.

IEEE *Access*

Multidisciplinary : Rapid Review : Open Access Journal

Self-organized maps and high-frequency image detail for MRI image enhancement

KHODABAKHSH AHMADIAN[1], HAMID-REZA REZA-ALIKHANI[2].

[1]Department of Electrical and Computer Engineering, Mahshahr Branch, Islamic Azad University, Mahshahr, 6351977439, Iran (e-mail: kh.ahmadian@mhriau.ac.ir)

[2]Department of Electrical and Computer Engineering, Tafresh Branch, National University, Tafresh,39518-79611, Iran (e-mail: Alikhani.hamid@aut.ac.ir)

Corresponding author: Hamid-reza Reza-Alikhani (e-mail: Alikhani.hamid@aut.ac.ir).

ABSTRACT Magnetic resonance imaging (MRI) is a medical technology that uses powerful magnets, radio waves, and a computer to produce images of the body's internal organs. The patient should be quiet and motionless during the scanning period, as unavoidable movements, such as breathing and heart rate, cause motion artifacts in the image, which cause contrast instability and low-resolution MRI images. Imaging in the clinical setting is performed at low resolution because the scanning time for high-resolution MR imaging would be very long and cumbersome, and imaging is also very expensive. Learning-based image superresolution methods can reconstruct MR images with optimal quality and resolution. However, these methods have problems such as the inability to find the intrinsic relationship between low- and high-resolution image patches in the training dictionary, specification of the proper amount of error in the training and testing stage due to variability in MR image contrast, inability to reconstruct objects by smoothed edges, and use of the backpropagation method in updating their weight vectors. In this paper, we propose a new superresolution method according to competitive learning-based approaches to overcome the problems of previous superresolution methods, which do not have the problems and complexities of those methods. The proposed method includes self-organizing maps as a preprocessor, the nearest neighbor algorithm as a classifier, and a high-frequency filter as a high-frequency image detail extractor. We constructed a single external dictionary from a combination of low-resolution and high-resolution feature patches and trained our SOM network. Next, we reconstruct the high-resolution image by converting the low-resolution input image into feature patch vectors, and for each vector, we find all corresponding neurons in the network and retrieve all their training feature vectors. Next, we train the nearest neighbor algorithm with the recovered vectors plus the input vector and find the best similarity vector to the input vector. After finding all the best similarity vectors to the input vectors, we reconstruct our high superresolution image. The proposed image superresolution method in practical experiments was trained, tested, and evaluated by the Div2k dataset and compared with other traditional and state-of-the-art image enhancement methods on various datasets, such as SET5, SET14, BSDS100, and URBAN100, and presented better results with higher accuracy and quality than traditional and state-of-the-art methods, both visually compared to each other by human and computational benchmarks, such as the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM), to compare image superresolution algorithms. This method is best for reconstructing high-resolution magnetic resonance images that require high-frequency details and sharp edges with a smooth slope of the imaging objects in their structures. The execution time of the proposed algorithm is slower than that of the other algorithms, so we use GPU hardware and parallel programming techniques to increase the algorithm speed.

INDEX TERMS Image Enhancement, MRI, Single-Image Superresolution, Self-organizing Maps, K-nearest Neighbor, Medical Imaging

I. INTRODUCTION

MAGNETIC resonance imaging (MRI) is a noninvasive diagnostic test that takes detailed images of the body's soft tissues. Images are generated using a magnetic field, radio waves, and a computer, unlike X-rays or CT scans. This test allows physicians to view the brain or spine as sliced layers and take a live picture of the target layer. They use these images and analyze them to identify tumors, strokes, and spinal discs [1]. MRI technology provides different information than other medical imaging techniques. The main advantages of MRI are that it can characterize and distinguish tissues based on their biochemical and physical properties, such as water, fat, iron, and extravascular blood, and their degradation products [2]. Imaging is performed at a low resolution because the scan time for high-resolution MRI images would be very long and impractical, and imaging is also expensive [3]. However, MRI systems cannot achieve sufficient sampling density in their images due to acquisition constraints, such as limited acquisition time or unavoidable movements, such as breathing and heart rate, which cause motion artifacts in the pictures. There are several different approaches to increasing the overall resolution of MRI scans. Hardware enhancements directly increase the resolution of acquired images, for example, by increasing the number of coil receiver channels, increasing the main magnetic field passing through the MRI core, and increasing the MRI signal [2], [4]. These methods are very efficient in reconstructing MRI images with good quality and fast scan time in clinical applications, but the upgrade of MRI hardware is physics-based, which leads to a long cycle time to realize its technological innovation and has a very high cost. Independent of hardware enhancements, postacquisition image processing techniques, such as image superresolution, are also used to increase the resolution of MRI images. They have the advantage that they can be applied on all MRI machinery and in many practical applications without the need to purchase new hardware equipment. The main advantage of superresolution imaging methods based on soft image processing techniques is their ability to reconstruct high signal-to-noise ratios and high-resolution object representations, which, due to object motion, limited sampling time, or signal-to-noise ratio considerations, can only be achieved at low resolution. Image superresolution is an IL-posed problem because many high-resolution(HR) images can produce the same low-resolution(LR) image. This problem can be solved by assuming that the high-frequency information and details of neighboring pixels are enough to recover the high-frequency information of the lost pixel by downsampling [1]. In recent years, several image superresolution algorithms have been introduced to improve the quality and resolution of natural and medical images [5]. These existing image superresolution methods can be classified into three main categories: interpolation-based methods, reconstruction-based methods, and learning-based methods [6]. Interpolation-based methods are usually based on signal processing methods. They explore the relationship of pixel values within a single-image
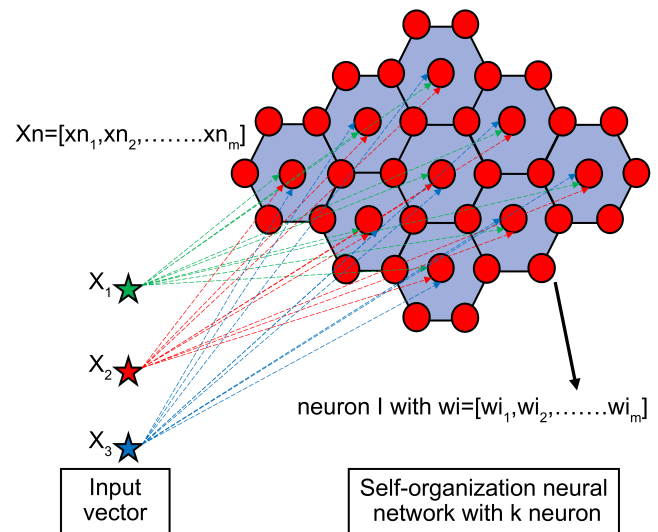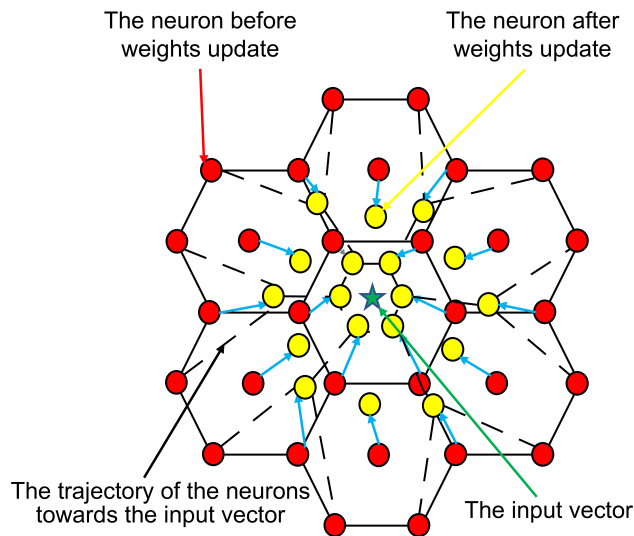

FIGURE 1. The self-organization map neural network structure.

neighborhood and then predict subpixels based on these selected pixels. These superresolution imaging methods are very fast and do not have much computational complexity [7]–[9]. However, these traditional methods often result in excessive smoothing and do not introduce more high-frequency and edge information into the new image. They usually produce images with various effects along object boundaries, such as aliasing, blurring, and zigzag edges [10]. Reconstruction-based algorithms reconstruct high-resolution images by modeling the image creation sequence and usually create a linear system for image reconstruction [11], [12]. These methods are better than interpolation techniques in many image enhancement applications, and their performance for reconstructing a high-resolution image is good. However, the computational cost of this approach is very high, the convergence time is slow, the real-time efficiency is not high, and the image quality decreases as the scale factor increases. In such cases, the result may be too smooth with a lack of significant detail at high frequencies. Example-based methods use the relationships between low-resolution image patches and high-resolution image patches as prior superresolution information. These methods can restore high-resolution images with sharp edges with reasonable computational cost by learning the statistical relationships between the low-resolution and high-resolution image patches from a large training dataset [13]. In 2000, Becker and Kanade [14] and Freeman and Pasztor [13] separately recommended image enhancement algorithms based on patch dictionaries.

In the example-based method, the main idea is to learn the corresponding relationship between low-resolution (LR) patches and high-resolution (HR) patches in a dataset and then select the most suitable HR patches in the dictionary for each input patch (LR) as a result of the reconstruction. Freeman and Pasztor used their proposed method and split an entire image into multiple overlapping image patches. After

The neuron before weights update

The neuron after weights update

The trajectory of the neurons towards the input vector

The input vector

FIGURE 2. Trajectory diagram of the winning neuron trajectory and its neighborhoods to the input vector. (The solid black lines show the neighborhood distance before the weight vector update, and the dashed black lines show the neighborhood distance after the weight vector update).

that, they searched the dictionary for the low-resolution image patches that most nearly matched the low-resolution input patch. Next, they used the corresponding high-resolution patches to construct a Markov random field. Next, they obtained the high-resolution image by solving the Markov random fields. In 2004, Chang, Yeung, and Xiong [15] proposed the neighbor embedding method for image enhancement, in which each input low-resolution image patch can be represented by a linear combination of a series of image patches within a dictionary neighborhood. This relationship is mapped to the high-resolution space to reconstruct a high-resolution image. Example-based superresolution methods usually work well on low-resolution images containing duplicate features and textures; as such, they lack the richness of image structures outside of the input image structure and, therefore, cannot produce a successful representation for other class images [16]. In 2010, Yang and his team [17] proposed a new image superresolution method to improve image resolution and called it the sparse dictionary learning method. The sparse-based method learns an overcomplete dictionary containing only a small number of atoms. This technique identifies the linear combination of the low-resolution dictionary atoms to represent each input patch. Additionally, the high-resolution patch can be reconstructed using the corresponding linear combination of atoms from the high-resolution atom dictionary. This approach is very useful for image reconstruction, and their learning dictionary atoms are concise, but the extracted features and mapping functions are not adaptive, which may not be optimal for generating high-resolution images [18], [19]. Optimizing this method was performed by Zeyde and his group [20] in 2010. It includes training the low-resolution dictionary using the K-SVD method, reduces the computational complexity by

using the principal component analysis algorithm, and solves the sparse vector quickly by orthogonal matching analysis. Li-Wei Kang and his group [21] optimized their method in 2015. They suggested learning sparse image representations to model the relationship between low- and high-resolution image patches in terms of the dictionaries learned for image patches with and without blocking artifacts simultaneously. In the new optimization method, image enhancement and deblocking were achieved in simultaneous sparse representation and morphology-based image decomposition. Yang and his research team [22] proposed a new image superresolution method based on redundant compact dictionaries learned from classified samples by K-means clustering to provide each sample with a more appropriate dictionary for image reconstruction. Yang, in 2012, optimized the proposed method by clustering the patterns and training multiple dictionaries according to the geometric similarity of the training patterns and then selecting a dictionary that has a similar geometric structure to the input low-resolution patch to reconstruct the high-resolution image patch [23]. Yang, in 2018, proposed an improved fuzzy clustering and weighted schema reconstruction framework to solve a variety of image structures. In this method, training patches are divided into multiple clusters by jointly learning multiple dictionary pairs with an improved fuzzy clustering method to minimize the error function. Then, high-resolution patches are estimated according to several more accurate dictionary pairs [24]. These methods have better performance than previous image superresolution methods. However, these methods cannot find the intrinsic relationship of low-resolution and high-resolution patches, and they cannot reconstruct the edges and borders of objects in an image with a smooth slope. Artificial neural network-based image superresolution approaches were proposed by LeCun and his group in 1989. LeCun's image superresolution method [25] involves a single network that learns the entire recognition operation, from the normalized image of the character to the final classification. This method uses a backpropagation algorithm [26] to train its weight vectors on large image databases. It has been successfully applied to recognizing digits in handwritten zip codes by the U.S. Postal Service. Dong and his team [27] proposed a deep learning method for single-image superresolution in 2014. Their image enhancement method directly learns an end-to-end mapping between the low-resolution and high-resolution images. The mapping is represented as a deep convolutional neural network that takes the low-resolution image as input and produces a high-resolution image as output. This paper attempts to find the intrinsic relationship of low- and high-resolution patches, which in the previous algorithms were inaccessible, and to overcome the dictionary training methods according to the statistical calculation, which has uncertainty in its calculations. Furthermore, to overcome the error criteria-based training methods in the neural network-based image superresolution methods that use backpropagation methods to train their weight vectors, we proposed the new image superresolution method according to the competitive

approaches that do not have the problems and complexities of the previous methods.

