Bayesian Anchored Neighborhood Regression for Single-Image Super-Resolution

Yinggan Tang[1,2] · Ailian Fan[1]

Abstract

Anchored neighborhood regression (ANR) single-image super-resolution (SISR) method demonstrates great superiority in speed and performance. However, ridge regression adopted by ANR didn't take the *prior* knowledge of regression parameters into account, which easily leads to over-fitting and is not robust to noise. To mitigate this problem, a Bayesian ANR method is proposed for SISR, called B-ANR. B-ANR uses sparse Bayesian regression model to build the mapping relationship between the low resolution (LR) image patch and its neighbors. The parameters of Bayesian regression model as well as its variance are estimated using maximum *a posterior*. Since our proposed B-ANR is deduced in the probabilistic framework, it has stronger generalization performance and robustness to noise. Experimental results verify that our method is superior to ANR in performance.

1 Introduction

The primary purpose of single image super-resolution (SISR) is to restore a high-resolution (HR) image from a single low-resolution (LR) image. Since HR image contain abundant high-frequency detailed information, it can facilitate many computervision tasks. Image SR techniques has been extensively adopted to enhance the

✉ Yinggan Tang
ygtang@ysu.edu.cn

Ailian Fan
2319898815@qq.com

1 School of Electrical Engineering, Yanshan University, The Westward Direction of Hebei Street, Qinhuangdao 438, Hebei Province, China

2 Key Lab of Industrial Computer Control Engineering of Hebei Province, Yanshan University, The Westward Direction of Hebei Street, Qinhuangdao 438, Hebei Province, China

image's resolution to overcome the limitation of low-quality imaging devices [24, 27, 29]. For example, the LR images such as positron emission tomography (PET) images [16], magnetic resonance images (MRI) [3], can be reconstructed using image SR technique to improve its resolution, leading to an improved diagnosis accuracy.

Due to it's ill-posed characteristic, image SR still remains a challenging problem. Over the past few decades, many researchers have conducted extensive studies on SR and a large amount of methods have been proposed in the literature. These approaches can be categorized into two broad categories, interpolation-based SR methods and learning-based SR methods.

Interpolation-based SR methods [15, 20, 23, 35] use different interpolation approaches, such as nearest neighbor interpolation, bilinear and bicubic interpolation, to approximate the unknown pixels in the HR grid. Interpolation-based methods are difficult to recover lost high-frequency details due to their low-pass filtering characteristics, resulting in excessively smooth HR images. Even more, jagged artifacts tend to arise at the edges of the reconstructed HR image.

Freeman et al. [11] believed that it is possible to recover lost details in the LR image by learning from an external training set and proposed an example-based SR method. The basic idea of example-based SR method is to establish a mapping relationship between HR and LR image patches. Following Freeman's idea, many learning-based SR methods had been proposed, which can be divided into three categories: neighborhood embedding (NE) based methods, sparse coding based methods and regression-based methods. NE SR method was presented by Chang et al. in [1], which assumes that the two manifolds in the HR and LR space have the same geometric structure, and the target LR patch can be represented by a linear combination of its neighborhood patches. NE SR method can recover a HR image with less complexity of computation, however it's SR performance still remains great room for improvement. Li et al. [21] proposed an neighbor embedding SR method based on local preservation constraints (LPC), which explicitly emphasize the local consistency on the two manifolds, resulting in reduced blurring and artifacts. Gao et al. [12] divided the entire training set into several subsets using histogram of gradient clustering (HOG), and then, the method finds neighbors for a given patch in the corresponding subset using Robust-SL0 algorithm. Chen et al. [2] use kernel trick to learn the nonlinear mapping between LR and HR image patches. Yu et al. [38] presented an improved NE method by enlarging neighborhood range to all neighbors.

Another class of learning-based SR methods is sparse coding based methods, which was originally proposed by Yang et al. [36]. The sparse coding based SR (ScSR) method encodes the LR or HR image patches in the training set as a linear combination of a LR or HR dictionary, respectively. Under the premise that the LR and HR patches have the same sparse representation coefficients, the target HR patch for a given LR patch can be reconstructed by linearly combining the sparse representation coefficients of the given LR patch and HR dictionary. ScSR method has demonstrated it's excellent performance and possesses robustness to noise. However, several disadvantages are still associated with ScSR. The sparse coding of LR patch is time-consuming. To speed up the sparse coding process, Zeyde et al. [39] proposed to use principal component analysis (PCA) and orthogonal matching pursuit (OMP) for LR patches sparse coding. As well, a coupled dictionary method [37], Bayesian

dictionary [14] learning method had also been proposed for addressing this problem. In addition, the hypothesis that the LR and HR patches have the same sparse representation coefficients is not fully reasonable, which leads to performance drop. In [6], Dong et al. incorporate the difference of sparse coefficients between LR and HR features into the sparse coding of process LR patches.

Regression-based methods also play an essential role in learning-based methods. Kim et al. [18] first adopted the kernel ridge regression to learn the mapping relationship between LR and HR patches. The sparse solution is derived by combining kernel matching pursuit with gradient descent. To acquire sharper edge information in reconstructed HR image, Tang et al. [28] utilized multiple matrix-valued kernel regression for nonlinear mapping. In [22], the linear kernel regression is employed to establish the regression relationship between HR image and fuzzy HR image patches. Furthermore, a kernel regression method with an adaptive Gaussian kernel was proposed in [26]. To overcome over-fitting problem, Ogawa et al. [25] presented a new SR method which combines Gaussian mixture model (GMM) and partial least squares (PLS) regression. The Gaussian process regression (GPR) [13] is introduced into image SR to improve the generalization ability of regression models.

Apart from the those learning-based SR methods mentioned above, Timoft et al. [31] proposed ANR for image SR. This method combines NE method with sparse coding method, which exhibits outstanding SR performance. So far, many improved ANR variants have been proposed. With the aim of exploring the nonlinear relationship between the LR and HR spaces, Jiang et al. [17] exploited the local geometric *prior* to regularize the neighborhood regression, and meanwhile, the quality of reconstructed HR image is further enhanced using non-local self-similarity. In [40], MI-KSVD method is introduced into dictionary training, and the neighborhoods of a given LR patch is found according to the coherence between dictionary atoms and training samples. In [41], a modified ANR method was proposed using clustering and collaborative representation, in which the whole training set is divided into 1024 clusters using k-means clustering algorithm and each center of clustering represents an atom. In [42], Zhang et al. introduced a collaborative representation cascade (CRC) method to learn the multilayered mapping model between the LR and HR feature space.

