Signal Processing: Image Communication 67 (2018) 79–89

Contents lists available at ScienceDirect

Signal Processing: Image Communication

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

Sparse representation and adaptive mixed samples regression for single image super-resolution

Chaopeng Zhang [a], Weirong Liu [a,*], Jie Liu [b], Chaorong Liu [c], Changhong Shi [a]

[a] College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China
[b] National Demonstration Center for Experimental Electrical and Control Engineering Education, Lanzhou University of Technology, Lanzhou 730050, China
[c] Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou 730050, China

ARTICLE INFO

Keywords:
Adaptive mixed samples
Ridge regression
Sparse representation
Super-resolution

ABSTRACT

The example-based super-resolution (SR) methods can be mainly categorized into two classes: the internal SR methods and the external SR methods. The internal SR methods only use samples obtained from a single low resolution (LR) input, while the external SR methods only utilize an external database. The complementary information included in internal and external samples is rarely taken into account. This paper presents a novel extraction and learning method about the complementary information between external samples and internal samples, and then the learned complementary information is used to improve the single image SR performance. Firstly, we construct an initial high resolution (HR) image via sparse coding over the learned dictionary pair with external samples. Secondly, we propose an adaptive sample selection scheme (ASSS) to acquire the mixed samples. Thirdly, we present a novel adaptive mixed samples ridge regression (AMSRR) model to effectively learn the complementary information included in the mixed samples. Finally, we optimize the SR image. Extensive experimental results validate the effectiveness of the proposed algorithm comparing with the state-of-the-art methods.

1. Introduction

Image super-resolution is a technique that increases the spatial resolution from a LR image or a series of LR images. The earliest SR method can be dated back to 1980s [1] by signal processing techniques. Since image SR problem has profound influence on various fields, such as satellite imaging and medical imaging, researchers have proposed a wide range of methods in the last three decades. The existing SR methods can be roughly categorized into multi-frame super-resolution [2–4] and single image super-resolution. Approaches of single image super-resolution can be further classified into interpolation-based methods [5,6], reconstruction-based methods [7] and example-based methods [8–15]. In this paper, we focus on example-based single image super-resolution. The classical example-based SR methods aim at learning the mapping relationship between LR and HR image patches from a preselected database. According to the different resources of the database, the example-based image SR methods can also be classified into external SR methods and internal SR methods. The external SR methods usually utilize a gigantic database collected from multifarious images. The internal SR methods focus on database generated from the input image exploiting self-similar examples. However, both external samples and internal samples have inherent defects. With respect to external samples, the mapping learned from an image of cityscape may not be suitable to reconstruct an image of nature scenery. While the internal samples tend to be insufficient, only fewer information can be transmitted to reconstruction. Moreover, the generation process of internal samples may introduce unsuitable patches. The following is the detailed discussion about external and internal SR methods.

In order to learn the mapping between the LR and HR samples, a representative method proposed by Yang et al. [8] exploited an external database to train a LR and HR dictionary pair via sparse coding. With the assumption that the LR input and its HR output shares the same sparse coefficients, a HR image can be recovered by the sparse coefficients of the LR input and the trained HR dictionary. A new mapping learning method proposed by Wang et al. [11] simultaneously learned a dictionary pair and a mapping function. The dictionary pair reveals the structural domain, while the mapping function reveals the intrinsic relationship between the LR and HR images. Zeyde et al. [9] provided a dictionary pair learning method via K-singular value decomposition (SVD), and then they reconstructed the HR image using the orthogonal matching pursuit (OMP) algorithm. To allow the

* Corresponding author.
E-mail address: liu_weirong@163.com (W. Liu).

https://doi.org/10.1016/j.image.2018.06.001
Received 20 April 2017; Received in revised form 4 June 2018; Accepted 4 June 2018
Available online 15 June 2018

same sparsity corresponding to the same dictionary atoms but different values in the coupled feature spaces, He et al. [14] employed a beta process model. Methods in [8,9,11] and [14] essentially concentrate on changes in dictionary learning. Moreover, the utilization of various prior knowledge is an active research area. Dong et al. [10] introduced the adaptive sparse domain selection (ASDS) scheme into the SR and exploited two adaptive regularization terms (i.e. the autoregressive (AR) models and the nonlocal (NL) self-similarity regularization term). Zhang et al. [12] took a high-order Markov random field (MRF) model as the prior of nature images. They proposed the Markov chain Monte Carlo-based sampling algorithm to estimate the HR image. Fang et al. [16] utilized the sparse representation dictionaries constructed from previously collected datasets to complete the task of simultaneous denoising and super-resolution. With respect to regularization methods in sparse domain, Liu et al. [17] proposed a novel morphologic regularization method for different morphologic features. A soft information and soft decision-based method [18] maps an LR patch to a pixel-wise distribution of all its possible HR counterparts to get the soft information. The final HR pixel value is made upon the maximum posteriori estimation relying on the soft information. In terms of depth image SR, a coupled dictionary learning method with locality coordinate constraints was proposed in [19]. Timofte et al. [20] proposed an anchored neighborhood regression (ANR) method to complete fast super resolution reconstruction. The ANR anchored the neighborhoods of a low resolution patch and embedded the nearest atom in the dictionary. Subsequently, Timofte et al. [21] proposed the improved ANR named A+, which unified the ANR and the simple functions. Their algorithms accelerate the reconstruction speed by a big margin. Schulter et al. [22] directly mapped the relationship between LR patches and HR patches using the random forests. Zhang et al. [23] introduced a clustering and collaborative representation to speed up the reconstruction. In essence, there is a problem in the mapping learning from the external samples. If the mapping learned from the external samples has high similarities to the LR input, the SR output tends to be more reliable. On the contrary, the SR output tends to be poor.

Instead of external SR methods, internal SR methods search for example patches from the input image itself, based on the fact that patches often tend to recur within the image or across different image scales. For example, Glasner et al. [24] exploited above fact to combine the multi-image SR and the example-based SR. Yang et al. [25] generated the LR and HR image patches from an image pyramid using the self-similarities. Furthermore, they exploited the group sparsity constraints to regularize the ill-posed characteristic in SR reconstruction. Zhu et al. [26] proposed an effective self-example learning method using the improved K-SVD algorithm and the straightforward orthogonal matching pursuit algorithm for single image super-resolution. The exploitation of self-similarities has blazed a new path for image super-resolution, however, these internal SR methods suffer from immanent problems. For example, the patches generated from internal SR methods contain limited information, so that the learned mapping cannot sufficiently represent various image structures.

According to the discussion about external SR and internal SR, external SR methods are usually sensitive to the correlation between the LR input and the training set, while internal SR methods suffer from the limited information and mismatch problems. It takes mutual efforts to deal with the intrinsic defects of external and internal SR methods. Recently, some researchers have made attempts to combine the internal SR methods and the external SR methods. Based on in-place self-similarity, Yang et al. [27] learned a first-order approximation of a nonlinear mapping function between the external low patches and the high resolution patches. In order to associate external samples with internal samples, Wang et al. [28,29] proposed a joint model, which defined two loss functions according to the external SR methods and the internal SR methods. They also exploited a weight function to adaptively balance the contributions of the two types of methods.

To effectively take the advantage of complementary information between internal and external samples, we present a novel joint SR method based on sparse representation and the adaptive mixed samples regression. The overview of the proposed method is presented in Fig. 1. Firstly, we generate the low frequency (LF) of an initial HR image via sparse representation and low-pass filter. Secondly, we combine the self-similarities image pyramid in [25] and the fast library for approximate nearest neighbors (FLANN) [30] to quickly construct internal samples using self-similarity across different scales, then we propose an ASSS to adaptively select internal and external samples by minimizing the Euclidean distances. Thirdly, we propose an AMSRR to model the mapping between low frequency (LF) components and high frequency (HF) components of the same mixed samples, and then reconstruct the HF component by the learned AMSRR model. Finally, we combine the LF component with the reconstructed HF component to generate the HR image, and optimize the final HR image via the global means (GLM) and the global reconstruction constraint (GRC) in [8].

The main contributions of the proposed method are presented as follows:

- An ASSS is presented to adaptively select both internal and external samples. The proposed ASSS can effectively organize the internal and external samples to form the mixed samples.
- An AMSRR is proposed to model the relationship between the low and the high frequency components. The AMSRR model can take the advantage of the complementary information included in the mixed samples.

The remainder of this paper is organized as follows. In Section 2, we review the related work. In Section 3, details of the proposed algorithm including dictionary learning, sparse reconstruction and optimization are presented. Experiment results and discussions of parameters can be found in Section 4. Finally, we conclude this paper in Section 5.

