Maximizing Nonlocal Self-Similarity Prior for Single Image Super-Resolution

Jianhong Li,1 Kanoksak Wattanachote,2 and Yarong Wu3

1Laboratory of Language Engineering and Computing, Guangdong University of Foreign Studies, Guangzhou 510006, China
2School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou 510006, China
3Zhongkai Science & Technology Development Company, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China

Correspondence should be addressed to Yarong Wu; wyrljh@163.com

Received 22 November 2018; Accepted 7 February 2019; Published 28 February 2019

Prior knowledge plays an important role in the process of image super-resolution reconstruction, which can constrain the solution space efficiently. In this paper, we utilized the fact that clear image exhibits stronger self-similarity property than other degraded version to present a new prior called maximizing nonlocal self-similarity for single image super-resolution. For describing the prior with mathematical language, a joint Gaussian mixture model was trained with LR and HR patch pairs extracted from the input LR image and its lower scale, and the prior can be described as a specific Gaussian distribution by derivation. In our algorithm, a large scale of sophisticated training and time-consuming nearest neighbor searching is not necessary, and the cost function of this algorithm shows closed form solution. The experiments conducted on BSD500 and other popular images demonstrate that the proposed method outperforms traditional methods and is competitive with the current state-of-the-art algorithms in terms of both quantitative metrics and visual quality.

1. Introduction

The technology of single image super-resolution (SISR) has been widely used in many fields, such as medical imaging, remote sensing and digital home etc., which refers to estimating the corresponding high-resolution (HR) image just according to an input low resolution (LR) one through software computing way. Due to the one-to-many mapping relationship between the LR and HR images captured from the same scene, the process from LR image to HR one is obviously a typically ill-posed problem. For obtaining HR image with sharp edges and fine details, researchers generally introduce some priors explicitly or implicitly to impose on the inverse imaging process; thus the quality of the reconstructed HR image is closely related to the introduced priors. The common priors for SISR include edge prior [1], gradient profile prior [2], sparse prior [3, 4], etc.

Except a handful of SISR algorithms designed for particular types of images [5–7], natural images are the main target for most SISR algorithms. Generally speaking, there are mainly three categories of SISR approaches including interpolation-based methods, reconstruction-based methods, and learning-based methods.

Interpolation-based methods assumed that the structures of one natural image are piece-wise smoothed [8]. According to this assumption, these methods take a kernel interpolator or base function to approximate the pixel values in HR grid and then transform the discrete digital image into continuous surface for estimating the unknown pixel values at specific position. This category is simple, efficient, even real-time processing because they do not discriminate edge or texture from smoothed region. The drawback of this category is that they cannot restore the high frequency information which was lost during imaging procedure and will lead to visible blur and noise along the edge in the reconstructed results to degrade its perception.

The reconstruction-based methods usually incorporate priors explicitly [2, 9, 10], which require not only making the reconstructed HR image to satisfy the prior but also promoting the image degenerated from the reconstructed HR
image by imaging model as similar as possible to the input one. The priors used in reconstruction-based methods are obtaining from edge statistical information of the desired HR images [2, 10]; hence this category is good at reconstructing edges and can suppress noises and artifacts near edges effectively. However, when the super-resolved magnification factor is larger, the prior term provides less and less effective information, which leads to be constrained for the ability to restore high frequency information, and exhibits displeasing artifacts in the reconstructed texture and detailed regions.

