

Single-image Super Resolution based on Group Sparse Representation via GAUSSIAN

Shuhua Xu, Fei Gao

Abstract—As recently newly techniques, Group based Sparse Representation (GSR) algorithms were proposed, which achieved an excellent performance of sparse representation, exploiting the concept of group as the basic unit of sparse representation which is composed of nonlocal image patches with similar structures and capturing intrinsic local sparsity and nonlocal self-similarity of images simultaneously in one unified framework. Inspired by this, we apply GSR to single image super resolution reconstruction. However, the Euclidean distance metric applied in the process of group construction in traditional GSR failed to capture nonlinear structural information between image patches, leading to that the performances of these algorithms were sensitive to the geometric structure of images. In order to solve the problem, on basic of existing GSR, the nonlinear nonlocal self-similarity and local information of image patches were captured by exploiting effectively Gaussian kernel distance metric instead of the Euclidean distance metric in the paper. The paper presents Single-image Super Resolution based on Group Sparse Representation via GAUSSIAN (GSRGSiSR) algorithm. Compared with many state-of-art SISR methods, extensive experimental results validate that the proposed method can obtain better peak signal-to-noise ratio (PSNR) and structural similarity (SSIM).

Keywords—single-image super resolution, group sparse representation, similar structures, Gaussian kernel distance, nonlocal nonlinear self-similarity.

I. INTRODUCTION

High resolution (HR) images are used in many practical applications, such as medical image analysis, computer vision, remote sensing, and so on. HR images can be obtained by single image super-resolution (SISR) methods which obtained the corresponding high resolution image of the low resolution (LR) image through an operation from a single low resolution (LR) image.

Single image super resolution method can be divided into three categories: the interpolation based methods [1], the reconstruction based methods [2] and the example based

methods [3]. Although the interpolation based methods are simple to perform, the reconstructed HR images are often blurred with jagged artifacts and ringing phenomenon. The reconstruction based methods introduced some prior knowledge during the reconstruction process, but the reconstructed HR images had excessive smoothing phenomenon or been lack of some important details, the blur effect was more obvious in the amplification of the reconstructed HR image, it may lose the real vision of images. The essence of the example based methods is to assume that the high frequency details lost in LR images can be obtained by studying the relationship between the LR image blocks and the corresponding HR image blocks, and the reconstructed HR image can keep the sharpness of the image expanding the size of the reconstructed HR image, so that they have become a hot research topic. However, the effect of these methods is mainly dependent on one large image database [4]. Recently, in order to solve this problem, Yang et al. [4] proposed a sparse representation based super resolution method(SCSR), which consisted of two stages: coding and linear combination.

The sparse model assumes that each image block can be represented sparsely by several elements in the dictionary, which are derived from the natural image. At present, there are two main problems in the sparse representation model based on image blocks [5]. First, dictionary learning is one large-scale and highly non-convex problem, and the computational complexity is high [6]. Secondly, the image block is one unit of sparse representation, and each image block is usually considered independently in dictionary learning and sparse coding process, ignoring the relationship between similar image blocks in essence, such as self-similarity [7]. In addition, according to the dictionary set, the relatively expensive nonlinear estimation method is generally used for calculating the sparse representation coefficient of image block, such as match pursuits [8], which may be unstable and inaccurate due to the coherence of the dictionary [8].

For the above shortcomings of sparse image block based natural image model, Zhang et al. [5] proposed group based sparse representation (GSR) model classifying similar local structure of image blocks as one group, which is the unit of sparse representation. In order to classify similar image blocks into one group, Zhang et al. [5] used Euclidean distance to measure the similarity between image blocks.

Compared with manifold distance, Euclidean distance metric is low computational complexity, but it cannot well

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61402410 and C12412135, the National Natural Science Youth Foundation of China under Grant No.11501375, 61602307 and 11401388, and the Zhejiang Provincial Natural Science Foundation of China under Grant No. LQ14A010005, LY13F020029, LQ14F020004 and Q14A010009.

Shuhua Xu is with College of Computer Science & Technology, Zhejiang University of Technology, Hang Zhou 310000, Zhejiang, China, and with School of Mathematics, Physics and Information Science, Shao Xing University, Shao Xing 312000, Zhejiang, China.

Fei Gao is with College of Computer Science & Technology, Zhejiang University of Technology, Hang Zhou 310000, Zhejiang, China (corresponding author; e-mail: webqmm1974@163.com).

reflect non-linear relationship between image blocks. Based on above analysis, this paper proposes group sparse representation based single image super-resolution algorithm via Gauss metric on the basis of the existing group sparse representation(GSR). In order to measure better the distance or similar between image blocks, the algorithm uses Gauss kernel distance to obtain nonlinear information between image blocks, making full use of the natural image space geometry to capture effectively local similarities and differences between image blocks. Experiments on FRGC Ver1.0 natural image sets show that the proposed algorithm has better robustness and super-resolution performance in both PSNR and visual perception.

The remainder of this paper is organized as follows. Single image super-resolution, traditional patch-based sparse representation and traditional patch-based sparse representation based single image super-resolution are introduced in Section II. Section III proposes single image super-resolution of group-based sparse representation (GSR) modeling. Extensive experimental results are reported in Section IV. In Section V, we summarize this paper.

II. RELATED WORK

A. Single image Super-resolution

The task of single image super-resolution is to restore one HR image from one input LR image through an operation. Given an observed image y , the model of image super resolution is defined as:

$$y = Hx + v \quad (1)$$

Where the degradation matrix H is a composite operator of blurring and down-sampling, x is the original image, and v is the noise term. In past decades, researchers have proposed a lot of single image super-resolution algorithms. Due to the ill-posed nature of the super-resolution inverse problem, the regularization method is introduced to eliminate uncertainty of image restoration. A lot of regularization based several techniques have been proposed in recent literatures[5,9,10]. Typical regularization models include total variation (TV) regularization[9], non-local similarity[10] and sparse regularization term [5]. Total variation (TV) regularization is introduced in image processing, and applied successfully to the solution of inverse problem. Due to TV regularization assuming the piecewise constant region, TV regularization[9] is more suitable for super resolution reconstruction of smooth images.

In order to restore the discontinuity or spatially inhomogeneous images, researchers propose sparse regularization, which is applied to super resolution image reconstruction[4]. For one LR image $y \in R^M$, let $x \in R^N$ and $\hat{x} \in R^N$ denote respectively HR image and reconstructed HR image. Accordingly, $y_i \in R^m$, $x_i \in R^n$, $\hat{x}_i \in R^n$ ($i=1, \dots, l$) represent respectively LR, HR and reconstructed HR image blocks, where l is total number of overlapping image blocks.

