Hierarchical Regression for Single Image Super Resolution via Clustering and Sparse Representation

Kang Qiu [1], Benshun Yi [1,2]*, Weizhong Li [1] and Taiqi Huang[1]
[1] School of Electronic Information, Wuhan University,
Wuhan 430072, China
[2] Collaborative Innovation Center for Geospatial Technology ,
Wuhan 430079, China
*Corresponding author: Benshun Yi
[e-mail: yibs@whu.edu.cn]

Received October 29, 2016; revised January 17, 2017; accepted February 26, 2017; published May 31, 2017

Abstract

Regression-based image super resolution (SR) methods have shown great advantage in time consumption while maintaining similar or improved quality performance compared to other learning-based methods. In this paper, we propose a novel single image SR method based on hierarchical regression to further improve the quality performance. As an improvement to other regression-based methods, we introduce a hierarchical scheme into the process of learning multiple regressors. First, training samples are grouped into different clusters according to their geometry similarity, which generates the structure layer. Then in each cluster, a compact dictionary can be learned by Sparse Coding (SC) method and the training samples can be further grouped by dictionary atoms to form the detail layer. Last, a series of projection matrixes, which anchored to dictionary atoms, can be learned by linear regression. Experiment results show that hierarchical scheme can lead to regression that is more precise. Our method achieves superior high quality results compared with several state-of-the-art methods.

Keywords: single imaeg super resolution, hierarchical regression, clustering, sparse coding, dictionary learning

https://doi.org/10.3837/tiis.2017.05.013 **ISSN : 1976-7277**

1. Introduction

The goal of image Super-Resolution (SR) is to reconstruct a high-resolution (HR) image from low-resolution (LR) image. In the past decades, many SR algorithms have been proposed and these algorithms can be mainly divided into the following three categories: interpolation-based methods, reconstruction-based methods and learning-based methods.

Interpolation-based methods, such as polynomial-based interpolation methods and edge-directed interpolation methods [13,14,15], estimate unknown pixels through their known neighbors. These methods have low complexity, but tend to produce blurring and jaggy. Reconstruction-based methods often use prior knowledge such as gradient profile prior [16], total variation prior [17] and nonlocal self-similarity prior [18] to constrain the SR process. The prior knowledge is usually represented as some regularization terms. Although these reconstruction-based methods have made some improvement in edge preserving and artifacts suppressing, the parameters for the regularization terms are usually difficult to estimate and the SR performance degrades rapidly when the desired magnification factor is large.

Learning-based methods are based on image patches which are extracted from original images with overlaps. The basic idea is that learning the mapping relationship between low-resolution (LR) image patches and corresponding high-resolution (HR) image patches as priori knowledge through example images. The priori knowledge can be used to estimate HR patches from corresponding input LR patches and then the result HR image can be constructed by positioning the estimated HR patches with their overlaps be averaged. The learning-based methods can be further divided into three categories: neighbor embedding (NE) methods, sparse coding (SC) methods and regression-based methods.

Inspired by manifold learning methods, NE methods[1,2] assume that image patches from LR images and corresponding HR images respectively form a low-dimensional nonlinear manifold with similar local geometry to each other. That means each HR patch can be reconstructed from its neighbors with the same weights as in the LR domain provided that sufficient samples are available. The main downside of these methods is that along with the growing of the number of sample images to improve SR performance, the dictionary of sampled patches can quickly become too big to save and to compute.

Sparse representation has gained great improvement in image processing and pattern recognition and many algorithms, such as l_p-norm $(0 < p < 1)$ [30], $l_{2,1}$-norm [29] and $l_{2,p}$-norm $(0 < p \leq 1)$ [28] based methods, are proposed in recent years. Inspired by the perspective of sparse representation, SC methods [3,4,5] are applied in single image SR based on the assumption that LR patches and corresponding HR patches have same sparse representations if the dictionaries are jointly learned. As an extension, some SC methods [6,7] were proposed without the constraint of representation invariance assumption. Instead, the dictionaries are learned respectively with a mapping function learned simultaneously. The main disadvantage of SC methods is that the computational burden of dictionary learning and sparse coding is still too heavy for practical applications.

In recent years, some regression-based methods are proposed to improve the computational efficiency of SR process while maintaining similar or even higher quality level. Clustering was used in [8,9,10] in training stage to split feature space into numerous subspaces and multiple effective mapping functions were learned for each subspace by linear regression. The anchored neighborhood regression (ANR) method [11] and its improved variant A+ [12] learn

the mapping functions through l_2-norm based sparse representation, which is also known as collaborative representation. The computational complexity of reconstruction process for regression-based methods can be greatly reduced since the mapping functions can be trained offline.

Motivated by previous regression-based methods [8,9,10,11,12], we propose a novel regression-based single image SR method which introduces the concept of hierarchy. First, training samples are grouped into different clusters according to their geometry similarity, which generates the structure layer. Then in each cluster, a compact dictionary can be learned by SC method and the training samples can be furtherly grouped by dictionary atoms to form detail layer. Last, a series of projection matrixes which are anchored to dictionary atoms can be learned by linear regression. The procedure of clustering will bring about a more compact dictionary that makes the dictionary atoms more representative and so the image can be more accurately reconstructed. Experiment results demonstrate that hierarchical scheme can lead to more precise regression. Our method achieves superior high quality results compared with several state-of-the-art methods.

The rest of the paper is organized as follows: related works are briefly introduced in Section 2, our proposed method is presented in detail in Section 3, experiment results are reported in Section 4 and the conclusions are made in Section 5.

2. Related Work

In this section, we will briefly present the mainstream of SC methods and regression-based methods on which our proposed method is based.

