

Research Article

Single Image Superresolution Using Maximizing Self-Similarity Prior

Jianhong Li,¹ Yarong Wu,² Xiaonan Luo,^{1,3} Chengcai Leng,⁴ and Bo Li⁴

¹National Engineering Research Center of Digital Life, Sun Yat-sen University, Guangzhou 510006, China

²Science and Technology Department, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China

³Beijing Key Laboratory of Multimedia and Intelligent Software Technology, College of Metropolitan Transportation, Beijing University of Technology, Beijing 100124, China

⁴School of Mathematics and Information Science, Nanchang Hangkong University, Nanchang 330063, China

Correspondence should be addressed to Xiaonan Luo; lnslxn@mail.sysu.edu.cn

Received 5 September 2015; Revised 10 November 2015; Accepted 12 November 2015

Academic Editor: Daniel Zaldivar

Copyright © 2015 Jianhong Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Single image superresolution (SISR) requires only one low-resolution (LR) image as its input which thus strongly motivates researchers to improve the technology. The property that small image patches tend to recur themselves across different scales is very important and widely used in image processing and computer vision community. In this paper, we develop a new approach for solving the problem of SISR by generalizing this property. The main idea of our approach takes advantage of a generic prior that assumes that a randomly selected patch in the underlying high-resolution (HR) image should visually resemble as much as possible with some patch extracted from the input low-resolution (LR) image. Observing the proposed prior, our approach deploys a cost function and applies an iterative scheme to estimate the optimal HR image. For solving the cost function, we introduce Gaussian mixture model (GMM) to train on a sampled data set for approximating the joint probability density function (PDF) of input image with different scales. Through extensive comparative experiments, this paper demonstrates that the visual fidelity of our proposed method is often superior to those generated by other state-of-the-art algorithms as determined through both perceptual judgment and quantitative measures.

1. Introduction

The task of single image superresolution (SISR) is concerned with generating a high-resolution (HR) image from an input image of low resolution (LR). Since the same LR image may hypothetically correspond to multiple HR images, the solution space is inherently ambiguous. To resolve the ambiguity, existing methods often rely on certain image priors for selecting the preferred HR reconstruction result.

Existing SISR methods can be broadly classified into three main categories: interpolation-based methods [1–4], reconstruction-based methods [5–9], and learning-based methods [10–16]. Interpolation-based methods usually apply some basic function or interpolation kernel to determine the unknown pixels in the HR version. Overall, these methods are relatively easy to implement. However, they tend to perform inefficiently near image edges and produce blurring artifacts.

Reconstruction-based methods derive HR images via imposing constraints according to prior knowledge of the inverse problem [6, 7]. Shan et al. [6] design a feedback control framework, which uses a continuous function to fit a gradient density distribution for the input image as a prior for the HR reconstruction process. However, these methods often produce undesirable artifacts in the HR results, especially along salient edges.

The basic assumption of the learning-based methods is that the high-frequency details lost in a reconstructed HR image can be learnt from a set of low- and high-resolution image pairs. This category of methods generates an HR image from a single LR image by referring to pairs of low- and high-resolution images as the training data. Extensive research results have demonstrated the promising potential of the approach. Freeman et al. [11, 15] proposed a learning based method applicable for processing generic images where

the mapping from LR to HR versions of an image is captured by a Markov random field inferred through belief propagation. Chang et al. [17] extended their method by applying the manifold learning theory onto the correspondence between the HR image patch space and the LR image patch space. The method generates HR patches as a linear combination of its nearest neighboring patches. However, the fixed number of nearest neighbors could cause over- or underfitting. To conquer this shortcoming, Yang et al. [13, 18, 19] proposed an approach that generates sparse linear combinations using a compact dictionary. The computed sparse representation adaptively selects the relevant patches in the dictionary to represent each patch. It is noted that constructing a compact dictionary suitable for the application is a time-consuming process. These methods severely depend on the quality and relevance of the image training data set. It tends to produce obvious artifacts and generate unwanted noise in its HR generation results.

Some recent studies show that natural images generally possess a great amount of self-similarities [14, 20–22], that is, local image structures tend to recur within and across different image scales [14, 20–22], and image superresolution can be regularized taken into account these self-similar examples instead of some external database [20–22]. Particularly, Yang et al. [14] use self-examples within and across multiple image scales to regularize the ill-posed classical superresolution problem. The method extends the example-based superresolution framework with self-examples and iteratively upscale the image. They show that the local self-similarity assumption for natural images holds better for small upscaling factors and the patch search can be conducted in a restricted local region, allowing very fast practical implementation. However, the main problem rests with the fact that the reconstructed edges are usually too sharp to look natural.

Our work is inspired by the observation that almost all small image patches usually reappear themselves at other positions in natural images or its different scales. In conformity with the above observation, the basic assumption we design in this study is that there should at least exist one resemble patch in the input low resolution image for a patch randomly extracted from the well-reconstructed HR image.

The main contributions of this paper include the following:

- (1) Inspired by the idea that each patch in the reconstructed SR result should resemble patches in the original input image, we propose a new algorithm that infers HR images from their LR versions through multiresolution self-similarity maximization. The results of the new method outperform those generated by the state-of-the-art algorithms through comparative experiments.
- (2) For finding the optimal HR image, we introduce GMM to train on a patch set for approximating the joint probability density function of input image with different scales and uses an iterative scheme to resolve the cost function. This framework is easy to be adapted to other problems such as image deblurring and image denoising.

The rest of this paper is organized as follows. We first state the study motivation in Section 2. Section 3 then introduces the proposed algorithm, including how to design the cost function, how to construct and train the reconstruction model, and how to solve the cost function. Section 4 presents the results of our approach as well as comparing them with those produced by the peer methods. Finally, we conclude this paper with discussions on future work directions.