With the continuous development of deep learning, more and more researchers have started to apply deep learning to SR reconstruction. Deep learning-based SR methods extract the high-level abstract features from LR/HR images in the train set through multiple-layer convolutional operation and nonlinear transformations, and then reconstruct a HR image by aggregating the extracted features. In 2014, Dong et al. proposed an image SR method based on deep convolutional networks (called SRCNN) [4]. After SRCNN, many deep learning models for image SR had been proposed. To speed up the convergence of SRCNN and obtain global optimal solution, Wang et al. introduced deep and shallow convolution networks [34] for image SR. Tian et al. proposed a lightweight enhanced SRCNN (LESRCNN) [30], which is more computationally frugal than SRCNN. Esmaeilzehi et al. [9] described image SR reconstruction as a three-priority optimization problem and developed an ultra-lightweight convolutional neural network for image SR. In [7], a new residual depth network called ComNet was proposed to solve image SR problem. In [8], Esmaeilzehi et al. proposed a new image restoration network, using pixel by pixel feature attention

to calibrate the feature maps extracted by the network. Compared with most existing SR methods, deep learning-based image SR methods have demonstrated great advantage in performance. Though it is, the deep learning-based SR methods have a great deal of demanding on computational resource and power. In addition, the training of deep learning models are time-consuming.

Though ANR and its variants have demonstrated excellent performance, however, there sill has certain disadvantages. In ANR and it's variants, the ridge regression is adopted to establish the relationship between the target patch and its neighbors in the LR/HR feature space. The coefficients of the ridge regression model are obtained by the least square method. The least square is not the optimal estimator because it does not take the *prior* distribution of the coefficients into account, which make it less robust to noise and lead to over-fitting. To address the above issues, we propose a new image SR method called B-ANR. B-ANR uses sparse Bayesian regression model to establish the mapping relationship between the LR image patch and its neighbors. The parameter's of Bayesian regression model and its variance are obtained by maximum a *posterior*, which is more accurate than the least square. Since our B-ANR is derived in the probabilistic framework, it performs better in terms of generalization and robustness to noise. Experimental results verify that our approach is superior to ANR.

The remainder of this paper is organized as follows. Section 2 characterizes the details our proposed B-ANR method, Sect. 3 presents the experimental results and analysis, Sect. 4 discusses the advantages and disadvantages of the proposed method and deep learning methods, and Sect. 5 derives conclusions to conclude.

2 The Proposed Method

In this section, the proposed method is detailed explained. For the convenience of narration, we first recall the fundamentals of ANR and then explain our proposed B-ANR model.

2.1 Anchored Neighborhood Regression(ANR)

In [31], Timofte et al. introduced an ANR based SR method, which can be viewed as a combination of sparse coding method and NE SR method. ANR starts from a learned LR and HR sparse dictionary pair. ANR takes each atom $\mathbf{d}_j$ in the learned LR dictionary (via the method of [39]) as an anchor point. Then, one finds K nearest neighbors N_l for anchor $\mathbf{d}_j$ in the LR dictionary based on the correlation between the dictionary atoms. Denotes N_l as

$$N_l = \left\{ d_l^1, d_l^2, ..., d_l^K \right\} \tag{1}$$

Using the same assumptions as [1], the input LR image patch x_t can be approximated as a linear combination of its neighbors N_l and the weight coefficients w, which can

be mathematically expressed as

$$x_t = N_l w + \epsilon. \tag{2}$$

In Eq. (2), the optimal coefficients w are found by using ridge regression, that is,

$$\min_{w} \parallel x_t - N_l w \parallel_2^2 + \lambda \parallel w \parallel_2 \tag{3}$$

where λ is a regularization parameter controlling the balance between the reconstruction error of x_t and the smoothness of w. The closed algebraic solution for ridge regression problem (3) is

$$w = (N_l^T N_l + \lambda I)^{-1} N_l^T x_t \tag{4}$$

Then, the HR output patch can be calculated using the same weight coefficients on HR neighborhood N_h as

$$y_t = N_h w \tag{5}$$

where y_t is the HR output patch and N_h is the HR neighborhood corresponding to N_l. Once the neighborhood is defined, an individual projection matrix P_j is calculated based on the neighborhood for each anchor d_j as

$$P_j = N_h (N_l^T N_l + \lambda I)^{-1} N_l^T. \tag{6}$$

After building the projection matrix for each anchor d_j, one can use the projection matrices to super-resolve a LR image. The process is described as following. First, for each input LR patch x_i, we calculate the correlation between x_i and all the atoms in the LR dictionary as

$$\text{corr}_j = \langle x_i, d_j \rangle. \tag{7}$$

Let

$$m = \max_{j} \text{corr}_j,$$

then, the m-th atom in the LR dictionary is selected as the neighbor atom of the input LR patch x_i. Then, the corresponding HR output y_i patch is reconstructed using the m-th projection matrix as

$$y_i = P_m x_i. \tag{8}$$

2.2 Bayesian Anchored Neighborhood Regression

In ANR [31], ridge regression is adopted to build the relationship between the target patch and its neighbors. The coefficients of the ridge regression model are calculated by the least squares method. From a probabilistic perspective, the least squares method is equal to maximum likelihood estimation when the regression error follows a Gaussian distribution. Maximum likelihood estimation does not take the parameter *prior* into
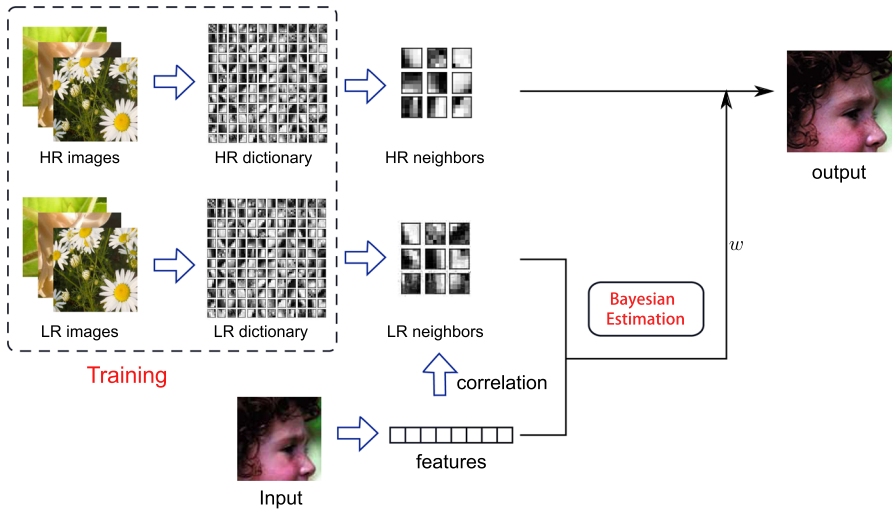
Fig. 1 The block diagram of SR algorithm proposed in this paper

account, leading to less robustness to noise and over-fitting. To build an accurate model to characterize the relationship between the input image patch and its neighbors, we propose B-ANR, which introduces the *prior* of the weight coefficients into the regression process. Meanwhile, we estimate the model weight coefficients and it's variance using maximum a *posterior* estimate method. The overall framework of the proposed B-ANR algorithm is shown in Fig. 1. The following will provide a detailed introduction to our proposed method.

