first see:
https://desktop.arcgis.com/en/arcmap/latest/tools/3d-analyst-toolbox/how-kriging-works.htm

Contents lists available at ScienceDirect

Optics Communications

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

Image super-resolution using windowed ordinary Kriging interpolation

Qianying Zhang *, Jitao Wu

School of Mathematics and Systems Science, Beihang University, Xueyuanlu 37, Haidian district, Beijing 100191, China

ARTICLE INFO

Article history:
Received 13 June 2014
Received in revised form
18 September 2014
Accepted 22 September 2014
Available online 1 October 2014

Keywords:
Super-resolution
Image interpolation
Ordinary Kriging
Intensity distance

ABSTRACT

This paper presents a novel interpolation approach for single image super-resolution based on ordinary Kriging interpolation, which has been widely used in geostatistics. The proposed method simultaneously considers the intensity distances and geometry of the pixel data. We employ a new intensity distance definition and local windows surrounding each unknown high-resolution pixel to implement the algorithm. The proposed approach is able to produce adaptive weights and edge preservation is achieved. Our experimental results show the efficiency of the proposed approach compared to conventional interpolation methods in terms of the peak signal-to-noise (PNSR) and visual perception.

1. Introduction

Image super-resolution (SR) is a significant image processing technique which aims to generate a high-resolution (HR) image from its low-resolution (LR) observations [1,2]. This technique is essential in many applications, such as medical image and satellite image processing. Conventional approaches employ a series of LR versions of the same scene to obtain the corresponding HR image [3]. Current image SR researches can be divided into three classes which are based on reconstruction [4,5], interpolation [6–11] and machine learning technique [12–14]. Here we mainly focus on the approaches based on interpolation.

Early image interpolation methods include the nearest-neighbor interpolation, bilinear and bicubic interpolation [6]. The greatest advantage of these methods is the low computational complexity. However, these methods tend to produce results with artifacts, such as jagged edges and blurry effects, due to degradation of the high-frequency components of the image. Some edge-guided image interpolation algorithms, which focus on preserving the edge structures, have been p [7] proposed an edge-directed inter ocal covariance, and provided a solut nce from the LR counterpart based on geometric duality. Zhang and Wu [8] proposed to interpolate the LR image in two orthogonal directions, and then adaptively fused the results to reconstruct a single HR image. Jing and Wu [9] introduced a fast image interpolation algorithm motivated by the inverse-distance weighting method [15]. Recently, sparse representation and dictionary learning have been successfully applied to image SR. Yang et al. [12] proposed a sparse representation model (SRM) to reconstruct HR image from its down-sampled LR version. By introducing a nonlocal autoregressive model into SRM, Dong and Zhang [10] proposed a sparse coding based image interpolation method, which effectively reconstructs the edges structures and reduces artifacts. Inspired by the impressive results of [10], Romano et al. [11] proposed a two-stage interpolation algorithm based on SRM and adaptive nonlocal self-similarities. Generally these methods can improve the visual quality of the interpolated images, but they are too complex and time-consuming compared to conventional linear methods.

between the interpolated value and a **model** of the ideal data

In this paper, we propose a new image interpolation method based on ordinary Kriging interpolation technique. Unknown HR pixels are estimated by weighting their four nearest diagonal pixels. We propose to calculate the interpolation weights using ordinary Kriging, which attempts to minimize the variance of the expected error. In the proposed method, we estimate the intensity distances in local windows surrounding the unknown HR pixel, and choose a simple linear semivariogram to achieve a good balance between computational efficiency and restoration performance. Our experimental results clearly demonstrate that the proposed method can generate visually attractive interpolated images of better edge preservation when compared to the results of other existing literature techniques.

The rest of this paper is organized as follows: Section 2 presents how to derive interpolation weights using ordinary Kriging; In Section 3, the proposed method based on ordinary Kriging interpolation is described in detail. Experimental results in Section 4 demonstrate the efficiency of this approach. Section 5 draws the conclusion.