2. Motivation

Recently, expected patch log likelihood (EPLL) framework [23] using GMM prior for image denoising was proposed with its performance comparable to the state-of-the-art algorithms. EPLL trains a GMM using external natural image patches and, for each patch extracted from the unclean image, finds its highest probability solution in the model by the iterative splitting algorithm (for more details, the reader can refer to [23]). GMM prior shows its powerful denoising ability. However, this scheme is unsuitable for solving SISR problem, except for the time consuming during training phase, also because there is only one-scale image patches in the training data; thus it cannot introduce high frequency in the SR result and adopt the Wiener filter solution leading to damaging the fine details. We design a simple experiment to illustrate this point which was displayed in Figure 1. In this experiment, first, we upsample the small natural image “bear” with bicubic interpolation method as the input one (but adding no noise in it). Processing it with EPLL (the result in Figure 1(c)), we can notice that the salient edges are preserved very well, but the texture and details are almost erased. The result of EPLL is in agreement with our analysis. Processing the small natural image “bear” with our method, the result (Figure 1(d)) is sharper than the results of bicubic interpolation and EPLL methods. The algorithm of EPLL [23] is specialized for image denoising, which can preserve edge and erase the noise, and, however, it cannot add any correct high-frequency content in the results. So if the input is an image without high frequency, the result of EPLL algorithm is undesirable and our method can refine the result by finding some high frequency in the input image itself. Obviously, it is unfair for the specific denoising algorithm EPLL to be used for finding high frequency; we just use this experiment to illustrate the function of our algorithm.

Inspired by EPLL work, and also due to the image property of self-similarity between different scales, thus the input LR image and its filtering version can serve as a sufficient training data set for learning the model of the image scene. We propose a SISR method which trains GMM on the training data constructed by concatenating the different scales of the input image and can introduce the correct high-frequency component and protect the details in the HR result.

3. Our Approach

We first explain the notations appearing in our SISR scheme with the help of Figure 2. In Figure 2, the input image is denoted by $Y \in R^{\sqrt{n} \times \sqrt{n}}$, from which we obtain its low-frequency component image $Y_0 \in R^{\sqrt{n} \times \sqrt{n}}$ by low pass Gaussian

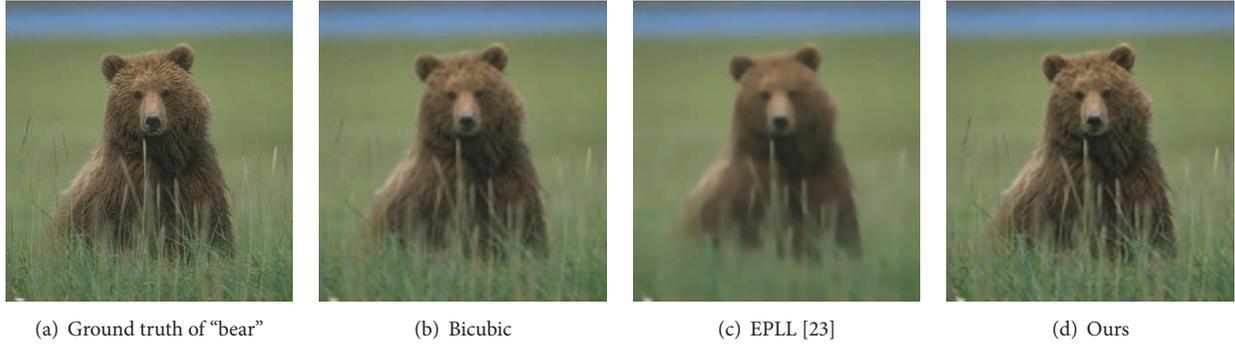


FIGURE 1: The comparison between the results of EPLL and the proposed method. (a) The ground truth image. Through downsampling and upsampling the image by the bicubic method, we derive (b), which is used as the input for producing (c) and (d) using the EPLL and the proposed method, respectively. From (c), we can see that EPLL fails to capture the image texture and its fine details. In comparison, the result of the proposed method (d) appears more natural and realistic; in particular the edges and textures regions are better preserved.

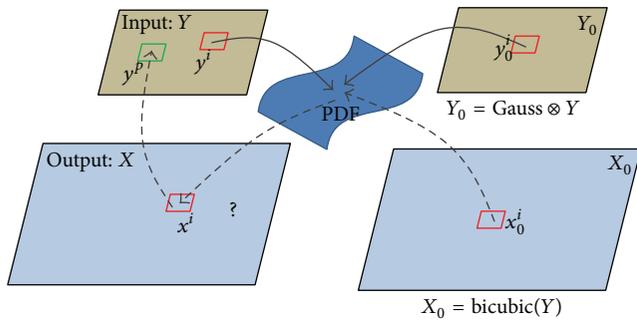


FIGURE 2: The framework of our approach. For a random patch x^i in X , we wish there exists at least one very similar patch y^p in Y . We find the joint PDF of the data pair $\{y_i, y_0^i\}_{i=1}^n$ and thus solve the optimal x^i when the data pair (x^i, x_0^i) substitute into the joint PDF.

filtering. We upsample Y using bicubic interpolation by a factor of s to get $X_0 \in R^{\sqrt{N} \times \sqrt{N}}$, $N = s^2 n$, and s is the upsampling factor. We use X_0 to approximate the low-frequency component version of the unknown high-resolution image $X \in R^{\sqrt{N} \times \sqrt{N}}$. From Y , Y_0 , and X_0 , the target is to estimate the HR image X . We use y^i and y_0^i to denote i th $a \times a$ image patch sampled from Y and Y_0 , respectively, and x^i and x_0^i to denote i th $a \times a$ image patch sampled from X and X_0 , respectively. $\{y_0^i\}_{i=1}^n$ and $\{x_0^i\}_{i=1}^N$ are thus referred to as the low-resolution image patches because they are missing high-frequency components, while $\{y^i\}_{i=1}^n$ and $\{x^i\}_{i=1}^N$ are referred to as their high-resolution counterparts separately. i represents the location of the patch in the corresponding images, and, in our scheme, the overlapped pixels between two neighbored patches are $(a-1)$; for the boundary pixels, we extend the image by symmetrical method. So the number of total patches extracted from one image is the same as the pixels of the image.

