

RESEARCH

Open Access



Group-based single image super-resolution with online dictionary learning

Xuan Lu¹, Dingwen Wang^{2*} , Wenxuan Shi³ and Dexiang Deng¹

Abstract

Recently, sparse representation has been successfully used in single image super-resolution reconstruction. Unlike the traditional single image super-resolution methods such as image interpolation, the super-resolution with sparse representation reconstructs image with one or several constant dictionaries learned from external databases. However, the contents can vary significantly across different patches in a single image, and the fixed dictionaries cannot suit for every patch. This paper presents a novel approach for single image super-resolution based on sparse representation, which uses group as the basic unit, and trains dictionary with external database and the input low-resolution image itself for each group to ensure that the dictionary is suitable for the patches in the group. Simultaneous sparse coding algorithm is used to accelerate the processing and improve the result. Extensive experiments on natural images show that our method achieves better results than some state-of-the-art algorithms in terms of both objective and human visual evaluations.

Keywords: Super-resolution, Sparse representation, Online dictionary learning, Non-local similarity

1 Introduction

Super-resolution (SR) is the method that uses one or several low-resolution (LR) images to reconstruct a high-resolution (HR) image. Denote the HR image as \mathbf{X} , and the LR image as \mathbf{Y} , then the degradation of \mathbf{X} to form an LR image can be generally formulated by

$$\mathbf{Y} = \mathbf{SHX} + \mathbf{v} \quad (1)$$

where H represents the blurring process and S represents the down-sampling process. \mathbf{v} is the additive noise. Super-resolution solves the inverse problem of the degradation while it remains extremely ill-posed, which means there are generally multiple solutions that can be degenerated to the same LR image.

The SR algorithms can be broadly classified into two classes: (i) the traditional method which focuses on the reconstruction with several LR images such as a short video; (ii) the example-based method which deals with a

single input LR image, which was called image hallucination by some articles [1–4]. In the traditional method, each LR image imposes a set of linear constraints to stabilize the solution of the unknown HR image. However, because the LR images lack high-frequency part, which was more sensitive to the human eyes, the traditional method cannot achieve a relatively high magnification factor [1, 5]. In example-based method, the single image super-resolution remains much ill-posed because it has only one constraint. To cope with the ill-posed nature of image super-resolution, prior knowledge of natural images is usually employed for regularizing the solution to the following minimization problem:

$$\mathbf{X} = \underset{\mathbf{X}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{SHX}\|_2^2 + \lambda J(\mathbf{X}) \quad (2)$$

where $J(\mathbf{X})$ is a regularization term specifying the prior knowledge of the HR image and λ is a scalar balancing between the quadratic fidelity term and the regularization term, such as the total variation (TV) regularization [6], edge smoothness [7], and gradient profile priors [8]. However, these methods cannot recover fine details and have unnatural edges.

*Correspondence: wangdw@whu.edu.cn

²International School of Software, Wuhan University, Wuhan 430072, People's Republic of China

Full list of author information is available at the end of the article

In the past several years, sparsity has been emerging as one of the most significant properties of natural images [9]. The sparsity prior suggests that image patch can be well-represented as a sparse linear combination of elements from an appropriately chosen over-complete dictionary [10]. The sparsity-based regularization has achieved great success both qualitatively and quantitatively. However, it still has a little jaggy and ringing artifact along the edges in the reconstructed image. One of the keys to improve the result is to find a more suitable dictionary. Different improvements were proposed [11–14], etc., and have gotten better results.

Another significant property exhibited in natural images is nonlocal self-similarity, which is based on an observation that patches in a single natural image tend to redundantly recur many times inside the image, both within the same scale, as well as across different scales [15]. In recent works, the sparsity and the self-similarity of natural images are usually combined to achieve better performance [13, 16, 17].

Traditional algorithms which are mentioned above often use patch as the basic unit of sparse representation and train redundant dictionaries with fixed sample image sets. Zhang et al. [18] and Zhang et al. [19] exploit the concept of group-based sparse representation for general image inverse problem and develop an efficient and effective algorithm for image restoration and image compressive sensing recovery. Inspired by these works, this paper uses group as the basic unit for image super-resolution. The main contribution of our proposed method is that we divide the input image into several groups to combine the sparsity and the self-similarity of natural images in a unified framework and improve the performance of the dictionary for each group with the novel online dictionary learning method, which is more suitable than the one trained with classic algorithms. Experiments show that the proposed algorithm outperforms many current state-of-the-art schemes.

The rest of the paper is organized as follows. Section 2 introduces the related works. Section 3 presents the proposed super-resolution method and gives its implementation details. Section 4 shows various comparison experiments. Section 5 gives the conclusions and discussions.

2 Background and preliminaries

2.1 Super-resolution via sparse representation

In this section, we review the work on the single image super-resolution via sparse representation which was first introduced by Yang et al. [20]. The basic unit of sparse representation for natural image is patch. Let $\mathbf{x} \in \mathbb{R}^n$ denote the HR image patches of size $\sqrt{n} \times \sqrt{n}$, and $\mathbf{y} \in \mathbb{R}^m$ denote the features of LR image patches. Use $\mathbf{D}_h \in \mathbb{R}^{n \times K}$ and $\mathbf{D}_l \in \mathbb{R}^{m \times K}$ to denote the over-complete dictionaries of K

atoms ($K > n, K > m$), which are trained from HR and the feature of LR patches from training images, respectively. And the patches \mathbf{x} can be represented as a sparse linear combination with respect to \mathbf{D}_h . Which means \mathbf{x} can be expressed as

$$\mathbf{x} = \mathbf{D}_h \boldsymbol{\alpha} \quad (3)$$

where $\boldsymbol{\alpha}$ is the sparse coefficient and $\|\boldsymbol{\alpha}\|_0 \ll K$. The l_0 -norm counts the number of nonzero coefficients in vector $\boldsymbol{\alpha}$. The sparse representation $\boldsymbol{\alpha}$ can be estimated from its observation \mathbf{y} by solving the following l_0 -minimization problem below:

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\mathbf{F}\mathbf{y} - \mathbf{D}_l \boldsymbol{\alpha}\|_2^2 + \gamma \|\boldsymbol{\alpha}\|_0 \quad (4)$$

where parameter γ balances the fidelity term and the sparsity of the solution and F is a feature extraction operator.

The l_0 -minimization is an NP-hard problem. Donoho [21] shows that it can be approximated by l_1 -minimization when $\boldsymbol{\alpha}$ is sufficiently sparse. It can be expressed as follows:

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\mathbf{F}\mathbf{y} - \mathbf{D}_l \boldsymbol{\alpha}\|_2^2 + \gamma \|\boldsymbol{\alpha}\|_1 \quad (5)$$

which is known in statistical literature as the lasso.

