Chapter 6
Sparse Representation for Image Super-Resolution

Xian-Hua Han and Yen-Wei Chen

Abstract. This chapter concentrates the problem of recovery a high-resolution (HR) image from a single low-resolution input image. Recent research proposed to deal with the image super-resolution problem with sparse coding, which is based on the well reconstruction of any local image patch by a sparse linear combination of an appropriately chosen over-complete dictionary. Therein the chosen LR (Low-resolution) and HR (High-resolution) dictionaries have to be exactly corresponding for well reconstructing the local image patterns. However, the conventional sparse coding based image super-resolution usually achieves a global dictionary $\mathbf{D}$=[$\mathbf{D}_l$; $\mathbf{D}_h$] by jointly training the concatenated LR and HR local image patches, and then reconstruct the LR and HR image as a linear combination of the separated $\mathbf{D}_l$ and $\mathbf{D}_h$. This strategy only can achieve the global minimum reconstructing error of LR and HR local patches, and usually cannot obtain the exactly corresponding LR and HR dictionaries. In addition, the accurate coefficients for reconstructing the HR image patch using HR dictionary $\mathbf{D}_h$ are also unable to be estimated using only the LR input and the LR dictionary $\mathbf{D}_l$. Therefore, this paper proposes to firstly learn the HR dictionary $\mathbf{D}_h$ from the features of the training HR local patches, and then propagates the HR dictionary to the LR one, called as HR2LR dictionary propagation, by mathematical proving and statistical analysis. The effectiveness of the proposed HR2LR dictionary propagation in sparse coding for super-resolution is demonstrated by comparison with the conventional super-resolution approaches such as sparse coding and interpolation.

1 Introduction

Sparse signal representation[1-8] has been proven to be a greatly powerful algorithm for coding, representing, and compressing high-dimensional signals. It is well

Xian-Hua Han · Yen-Wei Chen
Ritsumeikan University, 1-1-1, NojiHigashi, Kusatus-Shi, Shiga-ken, 525-8577, Japan
e-mail: hanxhua@fc.ritsumei.ac.jp, chen@is.ritsumei.ac.jp

Y.-W. Chen and L.C. Jain (eds.), *Subspace Methods for Pattern Recognition in Intelligent Environment*, Studies in Computational Intelligence 552,
DOI: 10.1007/978-3-642-54851-2_6,

known that the important properties of signals such as audio and images have naturally sparse representations with respect to fixed basis (i.e., Fourier, Wavelet), or concatenations of such basis. However, the Fourier, Wavelet basis etc. are mathematically fixed, and universal to any signal, and then cannot be adaptive to the processed signal. Therefore, researches on adaptively learning basis from the processed signals are actively taken on since 1990's. The most popular strategies for achieving adaptive basis to represent data mainly include Principle Component Analysis (PCA)[9-10], Independent Component Analysis (ICA)[11-14] and so on. PCA is a mathematical procedure that learns an orthogonal transformation from the proposed signal to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. ICA [14-17] is a method to find a linear nonorthogonal coordinate system in any multivariate data. The directions of the axes in this ICA coordinate system are determined by not only the second but also higher order statistics of the original data, unlike the principle component analysis (PCA), which considers only the second order statistics and can only deal with the variables that have Gaussian distributions. In computer vision, it is more preferable to extract the source signals produced by independent causes or obtained from different sensors; such signals are easily solved using ICA. These two classical adaptive base-learning strategies (PCA and ICA based) usually only produce non-overcomplete (the number of basis equals or is less than the dimension of the processed signal) basis, and then require to use all the learned basis for well representing the observed signal. In the other hand, understand processes in retina and primary visual cortex (V1) of human being [18] has been elucidated that early visual processes compress input into a more efficient form by activating only a few receptive fields in millions, which in mathematical theory can transfer this mechanism into sparse coding by learning an over-complete basis and only using a few of basis (sparsity) to represent the observed signal.

Thanks to the success of sparse coding strategy on representing, compressing high-dimensional signal, it is widely applied to pattern recognition, computer vision, image representation and so on, and has been proven its powerful advantage over conventional adaptive basis learning approaches such as PCA and ICA. Given only unlabeled input data in sparse coding, it learns basis functions that capture higher-level features in the data. When sparse coding is applied for natural image representation, the learned basis resemble the receptive fields of neurons in the visual cortex [1, 2]; in addition, sparse coding can also produce localized basis when applied to other natural stimuli such as speech and video [3, 4]. Different to the conventional unsupervised learning techniques such as PCA, sparse coding can learn overcomplete basis sets, in which the number of basis is larger than the dimensionality of the feature space. In order to learn the adaptive basis functions and achieve the sparse coefficients from the observed data, we use the popular strategy: the K-SVD algorithm [6, 19], a generalized algorithm from the K-Means clustering process. K-SVD is an iterative method that alternates between sparse coefficient calculation (sparse coding) of the observed signal based on the current dictionary, and a process of updating the dictionary atoms to better represent the data. The update of the dictionary columns is combined with an update of the sparse representations, thereby accelerating convergence. Furthermore, the K-SVD algorithm is flexible and can work with any pursuit method (e.g., basis pursuit, matching pursuit) for achieving adaptive basis and sparse coefficients. In this chapter, we use K-SVD and orthogonal matching pursuit (OMP) [20-23]- a smart improved version of matching pursuit [24-27] for dictionary updating and sparse coefficient calculation, and then apply the sparse representation for image super-resolution.

Super-Resolution (SR) is to generate a high resolution image from one or more low resolution input images. The super-resolution techniques are recently becoming a hot research topic due to many demanding applications such as biometric identification [27-28], medical imaging [29-30], remote sensing [31-32], etc.. There are mainly two types of super-resolution frameworks: the multiple-image super-resolution, which has several available low-resolution images with sub-pixel translation and rotation; and the single-image super-resolution, which has only one LR image. In this chapter, we focus on image super-resolution for a single image using the learning-based method, which can recover the lost information in LR images by exploring the co-occurrence prior between lots of available existing LR and HR image patches. The basic idea of learning-based super-resolution is to deduce the lost information by learning from training samples, which comprise HR and LR image pairs. In [33], Freeman et al. proposed an example-based super-resolution method to infer the HR images by the corresponding relationship of the prepared training HR and LR images pair, whose LR one is most similar to the input LR one. Stephenson extended this approach to predict the HR image from the LR one using Markov Random Field (MRF) solved by belief propagation [34]. However, the above methods typically require a huge amount of HR and LR training patch pair as prepared database, which makes ineffectiveness and un-efficiency for generating a HR image from the LR one. In [35], Locally Linear Embedding (LLE) [36] as a famous manifold learning was adopted to reconstruct the HR image patch as a linear combination of the training HR ones by mapping the local geometry of the LR space to the HR one, assuming similarity between two manifolds in the HR and LR patch spaces. With this strategy, more patch patterns can be represented using a moderate amount of training database, but usually results in blurring effect using the linear combination of the nearest K raw neighborhood patches due to the large variation of raw image patches. Then, Yang etc. [37-38] proposed to learn a structural dictionary using sparse coding, which can well reconstruct any image patch as a linear combination of several similar structural atoms in the learned dictionary. However, the conventional sparse coding based image super-resolution usually achieves a global dictionary $\mathbf{D}$=[$\mathbf{D}_l$; $\mathbf{D}_h$] by jointly training the concatenated LR and HR local image patches, and then reconstruct the LR and HR image as a linear combination of the separated $\mathbf{D}_l$ and $\mathbf{D}_h$. This strategy only can achieve the global minimum reconstructing error of LR and HR local patches, and usually cannot obtain the exactly corresponding LR and HR dictionaries. In addition, the accurate coefficients for reconstructing the HR image patch using HR dictionary $\mathbf{D}_h$ are also unable to be estimated using only the LR input and the LR dictionary $\mathbf{D}_l$. Therefore, this chapter proposes to firstly learn the HR dictionary $\mathbf{D}_h$ from the features of the training HR local patches, and then propagates the HR dictionary to the LR one, called as

HR2LR dictionary propagation, by mathematical proving and statistical analysis. The effectiveness of the proposed HR2LR dictionary propagation in sparse coding for super-resolution is demonstrated by comparison with the conventional super-resolution approaches such as sparse coding and LLE. Furthermore, we validate that the proposed algorithm is robust to noisy for generating the HR image from a noisy LR image.

The remaining parts of this chapter are organized as follows. We introduce the basic sparse coding for image representation in Sec. 2, and give a detail descriptors of the SC-based image super-resolution in Sec. 3. Sec. 4 describes the used LR and HR features for learning procedure, and explores the relationship between the used LR and HR features. The proposed HR2LR dictionary propagation strategy in sparse coding is given in Sec. 5, and experimental results are shown in Sec. 6. Finally, we conclude and summarize in Sec. 7.

2 Sparse Coding

In statistical analysis of image representation, recent works [39-40] show that any image local patch can be represented by a sparse linear combination of the atoms in an over-complete dictionary. Assuming $\mathbf{D} \in \Re^{n\times K}$ be an over-complete dictionary of K prototype atoms by statistical learning from some reshaped image patches, a reshaped vector $\mathbf{x}$ from one image patch can be written as $\mathbf{x} = \mathbf{D}\alpha_\mathbf{0}$, where $\alpha_0 \in \Re^K$ is a vector with very few ($\ll K$) nonzero entries.

Problem Formulation: Suppose that there are N data samples $\{\mathbf{y}_i \in \Re^n : i = 1, 2, \cdots, N\}$ of dimension n, and the collection of these N samples forms an n-by-N data matrix $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_N)$ with each column as one sample vector. The goal is to construct a representative dictionary for $\mathbf{Y}$ in the form of an n-by-K matrix $\mathbf{D} = (\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_K)$, which consists of K (usually $K \ll N$ and $K > n$) key features $\{\mathbf{d}_i \in \Re^n : i = 1, 2, \cdots, K\}$ extracted from $\mathbf{Y}$. In the dictionary context, $\mathbf{d}_i$ is also called an atom that represents one prototype feature for well-representing any input data. This dictionary $\mathbf{D}$ needs to be trained from Y, and should be capable to sparsely represent all the samples in Y. In other words, we want to find a dictionary $\mathbf{D}$ and corresponding coefficient matrix $\mathbf{A} = (\alpha_1, \alpha_2, \cdots, \alpha_N) \in \Re^{K\times N}$ such that $\mathbf{y}_i = \mathbf{D}\alpha_i$ and $\|\alpha_i\|_0 \ll K$ for all $i = 1, 2, \cdots, N$. This can be intuitively formulated as the following minimization problem:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{y}_i - \mathbf{D}\alpha_i\|_2 \quad \mathbf{s.t.} \quad \|\alpha_i\|_0 < T \tag{1}$$

where T is the predefined threshold which controls the sparseness of the representation and $\| \bullet \|_0$ denotes the l_0 norm which counts the number of non-zero element in the vector. The equation can also alternately be formulated as:

$$\min_{\mathbf{D},\mathbf{A}} \sum_{i=1}^{n} \|\alpha_i\|_0 \quad \mathbf{s.t.} \quad \|\mathbf{y}_i - \mathbf{D}\alpha_i\|_2 \le \varepsilon \tag{2}$$

where $\varepsilon > 0$ is the predefined tolerance of representation error. The solution $(\mathbf{D}, \mathbf{X})$ of Eq. 3 yields a dictionary $\mathbf{D}$ which extracts the representative features $\{\mathbf{d}_i \in \Re^n : i = 1, 2, \cdots, K\}$ from samples in $\mathbf{Y}$ and a coefficient matrix $\mathbf{A}$ with each column α_i representing the similarity between the sample $\mathbf{y}_i$ and the dictionary atoms in $\mathbf{D}$.

