ORIGINAL ARTICLE

Single-image super-resolution based on local biquadratic spline with edge constraints and adaptive optimization in transform domain

Danya Zhou[1] · Yepeng Liu[2] · Xuemei Li[2,3] · Caiming Zhang[2,3]

Abstract

This paper proposes a novel single-image super-resolution method based on local biquadratic spline with edge constraints and adaptive optimization in transform domain. The complex internal structure of the image makes the values of adjacent pixels often differ greatly. Using surface patches to interpolate image blocks can avoid large surface oscillation. Because the quadratic spline has better shape-preserving property, we construct the biquadratic spline surface on each image block to make the interpolation more flexible. The boundary conditions have great influence on the shape of local biquadratic spline surfaces and are the keys to constructing surfaces. Using edge information as a constraint to calculate them can reduce jagged and mosaic effects. To decrease the errors caused by surface fitting, we propose a new adaptive optimization model in transform domain. Compared with the traditional iterative back-projection, this model further improves the magnification accuracy by introducing SVD-based adaptive optimization. In the optimization, we convert similar block matrices to the transform domain by SVD. Then the contraction coefficients are calculated according to the non-local self-similarity, and the singular values are contracted. Experimental comparison with the other state-of-the-art methods shows that the proposed method has better performance in both visual effect and quantitative measurement.

Keywords Local biquadratic spline · Boundary conditions · Singular value contraction · Adaptive optimization

1 Introduction

The purpose of single-image super-resolution (SISR) is to recover high-frequency information from the low-resolution (LR) input images and generate reasonable high-resolution (HR) images that are not conflict with the LR versions. SISR is an important branch of image processing and has applications in many areas, such as medical imaging, security surveillance, satellite imagery, e-commerce and many others. There are two main types of SISR methods: interpolation-based methods and learning-based methods.

✉ Xuemei Li
xmli@sdu.edu.cn

Danya Zhou
danyazhou@yeah.net

1 School of Computer Science and Technology, Shandong University, Jinan, China

2 School of Software, Shandong University, Jinan, China

3 Shandong Co-Innovation Center of Future Intelligent Computing, Yantai, China

Classical polynomial interpolation methods [21–23] are extensively used due to their simplicity and speed of computation. However, since these methods use the commonly used polynomial fitting method to construct the interpolation surface of the image, the generated images are often too smooth and appear jagged and ringing effects in texture and edge areas. To solve this problem, some edge-oriented interpolation methods were proposed [5,20,29]. The adaptive interpolation method based on covariance was proposed by Li et al. [20]. The method first estimates the local covariance coefficient from the LR image, estimates the HR covariance according to the geometric duality between LR covariance and HR covariance, and then used these covariances to adaptively adjust the interpolation. Chen et al. proposed a fast edge guidance method [5]. Based on the analysis of the local structure of the image, this method divides the image into smooth regions and edge regions, and adopts a specific interpolation method in different regions. The method proposed by Wang et al. [29] realizes the super-resolution reconstruction of a single image through adaptive self-interpolation of gradient amplitude. The basic idea of this method is to first estimate the HR gradient field from the LR image, and then this gradient is

used as the gradient constraint in the reconstructed HR image, so that the reconstructed image can retain sharp edges. RFI [35] uses rational fractal interpolation for texture regions and rational interpolation for non-texture regions. These methods use the structural information of images to sharpen the edges, but they are easy to generate noise and distortion in the texture area. Other methods use some constraints to optimize the interpolated HR images. Iterative back projection optimizes the HR image by reducing the error between the down-sampled HR image and the input LR image iteratively, such as IBP [19], BFIBP [7], NLIBP [13], etc. NLFIBP [34] optimizes the interpolation image in combination with the nonlocal mean filter and back projection. The enlarged image has sharp edges, while the accuracy is relatively low. In addition, the nonlocal self-similarity of images is widely used in various image super-resolution methods [18,25,37]. BM3D-SR [14] enlarge the image by combining Wiener filtering in transform domain and iterative back projection framework. WSD-SR [6] is improved based on BM3D-SR. The Wiener filter in 3D transform domain is changed into one in 1D similar domain, which not only reduces the computational complexity, but also improves the image quality. LPRGSS [33] uses the self-similarity of images and the sparsity of the global structure as the constraints to optimize the interpolated image, and achieves remarkable results.

The learning-based SISR method learns the mapping from LR to HR images through training sets containing numerous HR and LR image pairs. Learning mapping methods can be divided into two categories: methods that rely on external images as the training data and methods that rely only on input LR images. These methods include sparse dictionary learning [24,27], manifold learning [4], local linear regression [26] and convolutional neural networks [11]. ANR [27] learns the sparse dictionary on fixed dictionary atoms. SF [30] achieves outstanding performance through clustering and the corresponding learned functions. A^+ [28] improves ANR based on the idea of SF, and combines the best characteristics of ANR and SF. The deep learning method SRCNN [10] based on convolutional neural network directly learns the end-to-end mapping between LR and HR images, and achieves remarkable results. FSRCNN [12] raises the speed of enlargement and achieves better image quality by improving the structure of SRCNN. SelfEx [16] is based on image self-similarity. Instead of relying on external data, it learns internal dictionaries from the given image. By expanding the search space of similar blocks, the limitations of internal dictionaries are improved. MMPM [17] proposes mixture prior model to learn image features, and uses the method based on curvature difference to group image blocks. DLD [9] builds a lightweight database and learns adaptive linear regression, which maps LR image blocks directly into HR blocks, and achieves excellent results.

In this paper, we propose a new method based on local biquadratic spline with edge constraints and adaptive optimization in transform domain for SISR. The local biquadratic spline interpolation is divided into four steps. First, we use the image edge information to calculate the boundary conditions. Second, for each 3×3 LR image block, a local biquadratic spline surface is constructed, namely, 9 surface patches. Third, the weighted average of all local biquadratic spline surfaces constitutes the overall bicubic fitting surface. Finally, the sampling density is selected according to the scale factor, and the output HR image is obtained by sampling on the bicubic fitting surface. Subdividing a surface into 9 surface patches gives each surface greater interpolation flexibility. Therefore, the error is smaller in the region where the pixel value varies greatly, which improves the accuracy of the enlarged image. It is inevitable to have approximation error and lose part of high-frequency information when fitting image blocks with surfaces. To overcome these two shortcomings, we use the non-local self-similarity and data fidelity as prior constraints to optimize the image in space and transform domains. For each iteration, the approximation error is reduced in the transform domain by contracting singular values of each similar block matrix. The contraction coefficient is adaptively updated in the iteration. Then in the space domain, the high-frequency information lost in the interpolation is made up by projecting the amplified error image back to the HR image.

The major contributions of our work are summarized as follows:

1. We propose a new local biquadratic spline method with edge constraints. Each 3×3 image block is interpolated by nine C^1 continuous surface patches, which improves the interpolation accuracy and the flexibility of surface to approximate the shape of the image block.
2. A new method for calculating the boundary conditions of biquadratic spline surfaces is presented. The edge information is introduced as a constraint to reduce the jagged and mosaic effect of the enlarged image edge. The two-step method improves the image accuracy and reduces the computation.
3. We construct a novel adaptive optimization in transform domain. Compared with the traditional iterative back-projection method, the proposed model incorporates the adaptive optimization based on SVD. By using the non-local self-similarity, the interpolation error is reduced, and the quality of the convergent image is improved greatly.

The rest of the paper is arranged as follows. Section 2 describes the problems and introduces the overall process of the proposed method. In Sect. 3, the details of local biquadratic spline with edge constraints are described. Sec-

tion 4 describes the adaptive optimization in transform domain and analyzes the structure of the optimization model. Section 5 evaluates the performance of the proposed method in terms of visual effect and quantitative measurement. A summary of the paper is given in Sect. 6.

2 Problem description and algorithm framework

The object of SISR is to recover HR image containing high-frequency information from the known degraded LR image. The image degradation process can be expressed as follows:

$$Y = (f_g \otimes X) \downarrow^s \tag{1}$$

where X and Y represent HR and LR images, respectively. f_g is fuzzy filtering, and '$\otimes$' is the convolution operation. '$\downarrow^s$' represents the down-sampling operation with scale factor s. Ideally, the reconstructed HR image should degrade to produce the same image as the known LR image. However, with only a small amount of LR information, it is difficult to restore an ideal HR image. There will be errors between the LR image generated by the actual generated HR image and the input image. Iterative back-projection is a method to obtain HR image by minimizing reconstruction errors. Specifically, after the initial HR image X^0 is obtained by interpolation method, the back-projection operation is performed iteratively. Each iteration is divided into two steps: calculating the error image and projecting the enlarged error image back to HR image. The formula is as follows:

$$X^{t+1} = X^t + U((Y - (f_g \otimes X^t) \downarrow^s), s) \tag{2}$$

where X^t represents the HR image obtained after the t-th iteration, and $U(\cdot, s)$ represents the interpolation for super-resolution, with the scale factor of s. The error image between the degraded image of X^t and the input LR image Y is calculated and enlarged. Then, the reconstructed error image is projected back to X^t to complete a back-projection operation of the error image. As the number of iterations increases, the error between the enlarged image and the input image decreases gradually. When the stop condition is reached, the final HR image is output.

