the features into two parts along the channel dimension and adopt different mechanisms to explore different information, thus realizing the different treatment of channel features. In addition, some other improved methods were introduced by increasing the depth of the base model or reusing the derived features to achieve improved reconstruction effectiveness; such networks include the deep recursive residual network (DRRN) [30], residual dense network (RDN) [31] and the wide residual network with a fixed skip connection (FSCWRN) [7]. However, in the methods mentioned above, the spatial features are treated equally, and the dependencies among the pixels are not considered.

To fuse the dependencies among the pixels of an input image, more methods have been proposed by researchers. The nonlocal recurrent network (NLRN) [32] was proposed to capture the available nonlocal self-similarity information by the combination of a nonlocal (NL) operation module and a CNN. The residual nonlocal attention network (RNAN) [33] was proposed to improve the ability of models to capture local features via residual local and nonlocal attention blocks. This approach can also maintain the dependencies between the attention feature maps of images. Subsequently, the second-order attention network (SAN) [34] was designed with a nonlocally enhanced residual

group structure and NLRs to capture long-distance spatial contextual information.

Although nonlocal self-similarity was studied deeply in the NLRN [32], RNAN [33] and SAN [34], local convolution cannot describe the interrelationships between blocks. That is, it may not be used to deal with non-Euclidean data. To process non-Euclidean data, researchers introduced graph neural networks (GNNs). Graph convolution based on spectrogram theory was first proposed in convolutional GNNs (ConvGNNs) [35]. To overcome the high complexity of ConvGNNs, ChebNet [36] and a graph convolutional network (GCN) [37] were developed by approximations and simplifications. The graph attention network (GAT) [38] uses an attention mechanism to capture the similarity levels of neighbor nodes relative to the current node. Graph convolution has the advantages of possessing a strong representation ability, capturing dependency relationships, and aggregating and delivering information [39, 40]. Therefore, it has also been studied in image processing tasks. To extract the correlation between features, Xu et al. [41] performed graph convolution on the features, thereby improving the SR reconstruction effect. Yan et al. [42] used GAT [38] to explore the interrelationships between different subregions in the feature map, helping to restore the texture structure and improve the reconstruction effect. However, when the above graph convolution model aggregates information, the current node aggregates the information of all neighboring nodes, and the aggregated information may interfere with itself.

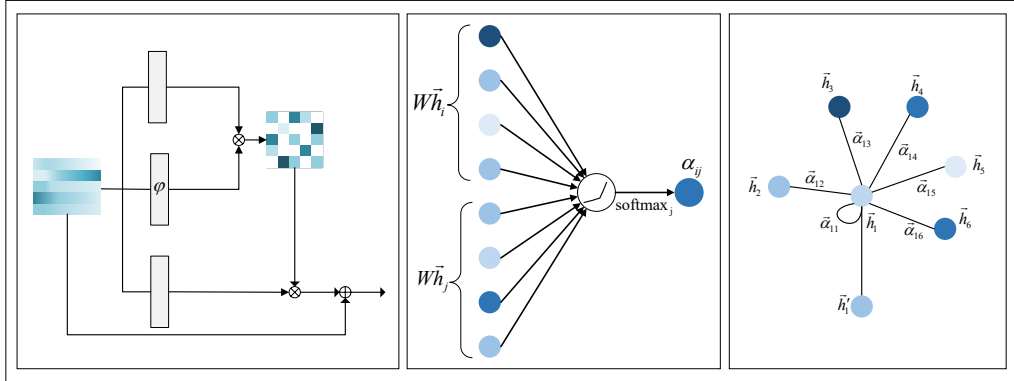
The current methods have the following shortcomings: (i) Nonlocal self-similarity information cannot be fully captured by local convolution; (ii) Most graph convolution operation may aggregate negative information to the current node. In view of the above problems, a nonlocal graph attention layer (NLGAL) is presented in this paper. It can fully capture the nonlocal self-similarity information of images by combining a GAL [38] and nonlocal self-similarity. Furthermore, a nonlocal graph network (NLGN) is designed based on the NLGAL for MR image reconstruction. The new model can capture the nonlocal self-similarity information and the remote dependencies between the obtained feature maps. Hence, it efficiently improves the quality of SR reconstruction. In addition, unfavorable information is avoided when the current node aggregates information. In this study, a new strategy is used. When information is aggregated, the top  $k$  neighbor nodes that are most similar to the current node are selected for aggregation. The IXI dataset<sup>2</sup> is chosen for verification and comparison with state-of-the-art methods, such as bicubic [9], the SRCNN [25], the VDSR [26], the RDN [31], CMSCN [29], the FSCWRN [7] and CSN [6]. The experimental results prove that the presented method achieves better reconstruction results than competing approaches.

The rest of this paper is organized as follows. In Section 2, we briefly review the related work. Section 3 and Section

<sup>2</sup> <http://brain-development.org/ixi-dataset/>

4 present a detailed analysis of the newly proposed schemes, followed by an extensive experimental comparison on the

IXI dataset. Finally, we conclude our work.



**Figure 1.** a) Nonlocal operation (NL), b) Schematic diagram of the graph attention layer (GAL) attention weight calculation, and c) The GAL information aggregation diagram

## 2. Related Work

### 2.1. Nonlocal Operation (NL)

The nonlocal operation was proposed in the NLRN [32], and it was formulated as follows:

$$Z = \frac{1}{\delta} \sum \Phi(X)G(X) \quad (1)$$

Where  $X \in R^{N \times m}$  is the input of the NL,  $Z \in R^{N \times k}$  is its output,  $N$  denotes the number of image pixels, and  $m$  and  $k$  are the input feature length and the output feature length, respectively.  $\Phi(X) \in R^{N \times N}$  is a nonlocal correlation matrix, and it is used to calculate the similarity relationship between the blocks.  $G(X) \in R^{N \times k}$  is a nonlocal transformation matrix. The output is normalized by  $1/\delta$  with a normalization factor of  $\delta$ .

As shown in Figure 1.a, the NL is implemented using a  $1 \times 1$  convolution kernel, where  $\theta$ ,  $\varphi$  and  $g$  are the weight parameters,  $\otimes$  represents matrix multiplication and  $\oplus$  represents elementwise addition.

### 2.1. Graph Attention Layer (GAL)

For a set of input nodes, the GAL first calculates the attention coefficient between two nodes. For a node  $i$  and its neighbor node  $j$ , the attention coefficient  $e_{ij}$  between them is defined as follows [38]:

$$e_{ij} = a(W\vec{h}_i, W\vec{h}_j) \quad (2)$$

Where  $a$  denotes the attention mechanism,  $\vec{h}_i$  and  $\vec{h}_j$  represent the feature vectors of nodes  $i$  and  $j$ , respectively, and  $W$  represents the linear transformation matrix.

Suppose that  $N_i$  represents all the neighbors of node  $i$ . To make the similarities of different nodes comparable, the *Softmax* function is used for normalization; this function is given as:

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \quad (3)$$

The attention mechanism  $a$  is a single-layer feedforward neural network. It is parameterized by a weight vector and the leaky rectified linear unit (*LeakyReLU*) function for nonlinear activation. The calculation of the attention mechanism is shown in Figure 1.b. Therefore,  $\alpha_{ij}$  can be further expressed as follows:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\vec{a}^T [W\vec{h}_i \| W\vec{h}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\vec{a}^T [W\vec{h}_i \| W\vec{h}_k]))} \quad (4)$$

Where  $T$  represents the transposition operation and  $\|$  is the concatenation operation.