A. MOTIVATIONS AND CONTRIBUTIONS

The single-image superresolution (SISR) method is one of the most controversial image processing techniques that attempts to acquire high-resolution (HR) images from low-resolution (LR) images by estimating and approximating all missing high-frequency details such as magnetic resonance images, which need high-frequency details and sharp edges with a smooth slope of the image objects in their structures. In this paper, we propose a hybrid single-image superresolution to enhance magnetic resonance imaging (MRI) details by using self-organizing neural maps [28] as a preprocessor, the nearest neighbor algorithm [29] as a classifier and a high-frequency filter as a high-frequency image detail extractor. The proposed image superresolution method uses a single external dictionary to train and reconstruct the high-resolution image. Although our proposed superresolution method is more robust and accurate than most of the previous learning-based methods, since the neural weight vectors are automatically adjusted, there is also no need to perform statistical calculations or manually adjust the error correction of the proposed algorithm parameters. Our method uses the nonparametric k-nearest neighbor algorithm for classification, which classifies feature patch vectors according to their similarity. The edges of the high-resolution image objects are sharpened by a gentle slope, which results in an enhancement of high-frequency image details. We increase the performance and execution speed of the proposed algorithm using CPU-GPU hardware technology and parallel programming. We use CPU hardware and serial programming to create the external dictionary, extract the feature patch vectors and reconstruct the high-resolution output image. In addition, we use GPU hardware and parallel programming to train the self-organizing maps and nearest neighbor algorithm and locate the high-resolution patch vector in the reconstruction stage. The main contributions of this work are as follows:

- The unique external dictionary is built using high- and low-resolution feature patch vectors from low- and high-resolution MRI images.
- We construct our self-organizing neural network and define its network parameters and initialize weight vectors by values close to zero. Next, we train our neural network by an external dictionary and store the neuron label and the content of each neuron according to the vector labels in another dictionary.
- We use the low-resolution input MRI image and make it into feature patch vectors, and for each feature patch vector, we locate the best matching neuron in the neural network and retrieve all the feature vectors pertaining to it from the training stage and all these vectors with the input test feature vector used to train the k-nearest neighbor algorithm and locate the best matching vector, which is very similar to the input vector, and retrieve the high-resolution patch from the dictionary. Then, after

locating all the high-resolution patches, we reconstruct our high-resolution image.

- We simulate our proposed superresolution method with medical MRI image datasets and ordinary ground-truth datasets and compare the results with other traditional and state-of-the-art image superresolution algorithms for different upscaling factors.
- We use CPU-GPU hardware technology and parallel programming to execute the proposed algorithm and evaluate the performance of GPU hardware technology and parallel programming for increasing the speed and accuracy of the image superresolution algorithms.

The remaining chapters of the paper are organized as follows. Section 2 describes the structure of the image model, the self-organizing neural network, and the k-nearest algorithm. Section 3 describes the proposed image superresolution method. Section 4 shows the experimental results on various ground-truth natural and MRI image data datasets to examine the performance of the proposed method. Section 5 is the discussion, and finally, Section 6 is the conclusion.

II. RELATED WORKS

A. FORWARD IMAGING SYSTEM MODEL

Modeling the imaging system accurately under mathematical equations is an excellent step toward superresolution and image recognition. The low-resolution image is the result of the warping, blurring, and subsampling performed on the high-resolution image x during the acquisition process. By considering the above effects on modeling the imaging system, the imaging system [30] can be represented by equation (1).

$$y = DBMx + n \qquad (1)$$

The low-resolution image $y \in R^{\mathrm{N}}$ is a blurred, warped, and subsampled version of the high-resolution image $x \in R^{\mathrm{N}}$, where B, W and D represent the blur, warping, and subsampling matrices, respectively. N is the number of images, and n represents noise. The image superresolution reconstruction approach can be an ill-posed problem due to ill-conditioned defocus operators. Regularization is the procedure adopted to stabilize the inverse of the ill-posed problem. It can be achieved by applying prior knowledge about the solution. There are two types of approaches: the deterministic approach and the stochastic approach. We can formulate the problem of recovering x from y by solving the following optimization problem by using the maximum a posteriori estimation (MAP) approach:

$$\min \quad \exp\left(-\frac{1}{2\sigma_{\mathrm{k}}^2}\sum_{\mathrm{k}=1}^{\mathrm{N}}(||y - DBMx||)^2\right) \qquad (2)$$

Several of the proposed regularization-based superresolution methods incorporate the regularization term in the maximum estimate. Therefore, the superresolution optimization problem can be overwritten by Equation (3), where (Ax) is the regularization term that contains prior information about the desired high-resolution image x, and the parameter λ
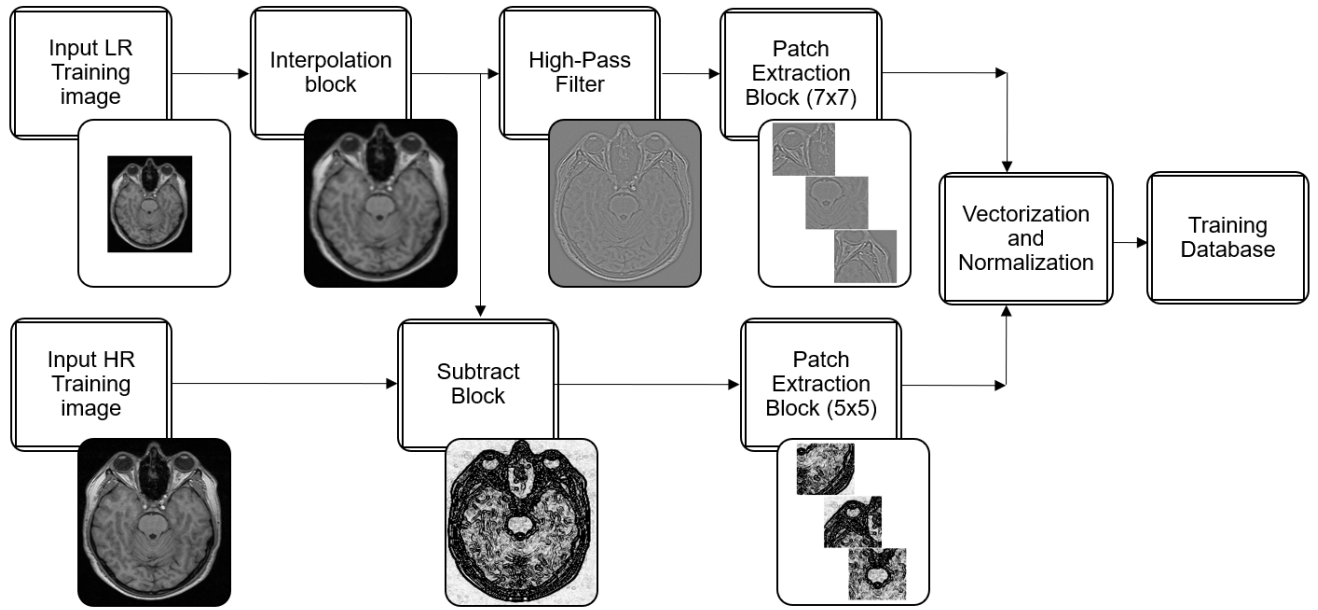
FIGURE 3. Block diagram of the feature extraction and dictionary reconstruction procedure.

is the regularization factor that specifies the weight of the regularization term [31].

$$\min \quad \frac{1}{2}\sum_{k=1}^{N}\left(||y - DBMx||\right)^2 + \lambda Ax \qquad (3)$$

In this paper, we use the first part of the equation of the superresolution optimization problem through equation (4) to reconstruct the high-resolution image without the regularization part and use the 2-norm to calculate the Euclidean distance.

$$\min \quad \frac{1}{2}\sum_{k=1}^{N}\left(||y - DBMx||\right)^2 \qquad (4)$$

B. THE ORGANIZATIONAL ARCHITECTURE OF SELF-ORGANIZING MAPS.

Self-organizing maps (SOMs) or Kohonen maps are a particular class of artificial neural networks first introduced in the 1980s by Teuvo Kohonen. They are unique types of neural networks that use an unsupervised competitive learning-based approach to update their weight vectors. This network does not learn by stochastic gradient descent (SGD) backpropagation like other artificial neural networks but uses competitive learning. A self-organizing map [32] consists of a set of neurons that are formed in a typical two-dimensional lattice. The number of neurons in a neural network varies depending on the application in which the neural network is used. The self-organizing map has two layers: the first is the input layer, and the second is the output layer or feature map. Unlike other kinds of artificial neural networks, the self-organizing map has no activation function for neurons. Instead, the weights are passed directly to the output

layer without any further action. For each neuron in a self-organizing map, the vector of weights has the same dimensionality "d" as the input space [33]. The neurons in a lattice of self-organizing maps are each connected by a connection specified by the topology of the neural network, such as the hexagonal lattice constructions illustrated in Figure 1. As mentioned above, the self-organizing map does not use the backpropagation method together with stochastic gradient descent to update the weights, and this type of unsupervised artificial neural network uses competitive learning to update its weights. The competitive learning method is based on three main approaches: competition, cooperation, and adaptation. In the competition stage, each neuron in the self-organizing map is assigned an initial vector of weights with a value close to zero in the same dimensionality as the input space. In this case, each neuron in the output layer has a vector of weights W=$[w_1, w_2, w_3, ..., w_d]$ with dimensionality "d". In each iteration of the training step, the self-organizing map algorithm calculates the distance between each neuron in the output layer and the input sampling data "X", and the neuron with the smallest distance is the winner of the competition; then, this winning neuron is called the best matching unit (BMU) according to Equation (6). The best matching unit is the neuron in the neural network whose weight vectors are very similar to the input vector X=$[x_1, x_2, x_3, ..., x_d]$. The Euclidean metric is often used to measure the distance of two vectors by the Equation (5).

$$Edistance_{(X,W)} = \sqrt{(x_1 - w_1)^2 + (x_2 - w_2)^2 ... (x_\mathrm{d} - w_\mathrm{d})^2}$$
$$(5)$$

$$WinningNeuron_{(c)} = \arg\min_{i}\{||X - W_\mathrm{i}||\} \qquad (6)$$

IEEE *Access*

During the cooperation stage, when the BMU is successfully located, the self-organizing map algorithm updates the weight vector of the winning neuron during the final adaptation process, but it is not the only one; its neighbors are also updated. We use the neighborhood kernel function to choose the neighboring neurons of the winning neuron in the lattice of the self-organizing map. The neighborhood kernel function depends on the distance between the winning neuron and the other neurons and is inversely proportional: increasing $d_{(}c, i)$ decreases h(t), the neighborhood radius, which in turn depends on time and decreases as time increases, and this causes the neighborhood kernel function to decrease as well. In this paper, we use the Gaussian neighborhood function according to Equation (7).

$$h_{\mathrm{c}(X),i}(t) = \exp\left(-\frac{d^2_{(c,i)}}{2\sigma^2(t)}\right) \qquad (7)$$

Since the number of neighbors of the winning neuron at each stage of the algorithm implementation, as well as the radius of the neighborhood of the winning neuron in time, is reduced, in this case, the value of the standard deviation of the Gaussian function is considered time-varying according to Equation (8).

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) n = 0, 1, 2, \ldots\ldots\ldots max \qquad (8)$$

The parameter τ_1 is the minimum number of iterations of the SOM-neural network in the training stage. Another parameter that plays a vital role in computing the self-organizing neural network algorithm is the learning rate of the neural network. The learning rate determines how many times we tune the neural network weights. The learning rate after time t, which is almost infinitely positive, converges to zero, so there is no update even for the winning neuron. The parameter τ_2 is the maximum number of iterations of the SOM-neural network in the training stage. The learning rate is estimated from Equation (9).

$$\eta(n) = \eta_0 \exp\left(-\frac{n}{\tau_2}\right) \qquad (9)$$

The weights of the winning neuron and all its neighboring neurons are updated according to equation (10). However, their weight vectors are not updated uniformly. The number of the vector of neuron weights farther away from the input vector is smaller. The trajectory of the winning neuron and its neighbors to the input vector is shown in Figure 2.

$$W_{\mathrm{i}}(n+1) = W_{\mathrm{i}}(n) + \eta(n)h_{\mathrm{c}(X),i}(n)(X - W_{\mathrm{i}}(n)) \quad (10)$$

One of the critical parameters in neural network algorithm calculations is the algorithm stopping condition. The closer the stopping condition is to reality, the more accurate the neural network will be in obtaining results. The stopping condition of the self-organizing network is that any change in the vector of weights is zero, i.e., the vector of weights from the previous stage is equal to the vector of weights computed in the next stage. In this paper, we use the stopping condition
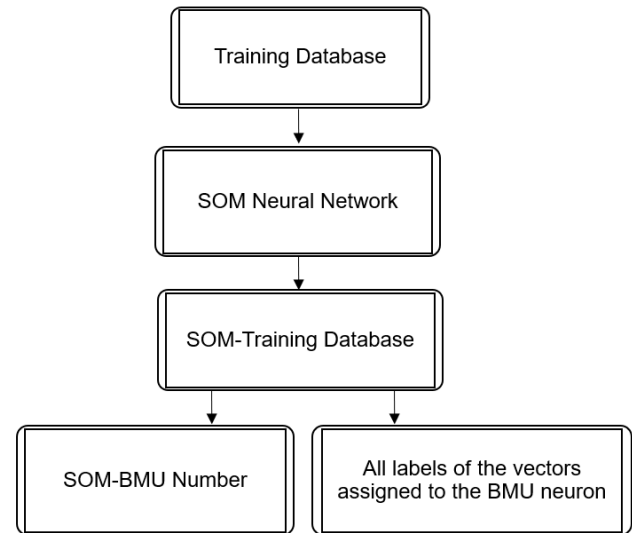

FIGURE 4. The block diagram of the dictionary training procedure.