We define I_h and I_l to be vectors of length N_h and N_l pixels that respectively represent high- and low-resolution images. The relation between I_h and I_l can be formulated as

$$I_l = DBI_h + \nu \tag{1}$$

where B: $R^{N_h} \to R^{N_h}$ is a blur operator , D: $R^{N_h} \to R^{N_l}$ is a decimation operator and ν is an additive Gaussian Noise denoted as $\nu \sim N(0, \sigma^2 I)$. The image SR problem is about how to find an estimation $\hat{I}_h$ closest to I_h from given observed measurement I_l.

As a feasible solution to this problem, SC methods assume that each patch pair from HR and LR image can be sparsely represented respectively by HR and LR dictionaries using the same sparse coefficient when the dictionaries are jointly learned. In literature [3], the dictionaries are learned by simply randomly sampling raw patches from training images of similar statistical nature. As an improvement, Zeyde et al. [5] apply K-SVD [22] for LR dictionary training and obtain HR dictionary by solving the following problem

$$\min_{D_h} \left\| X - D_h A \right\|_F^2 \tag{2}$$

where $\left\| \cdot \right\|_F$ is the Frobenius norm, matrix X contains HR training patches as its columns and matrix A is constructed by sparse representation vectors, which are obtained while training the LR dictionary, as its columns. The solution of (2) can be given by

$$D_h = XA^{\dagger} = XA^T (AA^T)^{-1} \tag{3}$$

In the testing phase, for each input LR patch, the sparse representation can be found through the following formulation

$$\min_{\alpha} \left\| D_l \alpha - y \right\|_2^2 + \lambda \left\| \alpha \right\|_0 \tag{4}$$

where D_l denotes learned LR dictionary, y denotes LR patches or features extracted from LR image, α denotes the sparse coefficient and λ is a small positive constant used to balance sparsity of the solution and fidelity of the approximation to y. The optimization problem (4) is a NP-hard problem and the solution is difficult to approximate because of the l_0 –norm regularization term. However, the literatures [19,20,21] have revealed that l_0 –norm constraint of problem (4) can be relaxed to l_1 –norm constraint while the solution is still content with the condition of sparsity. So problem (2) can be reformulated as

$$\min_{\alpha} \|D_l\alpha - y\|_2^2 + \lambda\|\alpha\|_1 \tag{5}$$

and this problem can be solved in polynomial time. Given the optimal solution $\hat{\alpha}$ to (5), the corresponding HR patch can be reconstructed by the following formulation

$$x = D_h\hat{\alpha} \tag{6}$$

where x denotes output HR patch and D_h denotes HR dictionary.

The solving of problem (5) with l_1–norm regularization term is much computationally demanding, so further improvements were proposed in [11,12] by using Collaborative Representation [23] and the problem was formulated as

$$\min_{\alpha} \|N_l\alpha - y\|_2^2 + \lambda\|\alpha\|_2 \tag{7}$$

where N_l denotes the neighborhood of input patch y in LR space. The l_2 -norm regularization term, which replaces the l_1 -norm regularization term in (5), is mainly used to make the least square solution stable and make sure the sparsity of the solution. The solution of (7) can be easily and analytically derived as

$$\hat{\alpha} = (N_l^T N_l + \lambda I)^{-1} N_l^T y \tag{8}$$

and the corresponding HR patch can be obtained by

$$\hat{x} = N_h\hat{\alpha} = N_h(N_l^T N_l + \lambda I)^{-1} N_l^T y \tag{9}$$

where N_h is neighborhood in HR space corresponding to N_l. In [11,12], (N_h, N_l) pairs are anchored to dictionary atoms, that is, given an LR patch y, its neighborhoods are selected by the dictionary atom which is nearest to y. So the projection matrixes are independent of y and can be pre-calculated in training phase as

$$P_j = N_h^j((N_l^j)^T N_l^j + \lambda I)^{-1}(N_l^j)^T,\ j = 1,2,\cdots M \tag{10}$$

where (N_h, N_l) is neighborhood pair anchored to the j-th atom and M is the dictionary size. Then in testing phase, input LR patch y can be directly projected to HR space via $P_j y$ just after selecting the proper projection matrix by comparing the distance between y and each dictionary atom and this can be much faster than previous learning-based methods. In [11], the neighborhood pairs anchored to dictionary atoms are represented by their k nearest neighbors in the dictionary, which makes the size of neighborhood be limited by the dictionary size. Instead, [12] represents each neighborhood straightly from full training material, similar to [8], and obtains improved quality.

3. Proposed Method

In this section, we propose a novel regression-based single image SR method, which implement a hierarchical structure via clustering and sparse representation to learn a series of projection matrixes more precisely for fast single image SR. To be specific, we split image patches which are extracted from training image set into numerous clusters, followed by jointly learning sparse dictionaries in each cluster domain. Then a series of projection matrixes, which are anchored to dictionary atoms, are learned for image SR. More details will be described in following discussion.

3.1 Clustering and Learning Sub-dictionaries

In this step, we first cluster training image patches into different groups according to their geometry similarity since it has been revealed that patches with similar patterns will bring about a more compact dictionary and so the image can be more accurately reconstructed [24]. Before clustering, we need extract patches from training image set. As in [5], we operate with image patches in feature space for robustness rather than straightly using raw patches. To obtain desired patches, we firstly carry out some preprocessing on full training images. Let $\left\{I_{l/h}^{i}\right\}_{i=1}^{N_I}$ denote LR/HR training image set and $\left\{I_{B}^{i}\right\}_{i=1}^{N_I}$ denote upscaling result of $\left\{I_{l}^{i}\right\}_{i=1}^{N_I}$ using bicubic interpolation algorithm (N_I is the number of all training images). Then we remove low frequency redundancies from HR image by

$$I_M^i = I_h^i - I_B^i, i = 1, 2, \cdots, N_I \tag{11}$$

so as to focus on edge and texture content which conveys more semantic information for human vision. For the same reason, we aim to extract high frequency details of LR image using

$$I_{f_j}^i = f_j * I_B^i, i = 1, 2, \cdots, N_I, j = 1, 2, 3, 4 \tag{12}$$

where $*$ denotes convolution operation, f_1 and f_2 represent one-order gradient filters in the horizontal and vertical directions, f_3 and f_4 represent two-order gradient filters in the horizontal and vertical directions, $I_{f_1}^i, ...I_{f_4}^i$ respectively represent the result image filtered by $f_1, ... f_4$.