Let $\mathbf{R}_k(\cdot)$ denote the operator that extracts the patch \mathbf{x}_k from the image \mathbf{X} at the k th position, and its transpose, denoted by $\mathbf{R}_k^T(\cdot)$, is the operator that puts back a patch into the k th position in the reconstructed image. The whole image \mathbf{X} can be reconstructed by averaging all of the reconstructed patches \mathbf{x}_k , which can be written as [18]

$$\mathbf{X} = \left(\sum_{k=1}^N \mathbf{R}_k^T \mathbf{1}_n \right)^{-1} \sum_{k=1}^N \mathbf{R}_k^T \mathbf{x}_k \quad (6)$$

where $\mathbf{1}_n$ is a vector of size n with all its elements being 1 and N is the total amount of the patches.

The image patches can be overlapped to better suppress noise and block artifacts. Considering Eqs. (3) and (6), we define the following operator “ \circ ” for convenience:

$$\mathbf{X} = \mathbf{D} \circ \mathbf{A} \stackrel{\text{def}}{=} \left(\sum_k \mathbf{R}_k^T \mathbf{1}_n \right)^{-1} \sum_{k=1}^N \mathbf{R}_k^T \mathbf{D} \boldsymbol{\alpha}_k \quad (7)$$

where \mathbf{A} denotes the concatenation of all $\boldsymbol{\alpha}_k$, i.e., $\mathbf{A} = [\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \dots, \boldsymbol{\alpha}_N]$.

Considering Eq. (5), the super-resolution via sparse representation can be formulated as follows:

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{F}\mathbf{Y} - \mathbf{D}_l \circ \mathbf{A}\|_2^2 + \gamma \sum_{k=1}^N \|\boldsymbol{\alpha}_k\|_1 \quad (8)$$

where γ is the regularization parameter. With $\hat{\mathbf{A}}$, the reconstructed image can be expressed by $\hat{\mathbf{X}} = \mathbf{D}_h \circ \hat{\mathbf{A}}$.

While $\hat{\mathbf{X}}$ may not satisfy the reconstruction constraint (2) exactly, it should be projected onto the solution space of (1), computing

$$\mathbf{X} = \underset{\mathbf{X}}{\operatorname{argmin}} \|\mathbf{SHX} - \mathbf{Y}\|_2^2 + \lambda \|\mathbf{X} - \hat{\mathbf{X}}\|_2^2 \quad (9)$$

where λ balances between the fidelity term and the regularization term.

2.2 Dictionary learning

The dictionary is usually learned from a set of training examples $X = \{x_1, x_2, \dots, x_t\}$, and it can be trained from the following formulation:

$$\begin{aligned} (\mathbf{D}, \mathbf{Z}) = \underset{\mathbf{D}, \mathbf{Z}}{\operatorname{argmin}} & \|\mathbf{X} - \mathbf{DZ}\|_2^2 + \lambda \|\mathbf{Z}\|_1 \\ \text{s.t.} & \|\mathbf{D}_i\|_2^2 \leq 1, i = 1, 2, \dots, K \end{aligned} \quad (10)$$

where \mathbf{Z} is the set of sparse representations of the training set \mathbf{X} and the l_1 -norm $\|\mathbf{Z}\|_1$ is used for enforcing sparsity. Eq. (10) is not convex in both \mathbf{D} and \mathbf{Z} but is convex in one of them when the other is fixed. So, it can be solved in an alternative manner over \mathbf{Z} and \mathbf{D} .

In the previous subsection, we saw the successful image SR should guarantee the situation that each pair of HR and LR image patch has the same sparse representation with respect to the two dictionaries \mathbf{D}_h and \mathbf{D}_l , respectively. Given a set of training sample pairs $P = \{X_h, Y_l\}$, where $X_h = \{x_1, x_2, \dots, x_n\}$ are the set of sampled HR image patches and $Y_l = \{y_1, y_2, \dots, y_n\}$ are the corresponding LR image patches. When the two dictionary learning processes combine and force the HR and LR representations to share the same code, they can be written as

$$\begin{aligned} \min_{\mathbf{D}_h, \mathbf{D}_l, \mathbf{Z}} & \frac{1}{N} \|\mathbf{X}_h - \mathbf{D}_h \mathbf{Z}\|_2^2 + \frac{1}{M} \|\mathbf{FY}_l - \mathbf{D}_l \mathbf{Z}\|_2^2 \\ & + \lambda \left(\frac{1}{N} + \frac{1}{M} \right) \|\mathbf{Z}\|_1 \end{aligned} \quad (11)$$

where N and M are the dimensions of the HR and LR image patches in vector form. (11) can be rewritten as follows:

$$\min_{\mathbf{D}_h, \mathbf{D}_l, \mathbf{Z}} \|\mathbf{X}_c - \mathbf{D}_c \mathbf{Z}\|_2^2 + \hat{\lambda} \|\mathbf{Z}\|_1 \quad (12)$$

where

$$\mathbf{X}_c = \begin{bmatrix} \frac{1}{\sqrt{N}} \mathbf{X}_h \\ \frac{1}{\sqrt{M}} \mathbf{FY}_l \end{bmatrix}, \quad \mathbf{D}_c = \begin{bmatrix} \frac{1}{\sqrt{N}} \mathbf{D}_h \\ \frac{1}{\sqrt{M}} \mathbf{D}_l \end{bmatrix} \quad (13)$$

Thus, the strategy of single dictionary learning can be used for training the two dictionaries for SR purpose.

3 The proposed algorithm

The non-local similarity prior for natural images is based on an observation that patches in a single natural image tend to redundantly recur many times inside the image. On the other hand, natural images are believed to be

composed of simple local image structures, observed as singular primitives, such as lines and arcs [3]. These local singular primitives are invariant to scale changes. So, a patch has good matches around its original location in the lower scale image [22].

These researches show that a natural image can be divided into several groups. The patches in the same group have similar image structures and can be presented by a relatively compact dictionary, which is more suitable for the patches in the group than a redundant dictionary. We use group as the basic unit instead of patch to gain a better result at the same time.