Given a target LR patch x_t and it's nearest neighbors N_l, in the ANR framework, x_t can be represented as

$$x_t = \sum_{k=1}^{K} w_k d_l^k + \epsilon \tag{9}$$

where ϵ represents mean-zero Gaussian with variance σ^2. Let $w = (w_1, ... w_K)^{\mathrm{T}}$ denote weight the coefficient matrix associated with the target patch, then, x_t can be written in matrix form as

$$x_t = N_l w + \epsilon. \tag{10}$$

The likelihood function of the target vector x_t could be expressed as

$$p(x_t | w, \sigma^2) = (2\pi)^{-N/2} \sigma^{-N} \exp\left\{ -\frac{\| x_t - N_l w \|^2}{2\sigma^2} \right\} \tag{11}$$

Directly estimating the weight coefficients w and variance σ^2 using maximum likelihood easily leads to over-fitting. To solve this issue, the prior distribution of w is incorporated into the estimation of w. The prior distribution of w is assumed to be a

zero-mean Gaussian distribution with variance of α_k^{-1} as

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = (2\pi)^{-K/2} \prod_{k=1}^{K} \sqrt{\alpha_k} \exp\left(-\frac{\alpha_k \boldsymbol{w}_k^2}{2}\right) \tag{12}$$

Using the Bayesian formula, the weight coefficients *posterior* $p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2)$ can be written as

$$p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2) = \frac{p(\boldsymbol{x}_t|\boldsymbol{w}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})}{p(\boldsymbol{x}_t|\boldsymbol{\alpha}, \sigma^2)} \tag{13}$$

Substituting Eqs. (11) and (12) into (13), and after some manipulations, one can get

$$p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2) = (2\pi)^{-N/2} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left\{-\frac{\|\boldsymbol{w} - \boldsymbol{\mu}\|^2}{2\boldsymbol{\Sigma}}\right\} \tag{14}$$

by ignoring the denominator of Eq. (13). In Eq. (14)

$$\boldsymbol{\Sigma} = (\boldsymbol{A} + \sigma^{-2} \boldsymbol{N}_l^T \boldsymbol{N}_l)^{-1}$$
$$\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \boldsymbol{N}_l^T \boldsymbol{x}_t \tag{15}$$

with $\boldsymbol{A} = \text{diag}(\alpha_1, \alpha_2, ...\alpha_k)$. The optimal weight coefficients $\boldsymbol{w}^*$ would be one that maximizes $p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2)$, i.e.,

$$\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2) \tag{16}$$

The optimal solution of problem (15) can be obtained by taking derivation of $p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2)$ with $\boldsymbol{w}$ setting it to zero. Since $p(\boldsymbol{w}|\boldsymbol{x}_t, \boldsymbol{\alpha}, \sigma^2)$ is a Gaussian function, therefore, the optimal solution of (15) is

$$\boldsymbol{w}^* = \boldsymbol{\mu}. \tag{17}$$

However, in Eq. (17), the hyper-parameter $\boldsymbol{\alpha}$ and σ are unknown. Therefore, one should make an estimation for them. This can be done using type-II maximum marginal likelihood procedure. The marginal likelihood is

$$p(\boldsymbol{x}_t|\boldsymbol{\alpha}, \sigma^2) = \int p(\boldsymbol{x}_t|\boldsymbol{w}, \sigma^2) p(\boldsymbol{w}|\boldsymbol{\alpha}) \mathrm{d}\boldsymbol{w} \tag{18}$$

or its logarithm

$$\mathcal{L}(\boldsymbol{\alpha}) = \ln p(\boldsymbol{x}_t|\boldsymbol{\alpha}, \sigma^2) = -\frac{1}{2}\left[N\ln(2\pi) + \ln|\boldsymbol{C}| + \boldsymbol{x}_t^T \boldsymbol{C}^{-1} \boldsymbol{x}_t\right] \tag{19}$$

where

$$C = \sigma^2 I + N_l A^{-1} N_l^T \tag{20}$$

To improve sparsity and reduce computational requirements, we simplify Eq. (20) using the same method as in [33], updating only a single α_i rather than the entire vector $\boldsymbol{\alpha}$ per iteration. Rewritten C as

$$
\begin{aligned}
C &= \sigma^2 I + \sum_{k \neq i} \alpha_k^{-1} d_l^k d_l^{kT} + \alpha_i^{-1} d_l^i d_l^{iT} \\
&= C_{-i} + \alpha_i^{-1} d_l^i d_l^{iT}
\end{aligned} \tag{21}
$$

where C_{-i} is C after removing the contribution of the basis vector i. Using the Woodbury identity for Eq. (21), it is possible to obtain

$$C^{-1} = C_{-i}^{-1} - \frac{C_{-i}^{-1} N_l^i N_l^{iT} C_{-i}^{-1}}{\alpha_i + N_l^{iT} C_{-i}^{-1} N_l^i} \tag{22}$$

and using the determinant identity, we get

$$|C| = |C_{-i}||1 + \alpha_i^{-1} N_l^{iT} C_{-i}^{-1} N_l^i| \tag{23}$$

Substituting Eqs. (22) and (23) into (19), then $\mathcal{L}(\boldsymbol{\alpha})$ can be written as

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{\alpha}) &= \mathcal{L}(\boldsymbol{\alpha}_{-i}) + \frac{1}{2}\left[\log \alpha_i - \log(\alpha_i + s_i) + \frac{q_i^2}{\alpha_i + s_i}\right] \\
&= \mathcal{L}(\boldsymbol{\alpha}_{-i}) + \ell(\alpha_i)
\end{aligned} \tag{24}
$$

For the convenience of simplicity, let

$$s_i = d_l^{iT} C_{-i}^{-1} d_l^i, \quad q_i = d_l^{iT} C_{-i}^{-1} x_t \tag{25}$$

Then, the objective function $\mathcal{L}(\boldsymbol{\alpha})$ function is decomposed into $\mathcal{L}(\boldsymbol{\alpha}_{-i})$ and $\ell(\alpha_i)$, in which the terms in α_i are now conveniently isolated. Obviously, the unique maximum value of $\mathcal{L}(\boldsymbol{\alpha})$ is [10]

$$
\begin{aligned}
\alpha_i &= \frac{s_i^2}{q_i^2 - s_i} \quad if \quad q_i^2 > s_i \\
\alpha_i &= \infty \qquad if \quad q_i^2 \leq s_i
\end{aligned} \tag{26}
$$

From Eq. (26) we can determine which basis vectors should be included in the model and which ones should be excluded. Once hyper-parameters α_i is updated using Eq.