Learning-based methods take advantage of an external HR image dataset to induce a mapping function from LR to HR image, which contains some implicit priors. For estimating the HR image, we need to substitute the LR image into the mapping function as a variable. Freeman et al. [11] designed a Markov Network model with HR and LR image patches, for each LR patch extracted from the input LR image, which selected the ‘best proper’ HR patch from some candidates as the reconstructed result. In essence, the whole reconstructed HR image was stitched by these ‘best proper’ HR patches from the training set, which makes noise and blur generated easily near the stitched position. Because of the nearest neighbors searching step, time-consuming is another drawback for this algorithm. Chang et al. [12] introduced an idea called ‘Local Linear Embedding’ (LLE) originated from manifold learning. For each LR image patch, its k nearest neighbors found in training set were used for fitting it. The fitting coefficients between the LR image patch and the corresponding HR one were assumed to be identical. Thus the unknown HR patches can be estimated by weighted combination of the HR versions sourced from the k nearest neighbors. This algorithm can successfully suppress the noise appeared near the stitched position and improve the reconstructed result to some extent. However, the algorithm needs to define the number of nearest neighbors k in advance, which make it cannot be adapted according to the image content. Therefore, some noise and blur can be observed along the salient edge in the reconstructed result. Yang et al. [3, 4] considered SISR as sparse representation problem and constructed two over-completed dictionaries for LR and HR image jointly. For each patch extracted from the input LR image, the algorithm selected as little as possible ‘atoms’ in the LR dictionary to sparsely represent it and transferred the sparse coefficients to the HR dictionary to estimate the corresponding HR patch. Actually, this algorithm solved the problem of ‘fixed k’ in [12] and significantly improved the quality of the reconstructed results. However, the condition for obtaining the accurate sparse coefficients is greatly rigorous, which is relaxed to choose ‘atoms’ by an approximately random style, and this will make the discriminative atoms leak into the fitting step to generate more errors unavoidably. In order to choose some specific atoms when sparse represent a patch, Dong et al. [13] proposed an adaptive sparse domain selection algorithm, which preclassify the training set and construct a subdictionary for each class. For each super-resolved patch, it would select one subdictionary according to the distant between the patch and these subdictionaries to sparsely represent it, which can alleviate the problem of existing noise along salient edges to some extent. Timofte et al. [14] combined neighbor embedding and sparse representation to find the nearest atom in the over-completed dictionary and take advantage of the k nearest neighbors of this atom in the dictionary to build a matrix operator. The matrix operator can speedily map the LR patch into HR space. Based on the above theory, Timofte et al. [15] additionally explored the original training images to further improve the algorithm's efficiency significantly. With the development of computer technology and the increasing enhancement of its ability to deal with data, deep learning as a resurgent technology is very popular in SISR [16–22]. However, Graphic Processing Unit (GPU) is a necessary accelerating installation for deep learning algorithm to optimize the weighted coefficients among neural network nodes by training a large number of HR images. Another weakness of deep learning technique is that prior knowledge is not effortless to be incorporated during the training process. For further improvement, the quality of the reconstructed HR image is restricted. As a summary, learning-based SISR algorithm exhibits huge potential for preserving edges and restoring details. However, this category relies too heavily on external dataset, which is a huge challenge for normal computer with the increasing complexity of SISR algorithm. The necessary number of HR images also shows an exponential rising tendency. Meanwhile, little research has been done on HR training images themselves, just to stack examples to enlarge the size of training set. For a specific super-resolved image, it is hardly to explain what roles an image from training set will play up to now. The only subjective conclusion is that an HR training image will supply more high frequency information when its content is more similar to the input one.

As far as image content is concerned, the input LR image itself is one of the closest ones to the corresponding HR image. Therefore, a kind of special learning-based algorithm called self-similarity based SISR has been presented [23–27]. The self-similarity of an image refers to a phenomenon when we observe an image from small patches perspective; each small patch will recur at other position or other scales with higher probability [23–29]. The self-similarity based SISR algorithms with the help of self-similarity property take the input image itself as training example to estimating the HR image. Glasner et al. [23] proposed a unified framework which considers the similar patches from the same scale as the different views from the same scenario and build the training patch pairs with similar patches from different scales and then integrated multiframe super-resolution technology with learning-based approach to magnify the LR input image. For some specific patches, there could exist obvious differences between the input patch and its nearest neighbors, which will generate blur and noise near the edges inevitably or introduce incorrect high frequency components in detailed regions. Concerning this issue, Freeman et al. [24] and Yang et al. [25] pointed out, respectively, that almost all patches will recur themselves in a small window at the same position in the input or lower scale image. This property can make the consuming time for finding nearest neighbors reduced significantly. However, due to the limited size of the training set, the edges of reconstructed HR image look too sharp to be natural. For solving the problem of small training
size, He et al. [30] constructed Gaussian process regression (GPR) training set to train parameters in a small window. However, this algorithm did not explore the relationship among different windows that leads to obvious noise and artifacts existed in the result. For the same purpose, Huang et al. [31] performed specific geometric deformation on the input image patch which not only extend the research space but also make the nearest neighbors matching more accurate. Due to the geometric deformation leading to distortion of the patch, there are visible noise and blur in the reconstructed HR image especially in complex texture region.

Self-similarity based SISR algorithm generally constructs the training set just with the help of self-similarity property of LR image. However, the self-similarity property of the unknown HR image has not been noticed. In this paper, we consider the self-similarity of the unknown HR image as a prior and propose a new SISR algorithm. The main contributions are as follows.

First, we utilize different scales of the input LR image to construct a joint Gaussian mixture model. Then an interesting prior is discovered by deviation with the help of this model. Combined with this discovery, we deliver a novel SISR algorithm, which is competitive to the state-of-the-art algorithms in both the quality of the reconstructed image and time cost.

Second, our proposed algorithm needs two steps: training and reconstructing. In training step, we train a joint Gaussian mixture model that contains a small number of Gaussian components. In reconstructing step, our algorithm can avoid nearest neighbor searching which is time-consuming and liable to generate errors.

At last, the presented prior in our algorithm can be self-adapted according to the input LR image, which make our algorithm more robust and can be uncomplicated expand into other fields such as image denoising, image restoration and so on.

The rest of this paper is organized as follows. Section 2 explains the prior term in SISR cost function. Section 3 details our proposed algorithm. The corresponding experiments and analysis are all exhibited in Section 4. Section 5 devotes to the conclusion of our work in this paper.

