



Super-resolution via supervised classification and independent dictionary training

Ronggui Wang¹ · Qinghui Wang¹ · Juan Yang¹ ·
Lixia Xue¹ · Min Hu¹

Received: 20 August 2017 / Revised: 12 February 2018 / Accepted: 27 March 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract Super-resolution (SR) reconstruction plays an important role in recovering the image details and improving the visual perception. In this paper, we propose a new and effective method based on the idea of classification reconstruction and independent dictionary training. Firstly, we extract some geometric features of images and design a new supervised classification method, which uses the decision tree to guarantee a better classification result. Secondly, the coefficients of the high-resolution (HR) and low-resolution (LR) patches are not equal strictly in fact, which enlighten us to train the HR and LR dictionaries independently. And then mapping matrices are learned to map LR coefficients into HR coefficients, which can not only help us improve reconstruction quality, but also just perform sparse coding one time in the reconstruction stage. At last, we enforce a global optimization on the initial reconstruction HR image based on the non-local means and the auto-regressive model. The experiments show that the method we proposed works better than other classic state-of-the-art methods.

Keywords Independent dictionary training · Decision tree · Mapping function · Super-resolution · Sparse coding

✉ Juan Yang
yangjuan6985@163.com

Ronggui Wang
wangrgui@foxmail.com

Qinghui Wang
1649759610@qq.com

Lixia Xue
xlzzm@163.com

Min Hu
jsjhumint@hfut.edu.cn

¹ College of Computer and Information, Hefei University of Technology, Hefei 230009, China

1 Introduction

Super-resolution reconstruction is an effective signal recovery technique which aims at constructing a HR image from one or multiple LR images. The HR image contains more details than the LR one, so the HR images are desired in many practical cases, such as medical diagnostic [31, 43], remote sensing [12, 33], intelligent surveillance [16, 18, 30]. The single-image SR reconstruction is mainly divided into three categories: interpolation-based methods, reconstruction-based methods, and learning-based methods.

The interpolation-based methods [2, 19, 27, 47] usually approximate the pixel value according to its neighborhood pixels. It is simple and fast, but the reconstructed images tend to be overly smooth. The reconstruction-based methods [5, 23, 26, 48] usually apply a certain prior knowledge to restrict the desired HR image and moreover maintain the consistency between the HR image and the LR input image. Although it has a clear effect on edge preserving and artificial influence suppression, the high-frequency details cannot be reconstructed effectively. Especially, the bigger the magnification of LR images, the worse the SR performance tends to be.

Compared with the two methods above, the learning-based methods can generate high-frequency details which are not available in the LR images, which has attracted many researchers in recent years. Freeman et al. [14] first proposed an example-based learning method, where the spatial relationship between patches is modeled by using a Markov network, and each pixel in the target HR image comes from only one nearest neighbor in the training set. Inspired by the idea of locally linear embedding proposed in [24], Chang et al. [4] proposed neighborhood embedding (NE) based on the principle that the HR and LR image have a similar local manifold. Specifically speaking, given a certain patch to be reconstructed, they obtained the K nearest neighbors and corresponding weight coefficients by searching from the LR image set, and then recover HR images with the linear combination of weights and neighbors. However, there are some drawbacks in the NE algorithm, one is that the reconstructed HR image tends to be under-fitting or over-fitting due to the fixed K , the other one is that the method is time-consuming and memory-intensive when searching K nearest patches. Recently, it is extended and developed as a sparse coding scheme [22, 37–39]. Yang et al. [37], enlightened from the idea of NE, first proposed a remarkable method which used the linear combination of different atoms in dictionary to represent various patches. In [38], the author proposed a classic method about dictionary training called joint dictionary training method which train HR and LR dictionaries together with a single dictionary training model based on the principle of invariance of HR and LR coefficients.

Without doubt, a critical stage in the learning-based methods is dictionary training, and it is studied widely by a lot of researchers [17, 28, 29, 34, 36, 41, 46] in order to further improve the training effect. Yang et al. [41] proposed a coupled dictionary training method based on patchwise sparse recovery, which trained and optimized dictionaries in turn. Xu et al. [36] preformed sparse coding on the HR and LR training set in turn via corresponding dictionaries, and trained HR and LR dictionaries iteratively. Zeyde et al. [46] and Timofte et al. [28, 29] trained LR dictionaries with KSVD [1, 25] method, and then applied the least squares to compute HR dictionaries based on the invariance above.

In summary, the training methods above are mostly based on the principle of invariance of HR and LR coefficients, however the dictionaries trained by these methods tend to have a certain degree of limitation on the expression ability of dictionary for patch features. In fact,

the two types of coefficients may not be strictly equal, since the HR image prone to have more details than the LR one. Therefore, Yuan et al. [45] trained dictionaries independently with principal component analysis (PCA), and estimated the corresponding HR coefficient with kernel support vector regression (KSVR). Yang et al. [44] applied ridge regression to train dictionaries independently, and utilized a pre-computed mapping matrix to reconstruct images. In this paper, we use KSVD to train dictionaries based on l_0 -norm independently so as to ensure the sparsity of coefficients.

On the other hand, the classic dictionary training tends to train HR and LR dictionaries directly with the entire training set and the atoms are arranged indifferently in the dictionaries, but it maybe not good enough. We can classify image patches which have similar geometric structure into the same category via the geometric features within patches. Then we use each sub-set to train the specialized dictionary which will have a more powerful and sufficient expression ability for corresponding geometric structure, so such dictionary will have a better work on the corresponding patches.

Clustering [8, 40, 42, 49, 50] is a rational strategy to accomplish this task, however the typical clustering algorithms (such as K-means, fuzzy C-means) are unsupervised classification methods that predicts fixed number of clusters, and the initial partitions may result in widely different final clusters. Additionally, it is difficult to evaluate the quality of the clusters produced. Therefore, it is necessary to introduce a supervised and deterministic clustering method into the classification of patches. On the other hand, the geometric information such as edges, contours, and textures in images, can remarkably influence the image perception quality, which can be used as a supervised prior to improve the performance of unsupervised methods. In this paper, we extract geometric features from images and apply them to our classification algorithm based on decision tree.

In recent years, some researchers apply the idea of deep learning on super resolution and get a good reconstruction result. For example, Dong et al. [7] designed several layers of network, and treated the hidden layers as a mapping which maps the input LR image to HR one. And his second version [10] improved on the reconstruction result due to the amendment on training set and other aspects. Liu et al. [20] designed a deep encoder-decoder symmetrical neural network for keeping the more image details. Wang et al. [35] combined the domain expertise of sparse coding and the merits of deep learning to achieve a good SR performance, which indicates that the domain expertise of sparse coding is useful in the area deep networks. In this paper, we mainly focus on the method based on sparse coding and make some improvements for enhancing the reconstruction image.

There are two stages in our reconstruction system, including the training stage and the reconstruction stage. In the training stage, we prepare the training set (generating the LR images, segmenting the images) for training independent dictionary. In other words, we extract some features, including variance, gradient feature and the angle of the gradient field, from these image patches for producing different categories, such as smooth patch, dominant orientation patch and stochastic patch. The dominant orientation patches can be further classified by the angle of the gradient field. Each patch set contains the HR and LR patches which can be trained independently corresponding HR and LR dictionaries based on K-SVD.

In the reconstruction stage, we interpolate the input image to get the LR image. Similarly, we segment the LR image and extract features to decide which category it belongs to for each patch. And then we choose the corresponding HR and LR dictionaries to reconstruct it. Next,

all patches reconstructed will be merged into a whole high frequency image which is added to the interpolation image to obtain the y-channel image. At last, we will enforce a global optimization, including non-local means and auto-regressive model, on this y-channel image. Figure 1 shows the reconstruction procedure of the proposed method, and the specific work of preparing data will be described in Section 2.4.

The main contributions of this paper are as follows:

1. We design a supervised classification algorithm based on image features, as we can see from our experiments, it works better than K-means under the same conditions.
2. We propose a new dictionary training method based on KSVD, which will train the HR and LR dictionaries independently for a further closer to fact of the difference of the HR and LR coefficients for the same patch.
3. We enforce a global optimization on the initially reconstructed image, which uses the non-local self-similarity and auto-regressive model, and get a not bad result. It indicates that exploiting the characteristic of image itself is very important, especially in some recent works over deep learning, we can get a lot of inspirations from it to further enhance the result.