We treat the input LR image as the image that contains some high-frequency contents but with unsatisfactory pixel resolution. So, it offers high-frequency information about the singular primitives and can be used in our group-based SR algorithm. Let $\mathbf{X}_l \in \mathbb{R}^K$ denote the input LR image, where K is the size of the whole image vector. We down-sample \mathbf{X}_l and then up-sample it using bi-cubic interpolation by the same factor of s to obtain the low-frequency band image $\mathbf{Y}_l \in \mathbb{R}^K$. Then, we up-sample \mathbf{X}_l with bi-cubic interpolation by the factor of s to obtain low-frequency band $\mathbf{Y}_h \in \mathbb{R}^{s^2 K}$ of the unknown HR image $\mathbf{X}_h \in \mathbb{R}^{s^2 K}$. Use $\mathbf{y}_l^k, \mathbf{y}_h^k, \mathbf{x}_l^k$, and \mathbf{x}_h^k to denote the vector representations of the image patch extracted from $\mathbf{Y}_l, \mathbf{Y}_h, \mathbf{X}_l$, and \mathbf{X}_h in the k th position, respectively.

Incorporating with the nonlocal similarity prior knowledge, for a patch \mathbf{y}_h^k , we search the similar patches around the corresponding place and make a group, which is denoted by $G_{y_h}^j$, where j is the group order, and the total number of the group is denoted by L . Euclidean distance is selected as the similarity criterion between different patches. The number of the patches in the group is c , that is to say we choose $c - 1$ patches that are most like the patch \mathbf{y}_h^k to make a group, and then delete them from the patch list. So $L = \lceil \frac{P}{c} \rceil$, where P is the total number of the patches and $\lceil \cdot \rceil$ is the ceiling function. The corresponding patches of $\mathbf{y}_h^k \in G_{y_h}^j (i = 1, 2, \dots, c)$ in image \mathbf{Y}_l can also make a group, which is denoted by $G_{y_l}^j$. The group $G_{y_l}^j$ and its high-frequency version $G_{x_l}^j$ provide the information about the lost high-frequency band to the unknown image \mathbf{X}_h .

Figure 1 shows the framework of our proposed algorithm. We divide our strategy into two phases: the first is the online dictionary learning phase (marked as red lines) and the second is the simultaneous sparse coding phase (marked as blue lines). We show the details of the first phase in Section 3.1 and the second phase in Section 3.2.

3.1 Online dictionary learning phase

In this paper, online dictionary learning is imported to train suitable dictionary for the group. Online dictionary learning was first presented by [23], which can handle

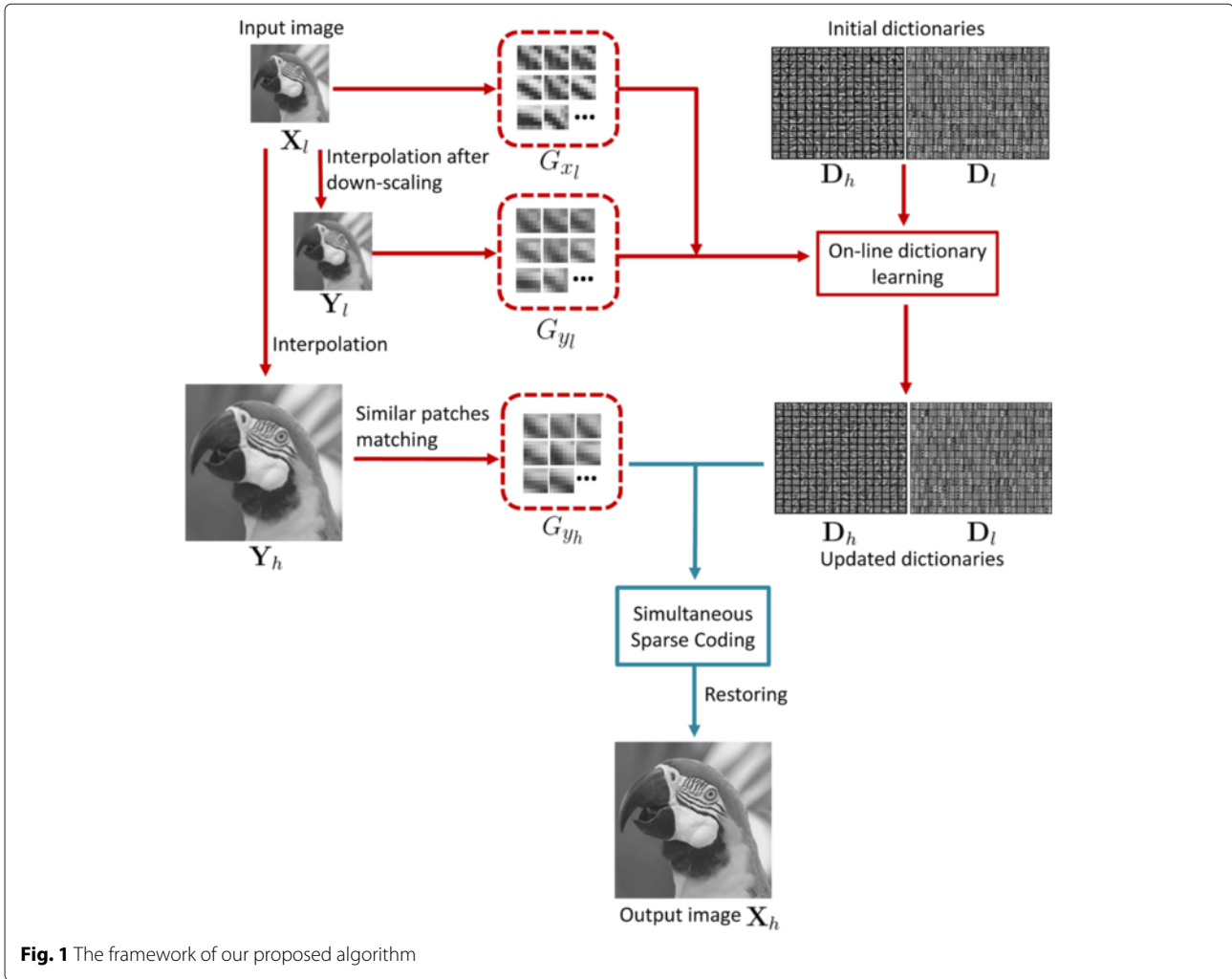


Fig. 1 The framework of our proposed algorithm

potentially infinite data sets, adapt to dynamic training sets, and it is dramatically faster than traditional algorithms. Instead of using a fixed dictionary, we try to update dictionary during the image processing of each patch. The patches in group $G_{x_l}^j$ can be treated as the corresponding patches in group $G_{y_l}^j$ with high frequency. And the group $G_{y_l}^j$ and the group $G_{x_l}^j$ compose the training set.

To ensure that the HR dictionary D_h and the LR dictionary D_l have the same sparse representation, we use the method mentioned in Section 2.2 and transform the joint dictionary learning into a single one.