Assuming that the matrix $R_i \in R^{n \times N}$ can extract image blocks from one image, so HR image blocks can be expressed as: $x_i = R_i x$. According to the observation model (1), the primary task of super resolution (SR) is to solve sparse coding a of y on dictionary Φ , which is defined as:

$$a_y = \arg \min_a \{ \|y - H\Phi a\|_2^2 + \lambda \|a\|_1 \} \quad (2)$$

Where the Lagrange multiplier λ is used to balance the tradeoff between fitting data perfectly and employing one sparse solution, and $\|a\|_1$ is the sparsity-inducing term. HR image x can be reconstructed by the following formula (3), and the reconstructed image \hat{x} can be obtained.

$$\hat{x} = \Phi \circ a_y = \left(\sum_{i=1}^l R_i^T R_i \right)^{-1} \sum_{i=1}^l (R_i^T \Phi a_{y,i}) \quad (3)$$

Where $\Phi \in R^{N \times M}$ ($N < M$) is an over-complete dictionary, most of sparse coding vector a are zero or close to zero. The purpose of introduction of symbol \circ is that the representation of $\Phi \circ a_y$ is more convenient than that of

$$\left(\sum_{i=1}^l R_i^T R_i \right)^{-1} \sum_{i=1}^l (R_i^T \Phi a_{y,i}).$$

The choice of dictionary is very important for sparse representation model. There are two main types of dictionaries for selection: analytical dictionaries and learning dictionaries.

B. Traditional Patch based Sparse Representation

Recently, sparse representation based model in image processing has been proved to be a promising model[7], which assumes that in some areas, natural images are sparse, and can be made by a group of bases or atoms of dictionary. In literatures, the basic unit of image sparse representation is image block. Let $x \in R^N$ and x_i ($i=1, 2, \dots, l$) denote respectively original image, image block with the size of $\sqrt{n} \times \sqrt{n}$, where $i=1, 2, \dots, l$ represents the location of image blocks, N is the size of image, n is the size of image block, and l is the total number of image blocks. Image blocks can be obtained by the following formula (4):

$$x_i = R_i(x) \quad (4)$$

Where $R_i(\cdot)$ is an operator extracting image blocks x_i from image x , transpose operation of $R_i(\cdot)$ is denoted as $R_i^T(\cdot)$, which can restore image block x_i to the i position of image x , the rest of which are padded with zeros in the reconstruction of HR image. Since there are usually overlapping parts between image blocks, reconstructed image from image blocks $\{x_i\}$ can be described as follows:

$$x = \sum_{i=1}^l R_i^T(x_i) ./ \sum_{i=1}^l R_i^T(1_n) \quad (5)$$

Where $./$ is on behalf of corresponding element division

of two vectors, vector 1_n is the same size as vector n , its value is 1, equation (5) is a kind of abstract mathematics for the average of all the overlapping image blocks.

Given one dictionary $\Phi \in R^{n \times s}$, where s is the total number of atoms in dictionary D , each image block can be coded sparsely as $\alpha_i = \Phi^T x_i$ by using sparse coding algorithm, the majority of coefficients in vector α_i are zero or close to zero. The whole image can be represented sparsely by a set of sparse encoding coefficients. In fact, α_i can be solved by the following equation (6):

$$\alpha_i = \arg \min_{\alpha} \frac{1}{2} \|x_i - D\alpha\|_2^2 + \lambda \|\alpha\|_p \quad (6)$$

Where, λ is one constant, p is 0 or 1. If $p=0$, the sparsity is represented by the norm l_0 of α_i , which is the number of non-zero elements in the vector α_i . However, the norm l_0 optimization problem is one non convex and NP hard problem, which can be solved by using the greedy algorithm such as orthogonal matching pursuit (OMP) algorithm [11]. If $p=1$, the norm l_0 optimization problem is similar to the convex norm l_1 , which can be solved by the large scale tool, which is recently proposed [8].

Similar to equation (5), the formula of reconstructed image x from sparse coding $\{\alpha_i\}$ of image x is as follows:

$$x = \Phi \circ \alpha = \sum_{i=1}^l R_i^T (\Phi \alpha_i) / \sum_{i=1}^l R_i^T (1_n) \quad (7)$$

Where α are concatenated by all α_i , i.e. $\alpha = [\alpha_1^T, \alpha_2^T, \dots, \alpha_l^T]^T$.

C. Patch-based Sparse Representation Single Image Super Resolution

According to the degradation equation (1), the reconstruction formula using traditional patch-based sparse representation single image super-resolution is as follows:

$$\hat{\alpha} = \arg \min_{\alpha} \frac{1}{2} \|H\Phi \circ \alpha - y\|_2^2 + \lambda \|\alpha\|_p \quad (8)$$

Where λ is regularization parameter, p is 0 or 1. The reconstructed image x using $\hat{\alpha}$ can be represented as: $\hat{x} = \Phi \circ \hat{\alpha}$.

The core of sparse representation model is the choice of dictionary. In other words, finding the best domain to sparsify one given image is very important. A redundant dictionary is usually learned from a set of training image blocks $X = [x_1, x_2, \dots, x_l]$, where l is the number of training image blocks. The goal of dictionary learning is to optimize dictionary Φ and representation coefficient matrix $A = [\alpha_1, \alpha_2, \dots, \alpha_l]$ together to let $x_k \approx \Phi \alpha_k$, where $\|\alpha_k\|_p \leq L$ and p is 0 or 1. Dictionary learning can be expressed in mathematical formula (9):

$$(\hat{\Phi}, \hat{A}) = \arg \min_{\Phi, A} \sum_{i=1}^l \|x_i - \Phi \alpha_i\|_2^2 \quad s.t. \|\alpha_i\|_p \leq L, \forall i \quad (9)$$

Obviously, even when p is 1, the solution of minimization problem of the above formula (9) is large-scale and highly non-convex. In order to solve the problem easy, some approximation methods, such as DGSDL [6], are proposed to optimize alternately the dictionary Φ and the representation coefficient matrix A , and improve results of image processing.

III. GROUP-BASED SPARSE REPRESENTATION VIA GAUSSIAN SINGLE IMAGE SUPER RESOLUTION

A. Basic Idea

This is a tendency to using sparse regularization based super-resolution to reconstruct one single image, which not only makes full use of the sparsity of image, but also can improve further the effect of super resolution reconstruction. But the traditional sparse regularization based single image super-resolution reconstruction is patch-based sparse regularization, there exist some shortcomings as follows:

(1) Large-scale optimization problems must be solved in highly redundant dictionary learning process. In order to make the solution more simple and convenient, researchers put forward some approximation methods, such as DGSDL [6], to optimize alternately the dictionary Φ and sparse coding α , and get good results in image processing. However, it is inevitable to require highly computational complexity by using these approximation methods to learn redundant dictionaries.

(2) Researchers also noted that, in the dictionary learning and sparse encoding stage, each patch in the image is actually considered separately, ignoring the relationship between patches with similar nature, such as self-similarity [7], resulting in inaccurate sparse encoding coefficients.