2. Background

The goal of single image SR is to recover a high resolution version $\boldsymbol{X}$ from a blurred and down-sampled low resolution image $\boldsymbol{Y}$ in the noisy environment. The relationship between LR image $\boldsymbol{Y}$ and its corresponding HR image $\boldsymbol{X}$ can be mathematically written as:

$$\boldsymbol{Y} = \boldsymbol{SBX} + \upsilon \tag{1}$$

where $\boldsymbol{S}$ is a down-sampling operator, $\boldsymbol{B}$ is a blurring operator, and υ is a noise term. Given a LR input $\boldsymbol{Y}$, there may exist many corresponding HR images $\boldsymbol{X}$ which satisfy the degradation model in Eq. (1). The SR reconstruction problem is dramatically ill-posed. Thus, prior knowledge should be incorporated into the reconstruction. A notable external SR method proposed by Yang et al. [8] utilized a sparsity prior to deal with single image SR. According to their method, the patches $\boldsymbol{x}$ of a HR image $\boldsymbol{X}$ can be represented as a sparse linear combination over a learned HR dictionary $\boldsymbol{D}_h$: $\boldsymbol{x} \approx \boldsymbol{D}_h\boldsymbol{\alpha}$ s.t. $\|\boldsymbol{\alpha}\|_0 \ll c$, where $\boldsymbol{\alpha} \in \mathrm{R}^c$ is a sparse coefficients matrix and the ℓ_0-pseudo-norm counts the number of non-zero coefficients in $\boldsymbol{\alpha}$. If the patch $\boldsymbol{y}$ of the LR image $\boldsymbol{Y}$ is able to share the same sparse coefficients, the HR patch $\boldsymbol{x}$of $\boldsymbol{X}$ corresponding to the LR patch $\boldsymbol{y}$ can be well recovered

$$\boldsymbol{x} = \boldsymbol{D}_h\hat{\boldsymbol{\alpha}},\ \text{s.t.}\ \hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left\|\boldsymbol{y} - \boldsymbol{D}_l\boldsymbol{\alpha}\right\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \tag{2}$$

where $\boldsymbol{D}_l$ is the LR dictionary, $\hat{\boldsymbol{\alpha}}$ is the optimal sparse coefficients, and λ is a regularization coefficient to balance sparsity and the error term. The ℓ_1-norm rather than the ℓ_0-norm is employed in Eq. (2), because ℓ_0-norm leads to a nonconvex problem.

Internal SR methods usually acquire the self-similarities in an image pyramid. The patches with high self-similarity may exist in the same scale or across different scales. Glasner et al. [24] successfully exploited the patch self-similarity and reconstructed the HR image using a single LR input. Moreover, Yang et al. [25] emphasized the self-similarity across scales. Let $\boldsymbol{In}_{i-1}$ denote the LR image and $\boldsymbol{In}_i$ denote its adjacent

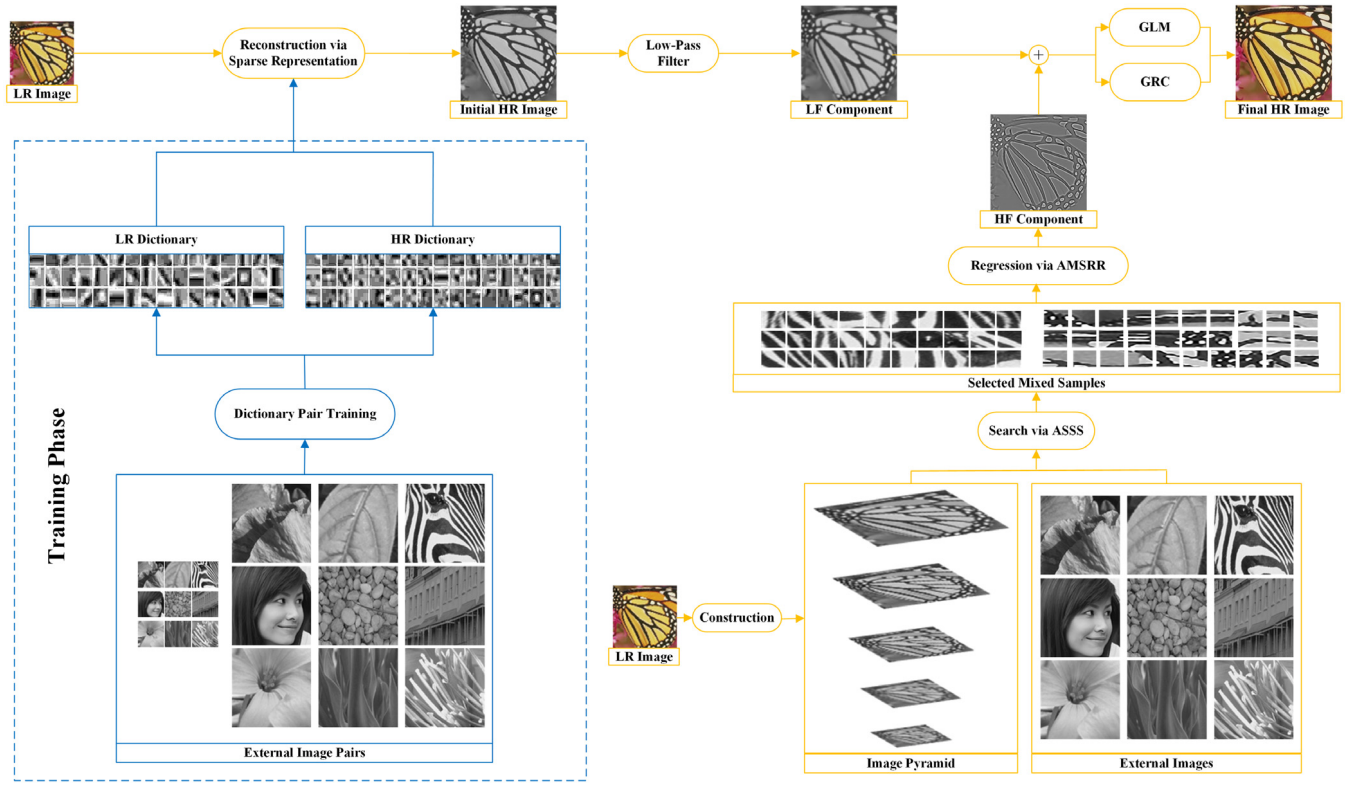

Fig. 1. Overview of the proposed method. The blue lines represent the training part and the yellow lines represent the reconstruction part.

HR version in an image pyramid. According to the image degradation model in Eq. (1), the relationship between $\boldsymbol{In}_{i-1}$ and $\boldsymbol{In}_i$ is given by

$$\boldsymbol{In}_{i-1} = (\boldsymbol{In}_i * \boldsymbol{B}_i)\boldsymbol{S}_\sigma \tag{3}$$

where $*$ is a convolution operator, $\boldsymbol{B}_i$ is a blur kernel and $\boldsymbol{S}_\sigma$ is a downsampling operator with scaling factor σ. The subscript i stands for the layer number in the image pyramid.

Given an input image $\boldsymbol{In}_0$, a series of LR images $\boldsymbol{In}_i$ ($i = -1, -2, \ldots, -a$) can be generated by Eq. (3) at first, where a is determined by upscaling factor s. The construction of HR image $\boldsymbol{In}_i$ ($i = 1, 2, \ldots, a$) is illustrated in Fig. 2. As shown in Fig. 2, let $\boldsymbol{In}_1$, $\boldsymbol{In}_2$ denote the HR layers and $\boldsymbol{In}_{-1}$, $\boldsymbol{In}_{-2}$ denote the LR layers. The patches of $\boldsymbol{In}_1$ and $\boldsymbol{In}_2$ are the copied and enlarged versions from the patches of $\boldsymbol{In}_0$. Since there are several similar patches (P_1 and P_2) to every source patch P_s in $\boldsymbol{In}_0$, P_1 and P_2 can be acquired via patch matching. Their corresponding regions R_1 and R_2 in $\boldsymbol{In}_0$ are also determined. In the same way, the patches D_1 and D_2 in the HR layers are determined by two factors: (1) the region of source patch P_s, (2) the layer index of found patches (-1 or -2). Finally, the pixel values of R_1 and R_2 are copied to D_1 and D_2 respectively with the enlarged areas.

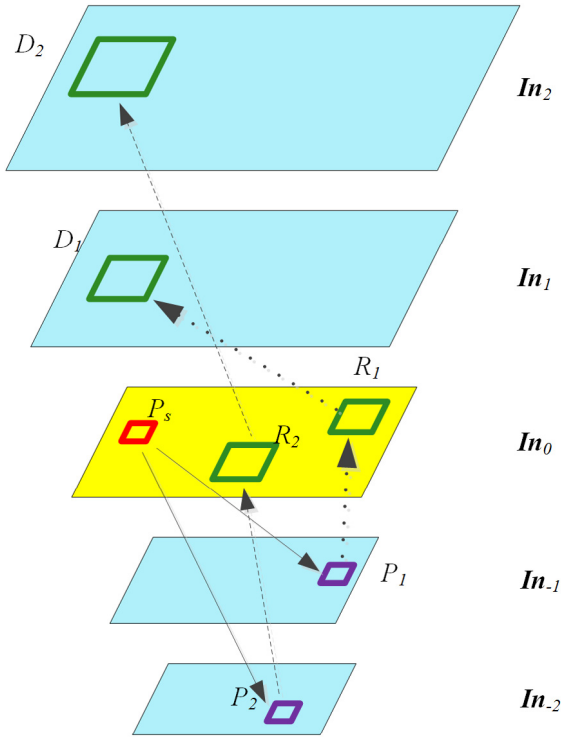

Fig. 2. The illustration of the image pyramid construction. P_s is a source patch in $\boldsymbol{In}_0$; P_1 and P_2 are the acquired similar patches in LR layers; R_1 and R_2 are the corresponding regions in $\boldsymbol{In}_0$; D_1 and D_2 are the regions to be construct in HR layers corresponding to P_s.

3. The proposed method

The proposed method named AMSRR-SCSR (sparse coding super-resolution) can be divided into three parts: dictionary learning via K-SVD, reconstruction and optimization. We divide the reconstruction into two sections: reconstruction via sparse representation and adaptive mixed samples regression model. In the following sections, we will present the proposed method in detail.

3.1. Dictionary learning via K-SVD

The dictionary plays a key role in the sparse representation-based SR. Various dictionary leaning methods have been proposed recently.

Yang et al. [31] provided a coupled dictionary training method for single image SR. In [10], Dong et al. utilized a series of PCA sub-dictionaries and adaptively selected the optimal sub-dictionaries to recover a HR image from a LR observation. Zhang et al. [32] employed SVD to learn an adaptive dictionary for each group. In this section, we follow the algorithm proposed by Aharon et al. [33] and design a pair of LR–HR dictionary.