Then,  $\vec{h}_i$  can be calculated by the following formula:

$$\vec{h}_i = \sigma\left(\sum_{j \in N_i} \alpha_{ij} W\vec{h}_j\right) \quad (5)$$

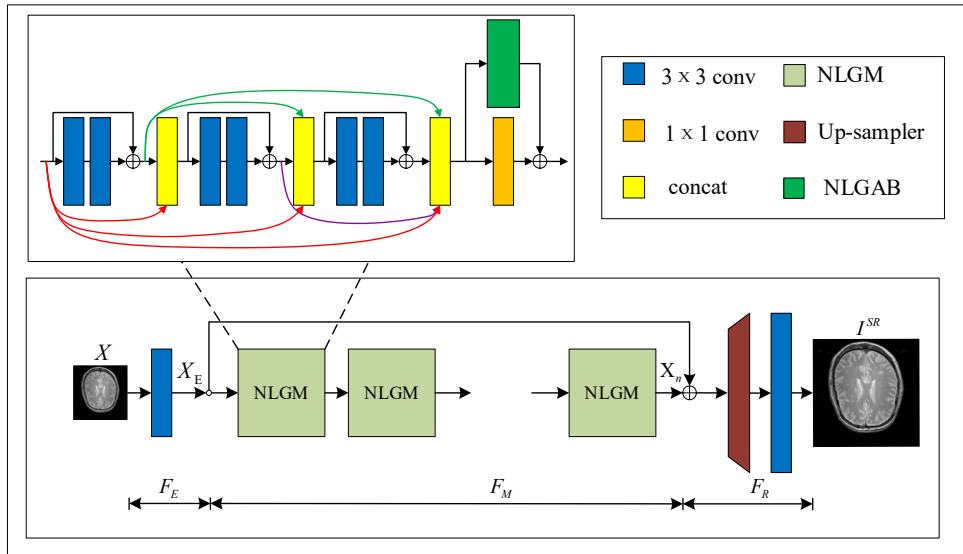
where  $\sigma(\cdot)$  stands for the nonlinear activation function. The information aggregation process is shown in Figure 1.c.

## 3. Nonlocal Graph Network (NLGN)

### 3.1. Network Architecture

The NLGN model proposed in the paper is shown in Figure 2.a. Similar to other image SR models, the NLGN mainly consists of three parts: a feature extraction network, a nonlinear mapping network, and a reconstruction network. The feature extraction network is used to extract the shallow features of the input image  $X$ . The deep features and

nonlocal self-similarity information are extracted and fused by the nonlinear mapping network. Finally, SR images are reconstructed by the reconstruction network.



**Figure 2.** The diagram of the proposed NLGN. a) The overall structure. b) The architecture of the nonlocal graph module

### Feature Extraction Network

The feature extraction network only uses a  $3 \times 3$  convolution to extract the input shallow features.  $F_E(\cdot)$  denotes the mapping function of the network, and the extracted feature  $X_E$  can be represented as:

$$X_E = F_E(X) \quad (6)$$

Where  $X$  represents a low-resolution input image.

### Nonlinear Mapping Network

The nonlinear mapping network contains a series of stacked nonlocal graph modules (NLGMs).  $X_E$  represents the input of the first NLGM. And the input and output of the  $i$ -th NLGM are  $X_{i-1}$  and  $X_i$ , respectively. Therefore, the output  $X_i$  of the  $i$ -th NLGM can be expressed as:

$$X_i = F_{nlgm}^i(X_{i-1}), i = 1, \dots, n, \quad (7)$$

Where  $F_{nlgm}^i(\cdot)$  corresponds to the operations of the  $i$ -th NLGM. For convenience,  $X_E = X_0$ . Then, the output of the last network, which is the output of the nonlinear mapping network, can be expressed as:

$$\begin{aligned} X_n &= F_M(X_E) \\ &= F_{nlgm}^n(X_{n-1}) = F_{nlgm}^n(F_{nlgm}^{n-1}(\dots(F_{nlgm}^1(X_0))\dots)) \end{aligned} \quad (8)$$

Where  $X_n$  represents the output of the  $n$ -th NLGM,  $X_{n-1}$  represents the input of the  $n$ -th NLGM, and similarly,  $X_0$  represents the input of the first NLGM.  $F_M(\cdot)$  denotes the mapping function of the nonlinear mapping network.

### Reconstruction Network

The reconstruction network consists of an upsampling module and a  $3 \times 3$  convolutional layer. The upsampling module first reconstructs the input feature maps into SR features through a subpixel shuffling layer [43]. Then, the network uses a  $3 \times 3$  convolutional layer to build the SR features into the final output. The mapping function of the reconstruction network can be expressed as follows:

$$I^{SR} = F_R(X_n + X_E) \quad (9)$$

Where  $X_n$  and  $X_E$  represent the deep and shallow features of the input images respectively,  $I^{SR}$  represents the output,  $F_R(\cdot)$  represents the mapping function of the reconstruction network.

### 3.2. Nonlocal Graph Module

The architecture of the NLGM is shown in Figure 2.b. It is composed of a dense residual block [44] and a nonlocal graph attention block (NLGAB).

#### Dense Residual Block

The dense residual block is composed of three densely connected residual blocks, and each residual block has two convolutional layers and two *ReLU* activation functions. The output  $X_{DR}$  of the dense residual block can be expressed as:

$$X_{DR} = F_{DR}(X_{i-1}) \quad (10)$$

Where  $X_{i-1}$  represents the input of the  $i$ -th NLGM and  $F_{DR}(\cdot)$  represents the mapping function of the dense residual block. Dense residual blocks can prevent the loss of

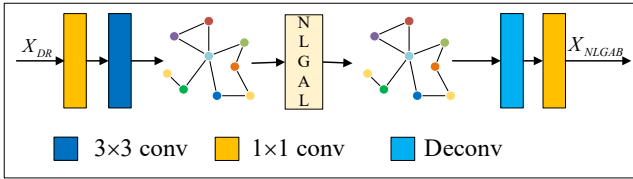
information during feature transfer. For more information, please refer to [44].

### NLGAB

The NLGAB architecture is shown in Figure 3. It is mainly composed of an NLGAL and convolutions. The NLGAB first uses a  $1 \times 1$  convolutional layer to reduce the dimensionality of  $X_{DR}$ . Then, it chooses a  $3 \times 3$  convolutional layer with a stride of 2 for downsampling, and the downsampled features are used as the inputs of the proposed NLGAL. After that, deconvolution is introduced to restore the generated output to its original size. Finally, a  $1 \times 1$  convolutional layer is applied to compress and reconstruct the deconvolution result. The whole procedure can be expressed as follows:

$$X_{NLGAB} = F_{NLGAB}(X_{DR}) = f_{c_{1 \times 1}}(f_{dc_{3 \times 3}}(F_{NLGAL}(f_{c_{3 \times 3, s=2}}(f_{c_{1 \times 1}}(X_{DR})))) \quad (11)$$

Where  $F_{NLGAB}(\cdot)$  represents the mapping function of the NLGAB,  $f_{c_{1 \times 1}}(\cdot)$  is a  $1 \times 1$  convolution,  $f_{dc_{3 \times 3}}(\cdot)$  is a  $3 \times 3$  deconvolution,  $F_{NLGAL}(\cdot)$  represents the NLGAL, and  $f_{c_{3 \times 3, s=2}}(\cdot)$  is a  $3 \times 3$  convolution with a stride of 2.