Then we extract LR and HR patches from preprocessed images. Let $\left\{x_i\right\}_{i=1}^{N}$ denotes HR patches and $\left\{y_i\right\}_{i=1}^{N}$ denotes LR patches in feature space (N is number of all patches extracted from LR/HR images). HR patches $\left\{x_i\right\}_{i=1}^{N}$ are vectors of length n_h obtained by vectorising image patches of $\sqrt{n_h} \times \sqrt{n_h}$ pixels which are extracted from $\left\{I_M^i\right\}_{i=1}^{N_I}$. For each HR patch $x_i (i = 1, 2, \cdots, N)$, four LR patches $\left\{p_1^i, p_2^i, p_3^i, p_4^i\right\}$ with the same size can be extracted from $I_{f_j}^i$ $(j = 1, 2, 3, 4)$ at the same location. After that, the corresponding LR patch y_i can be obtained from merging these four patches into one patch followed by a dimensionality reduction operation using Principal Component Analysis (PCA) algorithm to reduce computational burden since a large number of redundant information exists in merged patches. The length of y_i is supposed to be n_l.

Image patch pairs $\{x_i, y_i\}_{i=1}^{N}$ have been extracted from training images after the procedures above. Then we use k-means clustering algorithm on $\{y_i\}_{i=1}^{N}$ to split these patches into a fixed number of clusters. Here, we use Euclidean Distance as distance metric in k-means algorithm since it is simple and effective in our application. After that, the number of patches in some clusters are obviously less than others. The clusters with too less patches will result in overfitting after dictionary learning process and so lead to inaccurate regression. To overcome this problem, we merge these clusters with their nearest neighboring clusters.The initial clusters are compressed to K clusters with K centroids are recomputed. Let $\{c^k\}_{k=1}^{K}$ denote K centroids of these clusters and $Y^k = \{y_i\}_{i\in\Omega_k}$ is the LR patches assigned to the k-th cluster, where Ω_k is the index set of the k-th cluster from $\{y_i\}_{i=1}^{N}$. We split HR patches $\{x_i\}_{i=1}^{N}$ into K clusters $X^k (k = 1, 2, \cdots, K)$ with the same indices as Y^k and K clusters of HR/LR patch pairs $\{X^k, Y^k\}_{k=1}^{K}$ are established.

Once clustered image patch pairs are established, we come to the procedure of jointly learning sparse dictionaries in each $\{X^k, Y^k\}_{k=1}^{K}$ domain. Similar to [25], for the k-th cluster $\{X^k, Y^k\}$, the coupled HR/LR dictionaries can be obtained by solving the following optimization problem:

$$\min_{D_l^k, D_h^k, A^k} \frac{1}{n_l}\left\|D_l^k A^k - Y^k\right\|_F^2 + \frac{1}{n_h}\left\|D_h^k A^k - X^k\right\|_F^2 \text{ s.t. } \left\|\alpha_i^k\right\|_0 \le d, i = 1, 2, \cdots, \left|\Omega_k\right| \quad (13)$$

where $\{D_l^k, D_h^k\} \in \{R^{n_l \times M}, R^{n_h \times M}\}$ denotes coupled LR/HR dictionaries of the k-th cluster of training patches, $A^k \in R^{M\times N}$ denotes the matrix of coefficients, α_i^k is the i-th column vector of A^k and d is the limit of sparsity. Here M is size of the dictionary and $|\cdot|$ represents cardinality of a set. The problem can be reformulated as

$$\min_{D^k, A^k} \left\|D^k A^k - S^k\right\|_F^2 \text{ s.t. } \left\|\alpha_i^k\right\|_0 \le d, i = 1, 2, \cdots, \left|\Omega_k\right| \quad (14)$$

with

$$D^k = \begin{bmatrix} \frac{1}{\sqrt{n_l}} D_l^k \\ \frac{1}{\sqrt{n_h}} D_h^k \end{bmatrix}, S^k = \begin{bmatrix} \frac{1}{\sqrt{n_l}} Y^k \\ \frac{1}{\sqrt{n_h}} X^k \end{bmatrix} \quad (15)$$

This is a general dictionary learning problem which can be solved using K-SVD algorithm and the coupled dictionaries $\{D_l^k, D_h^k\}$ can be obtained from D^k.

3.2 Learning Projection Matrixes

In previous section above, we have established a series of clusters consisted of LR/HR patches with similar geometry structure and learned coupled dictionaries for each cluster. In this section, we aim to learn projection matrixes anchored to dictionary atoms in each cluster to

directly project LR patch to HR space in testing phase. We denote P_j^k as the projection matrix anchored to the j-th dictionary atom of the k-th cluster. First, we group the training patches in the k-th cluster into M subsets that the j-th $\left(j=1,2,\cdots,M\right)$ subset contains the patches to which the j-th dictionary atom has the highest correlation among the whole dictionary. Let the j-th subset is $X_j^k=\left\{x_i\right\}_{i\in\Omega_k^j}$ and $Y_j^k=\left\{y_i\right\}_{i\in\Omega_k^j}$ where Ω_k^j denotes index set of the j-th subset of Y^k. Then the problem of finding a projection matrix P_j^k can be formulated as

$$\min_{P_j^k}\sum_{i\in\Omega_k^j}\left\|x_i-P_j^k y_i\right\|_2^2, k=1,2\cdots,K,\ j=1,2,\cdots,M \tag{16}$$