Given an external database $\{\boldsymbol{Ex}_j\}_{j=1}^{b}$, where b is the total number of external images. Every external image $\boldsymbol{Ex}_j$ is down-sampled and then interpolated to recover a middle image $\boldsymbol{Mid}_j$ with bicubic kernel, where $\boldsymbol{Mid}_j$ has the same size as $\boldsymbol{Ex}_j$. After the database $\{\boldsymbol{Mid}\}_{j=1}^{b}$ is obtained, we get the corresponding error database $\{\boldsymbol{Err}_j\}_{j=1}^{b}$ via $\boldsymbol{Err}_j = \boldsymbol{Ex}_j - \boldsymbol{Mid}_j$. The filters $\boldsymbol{f}_1 = [1,0,-1]$, $\boldsymbol{f}_2 = [1,0,-1]^{\mathrm{T}}$, $\boldsymbol{f}_3 = [-1,0,2,0,-1]$, and $\boldsymbol{f}_4 = [-1,0,2,0,-1]^{\mathrm{T}}$ in [8,9,25,34] are used for feature extraction. Therefore, the gradient and Laplacian features can be extracted from $\{\boldsymbol{Mid}\}_{j=1}^{b}$ along horizontal and vertical directions respectively. We extract patches from feature images and concatenate four features into one to get the LR feature matrix $\boldsymbol{F}^L$. The corresponding HR feature matrix $\boldsymbol{F}^H$ can be acquired by extraction patches from $\{\boldsymbol{Err}_j\}_{j=1}^{b}$ directly. However, the concatenation of features results in a very high dimensionality of $\boldsymbol{F}^L$. It is necessary to introduce the principal component analysis (PCA) [35] to seek a subspace. The features should be projected on the subspace and preserved 99.9% of average energy [9]. Let $\boldsymbol{\Psi}$ denote the projection matrix that transforms $\boldsymbol{F}^L$ to its reduced eigenvectors, thus $\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{F}^L$ is the final training subspace. According to the definition above, the LR dictionary $\boldsymbol{D}^L$ and the sparse coefficient matrix $\boldsymbol{\Lambda}$ can be calculated via

$$\boldsymbol{D}^L, \boldsymbol{\Lambda} = \arg\min_{\boldsymbol{D}^L,\boldsymbol{\Lambda}} \left\|\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{F}^L - \boldsymbol{D}^L\boldsymbol{\Lambda}\right\|_F^2 \text{ s.t. } \left\|\boldsymbol{\Lambda}_\xi\right\|_0 \le \tau \ \forall\xi \tag{4}$$

where $\boldsymbol{\Lambda}_\xi$ is the ξth column of $\boldsymbol{\Lambda}$, τ controls the sparsity and $\|\cdot\|_F$ is the Frobenius norm. Considering $\boldsymbol{F}^H$ is acquired, we train the HR dictionary $\boldsymbol{D}^H$ with the assumption that HR and LR dictionaries share the same sparse coefficients $\boldsymbol{\Lambda}$

$$\boldsymbol{D}^H = \arg\min_{\boldsymbol{D}^H} \left\|\boldsymbol{F}^H - \boldsymbol{D}^H\boldsymbol{\Lambda}\right\|_F^2. \tag{5}$$

The solution of Eq. (5) is given by a Pseudo-Inverse expression

$$\boldsymbol{D}^H = \boldsymbol{F}^H\boldsymbol{\Lambda}^{\mathrm{T}}\left(\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{\mathrm{T}}\right)^{-1}. \tag{6}$$

3.2. Reconstruction

The reconstruction part can be divided into three phases. In the first phase, an initial HR image is reconstructed via sparse representation. In the second phase, an ASSS is proposed to select internal and external samples adaptively. Finally, an AMSRR model is proposed to refine the high frequency components of the initial HR image.

3.2.1. Reconstruction via sparse representation

Given an LR image $\boldsymbol{Y}$, it is scaled up by the bicubic interpolation with upscaling factor s. We extract the gradient and the Laplacian features along horizontal and vertical directions from the interpolated middle image. As shown in Section 3.1, we extract patches from feature images and concatenate four features to form the feature matrix $\boldsymbol{F}_{LR}$. In order to reduce the dimensionality of $\boldsymbol{F}_{LR}$, we apply PCA projection matrix $\boldsymbol{\Psi}$ to $\boldsymbol{F}_{LR}$. And then the orthogonal matching pursuit algorithm (OMP) [36] is employed to calculate the sparse coefficients. With the LR dictionary $\boldsymbol{D}^L$, we solve the sparse coefficients $\boldsymbol{A}$ by Eq. (7).

$$\boldsymbol{A} = \arg\min_{\boldsymbol{A}} \left\|\boldsymbol{\Psi}^{\mathrm{T}}\boldsymbol{F}_{LR} - \boldsymbol{D}^L\boldsymbol{A}\right\|_F^2 \text{ s.t. } \left\|\boldsymbol{A}_\rho\right\|_0 \le \tau \ \forall\rho \tag{7}$$

where $\boldsymbol{A}_\rho$ is the ρth vector of $\boldsymbol{A}$. Let $\boldsymbol{R}_o$ denote the patch extraction operator, so that $\boldsymbol{R}_o\boldsymbol{X}$ denote the oth patch of $\boldsymbol{X}$. Thus, we can reconstruct an initial HR image $\hat{\boldsymbol{X}}$ using the HR dictionary $\boldsymbol{D}^H$

$$\hat{\boldsymbol{X}} = \boldsymbol{\Delta}\boldsymbol{Y} + \varphi\left(\sum_{o=1}^{r}\boldsymbol{R}_o^{\mathrm{T}}\boldsymbol{R}_o\right)^{-1}\sum_{o=1}^{r}\boldsymbol{R}_o^{\mathrm{T}}\boldsymbol{D}^H\boldsymbol{A} \tag{8}$$

where $\boldsymbol{\Delta}$ is a bicubic interpolation operator, φ is a weight parameter, and r is the total number of the extracted patches.

3.2.2. Adaptive samples selection scheme

According to the description in Section 2, we build an image pyramid to generate the internal database via [25]. Since the construction speed in [25] is very slow, we introduce the FLANN in [30] to accelerate the search of the similar patches. Comparison experiments on the improved fast image pyramid and the method [25] in Section 4.3 demonstrate that the improved fast image pyramid significantly speed up the construction of the image pyramid. As a result, we can easily get the internal database $\{\boldsymbol{In}_i\}_{i=-a}^{a}$ by the improved fast image pyramid and the external database $\{\boldsymbol{Ex}_j\}_{j=1}^{b}$. Furthermore, we present an ASSS to seek mixed samples similar to the initial HR image $\hat{\boldsymbol{X}}$ from external and internal samples.

It is necessary to acquire the low and high frequency components of each image in $\{\boldsymbol{In}_i\}_{i=-a}^{a}$ and $\{\boldsymbol{Ex}_j\}_{j=1}^{b}$, as well as the low frequency components of the initial HR image $\hat{\boldsymbol{X}}$. We use a 7 × 7 Gaussian filter $\boldsymbol{H}_l$ with a standard deviation 1.6 to convolute each image. For example, $\boldsymbol{In}_i^l = \boldsymbol{H}_l * \boldsymbol{In}_i$ is the low frequency components and $\boldsymbol{In}_i^h = \boldsymbol{In}_i - \boldsymbol{H}_l * \boldsymbol{In}_i$ is the corresponding high frequency components of $\boldsymbol{In}_i$. Correspondingly, $\boldsymbol{Ex}_j^l$ and $\boldsymbol{Ex}_j^h$ are the low and high frequency components of $\boldsymbol{Ex}_j$. We extract internal sample pairs $\{\boldsymbol{p}_i^l, \boldsymbol{p}_i^h\}_{i=1}^{m}$ from $\boldsymbol{In}_i^l$, $\boldsymbol{In}_i^h$ and external sample pairs $\left\{\boldsymbol{q}_j^l, \boldsymbol{q}_j^h\right\}_{j=1}^{n}$ from $\boldsymbol{Ex}_j^l$, $\boldsymbol{Ex}_j^h$, where m and n are the number of internal samples and external samples respectively.

Let $\hat{\boldsymbol{x}}_o^l$ denote the patch extracted from the low frequency components of $\hat{\boldsymbol{X}}$. Since searching in the low frequency components $\boldsymbol{P}^l = \{\boldsymbol{p}_i^l\}_{i=1,2,\ldots,m}$ and $\boldsymbol{Q}^l = \left\{\boldsymbol{q}_j^l\right\}_{i=1,2,\ldots,n}$ is more robust than searching in the high frequency components $\boldsymbol{P}^h = \{\boldsymbol{p}_i^h\}_{i=1,2,\ldots,m}$ and $\boldsymbol{Q}^h = \left\{\boldsymbol{q}_j^h\right\}_{i=1,2,\ldots,n}$, we perform the K-nearest neighbor(KNN) algorithm to acquire the first k most similar samples that similar to $\hat{\boldsymbol{x}}_o^l$ in set $\boldsymbol{P}^l$ and set $\boldsymbol{Q}^l$

$$\boldsymbol{U}^l = \mathrm{KNN}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{P}^l, k\right), \boldsymbol{V}^l = \mathrm{KNN}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{Q}^l, k\right) \tag{9}$$

where $\boldsymbol{U}^l$ and $\boldsymbol{V}^l$ are the internal and external nearest neighbors satisfying the following conditions:

$$\begin{vmatrix}\boldsymbol{U}^l\\ \boldsymbol{V}^l\end{vmatrix} = k, \boldsymbol{U}^l \in \boldsymbol{P}^l \ \forall\boldsymbol{u} \in \boldsymbol{U}^l, \boldsymbol{p} \in \boldsymbol{P}^l - \boldsymbol{U}^l, \mathrm{D}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{u}\right) \le \mathrm{D}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{p}\right)$$
$$\begin{vmatrix}\boldsymbol{U}^l\\ \boldsymbol{V}^l\end{vmatrix} = k, \boldsymbol{V}^l \in \boldsymbol{Q}^l \ \forall\boldsymbol{v} \in \boldsymbol{V}^l, \boldsymbol{q} \in \boldsymbol{Q}^l - \boldsymbol{V}^l, \mathrm{D}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{v}\right) \le \mathrm{D}\left(\hat{\boldsymbol{x}}_o^l, \boldsymbol{q}\right) \tag{10}$$

where D is the Euclidean distance metric. We still use the FLANN lib [30] to effectively solve Eq. (9). The ASSS can be completed quite fast with FLANN [30], even the values of m and n are very large. Thus, we can combine internal samples and external samples to form the mixed samples

$$\boldsymbol{\Phi}_o^l = \begin{bmatrix}\boldsymbol{U}^l\\ \boldsymbol{V}^l\end{bmatrix}, \boldsymbol{\Phi}_o^h = \begin{bmatrix}\boldsymbol{U}^h\\ \boldsymbol{V}^h\end{bmatrix} \tag{11}$$

where $\boldsymbol{\Phi}_o^l$, $\boldsymbol{\Phi}_o^h$ are the low and high frequency components of the selected mixed samples corresponding to $\hat{\boldsymbol{x}}_o^l$. $\boldsymbol{U}^h$, $\boldsymbol{V}^h$ are the high frequency components corresponding to $\boldsymbol{U}^l$ and $\boldsymbol{V}^l$ respectively.

3.2.3. Adaptive mixed samples ridge regression model

It is well known that the high frequency components are the key of the SR reconstruction. To refine the high frequency components of the initial HR image $\hat{\boldsymbol{X}}$, we introduce an adaptive mixed samples ridge regression model. Inspired by the recent work [20], given a low frequency patch $\hat{\boldsymbol{x}}_o^l$, we formulate the reconstruction question as a least squares regression problem and use the l_2 regularization to deal with the extremely ill-posed characteristic. The ridge regression as a well-known method for solving ill-posed problems is used in the reconstruction. Because the proposed model is based on mixed samples and ridge regression, we name it as AMSRR. In order to take the advantage of the complementary information, we formulate the question as solving the coefficients of AMSRR

$$\boldsymbol{\beta}_o = \arg\min_{\boldsymbol{\beta}_o} \left\|\hat{\boldsymbol{x}}_o^l - \boldsymbol{\Phi}_o^l\boldsymbol{\beta}_o\right\|_2^2 + \eta\left\|\boldsymbol{\beta}_o\right\|_2^2 \tag{12}$$

where η is a regularization parameter. The solution of Eq. (12) is given by

$$\boldsymbol{\beta}_o = \left(\boldsymbol{\Phi}_o^{l\mathrm{T}}\boldsymbol{\Phi}_o^l + \eta\boldsymbol{I}\right)^{-1}\boldsymbol{\Phi}_o^{l\mathrm{T}}\hat{\boldsymbol{x}}_o^l \tag{13}$$

where $\boldsymbol{I}$ is an identify matrix. And then, we use the coefficients $\boldsymbol{\beta}_o$ to reconstruct the high frequency components via Eq. (13)

$$\widetilde{\boldsymbol{X}}^h = \left(\sum_{o=1}^{r}\boldsymbol{R}_o^{\mathrm{T}}\boldsymbol{R}_o\right)^{-1}\sum_{o=1}^{r}\boldsymbol{\Phi}_o^h\boldsymbol{\beta}_o \tag{14}$$

where $\boldsymbol{\Phi}_o^h$ is the high frequency components of the selected mixed samples. Ultimately, we reconstruct the HR image $\widetilde{\boldsymbol{X}}$ via

$$\widetilde{\boldsymbol{X}} = \boldsymbol{H}_l\hat{\boldsymbol{X}} + \rho\left(\sum_{o=1}^{r}\boldsymbol{R}_o^{\mathrm{T}}\boldsymbol{R}_o\right)^{-1}\sum_{o=1}^{r}\boldsymbol{\Phi}_o^h\boldsymbol{\beta}_o \tag{15}$$

where ρ is a weight parameter.

3.3. Optimization

Recent research [24] has shown that the nature images tend to contain repetitive patterns. The nonlocal redundancies are usually collected and employed to improve the quality of the reconstructed HR image. The nonlocal means (NLM) algorithm [37] searches the similar patches centered on a target patch with a limited radius, while we extend the search area to the global region named the GLM algorithm. Suppose that $\widetilde{\boldsymbol{x}}_o$ is a local patch of the current estimation, we employ the FLANN method [30] to search several similar patches $\{\widetilde{\boldsymbol{x}}_o^e\}_{e=1}^{\mu}$ corresponding to $\widetilde{\boldsymbol{x}}_o$ in the global region. The similar patches are used to predict $\widetilde{\boldsymbol{x}}_o$ by

$$\widetilde{\boldsymbol{x}}_o = \sum_{e=1}^{\mu}\widetilde{\boldsymbol{x}}_o^e g_o^e \tag{16}$$

where the weight g_o^e is calculated by

$$g_o^e = \frac{1}{\gamma}\exp(-\|\widetilde{\boldsymbol{x}}_o - \widetilde{\boldsymbol{x}}_o^e\|_2^2/w) \tag{17}$$

where w is a control factor of weight and $\gamma = \sum_{o=1}^{\mu}\exp(-\|\widetilde{\boldsymbol{x}}_o - \widetilde{\boldsymbol{x}}_o^e\|_2^2/w)$. Let $\boldsymbol{g}_o$ be the column vector containing all the weights g_o^e and $\boldsymbol{\chi}_o$ be the column vector containing all the similar patches $\widetilde{\boldsymbol{x}}_o^e$. We expect that the prediction error $\|\widetilde{\boldsymbol{x}}_o - \boldsymbol{g}_o^{\mathrm{T}}\boldsymbol{\chi}_o\|_2^2$ should be minimized. Let $\boldsymbol{I}$ be the identity matrix and $\boldsymbol{\Theta}$ is defined as

$$\boldsymbol{\Theta}(o,e) = \begin{cases} g_o^e, & \text{if } \widetilde{\boldsymbol{x}}_o^e \text{ is an element of } \boldsymbol{\chi}_o, g_o^e \in \boldsymbol{g}_o \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

The prediction error $\|\widetilde{\boldsymbol{x}}_o - \boldsymbol{g}_o^{\mathrm{T}}\boldsymbol{\chi}_o\|_2^2$ can be rewritten as $\|(\boldsymbol{I}-\boldsymbol{\Theta})\boldsymbol{X}\|_2^2$.

We adopt the gradient descent rule to enforce the GRC as in [8]. We project $\widetilde{\boldsymbol{X}}$ onto the solution $\boldsymbol{SBX} = \boldsymbol{Y}$ and incorporate the global similarity regularization term to the optimization question

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}}\left\{\|\boldsymbol{SBX}-\boldsymbol{Y}\|_2^2 + \theta\left\|\boldsymbol{X}-\widetilde{\boldsymbol{X}}\right\|_2^2 + \varepsilon\|(\boldsymbol{I}-\boldsymbol{\Theta})\boldsymbol{X}\|_2^2\right\}. \tag{19}$$

The solution of Eq. (19) can be effectively calculated using the gradient descent rule and the update equation can be formulated as

$$\boldsymbol{X}_{t+1} = \boldsymbol{X}_t + \upsilon\left[\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}^{\mathrm{T}}\left(\boldsymbol{Y}-\boldsymbol{SBX}_t\right) + \theta\left(\boldsymbol{X}-\widetilde{\boldsymbol{X}}\right) - \varepsilon(\boldsymbol{I}-\boldsymbol{\Theta})^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{\Theta})\boldsymbol{X}_t\right] \tag{20}$$

where $\boldsymbol{X}_t$ is the estimate of the high-resolution image after the tth iteration, υ is the step size of the gradient descent, θ and ε are the weight parameters. In the end, the optimal $\boldsymbol{X}^*$ is generated as the final HR output image.

We summarize the proposed method in Algorithm 1.

4. Experimental results

In this section, we conduct a batch of experiments to demonstrate the effectiveness of the proposed method. Parameters analysis and comparison experiments are presented as follows. We have completed the proposed algorithm in MATLAB R2014b and conducted the experiments on a PC with Intel Core i5 at 3.3 GHz and 12.0GB memory. Since human vision is more sensitive to luminance changes, the luminance channel of each color image is upscaled by all test methods and the chrominance channel of each color image is upscaled by bicubic interpolation method. Moreover, the peak signal to noise ratio (PSNR) and the structural similarity (SSIM) [38] are adopted to evaluate the quality of reconstructed images.

The training datasets are generated from the 91 training images in [34]. The "Set 5" and "Set 14" [20] containing 5 and 14 images respectively are used for SR performance testing. Note that the selected training set is only used for dictionary training offline, however, both training set and testing set are treated as the sources of external samples and internal samples respectively. The original HR images in "Set 5" and "Set 14" are blurred with the bicubic kernel and decimated by a factor of s to generate the input LR images.