Algorithm 1 : The training procedure of the self-organizing map

---

Input: X a set of P vectors of d components.
Data: W a set N of vectors with d-dimensions.
1: The weight vectors are all initialized with small values close to zero.
2: Compute $\eta(0)$;
3: Compute $\sigma(0)$;
4: **for** $n \leftarrow 1$ to n_{Max} **do**
5:     Choose $i \in 1\ldots\ldots\ldots P$ randomly;
6:     **for** $k \leftarrow 1$ to N **do**
7:         $d_k \leftarrow 0$;
8:         **for** $j \leftarrow 1$ to P **do**
9:             $d_k \leftarrow d_k + (x_{ij} - w_{kj})$
10:         **end for**
11:         $c \leftarrow arg\ min_k\ d$;
12:         **for** $k \leftarrow 1$ to N **do**
13:             $h_{c,k} \leftarrow \exp\left(-\frac{d^2_{(c,k)}}{2\sigma^2(n)}\right)$;
14:         **for** $k \leftarrow 1$ to N **do**
15:             $W_{\mathrm{k}}(j) = W_{\mathrm{k}}(j) + \eta(n)h_{c,k}(n)(X_{\mathrm{i}}(j) - W_{\mathrm{k}}(j))$;
16:         **end for**
17:         **end for**
18:     **end for**
19:     Decrease the $\eta(n)$ learning rate.
20:     Decrease $\sigma(n)$ neighborhood radius.
21: **end for**

---

of Equation (11) for the training and testing stages of the self-organizing neural network. To train the SOM-neural network, samples were chosen as the feature vector of pattern "X" from the input datasets, and the neural network algorithm started according to Algorithm (1).

$$If \quad h_{c(X),i}(t) \neq 0 \Rightarrow \forall i, E_t\{h_{c,i}(t)(X(t) - W_i(t))\} = 0 \tag{11}$$

$$W^* = \frac{\sum_t h_{c,i}(t)X(t)}{\sum_t h_{c,i}(t)} \tag{12}$$

To test the SOM-neural network, a random sample is selected as the feature vector of the pattern "X" from the input dataset. Then, the distances between it and all the weight vectors of the network are calculated using the Euclidean distance measurement criterion. Then, the unit with the best match is selected using the minimum function, and we use the feature vector of this neuron for classification in the future.

C. THE K-NEAREST NEIGHBOR ALGORITHM STRUCTURE

The nearest neighbor algorithm is a simple and easy-to-implement supervised machine learning approach that may be used to solve classification and regression problems. The supervised machine learning algorithm attempts to learn a function to make predictions from new unlabeled data. Unlike unsupervised learning, the supervised learning method relies on labeled input data to learn a function that produces an appropriate output when new unlabeled data are provided. The k-nearest neighbor algorithm is based on the assumption that similar patterns are close to each other, which means that similar patterns are located near each other. This algorithm involves the concept of similarity, which is sometimes defined in terms of distance, closest similarity, and closest proximity to each other, using mathematics to calculate the distance between points in a data graph. There are many methods for calculating the distance between two points, and depending on the problem we decide to solve, one of them may be preferable. However, the rectilinear distance, also known as the Euclidean distance, is the most popular and well-known option. To choose the correct value of k for our data, we execute the k-nearest neighbor algorithm many times at different values of k and then select the value of k that reduces the number of errors encountered while maintaining the algorithm's ability to make accurate predictions when given data it has not seen before. The "k" nearest neighbor algorithm is executed in accordance with the steps described in Algorithm 2. It is important to note that if we decrease the value of k to one, the predictions we make become less stable, and the k-nearest algorithm predicts patterns incorrectly. However, as we increase the value of k, due to majority voting or averaging, our predictions become more stable, allowing us to make more accurate predictions with higher probability to some extent. As time progresses, we see an increasing number of errors. At this point, we know that the value of k has been exceeded. In some cases where a majority decision is made, such as the choice of mode in a sorting problem between labels, we usually make k an odd number to break the tie. The k-nearest-nearest algorithm is uncomplicated and very easy to set up, and there is no need to build a model, make various adjustments or perform additional calculations. The k-nearest algorithm is

---

Algorithm 2 : The k-nearest neighbors algorithm

Input: Load the pattern data.
Data: Initialization k to the chosen number of neighbors.
1: **for** $q \leftarrow 1$ to $Query_{samples}$ **do**
2:     $d_k \leftarrow 0$;
3:     $indx_k \leftarrow 0$;
4:     **for** $p \leftarrow 1$ to P **do**
5:         Choose $p \in 1.........P$ randomly;
6:         $d_k \leftarrow d_k + (x_p - x_q)$
7:         $indx_k \leftarrow indx_k + 1$
8:     **end for**
9: **end for**
10: Sort the ordered collection of d_k and $indx_k$ from smallest to largest in ascending order by the distances.
11: Select the first k entries from the sorted collection.
12: Get the labels of the selected k entries.
13: Return the mode of the k labels for classification.

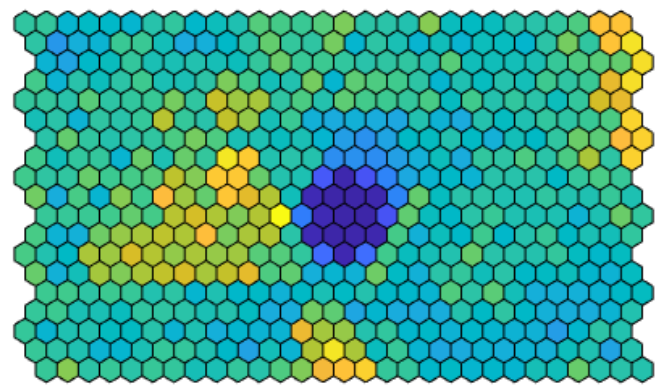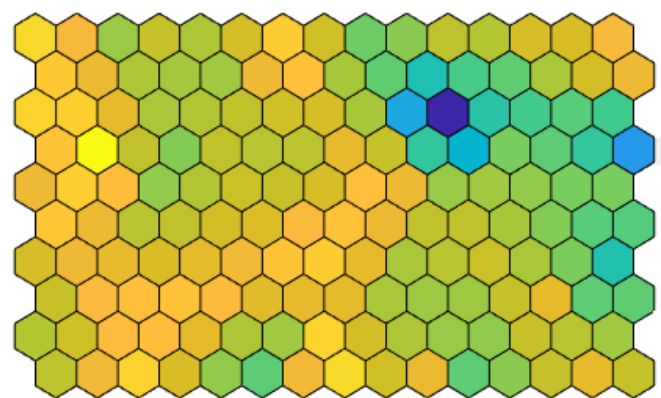**FIGURE 5.** SOM U-matrix.

FIGURE 6. SOM Component lattice diagram.

also multipurpose and is used for data sorting, regression, and search algorithms.
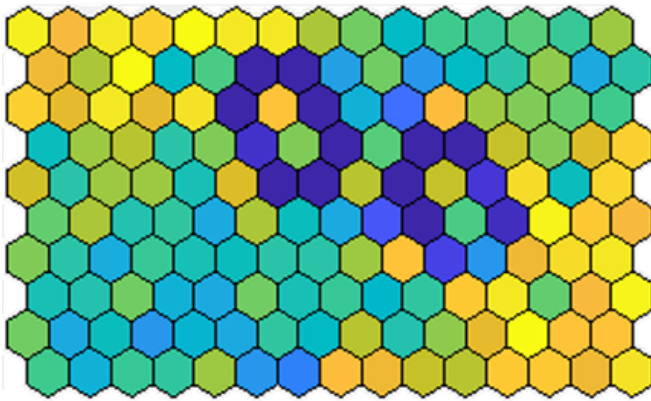
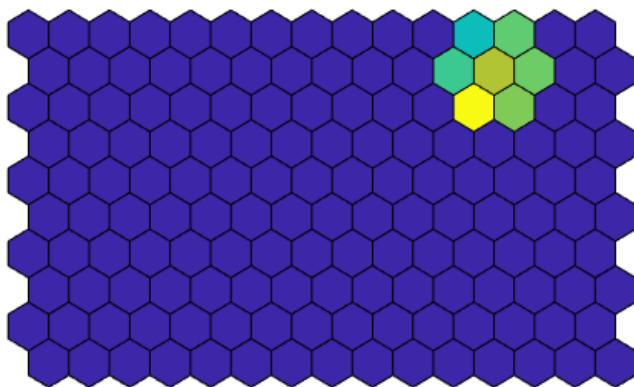**FIGURE 7.** Best matching units (BMU) of SOM in the lattice diagram.

FIGURE 8. Typical best matching units (BMUs) of SOM and their six neighborhoods in lattice diagrams.

III. PROPOSED IMAGE SUPERRESOLUTION METHOD

The proposed image superresolution method for reconstructing high-resolution images uses three stages: feature extraction and dictionary reconstruction, dictionary training, and image reconstruction.

A. FEATURE EXTRACTING AND DICTIONARY RECONSTRUCTION

We create a single overcomplete dictionary $D_{(HL)}$ containing two separate but related fields: the first is the feature patch vector field, which at this point is called a key field, and the second is the high-resolution patch vector, which at this point is called a value field. The procedures for creating an overcomplete dictionary and the associated field values can be seen in Figure 3. Preprocessing procedures for preparing low-resolution images from high-resolution images include several of these steps. In the first step, the best high-resolution images are selected from the high-resolution image dataset, and in the second step, a specific amount of noise is added to each image using a mean kernel to simulate the effects of the probability density function of the imaging system. Finally, the images are converted to low-resolution images with lower quality according to the required magnification factor using downsampling methods. The low-resolution images are indi-

vidually enlarged to their original high-resolution sizes with the appropriate scaling factor using interpolation methods. Then, each low-resolution image is used in two different parts. In one portion, the low-resolution image is subtracted from the high-resolution input image to extract the details of the high-resolution image. The resulting image is used as the high-resolution image to create high-resolution image patches. In the second portion, the low-resolution image is used to extract the "high-frequency details" by passing it through a high-pass filter, and the result is a low-resolution image, which is used to create low-resolution image patches. The image patch extraction procedure is the same for high- and low-resolution images and the only difference is the kernel size. A 5-by-5 extraction kernel is used for low-resolution imaging, and a 7-by-7 extraction kernel is used for high-resolution imaging. The patches are extracted from the high- and low-resolution images in a concentric manner by overlapping the patches in all dimensions with a size of one pixel and a stride length of one pixel. When the high-resolution patches and their corresponding low-resolution patches are each extracted with the above method, the fusion feature vector of both patches, which is based on image feature extraction using image intensity [34], is extracted by converting the contents of the 7x7 matrix of the low-resolution patch into a 49x1 column vector. Next, the contents of the first row and first column of the 5x5 matrix of the high-resolution image intensity patch are converted into a 9x1 column vector, and its contents are added to the end of the previous vector to obtain the 58x1 target feature vector. Next, we remove the low-frequency details that do not provide much information about the image from the constructed vector and only consume the amount of storage memory of the machine, using the normalization of the vector by the mean function. We store all feature vectors as the key fields and the entire contents of the intensity matrix of the high-resolution patches as value fields in the dictionary for the dictionary training stage.

B. THE DICTIONARY TRAINING

We label all input vectors and neurons in the self-organizing neural network in terms of numbers. We reconstruct a two-dimensional self-organizing neural network with the weight vectors proportional to the input vector dimension, as shown in Figure 4. The weight vectors of the self-organizing neural network are initialized with initial and minimum values that approach zero depending on the spacing of the input data distributions. We configure the SOM-neural network using the following parameters: The lattice network is a sheet with a two-dimensional lattice space, the network topology is a hexagon, the neighborhood function of the network is Gaussian, the size of the neurons in the network is two-dimensional [10 15], and it has two training stages. In the first stage, called the coarse stage, the neuronal neighborhood radius function decreases from 3 to approximately 0.5, and the number of training epochs is 50. In the second stage, called the fine stage, the neuronal neighborhood radius func-

tion decreases from 1 to approximately 0.5, and the number of training epochs is 10. We train the SOM-neural network using the vectors from the training database and by Algorithm 3. As shown in the U-matrix in Figure 5, the training data distribution is smoothly distributed across the winning neurons. This matrix shows a lattice of neurons with adjacent distances between them in different colors depending on the distribution of data between neurons. Figure 6 also shows how the input vectors are distributed among the neurons in the neural network. As can be seen in the figure, the higher the number of vectors assigned to a winning neuron, the lower and brighter the color density of that neuron, and the lower the number of neurons assigned to a vector, the thicker and darker the color. Figure 7 shows the distribution of winning neurons in the neural network. In this figure, the regions of the neural network where the distribution of winning neurons is higher are identified and can be used to estimate the efficiency and quality of the neural network training. We then store all the best matching units and their vector content in the second dictionary, which has two fields, the neuron weight vectors, and the input vector labels.

C. THE SUPERRESOLUTION IMAGE RECONSTRUCTION

The low-resolution input image is enlarged to the high-resolution output image by an appropriate enlargement factor through interpolation methods and is used in three different application areas. The first application of the low-resolution image is to extract low-resolution image patches from it using the 7x7 kernel, and then these patches are used to create a final feature vector along with a high-resolution image patch. The second application of the low-resolution image is the extraction of its high-frequency details by high-pass filtering, where the output image from this step is subtracted from the interpolated image, and the resulting image is added to the output image at the end of the image reconstruction algorithm. The third application of this image is at the end of the algorithm to create a high-resolution output image. Since we have no high-resolution image from the low-resolution input image, we create the image with the dimensions of the high-resolution output image with zero pixel intensity values (M, N, d). We convert it into high-resolution patches using a 5x5 patch extractor kernel. When the high-resolution patches and their corresponding low-resolution patches are extracted, the merging feature vector of both patches, which is based on image feature extraction using image intensity, is extracted by converting the contents of the 7x7 matrix of the low-resolution patch into a 49x1 column vector. Next, the contents of the first row and the first column of the 5x5 matrix of the high-resolution image intensity patch are converted into a 9x1 column vector, and its contents are added to the end of the previous vector to obtain the 58x1 target feature vector. Next, we remove from the constructed vector the low-frequency details that do not provide much information about the image and only consume the amount of storage memory in the machine, using the normalization of the vector by

the mean function. We test the SOM-neural network using Algorithm 4 for each experimental input vector to find the best matching unit, then find all the neighbors of the winning neuron, i.e., in this work, six neurons, as shown in Figure 8, and then retrieve the vector labels from the winning neurons plus six neighboring neurons. We use the nearest neighbor algorithm and take the k best training feature vectors that match this test feature vector, where k is 10 in this work. The algorithm sorts them in ascending order according to their distance to the input vector, selects the mode among them, and finds the high-resolution image field that maps to this key. Then, all the high-resolution test patches that best match the input image are taken. We reconstruct the high-resolution image as shown in Figure 9 by merging these three images: the image reconstructed from the patches, the image resulting from the difference of the high-pass filter output image, and finally the interpolation image.