We subtract patches in $G_{y_l}^j$ from corresponding patches in $G_{x_l}^j$ to obtain the high-frequency part as the HR sample group set, denoted by G_h^j . We extract the feature of patches in G_h^j with the feature extraction operator F in order to boost the prediction accuracy. In [2], high-pass filter was used as F to extract the edge information as the feature. In [24], a set of Gaussian derivative filters was used to extract the contours in the LR patches. In [25], the

wavelets of LR images were used to train dictionary. And in [10], the first- and second-order derivatives were used as the features:

$$\begin{aligned} f_1 &= [-1, 0, 1], & f_2 &= f_1^T \\ f_3 &= [1, 0, -2, 0, 1], & f_4 &= f_3^T \end{aligned} \quad (14)$$

where the superscript “ T ” means transpose. In this paper, we use Eq. (14) as F due to its simplicity and effectiveness.

Instead of applying the four filters to the patches in group $G_{y_l}^j$ directly, we apply these filters to Y_l to get the four gradient maps and extract the four feature patches from the corresponding location. Then, we concatenate the vector representations of the four patches in the same location and form the LR sample vector group set G_l^j for dictionary training. Thus, we get the training group set G_c^j for dictionary updating by rearranging the sample using the equation below:

$$G_c^j = \begin{bmatrix} \frac{1}{\sqrt{N}} G_h^j \\ \frac{1}{\sqrt{M}} G_l^j \end{bmatrix} \quad (15)$$

where N and M are the dimensions of the HR and LR image patches in vector form.

The formulation for the updating of the dictionary is shown below:

$$\mathbf{D}_i^j = \underset{\mathbf{D}}{\operatorname{argmin}} \frac{1}{c} \sum_{i=1}^c l(\mathbf{x}_i^j, \mathbf{D}_{i-1}^j) \quad (16)$$

where

$$l(\mathbf{x}, \mathbf{D}) \stackrel{\text{def}}{=} \min_{\alpha} \frac{1}{2} \|\mathbf{x} - \mathbf{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (17)$$

$$\mathbf{D}^j = \begin{bmatrix} \frac{1}{\sqrt{N}} \mathbf{D}_h^j \\ \frac{1}{\sqrt{M}} \mathbf{D}_l^j \end{bmatrix} \quad (18)$$

where \mathbf{x}_i^j is the sample vector with index i in the group set G_c^j and \mathbf{D}_0 is an initial dictionary which is learned using an external database. Thus, we have finished learning dictionary \mathbf{D}^j for the patches in the group with the index j .

The basic unit of dictionary learning is still patch, but each dictionary is only updated by the patches in the corresponding groups. Different from the traditional dictionary learning method proposed by Yang et al. [10] that uses a single dictionary for the construction of all patches, this method also contained the information of all patches in the group in dictionary learning phase, which made the dictionary more suitable for the patches in the corresponding group. Because the patches in the group are similar, the dictionary is relatively more compact than the initial one.

We randomly extracted 80,000 patches from 200 high-quality natural images from the *Berkeley Segmentation Database* [26]. With the same feature extractor F mentioned above and the method introduced in Section 2.2, we can calculate \mathbf{D}_0 .

Figure 2 shows the HR dictionary \mathbf{D}_h and the LR dictionary \mathbf{D}_l learned by our algorithm. The first row is the primal dictionaries we used as the initial \mathbf{D}_0 . The second row is the updated dictionaries during the process of the first group. It is obvious that some elements in the updated dictionaries are set to zero vectors, which reduce the size of the dictionaries. And the elements in the updated dictionaries are more specific than the initial ones.

3.2 Simultaneous sparse coding phase

The sparse coding phase attempts to magnify all of the patches in the input LR image \mathbf{X}_l . After estimating the low-frequency part \mathbf{Y}_h of the final HR image \mathbf{X}_h , we just need to restore the high-frequency part and then add it to \mathbf{Y}_h .

We apply simultaneous sparse coding to the sparse representation of each group. For traditional sparse coding,

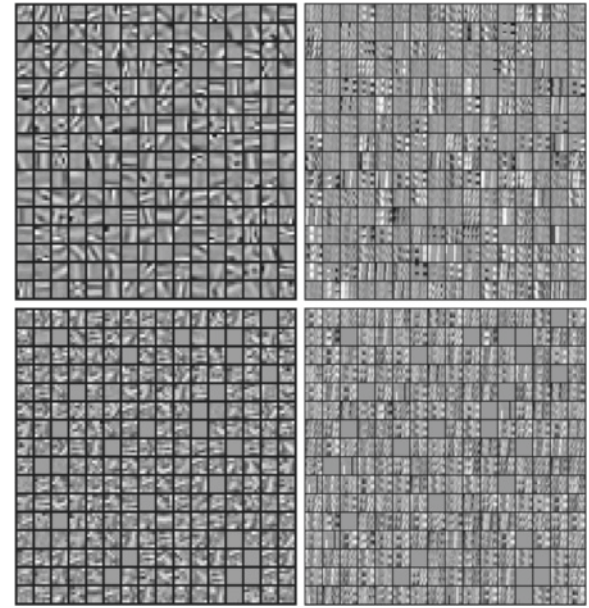


Fig. 2 Dictionaries used during the image process

similar patches in one group sometimes admit very different estimates due to the potential instability of sparse decomposition, which can result in noticeable reconstruction artifacts [27]. A simultaneous sparse coding algorithm makes approximation of several input signals at the same time using different linear combinations of the same elementary signals [28]. It solves the problem of the traditional sparse coding by forcing similar patches to admit similar decomposition.

The joint sparse representation \mathbf{A}^j of the group $G_{y_h}^j$ can be formulated as below:

$$\mathbf{A}^j = \underset{\mathbf{A}^j}{\operatorname{argmin}} \sum_{i=1}^c \|F\mathbf{y}_h^i - \mathbf{D}_l^j \alpha_i^j\|_2^2 + \gamma \|\mathbf{A}^j\|_{p,q} \quad (19)$$

where \mathbf{y}_h^i is the patch vector in the group $G_{y_h}^j$ with index i and α_i^j is the sparse representation of \mathbf{y}_h^i with dictionary \mathbf{D}_l^j ; where $\mathbf{A}^j = [\alpha_i^j]$. And $l_{p,q}$ is matrix norm defined as [27]

$$\|\mathbf{A}\|_{p,q} \stackrel{\text{def}}{=} \sum_{i=1}^M \|\alpha_i\|_q^p \quad (20)$$

where α_i is the i th row of \mathbf{A} . In practice, the value of the pair (p, q) is usually chosen as $(1, 2)$ or $(0, \infty)$, the former leading to a convex norm, while the latter actually counts the number of nonzero rows.

This optimization can be solved by simultaneous orthogonal matching pursuit (S-OMP) algorithm [28].