**Figure 3.** Nonlocal graph attention block (NLGAB)

Generally, for an NLGM, the dense residual block is first used to deal with the input information. Then, the NLGAB is presented to capture the nonlocal self-similarity information of  $X_{DR}$ . In addition,  $X_{DR}$  is further processed by the bottleneck layer for compression. Finally, the output of the NLGAB and the compression result are combined as

the input of the next NLGM. Furthermore, the output of an NLGM can be expressed as:

$$X_i = F_{NLGM}(X_{i-1}) = f_{c_{1 \times 1}}(X_{DR}) + F_{NLGAB}(X_{DR}) \quad (12)$$

Where  $X_i$  is the output of the  $i$ -th NLGM,  $f_{c_{1 \times 1}}(\cdot)$  is the bottleneck layer with  $1 \times 1$  convolution, and  $F_{NLGAB}(\cdot)$  represents the operation of NLGAB.

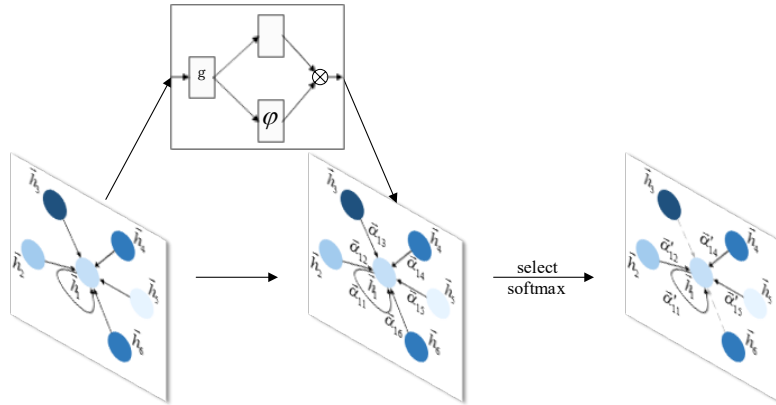
### 3.3. Nonlocal Graph Attention Layer (NLGAL)

Both the GAL [38] and NLS [32] can capture the context information and remote dependencies. However, they are different from each other. NLS are used to aggregate the information between all blocks. However, only the information between the neighboring nodes in each block is aggregated by the GAL. In addition, NLS cannot fully capture the relationship between two blocks by convolution. The GAL can effectively characterize and calculate the relationship between two blocks (that is, the adjacent nodes). In general, the GAL is more suitable for capturing nonlocal information than NLS. However, the space and time complexity of the GAL is very large due to the use of a parameter matrix and a single-layer feedforward neural network.

By the combination of NLS and the GAL, the NLGAL is proposed in this paper. Specifically, a linearly embedded Gaussian kernel is chosen to replace the calculation of the attention coefficient in the GAL. In addition, convolution is used to realize the linear transformation. Formula (4) is rewritten as:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}((W\vec{h}_i W_\theta)(W\vec{h}_j W_\varphi)^T))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}((W\vec{h}_i W_\theta)(W\vec{h}_k W_\varphi)^T))} \quad (13)$$

Where  $W$ ,  $W_\theta$  and  $W_\varphi$  are learnable parameters. And we used Formula (13) to calculate the similarity between nodes (that is attention coefficient). The operation of the NLGAL is shown in Figure 4.



**Figure 4.** The architecture of the nonlocal graph attention layer

In addition, the GAL aggregates the information of all neighboring nodes relative to the current node when performing information aggregation. We think this may cause the current node to aggregate information that is unfavorable to it. Notably, the similarities between the distant nodes are actually very small. Aggregating similar nodes with the current node will interfere with the final information of the current node. To prevent this impact, when the NLGAL aggregates information, it only aggregates the top  $k$  nodes that are most similar to the current node. Specifically, the first  $k$  nodes with the largest attention weights are first selected; second, these weights are normalized to obtain a new attention weight; and finally, information aggregation is performed based on the new attention weight. It is worth noting that the NLGAL first selects nodes for normalization because it can assign greater attention weights to similar nodes.

### 3.4. Loss Function

In view of the better performance of the  $L_1$  loss function [28], we chose it as the loss function of our NLGN. For a given training set  $D = \{x_{(i)}, y_{(i)}\}, i = 1, 2, \dots, |D|$ , where  $|D|$  is the total number of training samples, the  $L_1$  loss function is expressed as follows:

$$L(\theta) = \frac{1}{|D|} \sum_{i=1}^{|D|} \|y_{(i)} - F_{NLGN}(x_{(i)}; \theta)\| \quad (14)$$

Where  $\theta$  denotes the model parameters, and  $F_{NLGN}(\cdot)$  is the mapping function and  $y_{(i)}$  is the HR image corresponding to the input LR image  $x_{(i)}$ .

## 4. Experiments

### 4.1. Dataset and Implementation Details

In this study, we chose the same IXI dataset as that used for experiments using the CSN [8]. The dataset contained MR images of three types: T1-weighted, T2-weighted, and proton density (PD)-weighted images. In addition, two degradation models, bicubic downsampling (BD) and  $k$ -space truncation (TD), were chosen to simulate the LR images.

For convenience, the subdatasets with specific types and degradation levels were expressed by simplified forms. For example,  $T(PD, BD)$  denotes the PD-weighted test set under BD.  $SR \times r$  ( $r = 2, 3, 4$ ) denotes performing  $r$ -fold SR reconstruction. All 3D MR images were processed into  $240 \times 240 \times 96$  images, where 96 represents the number of slices in the 3D volume. During training, we used a random slice (i.e., a single channel image) of the 3D volume. The input of our model was  $48 \times 48$  image patches, and the ground truth (GT) images were the HR image patches corresponding to the LR image patches. These image

patches were randomly extracted from the LR image and the corresponding HR image, as shown in Figure 5.

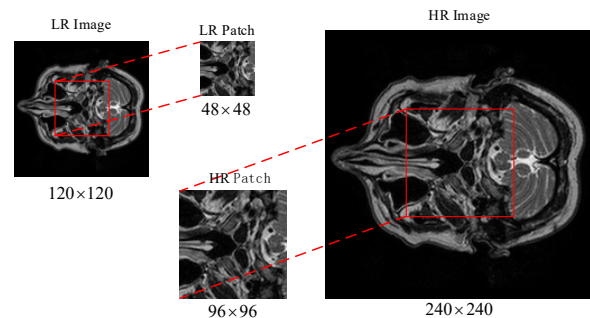


Figure 5. Data example

In the experiments, the batch size was set to 16. The number of modules of the nonlinear mapping network was set to 4 unless otherwise stated. The Xavier approach was chosen for initialization, and the Adam optimizer was selected for minimizing the loss ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and  $\varepsilon = 10^{-8}$ ). The learning rate was initialized to  $10^{-4}$ , and it was reduced by half every  $2 \times 10^5$  iterations. In addition, enhanced measures, such as random horizontal flipping, vertical flipping and  $90^\circ$  rotation, were chosen to ensure the diversification of the training data and to prevent overfitting.