Since the optimization problem in Eq. 2 is NP-hard in general, recent results suggest that several algorithms are able to be used for well approximating the solutions of Eq. 3 [39-40]. In this chapter, we use the recently developed K-SVD algorithm , which has proved to be very robust to solve Eq. 2, by iterating exact K times of Singular Value Decomposition (SVD). With an initial dictionary, K-SVD algorithm solves Eq. 2 by alternating the following two steps: the minimization with respect to $\mathbf{A}$ with the fixed $\mathbf{D}$, and atoms updating in $\mathbf{D}$ using the current $\mathbf{A}$. The formulate of the first step is same to Eq. 2 with the fixed $\mathbf{D}$, called the "sparse coding", which can be approximated by the orthogonal matching pursuit (OMP) [41]. in the following subsection, we will introduce how to calculate sparse coefficient using OMP strategy with initially selected dictionary, and update the dictionary $\mathbf{D}$ using K-SVD with the calculated sparse coefficients in the previous step.

2.1 Orthogonal Matching Pursuit

OMP is an extended orthogonal version of matching pursuit (MP), which is a type of numerical technique which involves finding the "best matching" projections of multidimensional data onto an over-complete dictionary $\mathbf{D}$. The OMP algorithm attempts to achieve the projected coefficients of the selected best basis vectors (atoms) iteratively to minimize the representation error, where the main difference from MP is that after every step, all the coefficients extracted so far are updated, by computing the orthogonal projection of the signal onto the set of selected atoms. Let $\mathbf{y}$ denotes an observed signal, and $\mathbf{D}$ denotes the fixed dictionary, the OMP algorithm attempts to find the sparse code vector α in four steps:

Step 1. Initialize the residual $\mathbf{r}_0 = \mathbf{y}$, and initialize the selected dictionary $\mathbf{D'} = \phi$ and the corresponding coefficients $\alpha_0(\mathbf{D'}) = \phi$. Let iteration counter $i = 1$, and the dictionary candidate $\mathbf{D}_0(\mathbf{D}_i) = \mathbf{D}$, from which one best basis (atom) is needed to be selected in following.

Step 2. Project the residual vector $\mathbf{r}_i$ to the dictionary candidate $\mathbf{D}_i$, and find the atom with the maximum projection value:

$$\mathbf{d} \Leftarrow \max_{\mathbf{k}} \mathbf{D}_i \mathbf{r}_i \tag{3}$$

Delete $\mathbf{d}$ from the dictionary candidate $\mathbf{D}_i$, and add it to the selected dictionary $\mathbf{D'} = [\mathbf{D'}, d]$.

Step 3. Update the coefficients $\alpha_i \Leftarrow \mathbf{D}_i^{\dagger} \mathbf{y}$ using the following equation:

$$\alpha_i(\mathbf{D'}) = \min_{\alpha_i} \|\mathbf{y} - \mathbf{D}_i \alpha_i\|^2 \tag{4}$$

Step 4. Update the residual $\mathbf{r}_i = \mathbf{y} - \mathbf{D}_i \alpha_i$.

The stop rule for OMP algorithm can be tuned to solve for either of the problem defined in Eq. 1, which iterates the step 2∼4 T times and Eq. 2, which would quit the iteration when $\|\mathbf{r}_i\|_2^2 < \epsilon$.

2.2 K-SVD Algorithm

As introduced in the above section, the sparse representation problem can be formulated by either Eq. 1 or Eq. 2. Let's assume the sparse representation problem formulated as Eq. 1, and the goal is to train the adaptive dictionary $\mathbf{D} \in R^{n*K}$ and the corresponding sparse coefficients $\alpha \in R^{K*N}$ from the observed dataset $\mathbf{Y} \in R^{n*N}$, where n is the dimension of the observed signal, N is the sample number, and K is the number of atoms or the dimension of the output sparse vector with $K >> n$. We introduce the K-SVD algorithm for extracting the adaptive dictionary $\mathbf{D}$, which is flexible and works in conjunction with any pursuit algorithm. The K-SVD algorithm is simply designed to be a truly direct generalization of the K-means. When forced to work with one atom per signal, it can train a dictionary for the gain-shape VQ. When forced to have a unit coefficient for this atom, it exactly reproduces the K-means algorithm. We start our discussion with a description of the K-means, and then derive the K-SVD algorithm as its direct extension.

A. K-means algorithm for vector quantization

K-means is to produce a codebook including codewords (representatives), which is used to represent a wide family of observed vectors (signals) by nearest neighbor assignment [42-48]. It can lead to efficient compression or description of those observed signals as clusters in surrounding the chosen codewords. Generally, K-means can be implemented based on the expectation maximization procedure, and intuitively it can be extended to the fuzzy assignment using similarity between an input signal and the codeword or normalized similarity by the covariance matrix per each cluster, where that the signal are modeled as a mixture of Gaussians [49]. Let's introduce the general K-means algorithm for learning the codebook matrix (dictionary) $\mathbf{D}$ with the codeword being in the columns from a set of observation signal $\mathbf{Y}$. In k-means, a signal $\mathbf{y}_i$ is represented as its closest codeword (under l^2-norm distance), and then its coded vector α_i include only one non-zero element with value 1, and all others zeros. Therefore, the objective function is to minimize the within-cluster sum of squares (WCSS):

$$\begin{aligned} &\arg\min_{\alpha} \sum_{i=1}^{N} \|\mathbf{y}_i - \alpha_i \mathbf{D}\| \\ &s.t. \|\alpha_i\|_{l^0} = 1, \sum_{j=1}^{K} \alpha_{ij} = 1, for\, all\, i \end{aligned} \tag{5}$$

Table 1 K-means algorithm

Goal: Find the best possible codebook to represent the observed signals $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$ using nearest neighbor $\arg\min_\alpha \sum_{i=1}^N \|\mathbf{y}_i - \alpha_i \mathbf{D}\|$ $s.t. \|\alpha_i\|_{l^0} = 1, \sum_{j=1}^K \alpha_{ij} = 1, for\, all\, i$
Initialization: initially the codebook $\mathbf{D}^{(0)} \in R^{n \times K}$ by randomly selecting K samples.   Set iteration number t=1, and repeat until convergence
Sparse coding step: Assign the observed sample to one of K codewords, and K cluster set can be achieved $(\mathbf{C}_1^{(t-1)}, \mathbf{C}_2^{(t-1)}, \cdots, \mathbf{C}_K^{(t-1)})$ where the sample $\mathbf{y}_i$ index i in k cluster $\mathbf{C}_k^{(t-1)}$ should satisfies the following condition: $\mathbf{C}_k^{(t-1)} = \{i \mid \forall_{l \neq k} \parallel \mathbf{y}_i - \mathbf{d}_k \parallel_2 < \parallel \mathbf{y}_i - \mathbf{d}_l \parallel_2\}$
Codebook update step: Update the k^{th} column $\mathbf{d}_k$ in the codebook by calculating the mean vector   in k^{th} cluster: $\mathbf{d}_k^{(t)} = \frac{1}{\lvert \mathbf{C}_k \rvert} \sum_{i \in \mathbf{C}_k} \mathbf{y}_i$
set $t = t + 1$

where $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_N]$ is the set of code vectors for the observed signal set $\mathbf{Y}$. The cardinality constraint $\|\alpha_i\|_{l^0} = 1$ means there will be only one non-zero element in each code vector $\mathbf{y}_i$, which corresponds to the most sparsity representation for the observed signal. The summation constraint $\sum_{j=1}^K \alpha_{ij} = 1$ imposes the coding weight for $\mathbf{y}_i$ is 1.

The K-means algorithm is generally implemented in an iterative strategy for designing the optimal codebook for vector quantization [39]. In each iteration there are two steps: one for assigning the observed signal to the codewords which can be called as sparse coding step, and one for updating the codebook by calculating the mean vector in each cluster, which can be considered as dictionary update. Table. 1 gives a more detailed description of these steps. The sparse coding step assumes a known codebook $\mathbf{D}^{(t-1)}$ and computes the coded coefficient αi that minimizes the representation error of Eq. (5). Similarly, the dictionary update step seeks an update of $\mathbf{D}$ by minimizing Eq. (5) with a fixed αi as known.

B. K-SVD: a generalized version of K-means

As introduced in the above, K-means algorithm quantize an observed signal to a codeword by vector quantization (VQ), which means that only one codeword is selected for representing the observed signal. The VQ strategy would results in a lot of representation error especially for the samples in the boundary areas of clusters. The sparse representation problem can be viewed as a generalization of the VQ problem (Eq. (5)), in which each observed signal is represented by a linear combination of codewords, called dictionary elements (atoms) in sparse coding (SC). Then, the coded coefficients vector is now allowed more than one nonzero entry, and these can have arbitrary values. In SC, the cost function can be relaxed as Eq. (1) or Eq. (2) mentioned in the Sec. 2.