The optimization model of iterative back-projection is based on the objective function of minimizing the error between the iterative convergence result and the input image. But the solution of this objective function is an indeterminate problem. In other words, there may be multiple different solutions that satisfy the optimal solution. Several different HR images may degenerate into the same LR image, and the convergence result may be quite different from the ideal HR image. Therefore, the interpolation function U has an important influence on the accuracy of the output image. Traditional image interpolation algorithms, such as bicubic spline interpolation, construct a whole surface to interpolate the entire image. The disadvantage of the overall construction surface is that much local information in the image is ignored. When the pixel values are greatly different between neighborhoods, the high-frequency components of the image are easily damaged. Thus the details degrade due to the wiggle of the polynomial spline functions. The main reason for this phenomenon is that the same kernel function is used in different regions. The methods lack adaptability and are prone to produce zigzag effect on the edge areas.

In addition, the errors between the HR image and the input image are mainly distributed in the image edges and texture areas. The back-projection operation of the reconstructed error image can make up the missing high-frequency information. However, there are inevitably errors in the way that polynomial interpolation surface is used to fit the original image surface. After the degradation of the HR image, partial errors will not be reflected in the error image. Therefore, this part of errors cannot be reduced by back-projection operation, but accumulate in the iterative process, making the final convergence result far from the ideal image. Therefore, ignoring the interpolation error and directly carrying out the back-projection operation will limit the improvement of the image accuracy, resulting in fuzzy effect.

Through the above analysis, we propose a SISR method based on local biquadratic spline with edge constrains and adaptive optimization model in transform domain. The proposed algorithm first obtains the initial HR image by local biquadratic spline with edge constraints (LBSEC), and then iteratively optimizes it to obtain the final HR image. Figure 1 shows the overall process.

Each iteration is divided into two steps: optimization in transform domain and back-projection of the error image. First, the optimization in the transform domain is carried out. The similar block matrices are obtained by matching blocks. And the singular values of each matrix are contracted adaptively to get the updated image blocks. Then we aggregate the blocks to obtain the optimized HR image. Second, back-projection is accomplished by enlarging the error image with LBSEC and then projecting it to the reconstructed HR image. Continue iterating until the stop condition is reached, and output the final HR image.

3 Local biquadratic spline with edge constraints

The traditional interpolation algorithms use the method of constructing the whole surface, and uses the same kernel function in different areas. In this way, the high-frequency information of the image is easily lost, resulting in smooth
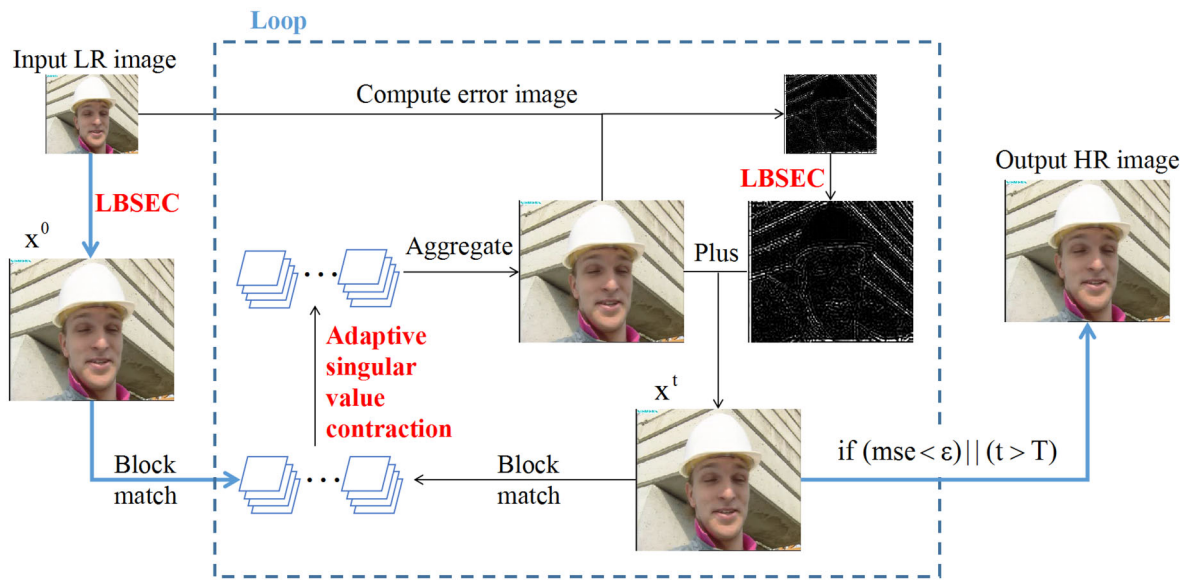
Fig. 1 The flowchart of the proposed algorithm. LBSEC represents the local biquadratic spline with edge constraints. X^0 is the initial HR image amplified by LBSEC from the input LR image, and X^t is the optimized HR image obtained in the tth iteration. Mse denotes the mean square error between the degraded HR image and the input LR image, and T is the maximum of iterations

and ringing effect. Therefore, we propose a new local biquadratic spline with edge constraints method. We construct a local biquadratic spline surface containing 9 surface patches on each 3×3 grid. Different boundary conditions are adopted in different areas. This way improves the shape preserving ability of the surfaces, and reduces the loss of the high frequency components in the regions with large image gradient. Because the biquadratic spline surface is constructed on a small 3×3 image block containing 9 pixels, the boundary conditions have great influence on the shape of the surface. They are the key to construct an ideal surface. The meaning of the ideal surface here is that the surface not only approximates the image block with high precision, but also has the shape of the image block. Constructing biquadratic spline surface is divided into two steps: calculating the boundary conditions and constructing surface. After constructing all local biquadratic spline surfaces, the integral bicubic fitting surface is formed by surface fusion.

3.1 Calculate the boundary conditions

The boundary conditions to be computed are the first partial derivatives at the pixel points. The interpolation accuracy of biquadratic splines can reach the accuracy of biquadratic polynomials. Therefore, the computational boundary conditions need to have the precision of biquadratic polynomials. When calculating the boundary conditions, we introduce the structure information of the image as a constraint. So the first partial derivative not only has high accuracy, but also keeps the shape of the edge, thus reducing the sawtooth effect of the

edge region. Assuming that the pixel values of the input LR image are sampled per unit area on the continuous original surface. The piecewise surface $g_{i,j}$ is used to approximate the original surface. Then, the pixel value can be defined as

$$P_{i,j} = \int_{j-0.5}^{j+0.5} \int_{i-0.5}^{i+0.5} g_{i,j}(x, y)dxdy \qquad (3)$$

To make the boundary conditions have biquadratic polynomial accuracy, we use the biquadratic polynomial to construct $g_{i,j}$ on the 5×5 image block (Fig. 2). That is, in the $[i-2.5,i+2.5] \times [j-2.5,j+2.5]$ region of xy plane. $g_{i,j}(x, y)$ can be written as

$$g_{i,j}(x, y) = a_0 u^2 v^2 + a_1 u^2 v + a_2 u v^2 + a_3 u^2 \\ + a_4 u v + a_5 v^2 + a_6 u + a_7 v + a_8 \qquad (4)$$

where $u = x - i$, $y = y - j$, $a_0, a_1, \ldots, a_8$ are the unknown surface coefficients.

The first partial derivatives a_6, a_7 in the surface patch $g_{i,j}(x,y)$ are the boundary conditions to be solved. There are 9 unknowns in (4). If they all participate in the calculation of a_6 and a_7, a large amount of calculation is required. To reduce the calculating amount, we solve a_6 and a_7 through the first-order difference quotient. There are 25 pixels on the 5×5 image block in Fig. 2. According to (3) and (4), 8 first-order difference quotients including a_1, a_2, a_6 and a_7 can be obtained from 8 directions in Fig. 2. And a_6 and a_7 can be solved by the 8 first-order difference quotients. In order to improve the fitting accuracy, we hope that the pixels on the
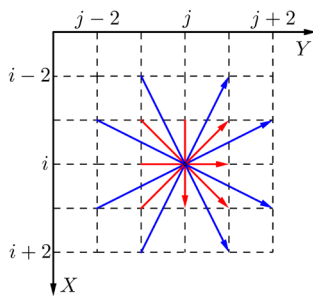
Fig. 2 First-order difference quotients in eight directions on the 5×5 image block
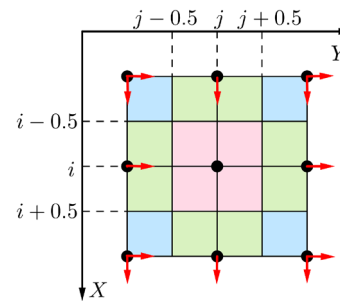

Fig. 3 The 3×3 image block is divided into 9 regions, and the adjacent regions are separated by different colors. The red arrows represent the boundary conditions. They are the first partial derivatives of the corresponding pixels. The arrow direction is the partial derivative direction

inner 3×3 image block play a greater role in determining a_6 and a_7. Therefore, we use the following two-step method to calculate a_6 and a_7.