3.1. The Cost Function Based on Maximizing Self-Similarity Prior. For SISR problem, the generation of an LR image from the underlying HR image is expressed as

$$Y = DHX + w, \quad (1)$$

where $Y \in R^n$ is the observed LR image and $X \in R^N$ is the original HR image, $s = \sqrt{N}/n$ is the up-sampling factor, and $w \in R^n$ is assumed to be an additive Gaussian noise. Y , X , and w are all represented in vectorial form. $D \in R^{n \times N}$, $H \in R^{N \times N}$ are downsampling and filtering matrix, respectively.

For a random patch x^i in X , Freeman et al. [11, 15], Chang et al. [17], Yang et al. [14], and Dong et al. [12] all want to find one nearest neighbor patch y^p in Y , by approximating the distance between x^i in X and y^p in Y to the distance between x_0^i in X_0 and y_0^p in Y_0 . However, the approximation is not accurate, since the correspondence between the LR image patches and their counterparts (HR image patches) and the neighborhood relationship cannot be preserved perfectly due to the ‘‘one-to-many’’ mapping existing between one LR image and many HR images.

If we have already got the reconstructed HR image X , according to the self-similarity property, a random patch x^i in X should at least have one nearest neighbor patch y^p in Y , and for another random patch x^j in X , being not afraid of the overlapping situation in the low resolution image Y , it should also at least have one nearest neighbor patch y^q in Y . We use $p(P_i X)$ to measure the similarity between x^i and y^p . P_i is a matrix to extract the patch x^i from X . The larger the probability value $p(P_i X)$, the more similar the appearance between x^i and y^p . For any patch x^i extracted from the reconstructed HR image, we hope the probability $p(x^i)$ of the patch obtains the maximizing value. And we call the property of the reconstructed HR image Maximizing Self-Similarity Prior (MSSP). So, for the whole reconstructed HR image, we can write out the cost function as

$$\arg \min_X \|DHX - Y\|_2^2 - \lambda \prod_{i=1}^N p(P_i X), \quad (2)$$

where λ is the parameter for balancing the effect of the fidelity term $\|DHX - Y\|_2^2$ and the regularization prior $\prod_{i=1}^N p(P_i X)$.

3.2. Solving. Since there are so many multiple operations in the second term of (2), it is difficult to solve, and $p(P_i X)$ represents probability which is always positive real value; thus

we can use log function with the second term in this equation, and the cost function is changed as follows:

$$\arg \min_X \|DHX - Y\|_2^2 - \lambda \sum_{i=1}^N \log p(P_i X). \quad (3)$$

The cost function is nonconvex and difficult to solve. For finding the optimization solution, we introduce a set of auxiliary variables $\{z_i\}_{i=1}^N$, the overlapped patch $P_i X$ corresponding to z_i . Thus the cost function is changed to

$$\begin{aligned} \arg \min_{X, \{z_i\}_{i=1}^N} & \|DHX - Y\|_2^2 - \lambda \sum_{i=1}^N \log p(z_i) \\ & + \beta \sum_{j=1}^N \|P_j X - z_j\|_2^2. \end{aligned} \quad (4)$$

Note that when $\beta \rightarrow +\infty$, since we want to find the minimize solution of (4), $P_j X$ is equal to z_j , and the solution of the equation will be converged. We solve this equation by taking an iterative way. Firstly, while we keep z_i fixed, solve for X . This can be solved by setting the derivative of this equation about X to zero and we can get the result

$$\begin{aligned} \widehat{X} = & \left(H^T D^T D H + \beta \sum_{j=1}^N P_j^T P_j \right)^{-1} \\ & \cdot \left(H^T D^T Y + \beta \sum_{j=1}^N P_j^T z_j \right), \end{aligned} \quad (5)$$

where $(A)^T$ represents transposing matrix A , $(A)^{-1}$ solves the inverse matrix of A , and j is for all overlapping patches in X . And then solving for $\{z_i\}_{i=1}^N$ gives \widehat{X} . After each iteration, we increase β and continue to do the next iteration. In next two following subsections, we will elaborate how to solve $\{z_i\}_{i=1}^N$.

3.2.1. Training. For solving each auxiliary variable z_i , it just means estimating the most likely image patch under prior of MSSP. However, we have no HR image X in practice; thus it is difficult to resolve directly. Here, we consider the relationship between Y and Y_0 and extract patches $\{y^p\}_{p=1}^n$ and $\{y_0^p\}_{p=1}^n$ from the input image Y and the Gaussian filtering version Y_0 separately. For representing patch textures and details rather than absolute intensities, we subtract the mean value for each patch. And then constructing the training data set D , in which each element $d^p \in R^{2a^2}$ is formed by concatenating y^p and y_0^p in vectorial form,

$$d^p = \begin{bmatrix} y_0^p \\ y^p \end{bmatrix}, \quad p = 1, \dots, N. \quad (6)$$

Gaussian mixture model (GMM) is a parametric probability density function (PDF) that is represented as a weighted sum of Gaussian densities. GMM can approximate an arbitrary probability density function accurately [24]. For

a random variable $d \in R^{2a^2}$, we learn a GMM over these concatenated vectors and write its PDF by GMM as

$$p(d) = \sum_{k=1}^K \alpha_k N(d | \mu_k, \Sigma_k), \quad (7)$$

where K is the number of Gaussian mixture components and α_k , μ_k , and Σ_k are the parameters of weight, mean vector, and covariance matrix of k th Gaussian mixture component separately. Using EM algorithm to estimate these parameters is not hard.