K-SVD is proposed to combine an approximation pursuit method to solve the minimization problem of Eq. (1). First, with a initialized fixed dictionary $\mathbf{D}$, a best sparse coefficient matrix is solved using a pursuit method by minimizing Eq. (1), called sparse coding step. With the calculated coefficient in sparse coding step, the second step is performed to search for a better dictionary. This step updates one column at a time, fixing all columns $\mathbf{D}$ except one, $\mathbf{d}_k$, which attempt to find the new column $\mathbf{d}_k$ and the new values for its coefficients that best reduce the mean square error (MSE). The process of updating only one column of at a time can lead to a straightforward solution based on the singular value decomposition (SVD), and allowing a change in the coefficient values while updating the dictionary columns accelerates convergence, since the subsequent column updates will be based on more relevant coefficients. Next we will give the detail description of K-SVD algorithm for dictionary update. Assuming we have already extract the sparse coefficients in an iteration step with an fixed dictionary in the preview step, let's update only one column $\mathbf{d}_k$ in the dictionary and the coefficients that correspond to it: the k^{th} row in α, denoted as α_R^k (not the k_{th} column vector α_k in α). Then the objective function can be rewritten as:

$$\begin{aligned} \|\mathbf{Y} - \mathbf{D}\alpha\|^2 &= \|\mathbf{Y} - \sum_{i=1}^{K} \mathbf{d}_i \alpha_R^i\|^2 \\ &= \|\mathbf{Y} - \sum_{i \neq k} \mathbf{d}_i \alpha_R^i - \mathbf{d}_k \alpha_R^k\|^2 \\ &= \|\mathbf{E}_k - \mathbf{d}_k \alpha_R^k\|^2 \end{aligned} \tag{6}$$

The above equation separates the error term into two parts: error when the atom $\mathbf{d}_k$ is not taken into account, and the error reduction due to its induction for representation. This also decompose the matrix multiplication $\mathbf{D}\alpha$ to the sum of K rank-1 matrices, among which $K-1$ terms are assumed fixed, and the k^{th} one if left for updating. Then the problem of minimizing the total error thus boils down to finding a rank-1 matrix which best approximates the error matrix $\mathbf{E}_k$. Estimation of such a matrix could simply be done by performing a singular value decomposition on $\mathbf{E}_k$ and using the largest singular value and its corresponding vector for this task. However, such a step will lead to an update of the coefficients: the row vector α_R^k being very likely to be filled, which would not be sparse any more. An intuitive remedy of this problem is to form the matrix $\mathbf{E}_k$ as the reconstruction error resembles, denoted as $\mathbf{E}_{+k}$, of the observed signals which use the k^{th} atom of the dictionary for reconstruction, since the reconstruction errors of the other samples do not any change due to deleting the atom $\mathbf{d}_k$. Therefore, in order to achieve the updated atom $\mathbf{d}_k$ and the sparse coefficient, SVD decomposition of $\mathbf{E}_{+k}$ can be directly conducted, where Eigenvector of the largest Eigenvalue is used for updating the k^{th} atom $\mathbf{d}_k$ with only updating the coefficients which used the k^{th} atom so far. To implement, we first identify all the observed signals that use the k^{th} atom of the dictionary for reconstruction. Then the total error term of Eq. 6 can be split into two terms, where one term is the resulted error of representation of those signals due to the $\mathbf{d}_k$ atom

Table 2 K-SVD algorithm

Goal: Find the best dictionary to represent the observed signals $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$ as a linear sparse combination by solving

$\arg\min_{\alpha} \sum_{i=1}^N \|\mathbf{y}_i - \alpha_i \mathbf{D}\|$

$s.t. \|\alpha_i\|_0 < T, for\, all\, i$

Initialization: initially the dictionary $\mathbf{D}^{(0)} \in R^{n\times K}$ with l^2 normalized column.

Set iteration number t=1. Repeat until convergence.

Sparse coding step: Using any pursuit method (such the OMP algorithm) to calculate the sparse vector α_i for each sample $\mathbf{y}_i$, by solving the following Equation with the fixed dictionary $\mathbf{D}$

$\arg\min_{\alpha_i} \sum_{i=1}^N \|\mathbf{y}_i - \alpha_i \mathbf{D}\|$

$s.t. \|\alpha_i\|_0 < T_0, \quad i = 1, 2, \cdots, N$

Dictionary update step: For each atom (each column) in Dictionary $\mathbf{D}^{(t-1)}$, update it by,

· Obtaining the index set $idx \Longleftarrow$ all non-zero indices of α_R^k or the sample indices that use the k^{th} atom

· Calculating the reconstruction error $\mathbf{E}_{+k}$ of the sample with indices idx that use the k^{th} atom, when remove the k^{th} atom

$\mathbf{E}_{+k} = \mathbf{Y}_{+k} - \sum_{i\in idx} \mathbf{d}_i \alpha_R^i - \mathbf{d}_k \alpha_R^k$

· Doing SVD decomposition on $\mathbf{E}_{+k}$, update the k^{th} atom $\mathbf{d}_k$ using the eigenvector with the largest Eigenvalue.

$\mathbf{E}_{+k} = \mathbf{U}\triangle\mathbf{V}^T$

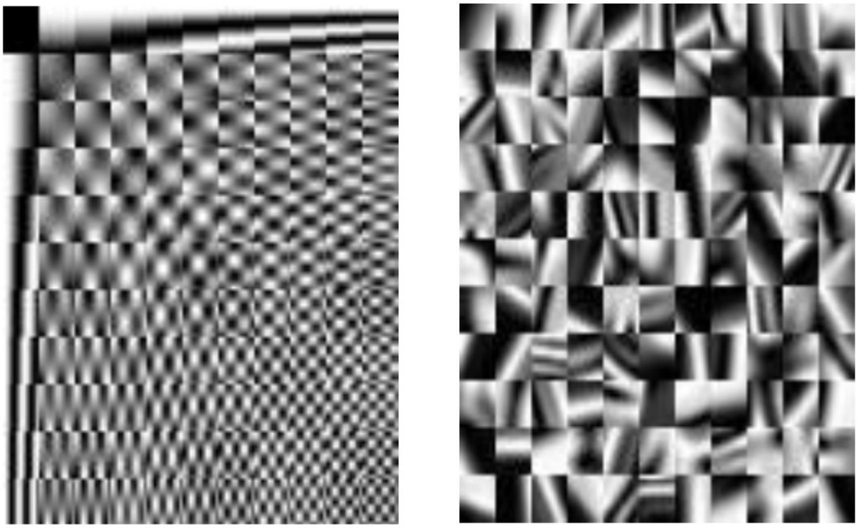
· updating the coefficient vector α_R^k using the first column of $\mathbf{V}$ multiplied with $\triangle(1,1)$.

set $t = t + 1$

being removed, and the other is the un-varied reconstruction error of the observed signals which do not use the k^{th} atom for reconstruction. The reconstruction error can be written as:

$$\begin{aligned} \|\mathbf{Y} - \mathbf{D}\alpha\|^2 &= \|\mathbf{Y} - \sum_{i=1}^{K} \mathbf{d}_i \alpha_R^i\|^2 \\ &= \|\mathbf{E}_{-k} + \mathbf{E}_{+k} - \mathbf{d}_k \alpha_R^k\|^2 \end{aligned} \tag{7}$$

where $\mathbf{E}_{-k}$ is the unchanged reconstruction error due to the deleting of the k^{th} atom, $\mathbf{E}_{+k}$ is the reconstruction error matrix with zero-vector for the observed signal without using the k^{th} atom but some reconstruction residual for the ones with using the k^{th} atom for representation. Let's firstly remove the zero-vector from the error matrix $\mathbf{E}_{+k}$, and decompose it using SVD for achieving the Eigenvector of the largest eigenvalue to update the k^{th} atom, and the corresponding vector to update the observed signals using the k^{th} atom. For all atoms, the procedure is iterated K times for updating each atom. Therefore, this procedure for dictionary updating is called 'K-SVD' to parallel the name K-means. While K-means applies computations of means to update the codebook, K-SVD obtains the updated dictionary by SVD computations, each determining one column. A detail description of the algorithm is given in Table. 2. Fig. 1 shows some 2-dimensional 8*8 DCT basis, and the learned K-SVD basis from some observed natural image patches.



(a) (b)

Fig. 1 Basis functions. (a) DCT basis; (b) the learned basis with K-SVD from the 8*8 natural image patches.

3 Sparse Coding Based Super-Resolution

The single-image super-resolution is to recover a high-resolution (HR) image $\mathbf{X}$ from a observed low-resolution image $\mathbf{Y}$, which is a blurred and downsampled version of the HR one $\mathbf{X}$:

$$\mathbf{Y} = \mathbf{LHX} \tag{8}$$

where $\mathbf{H}$ represents a blurring (smooth) filter, and $\mathbf{L}$ is the sown-sampling operator. The degradation model of the imaging procedure is shown in Fig. 2. In the learning-based super-resolution, the lost information in any test LR image can be recovered by learning using the corresponding relationship of the raw patches in the available LR and HR images. With same philosophy, given any LR image patch $\mathbf{y}$ well reconstructed by a sparse linear combination of an over-complete LR dictionary $\mathbf{D}_l$, the HR corresponding image patch $\mathbf{y}$ can also be approximated by the liner combination of corresponding HR dictionary $\mathbf{D}_h$ with the same sparse coefficients as the following:

$$\mathbf{y} = \mathbf{D}_l \alpha_0, \qquad \mathbf{x} \approx \mathbf{D}_h \alpha_0 \tag{9}$$

where α_0 is a vector with very few ($\ll K$) nonzero entries. In the conventional super-resolution using sparse coding, the image local patches are firstly reconstructed by the sparse linear combination of the pre-learned dictionary, and then remove the artifacts in the recovered global HR images formed from the local patches based on reconstruction constraints. Next, we will mainly introduce sparse representation of image local patches for super-resolution.

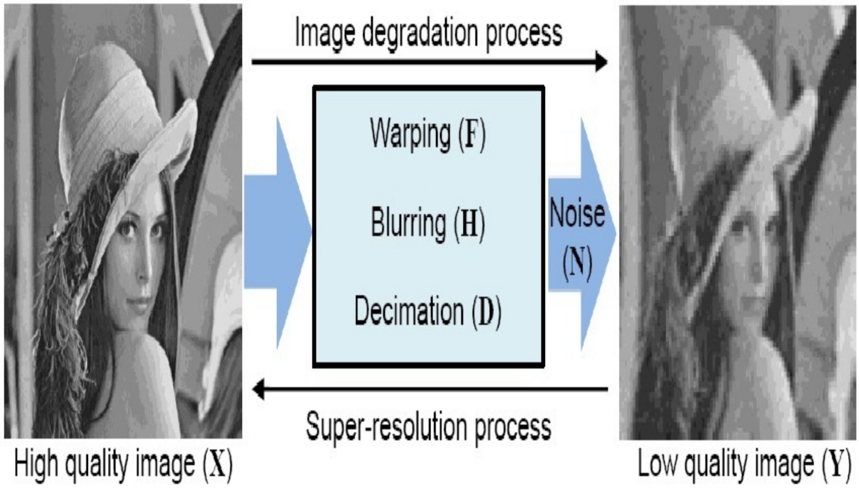


Fig. 2 The degradation model of imaging procedure

Similar to the conventional learning-based super-resolution framework which is shown in Fig. 2, the sparse coding based one also tries to infer the high-resolution patch from each low-resolution image patch of the input. In the sparse representation of the image local patches, there are two dictionaries $\mathbf{D}_h$ and $\mathbf{D}_l$, which are trained to have the similar sparse representations for each high-resolution and low-resolution image patch pair. Given any input low-resolution patch $\mathbf{y}$, we can find a sparse representation with respect to $\mathbf{D}_l$. The estimation of the corresponding high-resolution patch $\mathbf{x}$ can also be reconstructed by the sparse combination of these same coefficients but replacing $\mathbf{D}_l$ with $\mathbf{D}_h$.