The rest of the paper is organized as follows: Section 2 describes some works related to this paper. Section 3 details the classification of training set and the independent dictionary training. Section 4 describes the specific reconstruction procedure. Experimental results and discussions are showed in Section 5. And Section 6 concludes this paper.

2 Related works

In this Section, we first describe the constraint model of SR reconstruction. Then, we detail the sparse coding which will be applied to the dictionary training and image reconstruction stage. Next, orthogonal matching pursuit (OMP) is introduced for solving sparse coding. Finally, we introduce the preprocessing operations of image in this paper.

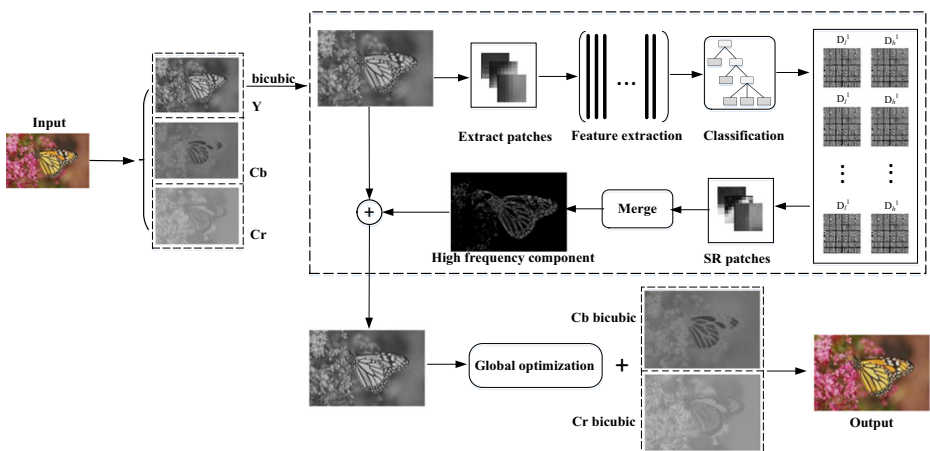


Fig. 1 The flowchart of proposed method

2.1 SR reconstruction constraint

In general, the SR reconstruction constraint assumes that the LR image Y is generated by HR image X which suffers from a series of degradation factors. The constraint model is:

$$Y = DHX + n \quad (1)$$

Where D , H and n represent down-sampling operator, blurring filter, and random noise respectively. The purpose of SR reconstruction is to recover the HR image X as accurately as possible according to the LR image Y . However, the degradation factors cause the loss of high-frequency detail, which leads to infinite solutions of Eq. (1). Therefore, we need to add a priori knowledge to constrain the solution space and determine the final solution.

2.2 Sparse coding

Each atom in the dictionary represents a certain geometric structure of patches. Given an over-complete dictionary $D \in \mathbb{R}^{d \times n}$ ($d \ll n$), the sparse coding utilizes a linear combination of certain atoms in the dictionary to approximate any patches x ($x \in \mathbb{R}^d$). There are two issues to note in this procedure:

Issue 1: The accuracy of target patches using dictionaries to approximate.

Issue 2: The sparsity of coding about the target patches.

The accuracy determines the quality of reconstructed HR images, and the sparsity determines the optimal degree of selected atoms. To some extent, the sparsity can alleviate the over-fitting problem of reconstructed images. Therefore, we always want to obtain the best approximation with the least atoms. The classical method achieves this goal by minimizing the reconstruction error in the condition of ensuring sparsity as follows:

$$\min \|\alpha\|_0 \quad s.t. \quad \|x - D\alpha\|_2^2 \leq \varepsilon \quad (2)$$

Where α represents the sparse coefficient of x , ε represents a very small constant used to ensure accuracy. Equation (2) can be converted into an equivalent form via the Lagrange multiplier method:

$$\min_{\alpha} \|x - D\alpha\|_2^2 + \lambda \|\alpha\|_0 \quad (3)$$

Where the parameter λ balances the accuracy and sparsity of solution. In this paper, we will choose l_0 -norm to constrain the sparsity, which can be solved by OMP algorithm.

2.3 OMP algorithm

OMP approximates the target patch using a series of dictionary atoms under a given sparsity T , and the residual after selecting atoms every time is orthogonal with current atoms. For a signal x , it can be denoted as follows:

$$x = \sum_{i=1}^k a_i^k d_i + r_k(x), \text{ with } \langle r_k(x), d_i \rangle = 0, i = 1, \dots, k \quad (4)$$

Where a_i^k represents the coefficient that is updated after selecting the k -th atom d_i , $r_k(x)$ represents the residual after approximating the image signal x with the current k atoms. Atoms

in the dictionary are selected continually according to the orthogonal rule, until meets the condition $k = T$. In this case, $\{a_i\}_{i=1}^T$ are the coefficients of current atoms.

2.4 Image preprocessing

In this section, we detail the preparation work of images in SR reconstruction, including feature extraction and image partition.

Firstly, human eyes have a stronger sensitivity to luminance compared with saturation and chrominance, so we only reconstruct the image of luminance channel in the consideration of efficiency. Specifically speaking, we convert original images to YCbCr color space, and then choose y channel as the image X to be rebuilt.

Secondly, the image X is down-sampled by the sampling factor s ($s \geq 1$) to obtain X_s , and then X_s is up-sampled with same factor s to obtain the LR image X_l . Next, we partition X_l into obtain image patches $X_l = \{x_l^1, x_l^2, \dots, x_l^N\}$. In addition, we extract the high frequency of the image X as the HR image X_h by $X_h = X - X_l$, and the HR patches $X_h = \{x_h^1, x_h^2, \dots, x_h^N\}$ can be obtained in the same way.

Thirdly, in order to better process the patch match and improve the accuracy of sparse coefficients, we extract four groups of feature vectors for LR patches $\{x_l^i\}_{i=1}^N$ according to the following filters, and then these vectors are concatenated into one vector as the final representation of the LR patch.

$$f_1 = [1, 0, 0, -1], f_2 = [1, 0, 0, -2, 0, 0, 1], f_3 = f_1^T, f_4 = f_2^T$$

Finally, we apply PCA method to reduce dimension under the condition of maintaining 99.99% energy of original images so as to speed up the efficiency of SR reconstruction. And the images after dimension reduction will be used as our final LR training set.

3 The classification of patches and dictionary learning

In this section, we first introduce how to start supervised classification with decision tree, and then describe how to train the specialized dictionary pairs with each training set. Finally, we introduce how we find their relationship between the HR and LR training results and apply it to subsequent image reconstruction.

3.1 Classifying the training set via decision tree

As discussed in Section 1, because of some characteristics of unsupervised clustering, it is necessary to introduce a supervised and deterministic clustering method into the classification of patches. So we extract geometric features from images and apply them to our classification algorithm based on decision tree.

There are so many classic geometric features we can use in the images, such as SFIT [21], HOG [6], non-separable wavelet [15] and so on. But in this paper, we will use other useful features including variance, gradient feature and the angle of the gradient field to classify these patches into three categories: smooth patch, dominant orientation patch, stochastic patch. The pixels in the smooth patch have little or no difference. A clear and specific direction exists in the dominant orientation patch, which appears in the region of texture transition frequently.

And the stochastic patch has neither smooth texture of the former nor specific direction of the latter, but multiple directions or random points which are different clearly from overall texture in the patch.

Now, we analyze geometric features in different categories, and then establish a decision tree according to them. The following geometric features are selected to apply in classification.

1. The pixels in the smooth patch have little or no difference, which leads to its variance τ smaller than the other two. So the variance is chosen as the first feature of decision tree.
2. The distinct difference between the latter two is whether patches have a specific texture direction. But it is difficult to distinguish them by simple statistical information. In this paper, we take the idea of decomposing the gradient field in [13], so as to extract texture direction within the patch. We denote $f(x, y)$ as an image, the gradient at the point (x_i, y_j) is given as follows:

$$g_i = \nabla f(x_i, y_j) = [f'_x(x_i, y_j), f'_y(x_i, y_j)] \quad (5)$$

Where $f'_x(x_i, y_j)$ and $f'_y(x_i, y_j)$ represent the partial derivative of $f(x, y)$ at x and y axes respectively. The local gradient field $G \in \mathbb{R}^{n \times 2}$ for each patch can be denoted as:

$$G = [g_1, g_2, \dots, g_N]^T \quad (6)$$

Then we perform the singular value decomposition (SVD) on the gradient field to obtain $G = USV^T$, where the first column v_1 and the second column v_2 in the matrix V represent dominant and sub-dominant directions, and the singular value σ_1 and σ_2 in the matrix S represent the corresponding energy values respectively. Therefore, the second geometric feature γ is defined as:

$$\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1 + \sigma_2} \quad (7)$$

Obviously, $\gamma \in [0, 1]$. And moreover, we can see from Eq. (7) that the greater the difference between two singular values is, the greater the geometric feature γ , and the greater the probability that this patch belongs to the dominant orientation patch. Otherwise, this patch is more likely to be the stochastic patch.

