Sustainable Cities and Society 37 (2018) 358–370

Contents lists available at ScienceDirect

Sustainable Cities and Society

journal homepage: www.elsevier.com/locate/scs

Medical image super-resolution by using multi-dictionary and random forest

Shuaifang Wei[a], Xinzhi Zhou[a,*], Wei Wu[a], Qiang Pu[b], Qionghua Wang[a], Xiaomin Yang[a,*]

[a] College of Electronics and Information Engineering, Sichuan University, Chengdu, Sichuan, 610064, PR China
[b] Department of Thoracic Surgery, West China Hospital, Sichuan University, Chengdu, Sichuan,610041, PR China

ARTICLE INFO

Keywords:
Medical image
Super-resolution
Multi-dictionary
Sparse coding
Random forest

ABSTRACT

Smart City has become the direction of the development of city. Telemedicine is an important part of Smart City. Telemedicine always provides clinical health care according to the medical images of the patient. High resolution images are expected for remote diagnosis. Super-resolution technology can improve the resolution of medical images. Recently, sparse coding based super-resolution has attracted more attentions. Sparse coding based super-resolution tries to find the sparse representation of low-resolution (LR) image patches from low resolution dictionary, then reconstructs high-resolution (HR) image patches using sparse representation and HR dictionary. In this paper, we propose a sparse-based scheme for medical image super-resolution. First, we jointly divide the training patches into several clusters. Multiple dictionaries are learned from each cluster to collectively provide the least super-resolution error for the training patches. Second, random forest is trained based on the training patches and their cluster labels. Finally, for an input LR image patches, we use trained random forest to determine which cluster the patch belong to, then use the corresponding dictionary to reconstruct the patch. Thus, all the input LR patches are reconstructed with smallest error. All the reconstructed HR patches are synthesized into a completed HR image. The proposed scheme is applied to test a set of medical images. Experimental results show that both objective evaluation (PSNR) and subjective evaluation (visual effect) are improved when compare to other example-based methods.

1. Introduction

HR images are expected both in human visual system and image processing. Many fields are pursuing HR images such as remote sensing, medical imaging, machine vision and so on. In order to increase the resolution of images, we can use hardware measures: upgrading the image sensor manufacturing technology, or using a huge size sensor. Due to the constraints of the physical system, upgrading the hardware is usually expensive and has a long cycle, especially in some special fields like Computerized Tomography (CT), Magnetic Resonance Imaging (MRI) in medical imaging. Therefore, another good option is to use a specific algorithm to improve the spatial resolution of images. In the recent two decades, an image processing technique: super-resolution (SR) has attracted more attentions. SR refers to introduce more details into observed LR images, thus improve the resolution of LR images.

With the growing demand for technology, Smart City has become the trend of urban development. Remote diagnosis is an important part of the Smart City. Remote diagnosis is usually done by sending medical images to medical specialist. The medical specialist diagnoses the patient based on the voice message and the medical images. The most important factor affecting the diagnostic effect is the resolution of the medical image. HR medical images can help specialist to analyze the condition of patients more accurately. Improving the resolution of medical images using SR can greatly improve the quality of the diagnosis while save the requirement of materials and funds used in upgrading devices. In the long run, SR for medical images will contribute to smart city. Several SR algorithms have been proposed.

Single-image super-resolution (SISR) algorithms could improve the resolution of LR image $\boldsymbol{Y}$ to achieve HR image $\boldsymbol{X}$. However, if there is no any prior information to regulate this process, super-resolution will become an ill-pose problem, which expressed as: an LR image $\boldsymbol{Y}$ may correspond to a set of different HR images. A common assumption is that $\boldsymbol{Y}$ is a "low version" of $\boldsymbol{X}$. It means that $\boldsymbol{Y}$ is the result of down-sampling and blurring $\boldsymbol{X}$ with an additive white Gaussian noise:

$$\boldsymbol{Y} = SH\boldsymbol{X} + w, \tag{1}$$

where H denotes a low-pass filter to make $\boldsymbol{X}$ blurred. S performs a down-sampling operator to reduce the size of $\boldsymbol{X}$. w represents the additive Gaussian noise (Katsaggelos, Molina, & Mateos, 2007). The problem is, given a $\boldsymbol{Y}$, finding an image $\hat{\boldsymbol{X}}$ with high resolution and have $\hat{\boldsymbol{X}} \approx \boldsymbol{X}$. Taking the existence of w into account, the target is to minimize $||SH\hat{\boldsymbol{X}} - \boldsymbol{Y}||_2$. In recent years, many methods have been

* Corresponding authors.
E-mail addresses: xz.zhou@scu.edu.cn (X. Zhou), arielyang@scu.edu.cn (X. Yang).

https://doi.org/10.1016/j.scs.2017.11.012
Received 29 July 2017; Received in revised form 14 October 2017; Accepted 9 November 2017
Available online 20 November 2017

proposed to solve this problem, existing SISR methods use various priors to make this inversion stable.

Popular SISR algorithms can be mainly divided into three categories: interpolation methods, multi-frame methods and example-based methods.

Interpolation methods use the relationship of adjacent pixels to predict and insert new pixels, such as Lanczos interpolation (Duchon, 1979), Cubic interpolation (Hou and Andrews, 1978; Keys, 2003), interpolation using in medical image (Lehmann, Gönner, & Spitzer, 1999). Simple interpolation like Bicubic and Bilinear may easily produce ringing and excessive smoothing, that is because interpolations do not introduce any new details into the enlarged images.

Multi-frame methods use the complementary information of multiple LR image from same scene to create one HR image (Farsiu et al., 2004; Fransens, Strecha, & Gool, 2007; Protter et al., 2008). Multi-frame methods require different LR images taken from different displaced viewpoints for one scene, it use more prior information from successive LR image frames of same scene to reconstruct the high-frequency details. Experiments show that multi-frame methods have better performance than interpolation methods in small magnification factor (Yang, Wu, Liu, Zhou, & Binyu, 2015). But the performance of multi-frame methods is highly dependent on the qualities of input LR images. We do not always have a serious of images of some scenes with subpixel offset, especially in practical applications. Moreover, multi-frame methods will lose its advantage when in the need for high magnification factor.

Example-based methods learn the relationship between corresponding LR and HR patches from example patches, then use this relationship to predict the HR patches. In example-based methods, neighbor embedding (NE) methods (Chang, Yeung, & Xiong, 2004; Chan et al., 2009; Freeman, Jones, & Pasztor, 2002; Tang et al., 2011) assume that corresponding LR and HR patches have similar local geometry. NE methods predict HR patches by finding the linear combination of HR patches neighbor atoms in dictionary. However, NE methods do not learn enough prior information of textures and details. The results of NE methods are lack of visualizing textures and details inevitably. Regression methods is another popular example-based method. In regression methods, some pixels in HR images are predicted by using a set of statistical models directly. Wu et al. (2011) solve the mapping error problem by using kernel partial least squares regression model. Wu's method needs to find neighbors throughout the entire database and must use the same complexity of principal components to obtain desired HR patches. These steps lead to high computational costs. Timofte et al., (2014a) use L2-norm to regulate sparse coefficients, thus have a closed-form solution for the SR process by using ridge regression. Timofte's method has a very high computing speed. Zhou, Wu, Huang et al. (2017) propose a fast near-duplicate image elimination method, which can be applied to make the dictionary atoms more independent.

Yang et al. (2010) first use sparse coding (SC) to solve super-resolution problems. Sparse coding refers to use the inner product of a sparse representation vector and an over-complete LR dictionary to represent each input image patch. Once LR and HR dictionary are learned from database jointly, the HR patch can be reconstructed by using the sparse vector of corresponding LR patch and the HR dictionary. Many variant methods based on Yang's algorithm (Jeon, Anisetti, & Kang, 2013; Jeon et al., 2010; Pu et al., 2013, 2014; Qu, Mayzel, Cai, Chen, & Orekhov, 2015; Tropp and Gilbert, 2007; Timofte, De, & Gool, 2014b; Wu, Yang, Yu Pang, & Jeon, 2013; Yang et al., 2012, 2015) have been proposed to improve the SR results. Zeyde and Elad (2010) improve both the execution speed and quality of SCSR by using K-SVD (Aharon, Elad, & Bruckstein, 2006) on dictionary training, applying OMP algorithm for sparse coding. Zhou, Wang, Wu et al. (2017) propose a global context verification scheme to filter false matches for copy detection of images. Paper (Zhou et al., 2016) propose a method for detecting the image copies of a given original image generated by arbitrary rotation. Methods of (Zhou, Wang, Wu et al., 2017; Zhou et al., 2016) can be applied to match the LR patches and HR patches.

All the sparse coding methods mentioned above are single-dictionary methods. There are numerous kinds of structures existing in intricate real images. It is difficult to use a single dictionary to obtain enough priori information to represent all of these structures. Since the richness of real-world image patches, the patches are diversity in patch space. It is difficult to use one dictionary and a sparse vector to cover the entire complex patch space. It is reasonable to learn different dictionaries for different kind of patches.

Dong et al. (2011) learn a series of dictionaries from a set of image patches which clustered by K-means algorithm. Yang et al. (2011) improve the performance of K-means based methods by introducing adaptive coding to prototype dictionary. K-means based methods train different dictionaries for different input patches to have better results. However, K-means clustering does not ensure that each input patch can successfully select the most appropriate reconstruction dictionary. Wang et al. (2016) introduce QWT to image forensics and improve the detection accuracy of images. This method can be applied to improve the cluster performance of multi-dictionary methods. Paper (Gu, Sheng, & Sheng, 2016) proposed a novel *v*-Support vector classification model, which achieve a better performance compared to general vector machine. Paper (Gu, Sun, & Sheng, 2017) propose a structural minimax probability machine, which can be used as a classifier to have a satisfactory performance. The method of (Gu et al., 2016;Gu et al., 2017) can be applied to get a better classification performance using multi-dictionary method.

The proposed scheme in this paper has the following contributions to improve the SR process:

1. To train a set of typical dictionaries, which collectively provide the least super-resolving error for the training patches, we cluster the patches and update dictionaries iteratively. This process is iteratively executed for several times to reduce the total error. Thus, we make all types of patches can have the most suitable dictionary for their own reconstruction.
2. To improve the speed and accuracy of finding the most suitable dictionary, we use random forest to find the suitable dictionary for each input LR patch. Thus, we avoid the process of computing the reconstruct error of all dictionaries of each input LR patch. We propose to train a random forest model, which as a classifier, to find out which cluster the input LR patch belong to. Random forest is trained offline. Using random forest, we can select the dictionary for input LR patch directly.