For sparse coding based super-resolution, the corresponding LR and HR dictionaries need to be learned from the training LR and HR image patches $\mathbf{Y}$ and $\mathbf{X}$, respectively. [37] modifies Eq. 3 as the following optimization formulation:

$$\min_{\mathbf{D}_l,\mathbf{D}_h,\mathbf{A}} \sum_{i=1}^{n} \|\alpha_i\|_0 \quad \mathbf{s.t.} \quad \|F\mathbf{y}_i - F\mathbf{D}_l\alpha_i\|_2^2 \leq \varepsilon_1 \\ \|P\mathbf{x}_i - P\mathbf{D}_h\alpha_i\|_2^2 \leq \varepsilon_2 \tag{10}$$

where F is a linear feature extraction operator, which is to provide a perceptually meaningful constraint on how closely the coefficients α approximates the input patch $\mathbf{x}_l$. In [38], The 1-order derivative operator are used for F. Sec. 3 will explore the choice of F in our proposed HR2LR dictionary propagation. The $\mathbf{P}$ is the operator for subtracting mean intensity of all pixels from the HR image patch.

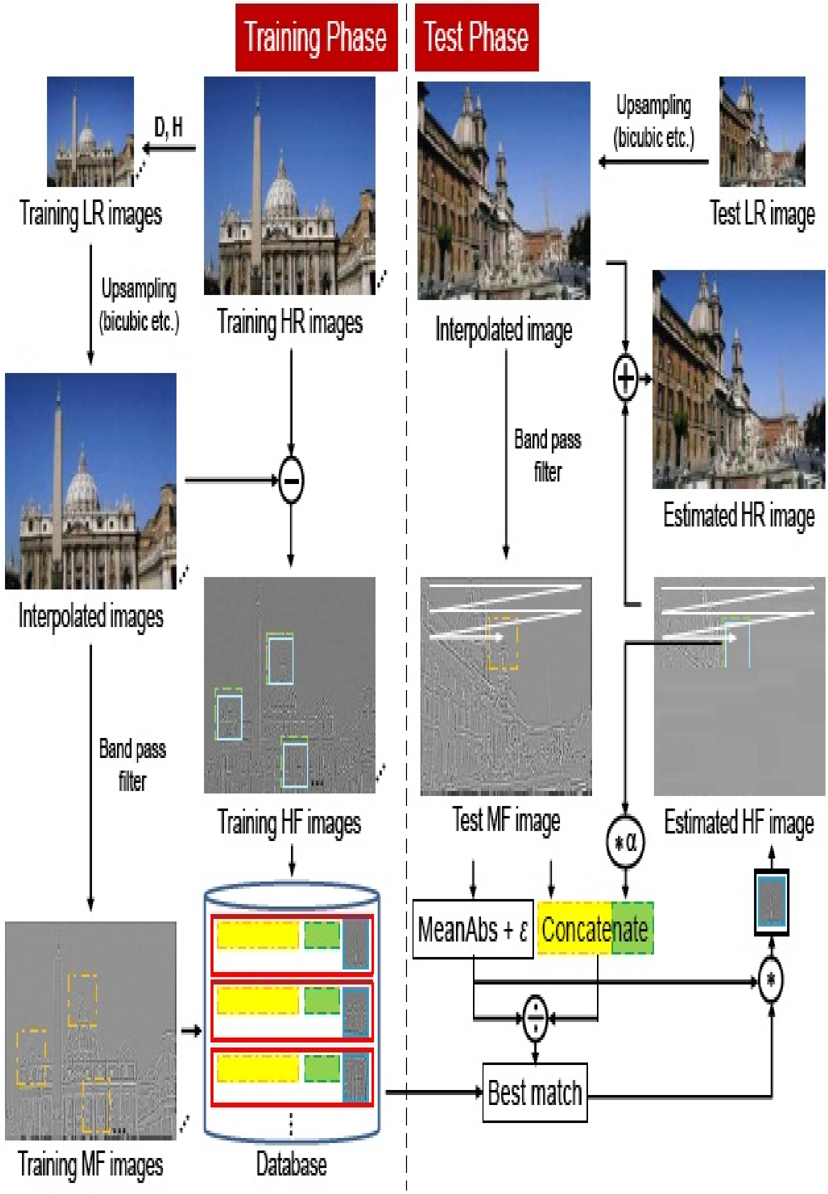


Fig. 3 The framework of learning-based Super-Resulution

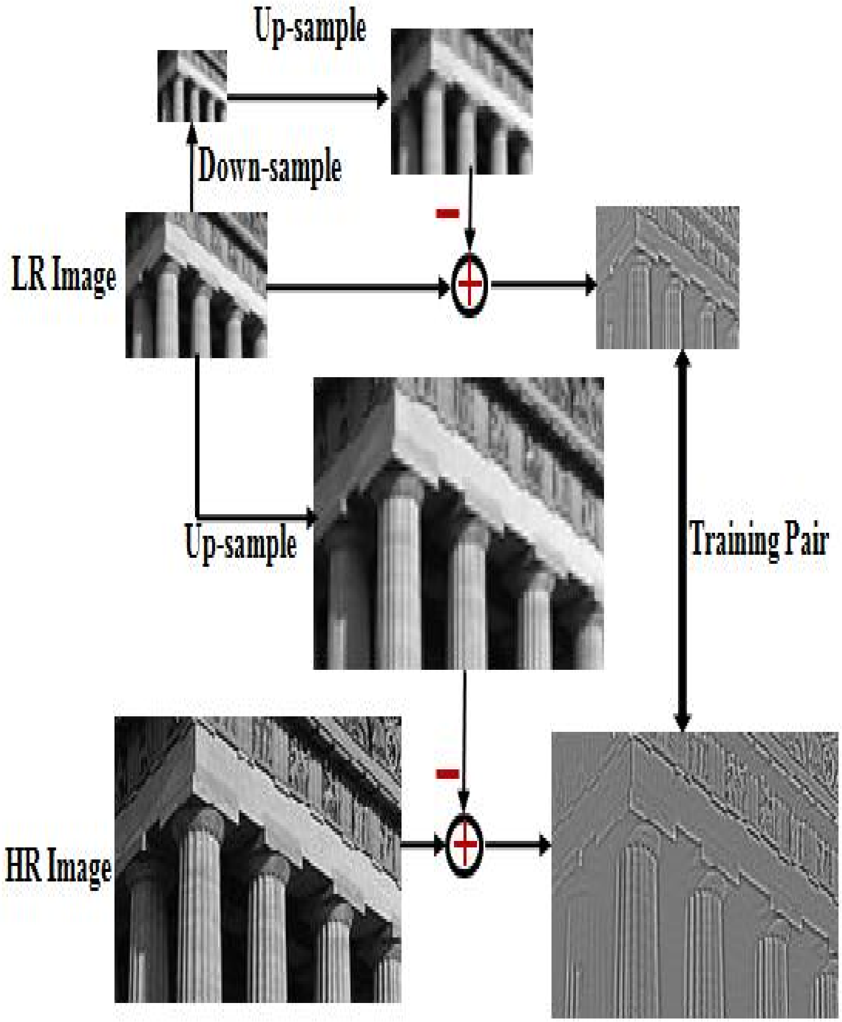


Fig. 4 The LR and HR feature extraction procedure

The constrained optimization (6) can be similarly reformulated as a jointly learning procedure for $\mathbf{D}_l$ and $\mathbf{D}_h$ [38]:

$$\min_{\bar{\mathbf{D}},\mathbf{A}} \sum_{i=1}^{n} \|\alpha_i\|_0 \quad \mathbf{s.t.} \quad \|\bar{\mathbf{y}}_i - \bar{\mathbf{D}}\alpha_i\|_2^2 \leq \varepsilon \tag{11}$$

where $\bar{\mathbf{D}} = [F\mathbf{D}_l; \beta P\mathbf{D}_h]$ and $\bar{\mathbf{y}} = [F\mathbf{y}_i; \beta P\mathbf{x}_i]$. The parameter β controls the tradeoff between reconstructing the LR and HR patches. With any input LR image patch $\mathbf{x}_t$, the sparse coefficient α_t can be achieved with the learned LR dictionary

$\mathbf{D}_l$, and then, the corresponding HR patch can be estimated with the same coefficients α_t and the learned HR dictionary $\mathbf{D}_h$. However, The above jointly learning procedure for $\mathbf{D}_l$ and $\mathbf{D}_h$ usually cannot achieve the accurate corresponding LR and HR dictionaries, and the sparse coefficients are also approximated estimation with only the available input LR feature $F\mathbf{y}_t$. Therefore, the following section investigates the corresponding LR and HR features for image patch representation, which invokes the proposed HR2LR dictionary propagation for achieving the accurate corresponding LR and HR dictionaries.