2. Prior in Cost Function

Imaging process will be disturbed by some degradation factors such as relative motion, focus on misalignment, atmosphere turbulence, etc. which is difficult to find an ideal mathematical model to explain. Actually, researchers generally describe the imaging procedure with a linear system model for SISR problem

\[ Y = AX + n, \]

where \( Y \) is the input LR observation, \( X \) is the unknown ideal HR image, \( A \) is the degraded matrix which should include filtering and down-sampling operations, and \( n \) is additive white Gaussian noise generated in imaging process. For an observed LR image, there are infinite number of solutions that can gratify the imaging model, consequently the problem of SISR is a representative ill-posed problem. For obtaining a satisfying solution, researchers often incorporate some priors to regularize the solution space. In terms of prior modality, there are primarily two ways to introduce them, explicitly, and implicitly, as shown in Figure 1. The explicit priors generated by summarizing external HR images are intuitive and comprehensible for their concrete expression formula. The SISR cost function combined with some explicit prior can be casted as follows:

\[ \arg \min_{X} \| Y - AX \|^2 + \lambda \rho(X), \]

where \( \| Y - AX \|^2 \) is the fidelity term, \( \rho(X) \) is the prior, and \( \lambda \) is a balance coefficient to tradeoff between the two terms. Natural images, as a whole, vary in content and structure, which are difficult to establish with a unified model to represent. However, when researching them from local region perspective (such as image patch with size \( 5 \times 5 \), \( 7 \times 7 \ldots \)), the content of each image patch becomes primitive.
and shows strong regularity, which makes modeling painless. Consequently, the prior term designed from the view of image patches can be written as

\[ \rho(X) = -\log \prod_{i=1}^{N} p(P_i, X) = -\sum_{i=1}^{N} \log \rho(P_i, X), \]  

(3)

where \( N \) is the total number of those extracted patches, \( P_i \) is an extracted matrix which takes charge of extracting \( i \)-th patch from HR image \( X \), and \( p(x_i) \) is the prior of \( i \)-th patch, where concrete expression of it will vary with the selected prior \([28, 29]\). On the whole, the formulated problem in formula (2) is nonconvex and difficult to solvedirectly. More typically, an auxiliary variable \( z_i \) is introduced foreach \( x_i \), and then formula (3) can be solved by ‘Half Quadratic Splitting’ algorithm \([32–37]\).

3. Maximizing Nonlocal Self-Similarity Prior for SISR

3.1. Problem Formulation. Self-similarity is an inherent property of natural image itself \([23–25, 28, 29, 33]\). Through experiments displayed in Figure 2, we can notice that what the clearer, sharper, and more detailed image are, what the more obvious this property is. The image patch extracted arbitrarily from the clear image has an extremely similar appearance to the nearest neighbor found in its lower scale (as shown in Figure 2(a)). However, in the image with noise (as shown in Figure 2(b)) or blur (as shown in Figure 2(c)) enforced, there are big differences between the selected patch and its nearest neighbors found in the corresponding lower scales. Based on this phenomenon, we believe that the clearer the image is, the stronger the self-similarity property manifests. In other words, on the premise that the reconstructed HR result is good enough, one image patch arbitrarily extracted from it will find similar patches at its lower scale with higher probability (in this paper, the lower scale refers to the input LR image). And the following is the concrete realization of this idea.

Supported by self-similarity, we use the input LR image to construct the training and testing dataset. As shown in Figure 3, first, upsample both the input LR image \( Y \) and its downsampling version \( Z \), \( s \) times to obtain the corresponding results \( X' \) and \( Y' \), and \( X \) is the unknown HR image for estimating. As a matter of fact, \( Y \) and \( Y' \) can be considered as the degraded LR image version of \( X \) and \( X' \) separately. It can be noticed according to the self-similarity property, for patches \( x_i \) and \( x'_i \) separately extracted from \( X \) and \( X' \) that there should exist recurrences in the corresponding LR images \( Y \) and \( Y' \) with the highest probability. Second, we can formulate this problem as follows: extracting all patches form \( Y \) and \( Y' \), then combining them to obtain the training set \( \{y_j^T, y'_j^T\}_{j=1}^{M} \), where \( y_j \) and \( y'_j \) are \( j \)-th patches extracted from \( Y \) and \( Y' \) separately. \( M \) is the total number of patches extracted from \( Y \) or \( Y' \), and \( T \) represents the transpose operator on a vector. Next, we combine the patches \( x_i \) and \( x'_i \) which are the \( i \)-th patches extracted form \( X \) and \( X' \) separately. According to the self-similarity theory, when \( [x_i^T, x'_i^T]^T \) make recurrence in training set \( \{y_j^T, y'_j^T\}_{j=1}^{M} \) with the highest probability, the residual core task is how to estimate the HR patch \( x_i \).
3.2. Nonlocal Self-Similarity Prior. Recently, GMM has been a popular and valid tool used in image processing [38] due to the fact that it can smoothly approximate the density distribution of arbitrary shape and is characterized as remarkable efficiency. Specifically, this property shows excellent performance in image modeling based on image patches [39, 40]. In this paper, for training set \( \{ y_j^T, y_j^T \}_{j=1}^M \) \( \in \mathbb{R}^2 \) and \( y_j^T \in \mathbb{R}^2 \), where \( a \) is an integer that represents the size of extracted patch \( a \times a \), we introduce GMM to approximate its probability density function, as shown in formula (4),

\[
p( y_j^T, y_j^T | \mu_k, \Sigma_k ) = \sum_{k=1}^{K} \pi_k N \left( \begin{bmatrix} y_j^T \\ y_j^T \end{bmatrix} | \mu_k, \Sigma_k \right),
\]

where \( K \) is the number of Gaussian components in GMM, \( \{ \pi_k, \mu_k, \Sigma_k \} \) is the weighted coefficient, mean vector and covariance matrix of \( k \)-th Gaussian component separately, according to the definition of GMM, \( \sum_{k=1}^{K} \pi_k = 1 \), and \( N(z | \mu_k, \Sigma_k) \) represents \( k \)-th Gaussian component, and its concrete expression is

\[
N(z | \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (z - \mu)^T \Sigma^{-1} (z - \mu) \right\}.
\]