* Corresponding author.
E-mail addresses: zhangqy55@126.com (Q. Zhang), wujitao555@buaa.edu.cn (J. Wu).

http://dx.doi.org/10.1016/j.optcom.2014.09.060
0030-4018/

The **semivariance** measures the dispersion of the difference between pairs of samples of a r.v.
Used to calculate the risk of a financial activity

sill: The value at which the model first flattens out
range: The distance at which the model first flattens out
nugget: The value at which the semi-variogram (almost) intercepts the y-value

2. The Kriging interpolation

The Kriging, a geostatistical interpolation technique, is an optimal and linear unbiased spatial interpolation method [16]. The characteristic of this interpolation technique is that semivariogram is introduced to measure the spatial correlation of the sample data with distance in the estimation of the interpolation coefficients. There are several Kriging methods differing in the interpolation formula [16]. In this paper, we choose ordinary Kriging interpolation, which is the most common type of Kriging.

Let random function $z(x)$ represent the value at point x in spatial region R, and $z(x+h)$ denotes the value of the same variable spacing distance h apart. Assume that $z(x)$ satisfies the conditions of intrinsic hypothesis, then the semivariogram of $z(x)$ is defined as [16]

$$r(h) = \tfrac{1}{2} Var[z(x) - z(x+h)] = \tfrac{1}{2} E[z(x) - z(x+h)]^2. \tag{1}$$

Several theoretical semivariogram models are available, such as

- The Gaussian model:

$$r(h) = \begin{cases} 0, & h = 0 \\ C_0 + C_1\{1 - \exp(-h^2/a^2)\}, & h > 0 \end{cases} \tag{2}$$

- The Spherical model:

$$r(h) = \begin{cases} 0, & h = 0 \\ C_0 + C_1\{\frac{3}{2}(h/a) - \frac{1}{2}(h/a)^3\}, & 0 < h \le a \\ C_0 + C_1, & h > a \end{cases} \tag{3}$$

- The Linear model:

$$r(h) = \begin{cases} 0, & h = 0 \\ C_0 + C_1 \dfrac{h}{a}, & h > 0 \end{cases} \tag{4}$$

where C_0 is the nugget effect, C_1 is the structured variance, $C_0 + C_1$ is the sill, and a is the variogram range ($C_0 \ge 0, C_1 \ge 0$, and $a \ge 0$) [16]. The above models are displayed in Fig. 1 using $C_0 = 0, C_1 = 1$, and $a = 1$. An experimental semivariogram can be calculated from samples and then fitted a chosen model by tuning the parameters. More details about constructing the semivariogram can be found in [17].

Given n sample observations $z(x_1), \ldots, z(x_n)$ at points $x_1, \ldots, x_n$, the ordinary Kriging estimate formula of an unknown point x_0 is [16]

$$\hat{z}(x_0) = \sum_{i=1}^{n} \lambda_i z(x_i), \tag{5}$$

where weights $\{\lambda_i\}_{i=1}^n$ are chosen such that the estimate is unbiased and the variance of the estimate error is minimum. The optimization problem can be formulated as [16]

$$\{\lambda_i\}_{i=1}^n = \arg\min_{\lambda_i} Var\left[z(x_0) - \sum_{i=1}^{n} \lambda_i z(x_i)\right] - 2\mu\left(\sum_{i=1}^{n} \lambda_i - 1\right), \tag{6}$$

where μ is a Lagrange multiplier that ensures $\sum_{i=1}^n \lambda_i = 1$. Then the optimal solutions can be obtained from [16]

$$\begin{pmatrix} r_{11} & \cdots & r_{1n} & 1 \\ \vdots & & \vdots & \vdots \\ r_{n1} & \cdots & r_{nn} & 1 \\ 1 & \cdots & 1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ \mu \end{pmatrix} = \begin{pmatrix} r_{01} \\ \vdots \\ r_{0n} \\ 1 \end{pmatrix}, \tag{7}$$

where $r_{ij} = r(x_i - x_j)$. Now the interpolation weights can be calculated and used to estimate the value of unknown point by solving the linear system of equations.