In order to solve these two problems, the literature [5] restored the image using group-based sparse regularization (GSR) model instead of patch-based sparse regularization model, the purpose of which is to use local sparse and nonlocal self-similarity of the image in a unified framework. However, GSR used the nearest neighbor sparse reconstruction to capture the local structure information of the image, and effectively improved deblurring effect of the image. But, GSR still has some shortcomings, and use Euclidean distance classifying patches with local similarity into one group. Euclidean distance only can get difference of pixels between two patches, which lacked consideration of image space, and is very sensitive to the deformation of patches. So, if patches exist the distortion or Euclidean distance between the two patches is too large, the result of patches grouping are inaccuracy.

At present, newly proposed Gauss kernel distance metric calculation method can obtain the nonlinear information between data sets, measuring the distance or similar between samples, and making full use of the image spatial relationships to effectively capture the similarity between images. Compared

with other nonlinear distance measure algorithms, the computation complexity of the Gaussian kernel distance metric is small and the performance of which is more stable [12].

To this end, this paper proposes GSR-based single image super-resolution with Gaussian algorithm in the base of the existing GSR using the Gaussian kernel distance to measure the distance between patches. Our approach contains three phases as follows:

(1) The first stage obtains Gaussian distance matrix between patches employing Gaussian distance, and then classify patches into groups by Gaussian distance matrix.

(2) The second stage uses effectively adaptive dictionary learning methods for learning patches in each group, and gets their dictionary atoms which are employed to encode sparsely patches in each group.

(3) Finally, the entire image is reconstructed by super resolution in the domain of the group.

B. Algorithm Steps

Let $x \in \mathbb{R}^N$ denotes the original high resolution image, which is processed by fuzzy and down sampling operation to obtain the degraded image y with low resolution as input data.

The specific implementation process of algorithm is as follows:

(1) Segmenting the image into groups with similar structure, as shown in figure 1. First, the low resolution image y with the size of N is divided into l overlapping patches, the size of each patch is $\sqrt{n} \times \sqrt{n}$, which is described as a vector $y_i \in \mathbb{R}^n, i=1,2,\dots,l$. Image patches of a training window $w \times w$ are denoted into a set of $S_{y_{blue}}$, the number of elements in which is denoted as l_w , shown as the blue box in Fig.1. For each patch y_i , as shown as the red box in Figure 1, h adjacent patches of y_i are selected as the Gaussian kernel distance measure formula (10):

$$d_G(y_i, y_j) = \exp\left(-\frac{\|y_i - y_j\|^2}{2\sigma^2}\right) \quad (10)$$

where $y_j \in S_{y_{blue}} (j \neq i)$, $\sigma = \frac{1}{l_w^2} \sum_{i,j} (y_i - y_j)^2$ is the width of

the Gaussian kernel distance, which is equal to the mean value of the Euclidean distance among image patches in a training window $w \times w$. h adjacent patches of y_i are denoted into a set as S_{y_i} . h patches with similar structures in S_{y_i} are represented as a matrix with the size of $n \times h$, and are composed of a group, denoted as $y_{G_i} \in \mathbb{R}^{n \times h}$, each patch in S_{y_i} is formed as a column of y_{G_i} , that is, $y_{G_i} = \{y_{G_i \otimes 1}, y_{G_i \otimes 2}, \dots, y_{G_i \otimes h}\}$. Constructing a group from the image y is defined as the formula (11):

$$y_{G_i} = P_{G_i}(y) \quad (11)$$

Where $P_{G_i}(\cdot)$ is an operator of forming the group y_{G_i} from the image y . The $P_{G_i}(\cdot)^T$ of $P_{G_i}(\cdot)$ transpose can return i th group to the position i of the image y , and any else except the position i is all with zero padding.

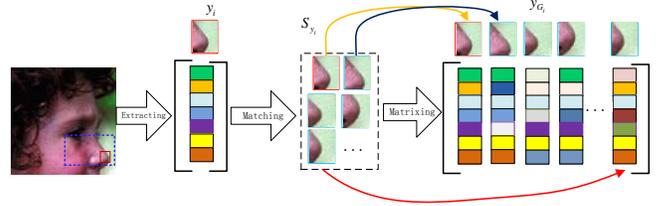


Fig. 1. Illustrations for classifying patches into one group

A complete image can be recovered from $\{y_{G_i}\}$ by averaging all the groups, as shown in the formula (12).

$$y = \sum_{i=1}^l P_{G_i}^T(y_{G_i}) \cdot \frac{1}{\sum_{i=1}^l P_{G_i}^T(1_{n \times h})} \quad (12)$$

Where, $\cdot /$ is on behalf of element division in the corresponding position of two vectors, every element value of matrix $1_{n \times h}$ with the size of $n \times h$ is 1.

(2) Learning the adaptive dictionary for $i(i=1,2,\dots,l)$ group by singular value decomposition (SVD). We utilize SVD to learn an adaptive dictionary for group y_{G_i} , and obtain the estimated value a_{G_i} of y_{G_i} , as shown in the formula (13).

$$a_{G_i} = U_{G_i} \sum_{G_i} V_{G_i}^T = \sum_{i=1}^m A_{a_{G_i \otimes i}}(u_{G_i \otimes i} v_{G_i \otimes i}^T) \quad (13)$$

Where U_{G_i}, V_{G_i} are respectively left and right singular value vectors of singular values in y_{G_i} , \sum_{G_i} is a diagonal matrix whose principal diagonal element is singular values, s is the number of principal singular values which are denoted as $A_{a_{G_i}} = [A_{a_{G_i \otimes 1}}; A_{a_{G_i \otimes 2}}; \dots; A_{a_{G_i \otimes s}}]$. $u_{G_i \otimes i}, v_{G_i \otimes i}$ are respectively column elements of U_{G_i}, V_{G_i} . The atoms $\phi_{G_i \otimes i}$ in dictionary Φ_{G_i} of group y_{G_i} are defined as shown in the formula (14).

$$\phi_{G_i \otimes i} = u_{G_i \otimes i} v_{G_i \otimes i}^T, i=1,2,\dots,s \quad (14)$$

Where, $\phi_{G_i \otimes i} \in \mathbb{R}^{n \times h}$. Finally, the adaptive learning dictionary Φ_{G_i} of the group y_{G_i} is shown in the formula (15).

$$\Phi_{G_i} = [\phi_{G_i \otimes 1}, \phi_{G_i \otimes 2}, \dots, \phi_{G_i \otimes s}] \quad (15)$$

(3) coding sparsely the i th ($i=1,2,\dots,l$) group for its sparse encoding α_{G_i} on the basis of dictionaries Φ_{G_i} of the

i th group. The sparse encoding α_{G_i} of the i th group y_{G_i} can be represented by a formula (16).

$$\hat{\alpha}_{G_i} = \arg \min_{\alpha_{G_i}} \frac{1}{2} \|y_{G_i} - \Phi_{G_i} \alpha_{G_i}\|_2^2 + \lambda \|\alpha_{G_i}\|_0 \quad (16)$$

Where λ is the sparse regularization parameter. Since the l_0 optimization problem is non-convex, and NP hard problem. l_1 Optimization is usually used to approximate the l_0 optimization problem. From a technical point of view, the l_1 optimization problem is equivalent to the l_0 optimization problem under certain constraints. In this paper, the SBI algorithm [9] is used to solve the l_1 optimization problem. The objective function of reconstructing the i th group is shown in the formula (17).

$$\min_{\alpha_{G_i}, x_{G_i}} \frac{1}{2} \|Hx_{G_i} - y_{G_i}\|_2^2 + \lambda \|\alpha_{G_i}\|_0, \text{ s.t. } x_{G_i} = \Phi_{G_i} \circ \alpha_{G_i} \quad (17)$$

SBI algorithm divides the solution of objective function into two sub problems, one is the solution of x_{G_i} , the other is the solution of α_{G_i} . Iterative solution algorithm steps are as shown in Algorithm1.