3.2.2. Predicting. We consider the parameters $\{\alpha_k, \mu_k, \Sigma_k\}$ and can rewrite $\mu_k = [\mu_k^{y_0}, \mu_k^y]^T$, where $\mu_k^{y_0}$ and μ_k^y are the mean vectors of image patches $\{y_0^p\}_{p=1}^N$ and $\{y^p\}_{p=1}^N$ separately; let us also introduce the notation $\Sigma_k = \begin{bmatrix} \Sigma_k^{y_0} & \Sigma_k^{y_0 y} \\ \Sigma_k^{y_0 y} & \Sigma_k^y \end{bmatrix}$ for the covariance matrix of the model component densities. According to the self-similarity between the different scales of one image, for a random patch z^i in the reconstructed HR image, we can predict the occurrence probability in the original input image as

$$p(z^i | x_0^i, \{\alpha_k, \mu_k, \Sigma_k\}_{k=1}^K) = N(z^i | \widehat{\mu}, \widehat{\Sigma}), \quad (8)$$

where

$$\begin{aligned} \widehat{\mu} &= \sum_{k=1}^K \varphi_k(x_0^i) \left[\mu_k^y + \Sigma_k^{y y_0} (\Sigma_k^{y_0})^{-1} (x_0^i - \mu_k^{y_0}) \right], \\ \widehat{\Sigma} &= \sum_{k=1}^K \varphi_k^2(x_0^i) \left[\Sigma_k^y + \Sigma_k^{y y_0} (\Sigma_k^{y_0})^{-1} \Sigma_k^{y_0 y} \right], \end{aligned} \quad (9)$$

$$\varphi_k(x_0^i) = \frac{\alpha_k N(x_0^i | \mu_k^{y_0}, \Sigma_k^{y_0})}{\sum_{k'=1}^K \alpha_{k'} N(x_0^i | \mu_{k'}^{y_0}, \Sigma_{k'}^{y_0})}.$$

Based on (3), prediction can be obtained by taking the expectation result:

$$z^i = E(z^i | x_0^i, \{\alpha_k, \mu_k, \Sigma_k\}_{k=1}^K) = \widehat{\mu}. \quad (10)$$

At last, the algorithm for SISR using maximizing self-similarity prior is summarized as in Algorithm 1.

4. Experiments and Analysis

4.1. Parameter Setting. Experiments have been conducted to evaluate the result of our method in comparison with several state-of-the-art algorithms. We start by using the original image as LR input and upsample it with a scale factor of 2. For further upsampling, we used the previous output as the input and solved its HR image. Note that, for a color image, it is first transformed from RGB to YIQ. Then, the Y channel (intensity) is upsampled by our algorithm because human vision is more sensitive to brightness change. I and Q channels are interpolated to the desired size by the bicubic interpolation method. Finally, the three channels

Input: LR image Y
(1) $X_0 = \text{Bicubic}(Y)$;
(2) $Y_0 = \text{Gaussian} \otimes Y$
(3) Extract patches $\{y^p\}_{p=1}^n$ and $\{y_0^p\}_{p=1}^n, \{x_0^i\}_{p=1}^N$;
(4) construct training data $d^p = [y_0^p, y^p]^T$;
(5) train GMM;
(6) initialize X with X_0 ;
(7) **for** $i = 1 : t // t$ represents the number of the iterative times
 Solve X with (5), keeping $\{z^i\}_{i=1}^N$ fixed;
 Update X ;
 Solve $\{z^i\}_{i=1}^N$ with (10), keeping X fixed;
 Update $\{z^i\}_{i=1}^N$;
end
Output: HR reconstructed Image X

ALGORITHM 1: SISR using MSSP.

are combined to form the final HR result. The input LR image Y is generated from the original HR image by a down-sampling and filtering process, the low-frequency band X_0 of the unknown HR image is approximated by bicubic interpolation from Y , and the low-frequency band of the input LR image Y_0 is obtained by low pass Gaussian filtering with a standard deviation of 0.5. In all experiments, the size of all patches is 5×5 ($a = 5$), $\lambda = 0.5$, $\beta = (t \times 10)^2$, and t represents the t th iteration. T represents the number of the iterative times; in this paper, we set $T = 5$. As we know, the amount of GMM component densities K is important to the performance of the joint PDF learning. However, the accuracy and the efficiency should be trade-off in practical implementation. In our experiments, we apply a Bayesian-based model selection criterion [25] to determine the number of GMM components by minimizing the following metric:

$$K = -2 \sum_{i=1}^N \log \left(p \left(\{d^p\}_{p=1}^N \mid \{\alpha_k, \mu_k, \Sigma_k\} \right) + \delta \log N, \quad (11)$$

where δ is the total number of model parameters.

4.2. Visual Analysis. Subjective visual effect is a significant evaluation to superresolution models, which could reflect the advantages and disadvantages intuitively. In this section, we test our model and make some comparisons to the state-of-the-art models.