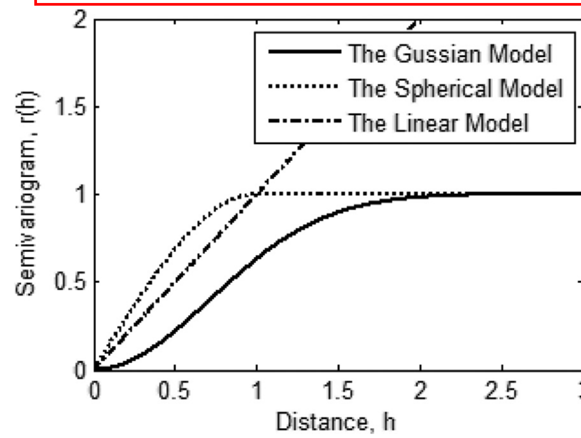

Fig. 1. Three semivariogram models, using nugget effect $C_0 = 0$, structured variance $C_1 = 1$ and variogram range $a = 1$.

3. The proposed method

Kriging for image becomes more computationally expensive as the data points to interpolate are much more than those in geostatistics. In order to reduce the computational complexity, only the four nearest diagonal pixels are included to estimate the unknown pixel. We assume that the LR image $X_{i,j}$ of size $H \times W$ is directly downsampled from the HR image $Y_{i,j}$ of size $2H \times 2W$, i.e. $Y_{2i,2j} = X_{i,j}$. As shown in [7], the interpolation problem can be solved in two steps. The first step is to interpolate the interlacing $Y_{2i+1,2j+1}$ from the lattice $Y_{2i,2j}$. Second, the other interlacing lattice $Y_{i,j}(i+j = odd)$ can be interpolated from the lattice $Y_{i,j}(i+j = even)$ in a similar method, as their geometric structures are isomorphic up to a scaling factor of $\sqrt{2}$ and a rotation factor of $\pi/4$. Therefore, we will only discuss the interpolation of $Y_{2i+1,2j+1}$. Following (5), the problem can be formulated as

$$\hat{Y}_{2i+1,2j+1} = \sum_{k=0}^{1} \sum_{l=0}^{1} \lambda_{2k+l} Y_{2(i+k),2(j+l)}. \tag{8}$$

where $Y_{2(i+k),2(j+l)}$ denote the four nearest diagonal pixels of $Y_{2i+1,2j+1}$, and λ_{2k+l} is the corresponding weights. Next, we will introduce how to calculate λ_{2k+l} using ordinary Kriging interpolation.

Note that as the fundamental element of Kriging method, the semivariogram is closely related to the distances of the sample data. Hence, it is critical to define a new distance in terms of pixel intensities. In this paper, we adopt a distance definition as follows [9]:

Definition 1 (*Intensity distance*). Let x and y be two pixels in an image, the intensity distance between them can be defined as

$$h(x,y) = \| I(x) - I(y) \|_1, \tag{9}$$

where $I(\cdot)$ denotes the pixel intensity, and $\| \cdot \|_1$ is the ℓ_1-norm.

As described in (7), to interpolate $\hat{Y}_{2i+1,2j+1}$, the first step is to calculate the corresponding intensity distances. However, since $Y_{2i+1,2j+1}$ is an unknown pixel, how to obtain the distances between $Y_{2i+1,2j+1}$ and its four nearest diagonal pixels (i.e. $\{h_{0p}\}_{p=0}^3$ shown in Fig. 2) is a problem. We propose to estimate the HR distance from its LR counterpart based on their intrinsic "geometric duality". Here, geometric duality refers to the correspondence between the HR intensity distance and the LR intensity distance that couples the pair of pixels along the same direction. It suggests that the intensity change of pixels in a local image region is consistent with their geometric trends. From the perspective of difference, the intensity distance satisfies [9]

$$h(x,y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2}\,\|\vec{n}\|_1, \tag{10}$$

where $\vec{n}$ is directional derivative along (x_1-y_1, x_2-y_2) [18]. We assume that the image function is differentiable and the gradients have small perturbations [19]. Thus, the intensity distances $\{h_{0p}\}_{p=0}^{3}$ can be estimated as

$$h_{0p}=\frac{h_p}{2},\quad p=0,1,2,3, \tag{11}$$

where $\{h_p\}_{p=0}^{3}$ denote the intensity distances along the same direction in the LR image, which are illustrated in Fig. 2.

In order to increase the accuracy of the proposed method and make it more robust to noise and fine details in the image, we propose to estimate the LR intensity distances in local windows. For example to compute h_1, we first employ two 5×5 windows W1 and W2 around the pixel $Y_{2i+1,2j+1}$. Then h_1 is estimated by averaging the intensity distances between the samples along the same direction within the windows (see Fig. 2). The other intensity distances can be derived similarly.