3. Finally, as we know, the dominant direction of patch can be obtained by rotating the gradient field's by 90 degrees. So we can utilize angle θ between horizontal direction and dominant direction to further classify the dominant orientation patch. The equation of θ can be written as:

$$\theta = \arctan \frac{v_1(y)}{v_1(x)} + \frac{\pi}{2} \quad (8)$$

Here, $v_1(x)$ and $v_1(y)$ represent corresponding coordinate values of dominant orientation within the gradient field.

In conclusion, we choose three types of geometric features as our priori knowledge and establish a decision tree shown in Fig. 2 for classification of image patches.

Specifically speaking, we divide the interval $[0, \pi]$ with step size $\pi/6$ and ensure that all patches fall into corresponding categories. And we assume the training set as $X = \{x_1, x_2, \dots, x_N\}$ in which each patch x_i corresponds to a record $x_i = [\tau_i, \gamma_i, \theta_i]$. As shown in Fig. 2, we first classify the training set with variance τ_i , if $\tau_i \leq \tau^*$, the current patch x_i belongs to the smooth patch, where τ^* is a constant threshold variance. Otherwise, we continue to judge by geometric feature γ_i whether the patch x_i belongs to stochastic patch or not. If $\gamma_i \leq \gamma^*$, it is stochastic patch. Otherwise, we need to use the last geometric feature θ_i to classify x_i to corresponding sub-interval.

We can obtain different categories of image patches by this decision tree. Each category contains the similar patches, but different from other categories. We can use each category of patches to training corresponding dictionaries, including the HR and LR dictionary. Such dictionaries will have better representation ability for corresponding kind of patches so that our algorithm works better for reconstructing images. Figure 3 shows a part of classification results based on decision tree, here, (a) and (e) represent smooth patch and stochastic patch respectively, (b)~(d) sequentially represent the dominant orientation intervals of $[0, \pi/6]$, $(\pi/6, \pi/3]$ and $(\pi/3, \pi/2]$, (f)~(h) represent sequentially intervals of $(5\pi/6, \pi]$, $(2\pi/3, 5\pi/6]$ and $(\pi/2, 2\pi/3]$.

3.2 Training independent dictionaries

Most of classic dictionary training methods are based on the assumption of coefficient invariance, such as the joint dictionary training in [38], anchor neighbor regression in [17, 28, 29] and so on. However, it tends to have a limitation on dictionary training based on this assumption, which restricts expression ability of dictionary atoms for corresponding feature

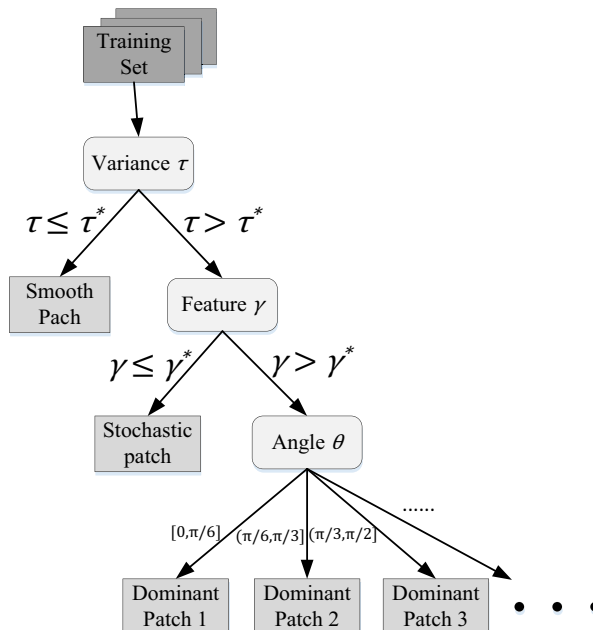


Fig. 2 The decision tree for classification

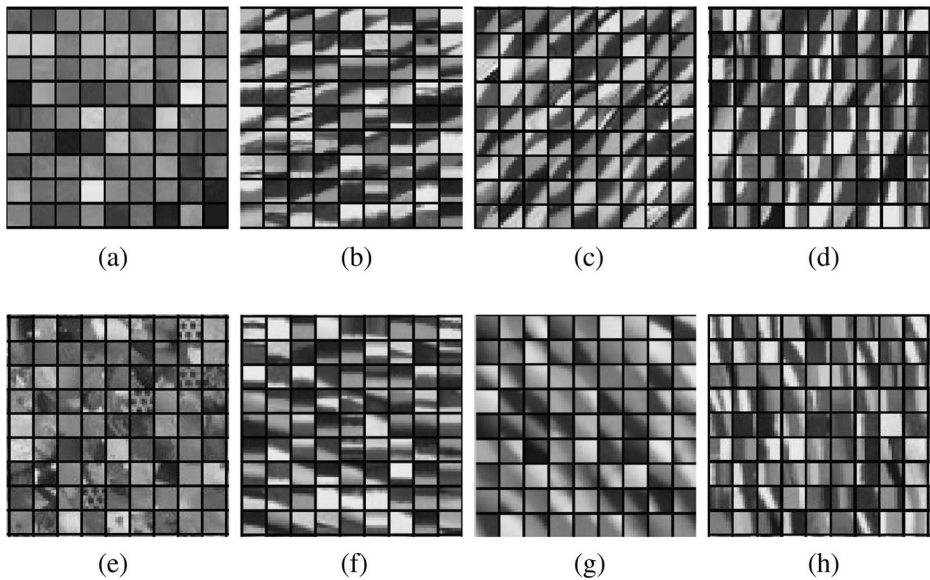


Fig. 3 The local classification results of classification

and consequently affects the quality of reconstructed images. Actually, the two types of coefficients may not be strictly equal, since the HR image prone to have more details than the LR one. So we propose a new effective method which independently trains the LR and HR sub-dictionaries according to Eqs. (9) and (10) respectively based on KSVD.

$$D_l^k, A_l^k = \arg \min_{D_l^k, A_l^k} \left\| x_i^{l,k} - D_l^k \alpha_i^{l,k} \right\|_2^2 \quad s.t. \quad \left\| \alpha_i^{l,k} \right\|_0 \leq T, i = 1, 2, \dots, N \quad (9)$$

$$D_h^k, A_h^k = \arg \min_{D_h^k, A_h^k} \left\| x_i^{h,k} - D_h^k \alpha_i^{h,k} \right\|_2^2 \quad s.t. \quad \left\| \alpha_i^{h,k} \right\|_0 \leq T, i = 1, 2, \dots, N \quad (10)$$

Here, $D_l^k \in \mathbb{R}^{d_l \times n}$ represents the LR dictionary trained by the k -th training set, $A_l^k \in \mathbb{R}^{n \times N}$ represents the LR coefficient matrix of the k -th training set, $D_h^k \in \mathbb{R}^{d_h \times n}$ and $A_h^k \in \mathbb{R}^{n \times N}$ represent the corresponding HR dictionary and coefficient matrix respectively, $\alpha_i^{l,k}$ and $\alpha_i^{h,k}$ represent coefficients of the LR patch $x_i^{l,k}$ and HR patch $x_i^{h,k}$ respectively. T represents sparsity constraint, which is an integer constant. d_l and d_h represent the dimension of LR and HR atoms respectively, n and N represent the number of atoms and image patches respectively.

We take D_l^k as an example for more details of our training method.