Algorithm 1

① Setting initial value of variables and illustrating parameters

$$t = 0, \mu > 0, u^{(t)} = A_{\alpha_{G_i}}, b^{(t)} = 0, \lambda > 0, o \text{ is the number of}$$

overlapping pixels between patches, h is the number of patches in one group,

$$\tau = \lambda \times h \times o / \mu, \alpha_{G_i}^{(t)} = \text{hard}(A_{\alpha_{G_i}}, \sqrt{\delta\tau}) = A_{\alpha_{G_i}} \odot 1_{(abs(A_{\alpha_{G_i}}) - \sqrt{\delta\tau})},$$

where $\sqrt{\delta\tau}$ is a threshold, $\text{hard}(\cdot)$ is one hard-threshold operator, \odot is on behalf of the corresponding elements multiplication in two vectors, $1_{(abs(A_{\alpha_{G_i}}) - \sqrt{\delta\tau})}$ is a vector with the same size as vector $A_{\alpha_{G_i}}$, the corresponding position on the vector $A_{\alpha_{G_i}}$ is set as 1, otherwise as 0 when the absolute value of which in $A_{\alpha_{G_i}}$ is greater than the threshold $\sqrt{\delta\tau}$.

② Repeat

$$\textcircled{3} u^{(t+1)} = \arg \min_u \frac{1}{2} \|Hu^{(t)} - y\|_2^2 + \frac{\mu}{2} \|u^{(t)} - \Phi_{G_i} \circ \alpha_{G_i}^{(t)} - b^{(t)}\|_2^2 \quad (18)$$

$$\textcircled{4} \alpha_{G_i}^{(t+1)} = \arg \min_{\alpha_{G_i}} \lambda \|\alpha_{G_i}\|_0 + \frac{\mu}{2} \|u^{(t+1)} - \Phi_{G_i} \circ \alpha_{G_i}^{(t)} - b^{(t)}\|_2^2 \quad (19)$$

$$\textcircled{5} b^{(t+1)} = b^{(t)} - (u^{(t+1)} - \Phi_{G_i} \circ \alpha_{G_i}^{(t+1)})$$

$$\textcircled{6} t = t + 1$$

⑦ Until the maximum number of iterations reached or the iterative stopping condition satisfied.

Equation (18) is a convex quadratic function, whose solution is as follow as the formula (20).

$$\hat{u} = (H^T H + \mu I)^{-1} (H^T y + \mu(\Phi_{G_i} \circ \alpha_{G_i}^{(t)} + b^{(t)})) \quad (20)$$

Where I is the unit matrix.

The solution of the equation (19) can be solved by the formula (21).

$$\hat{\alpha}_{G_i} = \text{hard}(\hat{u}, \sqrt{\delta\tau}) = u \odot 1_{(abs(\hat{u}) - \sqrt{\delta\tau})} \quad (21)$$

(4) Restoring and reconstructing the i th group of patches $\hat{x}_{G_i} = \Phi_{G_i} \circ \hat{\alpha}_{G_i}$. The rest group of patches are employing the same technology for super resolution reconstruction. A complete image can be recovered from $\{\hat{x}_{G_i}\}$ through averaging patches of all the groups, as shown in the formula (22).

$$\hat{x} = \sum_{i=1}^l P_{G_i}^T(x_{G_i}) \cdot \left/ \sum_{i=1}^l P_{G_i}^T(1_{n \times h}) \right. \quad (22)$$

Which is similar to the formula (11). Where $\cdot /$ is on behalf of element division in the corresponding position of two vectors, every element value of matrix $1_{n \times h}$ with the size of $n \times h$ is 1. Finally, the reconstructed image is the high resolution image \hat{x} .

IV. EXPERIMENTAL RESULT AND ANALYSIS

A. Experimental Dataset

Test images used in the experiment, include butterfly, vessels, flowers, leaves, raccoon, buildings, goldfish, house, Lena, peppers, windows, parrot, girl, bicycle, hat, coast guard, bridge, tiger, zebra, foreman and so on, some of which are shown in Fig. 2. For color images, single image super-resolution reconstruction is applied only to the luminance component.

B. Experimental Settings

In this paper, experimental parameters are set as follows: group size is 64×60 , patch size n is 64, the number of image patches in one group is 60, overlapping pixels between adjacent patches are 4 pixels. The size of training window $w \times w$ constructing one group is setting to 20×20 . All test images are first blurred by a 25×25 Gaussian filter with standard deviation $\sigma = 1.6$, and then downsampled by a decimation factor of $q = 2$ to produce the corresponding LR images. All experiments are conducted in MATLAB 7.11.0 software on a desktop PC with 2.60GHz Dual-Core CPU and 8.25G memory.

All experiments are repeated one hundred times and the average PSNR and SSIM values are taken as the final PSNR and SSIM.



Fig. 2. Some test images in experiment

C. Experimental Result and Analysis

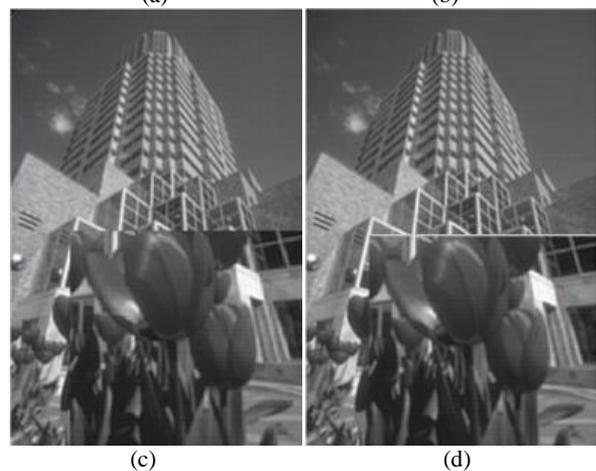
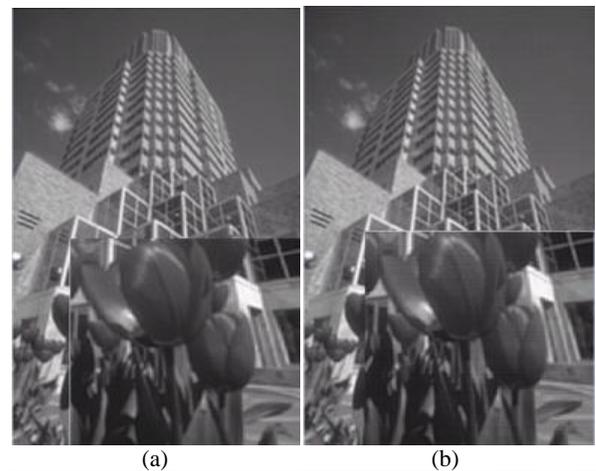
In order to estimate exactly the performance of GSRGSiSR algorithm, the effect of threshold parameter δ on GSRGSiSR is firstly discussed and then experimental result and detail analysis are given respectively on the different number of h nearest neighbors for MoE [13], LRT_SR[3] and GSRGSiSR algorithms.