This can be reformulated as

$$\min_{P_j^k}\sum\left\|X_j^k-P_j^k Y_j^k\right\|_F^2, k=1,2\cdots,K,\ j=1,2,\cdots,M \tag{17}$$

The solution of (17) can be given by

$$\hat{P}_j^k=X_j^k(Y_j^k)^{\dagger}=X_j^k(Y_j^k)^T(Y_j^k(Y_j^k)^T)^{-1} \tag{18}$$

In this way, we can learn a $K\times M$ matrix with $\hat{P}_j^k(k=1,2,\cdots,K,\ j=1,2,\cdots,M)$ as its elements and the learning process can also be finished offline without increasing computational burden to testing phase.

3.3 Reconstruction Process

In training phase, we have established K training clusters with $\left\{c^k\right\}_{k=1}^K$, $\left\{D_l^k,D_h^k\right\}_{k=1}^K$ as their centroids and sparse dictionaries and we have learned a $K\times M$ matrix consisted of projection matrixes which are anchored to dictionary atoms. In testing phase, we first extract feature patches from given input LR image using the method described in section 3.1. After then, each LR feature patch can be located to a cluster centroid by comparing the distances between the patch and the centroids $\left\{c^k\right\}_{k=1}^K$ and it can be furtherly located to a dictionary atom by comparing the relevancy between l_2-normalized LR feature patch and the dictionary atoms. Then we can select a projection matrix P_j^k from the learned $K\times M$ matrix with the located index and the corresponding HR patch can be obtained by directly multiplying the LR feature patch by P_j^k.

In this way, we can obtain all HR patches corresponding to LR feature patches. Then the HR image can be reconstructed by positioning these result HR patches at corresponding location with their overlaps be averaged.

4. Experimental Results

In this section, we carry out a set of experiments to evaluate performance of our proposed method. We compare our proposed method with Bicubic interpolation method and several state-of-the-art learning based SR methods such as NE+LLE [1], SCSR [25], the Zeyde's method [5], ANR [11], A+ [12] and SRCNN [27]. The peak signal-to-noise ratio (PSNR) and

structural similarity (SSIM) [26] are used as main objective quality evaluation indexes for SR performance evaluation.

4.1 Experiment Settings

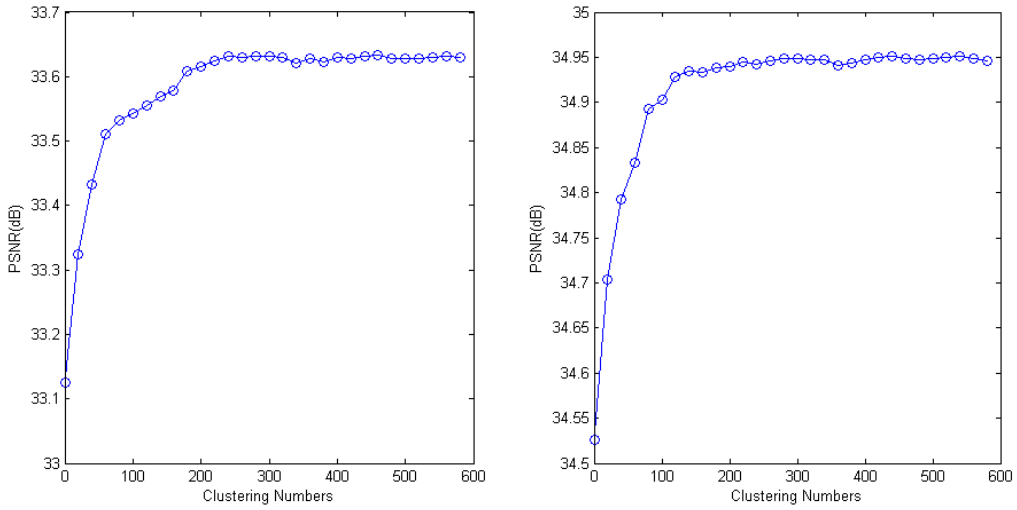
We select 16 images of representative scene as shown in **Fig. 1** to carry out our SR experiments. The objective indexes for performance evaluation are computed in the luminance channel of original and reconstructed HR images.

Fig. 1. Test 16 images. From left to right and top to down: coffee, sculpture, metal, hands, balls, bird, travelling, bedroom, boy, flower, indicator, cable stripe, model, car, boat.

Our proposed method is trained with 91 images provided in [25]. The training images are converted into YCbCr color space and LR/HR training patches are extracted from luminance component Y since human visual system is much more sensitive to intensity changes than to color changes. The patch size is set to 9×9, so the length of HR feature vectors $n_h = 81$. The size of LR feature vectors is $4\times81=324$ and it is reduced to $n_l = 30$ after using PCA algorithm. In the training phase, $N = 800,000$ LR/HR feature pairs are randomly extracted from multiple scales (12 scales) of training images.

The number of clusters is important for performance of our proposed method. If the number is too small, the training patches in each cluster will have less structure similarity, and then the precision of regression will reduce. If the number is too large, the time consumption of locating to cluster centroid in testing phase will increase. To set clustering number properly, we train our method with different clustering numbers from 20 to 580 at intervals of 20. Then we do SR reconstructions over two testing images (indicator and travelling) and the results are shown in **Fig. 2**. The results show that when set clustering number between 1 and 200, reconstruction performance in terms of PSNR grows rapidly along with the increasing of clustering number. When clustering number increase to larger than 200, the performance remains stable with very little fluctuation. Here we set clustering number to 300 and the training patch pairs are clustered by k-means method. Then the number of clusters is compressed to $K = 238$ by merging the clusters with little patches (e.g. less than 3000 patches) into their nearest neighboring clusters. For each cluster, a sparse dictionary is learned by K-SVD algorithm. The dictionary size is set to $M = 1024$ and the limit of sparsity is set to $d = 3$. The magnification factor is set to $s = 3$.