We take the expectation maximization (EM) algorithm to training the parameters \( \{ \pi_k, \mu_k, \Sigma_k \} \), as illustrated in Algorithm 1. The EM algorithm can be divided into two steps including E-step and M-step, which are executed alternately. E-step refers to estimating the probability that the \( j \)-th data point \( z_j \) in the training dataset is generated by the \( k \)-th Gaussian component if the model parameters are already known, which can be described as

\[
\gamma(j, k) = \frac{\pi_k N \left( z_j | \mu_k, \Sigma_k \right)}{\sum_{i=1}^{K} \pi_i N \left( z_j | \mu_i, \Sigma_i \right)}.
\]

Thus, the probability of the whole dataset generated by \( k \)-th Gaussian component can be casted as

\[
N_k = \sum_{j=1}^{N} \gamma(j, k).
\]

Next, the model parameters \( \{ \pi_k, \mu_k, \Sigma_k \} \) of GMM can be estimated by maximum likelihood in M-step.

\[
\pi_k = \frac{N_k}{N},
\]

\[
\mu_k = \frac{1}{N_k} \sum_{j=1}^{N} \gamma(j, k) z_j,
\]

\[
\Sigma_k = \frac{1}{N_k} \sum_{j=1}^{N} \gamma(j, k) (z_j - \mu_k) (z_j - \mu_k)^T.
\]

Because each example in training set was combined by two patches extracted from \( Y \) and \( Y' \) separately, the mean vector \( \mu_k \) and covariance matrix \( \Sigma_k \) can be rewritten as

\[
\mu_k = \begin{bmatrix} \mu_y^T \\ \mu_y'^T \end{bmatrix},
\]

\[
\Sigma_k = \begin{bmatrix} \Sigma_y & \Sigma_y' \\ \Sigma_y'^T & \Sigma_{y'} \end{bmatrix}.
\]

Thus the probability of testing example \( [x_i^T, x_i'^T] \) in the probability density function of the training set can be represented as

\[
p \left( x_i, x_i' | \mu_k, \Sigma_k \right) = \sum_{k=1}^{K} \pi_k N \left( x_i, x_i' | \mu_k, \Sigma_k \right).
\]

Thus, we discovered that the probability of patch vector \( x_i \) satisfied a specific Gaussian distribution

\[
p \left( x_i | x_i', \{ \pi_k, \mu_k, \Sigma_k \}_{k=1}^{K} \right) \sim N \left( x_i | \mu_x, \Sigma_x \right),
\]
where \( \mu_k \) and \( \Sigma_k \) can be denoted respectively as mean vector and covariance matrix for specifying this Gaussian distribution.

\[
\mu_k = \frac{1}{N} \sum_{i=1}^{K} \varphi_k(x'_i) \left[ \mu^y_k + \Sigma_k^y \left( \Sigma_k^y \right)^{-1} (x'_i - \mu^y_k) \right], \quad (15)
\]

\[
\Sigma_k = \frac{1}{N} \sum_{i=1}^{K} \varphi_k(x'_i) \left[ \Sigma_k^x - \Sigma_k^y \left( \Sigma_k^y \right)^{-1} \Sigma_k^y \right], \quad (16)
\]

And the coefficient \( \varphi_k(x'_i) \) appeared in formula (15) and formula (16) is

\[
\varphi_k(x'_i) = \frac{\pi_k \text{N} \left( x'_i \mid \mu^y_k, \Sigma_k^y \right)}{\sum_{k=1}^{K} \pi_k \text{N} \left( x'_i \mid \mu^y_k, \Sigma_k^y \right)}. \quad (17)
\]

Then the prior in this paper can be define as

\[
\rho(X) = -\sum_{i=1}^{N} \log p \left( P_{ij}X \right) = -\sum_{i=1}^{N} \log N \left( P_{ij}X \mid \mu_{x_i}, \Sigma_{x_i} \right), \quad (18)
\]

where \( N \) is the number of patches extracted from the unknown HR image \( X \) and \( P_{ij} \) is an extracted matrix to extract \( i \)-\( th \) patch from \( X \). According to the prior expression, we can conclude that the recurrence probability of each patch in unknown HR image is measured, and we call this prior as nonlocal self-similarity prior.