After obtaining \mathbf{A}^j , the super-resolved patch \mathbf{x}_h^i in the group $G_{x_h}^j$ can be written as

$$\mathbf{x}_h^i = \mathbf{D}_h^j \alpha_i^j + \mathbf{y}_h^i \quad (21)$$

With the extract and restore operator defined in (7), the whole image \mathbf{X}_h can be reconstructed by

$$\mathbf{X}_h = \sum_{j=1}^L \mathbf{D}_h^j \circ \mathbf{A}^j + \mathbf{Y}_h \quad (22)$$

Our method skips the back-project step mentioned in (9), because the sparsity prior is strong enough that we can already achieve good performance.

Algorithm 1 shows the complete process of our proposed method.

Algorithm 1 The proposed algorithm

Input: Low-resolution image \mathbf{X}_l , factor s , initial dictionary \mathbf{D}_0 , regularization parameter λ and γ , number of iterations T .

1. **Initialization:**

- (a) $\mathbf{D} \leftarrow \mathbf{D}_0$.
- (b) Scale up \mathbf{X}_l with factor s using bi-cubic interpolation to generate \mathbf{Y}_h , down-sample \mathbf{X}_l and then interpolate it with the same factor s to generate \mathbf{Y}_l . Extract the feature of \mathbf{Y}_l using (14).
- (c) Divide \mathbf{Y}_h into P patches and then collect them into L groups according to the Euclidean distance.

2. **for** $j = 1, \dots, L$ **do**

- (a) Collect the corresponding patches in \mathbf{X}_l and \mathbf{Y}_l of the patches in the j -th group of \mathbf{Y}_h , and generate the training group set G_c^j using (15).
- (b) **for** $i = 1, \dots, c$ **do**
Update the dictionary \mathbf{D} with (17), and then get the high-resolution dictionary \mathbf{D}_h and low-resolution dictionary \mathbf{D}_l using (18).
end for
- (c) Compute sparse representation \mathbf{A}^j of patches in the j -th group of \mathbf{Y}_h in (19) with \mathbf{D}_l using S-OMP algorithm.
- (d) Reconstruct the patches in the j -th group of \mathbf{X}_h with \mathbf{A}^j and \mathbf{D}_h using (21).

end for

- 3. Put the patches in all groups back together to generate the output image \mathbf{X}_h .

Output: High-resolution image \mathbf{X}_h .

4 Experimental results

Figure 3 shows the 15 images that are used for comparison. In our experiment, all the test images are applied a 7×7 Gaussian kernel of zero mean and standard deviation 1.0 and then down-sampled by a decimation factor of 2 to produce the corresponding LR images.

We compare the proposed method with other four algorithms to illustrate the efficiency of our proposed method. The compared algorithms are bi-cubic interpolation [13, 15, 20]. Specifically, for methods based on fixed external dictionaries, we choose the work of Yang et al. [20] for comparison; for methods based on non-local similarity, we choose the work of Glasner et al. [15] for comparison; for methods that combine sparse representation and non-local similarity, we choose the representative work ASDS method [13] for comparison.

A frequently used criterion, peak signal-to-noise ratio (PSNR), is used for the image quality analysis. But it is sometimes not a reliable metric for evaluating the image quality. Therefore, the structural similarity (SSIM) index [29] and the feature similarity (FSIM) index [30] are also adopted for the objective evaluation. A higher PSNR value implies less distortion compared with the ground truth, and an SSIM value or an FSIM value much closer to 1 indicates the structure or the feature of the reconstructed image is more similar to the ground truth image, respectively.

For color image super-resolution, we only apply our algorithm on the illuminance component and use bi-cubic interpolation for the chromatic components, because human visual system is not sensitive to the chromatic components. In experiments, the value of PSNR, SSIM, and FSIM are all conducted on the illuminance component of the image.

4.1 Experimental configuration

We magnify the input LR image by factor of 2 and use 6×6 patches with an overlap of two pixels between adjacent patches, both for the HR image \mathbf{Y}_h and \mathbf{X}_h and LR image \mathbf{Y}_l and \mathbf{X}_l ; and we learned the dictionary of size $K = 128$. In the online dictionary learning phase, the size L of training window is selected as 60, the number c of best matched patches is 128, the sparsity regularization parameter λ is 0.15, and the number T of iteration to train the dictionary is 32. In the simultaneous sparse coding phase, the parameter γ that balances the fidelity term and the regularization term is 0.15, and the value of (p, q) is chosen as 1, 2.

We use the images from the Berkeley Segmentation Dataset and Benchmark [26] to train dictionaries for our method and the method of Yang et al. [20]. Because there are large differences between our method and the methods being compared, we just use the default parameters as configurations of these methods.



Fig. 3 The input images we used in the comparison experiments

The proposed algorithm is implemented by MATLAB R2011b using *SPAMS toolbox* [31] for on-line dictionary learning and simultaneous sparse coding. The computer system used for simulation is Intel Core i7-4500U CPU at 1.80GHz with 8GB of RAM.

4.2 Noiseless experiment

4.2.1 Objective evaluations

For objective evaluations, Table 1 reports the results obtained by the proposed method and other super-resolution approaches. In this table, the proposed approach holds the best performance for most of the images. While for images such as cameraman or parrot, the proposed approach is not so good for the PSNR comparisons. That is because the PSNR value just counts the error between pixel values in the two images while ignores the characteristics of human visual system. The training samples of the proposed method need to be normalized before online dictionary training according to the requirement of the online training function. After the sparse approaching phase, the tonal range has a little bias in the histogram. This bias leads influences on the PSNR value.

The SSIM and FSIM value, which are more accurate than PSNR value to evaluate the image quality, simulate the characteristic of human visual system. The improvement of our results in SSIM and FSIM shows the structure and feature restoration are better than the other competing methods.

4.2.2 Visual quality evaluations

For visual quality evaluations, Figs. 4, 5, 6, and 7 show the visual comparison of different approaches. In the figures, the results of bi-cubic interpolation are very blurry and

have staircase artifacts on the ramp. The results of Yang et al. [20] restore lots of details and sharp edges, while introducing noise and ringing artifacts. This method uses a fixed dictionary pair to try to restore the image but it cannot be suitable for different kind of structures. The results of Glasner et al. [15] seem too smooth to retain enough details. This method uses non-local similarity prior and extracts information only from the input image itself. It makes full use of the input image, but the high-frequency information in the input image is limited. The results of ASDS [13] are better than bi-cubic for less staircase artifacts and less blurry edges. The ASDS [13] algorithm uses several sub-dictionaries to match different kinds of micro-structures, but the sub-dictionaries cannot guarantee the coverage of all the micro-structures.