All the compared learning based methods are also trained with the same training dataset for a fair comparison. The source codes are obtained from the authors' websites and the parameters are set according to the authors' recommendation.



(a) (b)

Fig. 2. Reconstruction performance in terms of PSNR with different clustering numbers. (a) the results over "indicator" (b) the results over "travelling".

4.2 Results and Analysis

We compare our proposed method with other state-of-the-art methods in terms of PSNR and SSIM respectively in **Table 1** and **Table 2**. Results show that our method obtains the best results for most of the testing images and achieves best results on average. The average PSNR gain of our proposed method over the second best method SRCNN [27] is 0.2588dB and the average SSIM gains of our proposed method on the second best method SRCNN [27] is 0.0040. These quantitative results demonstrate that our proposed method not only obtains smaller reconstruction error but also preserves better structural details than the other methods.

Table 1. Performance of $\times 3$ magnification in terms of PSNR (dB) on testing images. Best results in bold.

image	Bicubic	NE+LLE	SCSR	Zeyde	ANR	A+	SRCNN	Proposed
coffee	37.0438	38.5811	38.1036	38.3947	38.5564	39.4102	38.9855	**39.6196**
sculpture	26.0318	27.0121	27.2202	27.0133	27.0556	27.8219	27.7880	**27.9675**
metal	39.9263	42.7315	40.2590	42.8706	42.7453	43.7466	43.9355	**44.0996**
hands	33.9495	34.6234	34.1238	34.8985	34.6943	35.8155	35.8796	**36.0672**
balls	32.0054	33.8614	33.6078	33.9611	33.9514	35.0822	34.6007	**35.2374**
bird	30.1967	31.1995	30.6348	31.2376	31.2210	31.2666	**31.5551**	31.5249
travelling	32.1348	33.8787	33.3755	34.0614	33.9097	34.6904	34.6570	**34.9488**
bedroom	33.9631	35.8294	34.5874	35.8776	35.8326	35.5937	35.8539	**35.9551**
boy	39.1302	41.8170	40.0860	42.0290	41.9043	42.3609	42.3550	**42.5935**
flower	39.1534	40.8519	39.3268	40.7429	40.9110	41.5750	41.6288	**41.7196**
indicator	29.2828	31.5650	30.6459	32.0194	31.5942	33.2880	33.5718	**33.6316**
cable	27.4625	29.5415	30.1281	29.5261	29.5705	31.6945	32.0629	**32.1204**
stripe	27.8318	28.2772	28.3095	28.7943	28.3243	30.4701	30.2119	**30.8951**
model	25.5363	26.5819	26.7202	27.0268	26.6057	28.8418	28.7531	**29.0711**
car	28.3917	29.4651	29.3104	29.6262	29.4140	30.2529	30.0467	**30.4222**
boat	26.5110	27.4641	27.1823	27.7153	27.5282	28.2400	28.2919	**28.4440**
average	31.7845	33.3301	32.7263	33.4872	33.3637	34.3844	34.3861	**34.6449**

Table 2. Performance of $\times 3$ magnification in terms of SSIM on testing images. Best results in bold.

image	Bicubic	NE+LLE	SCSR	Zeyde	ANR	A+	SRCNN	Proposed
coffee	0.9591	0.9680	0.9614	0.9675	0.9684	0.9720	0.9679	**0.9729**
sculpture	0.8504	0.8776	0.8783	0.8785	0.8781	0.8983	**0.9033**	0.9007
metal	0.9869	0.9900	0.9793	0.9908	0.9905	0.9920	0.9924	**0.9928**
hands	0.9222	0.9317	0.9147	0.9325	0.9325	0.9364	0.9293	**0.9377**
balls	0.9175	0.9362	0.9269	0.9376	0.9368	0.9458	0.9445	**0.9466**
bird	0.8282	0.8540	0.8389	0.8543	0.8550	0.8543	**0.8585**	0.8581
travelling	0.9135	0.9348	0.9253	0.9360	0.9354	0.9421	0.9417	**0.9438**
bedroom	0.9458	0.9597	0.9482	0.9607	0.9607	0.9600	**0.9631**	0.9615
boy	0.9667	0.9757	0.9640	0.9764	0.9763	0.9775	**0.9790**	0.9782
flower	0.9647	0.9723	0.9574	0.9722	0.9727	0.9741	**0.9748**	0.9747
indicator	0.8976	0.9227	0.9069	0.9298	0.9221	0.9423	0.9432	**0.9448**
cable	0.9071	0.9301	0.9325	0.9315	0.9284	0.9513	0.9536	**0.9546**
stripe	0.7786	0.7866	0.7865	0.8170	0.7917	0.8645	0.8573	**0.8749**
model	0.7962	0.8200	0.8081	0.8307	0.8197	0.8590	0.8493	**0.8603**
car	0.8830	0.9041	0.8984	0.9078	0.9023	0.9194	0.9039	**0.9218**
boat	0.8250	0.8483	0.8424	0.8545	0.8509	0.8659	0.8677	**0.8692**
average	0.8964	0.9132	0.9043	0.9174	0.9138	0.9284	0.9268	**0.9308**