All experiments were conducted with PyTorch 1.50 on a PC with a 2.2 GHz Intel(R) Xeon(R) E5-2650 CPU, 96 GB of RAM and an NVIDIA TITAN V GPU (12 GB of memory). All compared methods mentioned in this study were trained for one million iterations. Finally, it is worth stating that all the experimental data listed in this study were the average results of the IXI test dataset.

### 4.2. Evaluation Metrics

To evaluate the performance of the tested models, the peak SNR (PSNR) and structural similarity index (SSIM) [45] were chosen as evaluation metrics.

The PSNR is defined as:

$$PSNR(x, y) = 10 \cdot \log_{10} \left( \frac{L^2}{MSE} \right) \quad (15)$$

Where  $L$  represents the maximum image pixel value and  $MSE$  denotes the mean squared error between the ground truth (GT) and the reconstructed image.

The SSIM is defined as:

$$SSIM(x, y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (16)$$

Where  $\mu_x$  and  $\mu_y$  represent the average pixel intensities of images  $x$  and  $y$ , respectively, and  $\sigma_x^2$  and  $\sigma_y^2$  represent the variance of the pixel intensities of images  $x$  and  $y$ , respectively. The covariance between images  $x$  and  $y$  is  $\sigma_{xy}$ , and  $c_1$  and  $c_2$  are constants used to prevent instability.

### 4.3. Ablation Experiments

In this section, the performance of the proposed NLGAL was evaluated first. Then, the selection of the number of neighbor nodes  $k$  in the NLGAL was tested. Finally, we tested the impact of the number of NLGMs in the NLGN on the reconstruction performance. The PD-weighted MR images under BD were chosen as the dataset, and  $r=2$  in the experiments.

#### Effectiveness of the NLGAL

To evaluate the effectiveness of the NLGAL, five structures were designed for comparison. a) The proposed NLGAB was not included in the model. b) The proposed NLGAL was not included in the model. c) Figure 1.a was used as the NLGAL. d) Figure 1.b was chosen as the NLGAL. e) Figure 4 was adopted as the NLGAL, and the strategy of utilizing the  $k$  nearest neighbor nodes was not considered. For convenience, the five structures were termed BL (Base line), BL+ (Base line+), BL+NL (Base line+NL), BL+GAL (Base line+GAL), and NLGN-. The experimental results are shown in Table 1.

Table 1. Experimental results yielded by the five structures

	BL	BL+	BL+NL	BL+GAL	NLGN-
PSNR	41.3268	41.3223	40.3186	41.3379	<b>41.3589</b>
SSIM	0.989594	0.989586	0.983282	0.989641	<b>0.989662</b>

It is apparent from Table 1 that NLGN- produced the best reconstruction results. Compared with BL+NL, it achieved approximately 1.04 dB and 0.0034 improvements in the PSNR and SSIM metrics, respectively. In addition, the results also show that BL and BL+ performed better than BL+NL. This reveals that the fusion of information from all neighborhood nodes is not the best choice for reconstruction. This is the main reason why we proposed selecting the top  $k$  nearest neighbor nodes for fusion.

In addition, the reason why the reconstruction result of BL+GAL is stronger than BL+NL is because GAL only aggregates the neighbors of the current node when performing information aggregation, while NL aggregates information for all nodes. The NLGN reconstruction effect proposed in this study is better than that of BL+GAL, mainly because the NLGAL can learn more comprehensive nonlocal self-similarity information than the GAL.

#### Selection of $k$

To verify the impact of  $k$  on the reconstruction results, we designed two groups of experiments. We first set  $k=5, 10$  and  $15$  for comparison. Then,  $k$  was further determined according to the previously obtained experimental results.

Table 2 provides the experimental comparison of the results obtained with  $k=5, 10, 15$  and NLGN-. It can be seen from Table 2 that both the PSNR and SSIM exhibited decreasing trends with the increase in  $k$ . The best scores

were achieved when  $k=5$ , and these results were approximately 0.0178 dB greater than those of NLGN- in terms of the PSNR metric. The experimental results prove the effectiveness of the proposed strategy of selecting only the  $k$  nearest neighbors for fusion. However, when  $k=15$ , the reconstruction effect is not as good as when using this strategy. This is mainly because the selection of the first  $k$  nodes will increase the weight of these nodes when they are aggregated.

Table 2. Experimental results of  $k=5, 10, 15$ , and NLGN-

	$k=5$	$k=10$	$k=15$	NLGN-
PSNR	<b>41.3767</b>	41.3624	41.3438	41.3589
SSIM	<b>0.989686</b>	0.989591	0.989463	0.989662

To further select the value of  $k$ , we set  $k=5, 6, 7, 8$ , and  $9$  according to the results in Table 2. The experimental results are given in Table 3.

Table 3. Experimental results of  $k=5, 6, 7, 8$ , and  $9$

	$k=5$	$k=6$	$k=7$	$k=8$	$k=9$
PSNR	41.3767	41.3789	41.3846	<b>41.3865</b>	41.3729
SSIM	0.989686	0.989691	0.989697	<b>0.989700</b>	0.98969

As seen in Table 3, the proposed model achieved the best scores with  $k=8$ . In addition, in the range of 5-10, both PSNR and SSIM show a trend of rising first and then falling. This phenomenon can be observed more intuitively from Figure 6. Therefore, we chose  $k=8$  in the subsequent experiments.

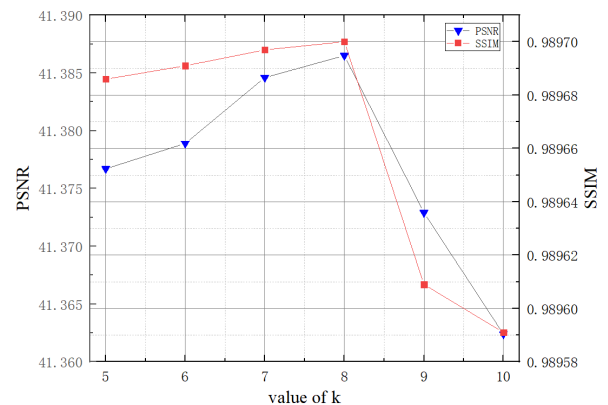


Figure 6. Results of PSNR and SSIM with increasing  $k$

In addition, it can be seen from the results of the above two sets of experiments that as the value of  $k$  increases, the reconstruction effect will improve because the information from more neighbor nodes can be aggregated. This is also

the reason for the improvement of the evaluation index from  $k=5$  to  $k=8$ . On the one hand, as  $k$  continues to increase, the information of the nearest neighbor node aggregated by the current node is reduced; on the other hand, the current node starts to aggregate the information of neighbor nodes that are relatively far away, and the current node begins to aggregate unfavorable information. Therefore, as  $k$  increases, the objective indicator shows a downward trend, which is why the result of  $k=15$  is lower than when this strategy is not used.

### Selection of the Number of NLGMs

In this subsection, we evaluated the impact of the number of NLGMs on the NLGN. In the experiments, the number of NLGMs was separately set to 2, 3, 4 and 5.

Figure 7 displays the PSNR comparison curves with respect to the number of iterations on the  $T(PD, BD)$  dataset and  $SR \times 2$ . Figure 8 shows the parameter comparisons. It is clear that the reconstruction performance improved gradually with the increase in the number of NLGMs. However, this led to a large increase in the number of parameters. Combining these findings, we chose 4 NLGMs in the following experiments.