1) Threshold parameter δ selection

In order to obtain the optimal value of threshold parameter δ , 6 test images are selects, and on which the different threshold parameter δ are selected for the experiment, μ is set to 0.005, and λ is set to 0.7532. Threshold parameter δ increases from 10 to 60 with the increment 10 and corresponding PSNR and SSIM are calculated. Concrete experimental results are showed in Table 1 and Fig.3.

Table 1 PSNR and SSIM results of reconstructed images by different values of δ

Values of δ	Measures	Images						
		Boats	Building	Bike	Leaf	Window	Parr ot	Avg.
$\delta = 10$	PSNR	30.01	25.45	24.22	38.69	29.86	31.03	29.88
	SSIM	0.8822	0.8754	0.8201	0.9553	0.8851	0.9341	0.8922
$\delta = 20$	PSNR	29.93	25.10	23.94	38.64	29.05	29.08	29.29
	SSIM	0.8791	0.8531	0.8080	0.9553	0.8616	0.9211	0.8797
$\delta = 30$	PSNR	29.92	25.06	23.86	38.63	29.03	29.82	29.39
	SSIM	0.8788	0.8521	0.8052	0.9553	0.8611	0.9202	0.8788
$\delta = 40$	PSNR	29.92	25.04	23.82	38.63	29.03	29.79	29.37
	SSIM	0.8787	0.8517	0.8040	0.9553	0.8610	0.9198	0.8784
$\delta = 50$	PSNR	29.92	25.04	23.80	38.63	29.02	29.79	29.37
	SSIM	0.8787	0.8515	0.8032	0.9553	0.8610	0.9198	0.8783
$\delta = 60$	PSNR	29.91	25.03	23.79	38.63	29.02	29.78	29.36
	SSIM	0.8767	0.8514	0.8029	0.9553	0.8609	0.9198	0.8778



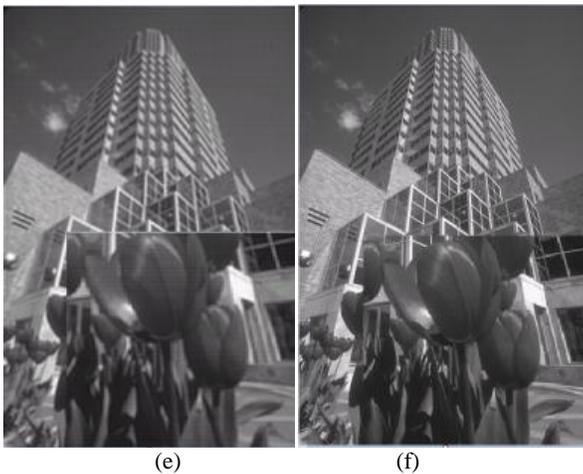


Fig.3. The reconstructed results by different threshold value δ on "Building". (a) $\delta = 60$ (PSNR=25.03, SSIM=0.8514). (b) $\delta = 50$ (PSNR=25.04, SSIM=0.8515). (c) $\delta = 40$ (PSNR=25.04, SSIM=0.8517). (d) $\delta = 30$ (PSNR=25.06, SSIM=0.8521). (e) $\delta = 20$ (PSNR=25.10, SSIM=0.8531). (f) $\delta = 10$ (PSNR=25.45, SSIM=0.8754).

As can be seen from Table 1, with increment in threshold parameter δ , the values of PSNR and SSIM fluctuation, and even decrease. The values of PSNR and SSIM reach most high when δ is set to 10. From Fig.3., we can see that the edge of reconstructed image becomes clearer with decreasing threshold parameter δ . When threshold parameter δ is set to 60, 50, 40, 30 respectively, the edge contour of reconstructed image changes little. Therefore, considering both objective and subjective effects, we adopt 10 as the value of threshold parameter δ .

2) Effect of the size of patch every group

In order to test the effect of patch size on the quality of reconstructed image in every group, we conduct experiments on different patch sizes, namely 3*3, 5*5, 7*7 and 9*9. The threshold parameter δ is set to 10. Experimental results are shown in Table 2 and Fig.4.

Table 2. PSNR and SSIM results of reconstructed images by different patch sizes of one group

Images	Measures	different patch sizes of one group			
		3*3	5*5	7*7	9*9
Boats	PSNR	27.79	30.01	28.64	28.24
	SSIM	0.8581	0.8822	0.8817	0.8803
Buildin g	PSNR	24.84	25.45	25.24	25.23
	SSIM	0.8523	0.8754	0.8551	0.8550
Bike	PSNR	23.98	24.22	24.11	23.90
	SSIM	0.8141	0.8201	0.8141	0.8135
Leaf	PSNR	38.06	38.69	38.69	38.64
	SSIM	0.9536	0.9553	0.9548	0.9543
Window	PSNR	28.73	29.86	29.16	29.13

	SSIM	0.8643	0.8851	0.8650	0.8647
Parrot	PSNR	27.13	31.03	26.69	26.41
	SSIM	0.9235	0.9341	0.9232	0.9008
Avg.	PSNR	28.42	29.88	28.76	28.59
	SSIM	0.8777	0.892	0.8823	0.8781

As can be seen from Table 2 and Fig.4., with increment in patch sizes, the values of PSNR and SSIM fluctuation, and even decrease. The values of PSNR and SSIM reach most high when patch size is set to 5*5. Therefore, considering both objective and subjective effects, we adopt 5*5 as the value of patch sizes.