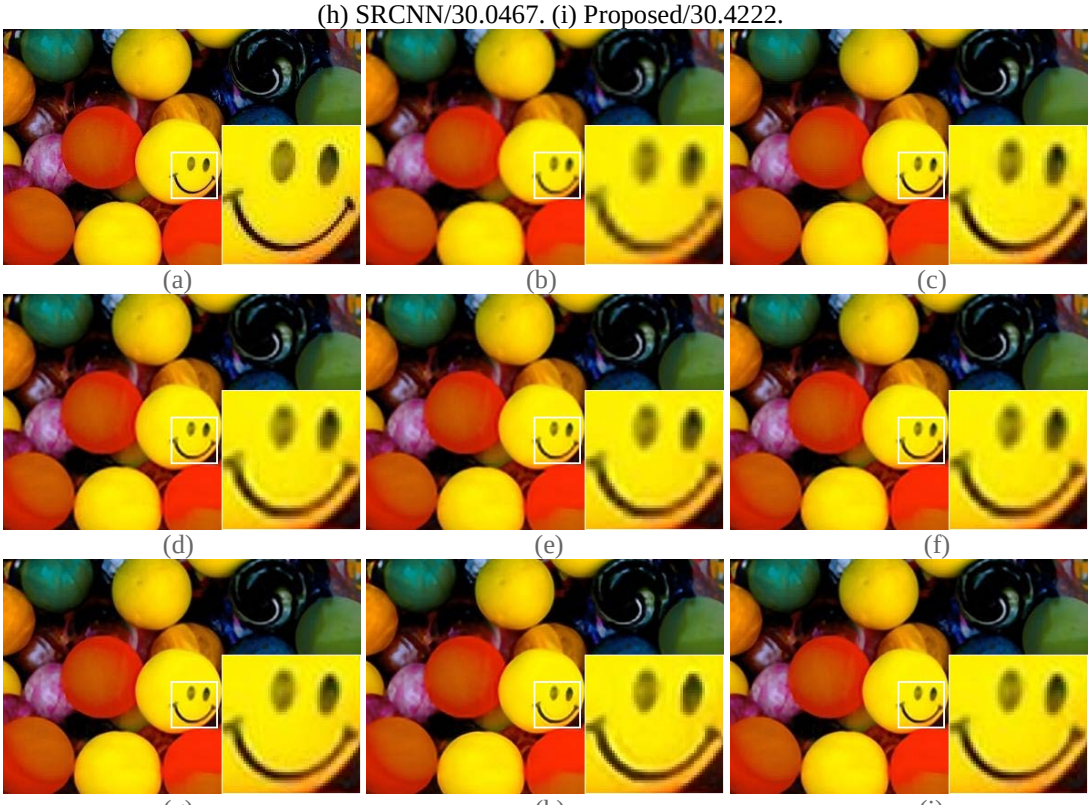
To further demonstrate the effectiveness of our proposed method, we compare our visual results with other state-of-the-art methods with magnification factor s = 3 in **Fig. 3, Fig. 4, Fig. 5** and **Fig. 6**. We select these images based on the variety of the scenes including car (see **Fig. 2**), balls (see **Fig. 3**), travelling (see **Fig. 4**) and indicator (see **Fig. 5**). The visual results show that the Bicubic interpolation produces the worst results with seriously blurring effects along the edges and over-smoothed textures. Although NE + LLE [1] can alleviate the blurring effects along the edges by partially reconstructing the high frequency components of the HR images, it still tends to produce ringing effects along edges by introducing inaccurate neighbors. SCSR [25], Zeyde's method [5] also generates ringing effects and blurred results mainly because that learned dictionaries fail to capture the details from training patches. ANR [11] would achieve very fast SR but still generates some unpleasing artifacts in **Fig. 4**(f) and blurred edges in **Fig. 5**(f). A+ [12], SRCNN [27] suppresses ringing effects significantly, however, it also produces jaggy artifacts in **Fig. 5**(g), **Fig. 6**(g) and smoothing details in **Fig. 3**(g), **Fig. 4**(g). As can be observed, our proposed method produces sharper edges without obvious ringing effects, less blurring effects, and finer textures with more details than other methods.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 3. Visual comparisons with different SR results on car by using different methods (magnification factor s = 3) (zoom in for better view). (a) Original/PSNR. (b) Bicubic/28.3917. (c) NE + LLE/29.4651. (d) SCSR/29.3104. (e) Zeyde/29.6262. (f) ANR/29.4140. (g) A+/30.2529. (h) SRCNN/30.0467. (i) Proposed/30.4222.