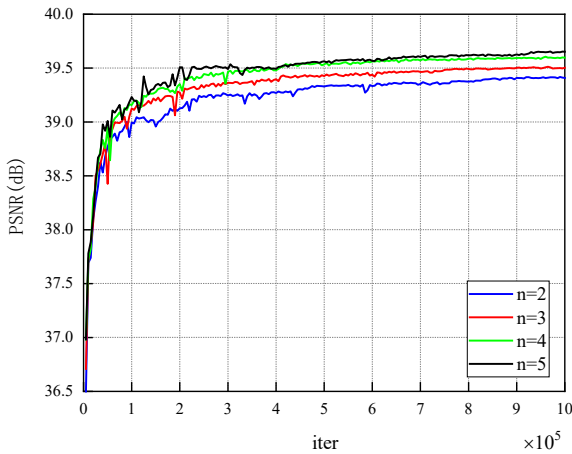


Figure 7. Impact of the number of NLGMs

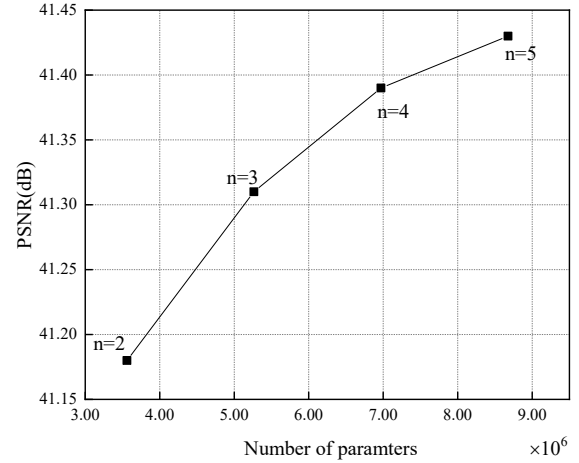


Figure 8. Comparison regarding the numbers of parameters

### 4.4. Comparison with the State-of-the-art Methods

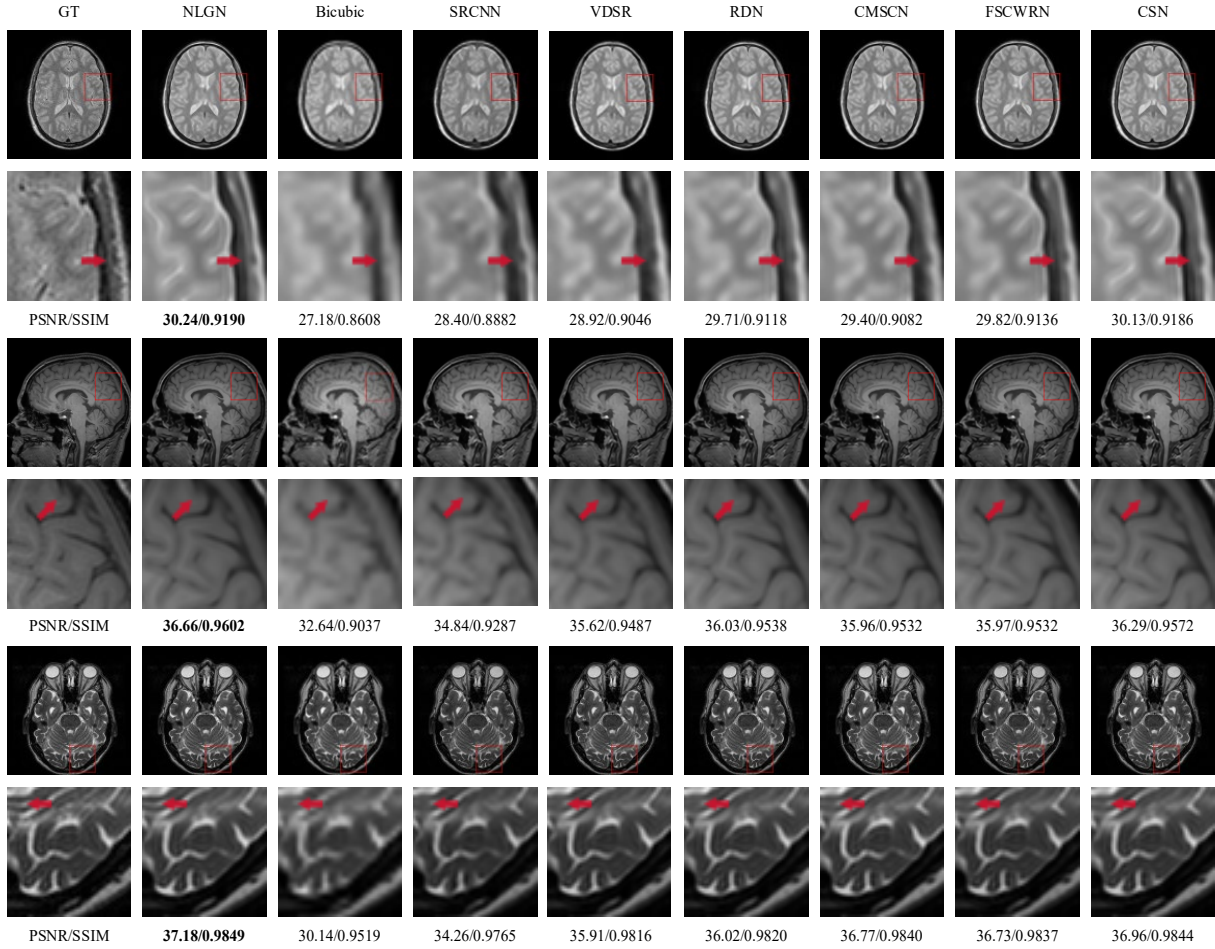
To further prove the effectiveness of the proposed NLGN, it was compared with state-of-the-art methods, including bicubic [9], the SRCNN [25], the VDSR [26], the RDN [31], the CMSCN [29], the FSCWRN [7] and the CSN [6].

Bicubic downsampling, which reduces the HR images to LR images with a bicubic kernel, is a method of simulating LR images that is widely used in the SR field. In this section, we first conduct experiments on the LR images generated by bicubic downsampling. Table 4 gives the PSNR and SSIM comparisons for the abovementioned methods on different dataset types and scales via BD. It can be seen that the proposed NLGN achieved the best scores. According to NLGN fully captured and utilizes information from the image itself. Compared to the comparative methods, the proposed NLGN achieves different improvements on other scales and datasets. For a scale of  $r=2$ , the proposed NLGN yielded improvements of 0.27 dB on the  $T(TI, BD)$  database over the results of the CSN [6].