As mentioned before, Glasner et al. [20] emphasized the importance of image self-similarity to superresolution problem. Therefore, in Figure 3, we present a comparison with respect to Glasner's patch recurrence (PR) model, our model, and the ground truth. As observed in Figure 3(b), Glasner's patch has some artifacts along the sharp edges and tends to be smoother. By contrast, our model could obtain an output closer to the ground truth. The difference between both of the results could be observed clearly by the close-up and its corresponding pseudocolor version.

In Figure 4, we adopt the natural images which have obvious self-similarity texture and structure components to

evaluate our model. For objectiveness, we select the Gaussian-type model to compare. Figure 4 presents the results which produced by He's Gaussian process regression (GPR) model [22] and ours. In the texture case, we note that our result has better detail performance rather than blurring. And in the structure case, we could see that some artifacts appear in GPR result but not in ours.

Furthermore, in 2x upsampling, we compare between Shan's method [6], Glasner's patch recurrence method [20], Yang's SCSR method [13], Peleg's statistical prediction model [26], and ours. The tested images include such characteristics of sharp edges, complicated textures, and repeated structures. In Figure 5, by the close-up, we observe that Shan's, Glasner's, and Peleg's results would yield some artifacts along the sharp edges, for example, the wheel image in the second line. And Yang's SCSR model would generate a relative blurred result. Generally speaking, our model could obtain a series of satisfactory results in these conventional tested images.

4.3. Objective Evaluation. Peak single-to-noise ratio (PSNR) and structure similarity (SSIM) index are used as the objective measures. PSNR is defined as

$$\text{PSNR} = 10 \log_{10} \frac{m \times 255^2}{\sum_i \hat{x}_i - x_i}, \quad (12)$$

where x is the original HR image, \hat{x} is the SR result we approximated, and m is the total number of pixels in the image.

SSIM is more consistent with human eye than PSNR. The SSIM metric is calculated on various windows of an image and the measure between two windows x and y of common size $N \times N$ is

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_x\sigma_y + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (13)$$

where μ_x, μ_y are the average values of x and y , σ_x^2, σ_y^2 are the variances of x and y , and c_1, c_2 are two variables to stabilize the division with weak denominator. From the definition, we can

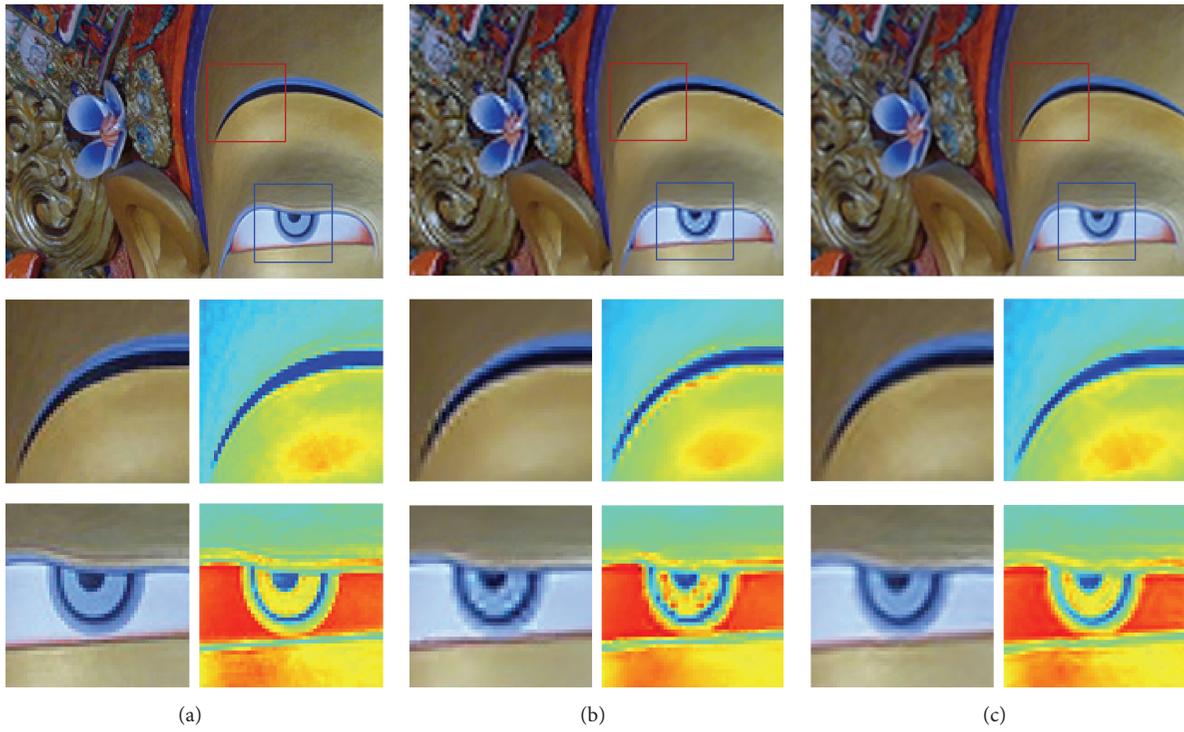


FIGURE 3: Comparisons with the self-similarity measurement. (a) Ground truth. (b) Glasner's PR [20]. (c) Our proposed method. Note the artifacts in (b) and ours are very close to the ground truth.