Next, we introduce the solution method of a_6, a_7. Based on (3) and (4), the following four equations can be obtained from the first-order difference quotient:

$$\frac{1}{12}a_2 + a_6 = \frac{P_{i+1,j} - P_{i-1,j}}{2} = c_1,$$

$$\frac{1}{12}a_1 + a_7 = \frac{P_{i,j+1} - P_{i,j-1}}{2} = c_2,$$

$$\frac{13}{12}a_1 + \frac{13}{12}a_2 + a_6 + a_7 = \frac{P_{i+1,j+1} - P_{i-1,j-1}}{2} = c_3,$$

$$\frac{13}{12}a_1 - \frac{13}{12}a_2 - a_6 + a_7 = \frac{P_{i-1,j+1} - P_{i+1,j-1}}{2} = c_4 \qquad (5)$$

As shown in Fig. 2, c_1, c_2, c_3, c_4 represent the first-order difference quotients in the four directions indicated by the red arrows, respectively.

The expressions of a_6, a_7 as linear combinations of a_1, a_2 are obtained by using the least square method with constraints. The objective function is defined as

$$
\begin{aligned}
G_1(a_1, a_2, a_6, a_7) = {} & w_1(\tfrac{1}{12}a_2 + a_6 - c_1) \\
& + w_2(\tfrac{1}{12}a_1 + a_7 - c_2) \\
& + w_3(\tfrac{13}{12}a_1 + \tfrac{13}{12}a_2 + a_6 + a_7 - c_3) \\
& + w_4(\tfrac{13}{12}a_1 - \tfrac{13}{12}a_2 - a_6 + a_7 - c_4)
\end{aligned}
\qquad (6)
$$

where w_1, w_2, w_3, w_4 are the weights. By minimizing the objective function $G_1(a_1, a_2, a_6, a_7)$, that is, setting its partial derivatives with respect to a_6, a_7 as zero, we can obtain the expression of a_6, a_7 as follow:

$$a_6 = \beta_{11}a_1 + \beta_{12}a_2 + \beta_{13},$$
$$a_7 = \beta_{21}a_1 + \beta_{22}a_2 + \beta_{23} \qquad (7)$$

The second step is to solve a_1, a_2. From (3) and (4), we can get that

$$\frac{13}{6}a_1 + \frac{49}{12}a_2 + a_6 + 2a_7 = \frac{P_{i+1,j+2} - P_{i-1,j-2}}{2} = c_5,$$

$$\frac{13}{6}a_1 - \frac{49}{12}a_2 - a_6 + 2a_7 = \frac{P_{i-1,j+2} - P_{i+1,j-2}}{2} = c_6,$$

$$\frac{49}{12}a_1 + \frac{13}{6}a_2 + 2a_6 + a_7 = \frac{P_{i+2,j+1} - P_{i-2,j-1}}{2} = c_7,$$

$$\frac{49}{12}a_1 - \frac{13}{6}a_2 - 2a_6 + a_7 = \frac{P_{i-2,j+1} - P_{i+2,j-1}}{2} = c_8 \qquad (8)$$

where c_5, c_6, c_7, c_8 correspond to the first-order difference quotients in the four directions indicated by the blue arrows in Fig. 3.

Substitute (7) into (5) and (8) to get the 8 equations $h_i(a_1, a_2) = c_i$, $i = 1, 2, \ldots, 8$ about a_1, a_2. And the a_1, a_2 are determined by least square method again, namely, by minimizing the following formula:

$$G_2(a_1, a_2) = \sum_{i=1}^{8} w_i(h_i(a_1, a_2) - c_i)^2 \qquad (9)$$

The weights in (6) and (9) are defined as follows:

$$w_i = \frac{1}{1 + (\Delta_i/d_i)^2} \qquad (10)$$

where Δ_i is the second-order difference quotients in corresponding directions, and d_i is the distance between adjacent pixels. The image edges have a significant impact on the visual effect. To reflect the image edge features better, the approximate surface should be closer to the original one in the edge directions than in other directions. A smaller change of the pixel values means that the direction is more likely to be the direction of the image edge, which should be given

Table 1 PSNR (dB) and running time results of LBSEC based on overall and two-step solutions on Set5, Set14, BSD100 and Urban100 by a factor of 2

Image set	Overall solution		Two-step solution	
	PSNR (dB)	Time (ms)	PSNR (dB)	Time (ms)
Set5	34.57	642	**34.75**	**486**
Set14	30.63	1199	**30.98**	**862**
BSD100	29.93	820	**30.19**	**599**
Urban100	26.96	934	**27.37**	**843**

The best results are shown in bold

a larger weight. Finally, the a_6, a_7 are obtained by bringing a_1, a_2 into (7).

In the two-step solution, a_6 and a_7 are expressed as functions of other coefficients a_1, a_2. This way makes full use of the structural information of the image, so that the pixels on the inner 3×3 image block play a greater role in determining a_6 and a_7. Thereby, the calculated boundary conditions are more reasonable. Compared with the overall solution, the two-step one improves the fitting precision. In addition, the overall solution is that 9 unknown coefficients are all involved in the calculation. For each pixel, the unknown coefficients are solved by a nine-element linear equation set. The two-step solution is to obtain a_6 and a_7 by solving the binary linear equation set twice, which requires less calculation. In order to test the performance, we magnify images by LBSEC based on overall and two-step solutions respectively. As shown in Table 1, the latter can effectively improve the quality of enlarged images while reducing the time cost.

3.2 Construct local biquadratic spline surface

For each 3×3 image block, we construct a biquadratic spline surface patch, which consists of 9 subsurface patches. As shown in Fig. 3, the neighborhood $[i-1, i+1] \times [j-1, j+1]$ of pixel point (i, j) is divided into 9 subregions, each containing a pixel point. We construct a biquadratic surface patch on each subregion, denoted by $s_{m,n}$, and the formula is as follows:

$$
\begin{aligned}
s_{m,n}(x, y) &= b_0^{m,n} u^2 v^2 + b_1^{m,n} u^2 v + b_2^{m,n} u v^2 + b_3^{m,n} u^2 \\
&\quad + b_4^{m,n} uv + b_5^{m,n} v^2 + b_6^{m,n} u + b_7^{m,n} v + b_8^{m,n}
\end{aligned}
\tag{11}
$$

where (m, n) is the coordinates of pixel points contained in the subregion, $m = i \pm 1, i, n = j \pm 1, j. u = x - m, v = y - n$, and $b_0^{m,n}, b_1^{m,n}, ..., b_8^{m,n}$ are the unknown surface coefficients.

Because of C^1 continuity of biquadratic spline surface, the function value and first partial derivative of subsurface patches are continuous at the junction. For example, at the boundary curves $x = i \pm 0.5$ in the y direction, the function value

being continuous is the sufficient condition for $\frac{\partial s_{m,n}}{\partial y}$ being continuous. So here, we only consider the continuity of $\frac{\partial s_{m,n}}{\partial x}$, regardless of $\frac{\partial s_{m,n}}{\partial y}$. Then, the adjacent subsurface patches at the junction of $x = i \pm 0.5$ satisfy the following conditions:

$$
\begin{aligned}
s_{m,n}(m + 0.5, y) &= s_{m+1,n}(m + 0.5, y), \\
\frac{\partial s_{m,n}}{\partial x}(m + 0.5, y) &= \frac{\partial s_{m+1,n}}{\partial x}(m + 0.5, y)
\end{aligned}
\tag{12}
$$

where $m = i - 1, i, n = j \pm 1, j$. The same is true for the interface curves in the x direction. In addition, for each pixel value $P_{m,n}, m = i \pm 1, i, n = j \pm 1, j$,

$$
s_{m,n}(m, n) = b_8^{m,n} = P_{m,n}
\tag{13}
$$

We use the known conditions to solve the unknown coefficients by group way. The first group is $b_3^{m,n}, b_6^{m,n}$. The first partial derivatives of x and y directions at point (m, n) obtained above are $a_6^{m,n}, a_7^{m,n}$ respectively. We assign $a_6^{m,n}, a_7^{m,n}$ to $b_6^{m,n}, b_7^{m,n}$, which are the first partial derivatives of the subsurface patches $s_{m,n}$ in the corresponding regions. As shown in Fig. 3, the red arrows indicate the first partial derivatives. Different regions adopt different boundary conditions. Then, the boundary conditions can be expressed by the formula as follows:

$$
\begin{aligned}
b_6^{m,n} &= a_6^{m,n}, m = i \pm 1, n = j \pm 1, j, \\
b_7^{m,n} &= a_7^{m,n}, m = i \pm 1, i, n = j \pm 1
\end{aligned}
\tag{14}
$$