### 3.3. Image Super-Resolution Reconstruction

In this paper, our algorithm requires the reconstructed HR result to be as similar as possible to the input LR image after the degeneration model processed; meanwhile the result also should satisfy the proposed nonlocal self-similarity prior. Therefore, the cost function of our algorithm is designed as

\[
\arg\min_X \frac{1}{2} \| Y - AX \|^2 - \lambda \sum_{i=1}^{N} \log N \left( P_{ij}X \mid \mu_{x_i}, \Sigma_{x_i} \right), \quad (19)
\]

where the first is fidelity term, the second is prior term, and \( \lambda \) is balance coefficient between the two terms. The cost function can be further simplified as

\[
\arg\min_X \frac{1}{2} \| Y - AX \|^2 + \lambda \sum_{i=1}^{N} \log \left| \Sigma_{x_i} \right| + \frac{1}{2} (X - \mu_{x_i})^T \Sigma_{x_i}^{-1} (X - \mu_{x_i}). \quad (20)
\]

The solution of (20) is relatively simple to solve; we directly take the derivative of (20) and set the derivative to zero

\[
A^TAX - A^TY + \lambda \sum_{i=1}^{N} \sum_{j=1}^{N} P_{ij} \Sigma_{x_j}^{-1} (X - \mu_{x_i}) = 0. \quad (21)
\]

The final estimation of the HR image can be obtained by further sorting the expression

\[
X = \left( A^TA + \lambda \sum_{i=1}^{N} P_{ij} \Sigma_{x_j}^{-1} \right)^{-1} \cdot \left( A^TY + \lambda \sum_{i=1}^{N} P_{ij} \Sigma_{x_j}^{-1} \mu_{x_i} \right). \quad (22)
\]

The above idea can be implemented by Algorithm 2 to estimate the whole HR image, and the complete framework of the proposed method is illustrated in Figure 4 to exhibit how to super-resolve a color image.

### 4. Experiments and Analyses

In this section, we verify the efficiency of our nonlocal self-similarity prior super-resolution algorithm by subjective visual observation and objective parameters comparisons. We firstly elaborate the parameters setting which is necessary in implementing our algorithm and then give the comparison results including the quality of the reconstructed HR images and their consuming time. At last, the performance of our algorithm is further discussed.

#### 4.1. Experiment Setting

We regard BSD500 dataset and some commonly used test images in image processing field as the ideal HR images and all these HR images were downloaded from the website https://www2.eecs.berkeley
Input: small image $Y$, magnification factor $s$

Output: HR image $X$

1. $X' = \text{upsampling}(Y, s)$;
2. $Y' = \text{upsampling}(\text{downsampling}(Y, s), s)$;
3. Partitioning $Y, Y'$ into overlapping patches $\{y_j^1\}_{j=1}^n$ and $\{y_j^2\}_{j=1}^n$;
4. $[\pi_k, \mu_k, \Sigma_k] = \text{Training}(Y, Y')$;
5. For each patch $x_i'$ extracted from $X'$;
6. For $k = 1 : K$
7. Computing $\varphi_k(x_i')$ using (17);
8. End For
9. Computing $\mu_k$ using (15);
10. Computing $\Sigma_k$ using (16);
11. End For
12. Computing $X$ using (22);
13. return $X$

Algorithm 2: SISR using nonlocal self-similarity prior.

Figure 4: Framework of our proposed algorithm.

The input LR image is transformed to the YUV color space because channel $Y$ representing brightness is sensitive to human eyes. The channel $Y$ is processed by our algorithm. The $U$ and $V$ channels are relatively less sensitive to human eyes. And the $U$ and $V$ channels can be magnified to the target size directly by bicubic method. Finally, the three reconstructed channel images are merged and transformed from YUV space to RGB space for displaying. In addition, the parameters involved in

.edu/Research/Projects/CS/vision/grouping/resources.html and http://people.csail.mit.edu/mrub/retargetme/download.html. In order to simulate the image degradation model to generate LR images for testing input, these HR images were downsampled $s$ times by bicubic method and then filtered by Gaussian filtering with mean value of 0 and variance of 0.5. Finally, we input these LR images into our proposed algorithm to upsample them $s$ times. As the self-similarity between two different scales of one image will be weaken with the decreasing scale [20, 25, 26, 30, 35], when the magnification factor is larger, zooming the input LR image to the target size directly will not achieve the best performance for the reconstructed HR image. Aiming at this problem, we adopted a stepwise super-resolving method, and the magnification factor is 1.25 for each time.

The input color image, we firstly transformed this image from RGB space to YUV space, because channel $Y$ representing brightness is sensitive to human eyes. The channel $Y$ is processed by our algorithm. The $U$ and $V$ channels representing color information are relatively sluggish to human eyes. And the $U$ and $V$ channels can be magnified to the target size directly by bicubic method. Finally, the three reconstructed channel images are merged and transformed from YUV space to RGB space for displaying. In addition, the parameters involved in
that, for almost all the Patches and will be explained latter. In the experiment, we noticed {\( \varphi \)} amount of calculation, formula (23) was used and shown in Figure 5. As a consequence, for reducing the amount of calculation, formula (23) was used and \( \varphi_k(x'_i) \) was set to

\[
\varphi_k(x'_i) = \begin{cases} 
\varphi_k(x'_i), & \varphi_k(x'_i) \geq K^{-1} \\
0, & \varphi_k(x'_i) < K^{-1}.
\end{cases}
\] (23)