4.1. Experimental settings

In the training and reconstruction phases, a window of size 3×3 multiplying up-scaling factor s is employed to extract patches. More specifically, the patch size is set as 6×6 when $s = 2$ and 9×9 when $s = 3$. The sparsity τ is set as 3. In the dictionary training phase, we set the overlap as half the patch size, i.e., the overlap is set as 3 for up-scaling factor 2 and 5 for up-scaling factor 3. In the end, we collect 573,353, 315,093 and 137,718 LR and HR feature pairs, respectively. In the reconstruction phase, the configurations of parameter η and φ vary according to different up-scaling factor s. For example, we set the regularization parameter η as 0.0001 for up-scaling factor 2 and 0.05 for up-scaling factor 3. The weight parameter ρ is set as 1 under up-scaling factor 2 and 3. The other parameters, such as the global similar patch number μ, the iteration number t, the step size υ, and the weight parameters φ, θ, ε are set as 10, 200, 1.8, 1, 1, and 0.05 respectively. The overlap is set as 4, 6, 10 respectively for the LR image to be coded over the learned dictionaries. In the sample selection scheme, we collect 576,284 low and high frequency image patch pairs as the external sample set for up-scaling factor 2, 317,719 for up-scaling factor 3, and 138,812 for up-scaling factor 4. Since the sizes of the input LR images vary from each other, the number of internal samples is quite different. Here, we report our overlap configuration. The overlap is set as 5, 8, 10 respectively. The parameter k stands for the number of external and internal samples that will be selected. We will discuss the selection of k in the next section.

4.2. Parameters analysis

In this section, we discuss the influence of the parameter k. In our experiments, the number of the selected internal samples and external samples is equal. To acquire the first k most similar samples from internal sample set $\boldsymbol{P}^l$ and external sample set $\boldsymbol{Q}^l$ for each low frequency $\hat{\boldsymbol{x}}_o^l$ respectively, we use the FLANN lib [30]. In essence, FLANN [30] can perform the searching task quite fast. Fig. 3 presents the average PSNR and SSIM values of images in Set 14 according to different values of sample number k. It is clear that both the average PSNR and SSIM values ascend at first and then descend as k increases. Because there is limited information in the selected mixed samples $\boldsymbol{\Phi}_o^l$ and $\boldsymbol{\Phi}_o^h$ when k is small. As k increases, more available information included in these similar samples is collected. The accumulation of k contributes to a better SR result. However, when k exceeds a certain number (such as 15 shown in Fig. 3), the accumulated dissimilar patches with much wrong information will degrade the quality of the reconstruction results.

Algorithm 1

1) Dictionary training:

   a) **Input**: The external training set $\{\boldsymbol{Ex}_j\}_{j=1}^{b}$ and the sparsity τ.

   b) Prepare LR and HR feature pairs $\{\boldsymbol{F}^L, \boldsymbol{F}^H\}$ and compute the PCA projection matrix $\boldsymbol{\Psi}$.

   c) Solve Eq. (4) to get the LR dictionary $\boldsymbol{D}^L$ and the spare coefficient $\boldsymbol{\Lambda}$.

   d) Compute the HR dictionary $\boldsymbol{D}^H$ via solving Eq. (6).

   e) **Output**: LR and HR dictionary pair $\{\boldsymbol{D}^L, \boldsymbol{D}^H\}$ and the PCA projection matrix $\boldsymbol{\Psi}$.

2) Reconstruction:

   a) **Input**: a LR image $\boldsymbol{Y}$, the sparsity τ, the PCA projection matrix $\boldsymbol{\Psi}$, the LR and HR dictionary pair $\{\boldsymbol{D}^L, \boldsymbol{D}^H\}$, the sample number k, the regularization parameter η, the global similar patch number μ, the control factor of weight w, the weight parameters φ, ρ, θ and ε, the iteration number t, the step size v.

   b) Prepare the LR feature $\boldsymbol{F}_{LR}$ and the internal database $\{\boldsymbol{In}_i\}_{i=-a}^{a}$.

   c) Compute the sparse coefficients $\boldsymbol{A}$ via solving Eq. (7).

   d) Construct an initial HR image $\hat{\boldsymbol{X}}$ with the HR dictionary $\boldsymbol{D}^H$ and the sparse coefficients $\boldsymbol{A}$.

   e) Acquire the low and high frequency components $\{\boldsymbol{P}^l, \boldsymbol{P}^h\}$ and $\{\boldsymbol{Q}^l, \boldsymbol{Q}^h\}$ using a 7×7 Gaussian filter $\boldsymbol{H}_l$ with a standard deviation 1.6, as well as the low frequency patch $\hat{\boldsymbol{x}}_o^l$ of the initial HR image $\hat{\boldsymbol{X}}$.

   For each $\hat{\boldsymbol{x}}_o^l$ **do**:

   f) Search the first k most similar samples in both internal set $\boldsymbol{P}^l$ and external set $\boldsymbol{Q}^l$, and construct the mixed samples $\{\boldsymbol{\Phi}_o^l, \boldsymbol{\Phi}_o^h\}$ via solving Eq. (9) and Eq. (11).

   g) Compute the coefficients $\boldsymbol{\beta}_o$ of AMSRR via solving Eq. (13).

   h) Construct the high frequency using $\boldsymbol{\beta}_o$ and $\boldsymbol{\Phi}_o^h$, such that the estimated HR image $\tilde{\boldsymbol{X}}$ can be generated.

   end for

3) Optimization:

   a) Search the similar patches for each $\tilde{\boldsymbol{x}}_o$ in the global region and compute the weight g_o^e.

   b) Enforce the GLM regularization and the global reconstruction constraint to acquire an optimal solution $\boldsymbol{X}^*$ of Eq. (19) via updating Eq. (20).

   c) **Output**: The final HR image $\boldsymbol{X}^*$.

Table 1
The construction time comparison between method [25] and the fast image pyramid.

Set14 Image name	Input size	Yang et al. [25] Time (s)	Ours Time (s)	**Set14** Image name	Input size	Yang et al. [25] Time (s)	Ours
baboon	160 × 166	1567.19	**120.58**	foreman	96 × 117	373.97	**52.14**
barbara	192 × 240	3370.95	**271.92**	lenna	170 × 170	1593.67	**178.5**
bridge	170 × 170	1613.00	**144.86**	man	170 × 170	1795.02	**174.97**
coastguard	96 × 117	374.83	**53.41**	monarch	170 × 256	2799.25	**236.38**
comic	120 × 83	385.81	**50.35**	pepper	170 × 170	1631.08	**167.40**
face	92 × 92	269.08	**40.39**	ppt3	218 × 176	2233.35	**229.00**
flower	120 × 166	944.57	**115.04**	zebra	130 × 195	1605.13	**131.72**

4.3. Experiments about fast image pyramid

Since we have improved the construction of image pyramid in [25], we compare the elapsed time of building the same image pyramid between our fast image pyramid and the method in [25]. When the upscaling factor s is 3, we create an image pyramid with 11 layers. In our experiments, we set the variance of Gaussian blur kernel as $\nabla = 1.0$ and the subscript $\nabla = 1.0$ as $\sigma = 1.25$ in Eq. (3). This setting is in accordance with [25]. Table 1 shows the sizes of the input images and the construction time of the corresponding pyramids. It is clear that our

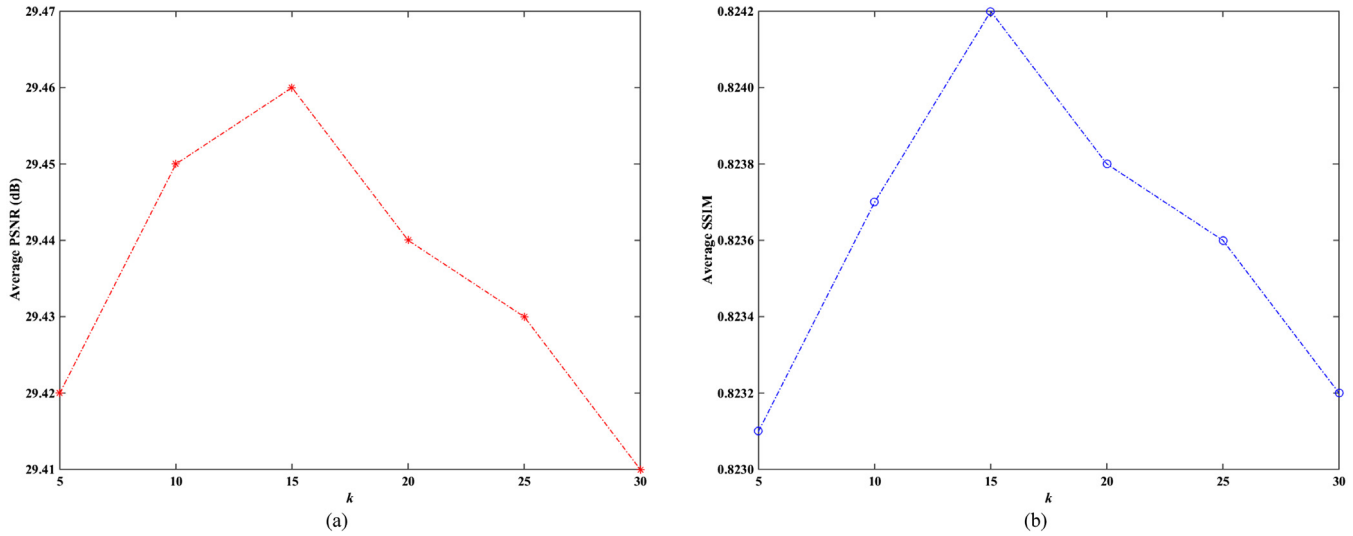

Fig. 3. The average PSNR (dB) and SSIM values for up-scaling ×3 in Set14 with different values of k. (a) The average PSNR values. (b) The average SSIM values.

method significantly accelerate the construction of the image pyramid. The fast image pyramid saves about 85%–95% time of the construction compared with the method in [25]. Indeed, our fast image pyramid boosts the construction speed without any performance degradation.