1. Preparation: Initialize the LR dictionary D_l^k with random samples in the training set.
2. The sparse coding phase

For a given training set $X_l^k = \{x_1^{l,k}, x_2^{l,k}, \dots, x_N^{l,k}\}$, we first compute the sparse coefficient of each item in X_l^k , and then arrange them by column to obtain the coefficient

matrix A_l^k . The problem with l_0 -norm can be solved by OMP, and the objective equations are given as follows:

$$\alpha_i^{l,k} = \arg \min_{\alpha_i^{l,k}} \left\| x_i^{l,k} - D_l^k \alpha_i^{l,k} \right\|_2^2 \quad s.t. \left\| \alpha_i^{l,k} \right\|_0 \leq T, \quad i = 1, 2, \dots, N \quad (11)$$

$$\alpha_i^{h,k} = \arg \min_{\alpha_i^{h,k}} \left\| x_i^{h,k} - D_h^k \alpha_i^{h,k} \right\|_2^2 \quad s.t. \left\| \alpha_i^{h,k} \right\|_0 \leq T, \quad i = 1, 2, \dots, N \quad (12)$$

3. The dictionary updating phase

In this phase, we start with updating each atom d_i ($d_i \in D_l^k, 1 \leq i \leq n$) in the current dictionary and the coefficients A_T^i ($A_T^i \in A_l^k, 1 \leq i \leq n$) associated with the atom d_i , where A_T^i represents the i -th row in the coefficient matrix A_l^k . The updating form can be written as follows:

$$D_l^k = \arg \min \left\| X_l^k - D_l^k A_l^k \right\|_F^2 = \arg \min \left\| X_l^k - \sum_{j=1}^n d_j A_T^j \right\|_F^2 \quad (13)$$

Then, we define error matrix E_i to indicate the direction towards which d_i should be updated. In other words, E_i represents the component that the sub-item $d_i A_T^i$ needs to be represented:

$$E_i = X_l^k - \sum_{j=1, j \neq i}^n d_j A_T^j \quad (14)$$

Therefore, we decompose E_i with SVD, which aims at approximating E_i with a singular value which possesses the most energy. However, when updating A_T^i , it tends to cause the non-sparseness of coefficients if E_i is decomposed by SVD directly. So we create the index ω_i to indicate the nonzero entries in A_T^i :

$$\omega_i = \{j | 1 \leq j \leq N, A_T^i(j) \neq 0\} \quad (15)$$

Next, we create a mapping matrix $\Phi_i \in R^{N \times |\omega_i|}$ with ones on $(\omega_i(j), j)$ th entries and zeros elsewhere according to Eq. (15), in order to retain the non-zero entries in A_T^i after mapping. It means $A_R^i = A_T^i \Phi_i$, where A_R^i represents the result after arranging the non-zero entries in A_T^i sequentially. Next, we treat E_i in the same way to obtain $E_R^i = E_i \Phi_i$, and then decompose E_R^i with SVD as follows:

$$E_R^i = U \Lambda V^T = \sum_{i=1}^k u_i \sigma_i v_i^T \quad (16)$$

Where σ represents singular values, u and v represent corresponding singular vectors. Generally speaking, the greater the singular value σ_i , the smaller the difference from the matrix E_R^i , and the more energy it contains. Therefore, we choose the greatest item $u_m \sigma_m v_m^T$ to update the atom and its coefficients. In other words, we update d_i with u_m and A_T^i with $\sigma_m v_m^T$ according to corresponding index ω_i .

Repeat steps 2 and 3 until meeting the stop condition of iteration. Now, we get the LR dictionary D_l^k and its coefficient matrix A_l^k . Similarly, the HR dictionary D_h^k and its coefficient matrix A_h^k can be obtained in the same way according to Eq. (10).

3.3 Learning multiple mapping functions

We can obtain K pairs of training set $\{X_l^k, X_h^k\}_{k=1}^K$ after classifying the training set based on decision tree, and then we train the dictionary pair $\{D_l^k, D_h^k\}$ and corresponding coefficient matrix $\{A_l^k, A_h^k\}$ according to each sub-set. However, there is an obvious difference between the LR coefficient and the HR one due to the independent dictionary training. Although the dictionary can represent respective training set well, the reconstructed images tend to be inferior if we insist on replacing the HR coefficient matrix A_h^k with the matrix A_l^k , due to not only the atom selection, but also under-fitting or over-fitting. So it is necessary to solve their mapping function between A_l^k and A_h^k in order to achieve the better reconstruction effect.

As we know, the patches in the same training set have similar geometric feature, so their sparse coefficients are similar too. Considering quality and efficiency, the linear mapping is sufficient to solve a relational mapping function. In this paper, we solve it with the least squares, and the mapping function of k-th training set can be written as:

$$M_k = \arg \min_{M_k} \|A_h^k - M_k A_l^k\|_F^2 \quad (17)$$

The closed-form expression of Eq. (17) is given as:

$$M_k = A_h^k A_l^{kT} (A_l^k A_l^{kT})^{-1} \quad (18)$$

Next, we apply Eq. (18) for each category to obtain corresponding mapping matrix $\{M_k\}_{k=1}^K$.

4 Image reconstruction and optimization

This section mainly introduces how to reconstruct images with independent dictionaries, and then describes the global optimization of reconstructed images.

4.1 Reconstructing image

For a given image $Y = \{y_1, y_2, \dots, y_N\}$ to be reconstructed, we first classify each patch in Y into k-th training set with decision tree, and then compute its coefficient α_i^l using corresponding LR dictionary D_l^k . The problem can be denoted as:

$$\alpha_i^l = \arg \min \|\gamma_i - D_l^k \alpha_i^l\|_2^2 + \lambda \|\alpha_i^l\|_0 \quad (19)$$

Then, we still compute the corresponding HR coefficient α_i^h by using the LR coefficient α_i^l and HR dictionary D_h^k , in order to ensure a better reconstruction effect. The specific method is given as follows:

$$\alpha_i^h = M_k \alpha_i^l \quad (20)$$

And our reconstruction method is shown as follows:

$$x_i = D_h^k \alpha_i^h \quad (21)$$

Next, we can obtain the reconstruction model with Eqs. (18), (20) and (21):

$$x_i = D_h^k M_k \alpha_i^l = D_h^k A_h^k A_l^{kT} (A_l^k A_l^{kT})^{-1} \alpha_i^l \quad (22)$$

From Eq. (22), we can obtain the reconstructed patch x_i with the coefficient α_i^l , during which we use sparse coding only one time. Finally, the HR image can be obtained by merging all reconstructed patches x_1, x_2, \dots, x_N .

4.2 Enforcing global optimization

There are three points to note when recalling our reconstruction procedure. Firstly, we approximate the LR patch y with the dictionary D based on Eq. (1), which leads to inequality between the approximation item $D\alpha$ and the patch y , which means $y \approx D\alpha$. Secondly, the pixel values in overlap region are average. Finally, the noise has a degradation effect on image. Therefore, the reconstructed image X may not be completely accurate to meet the reconstruction model shown in Eq. (1). Here, we adopt a Maximum a posteriori (MAP) framework to ensure that the reconstructed image X satisfies this reconstruction model with the given LR image Y . In other words, we add two constrains to this framework based on Eq. (1), which are the auto-regressive model and non-local means model.

4.2.1 Regularization by non-local self-similarity

There are many non-local redundancies in natural images which can greatly help us to suppress artifacts in image SR reconstruction. Therefore, we introduce the non-local means (NLMs) to our MAP framework for enhancing our reconstruction result, just like other SR methods (e.g., [9, 32, 44, 49]). This item can captures the non-local self-similarity well in natural images. The MAP framework can be expressed as:

$$X^* = \arg \min_X \|Y - DHX\|_2^2 + \delta \|(I - W)X\|_2^2 \quad (23)$$

Where the first term is fidelity term, the second term is NLMs term, D represents the down-sampling operator, H represents the blurring filter, I is an identity matrix and W is NLMs similar weight matrix. The δ aims at find a balance between the first and two terms, because a larger δ can make the reconstructed image the more smooth and a smaller δ will have no enough ability to suppress artifacts.

4.2.2 Regularization by auto-regressive model

A local area in a natural image can be viewed as a stationary process, which can be well modeled by the auto-regressive (AR) models. So, we learn a set of AR models from the classified high quality training image patches, and select one AR model to regularize the input

image patch. The AR model aims to predict the central pixel of the patch by using the neighboring pixels. It can be expressed as:

$$a_k = \arg \min_a \sum_{x_i \in S_k} (p_i - a^T q_i)^2 \quad (24)$$

Where S_k is one of the sets of classified patches, p_i is the central pixel of the patch x_i , q_i is the vector that consists of the neighboring pixels of p_i , and a is the corresponding weight vector. Based on the above considerations, the MAP framework can be written as:

$$X^* = \arg \min_X \|Y - DHX\|_2^2 + \rho \|(I - A)X\|_2^2 \quad (25)$$

Where A is the AR weight matrix defined in [9], ρ is the similar parameter with δ in Eq. (23). More details can be found in [9].