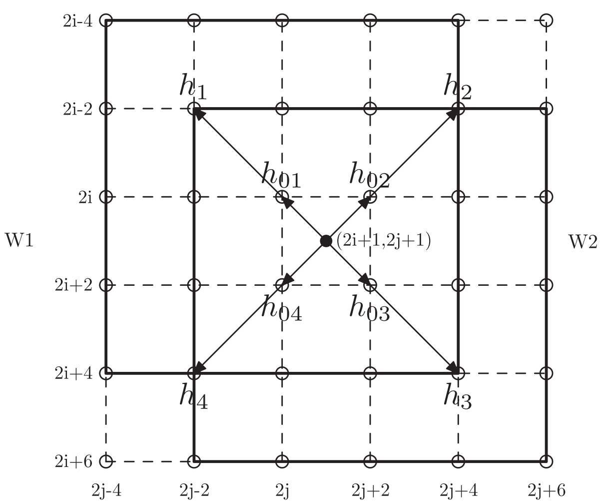

Fig. 2. Geometric duality when interpolating $Y_{2i+1,2j+1}$ from $Y_{2i,2j}$. The arrows denote intensity distances. The bold squares W1 and W2 denote the local windows employed to calculate h_1.

Thus far, we have derived the intensity distances used for (7), the next step is to calculate the semivariogram. Reviewing semivariogram models discussed in Section 2, we know that it requires much computational cost to obtain an experimental semivariogram by working with distances and corresponding values of semivariogram. On the other hand, we exploit the fact that semivariogram is a non-decreasing function. We propose to choose a simple linear semivariogram to improve the computational efficiency of our method while maintaining the visual quality. With the above considerations, we set $r(h)$ as

$$r(h)=h. \tag{12}$$

Then the weights λ_{2k+1} can be calculated according to (7) and (12). Finally, the interpolated value of $Y_{2i+1,2j+1}$ can be obtained using (8). Specifically, if the corresponding system of equations is ill-posed, we use bicubic interpolation as an alternative.

Following the approach mentioned above, we have proposed a novel image interpolation method based on windowed ordinary Kriging. We calculate the intensity distances in local windows, and obtain four diagonal weights using ordinary Kriging interpolation. The proposed method takes advantage of both geometrical structures and intensity distances of the pixel data, and attempts to minimize the variance of the expected error. It is able to produce adaptive weights according to the local image structures. It is also derived that our interpolation method has better edge preservation than other competitive and related algorithms. In the next section, the experimental results will illustrate its good performance.

4. Experimental results

We compare the proposed method with bicubic, NEDI [7], LMMSE [8], C2xinterp [20], DIDW [9], and NARM [10] to illustrate the efficiency of our proposal method. Six gray-level images (Cameraman, Zebra, Lena, Baboon, Boat and Clock) and two colorful images (Fence and Parrot) are used in the experiments (see Fig. 3). The test images are directly downsampled with a

Fig. 3. Test images from left to right and top to bottom: Cameraman, Zebra, Lena, Baboon, Boat, Clock, Fence and Parrot.

scaling factor of 2, and then interpolated back by various methods. The interpolation for a color image consists of three steps: (i) convert the image from RGB space to YCbCr space; (ii) recover the Y channel using the proposed method and interpolate Cb and Cr channels by bicubic interpolation; (iii) convert the interpolated channels back to the RGB color space. The structural similarity index (SSIM)[21] and PSNR are applied to measure the objective quality of the experimental results. All the experiments were tested using MATLAB on an Intel E8400, 3.0 GHz, 3 G RAM.