4.4. Comparisons with the state-of-the-art methods

In this section, we compare the proposed AMSRR-SCSR method with some state-of-the-art methods to validate the effectiveness of the proposed algorithm. The competing methods include the classical sparse representation-based method proposed by Zeyde et al. in [9], the regression-based method proposed by Timofte et al. in [21], the self-exemplars-based method proposed by Huang et al. in [39] and the deep learning-based method [40–42]. The sparse representation-based method [9] stands for the baseline and the four methods [21,39–42] stand for the state-of-the-art methods. According to their released source codes, all experimental results are compared subjectively and objectively. The quantified SR results (PSNR and SSIM scores) for different up-scaling factors (2 and 3) are listed in Tables 2, 3, 4, and 5, respectively. The visual comparisons are presented in Figs. 4, 5, 6, and 7.

Tables 2–5 show the proposed AMSRR-SCSR method outperforms SR methods [9,21,39,40] and is slightly better than the deep learning-based method [41], but cannot beat VDSR [42]. The gains are 1.24 dB higher than the baseline [9] on Set5 in terms of average PSNR value and 0.0078 higher in terms of average SSIM value for up-scaling ×2. When it comes to up-scaling ×3, the gains on Set5 are 1.19 dB and 0.0185 respectively. The gains on Set14 are 0.90 dB in terms of average PSNR value, 0.0109 in terms of average SSIM value for up-scaling ×2, and 0.79 dB in terms of average PSNR value, 0.0175 in terms of average SSIM value for up-scaling ×3. Compared with A^+ [21], the gains for up-scaling ×2 are 0.47 dB on Set5 and 0.43 dB on Set14 in terms of average PSNR value, 0.0027 on Set5 and 0.0040 on Set14 in terms of average SSIM value. For up-scaling ×3, the gains are 0.50 dB on Set5 and 0.33 dB on Set14 in terms of average PSNR value, 0.0068 on Set5 and 0.0060 on Set14 in terms of average SSIM value.

We also conduct experiments with and without the proposed ASSS and AMSRR model to demonstrate the effectiveness of the proposed method. We name the proposed algorithm without ASSS and AMSRR as SCSR. Tables 2–5 show that the proposed ASSS and AMSRR significantly promote the SR performances. Compared with our SCSR, the gains of the proposed AMSRR-SCSR method are 1.20 dB in terms of average PSNR value and 0.0069 in terms of average SSIM value on Set5 for up-scaling ×2; the gains are 1.09 dB in terms of average PSNR value and 0.0169 in terms of average SSIM value on Set5 for up-scaling ×3; the gains are 0.90 dB in terms of average PSNR value and 0.0078 in terms of average SSIM value on Set14 for up-scaling ×2; the gains are 0.77 dB in terms of average PSNR value and ,0.0129 in terms of average SSIM value on Set14 for up-scaling ×3.

From Figs. 4, 5, 6, and 7, we can see that the proposed AMSRR-SCSR method is competitive and usually produces shaper boundaries and more details than the competing methods. For example, the eyebrow of Fig. 4, i.e. the 'woman', reconstructed by our proposed AMSRR-SCSR method is shaper than others and is the closest to the ground truth; With respective to the 'ppt3' in Fig. 5, our proposed AMSRR-SCSR method provides the clearest words in the yellow rectangle; in the 'baby' image, i.e. Fig. 6, our proposed AMSRR-SCSR method produces more details of the eyelashes than other methods, while VDSR [42] provides the sharpest visual quality; In Fig. 7, both VDSR [42] and our proposed AMSRR-SCSR methods reconstruct more reliable texture of the legs than others.

4.5. Computational complexity

We analyze the computational complexity of the proposed AMSRR-SCSR method. The computational complexity mainly consists of four parts: (1) reconstruction via sparse coding with Eq. (4), (2) the proposed ASSS by solving Eq. (9), (3) computing the ridge regression coefficients for each patch by solving Eq. (13), (4) the optimization process.

(1) The complexity of reconstruction via sparse coding can be divided into two parts: the sparse coding over the LR dictionary $\boldsymbol{D}^L$ using Eq. (4) and the reconstruction over the learned HR dictionary $\boldsymbol{D}^H$ by solving Eq. (6). We use the Batch-OMP algorithm proposed in [36] with an explicit dictionary to solve Eq. (4). According to the complexity analysis of [36], the complexity is $\boldsymbol{O}\left(r\left(2d_1d_2+\tau^2d_2+3\tau d_2+\tau^3\right)\right)$, where d_1 and d_2 are the dimensions of the LR dictionary $\boldsymbol{D}^L$. In addition, the complexity of the reconstruction over the HR dictionary $\boldsymbol{D}^H$ is $\boldsymbol{O}\left(rd_3d_2\right)$, where d_3 is the dimension of the patch. Thus, the complexity of the reconstruction via sparse coding is $\boldsymbol{O}(r(2d_1d_2+\tau^2d_2+3\tau d_2+\tau^3+d_3d_2))$. (2) Following the analysis described in [43], the complexity of searching includes the tree construction time complexity and the searching time complexity, because we use the priority search k-means tree. The complexity is $\boldsymbol{O}\left(md_3\kappa h\left(\log m/\log\kappa\right)\right)+\boldsymbol{O}\left(rd_3\left(\log m/\log\kappa\right)\right)$ and $\boldsymbol{O}\left(nd_3\kappa h\left(\log n/\log\kappa\right)\right)+\boldsymbol{O}\left(rd_3\left(\log n/\log\kappa\right)\right)$ for the internal samples and the external samples in the ASSS phase, respectively, where κ is the branching factor and h is the maximum number of iteration. (3) In the high frequency components refinement phase, the complexity is $\boldsymbol{O}\left(r(8k^3+2d_3k)\right)$, which includes solving Eq. (13)

Table 2
The average PSNR (dB) and SSIM values of images in Set5 for up-scaling × 2 by using different SR approaches.

Image	Method															
	Zeyde et al. [9]		A+ [21]		SelfExSR [39]		SRCNN [40]		FSCNN [41]		VDSR [42]		Our SCSR		AMSRR-SCSR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
baby	38.25	0.9628	38.52	0.9647	38.46	0.9643	38.54	0.9651	38.56	0.9654	38.75	0.9667	38.46	0.9651	**38.78**	**0.9675**
bird	39.93	0.9836	41.12	0.9865	41.07	0.9864	40.91	0.9859	41.65	0.9869	**42.42**	**0.9890**	40.02	0.9845	41.47	0.9876
butterfly	30.65	0.9541	32.01	0.9652	31.94	0.9635	32.75	0.9652	33.34	0.9694	**34.49**	**0.9750**	30.36	0.9500	32.88	0.9684
head	35.59	0.8819	35.77	0.8867	35.68	0.8852	35.72	0.8862	35.76	0.8871	**35.93**	**0.8900**	35.69	0.8857	35.88	0.8896
woman	34.49	0.9646	35.31	0.9694	35.36	0.9690	35.37	0.9686	35.69	0.9702	36.05	0.9730	34.59	0.9661	**36.07**	**0.9731**
Average	35.78	0.9494	36.55	0.9545	36.50	0.9538	36.66	0.9542	37.00	0.9558	**37.53**	**0.9587**	35.82	0.9503	37.02	0.9572

Table 3
The average PSNR (dB) and SSIM values of images in Set5 for up-scaling × 3 by using different SR approaches.

Image	Method															
	Zeyde et al. [9]		A+ [21]		SelfExSR [39]		SRCNN [40]		FSCNN [41]		VDSR [42]		Our SCSR		AMSRR-SCSR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
baby	35.08	0.9202	35.21	0.9225	35.20	0.9236	35.25	0.9233	35.29	0.9238	35.38	0.9261	35.25	0.9242	**35.52**	**0.9283**
bird	34.57	0.9478	35.54	0.9560	35.76	0.9583	35.48	0.9550	36.01	0.9588	**36.66**	**0.9643**	34.63	0.9498	35.90	0.9601
butterfly	25.94	0.8803	27.24	0.9124	26.92	0.9060	27.95	0.9121	28.68	0.9266	**29.96**	**0.9423**	25.98	0.8749	28.39	0.9253
head	33.56	0.8197	33.77	0.8268	33.74	0.8272	33.71	0.8267	33.81	0.8286	**33.96**	**0.8337**	33.66	0.8252	33.86	0.8304
woman	30.37	0.9185	31.20	0.9296	31.45	0.9319	31.37	0.9297	32.01	0.9348	**32.36**	**0.9404**	30.48	0.9203	31.77	0.9347
Average	31.90	0.8973	32.59	0.9090	32.62	0.9094	32.75	0.9094	33.16	0.9145	**33.66**	**0.9213**	32.00	0.8989	33.09	0.9158

Table 4
The average PSNR (dB) and SSIM values of images in Set14 for up-scaling × 2 by using different SR approaches.