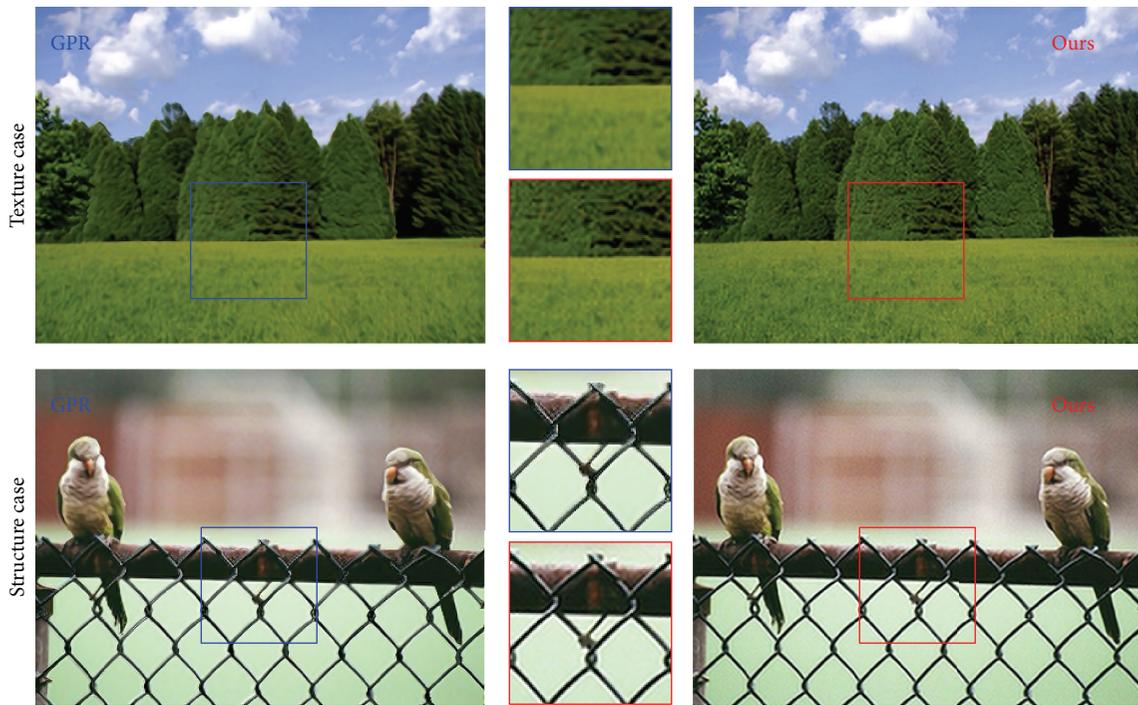


FIGURE 4: Texture and structure cases analysis. He's Gaussian process regression model [22] is compared with ours.