The following content of the paper is organized as follows: briefly review the sparse representation applying in SISR in Section 2; explain the proposed scheme in detail, including training and reconstruction step in Section 3; experiments results and comparison are shown in Section 4; the conclusions is placed in Section 5.

2. Related works

2.1. Sparse representation for SISR

The SR scheme should be able to recover the HR image $\boldsymbol{X}$ as much as possible. In traditional example-based methods, the dictionary will become very large when we need to take more images into the training database to improve the performance. Yang et al. (2010) proposed sparse representation method for this process. The input LR images are divided into several overlapping patches to reduce the error, overlapping areas was averaged after all the patches been reconstructed, then put all the HR patches into the $\hat{\boldsymbol{X}}$ final high-resolution image

Yang's method tends to reconstruct the LR patches $\boldsymbol{x}$ of HR image $\boldsymbol{X}$ by:

$$\boldsymbol{x} \approx \hat{\boldsymbol{x}} = \boldsymbol{D}_h \boldsymbol{\alpha} \boldsymbol{\alpha} \in R^k, \ \boldsymbol{\alpha}_0 \ll \mathrm{K}, \tag{2}$$

where $\hat{\boldsymbol{x}}$ is the patch reconstructed by the approach, $\boldsymbol{D}_h$ is HR dictionary of K atoms trained from a set of HR patches extract from training database, $\boldsymbol{\alpha}$ is the sparse representation, it can be obtained from the LR dictionary by minimizing:

$$\min_{\alpha} \boldsymbol{\alpha}_0 s.\ t.\ \boldsymbol{D}_l \boldsymbol{\alpha} - \boldsymbol{F}\boldsymbol{y}_2^2 \le \varepsilon, \tag{3}$$

where $\boldsymbol{y}$ is the LR patches of the input LR image $\boldsymbol{Y}$, $\boldsymbol{D}_l$ is LR dictionary learned from LR database. $\boldsymbol{F}$ is a feature extraction operator using on collecting the features to represent the main property (e.g. gradient or edge) of the LR patches. ε is the threshold of tolerance error. The formulation (3) become a NP-hard problem due to the existence of the l_0-norm. We can directly replace l_0-norm to l_1-norm but does not affect the performance of reconstruction in practical, thus (3) become:

$$\min_{\alpha} \boldsymbol{F}\boldsymbol{D}_l \boldsymbol{\alpha} - \boldsymbol{F}\boldsymbol{y}_2^2 + \lambda \boldsymbol{\alpha}_1. \tag{4}$$

The parameter λ is use as a weighting factor to balances the sparsity of $\boldsymbol{\alpha}$ thus control the effect of the result in formulation (2). The LR patches can be reconstructed to HR patches through jointly solving formulation (3) (4). Fig. 1 illustrates this process.

The sparse coding based methods assume that the LR patches and the corresponding HR patches have the same sparse representation for their respective dictionaries, so the $\boldsymbol{D}_l$ and $\boldsymbol{D}_h$ should be learned jointly from a set of corresponding LR patches and HR patches. Yang et al. (2010) constrains the sparse coefficient of $\boldsymbol{D}_l$ and $\boldsymbol{D}_h$ in training to ensure $\boldsymbol{D}_l$ and $\boldsymbol{D}_h$ share the same codes. In Yang et al. (2010), $\boldsymbol{D}_l$ and $\boldsymbol{D}_h$ should be learned jointly from a set of corresponding LR patches Y_l and HR patches X_h extracted from training database by minimizing:

$$\min_{\boldsymbol{D}_h, \boldsymbol{D}_l, Z} \frac{1}{N} X_h - \boldsymbol{D}_h Z_2^2 + \frac{1}{M} Y_l - \boldsymbol{D}_l Z_2^2 + \lambda \left(\frac{1}{N} + \frac{1}{M} \right) Z_1, \tag{5}$$

where N and M respectively represent the dimensions of X_h and Y_l, Z is a sparse vector to constraint the $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$ to be sparse dictionaries. The results of this process ensure that the dictionary is over complete, and a large number of training samples can be represented by a few atoms. However, this training process will take a long time because the high complexity of computation.

Zeyde and Elad (2010) improve the execution speed and the reconstruction quality by using K-SVD algorithm for LR dictionary training and using the pseudoinverse directly for HR dictionary training, and OMP algorithm for reconstruction. After the training, $\boldsymbol{D}_h$ and $\boldsymbol{D}_l$ will share the same codes. In Zeyde and Elad (2010), the sparse coefficient $\boldsymbol{\alpha}$ of input LR patch can be obtained from $\boldsymbol{D}_l$ directly by using OMP algorithm to solve formulation (4), then the HR patch can be obtained by solve formulation (2). Because we have already obtained $\boldsymbol{D}_h$ and $\boldsymbol{\alpha}$, formulation (2) is a simple matrix multiplication problem.

The proposed scheme is based on the approach of Zeyde and Elad (2010), we obtain all the dictionaries using K-SVD algorithm and reconstruct the HR patches using OMP algorithm.

2.2. Random forest

In our scheme, we need to use a classifier model to classify the input LR patches. This classifier can indicate which cluster the LR patch belongs to rapidly. Thus, we can choose the most suitable dictionary to reconstruct the LR patch according to the classification result.

K-means and decision tree are commonly used as classifier models. But in our scheme, the classifier model which used in patch classification need to learn the complex inner relationship of training patches. K-means model simply clusters and classifies the patches based on the distance of each sample, this method does not guarantee that the input samples are correctly assigned to the most suitable cluster. Decision tree can learn the regular pattern from each dimension of features of training samples. But because the training of decision tree simply splits each node based on the threshold, and the classification of decision tree is based on the "event or no event" rule strictly. Thus, if there is no proper pruning, decision tree is easy to fall into over-fitting, this drawback will cause the input LR patch failing to select the correct dictionary for reconstruction.

Random forest is a novel classifier proposed by Liaw and Wiener (2002), it can be applied in classification or regression. Random forest has good resistance to noise and not easy to fall into over-fitting. A random forest is trained by a set of samples. In our scheme, random forest is used as a classifier.

2.2.1. Training

A random forest model which used as a classifier is made up of a series of decision trees. These decision trees are randomly generated by two steps. All decision trees are independent of each other. Training a random forest model is essentially training several different decision trees. Assume we want to train a random forest with l decision trees, this model can be represented as $\boldsymbol{T} = \{T_1, T_2, \ldots T_l\}$, here, T_l is a decision tree and the subscript l indicates the serial number of the decision tree. Each decision tree is generated with double random processes.

The two steps of training decision tree are as follows: The first step

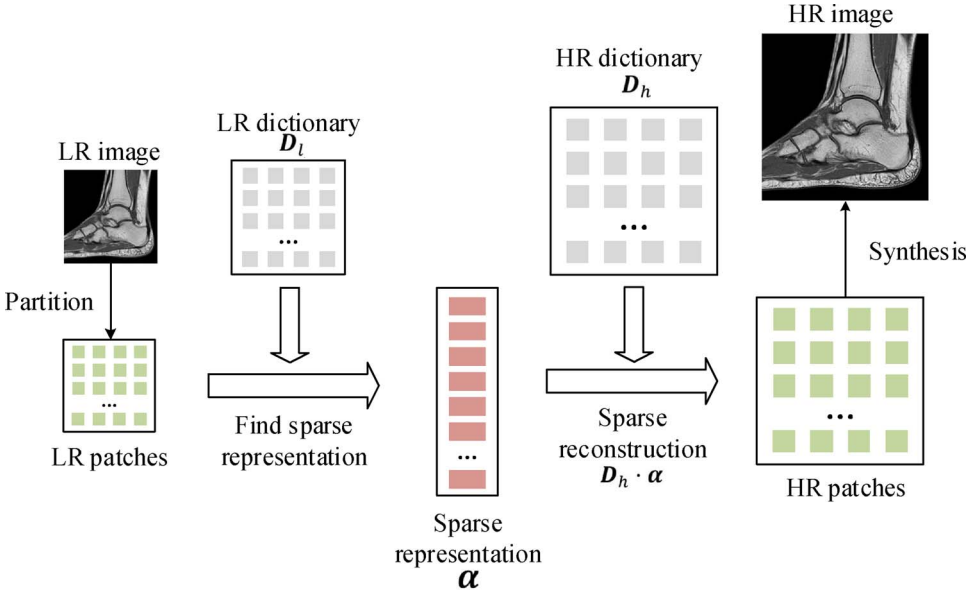

Fig. 1. Super-resolution via sparse representation.

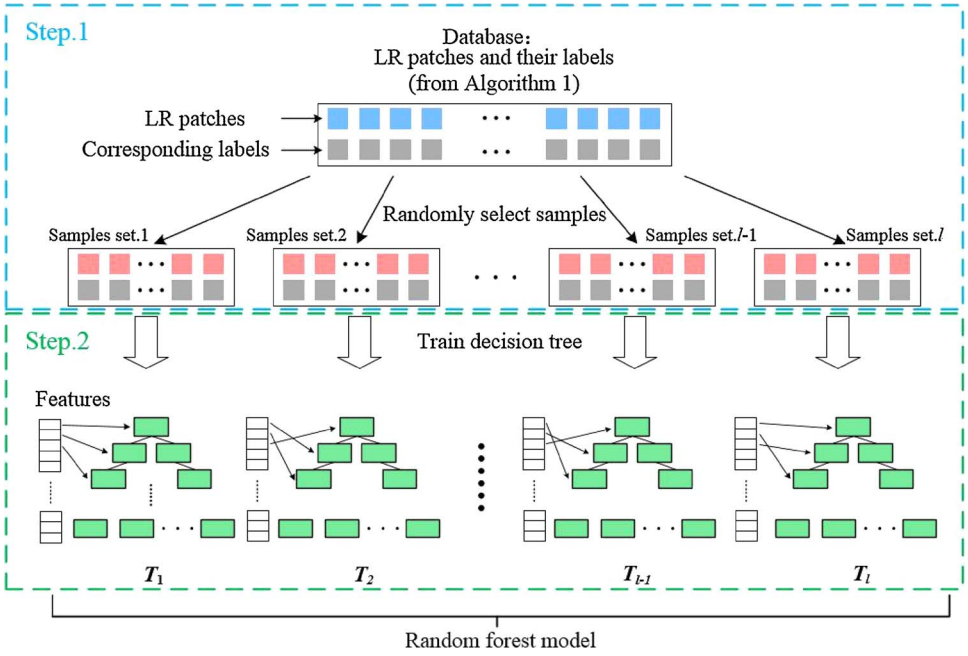

Fig. 2. The process of training a random forest model for patches classification. (a) N samples extracted from training database. Their labels are one of the output of Algorithm 1; (b) Samples selection. Follow the random forest theory in Section 2.2.1, randomly select samples for each decision tree; (c) Samples sets for decision tree training. Note that each set is different from each other; (d) Each sample set is used for train one decision tree; (e) The features is randomly selected for division of each node of decision tree.

is, randomly select samples from database. In a selected set, samples can be repeated between each other. The selected samples set are used for training a decision tree. This step is repeated several times, so that several different sets of samples are selected from the database. Each set of samples is used to train a decision tree. Illustrations (a) (b) (c) (d) in Fig. 2 show this process intuitively.