IV. PRACTICAL OUTCOMES

A. PREPARATION OF THE DATABASES

We used different image datasets to train, test, and validate the proposed image superresolution method depending on the application of the proposed image superresolution algorithm, such as for reconstructing natural images or for reconstructing medical images. In the application to reconstruct natural images, we used the natural images from known datasets to train, test, and validate the proposed algorithm. To train, test, and validate the proposed model using natural images, we used the Div2K dataset [35], which contains 1,000 high-quality 2k images divided into 100 images for validation, 800 images for training, and 100 images for testing the proposed model. In the application for reconstructing clinical diffusion-weighted and MRI images, we used 75 diffusion-weighted images and 90 brain MRI images from the brain development organization dataset in which each group of images had different high-frequency details in their image textures. In this paper, fifty images were used for training, 20 images for testing, and 5 images for validating the proposed model from the set of diffusion-weighted images. In addition, 65 images were used for training, 15 images were used for testing, and 10 images were used for validating the proposed model from the set of MRI images. First, by preprocessing the images and converting the high-resolution images to low-resolution images using blurred noise for blurring, the images were rotated under various angles, and the images were subsampled with an appropriate scale factor to create low-resolution DW and MR images. We used complete natural imaging datasets such as SET5 [36], which includes 5 color images, SET14 [20], which includes 14 color images, BSDS100 [37], which includes 100 black and color images, and URBAN100 [5], which includes 100 black and color images, to evaluate the proposed superresolution imaging algorithm against other superresolution imaging methods. We also used a dataset of DW and MRI medical images including several black images of the human brain in the organization of human brain development to compare the

---

Algorithm 3 : The proposed superresolution training stage

Input: A set of HR and LR images.

Output: All BMU units and all vector labels contained in each neuron.

 1: Upsampling of each LR image and passing through a high-pass filter to obtain a medium image (MR).
 2: Calculates the difference between the HR input image and the interpolated image.
 3: Divide HR and LR images into patches.
 4: **for** (*Each patch in the LR training set*) **do**
 5:     Extract each patch feature vector and combine the HR and LR image patch vectors to construct the final vectors.
 6:     Normalize the vectors using the mean intensity of each patch (contrast normalization).
 7:     Store the vector as a value and its HR patch vector as a key in the training database.
 8: **end for**
 9: Train the proposed SOM-neural network with all the vectors of the training database.
10: All the parameters of the SOM, as well as all the best matching units of the network and the data vector labels belonging to those units, are stored.

Algorithm 4 :The proposed image reconstruction stage

**Input:** A testing LR image.

Output: An HR reconstructed image.

 1: Upsample the LR image and pass it through a high-pass filter to obtain a medium image (MR).
 2: Calculates the difference between the MR image and the interpolated image
 3: Divide the HR framework with zero intensity of each pixel and LR images into patches.
 4: **for** (*Each LR patch*) **do**
 5:     Extract each patch feature vector and combine the HR and LR image patch vectors to construct the final vector.
 6:     Normalize the vector using the mean intensity of each patch (contrast normalization).
 7:     Run the SOM-neural network and enter the input vector.
 8:     Obtain the BMU neuron that matches the input vector.
 9:     Obtain six neighboring neurons for the BMU neuron.
10:     Retrieve the training vectors of these seven neurons.
11:     Run the K-nearest neighbor algorithm for this retrieved data and input vector.
12:     Obtain the best training vector that matches the input vector.
13:     Find the high-resolution training patch that belongs to this vector label and place it in its location on the high-resolution output image.
14: **end for**
15: Reconstruct a high-resolution image by reconstructing the image from patches, a difference between the high-frequency image and the interpolation image, and the interpolation image.

TABLE 1. Performance evaluation based on the benchmark for the DW-MRI dataset, Red is the best performance among all methods, and blue is the second-best performance among all methods.

Algorithms	Scale Factore	Parameters						
		MSE	PSNR	SSIM	GMSD	MI	FSIM	NIQE
Bicubic	x2	122.6	27.38	0.8746	0.0769	2.1069	0.8783	5.4412
RDN	x2	0.281	43.08	0.9832	0.016	4.3721	0.9310	5.1389
CSFM	x2	0.280	43.11	0.9838	0.013	4.3789	0.9388	5.1245
SRFBN	x2	0.283	43.012	0.9830	0.015	4.3645	0.9299	5.1438
CFSRCNN	x2	0.341	42.97	0.9813	0.019	4.2784	0.9201	5.1897
SOM-SR(Ours)	x2	0.2717	43.218	0.9845	0.011	4.4897	0.9456	5
Bicubic	x4	421.655	22.0423	0.6726	0.1716	1.6305	0.7734	6.5139
RDN	x4	274.54	37.50	0.8189	0.751	3.8281	0.9082	5.4269
CSFM	x4	273.87	37.65	0.8195	0.746	3.8285	0.9096	5.4256
SRFBN	x4	277.03	37.34	0.8183	0.788	3.8140	0.9071	5.4283
CFSRCNN	x4	278.92	37.26	0.8178	0.796	3.7982	0.9049	5.4388
SOM-SR(Ours)	x4	260.684	38.4032	0.8282	0.689	3.8303	0.9164	5.2738

proposed superresolution imaging algorithm with other superresolution imaging methods. We also used 25 DW images and 30 MRI images in addition to the training images to evaluate the proposed method with the other superresolution imaging algorithms.
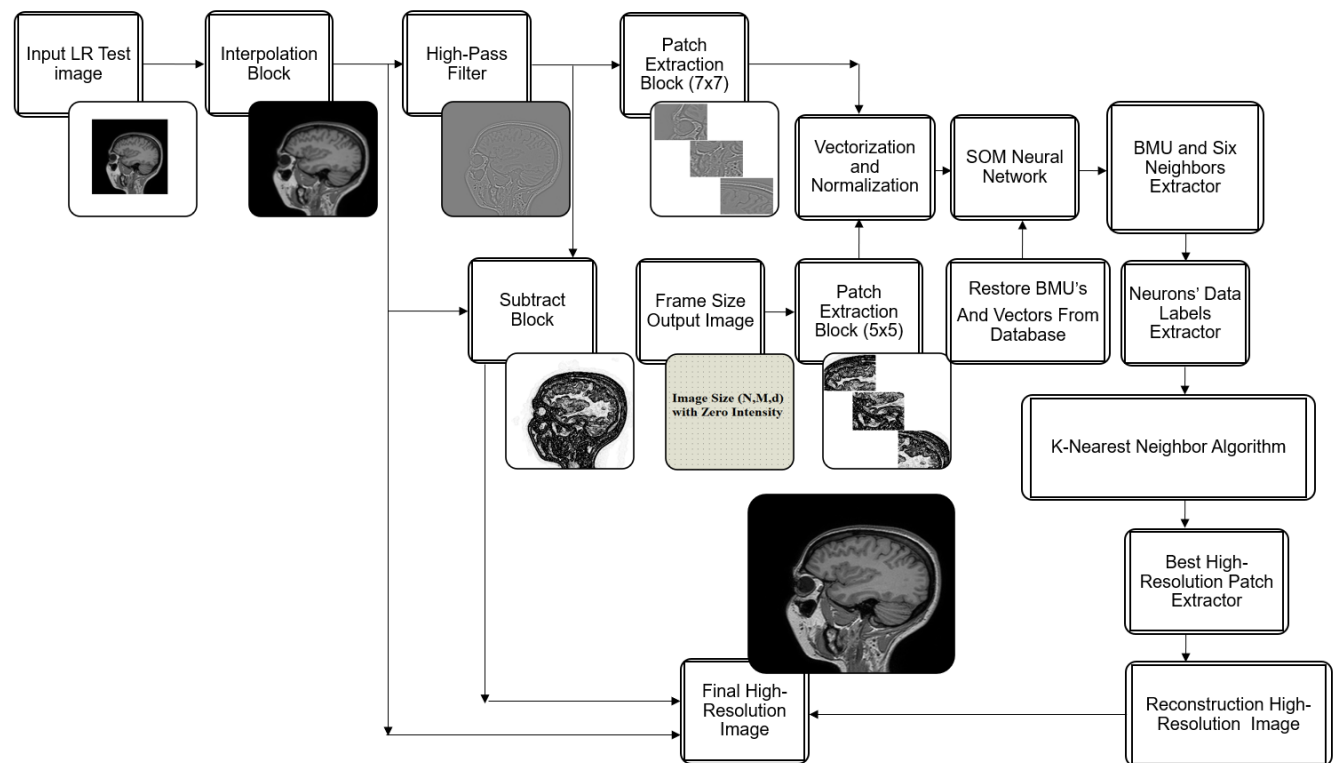
FIGURE 9. The proposed block diagram for the reconstruction of high-resolution images.

TABLE 2. Performance evaluation based on the MRI spectroscopic dataset benchmark, red shows the best performance among all methods, and blue shows the second-best performance among all methods.

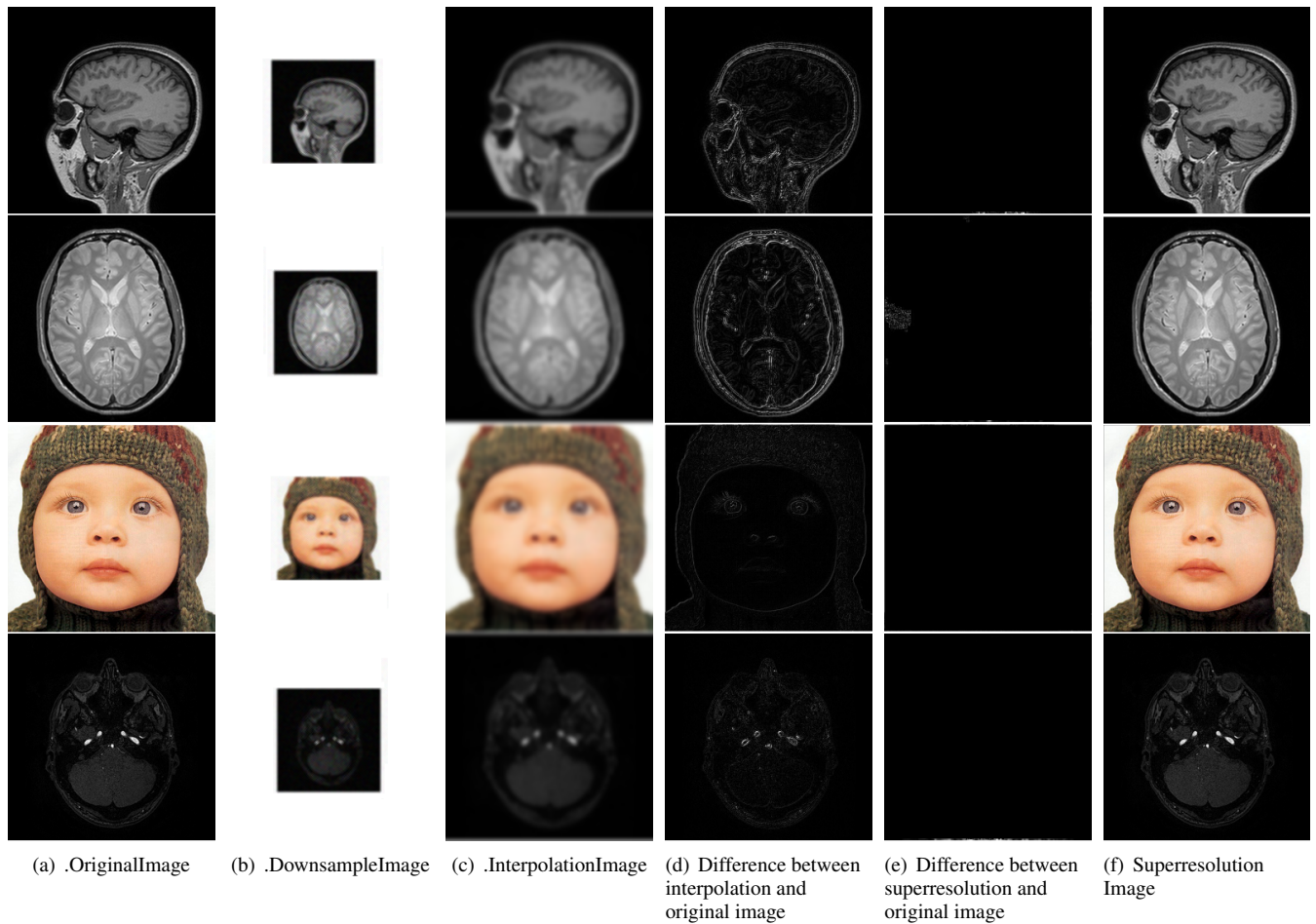| Algorithms	Scale Factore	Parameters						
		MSE	PSNR	SSIM	GMSD	MI	FSIM	NIQE
Bicubic	x2	86.3777	29.1952	0.8735	0.0668	1.8143	0.8819	5.6146
SRCNN	x2	0.183	41.77	0.9510	0.0195	4.8734	0.9401	5.3850
SRGAN	x2	0.174	41.99	0.9542	0.0185	4.9912	0.9412	5.3843
SRFBN	x2	0.191	41.65	0.9497	0.204	4.8001	0.9392	5.3855
CFSRCNN	x2	0.200	40.98	0.9488	0.264	4.6523	0.9299	5.4543
SOM-SR(Ours)	x2	0.163	42.9	0.9667	0.0045	5.2771	0.9544	5.2939
Bicubic	x4	297.1158	24.2542	0.704	0.1507	1.3629	0.7899	6.5151
SRCNN	x4	192.03	33.33	0.8200	0.0982	2.6412	0.8851	6.0654
SRGAN	x4	183.6	33.41	0.8212	0.0932	2.7645	0.8875	6.0001
SRFBN	x4	188.2	33.21	0.8189	0.1001	2.5856	0.8822	6.0832
CFSRCNN	x4	199.1	32.96	0.8132	0.1143	2.2895	0.8788	6.1432
SOM-SR(Ours)	x4	162.8227	34.3214	0.8252	0.0865	3.0192	0.8927	5.9431

B. THE SIMULATION ENVIRONMENT

All simulations of the proposed method were performed with MATLAB software version 2021a installed on Acer laptops with an Intel(R) Core (TM) i7–9750H CPU at 2.60 GHz Ram 16.0 GB and Windows 10 64 bit operating system and an Nvidia GeForce GTX 1650 graphics card with 4 GB GDDR5 Ram and 896 CUDA cores.