Table 1 shows the PSNR and SSIM values of different methods for the test images. It shows that our algorithm is competitive with the current state-of-the-art NARM interpolation method. The proposed method achieves higher PSNR and SSIM than the bicubic, NEDI, LMMSE, C2xinterp and DIDW. In Figs. 4–7, we show some cropped portions of the reconstructed HR images by the above algorithms. From Figs. 4–7, we can see that HR images by Bicubic have blurred edges and artifacts. The edge-guided interpolation methods NEDI, LMMSE and DIDW can generate sharp edges in most places, but fail in areas with multiple or small scale edges (e.g., Fig. 4(c), (f), Fig. 5(c), (d), Fig. 6(d) and Fig. 7(c), (f)). This is mainly because that it is difficult to accurately estimate edges' directions. By introducing contour stencils, the edge-adaptive method C2xinterp can successfully reconstruct those slight edges. However, some ringing artifacts can be clearly observed in the HR images (e.g., grasses in Fig. 4(e) and walls in Fig. 7(e)). NARM produces visually pleasant results, especially in reconstructing images with sufficient repetitive patterns (e.g., images Zebra and Parrot). It is because that NARM implicitly

Table 1
PSNR (dB) and SSIM values of different interpolation methods for the test images. For each column, the first row is PSNR, and the second row is SSIM.

Image	Technique						
	Bicubic	NEDI [7]	LMMSE [8]	C2xinterp[20]	DIDW [9]	NARM [10]	Proposed
Cameraman	23.5269	25.4183	25.6674	25.5228	25.4849	25.8783	**26.3129**
	0.8219	0.8632	0.8675	0.8633	0.8681	0.8690	**0.8694**
Zebra	22.7722	25.9359	26.1603	25.6713	25.2899	**27.6764**	26.6816
	0.7877	0.8525	0.8625	0.8428	0.8541	**0.8870**	0.8645
Lena	28.8737	31.7542	31.7948	31.7244	31.7278	**32.6481**	32.2498
	0.8328	0.8912	0.8862	0.8878	0.8875	**0.9016**	0.8932
Baboon	25.9986	27.7292	27.5704	27.3779	27.7745	27.8428	**28.0258**
	0.7408	0.8115	0.8039	0.8221	0.8033	0.8221	**0.8231**
Boat	25.3642	27.6797	27.7439	27.6082	27.7084	28.3583	**28.4188**
	0.8009	0.8569	0.8607	0.8557	0.8607	**0.8675**	0.8655
Clock	26.3245	28.0280	29.3998	28.9732	29.4331	**29.8162**	29.4465
	0.9068	0.9346	0.9380	0.9318	0.9357	**0.9401**	0.9386
Fence	22.5510	22.9362	24.5598	24.4429	24.5499	24.6776	**24.6823**
	0.6862	0.7420	0.7592	0.7578	0.7592	**0.7998**	0.7632
Parrot	24.7388	27.9427	27.8466	28.1980	27.6041	**29.8703**	28.6149
	0.8274	0.8936	0.8866	0.8953	0.8866	**0.9100**	0.8971

The best PSNR/SSIM result for each image is highlighted.