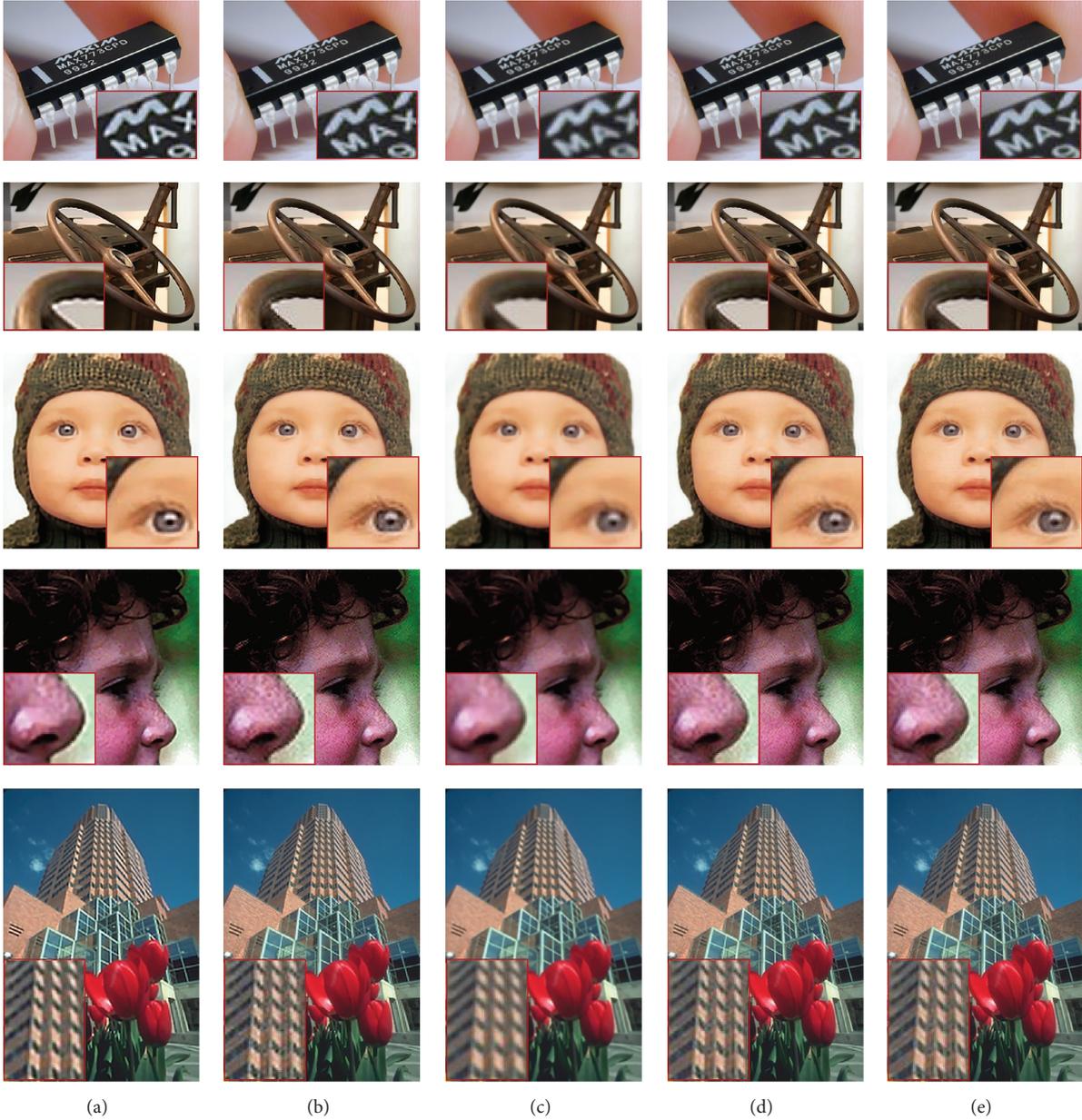


FIGURE 5: Comparisons with 2x magnification. (a) Shan's method [6]. (b) Glasner's patch recurrence method [20]. (c) Yang's SCSR [13]. (d) Peleg's SPM [26]. (e) Ours.

see that SR results with better quality should provide larger PSNR and SSIM values.

In our experiments, fifteen images are randomly selected from RetargetMe benchmark (<http://people.csail.mit.edu/mrub/retargetme/>), as illustrated in Figure 6. We first apply the Lanczos resampling to yield the low-resolution version (0.5x) and then recover to the original size to compare with the ground truth. Five state-of-the-art models are tested, including Glasner's patch recurrence method [20], SCSR [13], KRR [27], GPR [22], and SPM [26]. The PSNR and SSIM metrics are used to evaluate these results, and the corresponding scores are, respectively, recorded in Table 1. Note

that our method could obtain the best or the second-best performance in most of the test images.

For saving the computing time, we can compute $(H^T D^T D H + \beta \cdot \sum_{k=1}^N P_i^T P_i)^{-1}$, $(D H)^T$ previously, compute $\Sigma_k^{y_0} (\Sigma_k^{y_0})^{-1}$, $\varphi_k(x_0^i)$, and $(D H)^T Y$ after finishing the GMM training phrase, and use it directly in the iterative process. Figure 7 provides the numerical comparison on executing time with the state-of-the-art algorithms. The images are upscaled from 256×256 to 768×768 with the conventional images. The hardware configuration of our modern computer is Intel CPU 1.7 GHZ, 4 GB memory, and the software platform is Matlab 7.1.



FIGURE 6: Fifteen test images for effectiveness validation. From left to right: bells, bike, Buddha, butterfly, child, eagle, face, family, Fatem, fish, surfer, tiger, trees, two birds, and waterfall. All the test image are extracted from RetargetMe data set. And the validation data are recorded in Table 1.

TABLE 1: PSNR and SSIM for scale factor of 2.

	PR [20]	SCSR [13]	KRR [27]	GPR [22]	SPM [26]	Ours
Bells	25.401	29.477	29.527	28.213	29.568	29.511
	0.838	0.904	0.904	0.889	0.907	0.905
Bike	21.602	25.801	25.801	24.110	25.803	25.884
	0.713	0.845	0.847	0.779	0.846	0.848
Buddha	28.170	32.033	32.187	28.338	32.168	32.194
	0.851	0.915	0.919	0.844	0.918	0.916
Butterfly	27.337	31.743	32.128	26.352	31.883	31.983
	0.940	0.969	0.974	0.931	0.97	0.973
Child	26.435	29.941	30.154	26.392	30.021	30.248
	0.893	0.941	0.945	0.882	0.943	0.944
Eagle	27.715	30.293	30.676	27.740	30.280	30.339
	0.848	0.909	0.911	0.837	0.912	0.908
Face	30.307	37.054	36.953	30.453	37.209	37.381
	0.920	0.967	0.970	0.920	0.971	0.970
Family	25.767	30.307	30.295	25.972	30.443	30.465
	0.803	0.908	0.907	0.793	0.910	0.911
Fatem	29.091	35.384	35.620	28.701	35.788	35.857
	0.882	0.953	0.956	0.875	0.957	0.956
Fish	27.304	34.609	35.160	27.204	34.372	34.934
	0.859	0.950	0.954	0.860	0.950	0.952
Surfer	25.294	28.178	28.199	25.357	28.176	28.236
	0.736	0.820	0.820	0.712	0.818	0.822
Tiger	23.727	29.103	29.143	23.643	29.248	29.321
	0.883	0.946	0.948	0.880	0.949	0.948
Trees	27.848	30.678	30.651	27.928	30.794	30.780
	0.734	0.851	0.850	0.752	0.850	0.854
Two birds	21.112	26.262	27.024	21.525	26.309	26.487
	0.687	0.756	0.767	0.688	0.756	0.757
Waterfall	23.876	26.329	26.421	25.201	26.390	26.360
	0.771	0.838	0.841	0.794	0.840	0.840

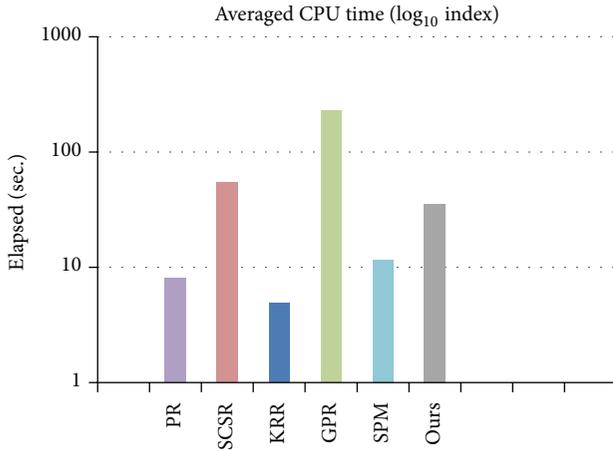


FIGURE 7: Comparison of the averaged CPU time between different methods.