4.2.3 Global SR reconstruction framework

By incorporating both the non-local self-similarity regularization and the auto-regressive constraint into the MAP framework, we can use the following method to enhance the SR reconstruction performance:

$$X^* = \arg \min_X \|Y - DHX\|_2^2 + \delta \|(I - W)X\|_2^2 + \rho \|(I - A)X\|_2^2 \quad (26)$$

The solution to (26) can be efficiently computed based on iterative optimization, and each step of iteration can be finished by the following processing:

$$X_{t+1} = X_t + \mu \left\{ (DH)^T (Y - DHX_t) - \delta (I - W)^T (I - W)X_t - \rho (I - A)^T (I - A)X_t \right\} \quad (27)$$

Where the μ is the step size of gradient descent algorithm, a too small μ will make the algorithm converge slowly and a too large one maybe cause the algorithm not to converge. We can iterate this method until the program reaches the requirements we set in advance.

5 Experimental results

In this section, we first introduce uniform configuration of experiments, and then we compare our experimental results with the sparse coding methods of Yang et al. [38] and Zeyde et al. [46], the implementations based on the least squares in Chang et al. [4] (NE + LLE) and Bevilacqua et al. [3] (NE + NNLS), the ANR method in Timofte et al. [28], as well as the deep network in Dong et al. [10]. In order to facilitate the narrative, we label the initial results reconstructed by our method as CISR-1 which will be compared with other methods by default, and the results after global optimization as CISR-2. All methods use the same training set which contains 91 natural images except SRCNN which uses the dataset of imageNet, and SRCNN's effect with imageNet is better than the effect of 91 natural images. The test set is divided into two categories: Set5 and Set14. Figures 4 and 5 show the pictures of Set5 and Set14 respectively. In addition, we try our best to keep the other experimental configuration the same. The specific description of experiments is organized as follows:

In Section 5.2, we compare and analyze the comparison results based on Set14 in terms of image quality. In Section 5.3, the comparison results are shown for the experiments with



Fig. 4 Test images in Set5. From left to right is baby, leaves, bird, woman and flower

magnification 2, 3 and 4 based on Set5. In Section 5.4, we mainly show the experiments about classification, including the comparison with the results based on K-means, the experiment with different values of classification properties, such as variance τ and geometric feature γ . In Section 5.5, we show the results of experiments with different parameters in the reconstruction phase, including dictionary size and sparsity T .

5.1 Experimental settings

In this paper, we choose the training set used in Yang et al. [38], which contains 91 HR natural images. We perform the preparation operations, such as image partition, extracting feature and dimensionality reduction, on these images with the method described in Section 2.4, so as to obtain the HR and LR training sets. The first-order difference is applied to solve the partial guidance by default in the classification phase. We made a lot of experiments about the choice of τ^* and γ^* , and then we found that it was rational to set $\tau^* = 0.001$, which could be got it from Section 5.4.2. And for the threshold of geometric feature γ , we will find suitable value with the probability density function (PDF) in [11]. And its equation is as follows:

$$f(\gamma) = 4(n-1)\gamma \frac{(1-\gamma^2)^{n-2}}{(1+\gamma^2)^n} \quad (28)$$

Where n represents the number of pixels in patches. Figure 6 shows corresponding curves of PDF about γ when the patch sizes are 6×6 、 9×9 and 12×12 respectively. As we can see from the trend of curves, the patches classified by decision tree belong to the dominant orientation patch basically when the patch size is 6×6 and $\gamma \geq 0.3$, so we choose $\gamma^* = 0.3$ as the threshold of the geometric feature γ . Similarly, we set $\gamma^* = 0.2$ when the patch size is 9×9 , and set $\gamma^* = 0.17$ when the patch size is 12×12 . In addition, the sparsity is set as $T = 20$. In the global optimization, We empirically set the parameters $\mu = 2$, $\delta = 0.05$, $\rho = 0.03$ for an effective optimization.

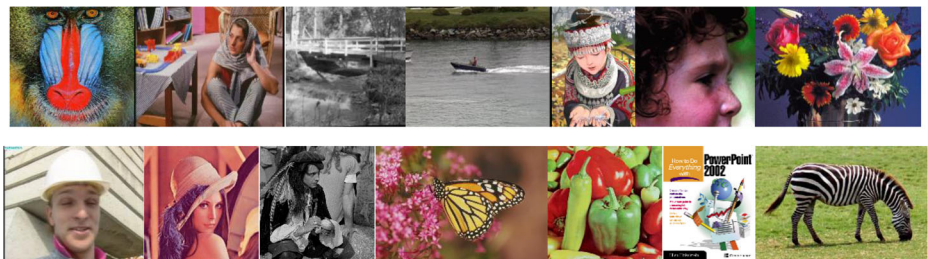


Fig. 5 Test images in Set14. The top line from left to right is baboon, barbara, bridge, coastguard, comic, face, flowers, the bottom line is foreman, lena, man, monarch, pepper, ppt, zebra

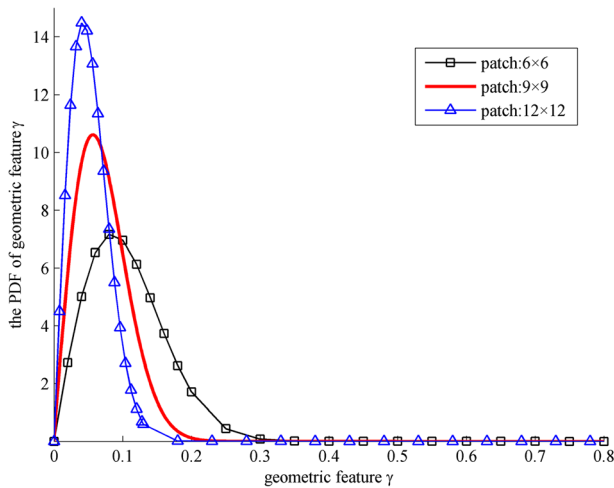


Fig. 6 The curves of PDF for geometric feature γ

5.2 The comparing results with different methods

This section focuses on the comparison with the results produced by other methods, and the analysis in terms of the quality of reconstruction results. Here, we partition images to obtain patches with the configuration that the patch size is 9×9 and the magnification is 3. Then we extract feature from these patches and reduce dimension for corresponding feature vectors. Next, these patches are classified by decision tree with the property $\tau^* = 0.001$ and $\gamma^* = 0.2$. Finally, these patches are trained as corresponding dictionaries with 1024 atoms. Table 1 shows the specific comparison results in this experiment, the top line for every image is PSNR, and the bottom line is SSIM.

As we can see from the Table 1, Zeyde's and ANR's reconstruction effects are very close, but better than Yang's, NE + LLE's and NE + NNLS's. Our reconstruction effect of CISR-1 is close to SRCNN's, and the version CISR-2 after global optimization is the best one and exceeds SRCNN's obviously. The average value of CISR-2 is 0.36 higher than SRCNN's in PSNR, and 0.0216 higher than SRCNN's in SSIM. But our results are not always the best, the reason for this, I think, can be explain as: 1. Our training set does not have many as imageNet's. 2. It is hard to reconstruct image for most methods if the input image is very complicated. So, the reconstruction effects are very close for good methods.

Additionally, we can see from the comparison between CISR-1 and CISR-2, the image quality is further improved after global optimization, which indicates that our global optimization have a positive effect, and the reconstructed image further meets the reconstruction model shown in Eq. (1).