According to (12), the coefficients of same order term on both sides of equal sign are same, we can obtain a set of equations about $b_3^{m,n}, b_6^{m,n}$:

$$
\begin{aligned}
\frac{1}{4} b_3^{m,n} + \frac{1}{2} b_6^{m,n} + b_8^{m,n} &= \frac{1}{4} b_3^{m+1,n} - \frac{1}{2} b_6^{m+1,n} + b_8^{m+1,n} \\
b_3^{m,n} + b_6^{m,n} &= -b_3^{m+1,n} + b_6^{m+1,n}
\end{aligned}
\tag{15}
$$

where $m = i - 1, i, n = j \pm 1, j$. Substitute (13) and (14) into (15), and the solutions of the rest $b_3^{m,n}, b_6^{m,n}$ can be obtained. Similarly, $b_5^{m,n}, b_7^{m,n}$ can be obtained.

Then, we solve for $b_0^{m,n}, b_1^{m,n}, b_2^{m,n}, b_4^{m,n}$. We set the second mixed partial derivatives of the surface at four vertices of the rectangular region as zero, which can be formulated as

$$
\frac{\partial^2 s_{m,n}}{\partial x \partial y} = b_4^{m,n} = 0, m = i \pm 1, n = j \pm 1
\tag{16}
$$

A set of equations about $b_0^{m,n}, b_1^{m,n}, b_2^{m,n}, b_4^{m,n}$ can be derived from the continuity constraint at the junction of x and y directions. Substituting the known $b_4^{m,n}, b_6^{m,n}$ and $b_7^{m,n}$ into them, the unknown $b_0^{m,n}, b_1^{m,n}, b_2^{m,n}, b_4^{m,n}$ can be solved.

The nine subsurface patches are jointed together to form the fitting surface patch at the center pixel point, denoted as $f_{i,j}(x, y)$.

3.3 Surface fusion

Each biquadratic spline surface patch $f_{i,j}(x, y)$ is divided into nine subsurface patches. In this way, the surface can fit the image block more flexibly and the interpolation precision is higher. All biquadratic spline surfaces are fused into the overall bicubic fitting surface $F(x, y)$ by weighted average. The formula is as follows:

$$q_{i,j}(x, y) = (1 - u)(1 - v)f_{i,j} + u(1 - v)f_{i+1,j} \\ + (1 - u)vf_{i,j+1} + uvf_{i+1,j+1} \quad (17)$$

where $u = x - i$, $v = y - j$, $f_{i,j}$, $f_{i+1,j}$, $f_{i,j+1}$ and $f_{i+1,j+1}$ are the surfaces at four vertices of quadrilateral region. The representation of $F(x, y)$ in this region is $q_{i,j}(x, y)$. All the surface $q_{i,j}(x, y)$ joint together to form the whole fitting surface $F(x, y)$. The HR image is obtained by sampling at the corresponding density on $F(x, y)$.

4 Adaptive optimization model in transform domain

The adaptive optimization model in transform domain is improved on the basis of traditional iterative back-projection framework. Before the operation of back-projection, we optimize the interpolated image by using the non-local self-similarity, which effectively reduces the approximation error of the interpolation surface. Compared with the traditional framework, the proposed model further improves the accuracy of enlarged image and reduces the sawtooth and mosaic effect of image edges.

4.1 Adaptive optimization in transform domain

In the process of biquadratic spline interpolation, the fitting surface inevitably has approximation errors, which make the sampled HR pixel easily distorted. Therefore, we design a new adaptive optimization in transform domain to reduce the error. The interpolation algorithm which uses only local information of the image cannot reconstruct HR images with high quality. So we use the non-local self-similarity to optimize the image. The optimization includes three steps: similar block matching; calculating the contraction coefficient and contracting the singular value; block fusion.

First, similar blocks are searched in a search window centered on the reference block R_i, $i = 1, 2, \ldots, n$. By comparing the distance between R_i and all blocks in the window, we find K blocks which are most similar to the central one and form a similar block matrix M_i.

Second, due to the similarity between the blocks, the column vectors of the similar block matrix are highly correlated. Thus, the similar block matrix is low-rank. Moreover, the

singular value often contains important information of the corresponding matrix, and the importance is positively correlated with the singular value. Thus, the low rank indicates that most energy of the matrix is contained in a few large singular values. However, the fitting error will cause the fluctuation of singular values. This problem leads to small singular values, and increases the rank of similar block matrices. According to this principle, we contract the singular values to ensure the low rank of the matrices and reduce the errors. The formula of SVD for similar block matrices is:

$$M_i = U_i \Sigma_i V_i, i = 1, 2, \ldots, n \quad (18)$$

where M_i is the similar block matrix of the reference block R_i and Σ_i is the singular value matrix. The greater the singular value is, the more effective information it contains, which should be dealt with a smaller contraction. So we set the soft threshold as $thr_j = \zeta/(\sigma_j + \tau)$. σ_j is the first j singular value from big to small order. τ is a small positive number to ensure that the denominator is not zero. ζ is the contraction factor. The greater ζ the more information amputated, and the optimized image will be smoother. We design the contraction factor to be adaptively updated, rather than predetermined. The formula is as follows:

$$\zeta = \frac{\left\| Y - (f_g \otimes X) \downarrow^s \right\|_2^2}{2m} \quad (19)$$

where Y denotes the input LR image, and X is the HR image to be optimized. f_g represents a fuzzy filter, "$\downarrow^s$" is a down-sampling operation with scale factor s, and m is the number of pixels in Y. In the initial iteration, the fitting error of interpolation surface and the error between the degraded HR image and the input LR image are both large, so a larger scaling factor is applied. As the number of iterations increases, the sawtooth of HR images decreases, and fitting errors are gradually eliminated. Then, a smaller contraction factor is applied. So ζ shrinks gradually in the iterations. The diagonal matrix of contracted singular value is donated as $\hat{\Sigma}_i$. And the first j diagonal element of it is as $\hat{\sigma}_j = max\{\sigma_j - thr_j, 0\}$. Then, by replacing Σ_i with $\hat{\Sigma}_i$ in (18), the updated similar block matrix is obtained.

Third, for different estimates $p^k, k = 1, 2, \ldots, K$ of the image block p, their weights are set according to the ranks r^k of the matrices where they are located. The low rank matrix contains blocks with high similarity, and the estimated results are more reliable. So we set a larger weight for it. The fused image block $\hat{p}$ is obtained by weighted averaging of multiple estimates as follows:

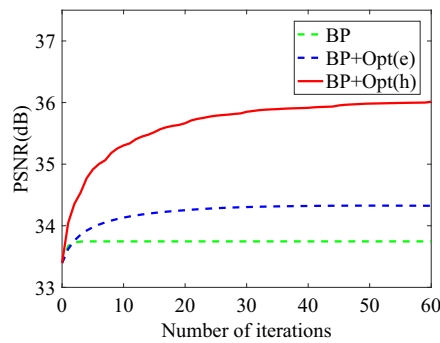$$\hat{p} = \frac{1}{W} \sum_{k=1}^{K} (1 - \frac{r^k}{N + 1}) p^k \quad (20)$$

Fig. 4 Performance comparison of 'BP', 'BP+Opt(e)' and 'BP+Opt(h)' optimization sructures on woman image with the magnification ×2

where $r \le N$, and the denominator is set as $N + 1$ to ensure weights are not zeros. $\frac{1}{W}$ is the normalization factor. After the fusion of different estimated blocks, we fuse the overlapped pixels of image blocks with a simple averaging strategy. Then, the complete reconstructed HR image is obtained.

4.2 Analysis of optimization model structure

In this section, three different optimization model structures are compared. We use 'BP' to represent the traditional iterative back-projection structure, that is, each iteration contains only the back-projection operation of the reconstructed error image. The optimization model structure in [34] is to first optimize the enlarged error image, and then project it back to the HR image in each iteration. We call this structure 'BP+Opt(e)'. Different from [34], we carry out the optimization for the intact HR image instead of the error image. We call it 'BP+Opt(h)'. We test the performance of three different structures of optimization model, as show in Fig. 4.