The second step is applied in node division: Randomly select a portion of features for node division step of building decision tree. This step ensures that the building of each decision tree is based on random selected features. Thus, each decision tree has its own decision-making process. Mathematically, for the samples with v-dimensional features, randomly select S features for node division ($S < v$). Illustration (e) in Fig. 2 shows this process intuitively.

This above process is repeated several times to generate a set of different decision trees. These decision trees form a random forest. Due to the introduction of random process in above two steps, random forest has good resistance to noise and not easy to fall into over-fitting, thus all decision trees do not need pruning.

2.2.2. Classification

When input a new sample to a trained random forest model, each decision tree is used to predict which cluster this sample belongs to. The sample is divided into a suitable cluster according to the "vote" result of all decision trees.

3. Proposed scheme

The proposed scheme has three main aspects: 1) Learning multi-dictionary from the training database iteratively. The learning results endure that each dictionary performs smallest reconstruct error to a cluster of LR patches. 2) using the LR patches and the cluster labels (point out which cluster the patch belong to) to train the random forest model; 3) for the LR patches extracted from a new LR images, use the trained random forest model to predict which cluster it belongs to, then choose the corresponding dictionary for its reconstruction.

3.1. Learning multi-dictionary iteratively

We have assumed that some LR patches can be reconstructed to HR patches with better effect by using classified dictionary, the multi-dictionary learning is based on this spirit. For a set of given training images, we treat them as the HR images, and the corresponding LR images is their down-sampled versions. Following the Yang et al. (2010), we use bicubic method to get the LR images. Divide the LR and HR images into several small LR patches $\boldsymbol{p}_l$ and HR patches $\boldsymbol{p}_h$, these patches are used as the training samples. Our goal is to learn j pairs of LR and HR dictionaries $\mathbf{D} = \{(\boldsymbol{D}_{l,1}, \boldsymbol{D}_{h,1}), (\boldsymbol{D}_{l,2}, \boldsymbol{D}_{h,2}), ..., (\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})\}$, which make all the samples can have smallest reconstruct error to its corresponding dictionary. Here, $(\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})$ is the j-th cluster dictionary pair, including one LR dictionary $\boldsymbol{D}_{l,j}$ and corresponding HR dictionary $\boldsymbol{D}_{h,j}$. The subscript j represents the serial number of dictionary pair. These two dictionaries are used for the reconstruction of j-th cluster input patches. Correspondingly, $\boldsymbol{p}_l$ and corresponding $\boldsymbol{p}_h$ are also divided into j classes $\mathbf{P} = \{(\boldsymbol{p}_{l,1}, \boldsymbol{p}_{h,1}), (\boldsymbol{p}_{l,2}, \boldsymbol{p}_{h,2}), ...(\boldsymbol{p}_{l,j}, \boldsymbol{p}_{h,j})\}$. Here, $(\boldsymbol{p}_{l,j}, \boldsymbol{p}_{h,j})$ is the j-th cluster patch pairs, including a set of LR patches $\boldsymbol{p}_{l,j}$ and a set of corresponding HR patches $\boldsymbol{p}_{h,j}$. The subscript j represents the serial number of patch set. The patch set $(\boldsymbol{p}_{l,j}, \boldsymbol{p}_{h,j})$ is used to train dictionary pair $(\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})$, using K-SVD algorithm as Zeyde and Elad (2010).

This problem can be treat as a clustering problem: Given the collection of pairs of patches, we group the patches into j cluster. The HR and LR dictionaries are learned for each cluster. These HR and LR dictionaries could precisely reconstruct the patches in each cluster. This is, using the dictionary $(\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})$ to reconstruct the LR patches $\boldsymbol{p}_{l,j}$, the result $\hat{\boldsymbol{p}}_{h,j}$ should have the smallest error. Mathematically, for all the clusters of the patches, we should solve this problem by minimize the function:

$$\min_{\boldsymbol{D},\boldsymbol{P}} \sum_{J=1}^{j} \hat{\boldsymbol{p}}_{h,J} - \boldsymbol{p}_{h,J}{}^2 \tag{6}$$

To solve the problem, we should either have the patches $\mathbf{P}$ or the dictionary set $\mathbf{D}$. However, neither of these can be obtained in advance. Our scheme overcome this contradiction iteratively. Firstly, all the LR patches $\boldsymbol{p}_l$ and its corresponding HR patches $\boldsymbol{p}_h$ are clustered to j clusters randomly. These clusters are used to train the multi-dictionary $\mathbf{D}$, then $\mathbf{D}$ are applied to reconstruct all the LR patches. For one LR patch $\boldsymbol{y} \in \boldsymbol{p}_l$, the approaches mentioned in Section 2 are adopted to reconstruct the HR patch $\hat{\boldsymbol{x}}$ by solving:

$$\min_{J} \boldsymbol{F}\boldsymbol{D}_{l,J\,J} - \boldsymbol{F}\boldsymbol{y}_2^2 + \lambda_{J1}, \tag{7}$$

$$\hat{\boldsymbol{x}}_J = \boldsymbol{D}_{h,J}\boldsymbol{\alpha}, \tag{8}$$

where J is from 1 to j. In our scheme, this problem is solved by using approach of Zeyde and Elad (2010), formulation (7) will be solved by using the OMP to get the sparse coefficient $\boldsymbol{\alpha}$, and formulation (8) will be solved by using matrix multiplication directly. The reconstruct error is computed by:

$$e_J = \boldsymbol{x} - \hat{\boldsymbol{x}}_J{}^2 \tag{9}$$

where $\boldsymbol{x} \in \boldsymbol{p}_h$, $\boldsymbol{x}$ is the corresponding HR patch of $\boldsymbol{y}$ in the training samples, e_J note the reconstruct error of $(\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})$, thus each patch in $\boldsymbol{p}_l$ will have its own error vector $\boldsymbol{e} = \{e_1, e_2, ..., e_j\}$. The $\boldsymbol{y}$ will be classified as q-th cluster when $e_q = min\{e_1, e_2, ..., e_j\}$. After this step, all the patches are classified based on the smallest reconstruct error of same dictionary. The above steps (establish the dictionary and calculate the error) are performed iteratively until the total error (6) is converge to the tolerance threshold. Final clustering results and dictionaries will be used for training random forest model, and super-resolution of new images. The process of this iterative algorithm is illustrated in Algorithm 1.

Algorithm 1 (*Learning multi-dictionary iteratively*).

Input: LR patches $\boldsymbol{p}_l$, HR patches $\boldsymbol{p}_h$, number of iterations t, number of clusters j

Randomly cluster $\boldsymbol{p}_l$ and $\boldsymbol{p}_h$ into j clusters, each cluster is used to train a dictionary pairs.

For 1 to t:

- Use all the dictionary pairs to reconstruct all $\boldsymbol{p}_l$ by solving fomulation (7)(8).
- Compute the error of all $\boldsymbol{p}_l$ to all dictionary pairs by using fomulation (9).
- To all the $\boldsymbol{p}_l$, put it and its corresponding $\boldsymbol{p}_h$ into a new cluster according to the dictionary which have minimum error.
- Train new dictionary pairs according to the new patches cluster.

End

Output: Multi-dictionary pairs $\mathcal{D}$, patches cluster label J.

3.2. Training of random forest

In our scheme, the training of a random forest model needs a set of LR patches and their corresponding labels. The LR patches $\boldsymbol{p}_l$ are extracted from training database and the labels J is one of the output of Algorithm 1. Select part or all of the LR patches and labels as the samples to train the random forest (affection of the number of samples will be discussed in Section 4), so that we can get a random forest classifier model. The law of the classification of random forest is based on the inherent regular of **P**. The training step is shown in Fig. 2.

3.3. Reconstruction

After the training step, we have a set of dictionaries $\mathbf{D}= \{(\boldsymbol{D}_{l,1}, \boldsymbol{D}_{h,1}), (\boldsymbol{D}_{l,2}, \boldsymbol{D}_{h,2}), ..., (\boldsymbol{D}_{l,j}, \boldsymbol{D}_{h,j})\}$, and the random forest model. For an input new LR images, decomposes it into small patches with overlap, the size of the patches and overlap must same as the size used in training dictionary. Each patch will be classified by the random forest, the result of classification is in the range of $J \in \{1,2,...,j\}$.

In our scheme, the clustered LR patches are used to train the random forest model. For a trained model with i clusters $\boldsymbol{C} = \{c_1, c_2, ..., c_i\}$ and l decision trees, when input a new LR patch to the model, each tree will give its own classification result. The set of the result is $\boldsymbol{r} = \{r_1, r_2, ..., r_l\}$, $r_l \in \boldsymbol{C}$, after all the results are given, the cluster with the highest number of predictions is used as the final predict result.

For an q-th cluster LR patch, dictionary pairs $\boldsymbol{D}_{l,q}$, $\boldsymbol{D}_{h,q}$ is adopted to reconstruct the HR patch by solving formulation (7) (8). After all patches are rebuilt, the images are grouped into a complete high-resolution image according to the previously decomposes grid. Fig. 3 shows the reconstruction framework in detail, and the details of classification in random forest model are shown in Fig. 4.