4.2. Experiment Results. In order to verify the effectiveness of our proposed algorithm, we chose some state-of-the-art algorithms and conducted a series of comparison experiments. The selected algorithms for comparison include ScSR [4], Glasner [23], GPR [30], SPM [41], SelfExSR [31], A+ [15], SRCNN [17], and FSRCNN [18], where Glasner, GPR, SelfExSR, and our proposed are self-similarity based super-resolution algorithms. The codes of all these algorithms except Glasner were downloaded from the authors own websites separately. The code of Glasner was achieved by ourselves, and the results generated by this code are very similar to those image displayed in the original paper [23]. All these codes were designed by Matlab software. All these algorithms were executed in inter (R) core (TM) i7-5600 CPU @ 2.60 GHz, 8.0 GM cache and windows 7 professional, 64 bit operating system with Matlab 2015b. As shown in Figures 6, 7, and 8, these three groups of images are the comparison results of these algorithms that are magnified by 2 times, 3 times, and 4 times separately. Observing these three groups of images carefully, we can notice that the algorithms of ScSR, Glasner, GPR, and SPM did not preserve the edges well and made noise, artifacts, and distortion along the salient edges in the reconstructed HR images. Meanwhile, the generated noise and artifacts become more obvious with the increasing of the image magnification factor in these methods. On the contrary, the algorithms of selfExSR, A+, SRCNN, FSRCNN, and our proposed have a good effect on the edge and texture region in the images, and with the increasing of magnification factor, there is no significant noise and artifacts generated near the edges or in the texture region. Furthermore, by observing the smallest windows marked in the reconstructed results, it can be found that our method has the smallest deformation near the edge, which is closest to the original image and shows excellent texture processing capability. In order to further illustrate the performance of this algorithm, we randomly selected 10 images from ideal HR image dataset, as shown in Figure 9, which were super-resolved by these selected state-of-the-art algorithms and our proposed with 3 and 4 times separately. Then, the parameters of peak single to noise ratio (PSNR) and structure similarity index (SSIM) of each reconstructed image were calculated. PSNR and SSIM are two parameters to measure the image quality under laboratory conditions. Generally speaking, the higher PSNR and SSIM values represent the better quality of the reconstructed results. These PSNR and SSIM values of the reconstructed images are shown in Tables 1 and 2, where for each reconstructed image, the first line is PSNR and the second line is SSIM. By observing the two tables, it can be concluded that the quality of our proposed algorithm is superior to others in a majority of results.

In practical, the number of Gaussian components \( K \) in GMM has to be set manually in our algorithm. In order to reveal the relationship between Gaussian component and reconstructed results, we magnified the 10 testing images with 3 times when the number of Gaussian components is set to be 3, 4, ..., 11, 12, respectively, and then computed the average PSNR and SSIM values, and the results are shown in Figure 10. As can be seen from Figure 10, when \( K = 8 \), both the PSNR and SSIM values achieved their maximum separately, which mean the image quality is best at this point. How to set the parameter \( \lambda \) is a key factor affecting the quality of the reconstructed high-resolution image which balances between the fidelity term and the prior term. We randomly selected 10 high-resolution images from the training set to be downsampled 3 times as the inputs, then set the parameter \( \lambda \) as 0.1, 0.2, ..., 1.0, and keep other parameters to be fixed.
### Table 1: Quality evaluation results of 10 test images with super resolution 3x.

<table>
<thead>
<tr>
<th>Image</th>
<th>ScSR</th>
<th>Glasner</th>
<th>GPR</th>
<th>SPM</th>
<th>SelfExSR</th>
<th>A+</th>
<th>SRCNN</th>
<th>FSRCNN</th>
<th>Our</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fence</td>
<td>28.31</td>
<td>28.12</td>
<td>27.72</td>
<td>27.77</td>
<td>29.06</td>
<td>28.69</td>
<td>28.99</td>
<td>29.17</td>
<td>29.41</td>
</tr>
<tr>
<td>Building</td>
<td>28.79</td>
<td>27.72</td>
<td>27.57</td>
<td>28.55</td>
<td><strong>30.83</strong></td>
<td>29.49</td>
<td>29.39</td>
<td>30.36</td>
<td>30.47</td>
</tr>
<tr>
<td>Baby</td>
<td>36.99</td>
<td>32.23</td>
<td>32.24</td>
<td>36.55</td>
<td>37.20</td>
<td>37.14</td>
<td>37.14</td>
<td>37.20</td>
<td>37.55</td>
</tr>
<tr>
<td>Lena</td>
<td>34.88</td>
<td>34.37</td>
<td>33.69</td>
<td>35.16</td>
<td>35.05</td>
<td>35.14</td>
<td>35.14</td>
<td>35.48</td>
<td>35.21</td>
</tr>
<tr>
<td>Tiger</td>
<td>28.94</td>
<td>28.22</td>
<td>27.43</td>
<td>28.39</td>
<td>29.00</td>
<td>29.12</td>
<td>29.12</td>
<td>29.28</td>
<td>29.48</td>
</tr>
<tr>
<td>Kola</td>
<td>34.01</td>
<td>33.93</td>
<td>32.11</td>
<td>33.98</td>
<td>34.07</td>
<td>34.17</td>
<td>34.27</td>
<td>34.43</td>
<td>34.43</td>
</tr>
<tr>
<td>Butterfly</td>
<td>31.47</td>
<td>31.92</td>
<td>30.70</td>
<td>31.83</td>
<td>32.05</td>
<td>32.28</td>
<td>32.28</td>
<td>32.63</td>
<td>32.58</td>
</tr>
<tr>
<td>Parrot</td>
<td>31.69</td>
<td>30.28</td>
<td>32.20</td>
<td>33.02</td>
<td>33.39</td>
<td>34.02</td>
<td>34.02</td>
<td>34.24</td>
<td>34.24</td>
</tr>
<tr>
<td>Zebra</td>
<td>30.76</td>
<td>29.77</td>
<td>30.71</td>
<td>31.01</td>
<td><strong>31.74</strong></td>
<td>31.17</td>
<td>31.17</td>
<td>31.33</td>
<td>31.33</td>
</tr>
<tr>
<td>Average</td>
<td>28.93</td>
<td>28.22</td>
<td>27.43</td>
<td>28.39</td>
<td>29.00</td>
<td>29.12</td>
<td>29.12</td>
<td>29.28</td>
<td>29.48</td>
</tr>
</tbody>
</table>