Compared with 'BP' that only has a back-projection operation, the optimization of HR image can reduce the fitting error and make the image accuracy higher when iterative convergence. It can be seen from the figure that 'BP+Opt(h)' effectively increases the growth rate of PSNR during iteration and further improves image quality.

In addition, it can be seen from the figure that 'BP + Opt (h)' achieves much better amplification effect than 'BP + Opt (e)'. Through adaptive optimization in transform domain, 'BP+Opt(e)' structure can use the non-local self-similarity of the image to guide the direction of iterative convergence. It can improve the quality of the final output image. However, because HR images contain more complete structural information, the results of non-local optimization for HR images are more reliable than those for error images. In this way, the error image obtained from the optimized HR image contains more useful information than that from the non-optimized HR image. Thus, the improvement space will be larger when the error image is back projected. In summary, we adopt the 'BP + Opt (h)' optimization model structure.

The overall process of the proposed single-image super-resolution algorithm based on local biquadratic spline with edge constraints and adaptive optimization in transform domain (LBSAO) is shown in Algorithm 1.

Algorithm 1: LBSAO algorithm

Input: LR image Y, scaling factor s
Output: HR image X
1) $t = 0$, initialize ε, T;
2) Compute the boundary conditions on Y;
3) Construct local biquadratic spline surface on each 3×3 grid;
4) Fuse surface patches into global bicubic surface;
5) Sample to get HR image X^0;
6) while $(mse > \varepsilon) \, \&\& \, (t < T)$ do
7)   If (t%10 == 0), update the location of similar blocks;
8)   Calculate contraction factor ζ^t by (19);
9)   Singular value contraction for similar block matrices of X^t;
10)   Aggregate the updated image blocks to get $\hat{X}^t$;
11)   Compute the error image, $E_l^t = Y - (f_g \otimes \hat{X}^t) \downarrow^s$;
12)   Compute the boundary conditions on E_l^t;
13)   Construct local biquadratic spline surface on each 3×3 grid;
14)   Fuse surface patches into global bicubic surface;
15)   Sample to get HR image E_h^t;
16)   Project E_h^t back to $\hat{X}^t$, $X^{t+1} = \hat{X}^t + f_g \otimes E_h^t$;
17)   Calculate the mean square error (mse) between the degraded image of X^{t+1} and Y;
18)   t=t+1;
19) end while

---

5 Experiments and analysis

This section verifies the effectiveness of the proposed algorithm. We first test the performance of local biquadratic spline with edge constraints (LBSEC) and adaptive optimization in transform domain (AOTD) respectively. Second, we give the setting of experimental parameters. Finally, the overall effectiveness of the proposed SISR method is verified by comparing it with different state-of-the-art methods in terms of quantitative measurement and visual effect.

5.1 LBSEC and AOTD performance

This section is divided into two parts to evaluate the effectiveness of the proposed LBSEC and AOTD respectively.

In the first part, we compare LBSEC with bicubic, CCEM [3], and EDD [32]. CCEM combines the edge information to construct the cubic fitting surface, which makes the edge of the interpolation image smoother and the visual effect is better. EDD can effectively preserve the edge structure of the image through directional filtering and data fusion based on linear minimum mean square-error estimation (LMMSE)

Table 2 PSNR (dB) comparison among different SISR interpolation algorithms on Set5 by a factor of 2

Image	Bicubic	CCEM	EDD	LBSEC
Baby	37.03	37.24	36.38	37.76
Bird	36.77	37.67	36.86	38.09
Butterfly	27.43	28.30	28.29	28.54
Head	34.84	35.00	33.85	35.29
Woman	32.13	32.88	32.46	33.18
Average	33.64	34.22	33.57	34.57

Table 3 PSNR (dB) comparison between NMF and AOTD on Set5 and Set14 ($\times 2$, $\times 3$, $\times 4$)

Image set	NMF			AOTD		
	$\times 2$	$\times 3$	$\times 4$	$\times 2$	$\times 3$	$\times 4$
Set5	35.85	32.41	30.28	**37.17**	**33.38**	**30.90**
Set14	31.85	28.95	27.29	**32.69**	**29.51**	**27.68**

The best results are shown in bold

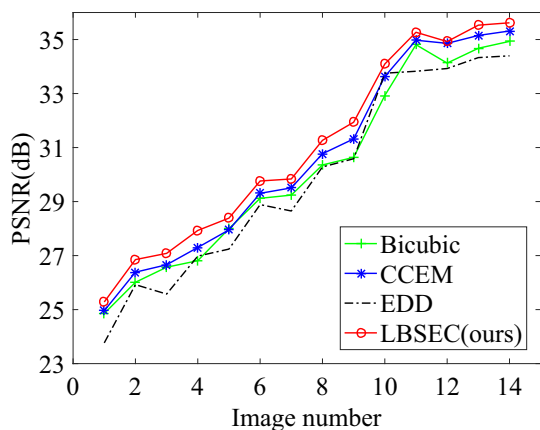**Fig. 5** PSNR (dB) comparison among different SISR interpolation algorithms on Set14 by a factor of 2

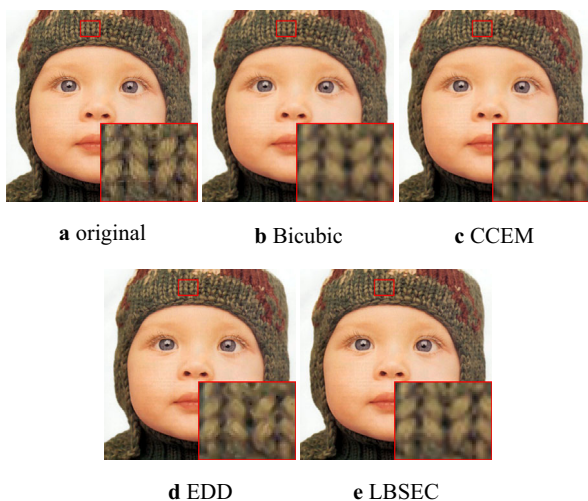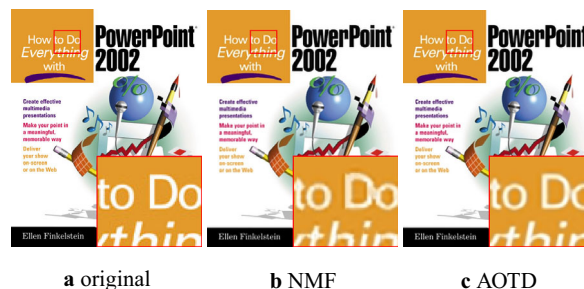**Fig. 7** Performance of NMF and AOTD on ppt3 image ($\times 3$)

Fig. 6 Performance of different SISR interpolation algorithms on Baby image ($\times 3$)

technology. Table 2 shows the PSNR comparison result on Set5. It can be seen that LBSEC achieves the highest magnification accuracy for each image. Also as shown in Fig. 5, in the PSNR comparison results on Set14, our method also surpasses other methods. Figure 6 is a visual comparison of the methods. At the texture of the woolen cap in the image, the results of bicubic and CCEM have different degrees of blurring. In comparison, the texture of LBSEC result is clearer. There are some color blocks in the result of EDD, while the result of LBSEC has more delicate texture and better visual effect. From the above, the proposed LBSEC can reduce the interpolation error effectively and improve the accuracy of the enlarged images.

Then, we compare the AOTD with non-local mean filter (NMF). NMF reconstruct the image by replacing the reference block with weighted average of the similar blocks. Both of NMF and AOTD make use of the self-similarity of the image and operate in blocks. The main difference of them is that the former operates in the spatial domain, while the latter operates in the transform domain. We replace AOTD of the proposed algorithm with NMF and compare it with the original algorithm. The comparison results in Table 3 are about Set5 [2] and Set14 [31] by different scale factors. It shows clearly that the SISR algorithm using AOTD has a improvement of magnification accuracy than the one using NMF, and achieves better performance. As can be seen from the results in Fig. 7, for the letter "D" in the figure, the edge of NMF result is blurred, and artifacts appear at the connection with the letter "o". However, the results of the method in this paper have a sharper and smooth edge, and the letters "D" and "o" are clearly separated, which is obviously better than the NMF result. Therefore, the enhancement of similarity in the transform domain can reduce the approximation error of interpolating surface and achieve sharper edges.