4 Analysis of the Represented Features for Local Image Patches

In Eq. 5, some features are needed to be extracted for image representation. The conventional sparse coding based super-resolution algorithm [37] uses the first order derivative as F in Eq. 5 for LR image representation, and the subtracted pixel intensity from the mean of the HR patch as P. It is obvious that the used features for LR and HR image patch have no accurate correspondence even after some prenormalization for removing scale variance [37]. As mentioned in Sec. 2, the low-resolution image $\mathbf{Y}$ is a blurred and down-sampled version of the high-resolution image $\mathbf{X}$: $\mathbf{Y} = SH\mathbf{X}$ with the blurring filter H and the down-sampling operator S. The most intuitive method for achieve the same size version of $\mathbf{X}$ from $\mathbf{Y}$ is to use up-sampling interpolation operator U: $\bar{\mathbf{X}} = U\mathbf{Y}$. Based on the interpolated version $\bar{\mathbf{X}}$ of $\mathbf{Y}$, the lost information of the high-resolution $\mathbf{X}$ can be considered as $\mathbf{X} - \bar{\mathbf{X}}$, which is the used feature for the training HR image, and at the same time, also is the estimated lost information of any LR input for achieving the HR one. The feature extraction, as linear operator P, for the HR image can be formulated as:

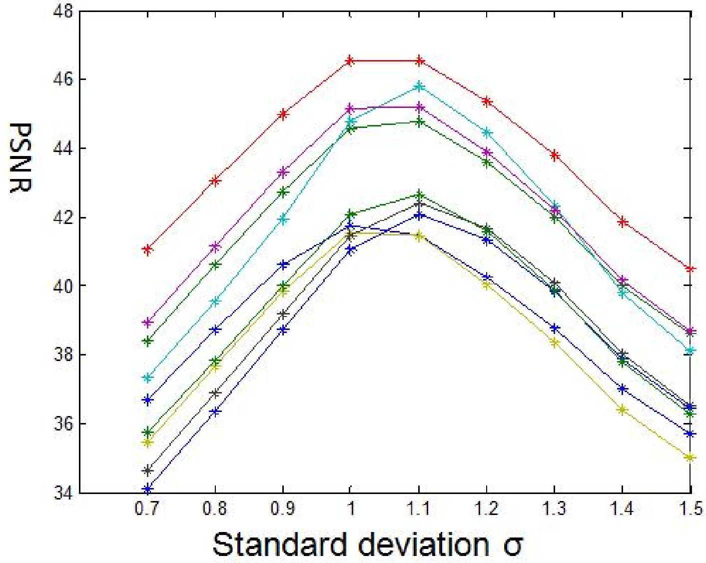
$$P\mathbf{X} = \mathbf{X} - \bar{\mathbf{X}} = \mathbf{X} - U\mathbf{Y} = \mathbf{X} - USH\mathbf{X} \tag{12}$$

In order to obtain the corresponding features of the LR image to those of the HR one, we impose the blurring and down-sampling operator on $\mathbf{Y}$: $\mathbf{Z} = SH\mathbf{Y}$, which is same on $\mathbf{X}$ to produce $\mathbf{Y}$, and then extract the LR feature by subtracting the interpolated version $\bar{\mathbf{Y}} = U\mathbf{Z}$ from the LR image $\mathbf{Y}$. Then the operator F for extracting feature from the LR image can be formulated as:

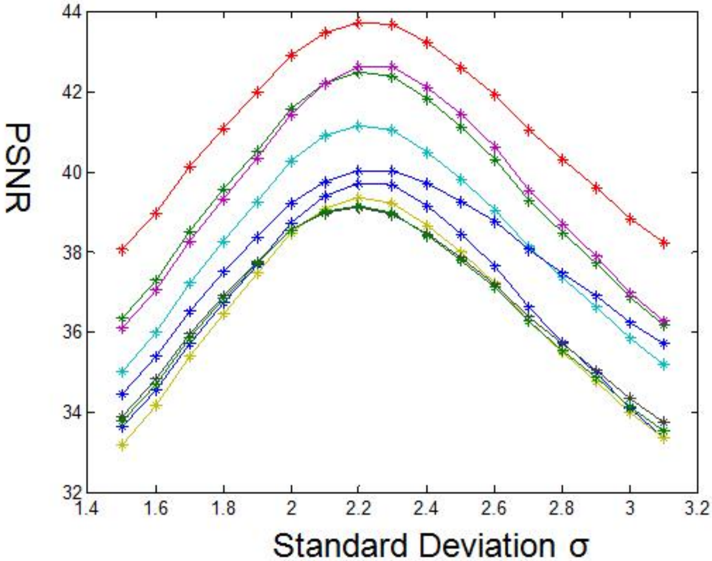
$$F\mathbf{Y} = \mathbf{Y} - \bar{\mathbf{Y}} = \mathbf{Y} - U\mathbf{Z} = \mathbf{Y} - USH\mathbf{Y} \tag{13}$$

The feature extraction procedures for the LR and HR image are shown in Fig. 3. From Eq. 7 and 8, it is obvious that the feature extractions for the LR and HR images follow the same process, prospecting corresponding relation between $F\mathbf{Y}$ and $P\mathbf{X}$. In addition, with $\mathbf{Y} = SH\mathbf{X}$ being the blurred and down-sampled version of the $\mathbf{X}$, the transformation from $P\mathbf{X}$ and $F\mathbf{Y}$ can be formulated as:

$$\begin{aligned} F\mathbf{Y} &= \mathbf{Y} - USH\mathbf{Y} = SH\mathbf{X} - USH(SH\mathbf{X}) \\ &= SH\{\mathbf{X} - USH\mathbf{X}\} = SH\{P\mathbf{X}\} \end{aligned} \tag{14}$$

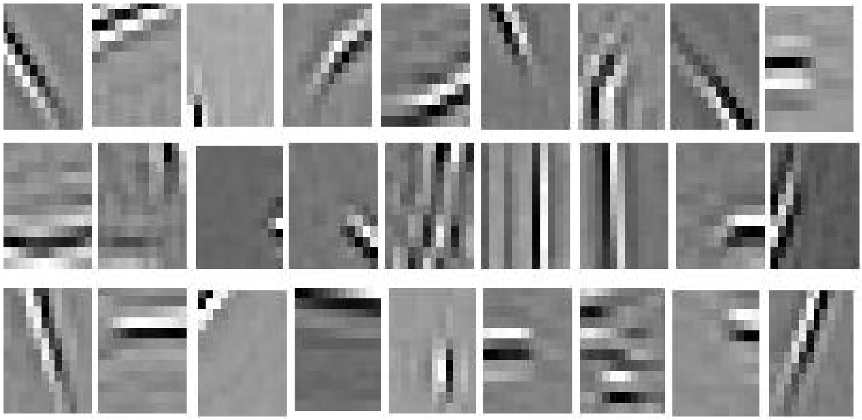


(a)The PSNR values between the interpolation versions of the LR images and the blurred versions of the HR images with different σ (Factor=2).

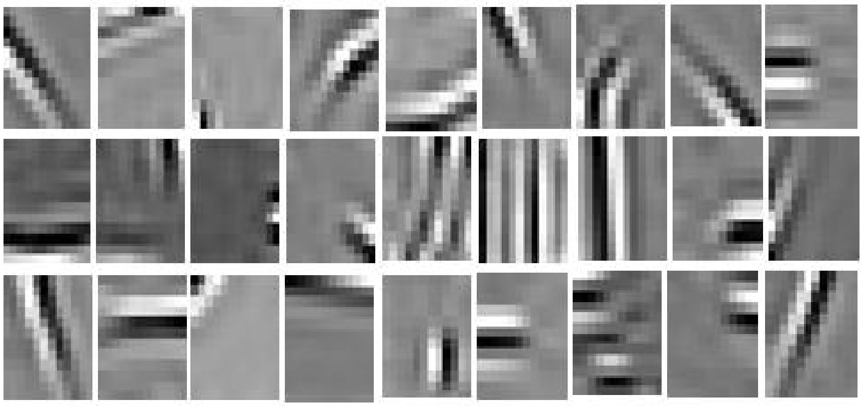


(b)The PSNR values between the interpolation versions of the LR images and the blurred versions of the HR images with different σ (Factor=4).

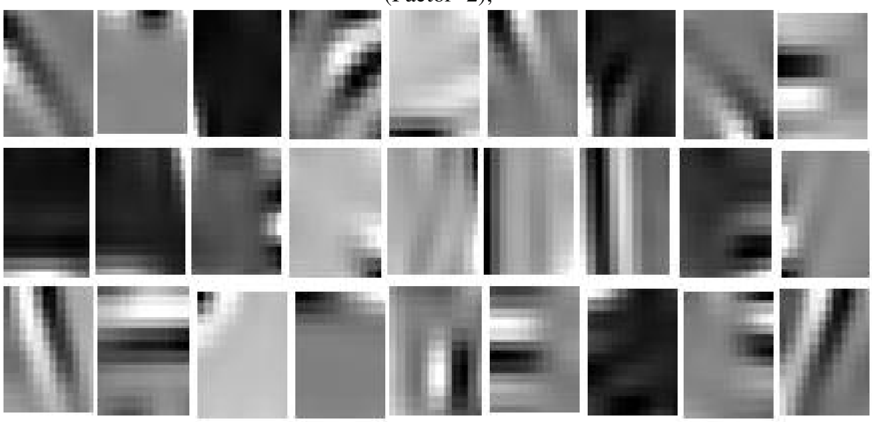
Fig. 5 The statistical analysis of the LR and HR images



(a) Example atoms of the HR dictionary;



(b)Corresponding example atoms of the propagated LR dictionary from the HR one (Factor=2);



(b)Corresponding example atoms of the propagated LR dictionary from the HR one (Factor=4);

Fig. 6 Example atoms of the LR and HR dictionaries

Fig. 7 The used example images for PSNR calculation

Therefore, the LR feature $F\mathbf{Y}$ is also a blurred and down-sampled version from the HR feature $P\mathbf{X}$. This means that the un-downsampled version of the LR features can be approximated by some suitable blurring version of the HR feature. If the HR feature is blurred by some suitable low-pass filter, the smoothed version should have high similarity with the un-downsampled version $H\{P\mathbf{X}\}$, which can be obtained by up-sampling the LR feature $F\mathbf{Y}$ using interpolation. Next, we investigate the similarity with PSNR (peak signal-to-noise ratio) between the interpolation version $\bar{\mathbf{Y}}$ of the LR images $\mathbf{Y}$ and the blurred version $H\mathbf{X}$ of the HR images $\mathbf{X}$ with low-pass filter.

We use the Gaussian kernel as the low-pass filter H with different standard deviation $\sigma = [0.7, 0.8, \cdots, 1.5]$, and utilize bilinear as the interpolation operator. The used 9 example images are shown in Fig. 4, and the PSNR values between between the interpolation version of the LR images and the blurred version of the HR image with different σ are shown in Fig. 5. It can be seen that the PSNR values of all images are larger than 40 with the largest one: more than 47 with about 1 or 1.1 σ value for expanding factor 2 (Fig. 5(a)), which means enough similarity and be difficult for distinguish from subjective assessment; larger than 37 with the largest one: more than 43 with about 2.1 or 2.3 σ value for expanding factor 4 (Fig. 5(b)). Based the statistical analysis, we will introduce the proposed HR2LR dictionary propagation approach of sparse coding for super-resolution.