4. Experimental results

4.1. Database

We present a series of experiments to validate the effectiveness of the proposed SR method. In order to compare our method to other example-base super-resolution method, we use the same training images as (Yang et al., 2010; Zeyde and Elad, 2010), which contains 91 high-resolution images with rich details. The LR and HR patches are extracted from these training images. To extract profuse image patches for multi-dictionary learning, the down-sample version of these 91 images are also used for training. The pictures used for testing are from Siemens Medical Image Gallery, which contain 7 MRI and 3 CT images, can be downloaded from https://www.healthcare.siemens.com, size of these images is set as 512 × 512 pixels. Fig. 5 shows the images we use in experiment.

4.2. Settings

The main settings of the experiment follow the idea of (Yang et al., 2010; Zeyde and Elad, 2010), including such aspects: For a color image, the low frequency component of the grayscale component and the color component can be reconstructed using a simple and fast traditional interpolation method, such as bicubic interpolation. This measure does not affect the reconstruction effect, but improve the operation speed. The training sample is constructed as follows: The HR patches are extract from HR images directly. Next, these HR images are down-sampling to low-version images, then the low-version images are up-sampling to an interpolated version (use the same factor as down-sampling). These interpolated pictures are used as LR images because they have missed high frequency information. These interpolated images are also divided into patches by same grid used in HR images, then use the feature extraction operator $\boldsymbol{F}$ to extract the features from interpolated patches. These features are used as the LR patches. We use 4 filters as the extraction operator $\boldsymbol{F}$, including first and second order gradient, respectively for horizontal and vertical directions: $f_1 = f_2^T = [1, -1]$ and $f_3 = f_4^T = [1, -2, 1]$.

In the training step of our experiment, the size of the LR patches is chosen as 3 × 3 pixels, the upscale factor is set as ×3, so the HR patches is 9 × 9 pixels. Because of the interpolation process mentioned before, the interpolated patches are used as the atom in LR dictionary and the size is 9 × 9 pixels. The overlap size is 2 pixels for all the images. All the data in the experiment is saved and processed in the form of column. In the super-resolution step, the settings are the same as in training, this is, using the same grid to extract LR patches from the testing images, and same filter are adopted to these LR patches.

4.3. Affection of parameters

The results of our scheme are affected by many variable parameters. The performance of random forest model is influenced by the number of decision trees and the number of training samples, the accuracy of the random forest classification directly affects the final reconstruction effect. Number of dictionary pairs may also affect the performance. This section will examine the impact of these parameters on the results.

In order to investigate the affection of parameters mentioned before, we fix some of the minor parameters, which either will not affect the final effect, or the impact is obvious, and has been confirmed. In training step of all the dictionaries, the K-SVD algorithm is iteratively performed 20 times because the error reduction caused by the increasing in the times of iterations has been less than 0.1% when to iterate more times. The proposed Algorithm 1 is executed cyclically 20 times to have an acceptable total error. The number of atoms in each

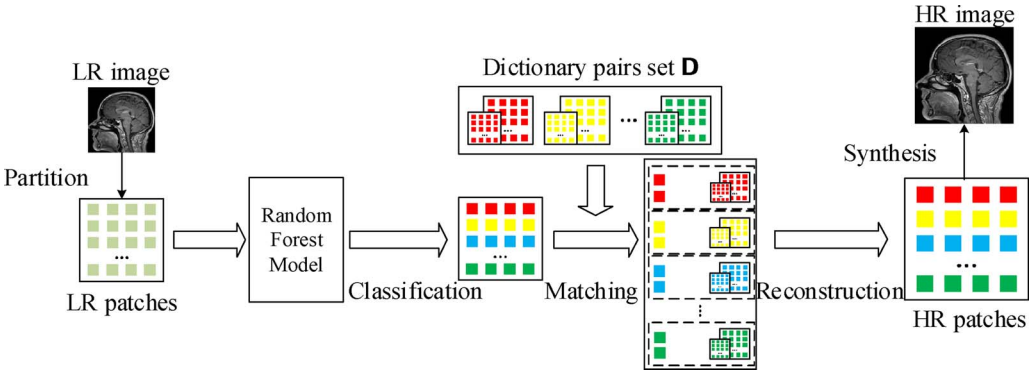

Fig. 3. Framework of super-resolution with random forest.

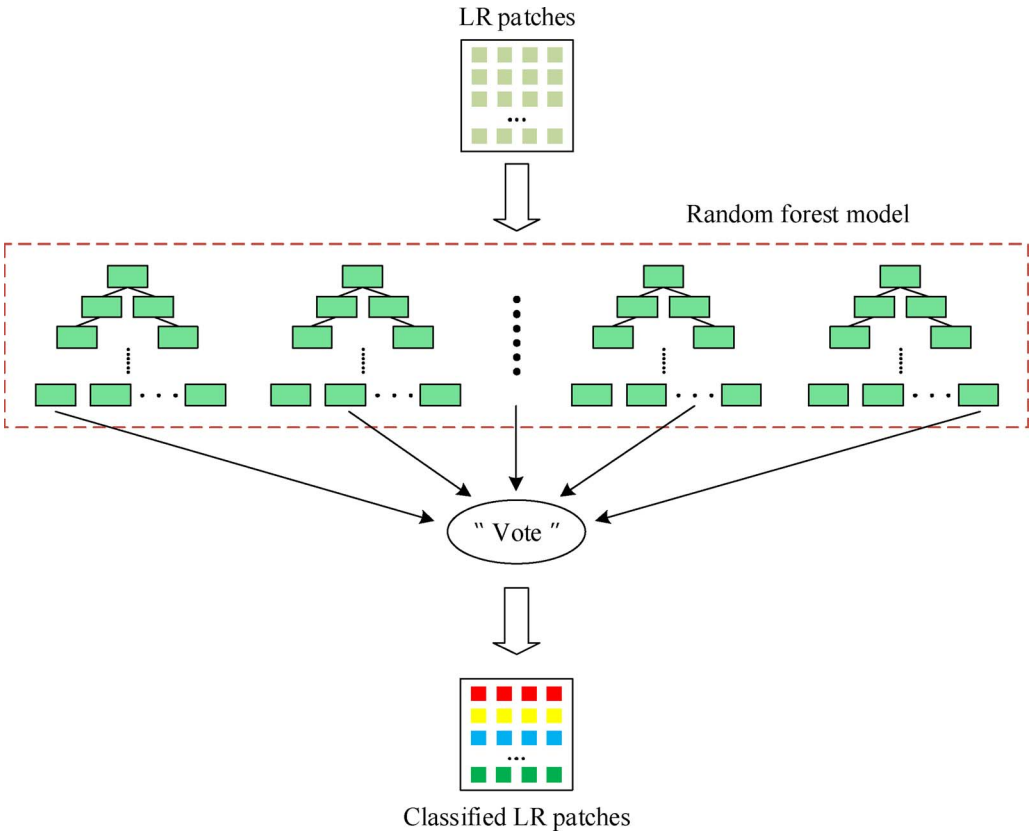

Fig. 4. Patches classification using random forest model. "Vote" means that: for each input LR patch, each decision tree in random forest model will give its own classification result, all the results are counted and the results of the voting are the final result of the classification.

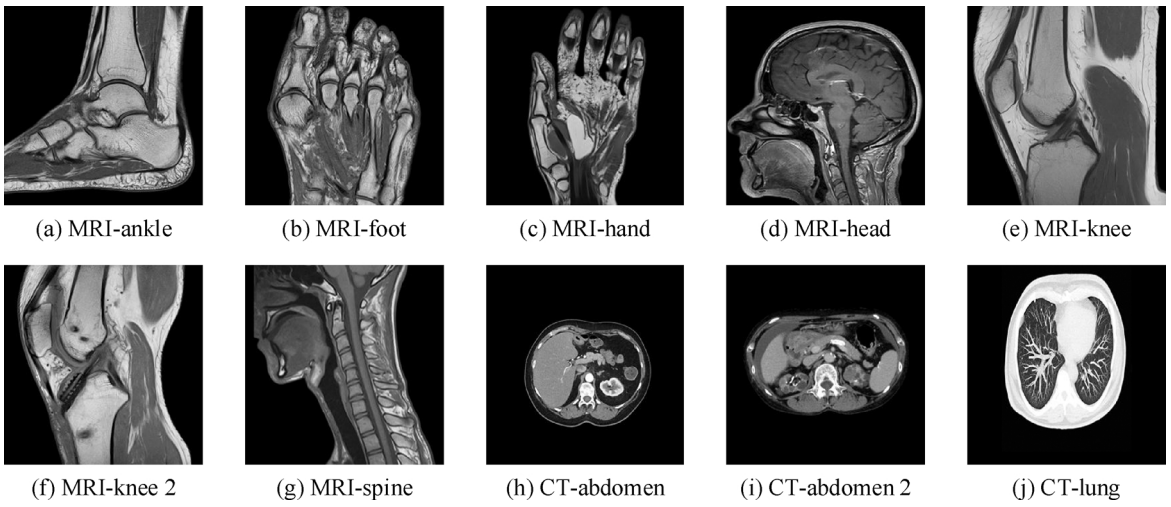

Fig. 5. Testing medical images in our experiment.

Table 1
Setting of all the parameters of experiments.

Testing images size	512 × 512 pixels	Overlap size	2 pixels
Upscale factor	×3	Dictionary size	1024 atoms
LR patch size	3 × 3 pixels	Algorithm 1 iteration times	20
HR patch size	9 × 9 pixels		

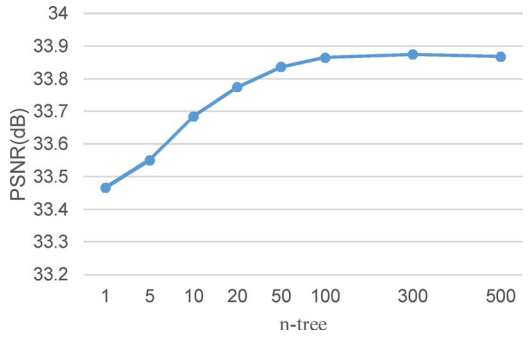

Fig. 6. Effect of *n*-tree.

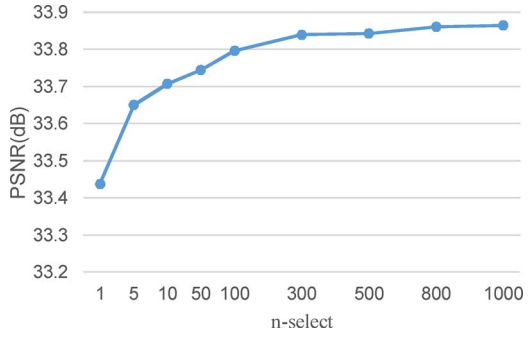

Fig. 7. Effect of *n*-select.