Image	Method															
	Zeyde et al. [9]		A+ [21]		SelfExSR [39]		SRCNN [40]		FSCNN [41]		VDSR [42]		Our SCSR		AMSRR-SCSR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
baboon	25.47	0.7522	25.65	0.7662	25.52	0.7625	25.74	0.7706	25.82	0.7757	**25.94**	**0.7787**	25.55	0.7629	25.61	0.7666
barbara	**28.70**	0.8707	**28.70**	0.8749	28.49	0.8742	28.64	0.8733	28.31	0.8719	28.41	**0.8777**	**28.70**	0.8746	28.61	0.8738
bridge	27.55	0.8398	27.78	0.8498	25.89	0.8258	27.84	0.8520	27.88	0.8548	**28.05**	**0.8585**	27.58	0.8459	27.90	0.8548
coastguard	30.41	0.8373	30.57	0.8428	30.70	0.8473	30.83	0.8497	30.81	0.8490	30.99	0.8549	30.46	0.8457	**31.18**	**0.8602**
comic	27.65	0.8959	28.29	0.9110	28.35	0.9149	28.52	0.9123	28.89	0.9197	**29.40**	**0.9319**	27.74	0.9002	28.61	0.9167
face	35.57	0.8820	35.74	0.8868	35.63	0.8851	35.70	0.8863	35.74	0.8872	**35.91**	**0.8900**	35.66	0.8859	35.86	0.8897
flowers	32.28	0.9271	33.02	0.9356	33.02	0.9354	33.32	0.9363	33.67	0.9400	**34.33**	**0.9459**	32.27	0.9292	33.45	0.9396
foreman	36.18	0.9687	36.94	0.9727	36.79	0.9689	36.42	0.9711	36.87	0.9723	37.40	0.9737	36.56	0.9694	**37.68**	**0.9741**
lenna	36.21	0.9262	36.60	0.9296	36.52	0.9289	36.64	0.9296	36.83	0.9312	**37.06**	**0.9326**	36.33	0.9289	36.86	0.9321
man	30.44	0.877	30.87	0.8859	30.83	0.8851	31.04	0.8877	31.16	0.8905	**31.43**	**0.8963**	30.44	0.8797	30.99	0.8878
monarch	35.75	0.9726	37.01	0.9767	37.22	0.9762	37.74	0.9770	38.31	0.9784	**39.40**	**0.9809**	35.64	0.9730	37.83	0.9785
pepper	36.59	0.9188	37.02	0.9216	37.00	0.9211	36.87	0.9211	37.03	0.9222	**37.37**	**0.9240**	36.39	0.9199	37.07	0.9224
ppt3	29.30	0.9694	30.09	0.9768	31.43	0.9821	31.52	0.9785	31.65	0.9806	**32.81**	**0.9871**	28.91	0.9631	32.03	0.9831
zebra	33.21	0.9388	33.59	0.9426	33.79	0.9431	33.49	0.9436	33.89	0.9446	**34.23**	0.9456	33.17	0.9418	34.18	**0.9492**
Average	31.81	0.8983	32.28	0.9052	32.23	0.9036	32.45	0.9064	32.63	0.9084	**33.03**	**0.9124**	31.81	0.9014	32.71	0.9092

Table 5
The average PSNR (dB) and SSIM values of images in Set14 for up-scaling × 3 by using different SR approaches.

Image	Method															
	Zeyde et al. [9]		A+ [21]		SelfExSR [39]		SRCNN [40]		FSCNN [41]		VDSR [42]		Our SCSR		AMSRR-SCSR	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
baboon	23.52	0.5903	23.62	0.6070	23.53	0.6042	23.67	0.6118	23.71	0.6167	**23.78**	**0.6205**	23.58	0.6047	23.60	0.6076
barbara	26.76	0.7785	26.47	0.7759	26.96	**0.7911**	26.55	0.7778	26.42	0.7759	26.21	0.7805	26.84	0.7844	**26.98**	0.7895
bridge	25.02	0.6985	25.17	0.7113	24.08	0.6865	25.24	0.7168	25.28	0.7192	**25.38**	**0.7238**	25.06	0.7087	25.29	0.7196
coastguard	27.15	0.6499	27.27	0.6584	27.23	0.6608	**27.36**	0.6618	27.24	0.6621	27.35	**0.6722**	27.16	0.6618	27.31	0.6625
comic	23.96	0.7524	24.38	0.7771	24.41	0.7838	24.55	0.7826	24.72	0.7911	**25.11**	**0.8121**	24.05	0.7611	24.50	0.7838
face	33.53	0.8190	33.76	0.8268	33.71	0.8265	33.72	0.8264	33.83	0.8283	**33.95**	**0.8330**	33.61	0.8243	33.89	0.8302
flowers	28.43	0.8364	29.05	0.8516	29.10	0.8550	29.26	0.8540	29.45	0.8593	**30.01**	**0.8720**	28.50	0.8411	29.33	0.8575
foreman	33.19	0.9323	34.3	0.9428	34.22	0.9386	33.89	0.9405	34.12	0.9438	**35.00**	**0.9484**	33.18	0.9332	34.74	0.9463
lenna	33.00	0.8771	33.52	0.8842	33.51	0.8860	33.67	0.8858	33.85	0.8878	**33.97**	**0.8911**	33.11	0.8811	33.72	0.8870
man	27.90	0.7852	28.28	0.7987	28.34	0.8023	28.42	0.8020	28.53	0.8058	**28.78**	**0.8151**	27.93	0.7899	28.42	0.8027
monarch	31.10	0.9371	32.14	0.9463	32.10	0.9476	32.81	0.9480	33.44	0.9524	**34.69**	**0.9601**	31.16	0.9377	33.12	0.9517
pepper	34.07	0.8859	34.74	0.8914	34.82	0.8929	34.71	0.8912	34.89	0.8928	**35.30**	**0.8969**	33.91	0.8865	34.88	0.8935
ppt3	25.23	0.9087	26.09	0.9319	27.08	**0.9481**	27.04	0.9372	27.11	0.9398	**27.86**	**0.9616**	24.97	0.8962	27.03	0.9432
zebra	28.49	0.8421	28.98	0.8513	29.18	0.8527	29.29	0.8561	29.42	0.8569	29.50	0.8585	28.52	0.8481	**29.67**	**0.8638**
Average	28.67	0.8067	29.13	0.8182	29.16	0.8197	29.30	0.8209	29.43	0.8237	**29.77**	**0.8314**	28.69	0.8113	29.46	0.8242

and refining the high frequency components with the ridge regression coefficients. 4) The main complexity in optimization process is to update the weight matrix $\boldsymbol{\Theta}$. The complexity of updating is $\boldsymbol{O}\left(zd_4\kappa h\left(\log z/\log\kappa\right)\right)+\boldsymbol{O}\left(zd_4\left(\log z/\log\kappa\right)\right)$for one iteration. The total complexity is $\boldsymbol{O}\left(t_1 zd_4\kappa h\left(\log z/\log\kappa\right)\right)+\boldsymbol{O}\left(t_1 zd_4\left(\log z/\log\kappa\right)\right)$, where z is the number of patches extracted from $\boldsymbol{X}_t$, t_1 is the updating weight of the GLM g_o^e, and d_4 is the corresponding patch size.

Fig. 4. SR results of the image 'bird' from Set5 for upscaling ×2. Images from top to bottom and left to right: LR input image, reconstructed HR image by Zeyde et al. [9], A+ [21], SelfExSR [39], SRCNN [40], FSCNN [41], VDSR [42], our SCSR, and our proposed AMSRR-SCSR, as well as the ground truth HR image.

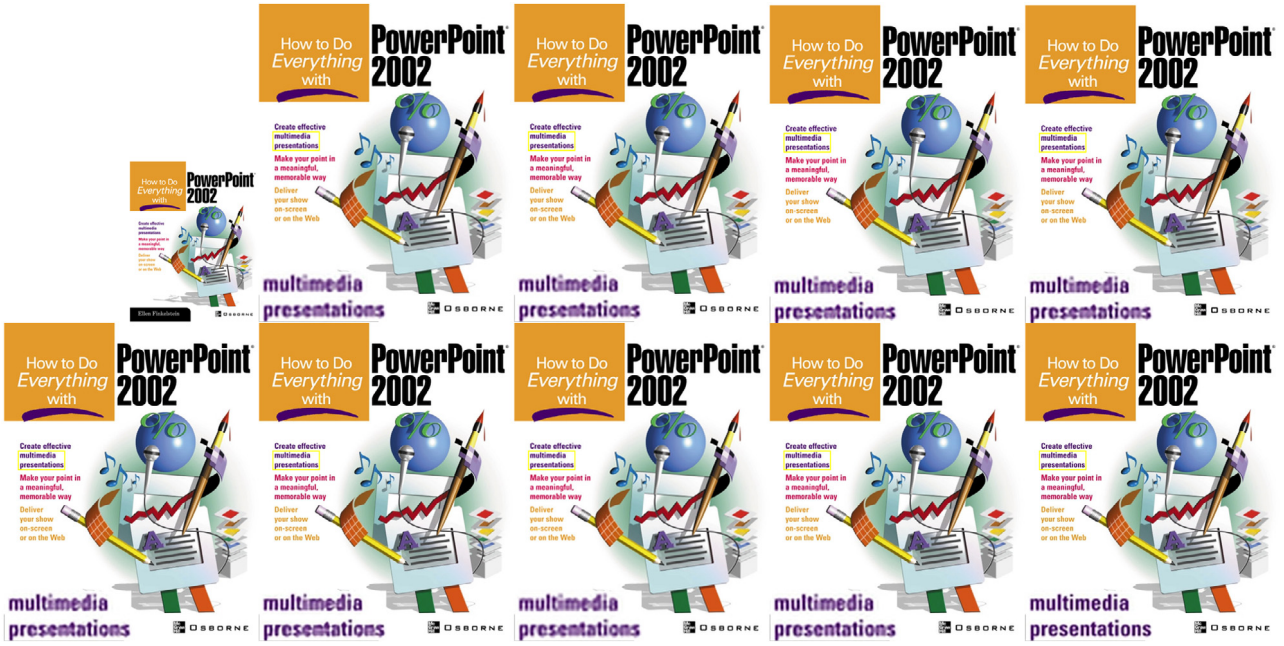

Fig. 5. SR results of the image 'ppt3' from Set14 for upscaling ×2. Images from top to bottom and left to right: LR input image, reconstructed HR image by Zeyde et al. [9], A+ [21], SelfExSR [39], SRCNN [40], FSCNN [41], VDSR [42], our SCSR, and our proposed AMSRR-SCSR, as well as the ground truth HR image.