Our algorithm uses the dictionary trained from external databases to offer the missing high frequency and the input image itself to offer the ground truth information. The online learning updates the dictionary for each patch to combine the information of external databases and the ground truth. The updated dictionary is more suitable for the patch and avoids adding some artifacts to the restored image.

It is obvious that the results of the proposed method have better details and edges. For long edges, there exist many similar patches along the edge which make the updated dictionary more precise, so the edges of the proposed results are much sharper than those of other algorithms. For the details, the input image offers important information about the details, especially the repeat similar details, to our super-resolution, while the fixed dictionaries cannot. However, the very short edges and the non-redundant details can hardly find enough similar patches.

Table 1 Comparisons of peak signal-to-noise ratio(PSNR) values, structural similarity (SSIM) values, and feature similarity (FSIM) values for 15 test images with different super-resolution approaches

Image		Bi-cubic	Yang et al. [20]	Glasner et al. [15]	ASDS [13]	Proposed
Avion	PSNR	27.01	29.13	28.48	<i>31.62</i>	30.92
	SSIM	0.875	0.905	0.918	0.934	<i>0.936</i>
	FSIM	0.863	0.902	0.907	0.930	<i>0.936</i>
Barnfall	PSNR	29.16	30.14	29.88	<i>30.76</i>	30.51
	SSIM	0.716	0.780	0.763	0.783	<i>0.788</i>
	FSIM	0.825	0.895	0.862	0.893	<i>0.899</i>
Blueeye	PSNR	33.39	34.00	35.28	36.37	<i>36.75</i>
	SSIM	0.923	0.923	0.939	0.939	<i>0.944</i>
	FSIM	0.946	0.957	0.960	0.962	<i>0.968</i>
Butterfly	PSNR	28.51	30.13	29.85	<i>32.22</i>	31.51
	SSIM	0.855	0.894	0.902	0.921	<i>0.926</i>
	FSIM	0.909	0.930	0.935	0.947	<i>0.950</i>
Cactusflower	PSNR	25.48	26.67	26.34	<i>27.47</i>	26.95
	SSIM	0.658	0.763	0.728	0.749	<i>0.764</i>
	FSIM	0.793	0.877	0.844	0.863	<i>0.885</i>
Cameraman	PSNR	25.55	26.83	26.68	<i>28.68</i>	27.94
	SSIM	0.827	0.862	0.869	0.892	<i>0.899</i>
	FSIM	0.821	0.883	0.864	<i>0.908</i>	0.901
Colomtn	PSNR	27.44	28.33	27.93	<i>28.80</i>	28.67
	SSIM	0.685	0.747	0.729	0.749	<i>0.757</i>
	FSIM	0.807	0.877	0.841	0.876	<i>0.886</i>
Desert	PSNR	24.10	24.86	24.89	<i>25.59</i>	25.41
	SSIM	0.677	0.757	0.737	0.780	<i>0.786</i>
	FSIM	0.812	0.879	0.852	0.890	<i>0.896</i>
Frog	PSNR	31.42	32.74	32.78	<i>34.45</i>	34.27
	SSIM	0.895	0.920	0.927	0.937	<i>0.944</i>
	FSIM	0.894	0.931	0.924	0.945	<i>0.953</i>
Goldgate	PSNR	33.05	34.46	34.05	<i>35.48</i>	35.10
	SSIM	0.873	0.899	0.907	0.908	<i>0.912</i>
	FSIM	0.877	0.919	0.909	0.922	<i>0.925</i>
House	PSNR	26.53	28.40	28.14	<i>30.33</i>	29.61
	SSIM	0.817	0.863	0.865	0.892	<i>0.894</i>
	FSIM	0.835	0.886	0.888	0.913	<i>0.919</i>
London	PSNR	29.27	31.32	30.34	<i>33.28</i>	33.13
	SSIM	0.850	0.894	0.894	0.915	<i>0.919</i>
	FSIM	0.837	0.903	0.877	<i>0.919</i>	0.918
Lostlake	PSNR	27.60	28.81	28.37	<i>29.71</i>	29.30
	SSIM	0.749	0.806	0.795	0.819	<i>0.824</i>
	FSIM	0.839	0.896	0.877	0.903	<i>0.907</i>
Parrot	PSNR	28.93	31.21	30.19	<i>34.05</i>	32.92
	SSIM	0.911	0.926	0.939	0.945	<i>0.949</i>
	FSIM	0.917	0.948	0.938	<i>0.959</i>	0.957
Redrock	PSNR	27.33	28.51	28.12	<i>29.35</i>	29.11
	SSIM	0.773	0.834	0.820	0.847	<i>0.851</i>
	FSIM	0.820	0.889	0.858	<i>0.894</i>	0.893

The data in italics indicates the best PSNR/SSIM/FSIM result



Fig. 4 Comparison of SR results of blueeye image with magnification factor 2. *Top row:* bi-cubic, the method in [20], and the method in [15]. *Bottom row:* the method in [13], the proposed method, and the ground truth image

With the fixed size of the search window, many patches that are not so similar are also added into the training sample set. This may mislead the dictionary updating, thus leads to a result that is not so fine. In addition, when the low-frequency part of the input image is too blurry, it also misleads the dictionary updating and produces blurry details.

In Fig. 4, the edges and the streaks of the petal of our method are clearer than that of Glasner et al. [15] and ASDS [13], while there is no additional noise and ringing artifact as which in the result of Yang et al. [20]. In Fig. 5, the fine lines in the wings of the butterfly are restored very well, and the round dot is not distorted when compared

with the first three methods. Because the textures in the butterfly wings are repeated, which is convenient for us to collect much useful ground truth information about these details. The same reason is for the restoration of Fig. 6. The short stripes on the parrot's face are separated in the results of bi-cubic, Yang et al. [20], ASDS [13], and our method, while the edges of other stripes in our result are sharper than that of bi-cubic and ASDS [13] and are more delicate than that of Yang et al. [20]. In Fig. 7, the details of the camera is restored better than the others, especially the white fine line in the left bottom corner of the camera. Besides, it is obvious that the edges of the cameraman's coat are sharper than others. These indicate that