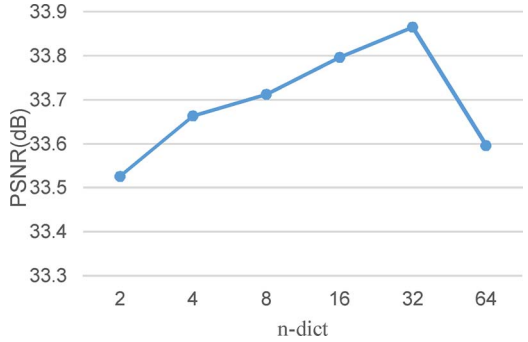

Fig. 8. Effect of *n*-dict.

dictionary is 1024 for fast training while ensuring the quality of reconstruction. All the images and patches are converted to YCbCr space. The chrominance channels Cb and Cr is reconstructed by bicubic interpolating, our scheme is only applied in luminance information Y. The quality of the super-resolution results is evaluated using Peak Signal-to-noise Ratio (PSNR), which have a high consistency with visual quality. The higher the PSNR that the better of the reconstruction effect. All the setting of parameters of our experiments are shown in Table 1.

The classification accuracy of random forest is an important indicator of the results of reconstruction. The more accurate the classification of each patch, the more matching the dictionary can be selected to achieve a smaller error. In our scheme, two parameters will affect the

Table 2
Comparison of PSNR value (dB) of different methods.

	Bicubic	Yang et al. (2010)	Zeyde and Elad (2010)	Proposed
MRI-ankle	30.49	31.28	31.51	31.60
MRI-foot	31.20	32.20	**32.28**	32.21
MRI-hand	33.25	34.57	**34.75**	34.73
MRI-head	31.84	33.65	33.84	**33.85**
MRI-knee	33.41	34.37	34.70	**34.79**
MRI-knee 2	33.46	34.51	34.85	**34.92**
MRI-spine	35.47	37.53	37.97	**38.23**
CT-abdomen	30.84	32.68	32.28	**33.34**
CT-abdomen 2	34.59	35.66	35.89	**36.11**
CT-lung	26.94	28.55	28.32	**28.90**
Average	32.15	33.50	33.64	**33.87**

performance of random forests: the number of decision trees and the number of the selected training samples. We control the performance of random forest as follow:

4.3.1. Number of decision trees

First, we vary the number of the decision trees (*n*-tree) from 1 to 500 when training a random forest model. The number of samples used for training random forest is set at 1,000.000, and use 32 dictionaries pairs trained by 5,000.000 patches pairs, these patches are extracted from database using Algorithm 1. The upscale factor is set as ×3. We observe the trend of average PSNR of all the test images. The results are shown in Fig. 6. According to (Liaw and Wiener, 2002), the more the *n*-tree, the more accurate the classification. As shown in Fig. 6, when *n*-tree increases from 1 to 100, the average PSNR value increases significantly, which from 33.47 dB to 33.87 dB. This performance indicate that the classification is become more accurate, so does the reconstruct effect. When the *n*-tree is greater than 100, even if the value of *n*-tree is greatly increased, the change of average PSNR will not change obviously, the variation range does not exceed 0.01 in this case. This phenomenon shows that when the *n*-tree is greater than 100, the classification performance of random forest is saturated. At this situation, increasing in the value of *n*-tree will not affect the results of classification. It is worth noting that the larger the *n*-tree, the longer the time to train the random forest model.

4.3.2. Number of training samples

The second way to control the performance of a random forest is to vary the number of selected training samples (*n*-select). We extract 5,000.000 patches from the training database for clustering (Algorithm 1), then randomly select a number of patches from which to train the random forest model. In theory, the greater the value of *n*-select, the random forest can more accurately learn the inherent characteristics of patches with same label. We vary the value of *n*-select from 1000 to 1,000.000, the *n*-tree is set as 100 for fast training and ensure the quality of reconstruction, the number of dictionaries pairs is set as 32, upscale factor is set as ×3. The trend of average PSNR is shown in Fig. 7. When the value of *n*-select increases from 1000 to 1,000.000, the average PSNR is from 33.44 dB to 33.87 dB. Note that when *n*-select is above 800,000, random forest model will reach saturation, which performance will no longer increase, the average PSNR is almost constant. The increases of *n*-select will also result in increased training time of random forest.

4.3.3. Number of dictionaries

Since the increasing of *n*-tree and *n*-select can improve the performance of random forest, the accuracy of classification for LR patches increases with the two parameters. As show in Figs. 6 and 7, the more accurate the classification results, the better the effect of reconstruction, this conclusion proves that the clustering in Algorithm 1 is effective. Each dictionary pair in **D** can reconstruct a cluster of LR patch with minimal error. As long as we have an accurate classification in

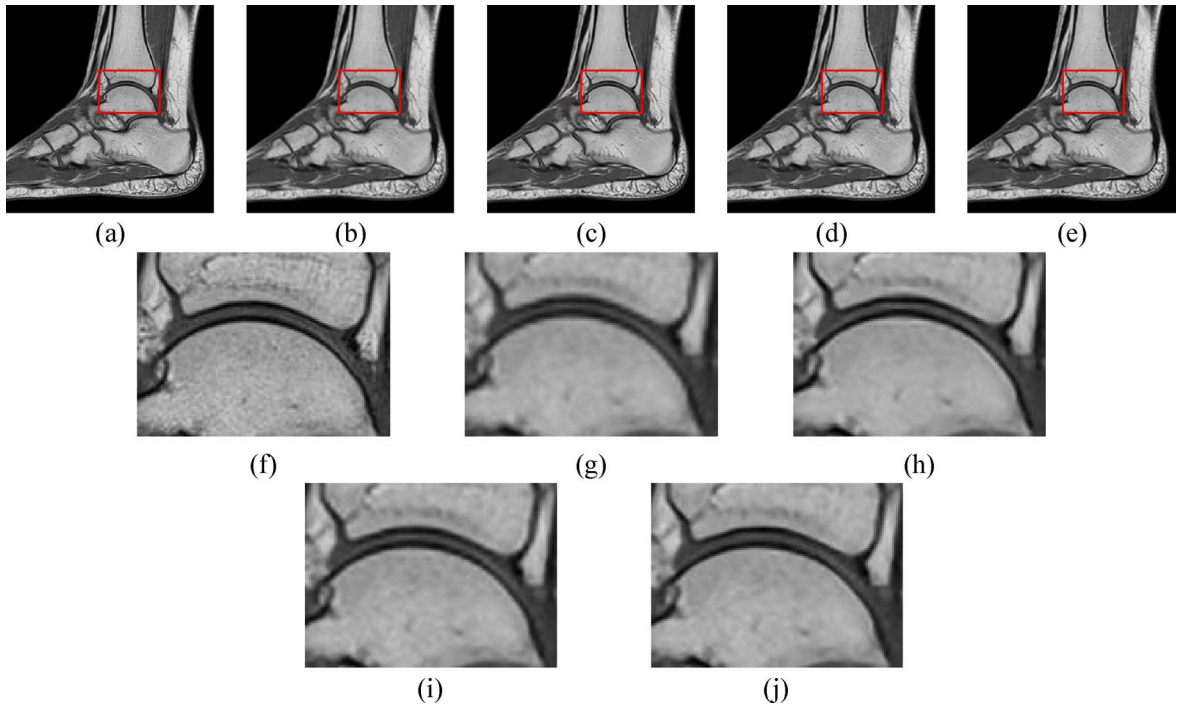

Fig. 9. Comparison of visual quality of MRI-ankle. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

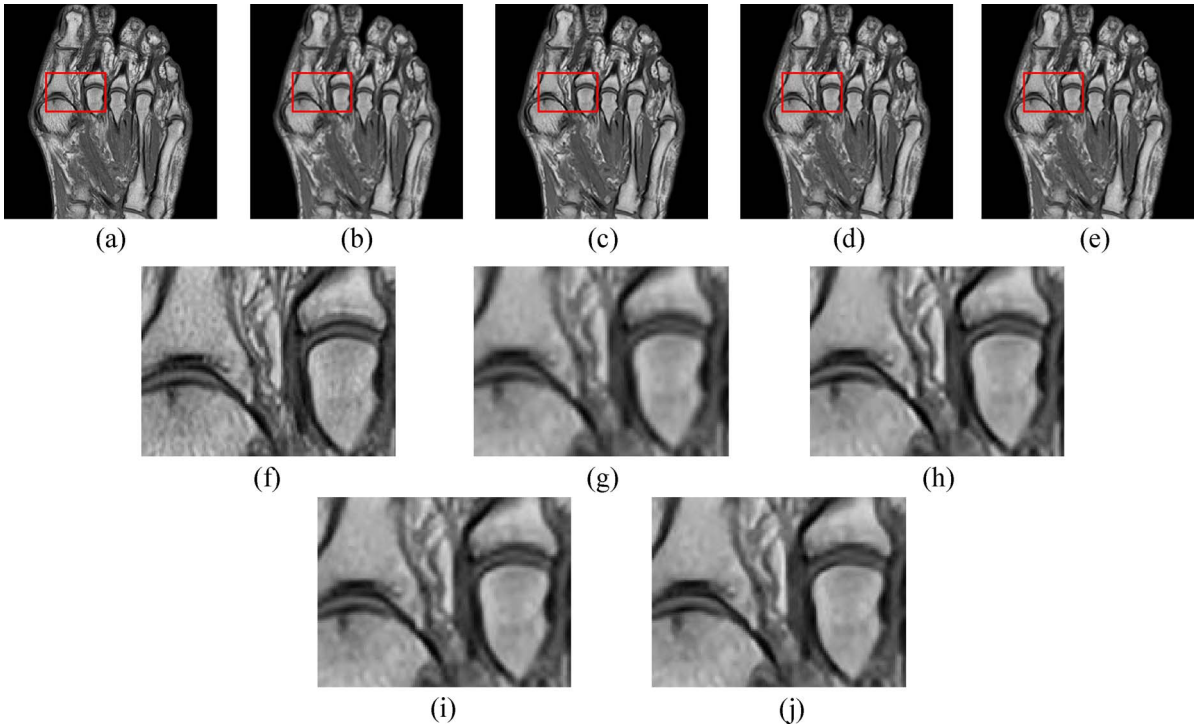

Fig. 10. Comparison of visual quality of MRI-foot. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

random forest model, the best dictionary pairs will be chosen to have a best reconstruct result for all the LR patches.