Table 4 PSNR (dB) comparison among different image block sizes ($3 \times 3 \sim 8 \times 8$) on lena image ($\times 2$, $\times 3$, $\times 4$)

Block size	3×3	4×4	5×5	6×6	7×7	8×8
$\times 2$	36.34	**36.78**	36.72	36.62	36.56	36.51
$\times 3$	33.49	**33.75**	33.67	33.58	33.48	33.43
$\times 4$	31.39	**31.67**	31.64	31.56	31.48	31.40

The best results are shown in bold

Table 5 PSNR (dB) comparison among different number of similar blocks ($25 \sim 50$) on lena image ($\times 2$, $\times 3$, $\times 4$)

Block number	25	30	35	40	45	50
$\times 2$	36.84	**36.85**	**36.85**	36.83	36.83	36.82
$\times 3$	33.81	**33.81**	33.80	33.78	33.77	33.77
$\times 4$	31.70	**31.70**	**31.70**	31.69	31.67	31.68

The best results are shown in bold

5.2 Experimental setting

To verify the effectiveness of the proposed algorithm, experiments for three different scale factors (2, 3 and 4) are carried out on the test images. The publicly available image sets Set5 [2], Set14 [31], BSD100 [1]and Urban100 [16] are used as the test benchmarks, which are widely used in performance testing of super- resolution methods. In the experiment, the corresponding LR images are obtained from these HR images by bicubic interpolation.

The proposed method is compared with eight different methods, which include interpolation-based methods and learning-based methods for SISR. Bicubic is a classical interpolation algorithm. And rational fractal interpolation (RFI) [35] is a method combining rational interpolation and fractal interpolation for image super-resolution. NLFIBP [34] is based on non-local feature interpolation and iterative back projection. LPRGSS [33] utilizes low-rank patch regularization and global structure sparsity for image restoration. SelfEx [16], FSRCNN [12], MMPM [17] and DLD [9] are four learning-based methods. Among them, FSRCNN and MMPM train the model by using external training data while SelfEx and DLD depend on internal data.

The parameter setting will affect the performance of the algorithm, and we choose the appropriate parameter values through experiments. With similar block matching, the larger the search window is, the more non-local information of the image will be obtained. However, it is also more likely to find blocks with small distances but different structures. We chose a compromise size 45×45 by experimental comparison. The selection of image block size also has a crucial impact on the performance of the method. Similar blocks matched with larger block size are more reliable, but the cost is that the algorithm will require more computation. According to the data in Table 4, we select the block size 4×4.

As for the number of similar blocks, if it is large, the information in the similar block matrix is more sufficient, but some image blocks which are not similar enough may be matched. From the comparison shown in Table 5, we set the number of similar blocks as 30.

5.3 Overall quantitative comparison

For quantitative evaluation, we use PSNR (dB) and SSIM indicators to evaluate quality. Table 6 shows the results of all compared methods on Set5, Set14, BSD100 and Urban100 image sets. The best results are shown in red and the secondary in blue. It can be seen from the table that the PSNR and SSIM of the proposed method are much higher than those of the interpolation-based methods with scale factors of 2, 3, and 4. Among the four learning-based comparison methods, FSRCNN achieves the best results in natural images in Set5 and Set14, while SelfEx has the most advantage in image with high self-similarity in Urban100. Compared with the learning-based SISR methods, the proposed method also achieves the best results on Set5, Set14 and Urban100 image sets with all scale factors. On BSD100 data set, the results of our method are only worse than the best. Figure 8 shows the comparison of PSNR of different methods on Urban100 image set. It can be seen that the proposed method achieves the best results on most images. It is further proved that the proposed method can improve the accuracy and quality of enlarged image.

5.4 Overall visual comparison

Figure 9, 10, 11 and 12 shows the visual effect of test images magnified by eight different SISR methods with scale factors 2, 3 and 4. The corresponding PSNR (dB) and SSIM values are below the pictures. Figure 9 shows the images with a scale factor 2. It can be seen that the edge contour of bicubic result at the butterfly wings is severely blurred, and the edge artifacts of NLFIBP and RFI results are also heavy. The results of the other four comparison methods show that the image edge is improved compared with the previous three methods, but there is also a fuzzy effect, and the proposed method achieves a sharper edge.

Figure 10 show the results of different methods using a scale factor 3. The bicubic and NLFIBP results show that the edge of parrot's beak is not clear and distorted, while the RFI and LPRGSS results show that an edge breaks into two

Table 6 PSNR (dB) (the first row) and SSIM (the second row) results of nine SISR methods on Set5, Set14, BSD100 and Urban100 image sets with scale factors 2, 3 and 4. The best results are in red and the secondary is in blue

Image set	scale	bicubic	NLFIBP	RFI	LPRGSS	SelfEx	FSRCNN	MMPM	DLD	proposed
Set5	×2	33.639	35.514	33.251	34.675	36.485	36.901	36.723	36.341	37.173
		0.929	0.945	0.928	0.937	0.953	0.954	0.955	0.953	0.956
	×3	30.384	30.717	29.434	30.303	32.620	33.117	32.644	32.914	33.375
		0.868	0.871	0.857	0.881	0.909	0.913	0.909	0.915	0.917
	×4	28.415	29.679	27.398	27.810	30.309	30.689	30.405	30.669	30.900
		0.810	0.844	0.806	0.837	0.862	0.865	0.862	0.872	0.875
Set14	×2	30.219	31.371	29.743	29.019	32.219	32.564	32.319	32.276	32.691
		0.868	0.893	0.868	0.864	0.903	0.908	0.906	0.905	0.909
	×3	27.531	27.714	26.394	25.920	29.146	29.385	29.111	29.268	29.505
		0.774	0.771	0.760	0.766	0.819	0.823	0.821	0.822	0.826
	×4	25.990	26.739	24.747	24.475	27.382	27.567	27.331	27.493	27.676
		0.702	0.739	0.694	0.704	0.752	0.753	0.752	0.754	0.760
BSD100	×2	29.548	30.451	29.025	28.718	31.173	31.465	31.335	31.489	31.485
		0.843	0.871	0.847	0.841	0.885	0.889	0.888	0.889	0.890
	×3	27.200	27.258	25.936	25.543	28.296	28.500	28.329	28.271	28.538
		0.739	0.738	0.724	0.723	0.785	0.789	0.785	0.786	0.791
	×4	25.956	26.387	24.521	24.226	26.835	26.952	26.849	27.085	27.001
		0.667	0.698	0.658	0.661	0.711	0.712	0.711	0.718	0.717
Urban100	×2	26.556	28.433	25.931	26.522	29.326	29.464	29.155	29.105	30.026
		0.837	0.888	0.830	0.859	0.902	0.902	0.899	0.898	0.912
	×3	23.989	24.334	22.803	22.980	25.957	25.834	25.510	25.929	26.107
		0.712	0.732	0.702	0.730	0.799	0.792	0.784	0.797	0.804
	×4	22.613	23.449	21.225	21.079	24.154	23.908	23.705	24.106	24.184
		0.619	0.674	0.617	0.635	0.711	0.695	0.687	0.709	0.713
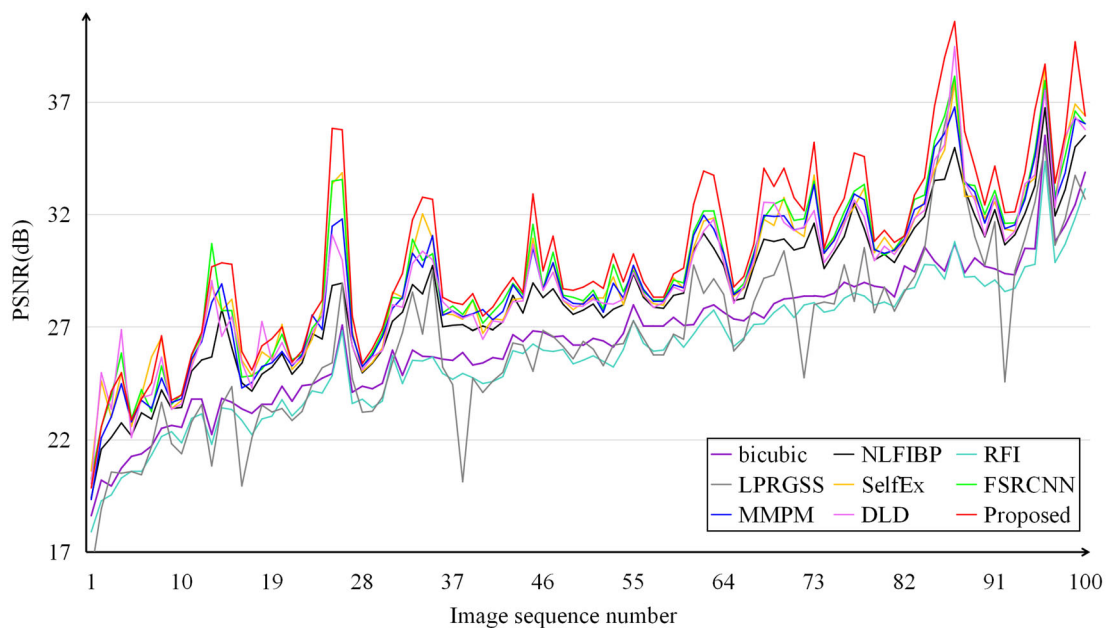
Fig. 8 PSNR (dB) comparison among nine SISR methods on Urban100 image set (×2)