5.3 The comparing results with different magnifications

In this section, we mainly show the experiments with the magnification 2, 3 and 4 based on Set5. In order to avoid serious block effect and ensure good reconstruction effect, we set the patch size 6×6 with 4 pixel overlap when the magnification is 2. Similarly, we set 9×9 with 6 pixel overlap when the magnification is 3, and set 12×12 with 8 pixel overlap when the magnification is 4. The threshold τ^* and γ^* follow the configuration described in Section 5.1,

Table 1 Comparing with other methods for magnification 3

Images	Bicubic	Yang	Zeyde	NE + LLE	NE + NNLS	ANR	SRCNN	CISR-1	CISR-2
Baboon	23.21	23.47	23.52	23.55	23.49	23.56	23.63	23.64	23.40
	0.5439	0.5878	0.5906	0.5969	0.5891	0.5988	0.6046	0.6043	0.6418
Barbara	26.25	26.39	26.77	26.74	26.67	26.71	26.63	26.65	27.16
	0.7531	0.7633	0.7821	0.7817	0.7779	0.7819	0.7813	0.7821	0.8040
Bridge	24.40	24.82	25.02	24.96	24.88	24.99	25.14	25.09	25.57
	0.6483	0.6921	0.6970	0.6992	0.6918	0.7009	0.7066	0.7046	0.7394
Coastguard	26.55	27.02	27.15	27.07	26.95	27.08	27.31	27.20	27.86
	0.6149	0.6393	0.6554	0.6566	0.6521	0.6582	0.6626	0.6580	0.7051
Comic	23.12	23.90	23.97	23.97	23.84	24.01	24.72	24.25	24.97
	0.6988	0.7557	0.7558	0.7571	0.7498	0.7592	0.7776	0.7710	0.8055
Face	32.82	33.11	33.52	33.55	33.44	33.60	33.57	33.68	34.17
	0.7984	0.8011	0.8198	0.8213	0.8177	0.8228	0.8224	0.8228	0.8396
Flowers	27.23	28.25	28.42	28.35	28.21	28.44	29.09	28.68	29.52
	0.8013	0.8297	0.8377	0.8376	0.8328	0.8395	0.8491	0.8436	0.8685
Foreman	31.18	32.04	33.16	33.10	32.91	33.12	33.70	33.73	34.34
	0.9058	0.9132	0.9292	0.9282	0.9250	0.9289	0.9347	0.9355	0.9425
Lena	31.68	32.64	33.00	33.01	32.84	33.06	33.67	33.45	34.16
	0.8582	0.8648	0.8784	0.8791	0.8755	0.8803	0.8582	0.8835	0.8962
Man	27.01	27.76	27.91	27.87	27.81	27.91	28.26	28.11	28.65
	0.7495	0.7749	0.7863	0.7872	0.7824	0.7886	0.7962	0.7933	0.8162
Monarch	29.43	30.71	31.09	30.89	30.75	31.02	32.68	31.63	32.57
	0.9198	0.9290	0.9381	0.9356	0.9342	0.9370	0.9469	0.9423	0.9510
Pepper	32.39	33.33	34.04	33.77	33.69	33.79	34.42	34.29	34.89
	0.8698	0.8669	0.8862	0.8842	0.8822	0.8848	0.8887	0.8887	0.8965
PPT	23.71	24.98	25.21	24.90	24.98	24.97	26.73	25.47	26.22
	0.8746	0.8918	0.9112	0.9003	0.9002	0.9006	0.9286	0.9154	0.9235
Zebra	26.63	27.95	28.51	28.30	28.10	28.42	28.83	28.90	29.99
	0.7942	0.8259	0.8416	0.8403	0.8336	0.8425	0.8486	0.8475	0.8787
Average	27.54	28.31	28.66	28.57	28.47	28.62	29.17	28.91	29.53
	0.7736	0.7954	0.8078	0.8075	0.8032	0.8089	0.8147	0.8138	0.8363

The Bold data is the best reconstruction result per line

and the dictionary size is set 1024. Table 2 shows the experiment results for magnification 2, 3 and 4, and the top line for each image is PSNR, the bottom line is SSIM.

As we can see from Table 2, when the magnification is 2, the SRCNN based on deep network has the best effect, it is because deep network can work better if the more details exist in image. But when the magnifications are 3 and 4, the interpolation images lost too many details, SRCNN cannot reconstruct the better images. However, the proposed method CISR-2 works better than SCRNN, and it takes advantage of image itself to reconstruct the HR image and gets a good effect. The average of CISR-2 is 0.67 higher than SRCNN in PSNR, and 0.0099 higher than SRCNN in SSIM when the magnification factor is 2. Similarly, the average of CISR-2 is 0.98 higher than SRCNN in PSNR, and 0.0227 higher than SRCNN in SSIM when the magnification factor is 3. Therefore, we can further infer from the comparison result that compared with other methods, the performance of our method will be better as the magnification increases.

Figure 7 shows the reconstruction images of different methods, all images use the magnification factor 3. As we can see from these images, NE + NNLS, NE + LLE and ANR generate the HR images with severely blurring artifacts, they are not good at solving deblurring problem. CISR-1 and SRCNN, especially CISR-2, which look much better than others, tend to generate the more shaper results. In addition, CISR-2 obviously contains the more details than CISR-1, which indicates our global optimization is useful.

Table 2 Comparison results for magnification 2, 3 and 4

Images	Scale	Bicubic	Yang	Zeyde	NE + LLE	NE + NNLS	ANR	SRCNN	CISR-1	CISR-2
Baby	×2	37.09	—	38.25	38.32	38.06	38.43	38.54	38.39	38.52
Leaves	×2	0.9520	—	0.9632	0.9631	0.9616	0.9645	0.9655	0.9643	0.9651
		40.02	—	41.27	41.16	41.04	41.30	41.37	41.30	41.46
		0.9510	—	0.9603	0.9597	0.9592	0.9610	0.9613	0.9604	0.9614
Bird	×2	36.83	—	39.97	40.01	39.50	40.02	40.91	40.35	40.66
		0.9721	—	0.9838	0.9836	0.9820	0.9843	0.9859	0.9847	0.9856
Woman	×2	32.15	—	34.44	34.43	34.22	34.47	35.37	34.94	35.09
		0.9476	—	0.9646	0.9644	0.9633	0.9651	0.9687	0.9668	0.9677
Flowers	×2	30.40	—	32.25	32.37	32.10	32.46	33.25	32.64	32.78
		0.8921	—	0.9262	0.9283	0.9240	0.9295	0.9363	0.9310	0.9326
Average	×2	35.30	—	37.24	37.26	36.98	37.34	37.89	37.52	37.70
		0.9430	—	0.9596	0.9598	0.9580	0.9609	0.9635	0.9614	0.9625
Baby	×3	33.92	34.29	35.09	35.09	34.85	35.13	35.25	35.24	36.06
		0.9039	0.9043	0.9213	0.9216	0.9185	0.9226	0.9241	0.9234	0.9371
Leaves	×3	37.28	37.49	38.49	38.41	38.26	38.48	38.58	38.63	39.46
		0.9174	0.9155	0.9321	0.9313	0.9296	0.9324	0.9331	0.9328	0.9427
Bird	×3	32.58	34.11	34.52	34.49	34.25	34.52	35.48	35.08	36.36
		0.9256	0.9391	0.9476	0.9475	0.9436	0.9484	0.9549	0.9522	0.9628
Woman	×3	28.57	29.94	30.34	30.22	30.02	30.27	31.37	30.93	31.78
		0.8896	0.9037	0.9175	0.9160	0.9131	0.9163	0.9291	0.9237	0.9347
Flowers	×3	27.38	28.42	28.53	28.56	28.36	28.64	29.25	28.92	29.62
		0.7865	0.8268	0.8300	0.8315	0.8251	0.8335	0.8494	0.8418	0.8628
Average	×3	31.95	32.85	33.39	33.35	33.15	33.41	33.99	33.76	34.66
		0.8846	0.8979	0.9097	0.9096	0.9060	0.9106	0.9181	0.9148	0.9280
Baby	×4	31.78	—	33.18	32.94	32.83	33.03	33.13	33.23	34.36
		0.8567	—	0.8835	0.8787	0.8765	0.8804	0.8824	0.8830	0.9090
Leaves	×4	35.37	—	36.33	36.16	36.13	36.22	36.41	36.42	37.67
		0.8864	—	0.9022	0.8989	0.8986	0.9006	0.9026	0.9031	0.9211
Bird	×4	30.18	—	31.79	31.60	31.44	31.66	32.52	32.22	33.56
		0.8729	—	0.8996	0.8965	0.8945	0.8978	0.9112	0.9070	0.9321
Woman	×4	26.47	—	27.88	27.66	27.56	27.73	28.89	28.47	29.64
		0.8318	—	0.8620	0.8566	0.8547	0.8571	0.8837	0.8737	0.8996
Flowers	×4	25.73	—	26.59	26.57	26.52	26.67	27.10	26.88	27.74
		0.6971	—	0.7421	0.7413	0.7371	0.7450	0.7658	0.7556	0.7971
Average	×4	29.91	—	31.15	30.99	30.90	31.06	31.61	31.44	32.59
		0.8290	—	0.8579	0.8544	0.8523	0.8562	0.8691	0.8645	0.8918

The Bold data is the best reconstruction result per line



Fig. 7 The top line, left to right: the original, bicubic, Yang, Zeyde, ANR. The bottom line, left to right: NE + NNLS, NE + LLE, SRCNN, CISR-1, CISR-2

5.4 Classification-related experiments

5.4.1 Comparison with reconstruction results based on K-means

To further verify the effectiveness of our classification algorithm, we compare our reconstruction results with the one of K-means in terms of PSNR and SSIM. Set5, Set14, B100(100 images) and kodak(24 images) are chosen as our test sets. Additionally, we set magnification 3, the patch size 9×9 , and the dictionary size 1024 in this experiment. To ensure the fairness of comparison, the training set is clustered 8 categories by K-means, which will be trained corresponding sub-dictionaries independently for reconstructing images. Then we compute the PSNR and SSIM of all images in each test set and then show their average for comparison as shown in Table 3. The experiment indicates that our decision tree have a better effect.