C. THE SIMULATION RESULTS

We performed experiments with traditional and state-of-the-art image superresolution algorithms to improve the quality and resolution of natural and medical magnetic resonance images at different image magnifications. In this paper, we present the evaluation results to compare the performance

and runtime of the algorithms on the reconstructed images. Initially, in the preprocessing stage, low-resolution natural and medical images were generated by subsampling the selected high-resolution images by a factor that is equal to the appropriate magnification factor. The low-resolution images were then blurred using a low-pass Gaussian filter (LPF) with a kernel size of 3x3 for sampling at Factor 2. Likewise, the size of the low-pass Gaussian filter was set to 3x3 for sampling at Factors 3 and 4. Next, to train a shared dictionary, we extracted more than 1,050,000 pairs of low- and high-resolution patches independently from each training dataset, such as Div2k and MRI. We considered a patch size of 5×5 with a one-pixel overlap between neighboring patches for the high-resolution images and a patch

| (a) .OriginalImage | (b) .DownsampleImage | (c) .InterpolationImage | (d) Difference between interpolation and original image | (e) Difference between superresolution and original image | (f) Superresolution Image |

FIGURE 10. Proposed image superresolution method for test images of different datasets with scale factor 2.

TABLE 3. Comparing the results of the proposed method with other algorithms for the complete databases with the image zoom factor 2. The numbers with a red mark indicate the best results, and the numbers with a blue mark indicate the second-best results.

Algorithms	SET 5		SET 14		BSDS 100		URBAN 100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	33.66	0.9299	30.24	0.8688	29.56	0.8431	26.88	0.8403
A+	36.54	0.9544	32.28	0.9056	31.21	0.8863	29.2	0.8938
RFL	36.54	0.9537	32.26	0.904	31.16	0.884	29.11	0.8904
SelfExSR	36.49	0.9537	32.22	0.9034	31.18	0.8855	29.54	0.8967
SCN	36.58	0.954	32.35	0.905	31.26	0.885	29.52	0.897
RED30	37.66	0.9599	32.94	0.9144	31.98	0.8974	30.91	0.9159
SRCNN	36.66	0.9542	32.42	0.9063	31.36	0.8879	29.5	0.8946
FSRCNN	37	0.9558	32.63	0.9088	31.53	0.892	29.88	0.902
VDSR	37.53	0.9587	33.03	0.9124	31.9	0.896	30.76	0.914
DRCN	37.63	0.9588	33.04	0.9118	31.85	0.8942	30.75	0.9133
LapSRN	37.52	0.959	33.08	0.913	31.8	-/-	30.41	0.91
DRRN	37.74	0.9591	33.23	0.9136	32.05	0.8973	31.23	0.9188
MemNet	37.78	0.9597	33.28	0.9142	32.08	0.8978	31.31	0.9195
CARN-M	37.53	0.9583	33.26	0.9141	31.92	0.896	3083	0.9233
RDN	38.24	0.9614	34.01	0.9212	32.34	0.901	32.89	0.9353
CSFM	38.26	0.9615	34.07	0.9213	32.37	0.9021	33.12	0.9366
SRFBN	38.11	0.9609	33.82	0.9196	32.29	0.901	32.62	0.9328
CFSRCNN	37.79	0.9591	33.51	0.9165	32.11	0.8988	32.07	0.9273
SOM-SR(Ours)	38.35	0.9881	34.6407	0.9265	32.45	0.9024	33.2	0.9377

size of 7 × 7 with a one-pixel overlap between neighboring patches for the low-resolution images. It is clear that as the number of patches increased, the quality of the reconstructed high-resolution image increased. Clearly, as the number of patches increased, the quality and resolution of the reconstructed high-resolution image increased. However, the

TABLE 4. Comparing the results of the proposed method with other algorithms for the complete databases with the image zoom factor 3. The numbers with a red mark indicate the best results, and the numbers with a blue mark indicate the second-best results.

Algorithms	SET 5		SET 14		BSDS 100		URBAN 100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	30.39	0.8682	27.55	0.7742	27.21	0.7385	24.46	0.7349
A+	32.58	0.9088	29.13	0.8188	28.29	0.7835	26.03	0.7973
RFL	32.43	0.9057	29.05	0.8164	28.22	0.7806	25.86	0.79
SelfExSR	32.58	0.9093	29.16	0.8196	28.29	0.784	26.44	0.8088
SCN	32.62	0.908	29.16	0.818	28.33	0.783	26.21	0.801
RED30	33.82	0.923	29.61	0.8341	28.92	0.7993	27.31	0.8303
HDRN	34.24	0.924	30.23	0.84	28.96	0.804	27.93	0.849
LSRCNN	34.05	0.9238	30.16	0.8384	28.94	0.8012	27.76	0.8424
FERN	34.36	0.9268	30.35	0.8422	29.1	0.8049	28.18	0.8521
SRCNN	32.75	0.909	29.28	0.8209	28.41	0.7863	26.24	0.7989
FSRCNN	33.16	0.914	29.43	0.8242	28.53	0.791	26.43	0.808
VDSR	33.66	0.9213	29.77	0.8314	28.82	0.7976	27.14	0.8279
DRCN	33.82	0.9226	29.76	0.8311	28.8	0.7963	27.15	0.8276
LapSRN	33.82	0.922	29.63	0.8269	28.82	0.798	27.07	0.828
DRRN	34.03	0.9244	29.96	0.8349	28.95	0.8004	27.53	0.8378
EDSR	34.37	0.927	30.28	0.8417	29.09	0.8025	28.15	0.8527
MemNet	34.09	0.9248	30	0.835	28.96	0.8001	27.56	0.8376
CARN-M	33.99	0.9255	30.08	0.8367	28.91	0.8	26.86	0.8263
RDN	34.71	0.9296	30.57	0.8468	29.26	0.8093	28.8	0.8653
CSFM	34.76	0.9301	30.63	0.8477	29.3	0.8105	28.98	0.8681
SRFBN	34.7	0.9292	30.51	0.8461	29.24	0.8084	28.73	0.8641
CFSRCNN	34.24	0.9256	30.27	0.841	29.03	0.8035	28.04	0.8496
SOM-SR(Ours)	31.04	0.91	24.1334	0.7699	29.11	0.7903	28.79	0.8672

TABLE 5. Comparing the results of the proposed method with other algorithms for the complete databases with the image zoom factor 4. The numbers with a red mark indicate the best results, and the numbers with a blue mark indicate the second-best results.

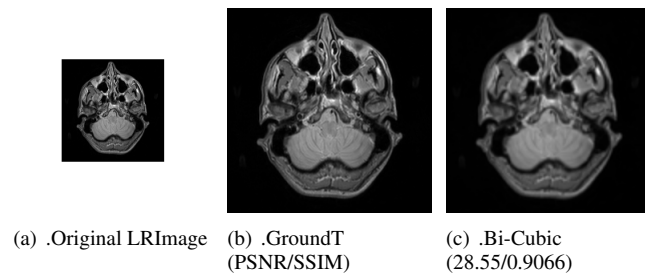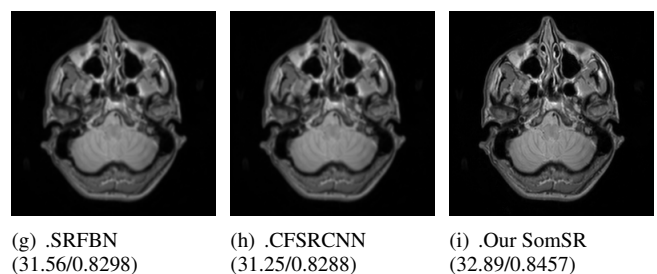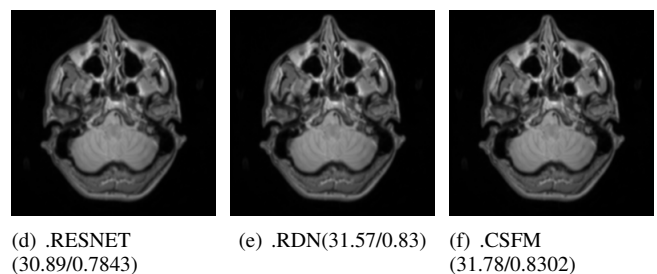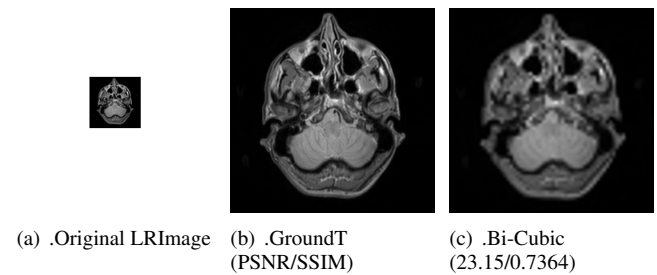| Algorithms	SET 5		SET 14		BSDS 100		URBAN 100	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	28.42	0.8104	26	0.7027	25.96	0.6675	23.14	0.6577
A+	30.28	0.8603	27.32	0.7491	26.82	0.7087	24.32	0.7183
RFL	30.14	0.8548	27.27	0.7451	26.75	0.7054	24.19	0.7096
SelfExSR	30.31	0.8619	27.4	0.7518	26.84	0.7106	24.79	0.7374
SCN	30.41	0.863	27.39	0.751	26.88	0.711	24.52	0.726
RED30	31.51	0.8869	27.86	0.7718	27.39	0.7286	25.35	0.7587
HDRN	32.23	0.896	28.58	0.781	27.53	0.737	26.09	0.787
LSRCNN	31.88	0.8907	28.43	0.7776	27.47	0.7321	25.78	0.7739
FERN	32.17	0.8944	28.59	0.7812	27.57	0.7364	26.11	0.7856
SRCNN	30.48	0.8628	27.49	0.7503	26.9	0.7101	24.52	0.7221
FSRCNN	30.71	0.8657	27.59	0.7535	26.98	0.715	24.62	0.728
SRDenseNet	32.02	0.893	28.5	0.778	27.53	0.733	26.05	0.781
VDSR	31.35	0.8838	28.01	0.7674	27.29	0.7251	25.18	0.7524
DRCN	31.53	0.8854	28.02	0.767	27.23	0.7233	25.14	0.751
SRGAN	29.46	0.838	26.6	0.718	25.74	0.666	24.5	0.736
LapSRN	31.54	0.885	28.19	0.772	27.32	0.728	25.21	0.756
ESPCN	29.1	0.851	26.4	0.744	25.5	0.696	24.02	0.726
DRRN	31.68	0.8888	28.21	0.772	27.38	0.7284	25.44	0.7638
EDSR	32.46	0.896	27.71	0.786	27.72	0.742	26.64	0.803
MemNet	31.74	0.8893	28.26	0.7723	27.4	0.7281	25.5	0.763
CARN-M	31.92	0.8903	28.42	0.7762	27.44	0.7304	25.63	0.7688
SR-RESNET	32.05	0.891	28.49	0.782	27.61	0.736	26.09	0.783
RDN	32.47	0.899	28.81	0.7871	27.72	0.7419	26.61	0.8028
CSFM	32.61	0.9	28.87	0.7886	27.76	0.7432	26.78	0.8065
SRFBN	32.47	0.8983	28.81	0.7868	27.72	0.7409	26.6	0.8015
CFSRCNN	32.06	0.892	28.57	0.78	27.53	0.7333	26.03	0.7824
SOM-SR(Ours)	25.1535	0.8202	20.5376	0.49054	21.43	0.603	22.43	0.6454

maximum efficient number of patches is finite, so increasing the number of patches beyond the maximum number did not affect the quality of the reconstructed images and only consumed machine memory with useless data. In addition, our attempts to extract more patches beyond the specific patch limit only increased the algorithm's execution time. By creating the image patches for natural and medical images

from their image training datasets and training the dictionary with feature vectors of the patches based on the desired application, we tested the proposed image superresolution algorithm with various natural and medical image datasets and next compared the results with other traditional and state-of-the-art image enhancement methods using quantitative and qualitative criteria. We used six quantitative and
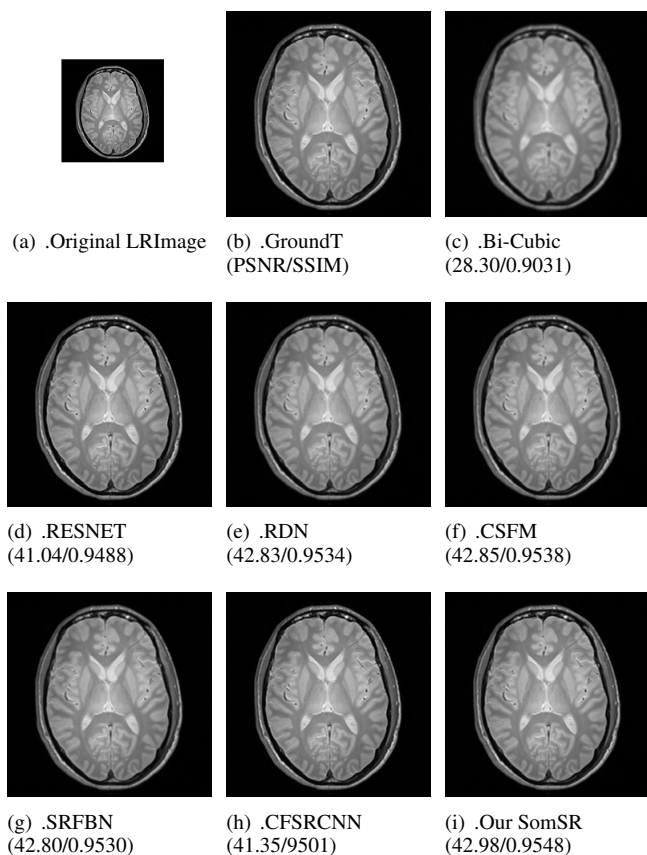
TABLE 6. Comparison of the average execution time(seconds) of the different algorithms, where red shows the best performance among all methods and blue shows the second-best performance among all methods.