Theoretically, the more the number of dictionaries pairs can cope with more clusters of patches, that the reconstruction errors of these patches are minimized, so the number of dictionaries pairs will also influence the effect of reconstruction. We vary the number of dictionaries pairs (n-dict) from 2 to 64, note that those dictionaries is trained by same 5,000.000 patches as before, just change the number of classes to run Algorithm 1. According to the results of Figs. 6 and 7, for quick and efficient reconstruction, n-sample and n-tree are set at 1,000.000 and 100 respectively. The trend of average PSNR is shown in Fig. 8. Reconstruction quality increases with n-dict when n-dict is less than 32, however, the average PSNR is plummeted when n-dict is 64. The reason is that all the dictionaries pairs is trained by the samples in the same class. When the samples are divided into too many clusters, some cluster may only have a few samples.

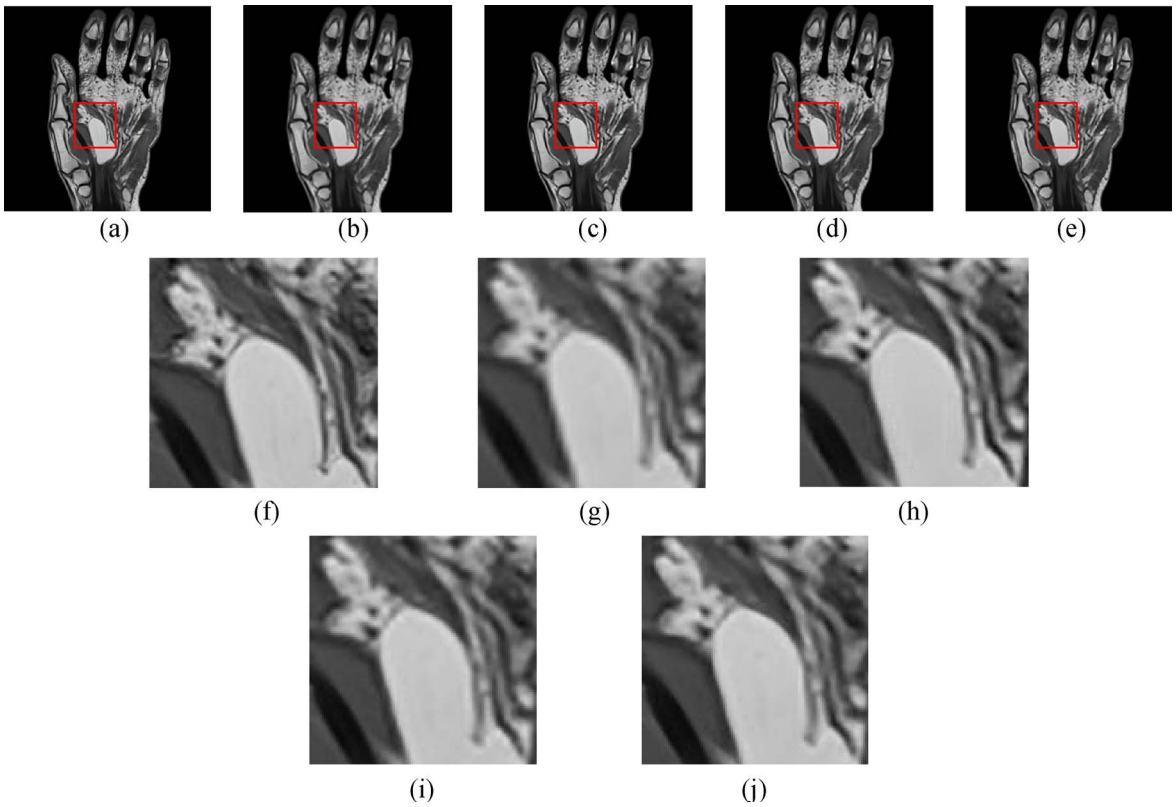

Fig. 11. Comparison of visual quality of MRI-hand. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

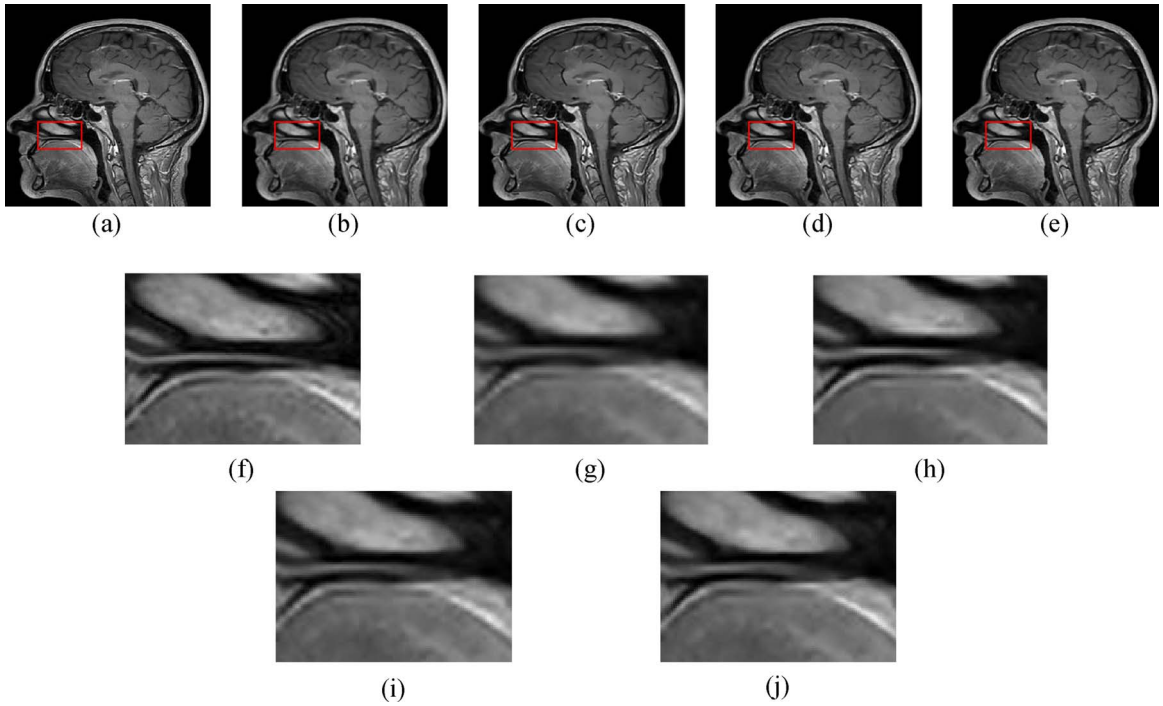

Fig. 12. Comparison of visual quality of MRI-head. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

It has been proven in (Yang et al., 2010) that the more the sample used to train the dictionary, the better the reconstruction performance. We check the number of samples in each cluster, when the *n*-dict is set as 64, some of the clusters just have several thousand of samples when other are hundreds of thousands, the absence of the sample led to a decline in the effect of reconstruction. If we want to improve the performance when *n*-dict is above 64, we should extract more samples from the database. But it can't ensure the increasing of samples in each cluster because some clusters may become "upset", and this will lead to a dramatic increase in computing time and memory consumption. In

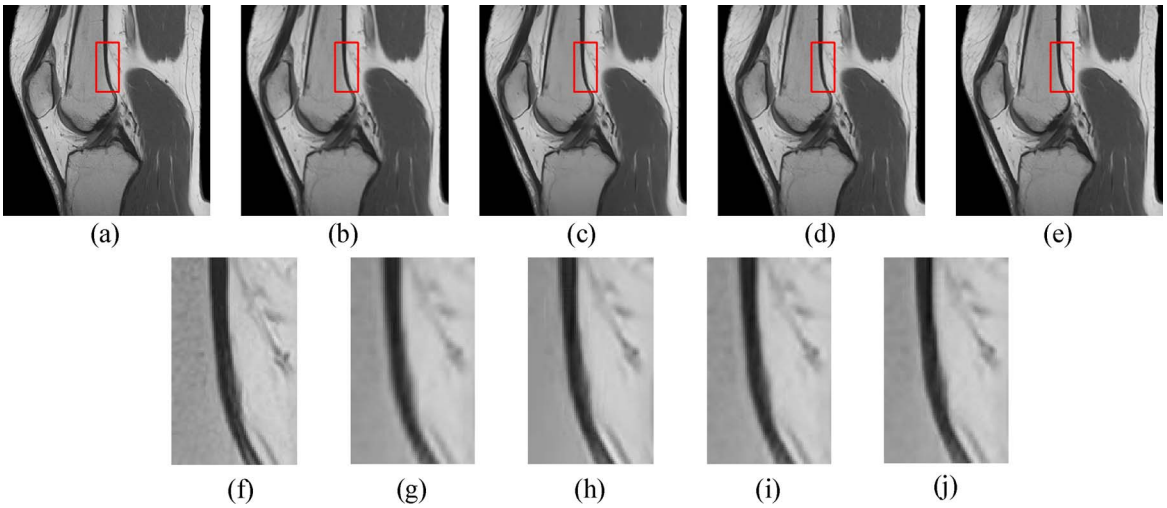

Fig. 13. Comparison of visual quality of MRI-knee. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