Table 4. Experimental comparison of the bicubic downsampling datasets. The maximal PSNR and SSIM of each subdataset are marked in bold.

scale	dataset	Bicubic[9]	SRCNN[25]	VDSR[26]	RDN[31]	CMSCN[29]	FSCWRN[7]	CSN[6]	NLGN(ours)
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
$\times 2$	PD	35.04/0.9964	38.96/0.9836	39.97/0.9861	40.31/0.9870	40.84/0.9883	40.72/0.9880	41.28/0.9895	<b>41.39/0.9897</b>
	T1	33.80/0.9525	37.12/0.9761	37.67/0.9783	37.95/0.9795	38.06/0.9800	37.98/0.9797	38.27/0.9810	<b>38.54/0.9818</b>
	T2	33.44/0.9589	37.32/0.9796	38.65/0.9836	38.75/0.9838	39.54/0.9857	39.44/0.9855	39.71/0.9863	<b>39.88/0.9866</b>
$\times 3$	PD	31.20/0.9230	33.60/0.9516	34.66/0.9599	35.08/0.9628	35.26/0.9641	35.37/0.9653	35.87/0.9693	<b>35.99/0.9700</b>
	T1	30.15/0.8900	32.17/0.9276	32.91/0.9378	33.31/0.9430	33.25/0.9423	33.24/0.9423	33.53/0.9464	<b>33.90/0.9497</b>
	T2	29.80/0.9093	32.20/0.9440	33.47/0.9559	33.91/0.9591	34.16/0.9613	34.27/0.9618	34.64/0.9647	<b>34.76/0.9649</b>
$\times 4$	PD	29.13/0.8799	31.10/0.9181	32.09/0.9311	32.73/0.9387	32.53/0.9374	32.91/0.9415	33.40/0.9486	<b>33.49/0.9492</b>
	T1	28.28/0.8312	29.90/0.8796	30.57/0.8932	31.05/0.9042	30.83/0.8997	30.96/0.9022	31.23/0.9093	<b>31.63/0.9153</b>
	T2	27.86/0.8611	29.69/0.9052	30.79/0.9240	31.45/0.9324	31.32/0.9312	31.71/0.9359	32.05/0.9413	<b>32.14/0.9415</b>





**Figure 9.** Visual effects under bicubic degradation. The maximal PSNR (dB) and SSIM values for each displayed image are in bold.

Figure 9 displays the visual effects of the compared methods. The first line shows the results obtained on the  $T(PD, BD)$  dataset with  $SR \times 4$ . It can be seen from the content indicated by the red arrow that in terms of details, the NLGN is closer to the GT image than the other methods. The middle line contains the visual effects produced on the  $T(T1, BD)$  dataset with  $SR \times 3$ . The bottom images are the visual effects obtained on the  $T(T2, BD)$  dataset with  $SR \times 2$ . As a whole, the images reconstructed by the NLGN contain more texture details and shapes than those yielded by the other tested methods.

The k-space truncation of the HR image is a process that simulates the actual image acquisition process, in which the LR image is scanned by reducing the acquisition line in the phase and slice encoding direction [6]. When the scaling factor is the same, the LR image generated by truncation degradation will lose more extensive information than the

LR image generated by BD. This can be demonstrated by the fact that bicubic interpolation works better in bicubic downsampling than in truncated degradation. Table 5 shows the PSNR and SSIM comparisons for the mentioned methods on different dataset types and scales via truncation degradation. It can also be seen that the proposed NLGN achieved the best scores in our experiments because NLGN uses graph convolution to fully capture nonlocal self-similarity information.

For a scale of  $r = 4$ , the proposed NLGN yielded improvements of 0.12 dB on the  $T(PD, TD)$  database over the results of the CSN [6]. In addition, the effect of the NLGN model in truncation degradation is better than that in bicubic downsampling, which means that the proposed NLGN is more suitable for MR image superresolution.

Table 5. Experimental comparison on truncated degradation datasets. The maximal PSNR and SSIM of each subdataset are marked in bold.

scale	dataset	Bicubic[13]	SRCNN[29]	VDSR[30]	RDN[35]	CMSCN[33]	FSCWRN[9]	CSN[8]	NLGN(ours)
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
×2	PD	34.65/0.9625	38.23/0.9802	39.89/0.9850	40.39/0.9862	41.14/0.9882	40.91/0.9876	41.77/0.9897	<b>41.86/0.9898</b>
	T1	33.38/0.9460	36.52/0.9705	37.58/0.9760	38.08/0.9784	38.23/0.9795	38.04/0.9786	38.62/0.9813	<b>38.79/0.9817</b>
	T2	33.06/0.9541	37.04/0.9773	38.74/0.9823	40.02/0.9826	39.63/0.9845	39.82/0.9851	40.47/0.9868	<b>40.56/0.9870</b>
×3	PD	30.88/0.9167	32.90/0.9432	34.27/0.9555	35.00/0.9609	35.41/0.9638	35.30/0.9636	36.09/0.9697	<b>36.19/0.9699</b>
	T1	29.79/0.8793	31.72/0.9187	32.57/0.9304	33.33/0.9416	33.18/0.9398	33.09/0.9390	33.68/0.9464	<b>33.99/0.9490</b>
	T2	29.50/0.9016	31.80/0.9381	33.23/0.9515	33.99/0.9576	34.45/0.9611	34.34/0.9603	34.95/0.9653	<b>35.06/0.9655</b>
×4	PD	28.82/0.8713	30.52/0.9078	31.69/0.9244	32.64/0.9362	32.23/0.9321	32.78/0.9387	33.51/0.9489	<b>33.63/0.9490</b>
	T1	27.96/0.8182	29.31/0.8616	30.14/0.8818	31.00/0.9018	30.55/0.8920	30.79/0.8973	31.27/0.9092	<b>31.72/0.9148</b>
	T2	27.60/0.8511	29.32/0.8960	30.51/0.9162	31.49/0.9301	31.28/0.9278	31.71/0.9334	32.28/0.9421	<b>32.41/0.9424</b>