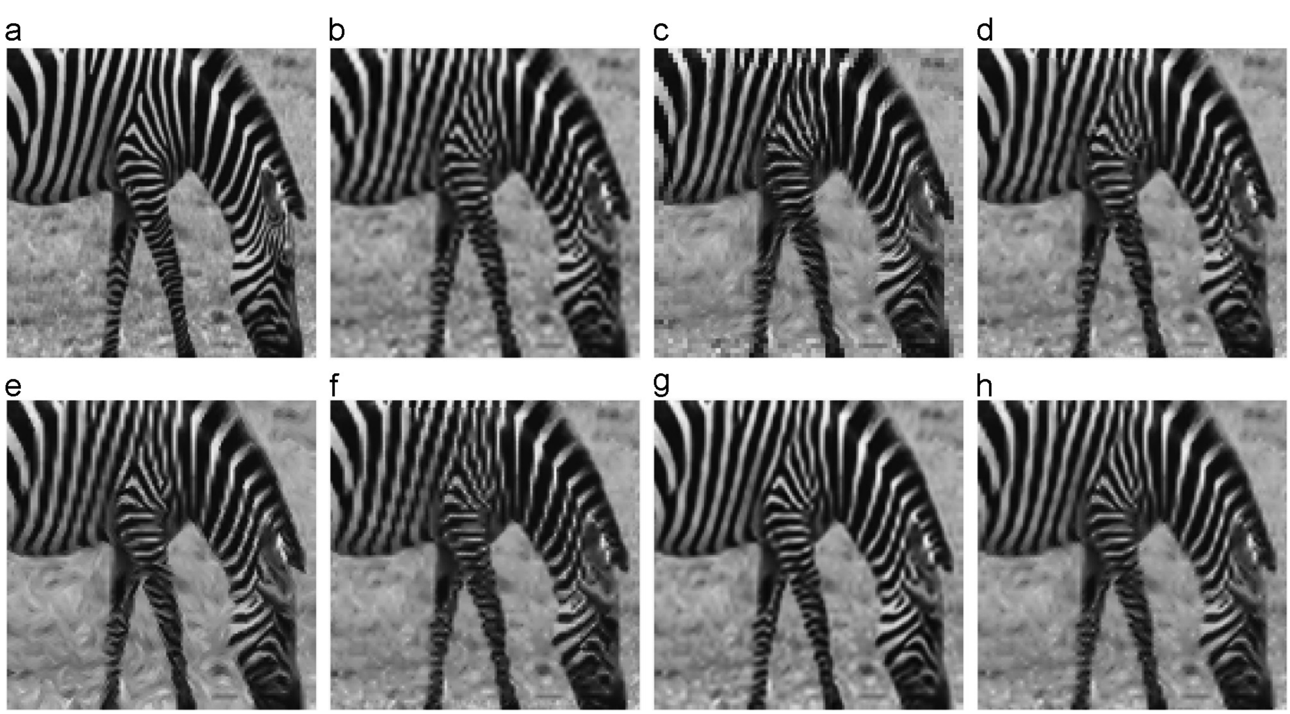

Fig. 4. Reconstructed HR images of Zebra by different interpolation methods. (a) Original image, (b) bicubic, (c) NEDI [7], (d) LMMSE [8], (e) C2xinterp [20], (f) DIDW [9], (g) NARM [10], and (h) Proposed method.