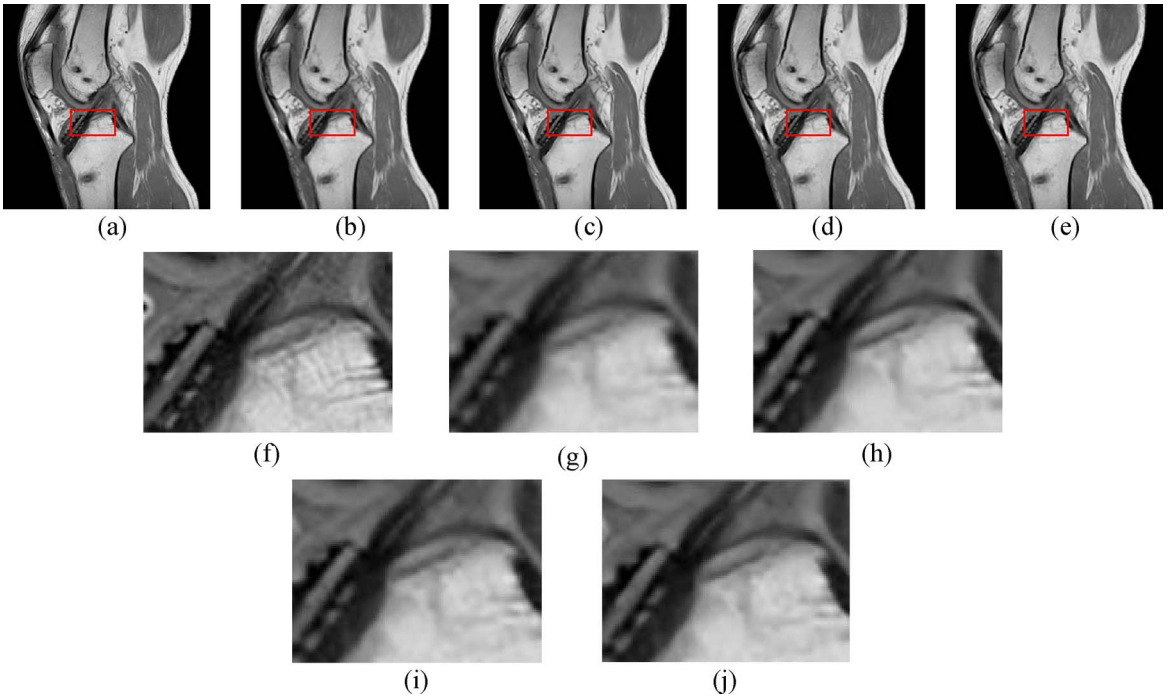

Fig. 14. Comparison of visual quality of MRI-knee2. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

summary, in the current hardware conditions, *n*-dict set to 32 is reasonable and efficient.

4.4. Comparison with the state-of-the-art algorithms

To prove that our scheme can get a smaller reconstruction error, we compare our reconstruction result with other single-dictionary method, including Yang et al. (2010) and Zeyde and Elad (2010), Bicubic method is also taken into account. All the experiments share the same database. In our scheme, we extract 5,000.000 patches from the database and *n*-dict is set as 32, so that each dictionary will trained by average nearly 150,000 samples, *n*-tree and *n*-select are set to 100 and 1,000.000 respectively for fast computation and best quality. In other scheme, 150,000 patches are extracted from the database to train their own dictionary. All the testing images are applied in those method, upscale factor is set to $\times 3$, the PSNR result is shown in Table 2. In proposed scheme, average PSNR is higher than Zeyde et al., 0.23 dB and Yang et al., 0.37 dB, and for most of the images, the proposed scheme has the best effect. The result show that proposed method can reach smaller reconstruction error and recover more information in the testing images.

Visual quality of result is shown in Figs. 9–18. By observing the details of the reconstructed results, we can find that proposed scheme is better in recovering the details and high frequency parts, and have shaper edges, clearer textures and fewer ringing artifacts. Among them, shaper edge information and less artifacts are very beneficial to improve the quality and amount of information of medical images. MRI and CT images are mostly grayscale images, and some areas of which are even very close to black and white images, without explicit color information. In this situation, clear edge information is the key to improving quality.

We select the reconstruction results of some testing images for

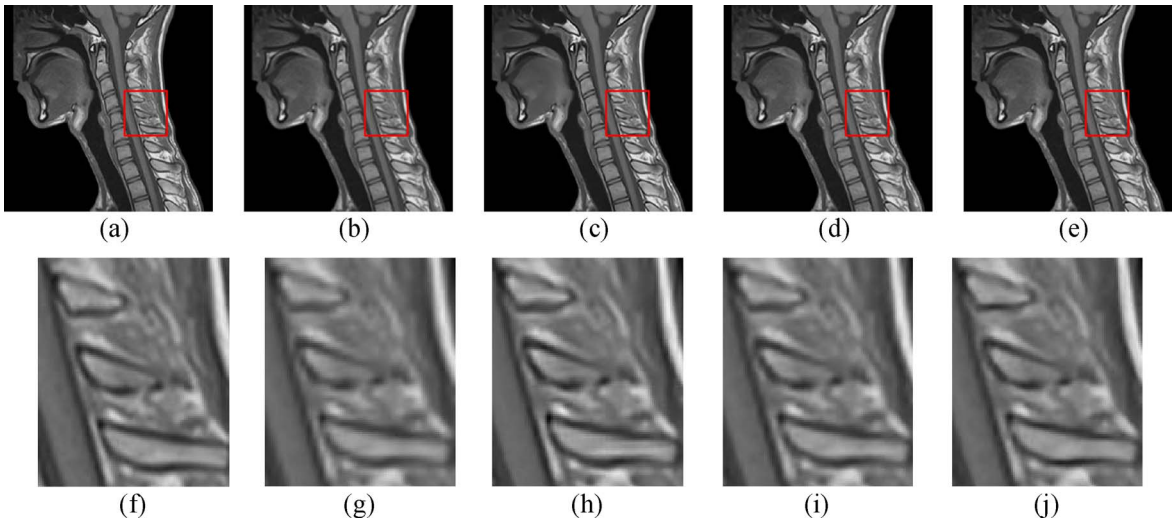

Fig. 15. Comparison of visual quality of MRI-spine. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

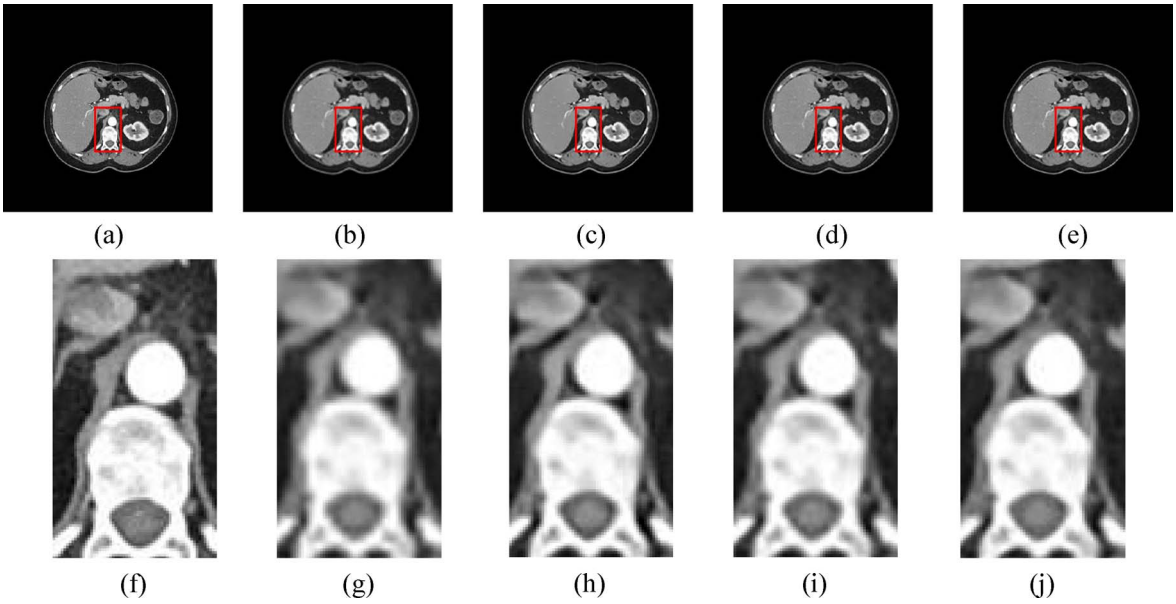

Fig. 16. Comparison of visual quality of CT-abdomen. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

careful analysis in order to prove that our scheme is superior to other methods. The analysis is as follows:

In MRI-ankle (Fig. 9), we mainly focus on the zoomed version of marked area of (a)-(e). In the result of Bicubic method (g), the entire image is too smooth and lost most of the details information. Moreover, there is a lot of jagged at the junction of the joint capsule and the bone. In the result of Yang's method (h), more details are restored, but artifacts appear in some areas, the entire image still lost some details. In the result of Zeyde's method (i), artifacts are overcome and have more details, but noise appear in some areas and these noises affect the visual effect. In the result of proposed method (j), artifacts and noise are overcome. The dividing line of joint capsule and bone is clearer and smoother, and (j) have the richest of details. The above analysis results can also be applied in the other images which have obvious boundary or edge, such as MRI-foot (Fig. 10), MRI-hand (Fig. 11), MRI-head (Fig. 12), MRI-knee (Figs. 13 and 14), MRI-spine (Fig. 15), and CT-lung (Fig. 18).

In some medical images like CT-abdomen (Figs. 16 and 17), the transition zones are very smooth. This kind of images have no obvious edges and lines. The proposed method also performs better effect in these kinds of medical images. In CT-abdomen2 (Fig. 17), we also compare the characteristics of each method. In the result of Bicubic method (g), the entire image become smooth and blur in visual effect. We can't see any details clearly in (g). In the result of Yang's method (h), more details are restored, but there are some burrs and ringing occur in the periphery of the vertebral, and some artifacts appear near the spine. Moreover, an excessive black line appears inside the vertebral. In the result of Zeyde's method (i), the defects of the Yang's method are overcome. But there are some distortions occur on the right side of the vertebral. Some curved boundaries are not smooth enough, a small amount of jagged attached to these curved boundaries. In the result of proposed method (j), curved boundaries of the image are smooth enough. The geometry of the reconstructed image is almost identical to the original image (f), and no ringing, artifacts, and distortions occur in these images. The results of proposed method have the richest details.