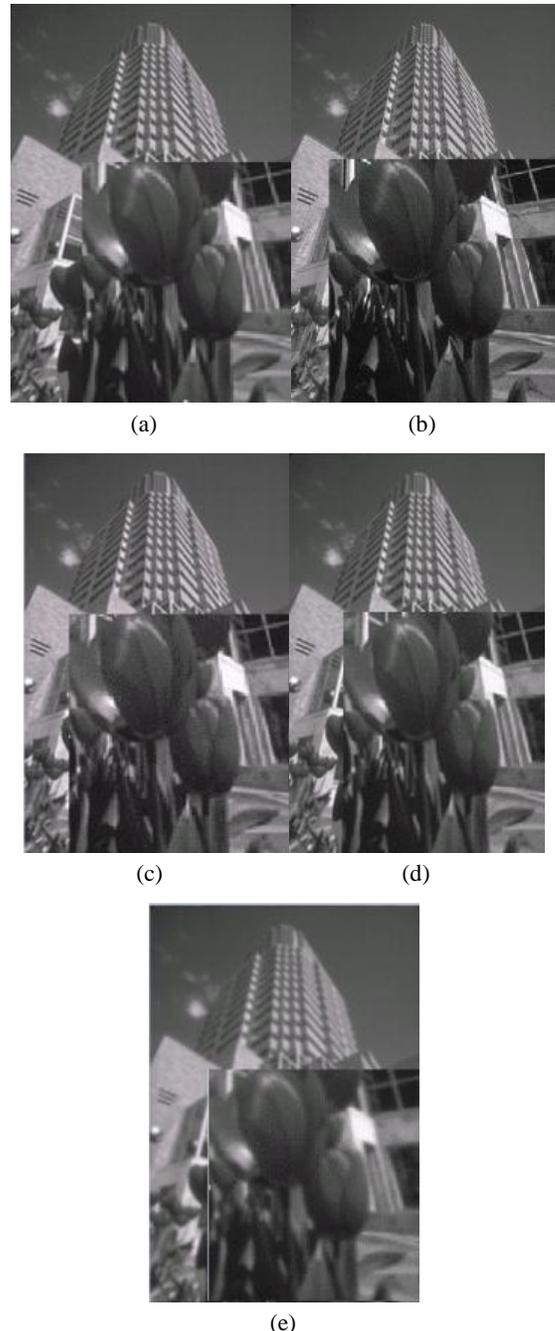


Fig.4. The reconstructed results by different patch sizes on "Building". (a) 3*3 (PSNR=24.84, SSIM=0.8523) (b) 5*5

(PSNR=25.45, SSIM=0.8754) (c)7*7 (PSNR=25.24,SSIM=0.8551)
 (d)9*9 (PSNR=25.23,SSIM=0.8550)(e) LR image

3) Effect of the size of training window every group

In order to test the effect of training window size on the quality of reconstructed image, we conduct experiments on different training window sizes, namely 19, 20, 21 and 22. The threshold parameter δ and patch size are set to 10 and 5*5 respectively. Experimental results are shown in Table 3 and Fig.5.

Table 3 PSNR and SSIM results of reconstructed images by the different size of training window of one group

Images	Measures	the different size of training window of one group			
		19	20	21	22
Boats	PSNR	29.96	30.01	29.95	29.94
	SSIM	0.8821	0.8822	0.8805	0.8805
Buildin g	PSNR	25.18	25.45	25.22	25.21
	SSIM	0.8549	0.8754	0.8562	0.8558
Bike	PSNR	24.09	24.22	24.09	24.08
	SSIM	0.8143	0.8201	0.8141	0.8139
Leaf	PSNR	38.67	38.69	38.68	38.68
	SSIM	0.9553	0.9554	0.9554	0.9554
Window	PSNR	29.13	29.86	29.13	29.13
	SSIM	0.8641	0.8851	0.8645	0.8647
Parrot	PSNR	29.98	31.03	29.97	29.95
	SSIM	0.9237	0.9341	0.9232	0.9232
Avg.	PSNR	29.5	29.88	29.51	29.5
	SSIM	0.8824	0.892	0.8823	0.8823



(a)

(b)



(c)

(d)



(e)

Fig.5. The reconstructed results by different training window sizes on "Window". (a)19 (PSNR=29.13, SSIM=0.8641) (b)20 (PSNR=29.86, SSIM=0.8651) (c)21 (PSNR=29.13, SSIM=0.8645) (d)22(PSNR=29.13,SSIM=0.8647)(e) LR image

As can be seen from Table 3 and Fig.5., with increment in training window sizes, the values of PSNR and SSIM fluctuation, and even decrease. The values of PSNR and SSIM reach most high when training window size is set to 20. Therefore, considering both objective and subjective effects, we adopt 20 as the value of training window size.

4)Effect of the different number of patches every group

In order to test the effect of different number of patches every group on the quality of reconstructed image, we conduct experiments on different number of patches every group, namely 70, 80, 90 and 100. The threshold parameter δ , patch size and training window size are set to 10, 5*5 and 20 respectively. Experimental results are shown in Table 4 and Fig.6.

Table 4 PSNR and SSIM results of reconstructed images by the different number of patches every group

Images	Measures	the different number of patches every group			
		70	80	90	100
Boats	PSNR	29.86	29.92	30.01	29.98
	SSIM	0.8791	0.8800	0.8822	0.8807
Buildin g	PSNR	25.19	25.25	25.45	25.27
	SSIM	0.8559	0.8562	0.8754	0.8570
Bike	PSNR	24.09	24.12	24.22	24.23
	SSIM	0.8141	0.8152	0.8201	0.8221
Leaf	PSNR	38.67	38.68	38.71	38.67
	SSIM	0.9553	0.9556	0.9558	0.9553
Window	PSNR	29.10	29.28	29.86	29.25
	SSIM	0.8641	0.8712	0.8851	0.8688
Parrot	PSNR	29.85	29.98	31.03	30.01
	SSIM	0.9218	0.9232	0.9341	0.9235
Avg.	PSNR	29.46	29.54	29.88	29.57
	SSIM	0.8817	0.8836	0.8921	0.8846

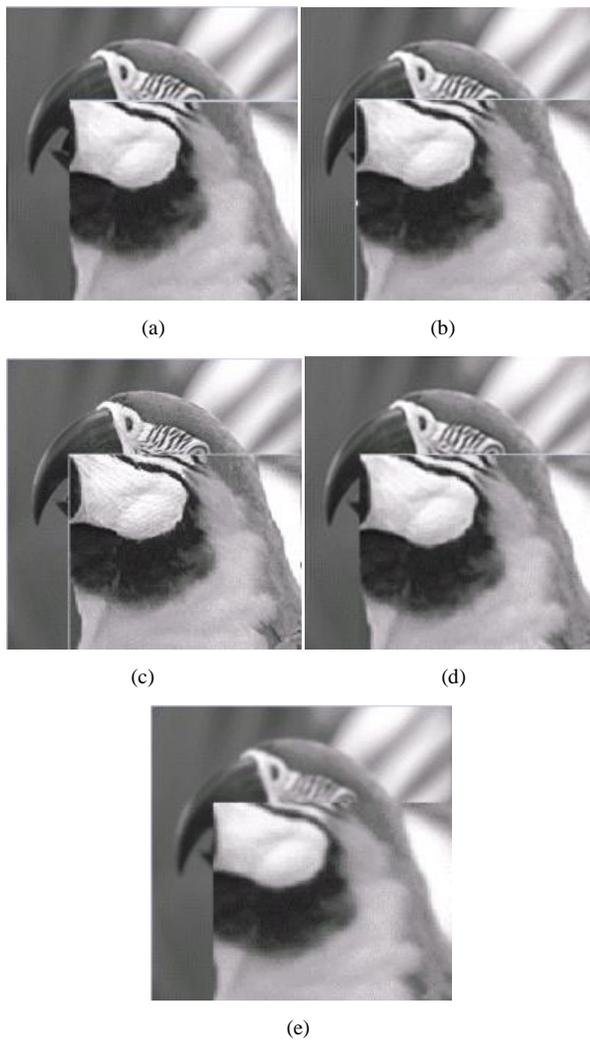


Fig.6. The reconstructed results by different number of patches every group on " Parrot". (a) 70 (PSNR=29.85, SSIM=0.9218) (b) 80 (PSNR=29.98,SSIM=0.9232) (c) 90 (**PSNR=31.03,SSIM=0.9341**) (d) 100(PSNR=30.01,SSIM=0.9235) (e) LR image.