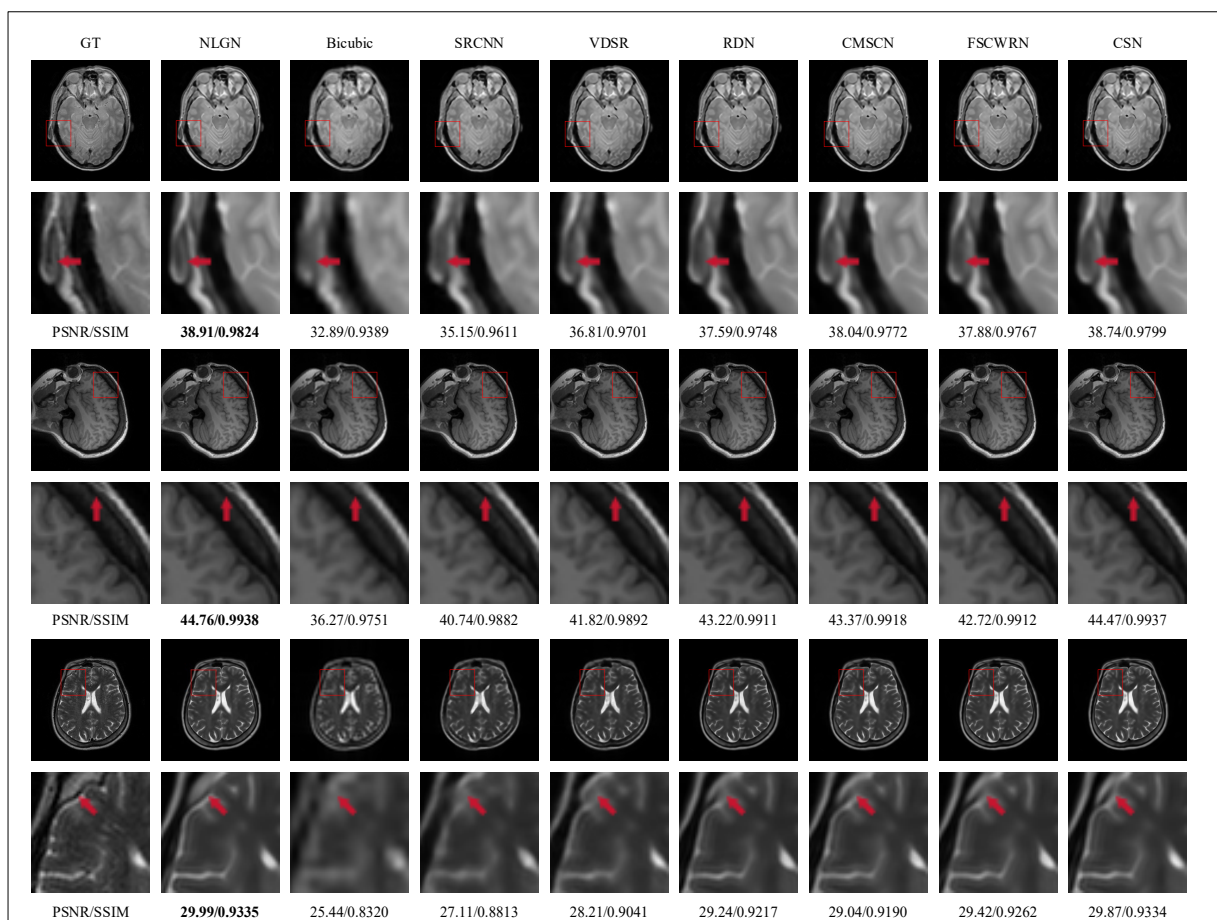


Figure 10. Visual effects under truncation degradation. The maximal PSNR (dB) and SSIM values for each displayed image are in bold.

Figure 10 shows the visual effects of the compared methods. The top images show the visual effects obtained on the  $T(PD, TD)$  dataset with a scaling factor of  $SR \times 3$ . The middle images show the results produced on the  $T(T1, TD)$  dataset with  $SR \times 2$ . The bottom images are the visual effects obtained on the  $T(T2, TD)$  dataset with  $SR \times 4$ . We can reach the same conclusion as that of the experiments

on the BD datasets. In the part indicated by the red arrow in the first row, the proposed NLGN reconstruction content is the closest to the reference image and is better than other methods. The middle row and the last row also show that the reconstruction effect of the NLGN is better than that of the other methods.

We further compared the complexity of the methods discussed in this paper. The comparison results obtained on  $T(PD, BD)$  with  $SR \times 2$  in terms of the number of parameters and the PSNR are shown in Figure 11. It can be seen that NLGN has the best performance, because NLGN captured nonlocal self-similarity information in the image and makes full use of it. The number of parameters in the NLGN model is only 6.97 M, and the number of parameters in the CSN model with similar performance is 13.64 M. The number of parameters of the NLGN is only half of that of the CSN, but it achieves better performance than the CSN. The number of parameters in the RDN model is 22.06 M. Therefore, the NLGN has a smaller number of parameters than the RDN, but its performance is much higher than the RDN. This proves an advantage of the NLGN from another perspective. In addition, the NLGN can achieve higher performance with fewer parameters because graph convolution excavates more nonlocal self-similar information.

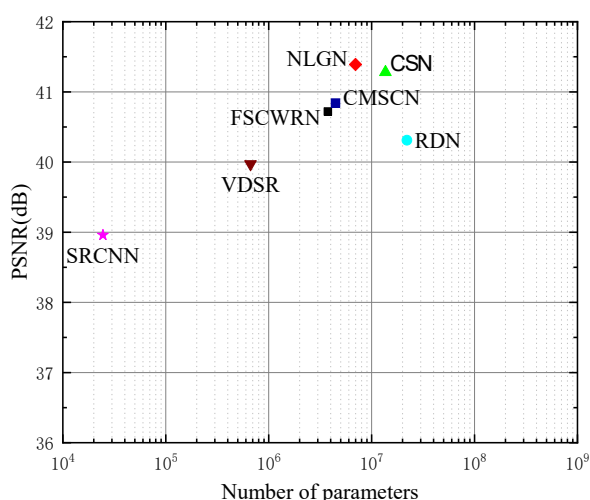


Figure 11. Comparison of parameters and performance

## 5. Conclusion

In this paper, MR image reconstruction based on nonlocal self-similarity was studied in detail. A novel model, an NLGN, was introduced to improve the quality of reconstructed images based on the combination of a graph network and an attention mechanism. First, dense residual blocks were introduced to prevent information loss during the feature transfer procedures. Then, we designed an NLGAL via the combination of NLs and a GAL, which is more effective than other approaches for mining nonlocal self-similarity information. Furthermore, we chose the  $k$  most similar nodes for fusion to reduce the impacts of the neighbors with small similarity values. The experimental results showed that the proposed model yielded the best scores among the state-of-the-art methods. In this study,

we also demonstrated the effectiveness of using graph convolution to mine nonlocal self-similarity information.

Currently, this work only conducts experiments on 2D slices of MR images, ignoring the structural information and implied nonlocal information in 3D volume. MR images are mostly in 3D volumes, which contain more nonlocal information and structural information in 3D space. In future work, we will study how to apply NLGN to MR images in 3D volumes. In this paper, we demonstrated the effectiveness of using graph convolution to mine nonlocal self-similar information and provides some inspiration for the development of other methods for mining nonlocal self-similar information.

## Acknowledgements.

This work was supported by Key Science and Technology Program of Henan Province, China (212102310084). Key Scientific Research Projects of Colleges and Universities in Henan Province, China (22A520027).

## References

- [1] Li, M. A coarse-to-fine face hallucination method by exploiting facial prior knowledge. In C. N. Proceedings. Proceedings of the 25th IEEE International Conference on Image Processing; October 7–10, 2018; Athens, Greece. Piscataway, NJ: IEEE: 2018. pp. 61–65.
- [2] Zhang, Z., Shi, Y., Zhou, X., Kan, H. and Wen, J. Shuffle block srgan for face image super-resolution reconstruction. Measurement and Control. 2020; 53(7-8): 1429–1439.
- [3] Pan, Z., Yu, J., Huang, H., Hu, S., Zhang, A., Ma, H. and Sun, W. (2013) Super-resolution based on compressive sensing and structural self-similarity for remote sensing images. IEEE Transactions on Geoscience and Remote Sensing. 2013; 51(9): 4864–4876.
- [4] Zhang, S., Yuan, Q., Li, J., Sun, J. and Zhang, X. Scene-adaptive remote sensing image super-resolution using a multiscale attention network. IEEE Transactions on Geoscience and Remote Sensing. 2013; 58(7): 4764–4779.
- [5] Hu, J. (2016) Single image super resolution of 3D MRI using local regression and intermodality priors. In Falco, C.M. Proceedings Volume 10033. Proceedings of the Eighth International Conference on Digital Image Processing; 2016; Chengdu, China. US; SPIE;2016. 10033. 866–871.
- [6] Zhao, X., Zhang, Y., Zhang, T. and Zou, X. Channel splitting network for single MR image superresolution. IEEE transactions on image processing. 2019; 28(11): 5649–5662.
- [7] Shi, J., Li, Z., Ying, S., Wang, C., Liu, Q., Zhang, Q. and Yan, P. Mr image super-resolution via wide residual networks with fixed skip connection. IEEE Journal of Biomedical and Health Informatics. 2019; 23(3): 1129–1140.
- [8] Song, T.A., Chowdhury, S.R., Yang, F. and Dutta, J. Super-resolution pet imaging using convolutional neural networks. IEEE Transactions on Computational Imaging. 2020; 6(1): 518–528.
- [9] Keys, R. Cubic convolution interpolation for digital image processing. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1981; 29(6): 1153–1160.