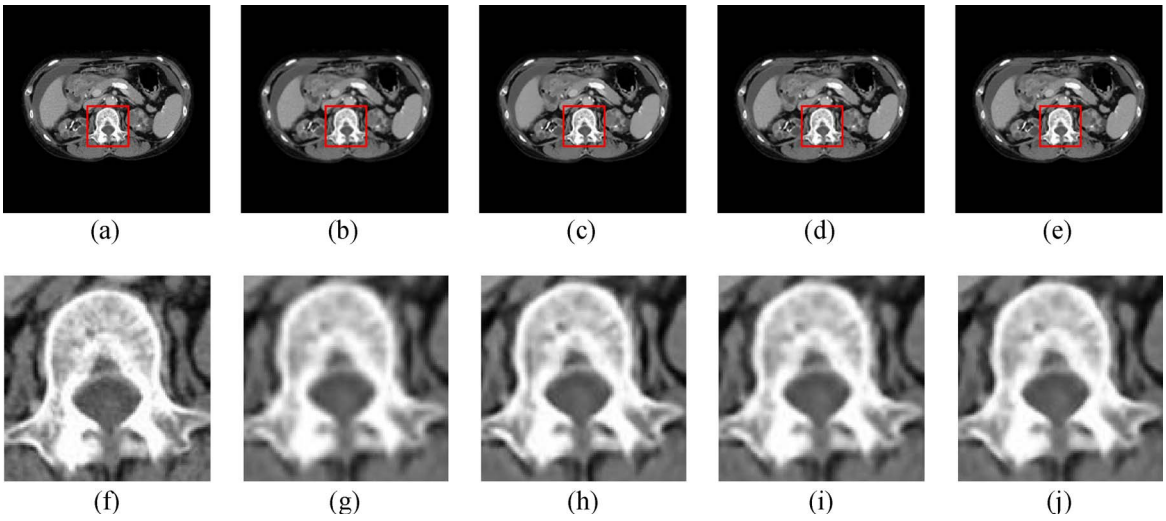

Fig. 17. Comparison of visual quality of CT-abdomen2. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

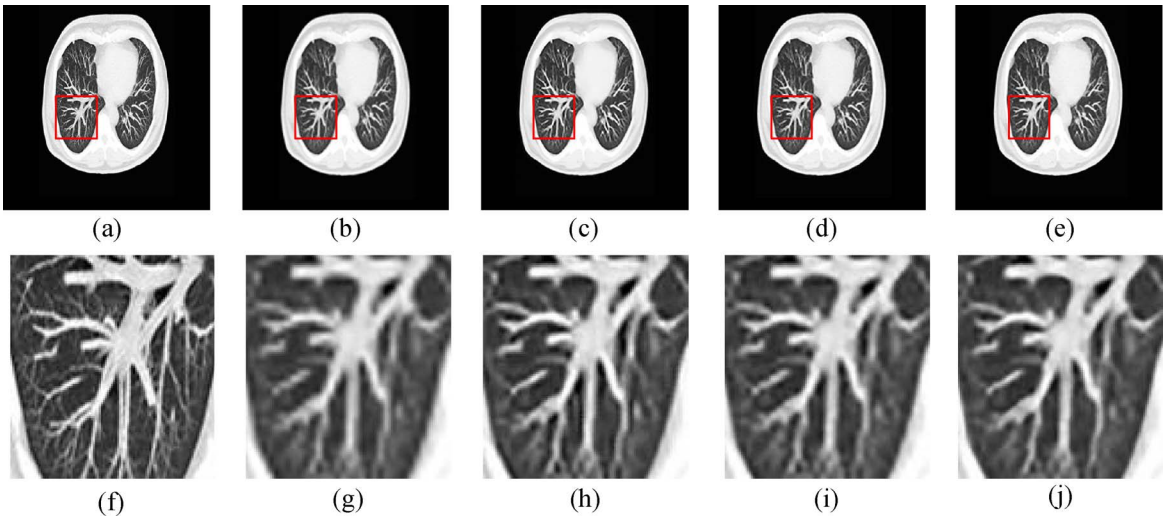

Fig. 18. Comparison of visual quality of CT-lung. (a)Original, (b)result of Bicubic method, (c)result of Yang's method, (d)result of Zeyde's method, (e)result of proposed method, (f)–(j) are the zoomed version of marked area of (a)–(e).

5. Conclusions

We have proposed an example-based single image super-resolution scheme for medical images. Our scheme clusters a number of low-resolution patches based on their reconstruction error of a set of dictionaries, then uses these patches to update their corresponding dictionaries. Random forest is trained by those patches to classify input low-resolution patches and then select the corresponding dictionary to reconstruct it to minimum reconstruction error. We have studied the effect of parameters on the performance, selected the most suitable parameters for reconstruction. The experimental results show that our scheme have better performance compare to state-of-the-art single-dictionary methods.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (#61701327, #61711540303, and #61473198), National Research Foundation of Korea (#NRF-2017K2A9A2A06013711), Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) Fund, Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

References

Aharon, M., Elad, M., & Bruckstein, A. (2006). *K-SVD. an algorithm for designing over-complete dictionaries for sparse representation.* IEEE Press.

Chan, T. M., Zhang, J., Pu, J., & Huang, H. (2009). Neighbor embedding based super-resolution algorithm through edge detection and feature selection. *Pattern Recognition Letters, 30*(5), 494–502.

Chang, H., Yeung, D. Y., & Xiong, Y. (2004). Super-resolution through neighbor embedding. *IEEE computer society conference on computer vision & pattern recognition* (pp. 275–282).

Dong, W., Li, X., Zhang, D., & Shi, G. (2011). Sparsity-based image denoising via dictionary learning and structural clustering[C]: IEEE Conference on Computer Vision and Pattern Recognition. *IEEE Computer Society,* 457–464.

Duchon, C. E. (1979). Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology, 18*(8), 1016–1022.

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

Fransens, R., Strecha, C., & Gool, L. V. (2007). *Optical flow based super-resolution: A probabilistic approach.* Elsevier Science Inc.

Freeman, W. T., Jones, T. R., & Pasztor, E. C. (2002). Example-based super-resolution. *IEEE Computer Graphics & Applications, 22*(2), 56–65.

Gu, B., Sheng, V. S., & Sheng, S. (2016). A robust regularization path algorithm for v-support vector classification. *IEEE Transactions on Neural Networks & Learning Systems, 28*(5), 1241.

Gu, B., Sun, X., & Sheng, V. S. (2017). Structural minimax probability machine. *IEEE*

Transactions on Neural Networks & Learning Systems, 28(7), 1646–1656.
Hou, H. S., & Andrews, H. (1978). Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics Speech & Signal Processing, 26*(6), 508–517.
Jeon, G., Min, Y. J., Anisetti, M., Bellandi, V., Damiani, E., & Jeong, J. (2010). Specification of the geometric regularity model for fuzzy if-then rule-based deinterlacing. *Journal of Display Technology, 6*(6), 235–243.
Jeon, G., Anisetti, M., & Kang, S. H. (2013). A rank-ordered marginal filter for deinterlacing. *Sensors (Basel), 13*(3), 3056–3065.
Katsaggelos, A., Molina, R., & Mateos, J. (2007). Super resolution of images and video. *Synthesis Lectures on Image Video & Multimedia Processing, 3*(1), 682–689.
Keys, R. (2003). Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics Speech & Signal Processing, 29*(6), 1153–1160.
Lehmann, T. M., Gönner, C., & Spitzer, K. (1999). Survey: interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging, 18*(11), 1049.
Liaw, A., & Wiener, M. (2002). Classification and regression with random forest. *R News, 23*(23).
Protter, M., Elad, M., Takeda, H., & Milanfar, P. (2008). Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 18*(1), 36–51.
Pu, Y. F., Zhou, J. L., Siarry, P., Zhang, N., & Liu, Y. F. (2013). Fractional partial differential equation: Fractional total variation and fractional steepest descent approach-Based multiscale denoising model for texture image. *Abstract and Applied Analysis, 2013*(5), 764–787.
Pu, Y. F., Siarry, P., Zhou, J. L., Liu, Y. G., Zhang, N., Huang, G., et al. (2014). Fractional partial differential equation denoising models for texture image. *Science China, 57*(7), 1–19.
Qu, X., Mayzel, M., Cai, J.-F., Chen, Z., & Orekhov, V. (2015). Accelerated nmr spectroscopy with low-rank reconstruction. *Angewandte Chemie. International Ed. In English, 54*(January (3)), 852–854.
Tang, Y., Yan, P., Yuan, Y., & Li, X. (2011). Single-image super-resolution via local learning. *International Journal of Machine Learning & Cybernetics, 2*(1), 15–23.
Timofte, R., De, V., & Gool, L. V. (2014a). Anchored neighborhood regression for fast example-Based super-Resolution. *IEEE international conference on computer vision* (pp. 1920–1927).
Timofte, R., De, V., & Gool, L. V. (2014b). Anchored neighborhood regression for fast example-Based super-Resolution. *IEEE international conference on computer vision* (pp. 1920–1927).
Tropp, J. A., & Gilbert, A. C. (2007). Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory, 53*(12), 4322.
Wang, J., Li, T., Shi, Y. Q., Lian, S., & Ye, J. (2016). Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. *Multimedia Tools & Applications,* 1–17.
Wu, W., Liu, Z., & Xiaohai, H. (2011). Learning-based super resolution using kernel partial least squares. *Image & Vision Computing, 29*(6), 394–406.
Wu, W., Yang, X., Yu Pang, J. P., & Jeon, G. (2013). A multifocus image fusion method by using hidden markov model. *Optics Communications, 287*(1), 63C72.
Yang, J., Wright, J., Huang, T. S., & Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 19*(11), 2861–2873.
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.
Yang, S., Wang, M., Chen, Y., & Sun, Y. (2012). Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding. *IEEE Transactions on Image Processing, 21*(9), 4016–4028.
Yang, X., Wu, W., Liu, K., Zhou, K., & Binyu, Y. (2015). Fast multisensor infrared image super-resolution scheme with multiple regression models. *Journal of Systems Architecture, 64*, 11–25.
Zeyde, R., & Elad, M. (2010). *Matan Protter On single image scale-up using sparse-representations. International conference on curves and surfaces*. Springer-Verlag711–730.
Zhou, Z., Yang, C. N., Chen, B., Sun, X., Liu, Q., & Wu, Q. M. J. (2016). Effective and efficient image copy detection with resistance to arbitrary rotation. *Ieice Transactions on Information & Systems, E99*(D(6)), 1531–1540.
Zhou, Z., Wang, Y., Wu, Q. M. J., Yang, C. N., & Sun, X. (2017). Effective and efficient global context verification for image copy detection. *IEEE Transactions on Information Forensics & Security, 12*(1), 48–63.
Zhou, Z., Wu, Q. J., Huang, F., & Sun, X. (2017). Fast and accurate near-duplicate image elimination for visual sensor networks. *International Journal of Distributed Sensor Networks, 13*(2) [155014771769417].