5 HR2LR Dictionary Propagation of SC

SC based image super-resolution requires two corresponding dictionaries $\mathbf{D}_l$ and $\mathbf{D}_h$ to be pre-learned for reconstructing the LR and HR image patches using the sparse combination of their atoms. A given feature of a HR image patch $\mathbf{x}_i$ is reconstructed as a sparse combination of atoms taken from the HR dictionary $\mathbf{D}_h$ as follows:

$$\mathbf{x}_i \approx \sum_{j=1}^{K} \alpha_{ij} \mathbf{D}_h^j, \quad \mathbf{s.t.} \|\alpha_i\|_0 \leq L \tag{15}$$

where L is a positive integer, meaning that the non-zero numbers of α_i are less than L. As analyzed in Section 3, the LR feature is a down-sampled version of the corresponding HR feature, formulated as

$$\begin{aligned} \mathbf{y}_i = SH\mathbf{x}_i \approx SH\sum_{j=1}^{K} \alpha_{ij} \mathbf{D}_h^j = \sum_{j=1}^{K} \alpha_{ij} [SH\mathbf{D}_h^j] \\ \mathbf{s.t.} \|\alpha_i\|_0 \leq L \end{aligned} \tag{16}$$

From Eq. 11, we conclude that the accurate corresponding LR dictionary can be propagated by the mathematical transformation if the HR dictionary is available. Because the corresponding HR and LR training images are available, we can first learn the HR dictionary $\mathbf{D}_h$ using SC strategy as follows:

$$\min_{\mathbf{D}_h, \mathbf{A}} \sum_{i=1}^{n} \|\alpha_i\|_0 \quad \mathbf{s.t.} \quad \|\mathbf{x}_i - \mathbf{D}_h \alpha_i\|_2 \leq \varepsilon \tag{17}$$

With the HR dictionary $\mathbf{D}_h$ obtained, the LR dictionary can then be simply propagated using $\mathbf{D}_l = SH\mathbf{D}_h$. In real applications, because of the boundary effects in small image patches, the blurred version $\bar{\mathbf{D}}_h$ of the HR dictionary $\mathbf{D}_h$, which corresponds to the interpolated up-sampled version of the LR image, is used to obtain the sparse coefficient of any LR image input $\mathbf{y}_t$ as follows:

$$\begin{aligned} \min \|\alpha_t\|_0 \quad \mathbf{s.t.} \quad & \|\mathbf{y}_t - \mathbf{D}_l \alpha_t\|_2^2 \leq \varepsilon \\ & \|U\mathbf{y}_t - U\mathbf{D}_l \alpha_t\|_2^2 \approx \|U\mathbf{y}_t - \bar{\mathbf{D}}_h \alpha_t\|_2^2 \leq \varepsilon \end{aligned} \tag{18}$$

where U is the up-sampling operator, and $\bar{\mathbf{D}}_h$ is the blurred version of $\mathbf{D}_h$, which in turn is the approximation of the up-sampling of $\mathbf{D}_l$. With the obtained α_t value for sparse reconstruction of the LR input $\mathbf{y}_t$, the HR estimation can be reconstructed with the same α_t but by replacing $\mathbf{D}_l$ with $\mathbf{D}_h$. Figure 6 shows a learned HR dictionary and the corresponding propagated LR dictionary for the magnification factors 2 and 4.

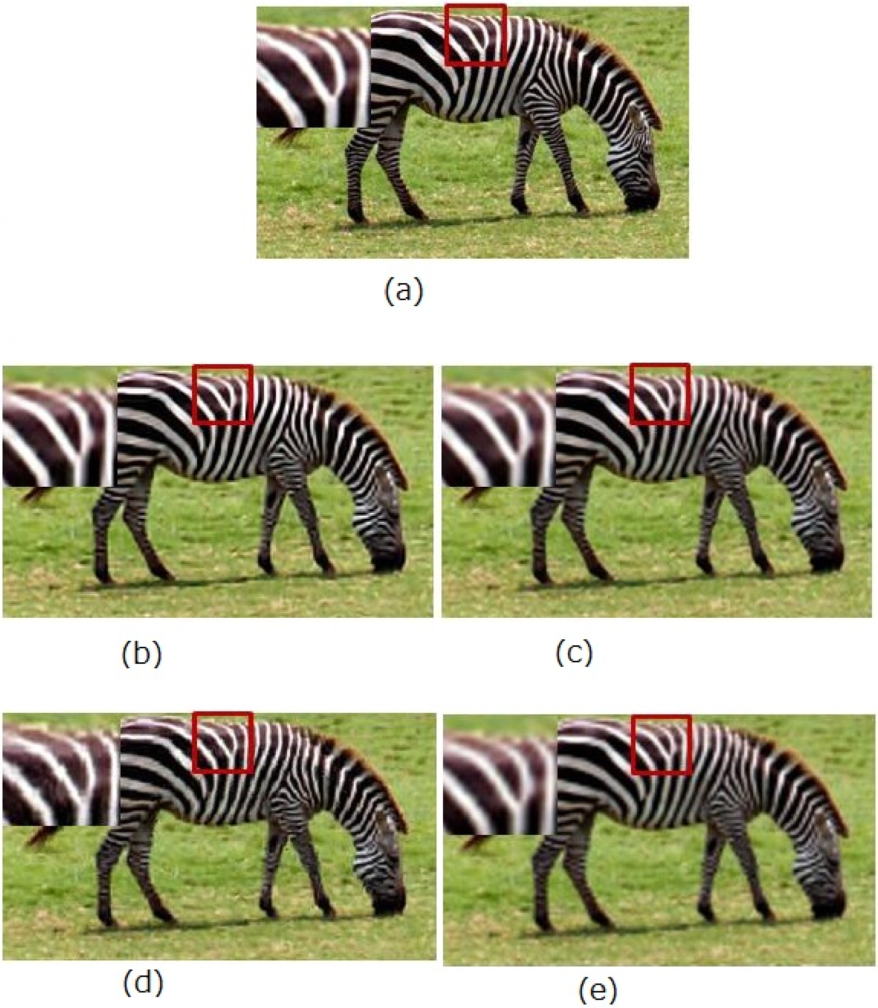


Fig. 8 Comparison of HR images of a zebra, reconstructed by different methods (magnification factor=2) with (a) the original HR image. Recovered images were obtained by (b) our proposed method, (c) the conventional SC-based method, (d) the NE-based method, and (e) the bicubic interpolation-based method.

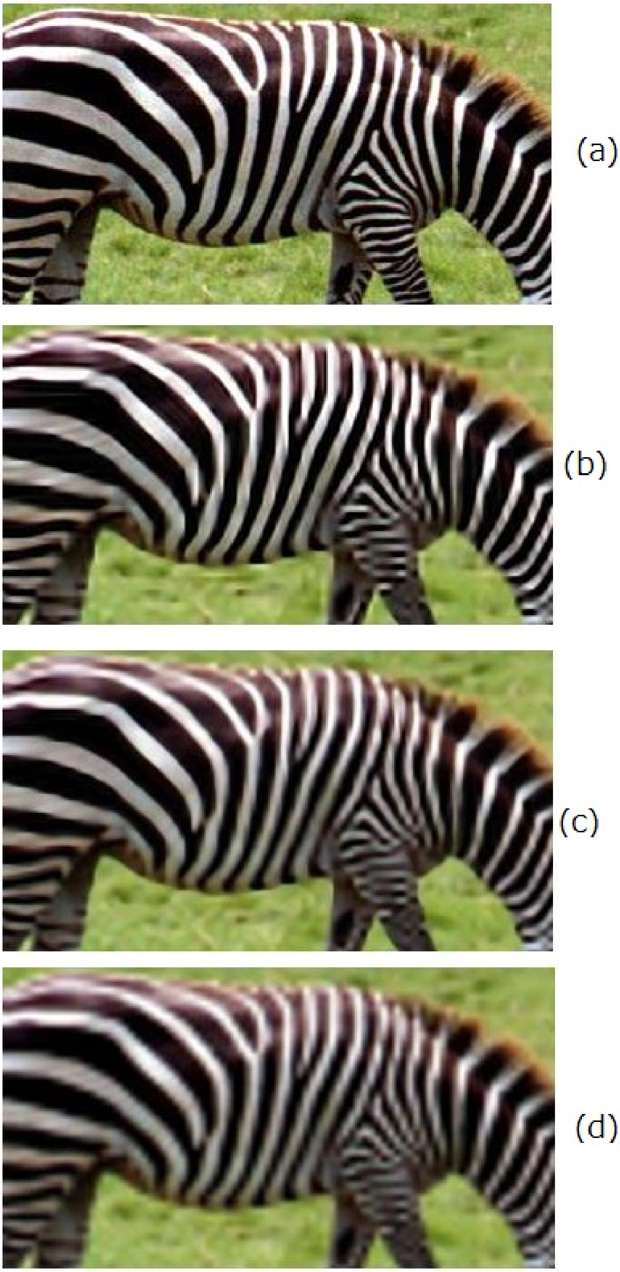


Fig. 9 Comparison of HR images of the zebra (magnification factor=4) reconstructed by different methods. (a) The Original HR zebra image and the HR recovered by (b) our proposed method (RMSE: 14.84), (c) the conventional SC-based method (RMSE: 15.40), and (d) the bicubic interpolation-based method (RMSE: 16.46).