- [10] Medoff, B.P., Brody, W.R., Nassi, M. and Macovski, A. Iterative convolution backprojection algorithms for image reconstruction from limited data. *J. Opt. Soc. Am.* 1983; 73(11): 1493–1500.
- [11] Stark, H. and Oskoui, P. High-resolution image recovery from image-plane arrays, using convex projections. *J. Opt. Soc. Am. A.* 1989; 6(11): 1715–1726.
- [12] Yang, J., Wright, J., Huang, T.S. and Ma, Y. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing.* 2010; 19(11): 2861–2873.
- [13] Zhu, Q. Image superresolution via sparse embedding. In Amir, A. Proceedings. Proceeding of the 11th World Congress on Intelligent Control and Automation; 29 June–4 July 2014; Shengyang, China. Piscataway, NJ: IEEE; 2014. 5673–5676.
- [14] Yang, L., Wang, S.H. and Zhang, Y.D. (2022) Ednc: Ensemble deep neural network for covid-19 recognition. *Tomography.* 2020; 8(2): 869–890.
- [15] Zhou, Q., Zhang, X. and Zhang, Y.D. Ensemble learning with attention-based multiple instance pooling for classification of spt. *IEEE Transactions on Circuits and Systems II: Express Briefs.* 2022; 69(3): 1927–1931.
- [16] Zhang, Y.D., Satapathy, S.C. and Wang, S.H. (2022) Fruit category classification by fractional fourier entropy with rotation angle vector grid and stacked sparse autoencoder. *Expert Systems.* 2022; 39(3), Article: e12701.
- [17] Wang, S.H., Satapathy, S.C., Zhou, Q., Zhang, X. and Zhang, Y.D. Secondary pulmonary tuberculosis identification via pseudo-zernike moment and deep stacked sparse autoencoder. *Journal of Grid Computing.* 2022; 20(1): 1–16.
- [18] Shui-Hua, W., Khan, M.A., Govindaraj, V., Fernandes, S.L., Zhu, Z. and Yu-Dong, Z. Deep rankbased average pooling network for covid-19 recognition. *Computers, Materials, & Continua.* 2022; 70(2): 2797–2813.
- [19] Shui-Hua Wang, Muhammad Attique Khan, Y.D.Z. Vispnn: Vgg-inspired stochastic pooling neural network. *Computers, Materials & Continua.* 2022; 70(2): 3081–3097.
- [20] Redmon, J. Yolo9000: better, faster, stronger. In G. L. Proceedings. Proceedings of the IEEE conference on computer vision and pattern recognition; July 21–26 2017; Honolulu, HI, USA. Piscataway, NJ: IEEE; 2017. pp. 7263–7271.
- [21] Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv: 2004.10934, 2020.
- [22] Lin, T.Y. Feature pyramid networks for object detection. In G. L. Proceedings. Proceedings of the IEEE conference on computer vision and pattern recognition; July 21–26 2017; Honolulu, HI, USA. Piscataway, NJ: IEEE; 2017. pp. 936–944.
- [23] Long, J. Fully convolutional networks for semantic segmentation. In Los Alamitos. Proceedings. Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition; June 7–15 2015; Boston, MA, USA. Piscataway, NJ: IEEE; 2015. pp. 3431–3440.
- [24] Ronneberger, O., Fischer, P. and Brox, T. U-net: Convolutional networks for biomedical image segmentation. arXiv preprint arXiv:1505.04597, 2015.
- [25] Dong, C. Learning a deep convolutional network for image superresolution. In Fleet, D. Proceedings. Proceedings of the 2014 European Conference on Computer Vision; Sep 6–12, 2014; Zurich, Switzerland. 2014 Cham: Springer International Publishing; 2014, pp. 184–199.
- [26] Kim, J., Lee, J.K. and Lee, K.M. Accurate image super-resolution using very deep convolutional networks. arXiv preprint arXiv: 1511.04587, 2015.
- [27] He, K. Deep residual learning for image recognition. In R. B. Proceedings. Proceedings of the IEEE conference on computer vision and pattern recognition; June 27–30, 2016; Las Vegas, NV, USA. Piscataway, NJ: IEEE; 2016. pp. 770–778.
- [28] Lim, B., Son, S., Kim, H., Nah, S. and Lee, K.M. (2017), Enhanced deep residual networks for single image super-resolution. arXiv preprint arXiv: 1707.02921, 2017.
- [29] Hu, Y., Gao, X., Li, J., Huang, Y. and Wang, H. Single image super-resolution with multi-scale information cross-fusion network. *Signal Processing.* 2021; 179: 107831.
- [30] Tai, Y. Image superresolution via deep recursive residual network. In G. L. Proceedings of the IEEE conference on computer vision and pattern recognition; July 21–26 2017; Honolulu, HI, USA. Piscataway, NJ: IEEE; 2017. pp. 2790–2798.
- [31] Zhang, Y. Residual dense network for image super-resolution. In Brown, M. Proceedings. the IEEE conference on computer vision and pattern recognition; Jun 18–22 2018, Salt Lake City, Utah, USA. Piscataway, NJ: IEEE; 2019. pp. 2472–2481.
- [32] Liu D, Wen B, Fan Y, et al. Non-local recurrent network for image restoration. *Advances in neural information processing systems*, 2018, 31: 1680–1689.
- [33] Zhang, Y., Li, K., Li, K., Zhong, B. and Fu, Y. Residual non-local attention networks for image restoration. arXiv preprint arXiv: 1903.10082, 2019
- [34] Dai, T. Second-order attention network for single image superresolution. In Davis L. Proceedings. Proceedings of the IEEE conference on Computer Vision and Pattern Recognition; Jun 16–20, 2019; Long Beach, CA, USA. Piscataway, NJ: IEEE; 2019. pp. 11057–11066.
- [35] Bruna J, Zaremba W, Szlam A, et al. Spectral networks and locally connected networks on graphs[J]. arXiv preprint arXiv:1312.6203, 2013.
- [36] Defferrard, M., Bresson, X. and Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 2016, 29: 3844–3852.
- [37] Kipf, T.N. and Welling, M. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv: 1609.02907, 2019.
- [38] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y. Graph attention networks. arXiv preprint arXiv: 1710.10903, 2017.
- [39] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. and Yu, P.S. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems.* 2021; 32(1): 4–24.
- [40] Chami, I., Abu-El-Hajja, S., Perozzi, B., Ré, C. and Murphy, K. (2021), Machine learning on graphs: A model and comprehensive taxonomy. arXiv preprint arXiv: 2005.03675, 2020.
- [41] Xu, B. and Yin, H. Graph convolutional networks in feature space for image deblurring and super-resolution. arXiv preprint arXiv: 2105.10465, 2105.
- [42] Yan, Y., Ren, W., Hu, X., Li, K., Shen, H. and Cao, X. SRGAT: Single image super-resolution with graph attention network. *IEEE Transactions on Image Processing.* 2021; 30: 4905–4918.
- [43] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D. et al. Real-time single image and video super-resolution using an efficient subpixel

convolutional neural network. arXiv preprint arXiv:1609.05158, 2016.

- [44] Anwar, S. and Barnes, N. Densely residual laplacian super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022; 44(3): 1192–11204.
- [45] Wang, Z., Bovik, A., Sheikh, H. and Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*. 2004; 13(4): 600–612.