5.4.2 The performance comparison of different classification parameter

This section focuses on the experiment with different τ^* and γ^* , in which the experimental configuration is the same as last section. Figure 8 shows the trend of image quality in terms of PSNR and SSIM as the property γ^* increases gradually. Each point on curves represents the average PSNR which is computed by 14 images in Set14.

As shown in the two figures in Fig. 8, the general trend of PSNR and SSIM is similar. With the increase of the threshold γ^* of geometric feature, the average PSNR rises at first, and then begins to decrease sharply when the property γ^* is about 0.4. This is because the dominant orientation patch becomes less and less as the threshold γ^* increases, which affects the effect of dictionary training directly and thus leads to bad reconstruction effect. On the contrary, the

Table 3 Comparing with K-means

Methods	Metrics	Set5	Set14	B100	kodak
K-means	PSNR	32.92	28.80	28.05	29.27
	SSIM	0.9201	0.8119	0.7769	0.8097
CISR-1	PSNR	33.29	28.91	28.13	29.38
	SSIM	0.9243	0.8138	0.7781	0.8113

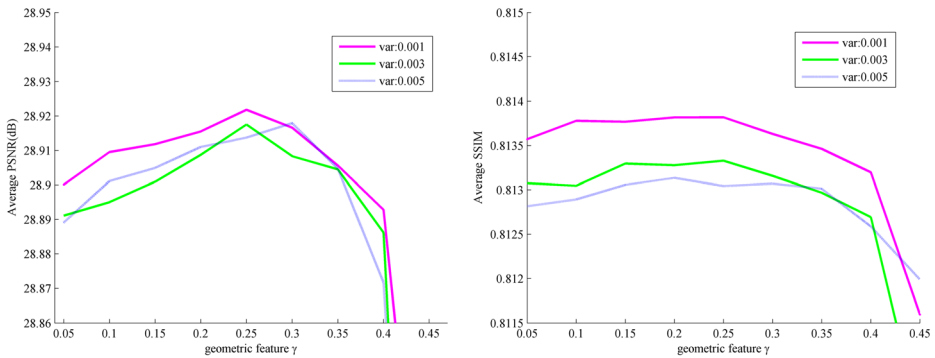


Fig. 8 The curves of experiment with different property τ and γ

small variance τ^* enables more patches to fall in the stochastic category or the dominant orientation category, which makes the better dictionary. Therefore, the magenta curve has an obvious advantage compared with other two.

5.5 Reconstruction parameter-related experiments

This section focuses on the experiments with different dictionary sizes and sparsity values, in which the patch size is set 9×9 and magnification 3. Additionally, we set sparsity $T = 20$ in the former experiment and the dictionary size 1024 in the latter experiment. We compute the PSNR average of images in Set14 as comparative indicator. And we compare our result with other methods in order to further illustrate the effectiveness of our method.

Dictionaries are very important to reconstruct images. Dictionaries contain all kinds of atoms and every atom expresses a certain feature of patches. If dictionaries contain few or less feature atoms, some details existing in natural images cannot be represented well using these atoms. However, if dictionaries contain too many or over-complete feature atoms, the redundancy of atoms will not good for dictionary training and image reconstruction due to the efficiency problem. So, a dictionary with a good size is important for reconstructing images. We make an experiment about dictionary size for a further comparison. Figure 9a shows the

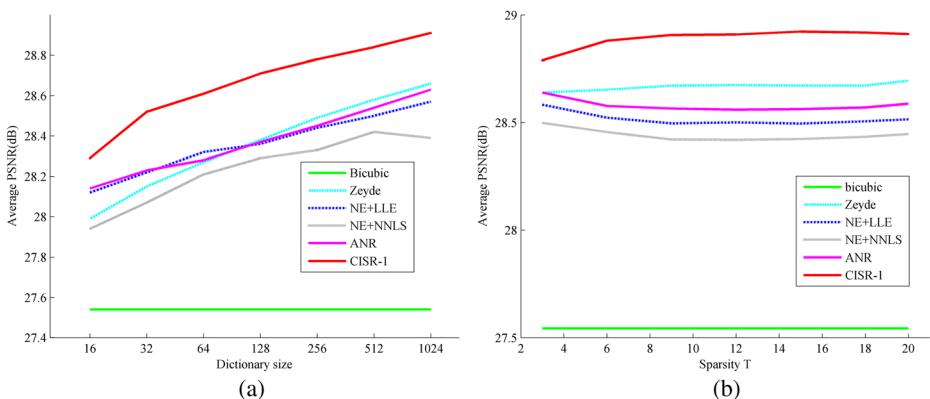


Fig. 9 The results of the experiments with different dictionary size and sparsity T

comparison result with other methods when the dictionary size is different. Obviously, the difference of reconstruction results for NE + LLE, Zeyde and ANR is very slight. Specifically speaking, when the dictionary size is greater than 128, the reconstruction result of Zeyde is better than the ones of NE + LLE and ANR, the performance of ANR is better than NE + LLE, and NE + NNLS is worse than the three above. But our reconstruction result is clearly superior to other methods, and the rising speed is also obvious as the dictionary size increases.

On the other hand, the sparsity T which is used to choose the number of atoms for reconstructing patch. If T is too small, it will not good for expressing corresponding patches. If T is too big, it will not good for reconstruction efficiency. Figure 9b shows the comparison result in the experiment about sparsity. As we can see from the figure, with the increase of sparsity T , the PSNR computed by our method increases gradually, then tends to be stable, and declines slightly at last. But from the location and the trend of the curve, our reconstruction result is clearly superior to others.

6 Conclusion

In this paper, a new and effective method about SR reconstruction is proposed for natural images. We analyze the characteristic of patches and extract three types of geometric features (variance, feature γ and angle θ) which works well on distinguishing patches. So these features are applied to create a decision tree for classification. We classify all kinds of patches into different categories to obtain corresponding sub-sets, which will be trained independently by K-SVD to acquire corresponding dictionaries. Additionally, we compute mapping matrices between the HR and LR training results based on the least squares to ensuring a good and rational reconstruction result. And the good experimental results demonstrate the effectiveness of our method.

Acknowledgements This work is partly supported by the National Natural Science Foundation of China under Project code (61672202).