### Table 2: Quality evaluation results of 10 test images with super resolution 4.

<table>
<thead>
<tr>
<th>Image</th>
<th>ScSR</th>
<th>Glasner</th>
<th>GPR</th>
<th>SPM</th>
<th>SelfExSR</th>
<th>A+</th>
<th>SRCNN</th>
<th>FSRCNN</th>
<th>Our</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fence</td>
<td>26.83</td>
<td>26.00</td>
<td>26.84</td>
<td>26.98</td>
<td>27.65</td>
<td>26.94</td>
<td>27.59</td>
<td>27.78</td>
<td>27.86</td>
</tr>
<tr>
<td>Building</td>
<td>26.38</td>
<td>26.53</td>
<td>26.06</td>
<td>26.13</td>
<td>27.03</td>
<td>27.27</td>
<td>27.24</td>
<td>27.84</td>
<td>27.88</td>
</tr>
<tr>
<td>Baby</td>
<td>33.20</td>
<td>33.86</td>
<td>33.04</td>
<td>33.62</td>
<td>34.81</td>
<td>34.73</td>
<td>34.82</td>
<td>34.89</td>
<td>34.97</td>
</tr>
<tr>
<td>Barbara</td>
<td>25.82</td>
<td>23.59</td>
<td>23.78</td>
<td>23.83</td>
<td>24.04</td>
<td>24.03</td>
<td>24.05</td>
<td>24.51</td>
<td>25.84</td>
</tr>
<tr>
<td>Lena</td>
<td>30.06</td>
<td>29.94</td>
<td>29.00</td>
<td>29.85</td>
<td><strong>31.90</strong></td>
<td>31.76</td>
<td>31.87</td>
<td>31.60</td>
<td>31.79</td>
</tr>
<tr>
<td>Butterfly</td>
<td>25.42</td>
<td>25.77</td>
<td>25.50</td>
<td>24.66</td>
<td>28.77</td>
<td>29.40</td>
<td>29.53</td>
<td><strong>29.66</strong></td>
<td>29.55</td>
</tr>
<tr>
<td>Parrot</td>
<td>27.85</td>
<td>27.64</td>
<td>27.37</td>
<td>27.84</td>
<td>29.18</td>
<td>29.18</td>
<td>29.28</td>
<td>29.52</td>
<td>29.59</td>
</tr>
<tr>
<td>Average</td>
<td>27.45</td>
<td>27.10</td>
<td>27.13</td>
<td>27.44</td>
<td>28.77</td>
<td>28.70</td>
<td>28.87</td>
<td>28.25</td>
<td>28.84</td>
</tr>
</tbody>
</table>
Figure 6: Super-resolved image (2×) of 'temple'.

Figure 7: Super-resolved image (3×) of 'penguin'.
to super-resolve them separately and finally compute the average PSNR and SSIM for each $\lambda$ about the 10 images. The result was illustrated below in Figure 11. We can discover from Figure 11 that the parameters both PSNR and SSIM were maximized at $\lambda = 0.5$. For verifying the effectiveness of our algorithm from statistical perspective, we randomly selected 200 images from BSD500 as testing images firstly which were super-resolved by these state-of-the-art algorithms and ours with 3 times. Then, calculate and restore these PSNR values for each reconstructed HR image and then count the HR results which entered into the top three maximums for each algorithm. The process of SSIM parameter is the same as PSNR. The statistical results were shown in Figure 12 where these algorithms were listed horizontally, and the number of best quality images for each algorithm was displayed vertically. It is effortless to find that the best image numbers of SRCNN, FSRCNN, and ours are more than half of the testing dataset, and our algorithm is the most one.

At last, we compared the consuming time of our algorithm with others. We also selected 200 images from BSD500 randomly, super-resolved them with those algorithms, and then recorded the average PSNR and the consuming time for each algorithm. The average PSNR and consuming time for each algorithm were shown in Figure 13. The X-coordinate represents the PSNR and the Y-coordinate represents the consuming time in Figure 13. The conclusion is that the average PSNR of our algorithm is the largest one, and its consuming time lies at midlevel which is less than GPR and self-EXSR.