5. Conclusion

In this paper, we present a novel joint SR algorithm to deal with the inherent defects of the internal SR and the external SR methods. The proposed ASSS adaptively generates the mixed samples based on input LR image and the proposed AMSRR model effectively learns the complementary information included in the mixed samples. Moreover, the final HR output image is optimized by the GLM and GRC. The qualitative and quantitative experimental results demonstrate that our algorithm reconstructs fine details and preserves sharp edges. Due to the fact that deep neural networks have powerful feature learning ability, the deep learning techniques have successfully been applied to image classification, acquisition and representation. We will research

Fig. 6. SR results of the image 'baby' from Set5 for upscaling ×3. Images from top to bottom and left to right: LR input image, reconstructed HR image by Zeyde et al. [9], A+ [21], SelfExSR [39], SRCNN [40], FSCNN [41], VDSR [42], our SCSR, and our proposed AMSRR-SCSR, as well as the ground truth HR image.

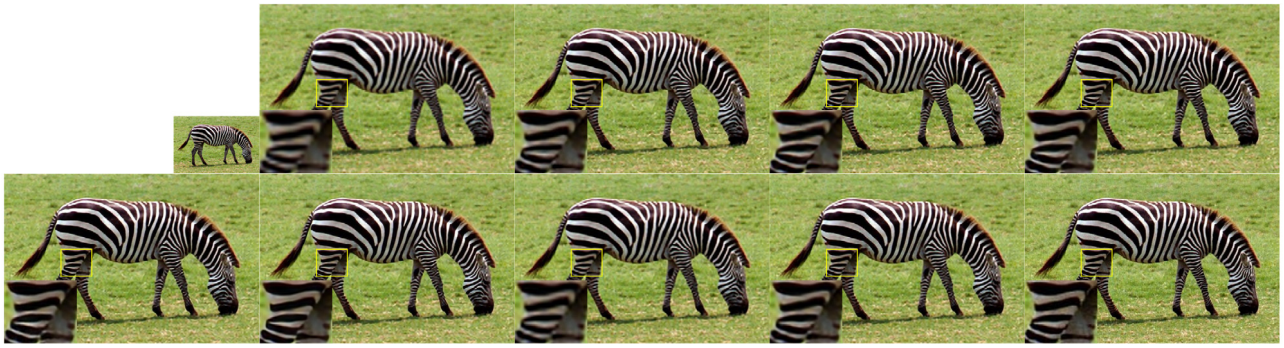

Fig. 7. SR results of the image 'zebra' from Set14 for upscaling ×3. Images from top to bottom and left to right: LR input image, reconstructed HR image by Zeyde et al. [9], A+ [21], SelfExSR [39], SRCNN [40], FSCNN [41], VDSR [42], our SCSR, and our proposed AMSRR-SCSR, as well as the ground truth HR image.

the internal and external sample priors within a deep neural network framework in the future work.

Acknowledgments

The authors would like to thank the editors and the anonymous reviewers, whose comments helped to improve the paper greatly. This work was supported by the National Natural Science Foundation of China (No. 61461028), the Natural Science Foundation of Gansu Province (No. 1508RJZA092) and the Gansu Province Basic Research Innovation Group Project (No. 1506RJIA031).

References

[1] R.Y. Tsai, T. Huang, Multiframe image restoration and registration, in: Advances in Computer Vision and Image Processing, 1984, pp. 317–339.
[2] S. Farsiu, M.D. Robinson, M. Elad, P. Milanfar, Fast and robust multiframe super resolution, IEEE Trans. Image Process. 13 (2004) 1327–1344.
[3] X. Li, Y. Hu, X. Gao, D. Tao, B. Ning, A multi-frame image super-resolution method, Signal Process. 90 (2010) 405–414.
[4] H. Yin, S. Li, L. Fang, Simultaneous image fusion and super-resolution using sparse representation, Inform. Fusion 14 (2013) 229–240.
[5] X. Li, M.T. Orchard, New edge-directed interpolation, IEEE Trans. Image Process. 10 (2001) 1521–1527.
[6] W. Tam, C. Kok, W. Siu, Modified edge-directed interpolation for images, J. Electron. Imaging 19 (2010) 13011.
[7] Z. Lin, H.Y. Shum, Fundamental limits of reconstruction-based superresolution algorithms under local translation, IEEE Trans. Pattern Anal. Mach. Intell. 26 (2004) 83–97.
[8] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution via sparse representation, IEEE Trans. Image Process. 19 (2010) 2861–2873.
[9] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: Proceedings of the 7th International Conference on Curves and Surfaces, 2010, pp. 711–730.
[10] W. Dong, L. Zhang, G. Shi, X. Wu, Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization, IEEE Trans. Image Process. 20 (2011) 1838–1857.
[11] S. Wang, L. Zhang, Y. Liang, Q. Pan, Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2216–2223.
[12] H. Zhang, Y. Zhang, H. Li, T. Huang, Generative Bayesian image super resolution with natural image prior, IEEE Trans. Image Process. 21 (2012) 4054–4067.
[13] W. Dong, L. Zhang, G. Shi, X. Li, Nonlocally centralized sparse representation for image restoration, IEEE Trans. Image Process. 22 (2013) 1620–1630.
[14] L. He, H. Qi, R. Zaretzki, Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 345–352.
[15] T. Peleg, M. Elad, A statistical prediction model based on sparse representations for single image super-resolution, IEEE Trans. Image Process. 23 (2014) 2569–2582.
[16] L. Fang, S. Li, R.P. McNabb, Q. Nie, A.N. Kuo, C.A. Toth, et al., Fast acquisition and reconstruction of optical coherence tomography images via sparse representation, IEEE Trans. Med. Imaging 32 (2013) 2034–2049.
[17] W. Liu, S. Li, Sparse representation with morphologic regularizations for single image super-resolution, Signal Process. 98 (2014) 410–422.
[18] Z. Xiong, D. Xu, X. Sun, F. Wu, Example-based super-resolution with soft information and decision, IEEE Trans. Multimedia 15 (2013) 1458–1465.
[19] J. Xie, R.S. Feris, S. Yu, M. Sun, Joint super resolution and denoising from a single depth image, IEEE Trans. Multimedia 17 (2015) 1525–1537.
[20] R. Timofte, V. De, L. Van Gool, Anchored neighborhood regression for fast example-based super-resolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1920–1927.

[21] R. Timofte, V. De, L. Van Gool, A+: adjusted anchored neighborhood regression for fast super-resolution, in: Proceedings of the Asian Conference on Computer Vision, 2014, pp. 111–126.
[22] S. Schulter, C. Leistner, H. Bischof, Fast and accurate image upscaling with super-resolution forests, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3791–3799.
[23] Y. Zhang, Y. Zhang, J. Zhang, Q. Dai, CCR: Clustering and collaborative representation for fast single image super-resolution, IEEE Trans. Multimedia 18 (2016) 405–417.
[24] D. Glasner, S. Bagon, M. Irani, Super-resolution from a single image, in: Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 349–356.
[25] C. Yang, J. Huang, M. Yang, Exploiting self-similarities for single frame super-resolution, in: Proceedings of the Asian Conference on Computer Vision, 2010, pp. 497–510.
[26] Z. Zhu, F. Guo, H. Yu, C. Chen, Fast single image super-resolution via self-example learning and sparse representation, IEEE Trans. Multimedia 16 (2014) 2178–2190.
[27] J. Yang, Z. Lin, S. Cohen, Fast image super-resolution based on in-place example regression, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1059–1066.
[28] Z. Wang, Z. Wang, S. Chang, J. Yang, . Yang, T. Huang, A joint perspective towards image super-resolution: unifying external- and self-examples, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 2014, pp. 596–603.
[29] Z. Wang, Y. Yang, Z. Wang, S. Chang, J. Yang, T.S. Huang, Learning super-resolution jointly from external and internal examples, IEEE Trans. Image Process. 24 (2015) 4359–4371.
[30] M. Muja, Fast approximate nearest neighbors with automatic algorithm configuration, in: Proceedings of the International Conference on Computer Vision Theory and Application Vissapp, 2009, pp. 331–340.
[31] J. Yang, Z. Wang, Z. Lin, S. Cohen, T. Huang, Coupled dictionary training for image super-resolution, IEEE Trans. Image Process. 21 (2012) 3467–3478.
[32] J. Zhang, D. Zhao, W. Gao, Group-based sparse representation for image restoration, IEEE Trans. Image Process. 23 (2014) 3336–3351.
[33] M. Aharon, M. Elad, A.M. Bruckstein, K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (2006) 4311–4322.
[34] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution as sparse representation of raw image patches, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[35] I.T. Jolliffe, Principal component analysis, Technometrics 44 (2005) 594–609.
[36] R. Rubinstein, M. Zibulevsky, M. Elad, Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit, CS, Technion, Haifa, Israels, Tech. Rep. 1–15, Apr. 2008.
[37] A. Buades, B. Coll, J.M. Morel, A non-local algorithm for image denoising, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 60–65.
[38] W. Zhou, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: From error visibility to structural similarity, IEEE Trans. Image Process. 13 (2004) 600–612.
[39] J.B. Huang, A. Singh, N. Ahuja, Single image super-resolution from transformed self-exemplars, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5197–5206.
[40] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2016) 295–307.
[41] C. Dong, C.C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: Proceedings of the European Conference on Computer Vision, 2016, pp. 391–407.
[42] J. Kim, J.K. Lee, K.M. Lee, Accurate image super-resolution using very deep convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1646–1654.
[43] M. Muja, D.G. Lowe, Scalable nearest neighbor algorithms for high dimensional data, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2014) 2227–2240.