(26), the parameters q_i, s_i μ and Σ are updated efficiently. As in [33], q_i, s_i can be updated as the following formula

$$S_i = d_l^{iT} B d_l^i - d_l^{iT} B N_l \Sigma N_l^T B d_l^i \tag{27}$$

$$Q_i = d_l^{iT} B x_t - d_l^{iT} B N_l \Sigma N_l^T B x_t \tag{28}$$

$$s_i = \frac{\alpha_i S_i}{\alpha_i - S_i} \tag{29}$$

$$q_i = \frac{\alpha_i Q_i}{\alpha_i - S_i} \tag{30}$$

where $B = \sigma^{-2} I$. Here N_l and Σ contain only the feature vectors currently included in the model, which is normally a very minor part of the whole K, thus reducing the computational effort considerably.

According to the nature of the marginal likelihood function, we use Eq. (26) as the decision criteria to add or delete the candidate nearest-neighbor features for maximizing the marginal likelihood objective function. In summary, the process of determining the weight coefficients w is shown in Algorithm 1.

In the reconstruction phase, we assume that LR space and HR space have the same geometry, so the resulting weight coefficients w can be used to construct the HR image. The HR image patch is calculated by using the HR neighborhood N_h corresponding to the LR neighborhood N_l and the weight coefficients w. The SR reconstruction of the image is carried out as follows:

$$y_t = N_h w \tag{31}$$

where y_t is the HR output patch and $N_h = (d_h^1, ... d_h^K)$ is the HR neighborhood.

Algorithm 1 Maximum a *posterior* estimation of w

Require: target vector x_t, a set of neighbors N_l.
Ensure: the weight coefficients w.
1: Initialize α and σ^2, explicitly calculate Σ and μ using (15), and initialize s_i and q_i for all K basic vector in N_l using (29), (30);
2: **while** not converged **do**
3:    Select a candidate basis vector d_l^i from the set of all K;
4:    Define $\theta_i = q_i^2 - s_i$;
5:    **if** $\theta \geq 0$ **then**
6:       $\alpha_i \leq \infty$, re-estimate α_i;
7:       $\alpha_i = \infty$, add d_l^i to the model with updated α_i;
8:    **else**
9:       $\alpha_i \leq \infty$, delete d_l^i from the model and set $\alpha_i = \infty$;
10:   **end if**
11:   Update σ^2 according to $\frac{\|x_t - N_l w\|^2}{(N - K + \sum_i \alpha_i \Sigma_{ii})}$ (Σ_{ii} is the i-th diagonal element of Σ)[32];
12: Compute Σ, μ using (15);
13: Compute s_k and q_k using (29), (30).
14: **end while**
15: Output the weight coefficients $w^* = \mu$.

---

3 Experimental Results and Analyses

To verify the validity of the proposed method, super-resolution experiments on several datasets are conducted. Meanwhile, the experimental results obtained by the method of this paper are compared with those obtained by several most advanced SR methods SR methods like Yang et al. [36], Zeyde et al. [39], ANR [31] , NE + LS, NE + LLE [1]. We use peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as evaluation criteria for measuring the performance of image reconstruction. PSNR represents the ratio of the maximum possible power of a signal to the power of the destructive noise that affects the accuracy of its representation. SSIM is a measure of similarity between two images.

3.1 Experimental Settings

Training Set Learning-based method requires a large amount of image patch pairs for training. In this paper, we use the same training set as Yang et al. [36], Zeyde et al. [39], ANR [31]. We use the images in the training set as HR images and the corresponding LR images are acquired by bicubic interpolation. We divide the HR images into HR image patches of 9×9 pixel size with an overlap of 6 pixels between neighboring patches. The size of the corresponding LR image patches is set to 3×3 pixels with an overlap of 2 pixels.

Testing Set To better compare with other methods, we uniformly use Set5 , Set14 and BSD100 as test sets, including 5, 14 and 100 common images, respectively. We perform bicubically interpolation on the images in the test set to obtain LR images for testing and segment the LR images into LR image patches of the same size as in the training set.

Features As the human eye is more susceptible to luminance constituents, we extract features from the luminance channel of the image patches. We use the identical features as Zeyde et al. [39] and Yang et al. [36], who started from the first- and second-order gradients of luminance and performed PCA to reduce the dimension and project the features into a low-dimensional subspace.

Since the features used in the LR patches are not reflective of absolute luminance, we subtract the mean from the luminance-based feature vector for each HR patch. When constructing the target HR image, the patches generated by the SR process are added to the average luminance value of the corresponding LR patches.

Dictionaries In our experiments, We train the dictionaries using the same external images as in Zeyde et al. [39]. For dictionary learning, we use the K-SVD/OMP learning method of Zeyde et al. [39]. To facilitate qualitative comparison with other methods, we uniformly use a dictionary of size 1024.

Neighborhoods As we are calculating weights and reconstructing them based on the neighborhood of the input patch in the dictionary, the method of neighborhood selec-

Table 1 Performance of different methods in terms of PSNR and SSIM per image on Set5 dateset with magnification factor ×3

Images	Bicubic	Yang et al.	Zeyde et al.	ANR	NE + LS	NE + LLE	Ours
baby	33.91	34.29	35.08	35.13	34.95	35.06	35.22
	0.9040	0.9046	0.9211	0.9226	0.9195	0.9210	0.9218
bird	32.57	34.10	34.56	34.59	34.36	34.56	34.83
	0.9262	0.9399	0.9428	0.9494	0.9466	0.9482	0.9463
butterfly	24.04	25.58	25.94	25.90	25.83	25.75	26.17
	0.8217	0.8620	0.8771	0.9718	0.8762	0.8706	0.8824
head	32.88	33.17	33.56	33.63	33.53	33.60	33.72
	0.8004	0.8026	0.8205	0.8241	0.8201	0.8228	0.8238
woman	28.56	29.93	30.36	30.32	30.20	30.21	30.56
	0.8897	0.9043	0.9177	0.9170	0.9165	0.9162	0.9152
average	30.39	31.42	31.90	31.92	31.78	31.60	32.09
	0.8684	0.8827	0.8969	0.8970	0.8958	0.8958	0.8979

All methods were trained on the same images, except for Bicubic interpolation and Yang et al. [39]. (1022 atoms), which share a dictionary of 1024 atoms

tion is important. For neighborhood selection, we choose neighborhoods on the basis of the correlation between dictionary atoms and image patches. Different sizes of neighborhoods also have different effects on performance, and when comparing methods, we choose the best neighborhood value for each method. In the learning dictionary, the best neighborhood for NN + LS is 12, for NE + LLE [1] and ANR [31] is 40, the best neighborhood chosen for our method is 80.