(a)PSNR/SSIM (b)27.43/0.91 (c)30.31/0.95

(d)27.30/0.92 (e)30.09/0.95 (f)32.36/0.97

(g)33.30/0.97 (h)31.90/0.96 (i)**33.59/0.97**

Fig. 9 Super-resolution comparison on butterfly image (×2): **a** original, **b** bicubic, **c** NLFIBP, **d** RFI, **e** LPRGSS, **f** MMPM. **g** FSRCNN, **h** DLD, **i** proposed
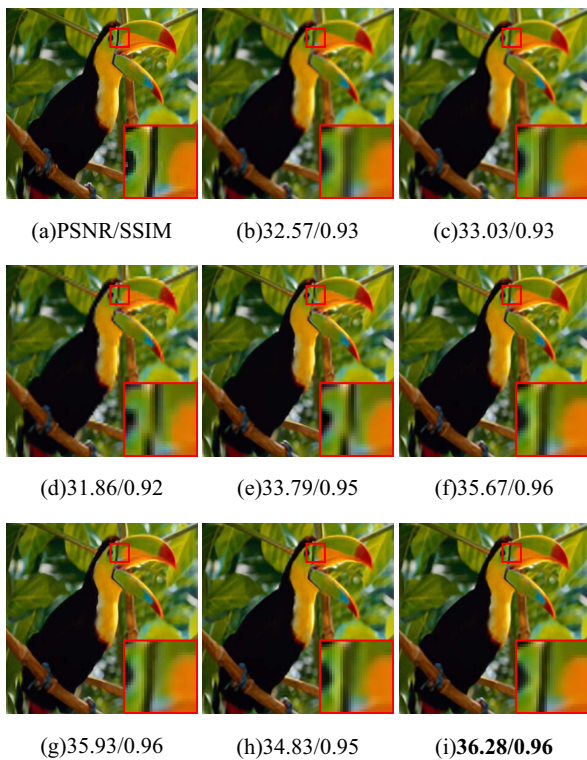

(a)PSNR/SSIM (b)29.83/0.81 (c)30.93/0.84

(d)28.85/0.80 (e)29.06/0.82 (f)31.56/0.85

(g)31.63/0.85 (h)31.48/0.85 (i)31.35/0.85

Fig. 11 Super-resolution comparison on lena image (×4): **a** original, **b** bicubic, **c** NLFIBP, **d** RFI, **e** LPRGSS, **f** MMPM, **g** FSRCNN, **h** DLD, **i** proposed

(a)PSNR/SSIM (b)32.57/0.93 (c)33.03/0.93

(d)31.86/0.92 (e)33.79/0.95 (f)35.67/0.96

(g)35.93/0.96 (h)34.83/0.95 (i)**36.28/0.96**

Fig. 10 Super-resolution comparison on bird image (×3): **a** original, **b** bicubic, **c** NLFIBP, **d** RFI, **e** LPRGSS, **f** MMPM, **g** FSRCNN, **h** DLD, **i** proposed

parts that are not connected, both of which produce unnatural edges. The result of MMPM method can see the outline of image edge, but it is not clear enough. The right side of the black edge of the DLD method is fused with the green mouth to produce a transition zone, and the resulting image is too smooth. Compared with MMPM and DLD, the result of FSRCNN has better details, but the edges are also fuzzy, and the results of the proposed method can better retain the edge details.

Figures 11 and 12 show the images with a upscaling factor 4. In Fig. 11, both the bicubic and RFI results produce severe jagged edges in the brim, resulting in missing image details. LPRGSS produces artifacts with severely distorted edges. The MMPM result is sharp on the outermost brim edge, but the brim on the upper left edge has a smooth effect, and the image does not contain fine textures. FSRCNN result also has a certain degree of fuzziness. The result of DLD has similar problems, with clear outer edges, but the inner edges are also clearly jagged and some details are distorted. Our method effectively reduces the jagged effect of the enlarged image at the texture and edge, and is closer to the original image.

As shown in Fig. 12, the boundary between collar and background in the bicubic image is not clear and the artifacts are serious. Serious mosaic effect appears in RFI images. LPRGSS results show a fault in the part below the collar edge,
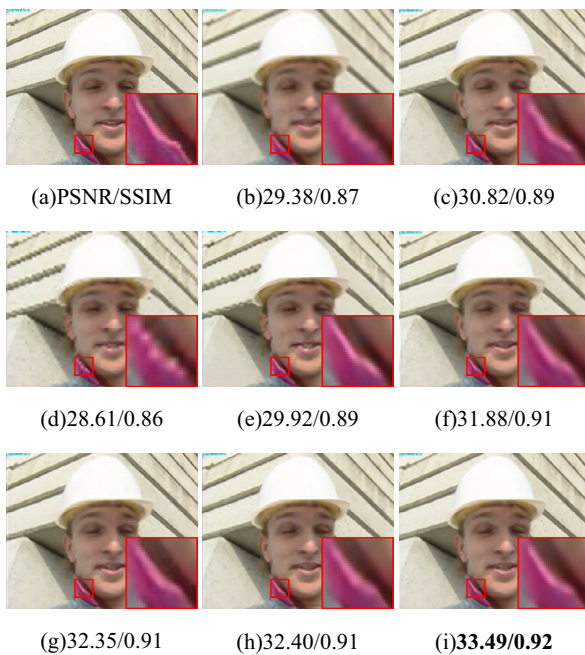
(a)PSNR/SSIM   (b)29.38/0.87   (c)30.82/0.89

(d)28.61/0.86   (e)29.92/0.89   (f)31.88/0.91

(g)32.35/0.91   (h)32.40/0.91   (i)**33.49/0.92**

Fig. 12 Super-resolution comparison on foreman image (×4): **a** original, **b** bicubic, **c** NLFIBP, **d** RFI, **e** LPRGSS, **f** MMPM, **g** FSRCNN, **h** DLD, **i** proposed

resulting in image distortion, while the magnification effect of the other four comparison methods has varying degrees of blurring effect. Compared with other methods, the method in this paper can better retain the edge, as shown in the collar part, and the algorithm can get the sharp edge without jagging.

5.5 Comparison with recent state-of-the-art methods

In this section, we compare the performance of the proposed method with that of recently proposed state-of-the-art denoising methods, including RDN [36], SAN [8], DBPN [15]. As we know, deep-learning-based techniques for SISR have been recently attracting considerable attentions due to its remarkable performance, and there are many successful schemes that have been published recently. For all the competing methods, the source codes are obtained from the original authors.

we compare the accuracy of the enlarged images by our method, RDN, SAN and DBPN in Table 7. The proposed method is lower than the other competing methods. The learning-based approach can achieve excellent results but it also has some problems. First, methods that rely on external training data will be limited in the application of image enlargement according to the type of training images. Second, some methods need a lot of time to train the model, especially the method based on CNN. Third, the learned model can only be used for the magnification of a fixed scale, and the model needs to be retrained for different scales. Our interpolation-based method is not as good as deep learning method in common image accuracy, which is the place we need to learn and improve in the next work.

5.6 Running time

LBSEC is mainly divided into three steps. The first step is to calculate the first-order partial derivatives at each pixel of the LR image. The second step is to construct the biquadratic spline surface at each pixel according to the boundary conditions. The last is to calculate the HR image pixels by weighting and averaging the pixels of four overlapping surfaces. Because the complexities of the above three steps are all O(n), the complexity of LBSEC is O(n), which is the same as Bicubic and Ecubic. In addition, AOTD uses the non-local structure information of the image, which can effectively reduce the interpolation error of the HR image. Same as NMF, its complexity is O(n). Although it is time-consuming to find similar blocks, the running time can be reduced by adjusting the step size and reducing the search frequency. In summary, the complexity of the entire algorithm is O(mn), and m is the number of iterations.

As for running time, the maximum of iterations T will control the times of iteratios, thus affecting the running time of the algorithm. In a certain range, if the value of T is larger, the magnified image will have higher accuracy, but the corresponding running time will be longer. So here we choose a compromise value $T = 100$. Meanwhile, the threshold of mean square error ε can also control the number of iterations. Through experiments, we set $\varepsilon = 0.25$. We compare the proposed method with different image super-resolution methods. As shown in Table 8, LPRGSS takes much longer

Table 7 PSNR (dB) result of different SISR methods (×3)

	RDN	SAN	DBPN	proposed
Set5	34.695	34.891	34.455	33.375
Set14	30.533	30.711	30.306	29.505
BSD100	29.310	29.365	29.274	28.538
Urban100	28.667	29.875	28.898	26.107

Table 8 PSNR (dB),SSIM and running time (min) results of different SISR methods on lena image (×3)

	NLFIBP	RFI	LPRGSS	Proposed
PSNR (dB)	31.998	31.05	31.05	33.856
SSIM	0.859	0.851	0.851	0.889
Time (min)	4.72	1.40	52.76	2.72

time than ours, but the image magnification accuracy is much lower. The block matching and SVD operations of proposed method require a large amount of computation. So ours is slower than RFI, but it has more advantages in enlarged image quality.