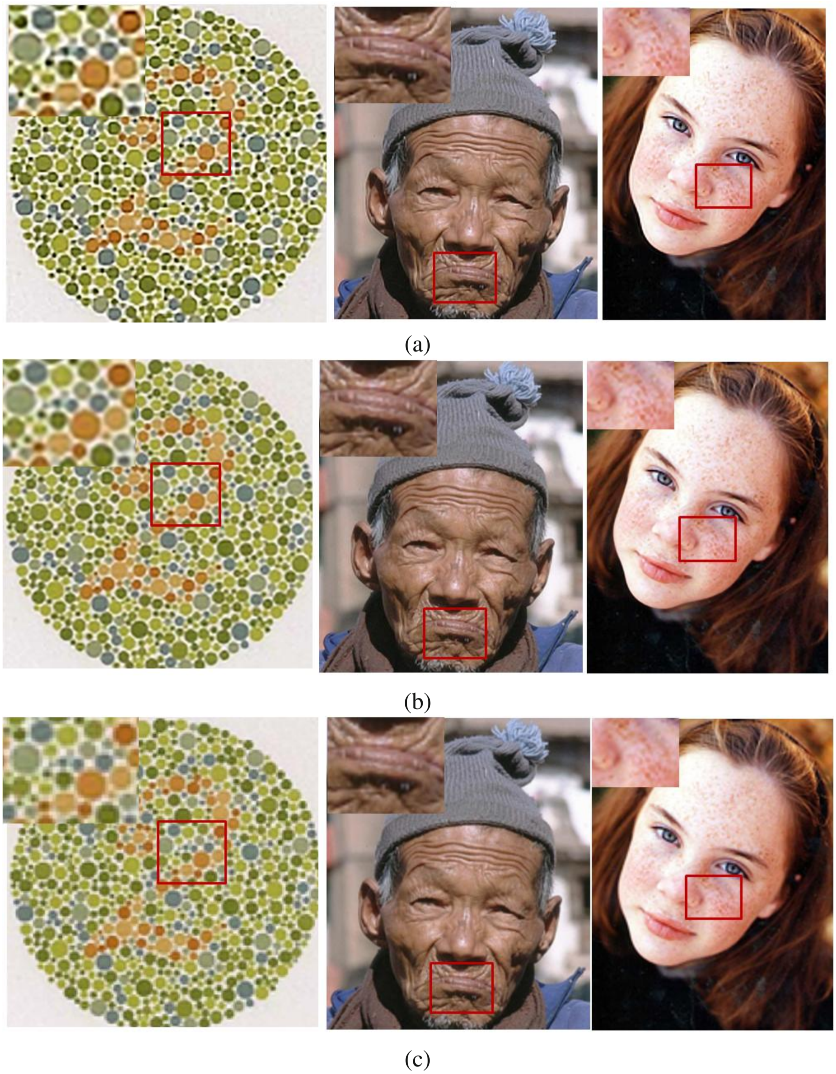


Fig. 10 Other examples of HR images (magnification factor=2 recovered by (a) our proposed method, (b) the conventional SC-based method and (c) the bicubic interpolation method

Table 3 Comparison of the RMSE and PSNR for the zebra images in Fig. 8 recovered using different SR methods

Evaluated measures	RMSE	PSNR
Our method	11.21	27.14
Conventional SC	11.71	26.76
NE-based method	15.13	23.97
Interpolation	13.58	25.47

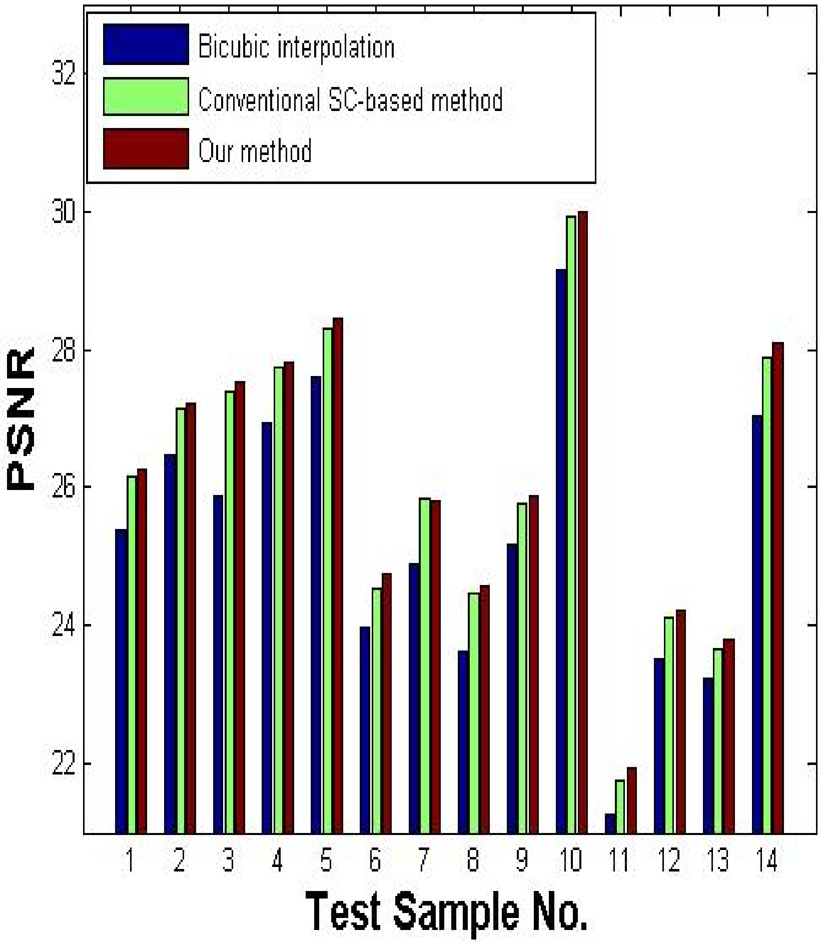


Fig. 11 The compared PSNRs of 14 test samples

6 Experiments

In our experiments, we magnified the LR input image by a factor of 2 or 4. We first interpolated the LR input to the same size. In the interpolated LR image and the corresponding HR image, we always use patches of size 12×12, with adjacent patches overlapping by 3 pixels. The features were then extracted as shown in Fig. 4. For color images, we only applied the SR strategy to the illuminance component, and the interpolated color components were used for reconstructing the HR final color image. To propagate the HR dictionary to the LR one, we used a Gaussian filter with a standard deviation $\sigma = 1.0$ for magnification factor 2, and $\sigma = 2.0$ for

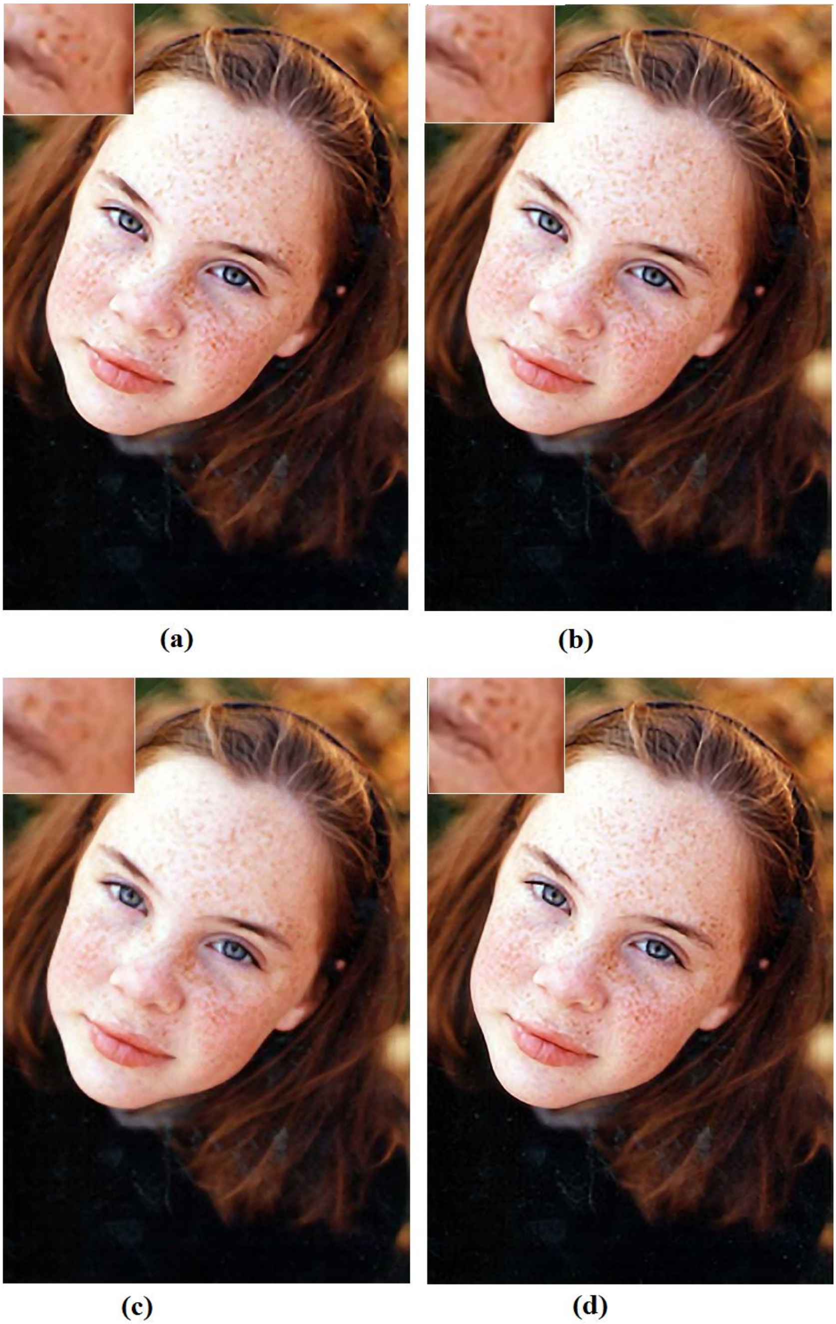


Fig. 12 The recovered HR images with other state-the-art methods. (a) our method, (b) Freedman's method [**?**], (c) Genuine Fractals (a state-of-the-art commercial product), (d) Glasner's method [**?**].

magnification factor 4, as shown in Fig. 6. Fig. 8 shows the HR images of the zebra recovered by our proposed strategy, the conventional SC-based [37], NE-based [35] and interpolation-based methods for magnification factor 2. Figure 9 shows a section of the HR images of the zebra (magnification factor 4) reconstructed by our proposed algorithm and by the conventional SC-based and interpolation-based methods. Figures 4 and 5 demonstrate that the proposed HR2LR dictionary propagation method in SC can yield much clearer HR images than yielded by the conventional SC-based, NE-based, and interpolation-based methods. We also evaluate the quantitative quality of the recovered HR images in Figs. 4 using root mean square error (RMSE) and the peak signal-to-noise ratio (PSNR) in Table 3. Figure 10 compares the reconstructed HR images derived from other LR inputs using our proposed strategy, the conventional SC-based and bicubic-interpolation-based methods. Here again, our proposed approach yields great clarity. In addition, in Fig. 11, we show the compared PSNR of the recovered HR images for other 14 test samples, which obviously validate most test images by our method can achieve better PSNR than the conventional SC-based method except for a similar PSNR for one sample. In order to validate effectiveness of the proposed strategy compared with other the state-of-the-art method [50-51], we also use the recovered HR images with expand factor 3 in Fig. 12. It is obvious that the recovered HR image is much better than the ones by Glasner's method [50], and has similar performance visually but sharper in some detail regions compared with Freedman's work [51].