Dataset	Scale	Bicubic	A+	SelfExSR	SCN	VDSR	MSLAPSRN	SRCNN	DRCN	SOM-SR(Ours)
SET 5	2	0.0010	0.3525	83.866	1.1632	0.2345	0.2273	0.1805	0.2992	281.134
	3	0.0009	0.3517	83.725	1.1606	0.2342	0.2270	0.1812	0.2987	281.092
	4	0.0009	0.3520	83.585	1.1616	0.2340	0.2266	0.1852	0.2982	281.063
	8	0.0009	0.3513	84.033	1.1592	0.2348	0.2278	0.1841	0.2998	282.156
SET 14	2	0.0016	0.6917	65.982	2.2826	0.1794	0.1789	0.3923	0.2354	363.867
	3	0.0020	0.6912	66.066	2.2809	0.1796	0.1791	0.3926	0.2357	364.151
	4	0.0017	0.6913	66.178	2.2812	0.1798	0.1794	0.3932	0.2361	364.202
	8	0.0013	0.6924	66.122	2.2849	0.1797	0.1792	0.3926	0.2359	364.241
BSDS 100	2	0.0012	0.4603	90.845	1.5189	0.2482	0.2463	0.2315	0.3241	487.686
	3	0.0012	0.4607	90.789	1.5203	0.2481	0.2461	0.2317	0.3239	487.541
	4	0.0012	0.4615	90.901	1.5229	0.2483	0.2464	0.2361	0.3243	487.438
	8	0.0012	0.4601	90.733	1.5183	0.2480	0.2460	0.2357	0.3237	487.239
URBAN 100	2	0.0038	0.5433	120.837	1.7928	0.3371	0.3276	0.3311	0.4311	617.98
	3	0.0037	0.5426	121.005	1.7905	0.3374	0.3280	0.3305	0.4317	618.65
	4	0.0035	0.5429	121.117	1.7915	0.3376	0.3283	0.3308	0.4321	618.50
	8	0.0031	0.5430	120.977	1.7919	0.3373	0.3280	0.3315	0.4316	618.66
DW	2	0.0009	0.3365	80.672	1.1043	0.2188	0.2142	0.1769	0.2885	279.892
	3	0.0007	0.3338	80.613	1.0881	0.2182	0.2131	0.1772	0.2889	279.888
	4	0.0006	0.3302	80.603	1.0677	0.2177	0.2126	0.1764	0.2883	279.893
	8	0.0006	0.3287	80.596	1.0598	0.2179	0.2127	0.1775	0.2878	279.796
MRI	2	0.0007	0.3327	80.312	1.0796	0.2147	0.2129	0.1745	0.2884	278.991
	3	0.0007	0.3326	80.308	1.0787	0.2148	0.2127	0.1746	0.2877	278.987
	4	0.0007	0.3228	80.310	1.0788	0.2145	0.2120	0.1747	0.2879	278.990
	8	0.0006	0.3210	80.298	1.0774	0.2140	0.2122	0.1739	0.2770	278.983

(a) .Original LRImage

(b) .GroundT (PSNR/SSIM)

(c) .Bi-Cubic (28.55/0.9066)

(d) .RESNET (43.38/0.9487)

(e) .RDN (44.24/0.9626)

(f) .CSFM (44.26/0.9635)

(g) .SRFBN (44.06/0.9628)

(h) .CFSRCNN (43.65/0.9488)

(i) .Our SomSR (44.98/0.9648)
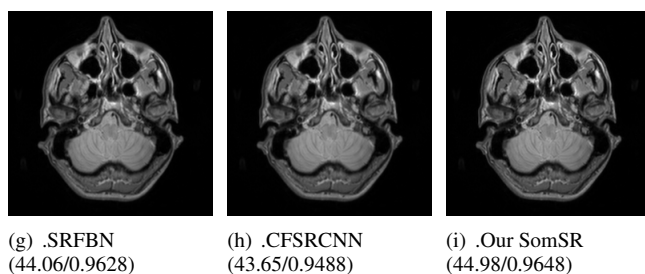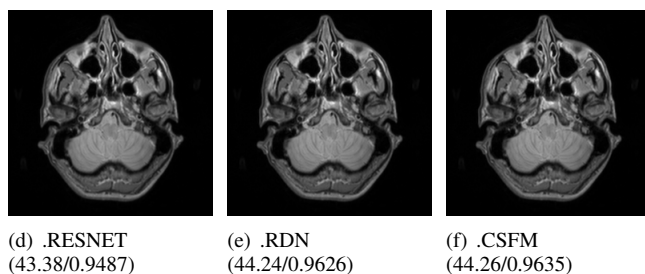
FIGURE 11. Comparison of the proposed superresolution with other methods for sample images in the DW-MRI database with magnification x2.

(a) .Original LRImage

(b) .GroundT (PSNR/SSIM)

(c) .Bi-Cubic (23.15/0.7364)

(d) .RESNET (30.89/0.7843)

(e) .RDN(31.57/0.83)

(f) .CSFM (31.78/0.8302)

(g) .SRFBN (31.56/0.8298)

(h) .CFSRCNN (31.25/0.8288)

(i) .Our SomSR (32.89/0.8457)

FIGURE 12. Comparison of the proposed superresolution with other methods for sample images in the DW-MRI database with magnification x4.

qualitative parameters to compare the results of the proposed algorithm with other traditional and state-of-the-art image superresolution methods. In addition to commonly used evaluation metrics, such as the mean squared error (MSE), peak

(a) .Original LRImage (b) .GroundT (PSNR/SSIM) (c) .Bi-Cubic (28.30/0.9031)

(d) .RESNET (41.04/0.9488) (e) .RDN (42.83/0.9534) (f) .CSFM (42.85/0.9538)

(g) .SRFBN (42.80/0.9530) (h) .CFSRCNN (41.35/9501) (i) .Our SomSR (42.98/0.9548)
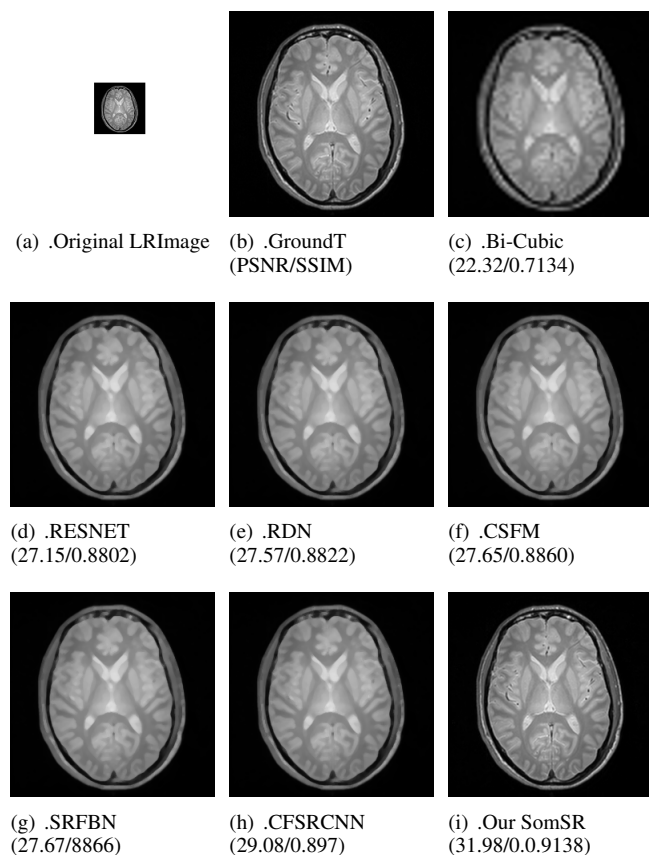
FIGURE 13. Comparison of the proposed superresolution with other methods for sample images in the spectroscopic MRI database with magnification x2.

(a) .Original LRImage (b) .GroundT (PSNR/SSIM) (c) .Bi-Cubic (22.32/0.7134)

(d) .RESNET (27.15/0.8802) (e) .RDN (27.57/0.8822) (f) .CSFM (27.65/0.8860)

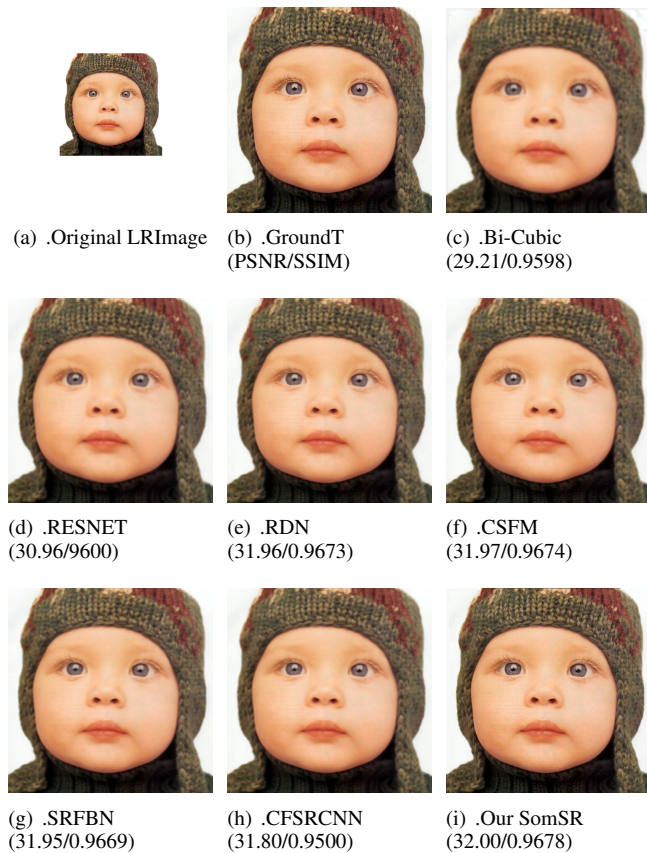(g) .SRFBN (27.67/8866) (h) .CFSRCNN (29.08/0.897) (i) .Our SomSR (31.98/0.0.9138)

FIGURE 14. Comparison of the proposed superresolution with other methods for sample images in the Spectroscopic MRI database with magnification x4.

signal-to-noise ratio (PSNR), and structural similarity index (SSIM), we used some other benchmarks to evaluate medical imaging enhancement approaches, such as the mutual information (MI) measure, which measures the degree of statistical dependence between two random variables [38]; the feature similarity index measurement (FSIM), which is based on the measurement of phase congruence and gradient magnitude to characterize local image quality [39]; gradient magnitude similarity deviation (GMSD), which is based on the exploration of the global variation in the local image quality map based on the gradient for the prediction of the global image quality [40]; and the natural image quality evaluator, which determines the image quality by calculating the local statistics of the input image (NIQE) [41]. We considered several approaches to compare the performance of the proposed image superresolution algorithm for medical MR imaging compared to traditional and state-of-the-art methods. Therefore, for better image superresolution reconstruction, the values of PSNR, MSSIM, MI, FSIM should be high, and MSE, GMSD, NIQE should be small.

1) The Medical Diffusion-Weighted MRI Image Superresolution

We tested the proposed image superresolution algorithm and other image superresolution methods on DW-MRI images,

and the corresponding results of the algorithms on DW-MRI image reconstruction for visual comparison and image comparison according to quantitative image benchmarks are presented in Figure 11 for the scale factor of 2 and Figure 12 for the scale factor of 4. The proposed method in image reconstruction successfully provided the best performance in reconstructing object edges from images and also reconstructed the edges of objects with a smooth slope and not with a fast and steep slope, similar to other superresolution algorithms, which can be seen in the figures. The proposed method achieved an average improvement of approximately 0.7532 and 0.118 dB in PSNR over that of the CSFM method for 4-factor and 2-factor scaling, respectively, as shown in Table 1. For the proposed method, significant improvements were observed in PSNR, SSIM, and many other reference parameters. Because of the above results and careful analysis of the results by expert clinical operators, it is evident that for DW-MRI superresolution imaging, the proposed method is much more effective than traditional and state-of-the-art superresolution imaging techniques.

2) The Medical Spectroscopic MRI Image Superresolution

The results of the proposed method compared with other image enhancement methods for spectroscopic MR images are presented in Figure 13 and 14 for scale ratios of 2 and 4,

IEEE *Access*

(a) .Original LRImage   (b) .GroundT (PSNR/SSIM)   (c) .Bi-Cubic (29.21/0.9598)

(d) .RESNET (30.96/9600)   (e) .RDN (31.96/0.9673)   (f) .CSFM (31.97/0.9674)

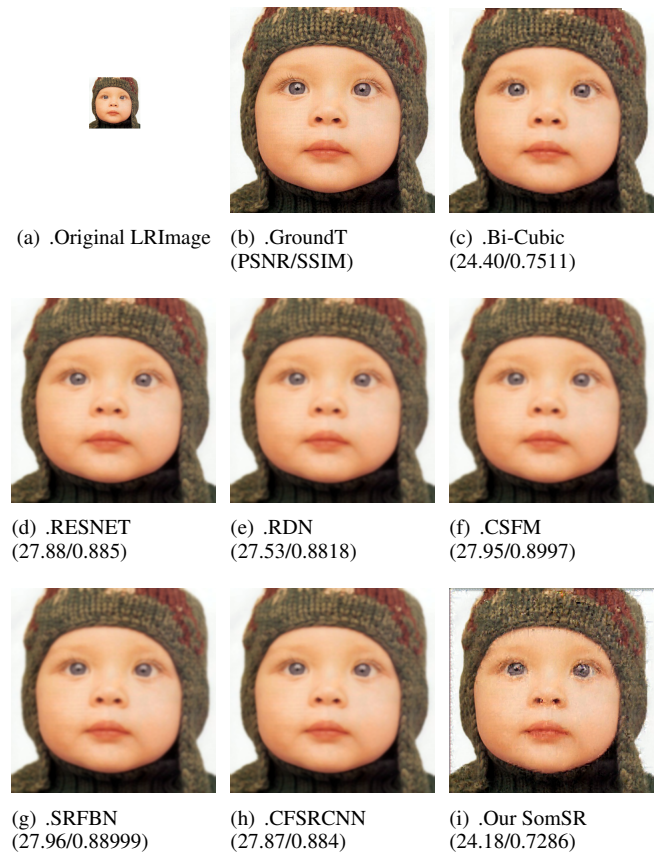(g) .SRFBN (31.95/0.9669)   (h) .CFSRCNN (31.80/0.9500)   (i) .Our SomSR (32.00/0.9678)

FIGURE 15. Comparison of the proposed superresolution with other methods for sample images in the real ground-truth database with magnification x2.