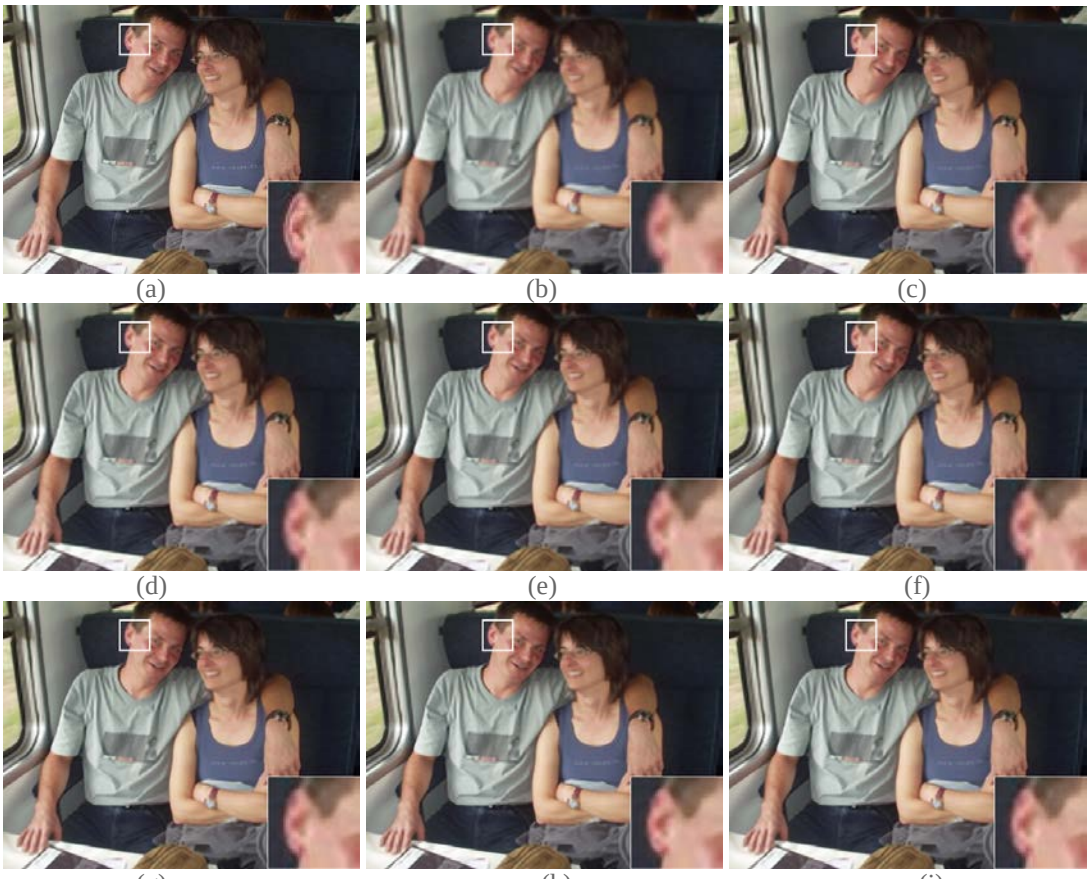


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 4. Visual comparisons with different SR results on balls by using different methods (magnification factor s = 3) (zoom in for better view). (a) Original/PSNR. (b) Bicubic/32.0054. (c) NE + LLE/33.8614. (d) SCSR/33.6078. (e) Zeyde/33.9611. (f) ANR/33.9514. (g) A+/35.0822. (h) SRCNN/34.6007. (i) Proposed/35.2374.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 5. Visual comparisons with different SR results on travelling by using different methods (magnification factor s = 3) (zoom in for better view). (a) Original/PSNR. (b) Bicubic/32.1348. (c) NE + LLE/33.8787. (d) SCSR/33.3755. (e) Zeyde/34.0614. (f) ANR/33.9097. (g) A+/34.6904. (h) SRCNN/34.6570. (i) Proposed/34.9488.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 6. Visual comparisons with different SR results on indicator by using different methods (magnification factor s = 3) (zoom in for better view). (a) Original/PSNR. (b) Bicubic/29.2828. (c) NE + LLE/31.5650. (d) SCSR/30.6459. (e) Zeyde/32.0194. (f) ANR/31.5942. (g) A+/33.2880. (h) SRCNN/33.5718. (i) Proposed/33.6316.

4.3 Running Time

We analyze the computing complexity of our proposed method using big O notation. As the hierarchical regression model can be trained offline, so we focus on analyzing the complexity of testing phase. It is simple to conclude that the cost for locating an LR patch to a cluster is $O(Kn_l)$, the cost for furtherly locating to final projection matrix is $O(Mn_l)$ and the cost for projecting an LR patch to corresponding HR patch $O(n_h n_l)$. K, M, n_h, n_l respectively represent the number of clusters, dictionary size and the dimension of HR and LR patch. So the whole cost is $O(N(K+M+n_h)n_l)$. N denotes the total number of patches extracted from original LR image.

Table 3 shows the average running time of our proposed method and other state-of-the-art methods on 16 testing images. All the experiments are conducted on an Intel(R) Core(TM) i7-5600U @ 2.60 GHz CPU with 16 GB RAM under MATLAB R2013a programming environment. Results show that the running time of our proposed method is more than ANR [11] and A+ [12] mainly because that our proposed method increases an extra procedure of locating to cluster centroid after feature extraction. This is reasonable cost for quality improvement. Even so, our proposed method still consumes less time than the other learning-based methods while achieving state-of-the-art SR quality.

Table 3. Average running time of different methods.

Methods	NE+LLE	SCSR	Zeyde	ANR	A+	SRCNN	Proposed
Times(s)	2.86	25.63	1.86	0.70	0.75	8.63	1.22

5. Conclusion

In this paper, a novel single image SR method based on hierarchical regression was proposed to further improve the performance of regression-based methods. The hierarchical scheme introduced here groups the training dataset into structure layer and detail layer, which leads to learning of more compact dictionaries and thus more precise regression. Bicubic interpolation method and 6 state-of-the-art learning based SR methods were compared with our proposed method in terms of PSNR and SSIM. The experimental results show that our proposed method have the best overall quality compared to other state-of-the-art methods at the cost that the running time increases compared to [11] and A+ [12], but still much faster than the other learning based methods.

References

[1] H. Chang, D.-Y. Yeung and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp.275-282, June 27- July 2, 2004. Article (CrossRef Link).

[2] M. Bevilacqua, A. Roumy, C. Guillemot and M.-L. Alberi Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. of British Machine Vision Conference*, pp.135.1-135.10, September 3-7, 2012. Article (CrossRef Link).

[3] J. Yang, J. Wright, T. Huang and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp.1-8, June 23-28, 2008. Article (CrossRef Link).

[4] J. Yang, Z. Wang, Z. Lin, S. Cohen and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp.3467–3478, August, 2012. Article (CrossRef Link).

[5] R. Zeyde, M. Elad and M. Protter, "On single image scale-up using sparse-representations," in *Proc. of 7th Int. Conference on Curves and Surfaces*, pp.711–730, June 24-30, 2010. Article (CrossRef Link).