3.2 Experimental Results

Timofte et al. [31] demonstrate that for the proper set of parameters, such as feature representation, neighborhood size, most current neighborhood embedding methods are capable of achieving a certain quality of performance. Tables 1 and 2 display the PSNR and the SSIM values for each test image at magnification factor ×3 for the Set5, Set14 and BSD100 dateset. In addition to deep learning-based SR method, it is evident that our results obtain the highest PSNR and SSIM values compared to other methods, both PSNR and SSIM have different degrees of improvement. In Table 3, we show the average PSNR and SSIM values for the amplification factors ×2, ×3, ×4 on the Set5, Set14, BSD100 datasets. With the exception of TPCNN [4], our proposed method is the best among the others in terms of quality, with an average improvement PSNR of 0.14 dB (BSD100, ×2) to 0.09 dB (Set5, ×2) over the ANR [31] method. Different amplification factors of SSIM on Set5, Set14 and BSD100 data sets have an average increase of 0.0023 compared to ANR [31].

In order to demonstrate the superior visual quality of our method compared to other methods, we compare our method with others from a visual perspective in Figs. 2, 3, 4, 5 and 6. We select 'butterfly' in Set 5 and 'baboon', 'zebra', 'foreman' and 'pep-

Table 2 Performance of different methods in terms of PSNR and SSIM per image on Set14 dataset with magnification factor ×3

Images	Bicubic	Yang et al.	Zeyde et al.	ANR	NE + LS	NE + LLE	Ours
baboon	23.21	23.47	23.52	23.56	23.51	23.55	23.62
	0.5443	0.5882	0.5902	0.5994	0.5910	0.5973	0.5983
barbara	26.25	26.39	26.76	26.69	26.71	26.74	26.81
	0.7531	0.7633	0.7816	0.7811	0.7794	0.7813	0.7850
bridge	24.40	24.82	25.02	25.01	24.92	24.98	25.10
	0.6675	0.7088	0.7126	0.7169	0.7089	0.7149	0.7195
coastguard	26.55	27.01	27.15	27.07	27.05	27.07	27.18
	0.6150	0.6396	0.65411	0.6570	0.6514	0.6557	0.6588
comic	23.12	23.90	23.90	24.04	23.92	23.98	24.11
	0.6999	0.7576	0.7561	0.7621	0.7543	0.7584	0.7633
face	32.82	33.11	33.53	33.61	33.50	33.57	33.59
	0.7985	0.8013	0.8197	0.8234	0.8189	0.8217	0.8237
flowers	27.23	28.25	28.43	28.49	28.35	28.38	28.52
	0.8027	0.8312	0.8388	0.8416	0.8365	0.8390	0.8429
foreman	31.16	32.00	33.18	33.21	33.18	33.19	33.46
	0.9059	0.9131	0.9294	0.9302	0.9297	0.9294	0.9333
lenna	31.68	32.64	33.00	33.08	33.98	33.01	33.23
	0.8583	0.8650	0.8782	0.8805	0.8778	0.8791	0.8815
man	27.01	27.76	27.90	27.92	27.85	27.87	28.01
	0.7500	0.7760	0.7868	0.7900	0.7855	0.7881	0.7918
monarch	29.43	30.71	31.10	31.09	30.94	30.95	31.26
	0.9198	0.9292	0.9383	0.9377	0.9365	0.9362	0.9369
pepper	32.38	33.31	34.06	33.81	33.91	33.80	34.21
	0.8721	0.8692	0.8882	0.8872	0.8868	0.8865	0.8901
ppt3	23.71	24.98	25.23	25.03	25.15	24.94	25.29
	0.8838	0.9078	0.9210	0.9146	0.9195	0.9136	0.9225
zebra	26.63	27.95	28.49	28.43	28.25	28.31	28.63
	0.7961	0.8298	0.8427	0.8449	0.8387	0.8447	0.8482
average	27.54	28.31	28.67	28.65	28.59	28.60	28.77
	0.7762	0.7986	0.8098	0.8120	0.8082	0.8103	0.8142

All methods were trained on the same images, except for Bicubic interpolation and Yang et al. [39]. (1022 atoms), which share a dictionary of 1024 atoms

per' in Set 14 with the magnified ×3. From the figure it can be seen that the Bicubic interpolation method has the worst visual results, producing blurred edge information. Yang et al. [36] reconstructed HR images based on dictionary learning, which also resulted in some ringing and artefact effects. Zeyde et al. [39] used PCA dimensionality reduction to greatly reduce the computational complexity, but it did not have good information about the details of the recovered. NE+LS produces serrated edge information and annoying texture details irritating texture details. NE+LLE method can

Table 3 Performance of different methods in terms of PSNR and SSIM at different magnifications for Set5, Set14 and BSD100 dateset

Dataset	Scale	Bicubic	Zeyde et al.	ANR	NE + LS	NE + LLE	TPCNN [9]	Ours
Set5	×2	33.66	35.78	35.83	35.66	35.77	37.18	36.02
		0.9299	0.9494	0.9500	0.9487	0.9491	0.9589	0.9514
	×3	30.39	31.90	31.92	31.78	31.84	33.41	32.09
		0.8684	0.8969	0.8970	0.8958	0.8958	0.9189	0.8979
	×4	28.42	29.69	29.69	29.55	29.61	31.15	29.85
		0.8108	0.8433	0.8426	0.8401	0.8408	0.880	0.8468
Set14	×2	30.23	31.81	31.80	31.69	31.76	32.98	31.96
		0.8700	0.8998	0.9016	0.8986	0.9005	0.9135	0.9030
	×3	27.54	28.67	28.65	28.59	28.60	29.88	28.79
		0.7762	0.8098	0.8120	0.8082	0.8103	0.8336	0.8142
	×4	26.00	26.88	26.85	26.81	26.81	28.12	26.98
		0.7054	0.7379	0.7393	0.7354	0.7371	0.7707	0.7420
BSD100	×2	29.56	30.77	30.80	30.68	30.59	31.67	30.89
		0.8431 8773		0.8782	0.8758	0.8743	0.8950	0.8811
	×3	27.21	27.98	27.96	27.91	27.84	28.68	27.96
		0.7388	0.7728	0.7743	0.7707	0.7685	0.7956	0.7770
	×4	25.96	16.57	26.58	26.51	26.38	27.18	26.62
		0.6674	0.6979	0.6990	0.6954	0.6900	0.7247	0.7017