4.3. Experimental Analysis. From the above experimental results, we can find that our algorithm has certain advantages in the quality of reconstructed image compared with other state-of-the-art algorithms, and the consuming time is at the intermediate level. By literature retrieval, we noticed that there are some formal resemblances between the algorithms [32, 33, 38, 40, 42] and our proposed, but they are different in nature. The algorithms [32, 38, 40, 42] achieved the task of image restoration with the help of GMM. For an image patch to be restored, the assumption that this image patch...
The number of Gaussian component $K$ and the quality of reconstructed images (when $K = 8$, the quality of the reconstructed image is best).

The relationship between the balance weight $\lambda$ and the quality of reconstructed images (when $\lambda = 0.5$, the quality of the reconstructed image is best).

is generated by just some Gaussian component with highest probability is necessary, and then to estimate the clear version of this patch by wiener filtering. However, our algorithm did not need such assumption, and the cost function of our algorithm has a closed form solution. In paper [33], a prior that ‘clearer image has stronger self-similarity’ is used to deblur the natural image. This prior proposed in [33] is similar to the nonlocal self-similarity algorithm proposed in this paper, but the expressions of the two priors are completely different. The prior proposed in [33] directly calculated the distance between the blurred image patch and its nearest neighbor found in the lower scale of the input image. The result of this estimation is essentially the weighted combination of these nearest neighbors of the blurred patch. However, our proposed algorithm is to estimate the HR image patch with maximum value of the probability density function. Meanwhile, the algorithm in [33] still needs the step of searching nearest neighbors which is time-consuming and inaccurate; however, the algorithm in this paper can avoid this step. The algorithm presented in [40] is formally closest to our proposed algorithm; both paper [40] and our proposed algorithm resorted to probability density function trained with LR/HR image patch pairs to estimate HR image. However, the algorithm presented in [40] still needs the restricted assumption and the wiener filtering solution used in papers [32, 38, 42] to estimate. More importantly, the wiener filtering solution is equivalent to the mean vector $\mu_x$ of the specific Gaussian distribution prior of the algorithm in this paper, and the information of its covariance matrix $\Sigma_x$ is not used in paper [40], which reduces the accuracy of the results. Based on the experimental results presented in paper [40], our algorithm is far more efficient because it exploits the information of covariance matrix $\Sigma_x$.

The time complexity of one algorithm is an important index to evaluate its quality. The algorithm in this paper is mainly composed of two parts: model training and super-resolution. EM algorithm was used to train the Gaussian mixture model, and the time complexity is $O(n^2)$, where $n$ is the number of low resolution and high-resolution patch pairs. Since the algorithm in this paper only uses the input image itself for training, the number of image pairs will
not be too large to consume unacceptable time. The super-resolution part would take up most of the consuming time of the algorithm, since a large number of matrix inversion and matrix multiplication operations were necessary in this step, and the time complexity of this step is \(O(n^3)\). However, the weight \(\phi_k(x_i^t)\) is sparse, which make a large number of \(\mu_k\) and \(\Sigma_k\) set to be 0 directly when \(\phi_k(x_i^t)\) is very small or even 0. \(P^T_i \Sigma_i^{-1}\) was used in both of the two terms of (22) which can be computed one time and then restored the result for the second term. As a whole, the total consuming time by the proposed algorithm is acceptable.

5. Conclusion

In this paper, the property that the self-similarity of clear natural images is stronger than other versions is considered as a prior and integrated into the image super-resolution reconstruction framework. With the help of training joint GMM using LR and HR patch pairs, the prior is represented as a special Gaussian distribution. The algorithm in this paper has a closed solution, which does not require sophisticated training and time-consuming nearest neighbor searching. Therefore, the quality of reconstructed HR image with our proposed is better than other state-of-the-art algorithms. In the following future work, we hope to combine the image sparsity with the prior presented in this paper to further improve the quality of the reconstructed images.

Data Availability

The datasets involved are BSD500 (Figures 1, 2, 6, 7, and 9), RetargetMe (Figures 1, 3, and 8), and some common used images (Figures 4, 5, and 9) in image processing field, and all the experiments (such as Tables 1 and 2 and Figures 9, 10, 11, and 12) were finished with the help of Dataset BSD500 and the common used images. All these datasets were downloaded from the third-party websites. The specific addresses are as follows: BSD500: https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html; RetargetMe: http://people.csail.mit.edu/mrub/retargetme/; and the common used images: http://vllab.ucmerced.edu/wlai24/LapSRN/; these dataset provided by the three websites are all free for any researchers.

Conflicts of Interest

The authors declare that they have no conflicts of interest. Thus, there are no conflicts between the manuscript and these dataset providers.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61877013), the Natural Science Foundation of Guangdong Province (2017A030310618), the Science and Technology Planning Project of Guangdong Province (2016A020210131), and Laboratory of Language Engineering and Computing Project in Guangdong University of Foreign Studies (LEC20162BKT004). Thanks are due to dataset providers for their generosity.

References


Mathematical Problems in Engineering 13