References

1. Aharon M, Elad M, Bruckstein A (2006) K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans Signal Process* 54(11):4311–4322
2. Allebach J, Wong PW (1996) Edge-directed interpolation. *International conference on image processing, 1996. Proceedings IEEE Xplore*, vol 3, pp 707–710
3. Bevilacqua M, Roumy A, Guillemot C, Alberi Morel M-L (2012) Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: *BMVC, Guildford*, pp 1–10
4. Chang H, Yeung DY, Xiong Y (2004) Super-resolution through neighbor embedding. *Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on IEEE Xplore*, vol 1, pp I-275-I-282
5. Dai S et al (2009) SoftCuts: a soft edge smoothness prior for color image super-resolution. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 18(5):969–981
6. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. *Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on IEEE*, pp 886–893
7. Dong C, Chen CL, He K, Tang X (2014) Learning a deep convolutional network for image super-resolution. In: *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, Cham, pp 184–199
8. Dong W et al (2011) Sparsity-based image denoising via dictionary learning and structural clustering. *IEEE conference on computer vision and pattern recognition IEEE computer society*, pp 457–464
9. Dong W et al (2011) Image Deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 207:1838–1857

10. Dong C, Loy CC, He K, Tang X (2016) Image super-resolution using deep convolutional networks. *IEEE Trans Pattern Anal Mach Intell* 38(2):295–307
11. Ducournau A, Fablet R (2016) Deep learning for ocean remote sensing: an application of convolutional neural networks for super-resolution on satellite-derived SST data. 2016 9th IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS). Cancun, pp 1–6
12. Edelman A (1988) Eigenvalues and condition numbers of random matrices. *SIAM J Matrix Anal Appl* 9(4):543–560
13. Feng XG, Milanfar P (2003) Multiscale principal components analysis for image local orientation estimation. In: *Proceedings of the 36th Asilomar Conference on Signals, Systems and Computers*. University of California, Santa Cruz, pp 478–482
14. Freeman WT, Jones TR, Pasztor EC (2002) Example-based super-resolution. *Comput Graph Appl IEEE* 22(2):56–65
15. Huang J, You X, Yuan Y, Yang F, Lin L (2010) Rotation invariant iris feature extraction using Gaussian Markov random fields with non-separable wavelet. *Neurocomputing* 73(4):883–894
16. Jiang J et al (2016) Noise robust position-patch based face super-resolution via Tikhonov regularized neighbor representation. *Inf Sci* 367–368:354–372
17. Jiang J et al (2016) Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means. *IEEE Trans Multimedia* 19(1):15–26
18. Jiang J, Chen C, Ma J, Wang Z, Wang Z, Hu R (2017) SRLSP: a face image super-resolution algorithm using smooth regression with local structure prior. *IEEE Trans Multimedia* 19(1):27–40
19. Li X, Orchard MT (2001) New edge-directed interpolation. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 10(10):1521–1527
20. Liu H, Han J, Hou S, Shao L, Ruan Y (2017) Single image super-resolution using a deep encoder-decoder symmetrical network with iterative back projection. *Neurocomputing* 282:52–59
21. Lowe DG (2004) Distinctive image features from scale-invariant Keypoints. *Int J Comput Vis* 60(2):91–110
22. Mairal J, Sapiro G, Elad M (2008) Learning multiscale sparse representations for image and video restoration (PREPRINT). *Siam J Multiscale Model Simul* 7(1):214–241
23. Protter M, Elad M, Takeda H, Milanfar P (2009) Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 18(1):36–51
24. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(5500):2323–2326
25. Rubinstein R, Zibulevsky M, Elad M (2008) Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. *Cs Technion* 40(8):1–15
26. Sun J, Xu Z, Shum HY (2008) Image super-resolution using gradient profile prior. *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE Conference on IEEE*, pp 1–8
27. Takeda H, Farsiu S, Milanfar P (2007) Kernel regression for image processing and reconstruction. *IEEE Trans Image Process* 16(2):349–366
28. Timofte R, De V, Gool LV (2013) Anchored neighborhood regression for fast example-based super-resolution. *IEEE international conference on computer vision IEEE*, pp 1920–1927
29. Timofte R, Smet VD, Gool LV (2014) A+: adjusted anchored neighborhood regression for fast super-resolution. *Lect Notes Comput Sci* 9006:111–126
30. Uiboupin T, Rasti P, Anbarjafari G, Demirel H (2016) Facial image super resolution using sparse representation for improving face recognition in surveillance monitoring. 2016 24th Signal Processing and Communication Application Conference (SIU). Zonguldak, pp 437–440
31. Wang YH, Li JB, Fu P (2012) Medical image super-resolution analysis with sparse representation. *Eighth international conference on intelligent information hiding and multimedia signal processing IEEE computer society*, pp 106–109
32. Wang S et al (2012) Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. *IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI* 157(10):2216–2223
33. Wang L, Lu K, Liu P (2015) Compressed sensing of a remote sensing image based on the priors of the reference image. *IEEE Geosci Remote Sens Lett* 12(4):736–740
34. Wang Z, Yang Y, Wang Z, Chang S, Yang J, Huang TS (2015) Learning super-resolution jointly from external and internal examples. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 24(11):4359–4371
35. Wang Z, Liu D, Yang J, Han W, Huang T (2015) Deep networks for image super-resolution with sparse prior. 2015 I.E. International Conference on Computer Vision (ICCV). Santiago, pp 370–378
36. Xu J, Qi C, Chang Z (2015) Coupled K-SVD dictionary training for super-resolution. *IEEE international conference on image processing IEEE*, pp 3910–3914
37. Yang J et al (2008) Image super-resolution as sparse representation of raw image patches. *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE Conference on IEEE*, pp 1–8

38. Yang J et al (2010) Image super-resolution via sparse representation. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 19(11):2861–2873
39. Yang S et al (2011) Novel super resolution restoration of remote sensing images based on compressive sensing and example patches-aided dictionary learning. *International workshop on multi-platform/multi-sensor remote sensing and mapping IEEE*, pp 1–6
40. Yang S, Liu Z, Wang M, Sun F, Jiao L (2011) Multitask dictionary learning and sparse representation based single-image super-resolution reconstruction. *Neurocomputing* 74(17):3193–3203
41. Yang J et al (2012) Coupled dictionary training for image super-resolution. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 21(8):3467–3478
42. Yang S, Wang M, Chen Y, Sun Y (2012) Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 21(9):4016–4028
43. Yang X, Zhant S, Hu C, Liang Z, Xie D (2016) Super-resolution of medical image using representation learning. 2016 8th International Conference on Wireless Communications & Signal Processing (WCSP). Yangzhou, pp 1–6
44. Yang W et al (2016) Consistent coding scheme for single-image super-resolution via independent dictionaries. *IEEE Trans Multimedia* 18(3):1–1
45. Yuan T et al (2014) Image super-resolution via kernel regression of sparse coefficients. *IEEE international conference on acoustics, speech and signal processing IEEE*, pp 5794–5798
46. Zeyde R, Elad M, Protter M (2010) On single image scale-up using sparse-representations. *International conference on curves and surfaces Springer-Verlag*, pp 711–730
47. Zhang D, Wu X (2006) An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Trans Image Process* 15(8):2226–2238
48. Zhang K et al (2013) Image super-resolution via non-local steering kernel regression regularization. *IEEE international conference on image processing IEEE*, pp 943–946
49. Zhang K et al (2015) Learning multiple linear mappings for efficient single image super-resolution. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 24(3):846–861
50. Zhang Y et al (2015) CCR: clustering and collaborative representation for fast single image super-resolution. *IEEE Trans Multimedia* 18(3):1–1



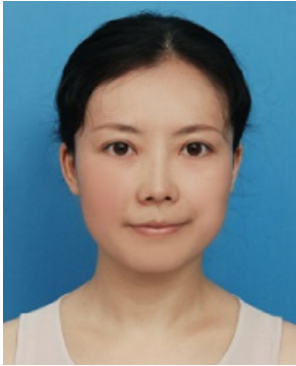
Ronggui Wang received the M.S. degree in mathematics from Anhui University, China, in 1997, and the Ph.D. degree in computer science from Hefei University of Technology, Hefei, China, in 2005. He is currently a Professor with School of Computer and Information, Hefei University of Technology. His research interests include digital image processing, artificial intelligence and data mining.



Qinghui Wang received the B.S. degree from Yantai University, Yantai, China, in 2015. He is currently pursuing the M.S. degree at School of Computer and Information, Hefei University of Technology. His research interests include digital image processing, artificial intelligence and machine learning.



Juan Yang received the B.S. and M.S. degrees in mathematics from Hefei University of Technology, Hefei, China, in 2004 and 2008, respectively. She received the Ph.D. degree with school of Computer and Information, Hefei University of Technology. She is currently a lecturer with school of Computer and Information, Hefei University of Technology. Her research interests include image processing and intelligent visual surveillance.



Lixia Xue received the B.S. degrees in geography education and M.S. degrees in geographic information system from Chongqing Normal University, Hefei, China, in 1999 and 2002, respectively. She received the Ph.D. degree in cartography and geographic information engineering from Southwest Jiao Tong University. She is currently an associate professor with school of Computer and Information, Hefei University of Technology. Her research interests include image segmentation.



Min Hu received the M.S. degree in industrial automation from Hefei University of Technology, China, in 1994, and the Ph.D. degree in computer science from Hefei University of Technology, Hefei, China, in 2004. He is currently a Professor with School of Computer and Information, Hefei University of Technology. Her research interests include digital image processing, artificial intelligence and data mining.