7 Conclusions

This chapter introduces the sparse signal representation, and a popular implementation: K-SVD algorithm combining orthogonal matching pursuit (OMP) for learning the adaptive dictionary and achieving the sparse coefficients. OMP is an extended orthogonal version of matching pursuit (MP), which is a type of numerical technique which involves finding the "best matching" projections of multidimensional data onto an over-complete dictionary $\mathbf{D}$, and can be combined into the K-SVD strategy for achieving sparse representation and the best adaptive dictionary. K-SVD is popularly used for solving the optimization problem in sparse coding. The procedure of K-SVD mainly include two steps: first, with a initialized fixed dictionary $\mathbf{D}$, a best sparse coefficient matrix is solved using a pursuit method, called sparse coding step. With the calculated coefficient in sparse coding step is achieved, the second step is performed to search for a better dictionary. This step updates one column at a time, fixing all columns $\mathbf{D}$ in except one, $\mathbf{d}_k$, which attempt to find the new column $\mathbf{d}_k$ and the new values for its coefficients that best reduce the MSE.

Next, we apply the sparse representation for learning-based image super-resolution for recovering the high-resolution image from only single LR one. Based on the couple dictionary learning for super-resolution, we proposed a HR2LR dictionary propagation algorithm in SC for image super-resolution. Conventional SC-based image SR usually yields a global dictionary $\mathbf{D}$=[$\mathbf{D}_l$; $\mathbf{D}_h$] by jointly training the concatenated LR and HR local image patches and then reconstructing the LR and HR images as sparse combinations of atoms taken from $\mathbf{D}_l$ and $\mathbf{D}_h$. This strategy can only achieve the global minimum reconstruction error of LR and HR local patches, and cannot usually obtain exactly corresponding LR and HR dictionaries. In addition, accurate-coefficients for reconstructing the HR image patch using $\mathbf{D}_h$ cannot be estimated using only the LR input and the $\mathbf{D}_l$. This chapter proposes an algorithm called HR2LR dictionary propagation that involves learning the HR dictionary $\mathbf{D}_h$ from the features of the HR training local patches and then propagating the HR dictionary to the LR one by mathematical proofs and statistical analysis. The experimental results for image SR demonstrate that the proposed HR2LR dictionary propagation yields much clearer HR images than those obtained using conventional SR approaches such as those based on SC, NE and bicubic interpolation.

Acknowledgement. This work was supported in part by the Grant-in Aid for Scientific Research from the Japanese MEXT under the Grant No. 2430076, 24103710, 24700179 and in part by the R-GIRO Research fund from Ritsumeikan Univ.

References

1. Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381, 607–609 (1996)
2. Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision Research 37, 3311–3325 (1997)
3. Lewicki, M.S., Sejnowski, T.J.: Learning overcomplete representations. Neural Comp. 12(2) (2000)
4. Olshausen, B.A.: Sparse coding of time-varying natural images. Journal of Vision 2(7), Article 130 (2002)
5. Olshausen, B.A., Field, D.J.: Sparse coding of sensory inputs. Cur. Op. Neurobiology 14(4) (2004)
6. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing 54(11), 4311–4322 (2006)
7. Aharon, M., Elad, M.: Image denoising via sparse and re- dundant representations over learned dictionaries. IEEE Transactions on Image Processing 15(12), 3736–3745 (2006)
8. Donoho, D.L., Elad, M., Temlyakov, V.: Stable recovery of sparse overcomplete representations in the presence of noise. IEEE Transactions on Information Theory 52(1), 6–18 (2006)
9. Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics 2, 433–459 (2010)
10. Roweis, S.: M Algorithms for PCA and SPCA. In: Jordan, M.I., Kearns, M.J., Solla, S.A. (eds.) Advances in Neural Information Processing Systems. The MIT Press (1998)
11. Bell, A.J., Sejnowski, T.J.: The 'Independent Components' of natural scenes are edge filters. Vision Research 37, 3327–3338 (1997)
12. Bell, A.J., Sejnowski, T.J.: An information-maximization approach to blind separation and blind deconvolution. Neural Computation 7, 1129–1159 (1995)
13. Common, P.: Independent component analysis-a new concept? Signal Processing 36, 287–314 (1994)
14. Hyvarinen, A., Oja, E.: A fast fixed-point algorithm for indepepndent component analysis. Neural Computation 9, 1483–1492 (1997)

15. Hyvarinen, A., Oja, E., Hoyer, P.: Image Denoising by Sparse Code Shrinkage. In: Haykin, S., Kosko, B. (eds.) Intelligent Signal Processing. IEEE Press (2000)
16. Han, X.-H., Nakao, Z., Chen, Y.-W.: An ICA-Domain Shrinkage based Poisson-Noise Reduction Algorithm and Its Application to Penumbral Imaging. IEICE Trans. Inf. & Syst. E88-D(4), 750–757 (2005)
17. Han, X.-H., Chen, Y.-W., Nakao, Z.: Robust Edge Detection by Independent Component Analysis in Noisy Images. IEICE Trans. Inf. & Syst. E87-D(9), 2204–2211 (2004)
18. Sceniak, M.P., Hawken, M.J., Shapley, R.: Visual spatial characterization of macaque V1 neurons. The Journal of Neurophysiology 85(5), 1873–1887 (2001)
19. Aharon, M., Elad, M.: Image denoising via sparse and re- dundant representations over learned dictionaries. IEEE Transactions on Image Processing 15(12), 3736–3745 (2006)
20. Cai, T.T., Wang, L.: Orthogonal Matching Pursuit for Sparse Signal Recovery With Noise. IEEE Transactions on Information Theory 57(7), 4680–4688 (2011)
21. Tropp, J.A., Gilbert, A.C.: Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. IEEE Transactions on Information Theory 53(12) (December 2007)
22. Donoho, D.L., Tsaig, Y., Drori, I., Starck, J.-l.: Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit, Technique Report (2006)
23. Pati, Y.C., Rezaiifar, R., Krishnaprasad, P.S.: Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In: Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput., vol. 1 (1993)
24. Bergeaud, F., Mallat, S.: Matching pursuit of images. In: Proc. International Conference on Image Processing, vol. 1, pp. 53–56 (1995)
25. Neff, R., Zakhor, A.: Very low bit-rate video coding based on matching pursuits. IEEE Transactions on Circuits and Systems for Video Technology 7(1), 158–171 (1997)
26. Mallat, S.G., Zhang, Z.: Matching Pursuits with Time-Frequency Dictionaries. IEEE Transactions on Signal Processing, 3397–3415 (December 1993)
27. Gunturk, B., Batur, A.U., Altunbasak, Y., Hayes, M.H., Mersereau, R.M.: Eigenface-domain super-resolution for face recognition. IEEE Transaction on Image Processing 12(5), 137–147 (2003)
28. Zhang, J., Pu, J., Chen, C., Fleischer, R.: Low-resolution gait recognition. IEEE Transaction on Systems, Man, and Cybernetic–Part B: Cybernetics 40(4), 986–996 (2010)
29. Galbraith, A., Theiler, J., Thome, K., Ziolkowski, R.: Resolution enhancement of multi-look imagery for the multispectral thermal image. IEEE Transaction on Geoscience and Remote Sensing 43(9), 1964–1977 (2005)
30. Boucher, A., Kyriakidis, C., Collin, C.: Geo-statistical solutions for super-resolution land cover mapping. IEEE Transaction on Geoscience and Remote Sensing 46(1), 272–283 (2008)
31. Greenspan, H.: Super-resolution in medical imaging. The Computer Journal 52, 43–63 (2009)
32. Kennedy, J.A., Israel, O., Frenkel, A., Bar-Shalom, R., Azhari, H.: Super-resolution in pet imaging. IEEE Transaction on Medical Imaging 25(2), 137–147 (2006)
33. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision IJCV (2000)
34. Sun, J., Zheng, N.-N., Tao, H., Shum, H.: Image hallucinationwith primalsketch priors. In: Proc. CVPR (2003)
35. Chang, H., Yeung, D.-Y., Xiong, Y.: Super-resolution through neighbor embedding. In: CVPR (2004)
36. Roweis, S.T., Saul, L.K.: Nonlinear dimentionality reduction by locally linear embedding. In: Proc. CVPR (2003)

37. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: Proc. CVPR (2008)
38. Yang, J., Wright, J., Huang, T., Ma, Y.: Image Super-resolution via Sparse Representation. IEEE Transaction on Image Processing 19 (2010)
39. Elad, M., Aharon, M.: Image denoising via sparse and redundant representation over learned dictionaries. IEEE Transaction on Image Processing 15, 3736–3745 (2006)
40. Mairal, J., Sapiro, G., Elad, M.: Learning multiscale sparse representation for image and video restoration. Multiscale Modeling and Simulation 7, 214–241 (2008)
41. Tropp, J.: Greed is good: Algorithmic results for sparse approximation. IEEE Trans. Inf. Theory 50, 2231–2242 (2004)
42. Gersho, A., Gray, R.M.: Vector Quantization and Signal Compression. Kluwer Academic, Norwell (1991)
43. Hamerly, G., Elkan, C.: Alternatives to the k-means algorithm that find better clusterings. In: Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM) (2002)
44. Vattani, A.: k-means requires exponentially many iterations even in the plane. Discrete and Computational Geometry 45(4), 596–616 (2011)
45. Arthur, D., Manthey, B., Roeglin, H.: k-means has polynomial smoothed complexity. In: Proceedings of the 50th Symposium on Foundations of Computer Science (FOCS) (2009)
46. Hartigan, J.A., Wong, M.A.: Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the Royal Statistical Society, Series C 28(1), 100–108 (1979)
47. Dasgupta, S., Freund, Y.: Random Projection Trees for Vector Quantization. IEEE Transactions on Information Theory 55, 3229–3242 (2009)
48. Mahajan, M., Nimbhorkar, P., Varadarajan, K.: The Planar k-Means Problem is NP-Hard. In: Das, S., Uehara, R. (eds.) WALCOM 2009. LNCS, vol. 5431, pp. 274–285. Springer, Heidelberg (2009)
49. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc., Ser. B 39(1), 1–38 (1977)
50. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: Proc. ICCV (2009), http://www.wisdom.weizmann.ac.il/~vision/SingleImageSR.html
51. Freedman, G., Fattal, R.: Image and video upscaling from local self-examples. ACM Trans. Graph. 28(3), 1–10 (2010)

List of Acronyms

DCT Discrete Cosine Transform
HR High-Resolution
ICA Independent Component Analysis
K-SVD K-Singular Value Decomposition
LEE Locally Linear Embedding
LR Low-Resolution
MP Matching Pursuit
MSE Mean Square Error
MRF Markov Random Field
NE Neighborhood Embedding
OMP Orthogonal Matching Pursuit

PCA Principle Component Analysis
PSNR Peak Signal-to-Noise Ratio
RMSE Root Mean Square Error
SC Sparse Coding
PSNR Super-Resolution
RMSE Singular Value Decomposition
VQ Vector Quantization
WCSSR Within Cluster Sum of Squares