4.4. Analysis. By analyzing (8) and (10), we can notice that z_i is decomposed into K layers, and, for each layer, the weight is $\varphi_k(x_0^i)$. This operation can avoid the influence of different frequencies. In each layer, we can see that if $\Sigma_k^{y_0}(\Sigma_k^{y_0})^{-1} = I$, the result of z_i is degenerated into the result of the bicubic interpolation method. In fact, we can consider $\Sigma_k^{y_0}(\Sigma_k^{y_0})^{-1}$ as transforming matrix in which $\Sigma_k^{y_0}(\Sigma_k^{y_0})^{-1}y_0^i = y^i$. We can assume that the relationship between $\{y_0^i\}_{i=1}^N$ and $\{y^i\}_{i=1}^N$ is similar to the relationship between $\{x_0^i\}_{i=1}^N$ and $\{x^i\}_{i=1}^N$. Thus $x^i \approx \Sigma_k^{y_0}(\Sigma_k^{y_0})^{-1}x_0^i$. So our approach is often superior to those generated by other state-of-the-art algorithms.

5. Conclusion

In this paper, a novel algorithm is proposed for SISR based on the concept of image self-similarity maximization. The proposed method carefully observes and leverages a pair of local and global image constraints between different resolutions of an image. The local constraint dictates that a random patch in the reconstructed HR image should exhibit visual similarity with its corresponding image region in the LR image; the global constraints state that the reconstructed HR image should perceptually resemble the input LR image through downsampling and filtering. The comparative experimental results show that the results generated by the new method preserve details of image more realistically, as determined both perceptually through subjective evaluation and quantitatively via numeric measurements. In the future, we plan to adapt and migrate the method for the problem of video superresolution.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (61320106008, 61262050, and

61363049), NSFC-Guangdong Joint Fund (no. U1135003), Guangdong Province Special Fund Research Project for Scientific and Professional Construction of University (2013WYXM0058), and Guangdong Province Science and Technology Planning Project (2013B020314019).

References

- [1] H. S. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 6, pp. 508–517, 1978.
- [2] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1521–1527, 2001.
- [3] L. Zhang and X. Wu, "An edge-guided image interpolation algorithm via directional filtering and data fusion," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2226–2238, 2006.
- [4] M. Li and T. Q. Nguyen, "Markov random field model-based edge-directed image interpolation," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1121–1128, 2008.
- [5] R. Fattal, "Image upsampling via imposed edge statistics," *ACM Transactions on Graphics*, vol. 26, no. 3, article 95, 2007.
- [6] Q. Shan, Z. Li, J. Jia, and C.-K. Tang, "Fast image/video upsampling," *ACM Transactions on Graphics*, vol. 27, no. 5, article 153, 2008.
- [7] J. Sun, J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, Anchorage, Alaska, USA, June 2008.
- [8] C. Chen, H. Liang, S. Zhao, Z. Lyu, and M. Sarem, "A novel multi-image super-resolution reconstruction method using anisotropic fractional order adaptive norm," *The Visual Computer*, vol. 31, no. 9, pp. 1217–1231, 2015.
- [9] X. Lu, Y. Sun, and Y. Yuan, "Image reconstruction by an alternating minimisation," *Neurocomputing*, vol. 74, no. 5, pp. 661–670, 2011.
- [10] B. Shi, H. Zhao, M. Ben-Ezra et al., "Sub-pixel layout for super-resolution with images in the octic group," in *Proceedings of the European Conference on Computer Vision*, pp. 250–264, Zurich, Switzerland, September 2014.
- [11] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.
- [12] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part IV*, vol. 8692 of *Lecture Notes in Computer Science*, pp. 184–199, Springer, 2014.
- [13] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, 2012.
- [14] J. Yang, Z. Lin, and S. Cohen, "Fast image super-resolution based on in-place example regression," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 1059–1066, Portland, Ore, USA, June 2013.
- [15] W. T. Freeman and C. Liu, "Markov random fields for super-resolution and texture synthesis," in *Advances in Markov Random Fields for Vision and Image Processing*, The MIT Press, 2011.
- [16] M.-T. Chi, C.-Y. Yao, E. Zhang, and T.-Y. Lee, "Optical illusion shape texturing using repeated asymmetric patterns," *The Visual Computer*, vol. 30, no. 6–8, pp. 809–819, 2014.

- [17] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 1275–1282, Washington, DC, USA, July 2004.
- [18] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, IEEE, Anchorage, Alaska, USA, June 2008.
- [19] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [20] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proceedings of the 12th IEEE International Conference on Computer Vision (ICCV '09)*, pp. 349–356, IEEE, Kyoto, Japan, September–October 2009.
- [21] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Transactions on Graphics*, vol. 30, no. 2, article 12, 11 pages, 2011.
- [22] H. He and W.-C. Siu, "Single image super-resolution using Gaussian process regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 449–456, IEEE, Providence, RI, USA, June 2011.
- [23] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 479–486, IEEE, Barcelona, Spain, November 2011.
- [24] J. A. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *International Computer Science Institute*, vol. 4, no. 510, pp. 2–7, 1998.
- [25] G. Schwarz, "Estimating the dimension of a model," *Institute of Mathematical Statistics*, vol. 6, no. 2, pp. 461–464, 1978.
- [26] T. Peleg and M. Elad, "A statistical prediction model based on sparse representations for single image super-resolution," *IEEE Transactions on Image Processing*, vol. 23, no. 6, pp. 2569–2582, 2014.
- [27] K. I. Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 6, pp. 1127–1133, 2010.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