[6] L. He, H. Qi and R. Zaretzki, "Beta process joint dictionary learning for coupled feature spaces with application to single image super resolution," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp.345–352, June 23-28, 2013. Article (CrossRef Link)

[7] W. Yang, Y. Tian and F. Zhou, "Consistent coding scheme for single-image super-resolution via independent dictionaries," *IEEE Transactions on Multimedia*, vol. 18, no. 3, pp.312-325, March, 2016. Article (CrossRef Link).

[8] C.-Y. Yang and M.-H. Yang, "Fast direct super-resolution by simple functions," in *Proc. of IEEE International Conference on Computer Vision*, pp.561–568, Dec 1-8, 2013. Article (CrossRef Link).

[9] Y. Zhang, Y. Zhang and J. Zhang, "CCR-Clustering and collaborative representation for fast single image super-resolution," *IEEE Transactions on Multimedia*, vol. 18, no. 3, pp.405-417, March, 2016. Article (CrossRef Link).

[10] K. Zhang, D. Tao, X. Gao, X. Li and Z. Xiong, "Learning multiple linear mappings for efficient single image super-resolution," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp.846–861, March, 2015. Article (CrossRef Link).

[11] R. Timofte, V. D. Smet and L.V. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Proc. of IEEE International Conference on Computer Vision*, pp.1920–1927, December 1-8, 2013. Article (CrossRef Link).

[12] R. Timofte, V. D. Smet and L. V. Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Proc. of 12th Asian Conference on Computer Vision*, pp.111-126, November 1-5, 2014. Article (CrossRef Link).

[13] K.-W. Hung and W.-C. Siu, "Fast image interpolation using bilateral filter," *IET Image Processing*, vol. 6, no. 7, pp.877-890, October, 2012. Article (CrossRef Link).

[14] K.-W. Hung and W.-C. Siu, "Robust soft-decision interpolation using weighted least squares," *IEEE Transactions on Image Processing*, vo1. 21, no. 3, pp.1061-1069, March, 2012. Article (CrossRef Link).

[15] W.-S. Tam, C.-W. Kok and W.-C. Siu, "A modified edge directed interpolation for images," *Journal of Electronic Imaging*, vol. 19, no. 1, pp.013011-013011-20, March 2010. Article (CrossRef Link).

[16] J. Sun, J. Sun, Z. Xu and H.-Y. Shum, "Gradient profile prior and its applications in image super-resolution and enhancement," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp.1529-1542, June, 2011. Article (CrossRef Link).

[17] A. Marquina and S.J. Osher, "Image super-resolution by TV-regularization and Bregman iteration," *Journal of Scientific Computing*, vol. 37, no. 3, pp.367–382, December, 2008. Article (CrossRef Link).

[18] K. Zhang, X. Gao, D. Tao and X. Li, "Single image super-resolution with non-local means and steering kernel regression," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp.4544–4556, November, 2012. Article (CrossRef Link).

[19] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp.797–829, March, 2006. Article (CrossRef Link).

[20] E. J. Candès, J. K. Romberg and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp.1207–1223, March, 2006. Article (CrossRef Link).

[21] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp.5406–5425, December, 2006. Article (CrossRef Link).

[22] M. Aharon, M. Elad and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp.4311-4322, November, 2006. Article (CrossRef Link).

[23] L. Zhang, M. Yang and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *Proc. of IEEE International Conference on Computer Vision*, pp.471-478, November, 6-13, 2011. Article (CrossRef Link).

[24] W. Dong, D. Zhang, G. Shi and X. Wu, "Image deblurring and super resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp.1838–1857, July, 2011. Article (CrossRef Link).

[25] J. Yang, J. Wright, T. Huang and Y. Ma., "Image super resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp.2861–2873, November, 2010. Article (CrossRef Link).

[26] Z. Wang, A. Bovik, H. Sheikh and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp.600-612, April, 2004. Article (CrossRef Link).

[27] C. Dong, C. Loy, K. He and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp.295–307, February, 2016. Article (CrossRef Link).

[28] Z. Li and J. Tang, "Unsupervised feature selection via nonnegative spectral analysis and redundancy control," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp.5343-5355, December, 2015. Article (CrossRef Link).

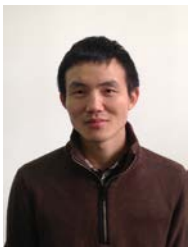
[29] Z. Li, J. Liu, J. Tang and H. Lu, "Robust structured subspace learning for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 10, pp.2085-2098, October, 2015. Article (CrossRef Link).

[30] F. Cao, M. Cai, Y. Tan and J. Zhao, "Image super-resolution via adaptive lp(0 < p < 1) regularization and sparse representation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 7, pp.1550 – 1561, July, 2016. Article (CrossRef Link).

Kang Qiu received the B.S degree in computer science and the M.S degree in electronic information engineering from Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2014, respectively. He is now pursuing the Ph.D. degree in Wuhan University, Wuhan, China. His research interests include multimedia communication and video compression.

Benshun Yi received the B.S, M.S and Ph.D. degrees in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 1986, 1989 and 1996, respectively. He is a Professor of Electronic Information school of Wuhan University, Wuhan, China. His research interests fall in the general areas of multimedia network communication, wireless network, and channel coding and he has published extensively in these areas.

Weizhong Li received the B.S degree in electronic information engineering and the M.S degree in communication and information system from China University of Geosciences, Wuhan, China, in 2005 and 2008, respectively. He is now pursuing the Ph.D. degree in Wuhan University, Wuhan, China. His research interests include multimedia communication and video compression.



Taiqi Huang received his B.S degree in electronic information school of Wuhan University in 2012 and he is pursuing the Ph.D. degree at Wuhan University, Wuhan, China. His research interests are in the area of channel coding and multimedia communication.