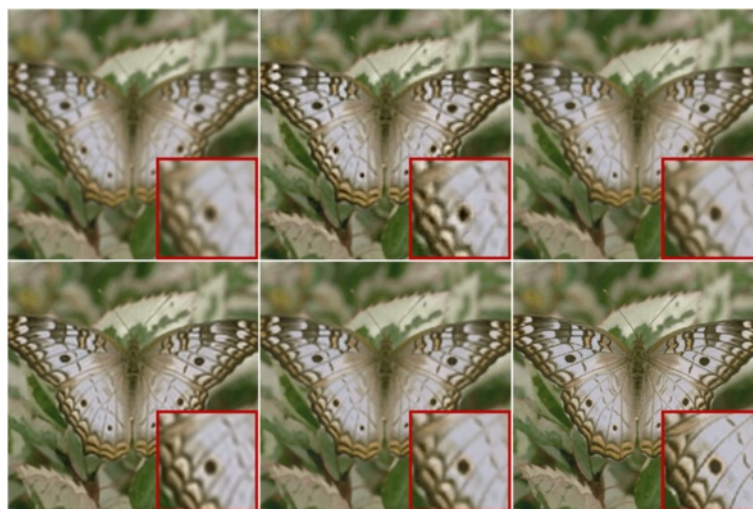


Fig. 5 Comparison of SR results of butterfly image with magnification factor 2. *Top row:* bi-cubic, the method in [20], and the method in [15]. *Bottom row:* the method in [13], the proposed method, and the ground truth image

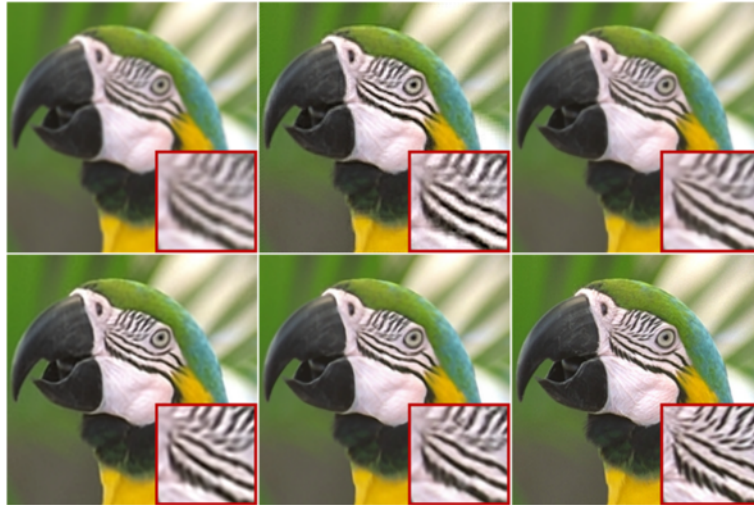


Fig. 6 Comparison of SR results of parrot image with magnification factor 2. *Top row*: bi-cubic, the method in [20], and the method in [15]. *Bottom row*: the method in [13], the proposed method, and the ground truth image

our algorithm can recover fine details and sharp edges at the same time.

4.3 Noisy experiment

In this subsection, we added Gaussian white noise with zero mean and standard deviation of 1, 3, and 5 to the LR image parrot and then compare the result of the five methods. The objective evaluation is reported in Table 2 and the results of the LR image with noise level $\sigma_v = 5$ with different approaches are shown in Fig. 8.

From the table and the figure, we can see that the results of Yang et al. [20] and Glasner et al. [15] enhanced the noise. The result of ASDS [13] performs the best on noise

suppressing, but this noise suppression also affects the image reconstruction and makes the result image more blurry than its result of noiseless image. The proposed method achieves a good result both on noise suppressing and image reconstruction. The proposed method is better than the other because the group-based dictionary learning phase gets a relatively compact and suitable dictionary for the patches in the group to be reconstructed. The sparse coding in each iteration in the online dictionary learning method helps suppress the noise in the training patches. Thus, the elements in the dictionary and the structures of the group have a correlation between them, while the elements are independent with the noise.



Fig. 7 Comparison of SR results of cameraman image with magnification factor 2. *Top row*: bi-cubic, the method in [20], and the method in [15]. *Bottom row*: the method in [13], the proposed method, and the ground truth image

Table 2 Comparisons of peak signal-to-noise ratio (PSNR) values, structural similarity (SSIM) values, and feature similarity (FSIM) values for image with different noise level with different super-resolution approaches

Noise level		Bi-cubic	Yang et al. [20]	Glasner et al. [15]	ASDS [13]	Proposed
$\sigma_v = 0$	PSNR	28.93	31.21	30.19	<i>34.05</i>	32.92
	SSIM	0.911	0.926	0.939	0.945	<i>0.949</i>
	FSIM	0.917	0.948	0.938	<i>0.959</i>	0.957
$\sigma_v = 1$	PSNR	28.92	31.13	30.17	31.34	32.92
	SSIM	0.910	0.918	0.935	0.908	<i>0.947</i>
	FSIM	0.917	0.945	0.937	0.934	<i>0.956</i>
$\sigma_v = 3$	PSNR	28.83	30.49	29.94	31.24	32.67
	SSIM	0.897	0.863	0.908	0.907	<i>0.934</i>
	FSIM	0.913	0.918	0.925	0.936	<i>0.952</i>
$\sigma_v = 5$	PSNR	28.66	29.41	29.46	30.96	32.18
	SSIM	0.875	0.779	0.860	0.896	<i>0.911</i>
	FSIM	0.904	0.872	0.901	0.934	<i>0.941</i>

The data in italics indicates the best PSNR/SSIM/FSIM result

Combined with the simultaneous sparse coding phase, the noise is suppressed and the structure and the details are retained.

4.4 Time comparisons

In this subsection, we compare the running time of the four methods. We have ignored the time of training dictionaries for Yang et al. [20] and ASDS [13] and the time of training initial dictionaries for the proposed method.

We count the mean times for all configuration of the 15 images experiments, and Table 3 shows the running time of the compared methods. From the table, we can see that the shortest running time is that of Yang et al. [20]. The running time of Glasner et al. [15], ASDS [13], and the proposed method is comparable. Although we use

online dictionary learning algorithm during the processing which costs a large amount of time, we use group as the basic unit to reduce the dictionary learning times, and the simultaneous sparse coding algorithm also accelerates the processing.

5 Conclusions

This paper describes a novel method of group-based single image super-resolution. With the property of non-local self-similarity, we divide the input image into several groups. Combining with the information from external databases, we train suitable dictionaries for each group using online dictionary learning method. Simultaneous sparse coding algorithm is used to accelerate the processing and improve the result. Experiments show that



Fig. 8 Comparison of SR results of parrot image with noise level $\sigma_v = 5$ with magnification factor 2. Top row: input image, bi-cubic, and the method in [20]. Bottom row: the method in [15], the method in [13], and the proposed method

Table 3 Time comparisons of different approaches

Method	Time(s)
Yang et al. [20]	15.1332
Glasner et al. [15]	427.5111
ASDS [13]	173.2815
The proposed	257.0103

the proposed method can restore sharp edges and fine details and achieve good result on noise suppressing. The running time is comparable with other state-of-the-art algorithms.