6 Conclusions

This paper presents a new method for SISR which is realized by two steps. First, a local biquadratic spline surface based on edge constraints is constructed on each small area of the image. And then, the enlarged image generated by biquadratic spline surface is adaptive optimized in transform domain. Biquadratic spline interpolation computes nine C^1 continuous biquadratic surface patches in the 3×3 neighboring region of each pixel, constrained by the edge information. The whole fitting surface is constructed by weighted averaging of the surface patches. The interpolation algorithm improves the accuracy of the enlarged image, and has good shape preserving ability, which reduces the sawtooth shape and mosaic effect of the image. A new adaptive optimization in transform domain is proposed to reduce the unavoidable approximation error of the interpolation surface. The optimization model combines the proposed optimization with the iterative back-projection framework to optimize the enlarged image iteratively. According to the non-local self-similarity, we compute the contraction factor and adaptively contract the singular values of similar block matrices to reduce errors. The global structure can make up for the limitation of biquadratic spline algorithm that constructing HR pixels with only local pixels. Finally, the reconstructed error image is projected back to make up the lost high-frequency information. By introducing the adaptive optimization into the traditional iterative back-projection framework, the quality of output image can be greatly improved. Compared with the state-of-the-art methods, our method achieves better results in terms of visual effects and quantitative indicators. In future work, we will consider adding more constraints to the optimization model to improve the amplification effect.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2010)

2. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012)

3. Caiming, Z., Xin, Z., Xuemei, L., Fuhua, C.: Cubic surface fitting to image with edges as constraints. In: 2013 IEEE International Conference on Image Processing, pp. 1046–1050. IEEE (2013)

4. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., vol. 1, pp. I–I. IEEE (2004)

5. Chen, M.J., Huang, C.H., Lee, W.L.: A fast edge-oriented algorithm for image interpolation. Image Vis. Comput. **23**(9), 791–798 (2005)

6. Cruz, C., Mehta, R., Katkovnik, V., Egiazarian, K.O.: Single image super-resolution based on wiener filter in similarity domain. IEEE Trans. Image Process. **27**(3), 1376–1389 (2018)

7. Dai, S., Han, M., Wu, Y., Gong, Y.: Bilateral back-projection for single image super resolution. In: 2007 IEEE International Conference on Multimedia and Expo, pp. 1039–1042. IEEE (2007)

8. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

9. Ding, N., Liu, Y.P., Fan, L.W., Zhang, C.M.: Single image super-resolution via dynamic lightweight database with local-feature based interpolation. J. Comput. Sci. Technol. **34**(3), 537–549 (2019)

10. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European conference on computer vision, pp. 184–199. Springer (2014)

11. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 295–307 (2016)

12. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: European Conference on Computer Vision, pp. 391–407. Springer (2016)

13. Dong, W., Zhang, L., Shi, G., Wu, X.: Nonlocal back-projection for adaptive image enlargement. In: 2009 16th IEEE International Conference on Image Processing (ICIP), pp. 349–352. IEEE (2009)

14. Egiazarian, K., Katkovnik, V.: Single image super-resolution via bm3d sparse coding. In: 2015 23rd European Signal Processing Conference (EUSIPCO), pp. 2849–2853. IEEE (2015)

15. Haris, M., Shakhnarovich, G., Ukita, N.: Deep back-projection networks for super-resolution. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)

16. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5197–5206 (2015)

17. Huang, Y., Li, J., Gao, X., He, L., Lu, W.: Single image super-resolution via multiple mixture prior models. IEEE Trans. Image Process. **27**(12), 5904–5917 (2018)

18. Irani, D.G.S.B.M.: Super-resolution from a single image. In: Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan, pp. 349–356 (2009)

19. Irani, M., Peleg, S.: Motion analysis for image enhancement: Resolution, occlusion, and transparency. Journal of Visual Communication and Image Representation **4**(4), 324–335 (1993)

20. Li, X., Orchard, M.T.: New edge-directed interpolation. IEEE Trans. Image Process. **10**(10), 1521–1527 (2001)

21. Meijering, E.H., Niessen, W.J., Viergever, M.A.: Piecewise polynomial kernels for image interpolation: A generalization of cubic convolution. In: Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348), vol. 3, pp. 647–651. IEEE (1999)

22. Park, S.K., Schowengerdt, R.A.: Image reconstruction by parametric cubic convolution. Computer vision, graphics, and image processing **23**(3), 258–272 (1983)
23. Parker, J.A., Kenyon, R.V., Troxel, D.E.: Comparison of interpolating methods for image resampling. IEEE Trans. Med. Imaging **2**(1), 31–39 (1983)
24. Peleg, T., Elad, M.: A statistical prediction model based on sparse representations for single image super-resolution. IEEE Trans. Image Process. **23**(6), 2569–2582 (2014)
25. Protter, M., Elad, M., Takeda, H., Milanfar, P.: Generalizing the nonlocal-means to super-resolution reconstruction. IEEE Trans. Image Process. **18**(1), 36–51 (2009)
26. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**(5500), 2323–2326 (2000)
27. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: Proceedings of the IEEE international conference on computer vision, pp. 1920–1927 (2013)
28. Timofte, R., De Smet, V., Van Gool, L.: A+: Adjusted anchored neighborhood regression for fast super-resolution. In: Asian Conference on Computer Vision, pp. 111–126. Springer (2014)
29. Wang, L., Xiang, S., Meng, G., Wu, H., Pan, C.: Edge-directed single-image super-resolution via adaptive gradient magnitude self-interpolation. IEEE Trans. Circuits Syst. Video Technol. **23**(8), 1289–1299 (2013)
30. Yang, C.Y., Yang, M.H.: Fast direct super-resolution by simple functions. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 561–568 (2013)
31. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: International Conference on Curves and Surfaces, pp. 711–730. Springer (2010)
32. Zhang, L., Wu, X.: An edge-guided image interpolation algorithm via directional filtering and data fusion. IEEE Trans. Image Process. **15**(8), 2226–2238 (2006)
33. Zhang, M., Desrosiers, C.: High-quality image restoration using low-rank patch regularization and global structure sparsity. IEEE Trans. Image Process. **28**(2), 868–879 (2019)
34. Zhang, X., Liu, Q., Li, X., Zhou, Y., Zhang, C.: Non-local feature back-projection for image super-resolution. IET Image Process. **10**(5), 398–408 (2016)
35. Zhang, Y., Fan, Q., Bao, F., Liu, Y., Zhang, C.: Single-image super-resolution based on rational fractal interpolation. IEEE Transactions on Image Processing **27**(8), 3782–3797 (2018)
36. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
37. Zheng, H., Bouzerdoum, A., Phung, S.L.: Wavelet based nonlocal-means super-resolution for video sequences. In: 2010 IEEE International Conference on Image Processing, pp. 2817–2820. IEEE (2010)

Danya Zhou received her B.S. degree in software engineering from Shandong University, Jinan, in 2017. Currently, she is a Master student in the School of Computer Science and Technology, Shandong University, Jinan. Her research interests include image super-resolution and computer vision.

Yepeng Liu is currently a Ph.D. candidate in the School of Software, Shandong University, Jinan. He received his B.S. degree from the School of Computer Science and Technology, Shandong University, Jinan, in 2014. He is engaged in image super-resolution, image denoising and image smoothing.

Xuemei Li received the Master's and Doctor's degrees from Shandong University, Jinan, China, in 2004 and 2010, respectively. She is currently an associate professor in the School of Computer Science and Technology, Shandong University, and a member of the GD&IV lab. She is engaged in research on Geometric Modeling, CAGD, Medical Image Processing and Information Visualization.

Caiming Zhang received B.S. and M.S. degrees in computer science from Shandong University, Jinan, China, in 1982 and 1984, respectively, and a Ph.D. degree in computer science from Tokyo Institute of Technology, Tokyo, Japan, in 1994. From 1998 to 1999, he was a Post-Doctoral Fellow with the University of Kentucky, Lexington, USA. He is currently a Professor with the School of Computer Science and Technology, Shandong University, and a Distinguished Professor with the School of Computer Science and Technology, Shandong University of Finance and Economics. His research interests include computer aided geometric design, computer graphics, information visualization, and medical image processing.