All algorithms are trained on the same image and use a dictionary of size 1024, each algorithm uses the best neighborhood size

Table 4 Comparison of training time and parameter amount between deep learning methods and our method

Methods	Training Set	Parameters	Training time
SRCNN[4]	Set91	8K	> 48 h
FSRCNN[5]	General-100	13K	> 48 h
TPCNN[9]	DIV2K	52K	> 24 h
ComNet[7]	Set91,BSD200	67K	> 24 h
RFDN[19]	DIV2K	534K	> 24 h
Ours	Set91	–	404 s

mitigate the fuzzy edges produced in [36] and [39] by the neighborhood embedding reconstruction method, but it also produces a ringing effect at the edges. ANR [31] combined with neighborhood embedding and sparse coding produced favorable image SR performance, but also did not reconstruct edge information well. In contrast, our method recovers the detail information well and yields sharp edges.

| Reference | Bicubic | Yang et al. | Zeyde et al. |

| NE+LS | NE+LLE | ANR | Our |
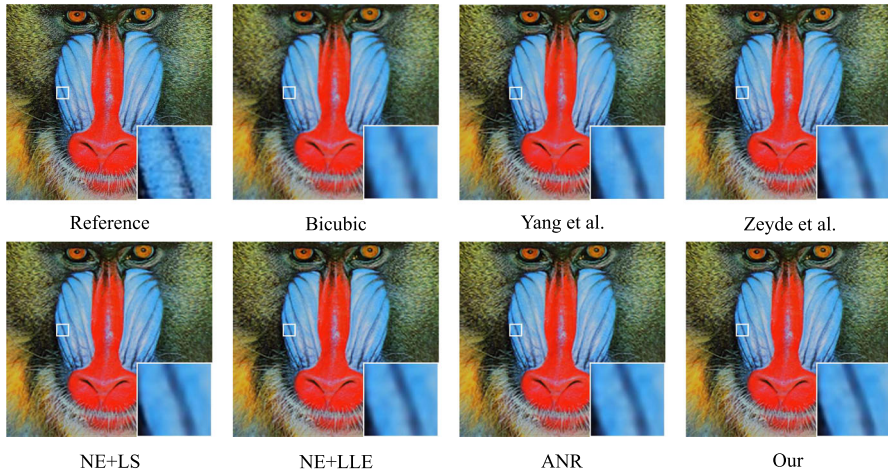
Fig. 2 The visual qualitative assessment of the 'baboon' image is from Set5 magnified ×3

4 Discussion

In the example-based image SR framework, ANR method achieves the best SR performance compared with other methods using traditional machine-learning techniques. ANR method combines the advantage of sparse coding theory and neighborhood embedding. In ANR, the sparse coding is adopted to extract the feature of image patches and neighborhood embedding to build the relationship among the features. In this paper, we further improved ANR method using Bayesian regression. From theoretic viewpoint and experimental results, one can see that the SR performance of the proposed method indeed has certain improvement.

Though it is, ANR and our proposed method are still associated with some disadvantages compared with deep learning based SR methods. For example, the feature extraction ability of ANR and our proposed method is still weak in a certain extent because only the sparse representation features are utilized. However, the deep learning based methods show powerful ability in extracting the shallow and deep features of images by utilizing a series of convolutional layers. Full utilization of the shallow features and semantic features are helpful for SR performance improvement and enhancement of robustness. This is the essential reason that the deep learning based methods can achieve the breakthrough SR performance.

On the other, the model complexity and computations of deep learning based methods are very large. Because a large number of convolutional layers are involved in deep learning based models, the computation burden in model training and test is huge, therefore, massive computing power are required for most of the deep learning based SR methods. Table 4 lists the rough time required for model training and parameter amount of several deep learning based models and our proposed method. The large computations hampers the application of deep learning based SR methods in the devices with limited computing power such as smart phone and edge-devices.
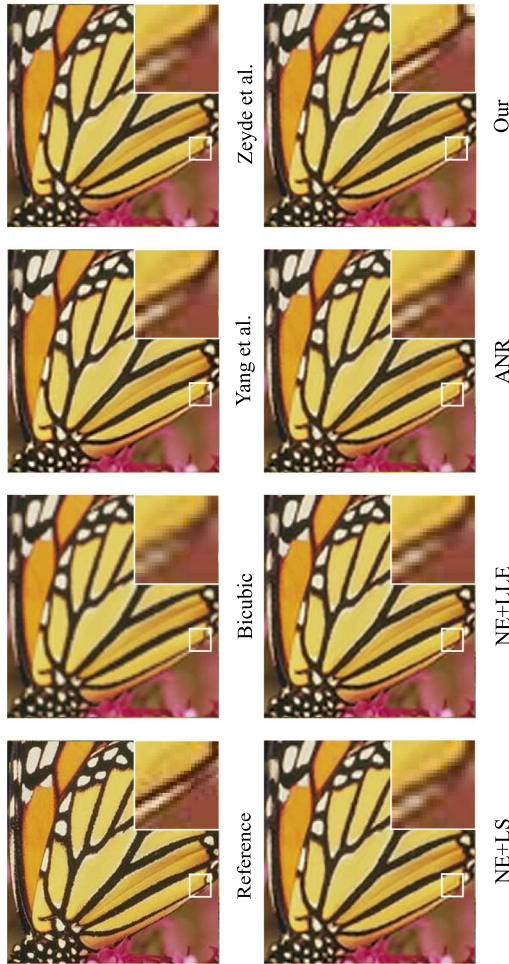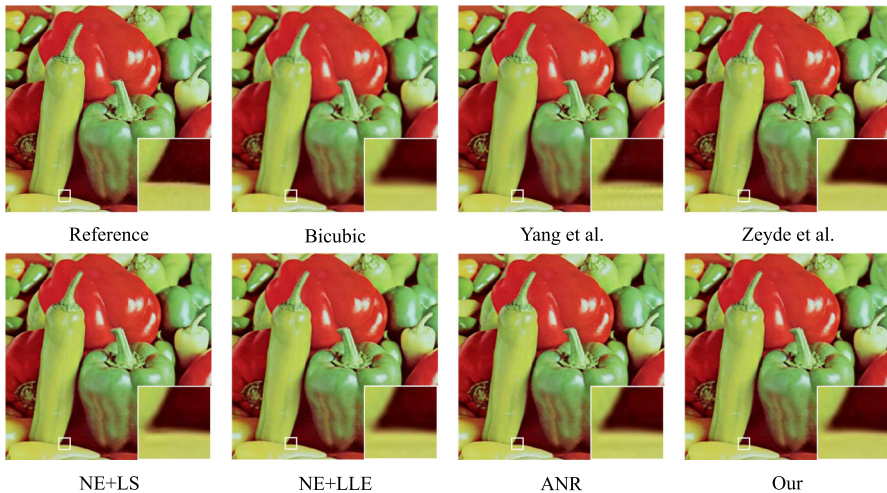
Fig. 3 The visual qualitative assessment of the 'butterfly' image is from Set5 magnified ×3
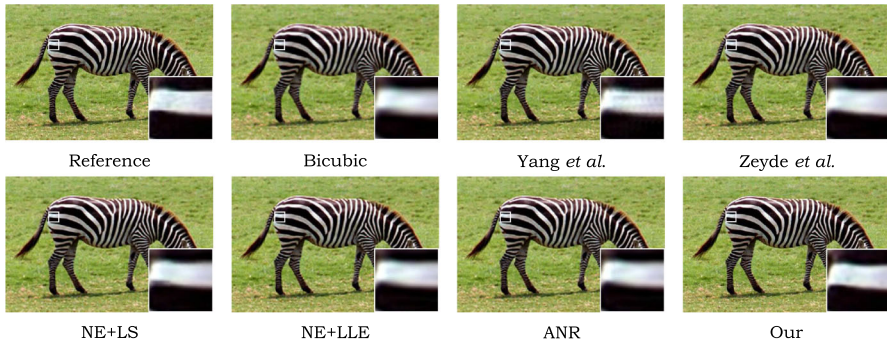
Fig. 4 The visual qualitative assessment of the 'zebra' image is from Set14 magnified ×3
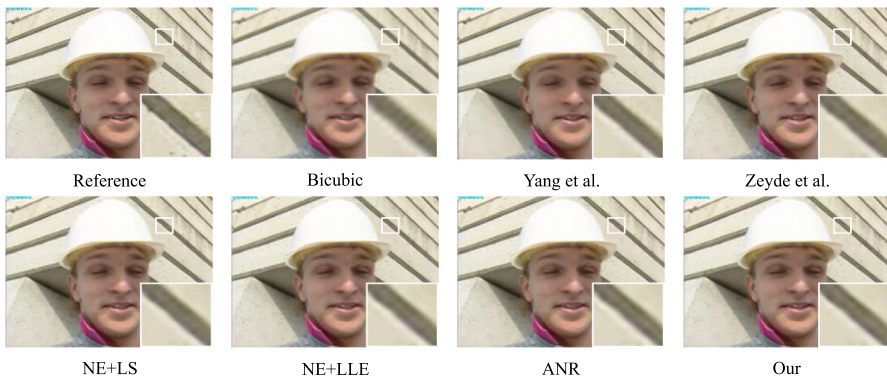
Fig. 5 The visual qualitative assessment of the 'foreman' image is from Set5 magnified ×3