(a) .Original LRImage   (b) .GroundT (PSNR/SSIM)   (c) .Bi-Cubic (24.40/0.7511)

(d) .RESNET (27.88/0.885)   (e) .RDN (27.53/0.8818)   (f) .CSFM (27.95/0.8997)

(g) .SRFBN (27.96/0.88999)   (h) .CFSRCNN (27.87/0.884)   (i) .Our SomSR (24.18/0.7286)

FIGURE 16. Comparison of the proposed superresolution with other methods for sample images in the real ground-truth database with magnification x4.

respectively. The proposed method provides a better visual representation of the reconstructed image, with increased edge and boundary details, compared to traditional and state-of-the-art techniques. For most parameters, the proposed method is superior to other methods. The proposed method achieves an average improvement of approximately 0.91 and 0.9114 dB in PSNR over the CSFM method for 2-factor and 4-factor scaling, respectively, as shown in Table 2. The proposed method shows significant improvements in terms of PSNR, SSIM, and several other parameters. It is determined that, in the case of MR superresolution spectroscopic imaging, the proposed method is much more efficient than traditional and modern superresolution imaging techniques.

3) The Real Ground-Truth Image Superresolution

We evaluated the performance of the proposed method both quantitatively by image matching criteria and qualitatively by visual comparison with traditional and state-of-the-art methods, including interpolation approaches (bicubic method), dictionary-based image enhancement (A+ [37], RFL [42]), self-similarity based methods [16], artificial neural network-based methods (Ms-LapSR [43], MemNET [44], RED30 [45], CARN-M [46], CFSRCNNN [47], HDRN [48], LSR-CNN [49], FERN [50], SRDenseNet [51], DRRN [44], SR-GAN [52], ESPCN [53], SCN [54], FSRCN [55], SRCNN

[56], DRCN [57], VDSR [58], LapSRN [5], EDSR [59], SR-Resnet [60], RDN [61], CSFM [62], and SRFBN [63] for a scale factor of 2, 3, and 4. We performed several comprehensive experiments on conventional datasets, of which the SET5, SET14, and BSDS100 datasets consist of natural scenes, and the URBAN100 dataset includes images of architecturally challenging scenarios with varying details at different frequencies. These experimental results of the proposed approach and other traditional and state-of-the-art methods in terms of signal-to-noise ratio (PSNR) and structural similarity index (SSIM) criteria are shown in Table 3 for scale Factor 2, Table 4 for scale Factor 3 and Table 5 for scale Factor 4. The experimental results are presented in Table 3 and show the superiority and efficiency of the proposed model for high-resolution image reconstruction compared to other image superresolution methods. As a result, our method with scale Factor 2 outperforms traditional and state-of-the-art approaches, such as SRFBN and CSFM, on four datasets and achieves a comparable result with the RDN method with zoom Factor 2. In particular, compared with SRFBN, CSFM achieves a remarkable gain of almost the same PSNR and SSIM for zoom Factor 2. The proposed method achieves an average improvement of approximately 0.14 dB at SET5, 0.57 dB at SET14, 0.08 dB at BSDS100, and 0.13 dB at URBAN100 at PSNR over the CSFM method for the zoom
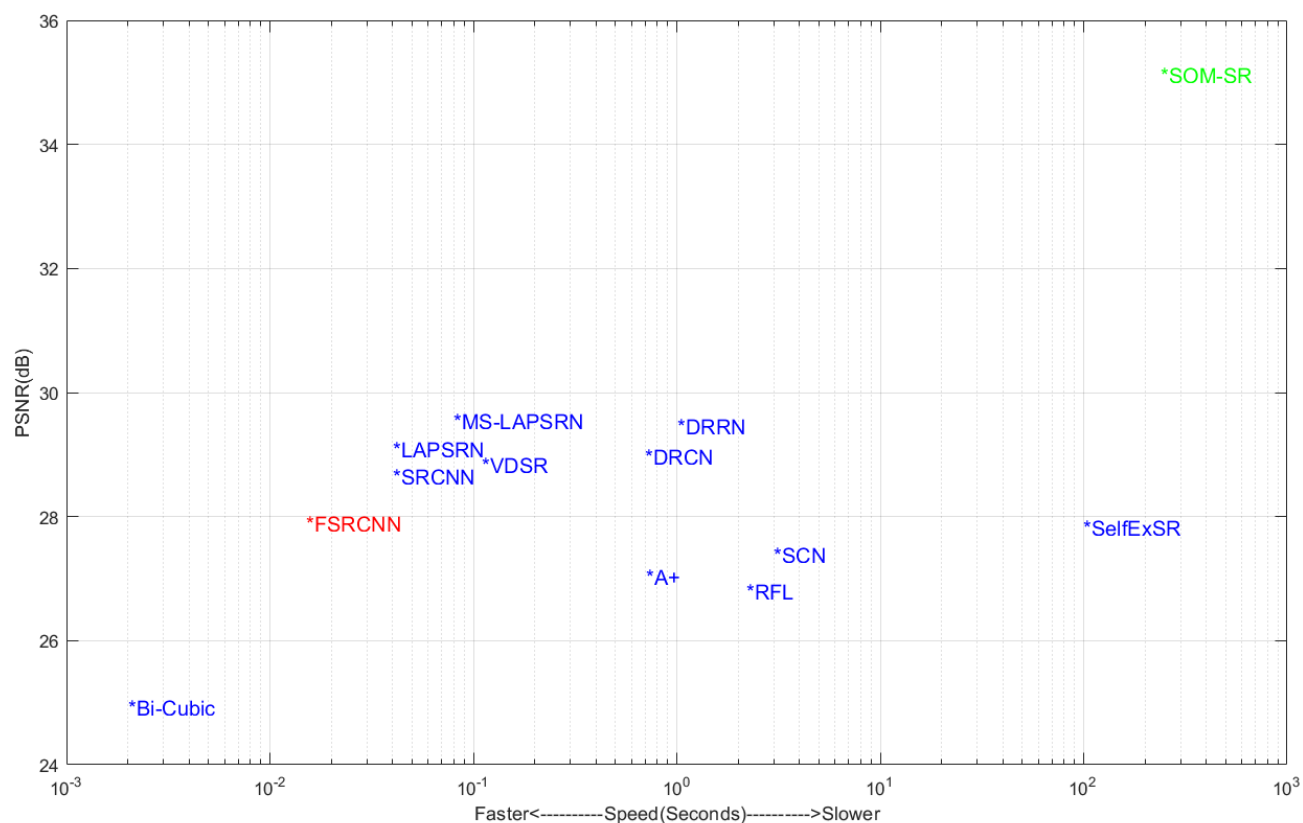
FIGURE 17. Running time versus PSNR of the proposed algorithm and other image enhancement algorithms on MRI images with a scale Factor x2. The method with the maximum PSNR is marked in green, and methods with a fast algorithm running time are marked in red.

scale of 2. The proposed method achieves at least an average improvement of approximately 0.0266 in SET5, 0.0052 in SET14, 0.0003 in BSDS100, and 0.0011 in URBAN100 in SSIM over the CSFM method for a scaling of 2. Moreover, as shown in Tables 4 and 5, the proposed method achieves weak results for scaling Factors 3 and 4. All these tables show that our image enhancement approach that is based on the SOM-KNN technique can present better results than traditional and some state-of-the-art image enhancement techniques and achieve reasonable PSNR and moderate SSIM at scale Factor 2. In practice, several experiments were performed with all image superresolution algorithms, the results of which, presented in Figures 15 and 16, show that our method is better than several others in reconstructing all details of an image, such as edges and lines. However, to present the best results in natural image reconstruction, the proposed method needs a very large training database, and the similarity of image details in the database should be higher.

4) Comparison of computing times

We used the existing state-of-the-art and traditional algorithm codes to compare the running time of our method with these algorithms, and the results of the average running time on various image datasets are shown in Table 6. We can see in Table 6 that our proposed method has a much lower running

time speed than other image enhancement algorithms. The execution speed of the proposed algorithm is much slower than that of other image enhancement algorithms due to the use of all three image color components in the algorithm's calculations, i.e., the R, G, and B components. As a result, the computational time is more than three times longer than that of other algorithms that use YCbCr space and employ only one image component, such as the Y component, in their calculations. The use of three color components increases the number of image patches up to three times, and as a result, the image patch feature vectors have more attributes. The proposed algorithm easily finds the correct high-resolution patches. Therefore, the proposed algorithm provides higher-quality images than those produced by other image enhancement algorithms. Figure 17 shows the peak signal-to-noise ratio values of the different superresolution imaging algorithms as a function of the execution speed of each algorithm over time. Our method is much slower than the other methods but has the maximum peak signal-to-noise ratio over that of other algorithms.

5) Serial and parallel processing in the CPU-GPU environment

In this work, we used both series programming on the CPU hardware and parallel programming on the GPU hardware to

execute the proposed image superresolution algorithm. We used CPU hardware and serial programming for image up- and downsampling with a predefined zoom factor, filtered images with different kernels, and extracted patches with multiple kernel numbers. We also used it for vectorization, vector normalization, and at the end of the algorithm, for high-resolution image reconstruction. We used GPU hardware and parallel programming to train the SOM algorithm, the k-nearest neighbor algorithm, the high-resolution image patch, and high-resolution image reconstruction at the end of the algorithm.

V. DISCUSSIONS

In this paper, to increase the quality and resolution of medical images, we introduced an image enhancement method and presented the results obtained in improving the quality and resolution of natural and medical images. Such a method is based on a dictionary learning-based medical image reconstruction approach. In recent years, several dictionary-based image superresolution algorithms have been used to reconstruct high-resolution images from low-resolution images based on a single dictionary. However, due to the complexity of high-frequency details and texture information present in real images, a single dictionary cannot successfully recover image details. Single dictionary-based image superresolution algorithms did not present good performance in reconstructing real images. Image enhancement methods based on regression and clustering using multiple dictionaries achieved better results in high-resolution image reconstruction than image enhancement methods using single dictionaries, but these methods did not find the inherent relationship between LR and HR patches. Deep neural network learning-based methods can reconstruct high-resolution images significantly better than clustering and regression-based methods, but estimating the amount of error is the major problem for these methods. In addition, they use the backpropagation technique to update their weight vectors. These weight vector update processes are time-consuming, which increases the execution time of the image enhancement algorithm in the training stage. However, all of the above methods for reconstructing object boundaries and edges in images with smoothed slopes are weak. This is especially true in medical images, which have spatially changing pixel intensities throughout an image and considerable variations, unlike natural images, whose pixel intensities throughout an image are almost uniform and hardly change. We attempted to solve the three main problems of the previous methods, such as the intrinsic relationship between low- and high-resolution patches, the limitation of image detail acquisition, the omission of updating the neural network weight vectors by the backpropagation approach, and the estimation of the optimal and minimum error values for calculating the neural network weight vectors in the new model. The proposed image superresolution method uses self-organizing maps as a preprocessor, and this neural network works as the tree search block in the regression and classification methods. Thus,

by using this preprocessor, we achieved the accuracy and speed of artificial neural networks in the single dictionary-based image reconstruction methods that work according to the classification and regression methods by using this preprocessor. Moreover, the proposed image superresolution method can reconstruct high-resolution images with better quality from a single dictionary, unlike the previous image superresolution method based on a single dictionary, which cannot present high-quality images. In addition, we used the nearest neighbor algorithm as a classifier to classify the feature patch vectors that previously belonged to the neurons of the neural network in the training stage. In particular, the vectors corresponding to the neurons at the boundary of the neural network, since these neurons, depending on the network topology, have incomplete neighbors. Therefore, the algorithm can accurately reconstruct the high-frequency details and texture information of the reconstructed images by unsupervised learning approaches and effectively remove the artifact and blocking effects of the reconstructed high-resolution images by classifying the input feature vectors into the correct categories. In addition, it can reconstruct the boundaries and edges of objects in images with good quality. We trained our model with 800 images from the Div2K dataset after preprocessing and processing the images. We then tested the model with 100 images from the Div2K dataset, and our model test results were satisfactory. We validated the algorithm with 100 images from the Div2K dataset with acceptable validation results. In addition, to train, test, and validate the proposed model with medical MRI and DW images, we used the medical imaging dataset from the brain development organization. All training, testing, and validation processes for medical imaging with the same as the previous training for natural imaging. We compared the results of the proposed method with those of other traditional and state-of-the-art image superresolution methods on natural and medical image datasets. Figure 10 shows the resulting images reconstructed by our image enhancement method from various image datasets for scale Factor x2. We can see in these images that the image details, such as edges and boundaries, were reconstructed very well. As seen in Column (e) of the figure, which shows the high-frequency difference between the output image of the proposed algorithm and the original high-resolution image, the proposed algorithm reconstructed all the details of the high-frequency image, and all images in this column are black and do not contain any remaining high-frequency image detail. We compared images reconstructed by the proposed algorithm in different image databases with images reconstructed by other image enhancement algorithms, both in terms of quality by expert persons and quantitatively using numerical image comparison criteria. The proposed image reconstruction algorithm is superior to other image enhancement methods for both natural and medical images with a zoom factor of 2. However, as the image zoom factor increases, the proposed algorithm gradually loses its advantage compared to other image enhancement algorithms in reconstructing high-

quality images because it only uses a single high-frequency filter to process the input image, unlike the other algorithms that use several image feature extractor filters, such as edge, Gradian, Laplace, high-frequency and angular filters. As seen in the results presented in the tables for the different datasets, the results of the proposed method on the datasets are very different from each other, which is due to the use of trial and error methods in estimating the size of high- and low-resolution image patches. In addition, higher texture similarity in the input images leads to the reconstruction of more details in the output images. The proposed method is more capable of reconstructing human faces and medical magnetic resonance images with the best resolution, as these images have many similar details in their structures. The ability of the method to reconstruct natural images, such as landscapes and forests, which do not have many similar textures, is reduced. We also estimated the running time of the algorithms in the reconstruction of existing images by testing the proposed algorithm, as well as other existing image enhancement algorithms, on images with dimensions of 64x64, 256 x 256, 512 x 512, and 1024 x1024. The results showed that the execution time of the proposed algorithm is high compared to other methods because the proposed algorithm uses three components R, G, and B of the color image, so the computational time is three times longer than other image enhancement methods. We also used MATLAB software to run the algorithm, which is much slower than the C and Python programming languages used by other algorithms. However, despite the slowness of the algorithm compared to other image enhancement methods, it is surprisingly effective in reconstructing the details of image objects, such as edges and contours, with a very smooth slope, especially in clinical images. In the future, we will use gradient kernels to improve the resolution of these images.