In this paper, we just use the traditional Euclidean distance for the searching of the similar patches to make a group. For further research, we will focus on developing a evaluation that directly measure the probability between two patches that belong to the same group to improve the performance of the proposed method.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant No. 61501334).

Availability of data and materials

The images supporting the conclusions of this article are available in the "Test Images of Computer Vision Group". All of the images are Copyright free. <http://decsai.ugr.es/cvg/index2.php>.

Authors' contributions

XL and DW designed and carried out the experiments. XL and DD analyzed the experimental results. XL wrote the manuscript. WS and DD gave the critical revision and final approval. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Author details

¹School of Electronics and Information, Wuhan University, Wuhan 430072, People's Republic of China. ²International School of Software, Wuhan University, Wuhan 430072, People's Republic of China. ³School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, People's Republic of China.

Received: 20 February 2016 Accepted: 14 July 2016

Published online: 29 July 2016

References

1. B Simon, K Takeo, Limits on super-resolution and how to break them. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(9), 1167–1183 (2002)
2. TF William, CP Egon, Learning low-level vision. *Int. J. Comput. Vis.* **40**(1), 25–47 (2000)
3. TF William, RJ Thouis, CP Egon, Example-based super-resolution. *IEEE Comput. Graph. Appl.* **22**(2), 56–65 (2002)
4. S Jian, Z Jieie, FT Marshall, in *IEEE Conference on Computer Vision and Pattern Recognition*. Context-constrained hallucination for image super-resolution (IEEE, San Francisco, CA, USA, 2010), pp. 231–238
5. L Zhouchen, S Heung-Yeung, Fundamental limits of reconstruction-base superresolution algorithms under local translation. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(1), 83–97 (2004)
6. M Antonio, JO Stanley, Image super-resolution by tv-regularization and Bregman iteration. *J. Sci. Comput.* **37**(3), 367–382 (2008)
7. D Shengyang, H Mei, X Wei, W Ying, G Yihong, KK Aggelos, Softcuts: a soft edge smoothness prior for color image super-resolution. *IEEE Trans. Image Process.* **18**(5), 969–981 (2009)
8. S Jian, S Jian, X Zongben, S Heung-Yeung, in *IEEE Conference on Computer Vision and Pattern Recognition*. Image super-resolution using gradient profile prior (IEEE, Anchorage, Alaska, USA, 2008), pp. 24–26
9. MB Alfred, LD David, E Michael, From sparse solutions of systems of equations to sparse modeling of signals and images. *Siam Rev.* **51**(1), 34–81 (2009)
10. Y Jianchao, W John, H Thomas, M Yi, Image super-resolution via sparse representation. *IEEE Trans. Image Process.* **19**(11), 2861–2873 (2010)
11. Y Jianchao, W Zhaowen, L Zhe, C Scott, H Thomas, Coupled dictionary training for image super-resolution. *IEEE Trans. Image Process.* **21**(8), 3467–3478 (2012)
12. Z Jian, Z Chen, X Ruiqin, M Siwei, Z Debin, in *IEEE International Symposium on Circuits and Systems*. Image super-resolution via dual-dictionary learning and sparse representation (IEEE, COEX, Seoul, Korea, 2012), pp. 1688–1691
13. D Weisheng, Z Lei, S Guangming, W Xiaolin, Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Trans. Image Process.* **20**(7), 1838–1857 (2011)
14. G Xinbo, Z Kaibing, T Dacheng, L Xuelong, Joint learning for single-image super-resolution via a coupled constraint. *IEEE Trans. Image Process.* **21**(2), 469–480 (2012)
15. G Daniel, B Shai, I Michal, in *Proceedings of the 12th International Conference on Computer Vision: 29 September–2 October 2009*. Super-resolution from a single image (IEEE, Kyoto, 2009), pp. 349–356
16. GN Aneesh, SN Madhu, R Jeny, Single image super resolution from compressive samples using two level sparsity based reconstruction. *Procedia Comput. Sci.* **46**, 1643–1652 (2015)
17. D Weisheng, S Guangming, Z Lei, W Xiaolin, Super-resolution with nonlocal regularized sparse representation. *Proc. SPIE Visual Commun. Image Process.* **7744**, 77440–17744010 (2010)
18. Z Jian, Z Debin, G Wen, Group-based sparse representation for image restoration. *IEEE Trans. Image Process.* **23**(8), 3336–3351 (2014)
19. Z Jian, Z Debin, J Feng, G Wen, in *Data Compression Conference (DCC), 2013*. Structural group sparse representation for image compressive sensing recovery (IEEE, Snowbird, UT, 2013), pp. 331–340
20. Y Jianchao, W John, H Thomas, M Yi, in *IEEE Conference on Computer Vision and Pattern Recognition*. Image super-resolution as sparse representation of raw image patches (IEEE, Anchorage, AK, USA, 2008), pp. 1–8
21. LD David, For most large underdetermined systems of equations, the minimal l_1 -norm near-solution approximates the sparsest near-solution. *Commun. Pure Appl. Math.* **59**(7), 907–934 (2006)
22. Y Jianchao, L Zhe, C Scott, in *IEEE Conference on Computer Vision and Pattern Recognition*. Fast image super-resolution based on in-place example regression (IEEE, Portland, OR, USA, 2013), pp. 1059–1066
23. M Julien, B Francis, P Jean, S Guillermo, Online learning for matrix factorization and sparse coding. *J. Mach. Learn. Res.* **11**, 19–60 (2010)
24. S Jian, Z Nan-Ning, T Hai, S Heung-Yeung, in *IEEE Conference on Computer Vision and Pattern Recognition*. Image hallucination with primal sketch priors, (Madison, Wisconsin, USA, 2003), pp. 729–736
25. A Na, P Jinye, Z Xuan, F Xiaoyi, Ssr via trained double sparsity dictionaries. *Multimedia Tools Appl.* **74**(6), 1997–2007 (2013)
26. The Berkeley Segmentation Dataset and Benchmark. <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench>. Accessed June 2007
27. M Julien, B Francis, P Jean, S Guillermo, Z Andrew, in *IEEE International Conference on Computer Vision*. Non-local sparse models for image restoration (IEEE, Kyoto, Japan, 2009), pp. 2272–2279
28. AT Joel, CG Anna, JS Martin, Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal Process.* **86**(3), 572–588 (2006)
29. W Zhou, CB Alan, RS Hamid, PS Eero, Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **13**(4), 600–612 (2004)
30. Z Lin, Z Lei, M Xuanqin, Fsim: a feature similarity index for image quality assessment. *IEEE Trans. Image Process.* **20**(8), 2378–2386 (2011)
31. SParse Modeling Software. <http://spams-devel.gforge.inria.fr/downloads.html>. Accessed July 2014