Fig. 6 The visual qualitative assessment of the 'pepper' image is from Set14 magnified ×3

If we comprehensively take the SR performance and computations into account, ANR and our proposed method are competitive to deep learning based methods.

5 Conclusions

This paper presents an image SR algorithm based Bayesian anchored neighborhood regression. So as to better suppress the noise and increase the generalization performance of SR, we consider the *prior* distribution of the weight coefficients, using sparse Bayesian regression model to model the mapping relationship between LR image patch and its neighbors. Qualitative comparisons with several state-of-the-art SISR methods are performed to verify the effectiveness and stability of the proposed algorithm. Additionally, we assume that the parameters obey a Gaussian *prior* distribution, and other probability distributions can also be assumed to verify the validity of the algorithm, which we will investigate in more detail in future work.

Data Availility Both training and test datasets used in research study are openly accessible.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. H. Chang, D. Y. Yeung, Y. Xiong, Super-resolution through neighbor embedding, in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (2004), pp. 275–282
2. X. Chen, C. Qi, Nonlinear neighbor embedding for single image super-resolution via kernel mapping. Signal Process. **94**, 6–22 (2014)
3. Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. G. Christodoulou, D. Li, Brain MRI super resolution using 3D deep densely connected neural networks, in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI)* (2018), pp. 739–742. https://doi.org/10.1109/ISBI.2018.8363679
4. C. Dong, C.C. Loy, K. He, X.O. Tang, Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 295–307 (2016). https://doi.org/10.1109/TPAMI.2015.2439281
5. C. Dong, C. C. Loy, X. O. Tang, Accelerating the super-resolution convolutional neural network, in *Computer Vision–ECCV 2016: 14th European Conference* (Amsterdam, The Netherlands, 2016b), pp. 391–407
6. W.S. Dong, L. Zhang, G.M. Shi, X. Li, Nonlocally centralized sparse representation for image restoration. IEEE Trans. Image Process. **22**(4), 1620–1630 (2013). https://doi.org/10.1109/TIP.2012.2235847
7. A. Esmaeilzehi, M.O. Ahmad, M. Swamy, Compnet: a new scheme for single image super resolution based on deep convolutional neural network. IEEE Access **6**, 59963–59974 (2018). https://doi.org/10.1109/ACCESS.2018.2874442
8. A. Esmaeilzehi, M. O. Ahmad, M. Swamy, DSEGAN: a deep light-weight segmentation-based attention network for image restoration, in *2022 IEEE International Symposium on Circuits and Systems (ISCAS)* (2022), pp. 1284–1288. https://doi.org/10.1109/ISCAS48785.2022.9937894

9. A. Esmaeilzehi, M.O. Ahmad, M. Swamy, Ultralight-weight three-prior convolutional neural network for single image super resolution. IEEE Trans. Artif. Intell. **4**(6), 1724–1738 (2023). https://doi.org/10.1109/TAI.2022.3224417

10. A. Faul, M. Tipping, Analysis of sparse Bayesian learning, in *Advances in Neural Information Processing Systems* (2001), pp. 383–389

11. W. Freeman, T. Jones, E. Pasztor, Example-based super-resolution. IEEE Comput. Graph. Appl. **22**(2), 56–65 (2002). https://doi.org/10.1109/38.988747

12. X.B. Gao, K.B. Zhang, D.C. Tao, X.L. Li, Image super-resolution with sparse neighbor embedding. IEEE Trans. Image Process. **21**(7), 3194–3205 (2012). https://doi.org/10.1109/TIP.2012.2190080

13. H. He, W. C. Siu, Single image super-resolution using Gaussian process regression, in *CVPR 2011* (2011), pp. 449–456. https://doi.org/10.1109/CVPR.2011.5995713