VI. CONCLUSION

In this paper, we proposed a novel hybrid single-image superresolution method for magnetic resonance image enhancement based on a self-organizing neural network as a preprocessor, the nearest neighbor algorithm as a classifier, and the use of a high-pass filter as a high-frequency image detail extractor. Unlike other image enhancement methods, our method uses a single overcomplete external dictionary to train the proposed method. We resolved the intrinsic relationship problem between low- and high-resolution image patches in the training dictionary by defining feature vectors that directly contain the low- and high-resolution patches. In addition, we solved the difficulty of specifying the appropriate amount of error at the training and testing stages due to the variability of MR image contrast. We also solved the use of backpropagation methods in updating the neural network weight vectors by competitive learning that does not require error correction. Since MRI images of internal body parts and organs of people are very similar in appearance and differ only in size, our method of reconstructing such images is very effective. This method's success lies in its ability to

reconstruct the boundaries and edges of the objects in the image with a smooth slope without the need to perform any statistical calculations or adjust the regularization parameters. Practical experiments on the proposed superresolution method compared with other methods illustrate that our image reconstruction method achieves high performance. Under limited computational resources, our method achieves better evaluation criteria among all traditional and state-of-the-art benchmarks. It also achieves more reliable performance and resolution on images with different scale factors. We used CPU-GPU hardware and parallel programming to train and test the proposed algorithm and obtain better performance in algorithm speed and execution time.

ACKNOWLEDGMENT

REFERENCES

[1] Wei. Lu, Zhijin. Song, and Jinghui. Chu. A novel 3d medical image super-resolution method based on densely connected network. Biomedical Signal Processing and Control, 62:102120, 2020.

[2] Bhabesh. Deka, Sumit. Datta, Helal. Uddin Mullah, and Suman. Hazarika. Diffusion-weighted and spectroscopic mri super-resolution using sparse representations. Biomedical Signal Processing and Control, 60:101941, 2020.

[3] Mi-Sun Kang, Eunju Cha, Eunhee Kang, Jong Chul Ye, Nam-Gu Her, Jeong-Woo Oh, Do-Hyun Nam, Myoung-Hee Kim, and Sejung Yang. Accuracy improvement of quantification information using super-resolution with convolutional neural network for microscopy images. Biomedical Signal Processing and Control, 58:101–846, 2020.

[4] Hamid Behnam and Mina Hosseinpour. Temporal super resolution of ultrasound images using compressive sensing. Biomedical Signal Processing and Control, 52, 03 2019.

[5] Kamal Nasrollahi and Thomas Moeslund. Super-resolution: A comprehensive survey. Machine Vision and Applications, 25:1423–1468, 08 2014.

[6] Hui Liu, Yuxiu Lin, Bulat Ibragimov, and Caiming Zhang. Low dose 4d-ct super-resolution reconstruction via inter-plane motion estimation based on optical flow. Biomedical Signal Processing and Control, 62:102085, 2020.

[7] Hsieh Hou and H. Andrews. Cubic splines for image interpolation and digital filtering. IEEE Transactions on Acoustics, Speech, and Signal Processing, 26(6):508–517, 1978.

[8] J. Wu, M. Anisetti, W. Wu, E. Damiani, and G. Jeon. Bayer demosaicking with polynomial interpolation. IEEE Transactions on Image Processing, 25(11):5369–5382, 2016.

[9] Hong Zheng, Xiaobo Qu, Zhengjian Bai, Yunsong Liu, Jiyang Dong, Xi Peng, and Zhong Chen. Multi-contrast brain magnetic resonance image super-resolution using the local weight similarity. BMC Medical Imaging, 17, 01 2017.

[10] Hani Nozari Mirarkolaei, Sten Roar Snare, and Anne H Schistad Solberg. A robust bidirectional motion-compensated interpolation algorithm to enhance temporal resolution of 3d echocardiography. Biomedical Signal Processing and Control, 65:102384, 2021.

[11] R. C. Hardie, K. J. Barnard, and E. E. Armstrong. Joint map registration and high-resolution image estimation using a sequence of undersampled images. IEEE Transactions on Image Processing, 6(12):1621–1633, 1997.

[12] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar. Fast and robust multiframe super resolution. IEEE Transactions on Image Processing, 13(10):1327–1344, 2004.

[13] Freeman. William T, Pasztor. Egon C, and Carmichael. Owen T. Learning low-level vision. International Journal of Computer Vision, 40(1):25–47, 2000.

[14] S Baker and T Kanade. Hallucinating faces. In Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580), pages 83–88, 2000.

[15] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. volume 1, pages I–275, 01 2004.

[16] J. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5197–5206, 2015.

[17] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. IEEE Transactions on Image Processing, 19(11):2861–2873, 2010.

[18] T. Peleg and M. Elad. A statistical prediction model based on sparse representations for single image super-resolution. IEEE Transactions on Image Processing, 23(6):2569–2582, 2014.

[19] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. IEEE Computer Graphics and Applications, 22(2):56–65, 2002.

[20] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In Proceedings of the 7th International Conference on Curves and Surfaces, page 711–730, Berlin, Heidelberg, 2010. Springer-Verlag.

[21] Li-Wei Kang, Chih-Chung Hsu, Boqi Zhuang, Chia-Wen Lin, and Chia-Hung Yeh. Learning-based joint super-resolution and deblocking for a highly compressed image. IEEE Transactions on Multimedia, 17:921–934, 07 2015.

[22] Shuyuan Yang, Zhizhou Liu, Min Wang, Fenghua Sun, and Licheng Jiao. Multitask dictionary learning and sparse representation based single-image super-resolution reconstruction. Neurocomputing, 74:3193–3203, 10 2011.

[23] Shuyuan Yang, Min Wang, Yiguang Chen, and Yaxin Sun. Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding. IEEE Transactions on Image Processing, 21(9):4016–4028, 2012.

[24] Xiaomin Yang, Wei Wu, Kai Liu, Weilong Chen, and Zhili Zhou. Multiple dictionary pairs learning and sparse representation-based infrared image super-resolution with improved fuzzy clustering. Soft Computing, 22:1–14, 03 2018.

[25] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4):541–551, 1989.

[26] Olga Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Zhiheng Huang, A. Karpathy, A. Khosla, Michael S. Bernstein, A. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115:211–252, 2015.

[27] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, Computer Vision – ECCV 2014, pages 184–199, Cham, 2014. Springer International Publishing.

[28] Rauber. Andreas and Merkl. Dieter. Automatic labeling of self-organizing maps: Making a treasure-map reveal its secrets. In Zhong.Ning and Zhou.Lizhu, editors, Methodologies for Knowledge Discovery and Data Mining, pages 228–237, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[29] Cunningham. Padraig and Delany. Sarah. k-nearest neighbour classifiers. Mult Classif Syst, 04 2007.

[30] Liming Xu, Xianhua Zeng, Zhiwei Huang, Weisheng Li, and He Zhang. Low-dose chest x-ray image super-resolution using generative adversarial nets with spectral normalization. Biomedical Signal Processing and Control, 55:101600, 2020.

[31] Yongjin Zhou, Zhengji Liu, Lei Tian, Menglin Guo, and Jizhou Li. Finer cornea characterization with improved spatial resolution in corvis st. Biomedical Signal Processing and Control, 65:102297, 2021.

[32] A. K. Jain, R. P. W. Duin, and Jianchang Mao. Statistical pattern recognition: a review. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1):4–37, 2000.

[33] Simon Haykin. Neural Networks: A Comprehensive Foundation. Prentice Hall PTR, USA, 1st edition, 1994.

[34] A. O. Salau and S. Jain. Feature extraction: A survey of the types, techniques, applications. In 2019 International Conference on Signal Processing and Communication (ICSC), pages 158–164, 2019.

[35] Andrey Ignatov, Radu Timofte, et al. Pirm challenge on perceptual image enhancement on smartphones: report. In European Conference on Computer Vision (ECCV) Workshops, January 2019.

[36] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings Eighth

[37] IEEE International Conference on Computer Vision. ICCV 2001, volume 2, pages 416–423 vol.2, 2001.

[37] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In Computer Vision – ACCV 2014, pages 111–126, Cham, 2015. Springer International Publishing.

[38] M. Ceccarelli, M. di Bisceglie, C. Galdi, G. Giangregorio, and S. L. Ullo. Image registration using non-linear diffusion. In IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium, volume 5, pages V – 220–V – 223, 2008.

[39] L. Zhang, L. Zhang, X. Mou, and D. Zhang. Fsim: A feature similarity index for image quality assessment. IEEE Transactions on Image Processing, 20(8):2378–2386, 2011.

[40] W. Xue, L. Zhang, X. Mou, and A. C. Bovik. Gradient magnitude similarity deviation: A highly efficient perceptual image quality index. IEEE Transactions on Image Processing, 23(2):684–695, 2014.

[41] Ruben Fernandez-Beltran, Pedro Latorre-Carmona, and Filiberto Pla. Single-frame super-resolution in remote sensing: a practical overview. International Journal of Remote Sensing, 38(1):314–354, 2017.

[42] K. I. Kim and Y. Kwon. Single-image super-resolution using sparse regression and natural image prior. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(6):1127–1133, 2010.

[43] W. Lai, J. Huang, N. Ahuja, and M. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5835–5843, 2017.

[44] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2790–2798, 2017.

[45] Xiao-Jiao Mao, Chunhua Shen, and Yubin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In NIPS, 2016.

[46] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. 03 2018.

[47] Chunwei Tian, Ruibin Zhuge, Zhihao Wu, Yong Xu, Wangmeng Zuo, Chen Chen, and Chia-Wen Lin. Lightweight image super-resolution with enhanced cnn. Knowledge-Based Systems, 205:106235, 2020.

[48] Kui Jiang, Zhongyuan Wang, Peng Yi, and Junjun Jiang. Hierarchical dense recursive network for image super-resolution. Pattern Recognition, 107:107475, 2020.

[49] C. Tian, Y. Xu, W. Zuo, B. Zhang, L. Fei, and C. Lin. Coarse-to-fine cnn for image super-resolution. IEEE Transactions on Multimedia, pages 1–1, 2020.

[50] Zheng Hui, Xinbo Gao, and Xiumei Wang. Lightweight image super-resolution with feature enhancement residual network. Neurocomputing, 404:50–60, 2020.

[51] T. Tong, G. Li, X. Liu, and Q. Gao. Image super-resolution using dense skip connections. In 2017 IEEE International Conference on Computer Vision (ICCV), pages 4809–4817, 2017.

[52] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. pages 105–114, 07 2017.

[53] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. 06 2016.

[54] Y. Wang, L. Wang, H. Wang, and P. Li. End-to-end image super-resolution via deep and shallow convolutional networks. IEEE Access, 7:31959–31970, 2019.

[55] C. Dong and X. Tang. Accelerating the super-resolution convolutional neural network : Supplementary file. pages 391–407, 2016.

[56] Fan Yang, Wei Xu, and Yapeng Tian. Image super resolution using deep convolutional network based on topology aggregation structure. volume 1864, page 020185, 08 2017.

[57] J. Kim, J. K. Lee, and K. M. Lee. Deeply-recursive convolutional network for image super-resolution. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1637–1645, 2016.

[58] J. Kim, J. K. Lee, and K. M. Lee. Accurate image super-resolution using very deep convolutional networks. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1646–1654, 2016.

[59] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. Enhanced deep residual networks for single image super-resolution. In 2017 IEEE Conference on

Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1132–1140, 2017.

[60] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 06 2016.

[61] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu. Residual dense network for image super-resolution. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2472–2481, 2018.

[62] Y. Hu, J. Li, Y. Huang, and X. Gao. Channel-wise and spatial feature modulation network for single image super-resolution. IEEE Transactions on Circuits and Systems for Video Technology, 30(11):3911–3927, 2020.

[63] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

HAMID-REZA REZA-ALIKHANI was born in Tehran, Iran in 1952, received his B.Sc. degree in electrical and electronic engineering from Nottingham University in the U.K., an M.Sc. degree in power electronics engineering, and a Ph.D. in digital communication engineering from Loughborough University in England. His main research interests include control (robotics control), medical monitoring (physiotherapy) system design, power electronics, and image processing. He has worked as an electronics design engineer in England, and he is currently lecturing as an assistant professor at Tafresh University, Tafresh, Iran.

KHODABAKHSH AHMADIAN was born in Eslamabade-Gharb, Iran in 1974, received his B.Sc. degree in electrical and electronic engineering from Razi University, an M.Sc. degree in control systems engineering, and a Ph.D. in electronic engineering from Mahshahr University in Iran. His main research interests include control (industrial automation systems), medical image processing systems, power electronics, image processing, and pattern recognition. He has worked as an electronics engineer and industrial automation programmer in oil and gas in Iran, and he is currently employed at Oil and Gas Company in Iran.

• • •