As can be seen from Table 4 and Fig.6., with increment in the number of patches every group, the values of PSNR and SSIM fluctuation, and even decrease. The values of PSNR and SSIM reach most high when the number of patches every group is set to 90. Therefore, considering both objective and subjective effects, we adopt 90 as the number of patches every group.

5) Reconstructed image results by different methods

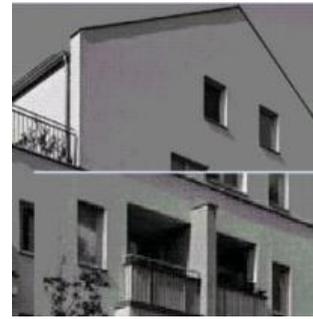
In order to display the visual effects of different methods reconstructing images, experimental results on reconstructed images and segmented local regions are shown in Table 5 and Fig.7.

Table 5PSNR and SSIM results of reconstructed images by different methods

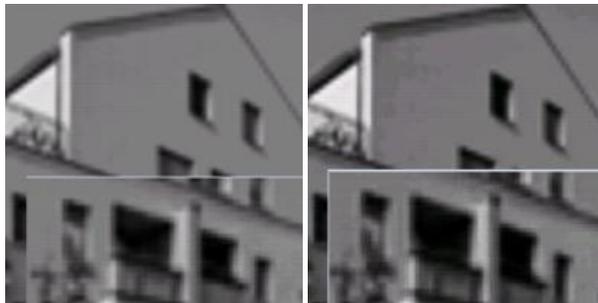
Images	Measures	Methods						
		Bicubic	SCSR	Zeyde's	NA RM	LRT_SR	MoE	Proposed
Boat	PSNR	27.8	29.	29.2	27.5	29.4	29.9	30.01

s	R	7	25	4	3	5	5	
	SSI M	0.81 42	0.8 542	0.85 30	0.78 24	0.86 06	0.87 17	0.882 2
Starfish	PSN R	26.5 3	27. 82	27.7 4	26.0 4	28.4 9	28.5 2	28.92
	SSI M	0.77 29	0.8 237	0.81 82	0.73 11	0.84 02	0.85 62	0.883 5
Houset	PSN R	24.7 5	24. 85	25.5 5	25.8 2	26.1 7	26.3 7	26.61
	SSI M	0.75 31	0.7 881	0.87 57	0.87 85	0.88 10	0.89 99	0.906 4
Lena	PSN R	29.4 8	30. 88	30.7 6	29.3 8	31.3 6	31.5 1	31.48
	SSI M	0.78 57	0.8 201	0.81 71	0.76 15	0.83 06	0.91 3	0.927 7
Window	PSN R	27.1 5	28. 48	28.3 5	26.6 9	28.7 8	29.1 8	29.86
	SSI M	0.76 44	0.8 174	0.81 20	0.73 03	0.82 79	0.84 13	0.885 1
Parrot	PSN R	27.4 5	29. 11	28.9 9	27.5 7	29.5 3	29.8 1	31.03
	SSI M	0.85 28	0.8 826	0.87 90	0.83 37	0.88 68	0.89 32	0.934 1
Butterfly	PSN R	23.8 5	26. 31	26.0 6	24.7 9	27.1 4	26.8 7	26.99
	SSI M	0.79 85	0.8 748	0.86 51	0.83 7	0.89 34	0.88 30	0.914 684
Flower	PSN R	34.2 1	36. 24	36.3 5	32.4 1	36.6 4	36.9 4	36.67
	SSI M	0.90 37	0.9 181	0.91 88	0.86 09	0.92 21	0.93 65	0.928 3
Plants	PSN R	30.7 5	32. 58	32.3 2	30.3 6	32.9 9	33.0 5	33.10
	SSI M	0.84 63	0.8 547	0.85 36	0.80 39	0.86 09	0.91 98	0.924 7
Leaf	PSN R	36.8 8	38. 04	38.1 8	35.3 1	38.4 0	38.5 7	38.69
	SSI M	0.88 42	0.8 940	0.89 42	0.84 94	0.89 81	0.91 13	0.955 3
Girl	PSN R	33.0 1	33. 90	33.8 9	30.4 8	34.0 3	33.8 0	35.24
	SSI M	0.76 06	0.7 836	0.78 14	0.59 55	0.78 55	0.75 4	0.880 2
Bike	PSN R	22.6 3	23. 92	23.7 8	22.7 4	24.2 8	24.1 7	24.22
	SSI M	0.67 46	0.7 561	0.74 61	0.65 64	0.77 21	0.79 21	0.820 1
Raccoon	PSN R	28.3 7	28. 91	28.9 2	27.5 9	29.1 6	29.6 1	29.92
	SSI M	0.70 01	0.7 402	0.73 51	0.62 51	0.74 18	0.80 89	0.824 7
Leaves	PSN R	23.3 1	25. 64	25.5 5	24.1 1	26.4 4	26.3 4	26.18
	SSI M	0.79 98	0.8 818	0.87 57	0.83 31	0.90 18	0.90 02	0.894 3

Building	PSNR	23.53	24.36	24.33	23.66	24.82	25.06	25.45
	SSIM	0.6842	0.7445	0.7370	0.6722	0.7611	0.8111	0.8754
Peppers	PSNR	29.43	31.91	31.75	29.60	32.23	32.54	31.19
	SSIM	0.8482	0.8616	0.8667	0.8254	0.8691	0.906	0.8960
Avg.	PSNR	28.08	29.51	29.49	27.76	29.99	30.14	30.35
	SSIM	0.7902	0.831	0.833	0.7673	0.8458	0.8686	0.895793