14. L. He, H. Qi, R. Zaretzki, Non-parametric Bayesian dictionary learning for image super resolution, in *2011 Future of Instrumentation International Workshop (FIIW) Proceedings* (2011), pp. 122–125. https://doi.org/10.1109/FIIW.2011.6476831

15. H.S. Hou, H.C. Andrews, Cubic splines for image interpolation and digital filtering. IEEE Trans. Acoust. Speech Signal Process. **26**(6), 508–517 (1978)

16. Z. L. Hu, T. Li, Y. F. Yang, X. Liu, H. R. Zhang, D. Liang, Super-resolution pet image reconstruction with sparse representation, in *2017 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)* (2017), pp. 1–3. https://doi.org/10.1109/NSSMIC.2017.8532893

17. J.J. Jiang, X. Ma, C. Chen, T. Lu, Z.Y. Wang, J.Y. Ma, Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means. IEEE Trans. Multimed. **19**(1), 15–26 (2017)

18. K.I. Kim, Y. Kwon, Single-image super-resolution using sparse regression and natural image prior. IEEE Trans. Pattern Anal. Mach. Intell. **32**(6), 1127–1133 (2010). https://doi.org/10.1109/TPAMI.2010.25

19. F. Kong, M. X. Li, S. W. Liu, D. Liu, J. W. He, Y. Bai, F. G. Chen, L. Fu, Residual local feature network for efficient super-resolution, in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2022), pp. 765–775

20. Z. Lei, X. Wu, An edge-guided image interpolation algorithm via directional filtering and data fusion. IEEE Trans. Image Process. **15**(8), 2226–2238 (2006)

21. B. Li, H. Chang, S. G. Shan, X. L. Chen, Locality preserving constraints for super-resolution with neighbor embedding, in *2009 16th IEEE International Conference on Image Processing (ICIP)* (2009), pp. 1189–1192. https://doi.org/10.1109/ICIP.2009.5413691

22. J. M. Li, Y. Y. Qu, Y. Gu, T. Z. Fang, C. H. Li, Super-resolution based on fast linear kernel regression, in *2013 International Conference on Machine Learning and Cybernetics* (2013), pp. 333–339. https://doi.org/10.1109/ICMLC.2013.6890490

23. M. Li, T.Q. Nguyen, Markov random field model-based edge-directed image interpolation. IEEE Trans. Image Process. **17**(7), 1121–1128 (2008). https://doi.org/10.1109/TIP.2008.924289

24. Z. Li, W.J. Leong, M. Durand, I. Howat, K. Wadkowski, B. Yadav, J. Moortgat, Super-resolution deep neural networks for water classification from free multispectral satellite imagery. J. Hydrol. **626**, 130248 (2023)

25. Y. Ogawa, T. Hori, T. Takiguchi, Y. Ariki, Super-resolution using GMM and PLS regression, in *2012 IEEE International Symposium on Multimedia* (2012), pp. 298–301. https://doi.org/10.1109/ISM.2012.62

26. M. E. Philip, G. S. Kumar, An improved color video super-resolution using kernel regression and fuzzy enhancement, in *2012 International Conference on Advances in Computing and Communications* (2012), pp. 114–117. https://doi.org/10.1109/ICACC.2012.25

27. H.P. Song, H. Ma, Y.X. Si, J.Y. Gong, H.Y. Meng, Y.P. Lai, Back projection deep unrolling network for handwritten text image super resolution. Comput. Electr. Eng. **111**, 108965 (2023)

28. Y. Tang, Z. Jiang, J. H. Chen, Single-image super-resolution via multiple matrix-valued kernel regression, in *2019 International Conference on Machine Learning and Cybernetics (ICMLC)* (2019), pp. 1–7. https://doi.org/10.1109/ICMLC48188.2019.8949261

29. Y. Tang, T. Wang, D. Liu, MFFAGAN: generative adversarial network with multilevel feature fusion attention mechanism for remote sensing image super-resolution. IEEE J-STARS. **17**, 6860–6874 (2024)

30. C.W. Tian, R.B. Zhuge, Z.H. Wu, Y. Xu, W.M. Zuo, C. Chen, C.W. Lin, Lightweight image super-resolution with enhanced CNN. Know-Based Syst. **205**, 106235 (2020)

31. R. Timofte, V. De Smet, 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
32. M.E. Tipping, Sparse Bayesian learning and the relevance vector machine. J. Mach. Learn. Res. **1**, 211–244 (2001)
33. M. E. Tipping, A. C. Faul, Fast marginal likelihood maximisation for sparse Bayesian models, in *International Workshop on Artificial Intelligence and Statistics* (2003), pp. 276–283
34. Y.F. Wang, L.J. Wang, H.Y. Wang, P.H. Li, End-to-end image super-resolution via deep and shallow convolutional networks. IEEE Access **7**, 31959–31970 (2019). https://doi.org/10.1109/ACCESS.2019.2903582
35. L. Xin, M.T. Orchard, New edge-directed interpolation. IEEE Trans. Image Process. **10**(10), 1521–1527 (2001)
36. J.C. Yang, J. Wright, T.S. Huang, Y. Ma, Image super-resolution via sparse representation. IEEE Trans. Image Process. **19**(11), 2861–2873 (2010)
37. J. Yang, Z. Wang, Z. Lin, S. Cohen, T. Huang, Coupled dictionary training for image super-resolution. IEEE Trans. Image Process. **21**(8), 3467–3478 (2012). https://doi.org/10.1109/TIP.2012.2192127
38. W.S. Yu, S.Q. Chen, An improved neighbor embedding method to super-resolution reconstruction of a single image. Proc. Eng. **15**, 2418–2422 (2011)
39. R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in *Curves and Surfaces* (Springer, 2012), pp. 711–730
40. Y. Zhang, K. Gu, Zhang, J. Zhang, Q. Dai, Image super-resolution based on dictionary learning and anchored neighborhood regression with mutual incoherence, in *2015 IEEE International Conference on Image Processing (ICIP)* (2015), pp. 591–595. https://doi.org/10.1109/ICIP.2015.7350867
41. Y. Zhang, Y. Zhang, J. Zhang, Q. Dai, Ccr: clustering and collaborative representation for fast single image super-resolution. IEEE Trans. Multimed. **18**(3), 405–417 (2016). https://doi.org/10.1109/TMM.2015.2512046
42. Y. Zhang, Y. Zhang, J. Zhang, X. Dong, F. Yun, Y. Wang, X. Ji, Q. Dai, Collaborative representation cascade for single-image super-resolution. IEEE Trans. Syst. Man Cybern. **49**(5), 845–860 (2019). https://doi.org/10.1109/TSMC.2017.2705480