(i) Original image



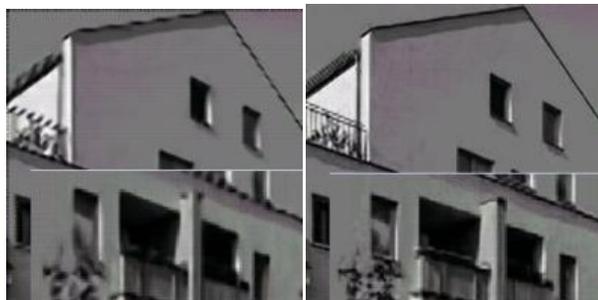
(a) LR image

(b) BI



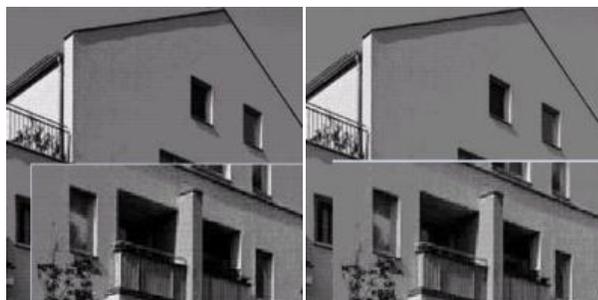
(c) SCSR

(d) Zeyde's



(e) NARM

(f) LRT_SR



(g) MoE

(h) The proposed

Fig.7. Comparison with results reconstructing on "house" image by different methods. (a)LR image (b) BI (PSNR = 24.75, SSIM = 0.7531). (c) SCSR [4] (PSNR = 24.85, SSIM = 0.7881). (d) Zeyde's [14] (PSNR = 25.55, SSIM = 0.8757). (e) NARM [1] (PSNR = 25.82, SSIM = 0.8785). (f) LRT_SR[3] (PSNR =26.17, SSIM = 0.8810). (g) MoE[13](PSNR = 26.37, SSIM = 0.8999). (h) The proposed (PSNR = **26.61**, SSIM = **0.9064**) (i) Original image

As can be seen from Fig.7, BI method has lost some important details of image, and the edge of which is blur. SCSR method recovers more details, but edges or details in the reconstructed image by which have some jaggy and ringing artifacts. Zeyde's method can suppress jaggy artifacts and sharp the edges, but it produces obvious ringing effects and leads to smooth reconstructed image. Although NARM method reconstructs the image relatively clear, the edges of which have obvious ringing effects. LRT_SR method can recover high-frequency details, and the contour edge of reconstructed image is quite clear. We can see that although the proposed method and MoE method can not only reconstruct fine details but also preserve correct edges. Also, an interesting observation is that by averaging the value of PSNR and SSIM of the super-resolved images of those two methods we can get a better result with average PSNR 30.35 dB and SSIM 89.58%, indicating that our method is somewhat complementary to MoE.

V. CONCLUSION

Failing to capture nonlinear nonlocal structure information and ignoring prior information in selecting the m nearest neighbors of every image patch as one group are main problems of existing group sparse representation algorithms. Therefore, Single-image Super Resolution based on Group Sparse Representation via GAUSSIAN (GSRGSiSR) is proposed on the basic of existing Group Sparse Representation in the paper. GSRGSiSR makes use of Gaussian kernel distances between image patches for analysis on similarity among them with these distances as group reconstruction, creating the group set of every image. Then GSRGSiSR designs an effective self-adaptive dictionary learning method for each group with low complexity, rather than dictionary learning from natural images, which achieves sparse representation of the image in the domain of group and gets their dictionary atoms which are employed to encode sparsely patches in each group. Finally, the entire image is reconstructed by super resolution in the domain of the group.

GSRGSiSR not only fuses performance of group sparse representation but also preserves nonlinear nonlocal self-similarity structure in the processing of single image super resolution. Experimental results on natural images demonstrate that the proposed GSRGSiSR method can get better PSNR and SSIM. Future work includes the extensions of GSR via Gaussian on a variety of applications, such as image deblurring and video restoration and so on.

REFERENCES

- [1] W. Dong, L. Zhang, R. Lukac, and G. Shi, "Sparse representation based image interpolation with nonlocal autoregressive modeling", *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp: 1382–1394, 2013.
- [2] C. Ren, X. He, Q. Teng, Y. Wu, and T.Q. Nguyen, "Single Image Super-Resolution Using Local Geometric Duality and Non-Local Similarity", *IEEE Transactions on Image Processing*, vol. 25, no. 5, pp. 2168–2183, 2016.
- [3] W. Gong, L. Hu, J. Li, and W. Li, "Combining sparse representation and local rank constraint for single image super resolution", *Information Sciences*, vol. 325, pp: 1-19, 2015.
- [4] J. Yang, J. Wright, T. Huang, and Y. Ma., "Image super-resolution via sparse representation", *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [5] J. Zhang, D. Zhao, and W. Gao, "Group-based Sparse Representation for Image Restoration", *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3336–3351, 2014.
- [6] Y. Sun, Q. Liu, J. Tang, and D. Tao, "Learning discriminative dictionary for group sparse representation", *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3816–3828, 2014.
- [7] Y. Zhang, J. Liu, W. Yang, and Z. Guo, "Image super-resolution based on structure-modulated sparse representation", *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2797–2810, 2015.
- [8] K. Zeng, G. Erus, A. Sotiras, R.T. Shinohara, and C. Davatzikos, "Abnormality detection via iterative deformable registration and basis-pursuit decomposition", *IEEE Transactions on Medical Imaging*, vol. 35, no. 8, pp. 1937–1951, 2016.
- [9] L. Li, Y. Xie, W. R. Hu, and W. S. Zhang, "Single image super-resolution using combined total variation regularization by split Bregman Iteration", *Neurocomputing*, vol. 142, pp. 551–560, 2014.
- [10] G. Vaksman, M. Zibulevsky, and M. Elad, "Patch-Ordering as a Regularization for Inverse Problems in Image Processing", *Siam Journal on Imaging Sciences*, vol. 9, no. 1, pp. 287–319, 2016.
- [11] A.P. Liao, Jiaxin, Xiaobo, Yang Peng Wang, THE EXACT RECOVERY OF SPARSE SIGNALS VIA ORTHOGONAL MATCHING PURSUIT, *Journal of Computational Mathematics*, 2016(1):70-86.
- [12] S. Xu, and F. Gao, "Weighted two-phase supervised sparse representation based on Gaussian for face recognition", *Discrete & Continuous Dynamical Systems - Series S*, vol. 6, no. 8, pp: 1385–1400, 2015.
- [13] K. Zhang, B. Wang, W. Zuo, and H. Zhang, "Joint Learning of Multiple Regressors for Single Image Super-Resolution", *IEEE SIGNAL PROCESSING LETTERS*, vol. 23, no. 1, pp: 102–106, 2016.
- [14] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations", in: *Proceedings of the 7th International Curves and Surfaces*, 2012, pp. 711–730.

Shuhua Xu was born on Jan. 30, 1977. She received the M.S. degree in computer science and technology from Guizhou University of China. Currently, she is a Ph.D. student of Computer Science & Technology at Zhejiang University of Technology and a lecturer of School of Mathematics, Physics and Information Science at Shaoxing University, China. Her major research interests include machine learning and image processing.

Fei Gao was born on Apr. 5, 1974. He received the PhD degree in mechanical engineering from Zhejiang University of China. Currently, he is a full Professor of College of Computer Science & Technology at the Zhejiang University of Technology, China. and a visiting scholar in industrial engineering at the Pennsylvania State University between November 1, 2007 and April 30, 2008. His current research interests include pattern recognition and image processing.