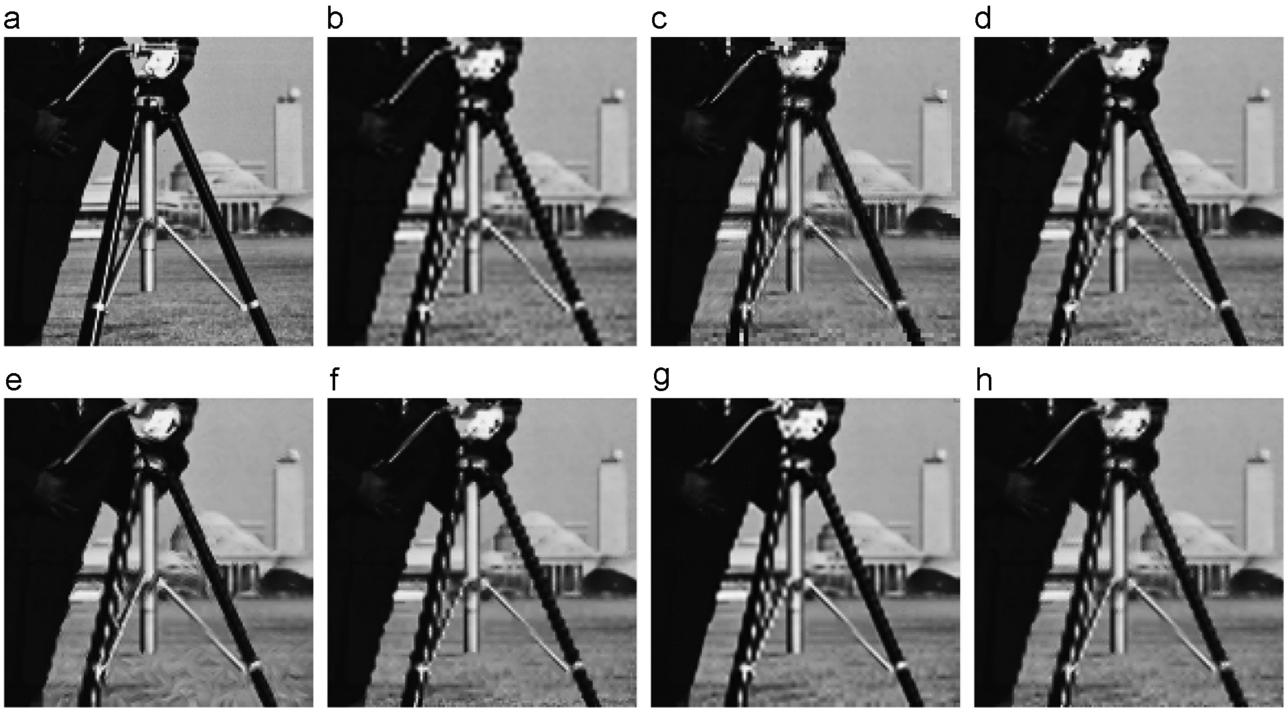

Fig. 5. Reconstructed HR images of Cameraman by different interpolation methods. (a) Original image, (b) bicubic, (c) NEDI [7], (d) LMMSE [8], (e) C2xinterp [20], (f) DIDW [9], (g) NARM [10], and (h) Proposed method.

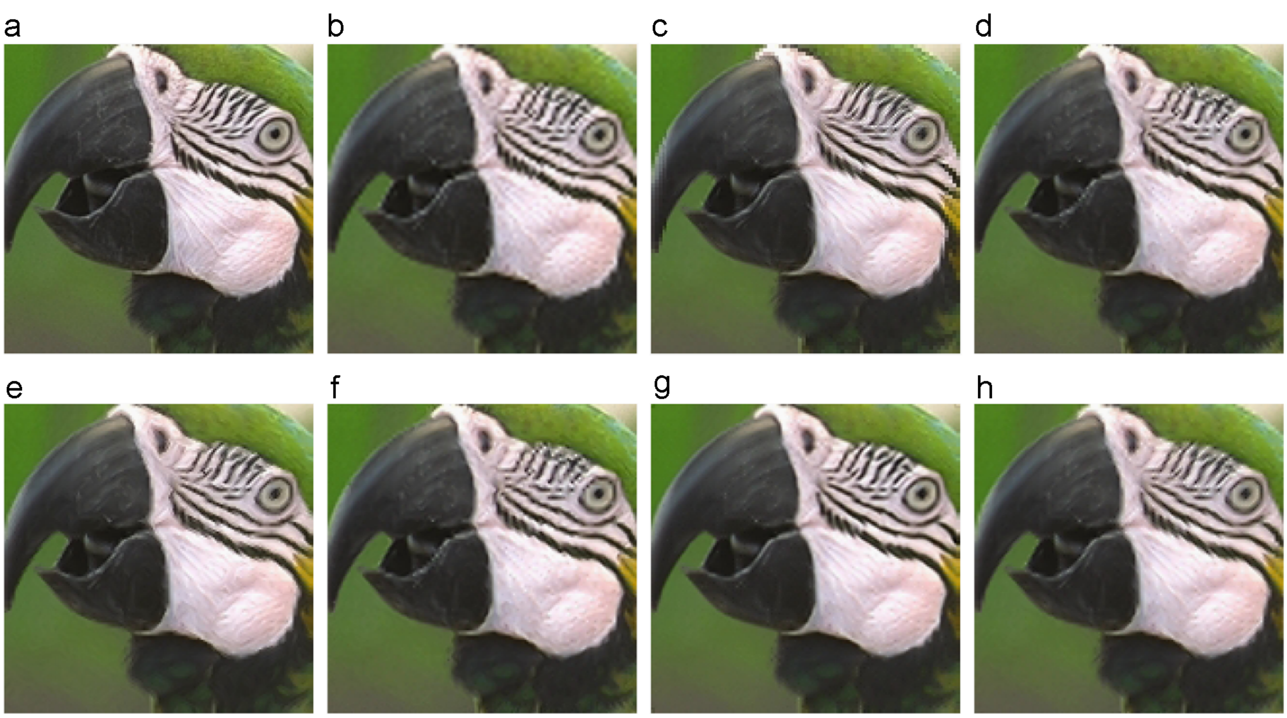

Fig. 6. Reconstructed HR images of Parrot by different interpolation methods. (a) Original image, (b) bicubic, (c) NEDI [7], (d) LMMSE [8], (e) C2xinterp [20], (f) DIDW [9], (g) NARM [10], and (h) Proposed method.

relies on the nonlocal self-similarity in natural images. According to these figures, the proposed method produces sharp edges, and has the advantage of preserving fine image details (e.g., images Cameraman and Fence). Visually, it outperforms the bicubic, NEDI, LMMSE, C2xinterp and DIDW, and is comparable to NARM. For some images, our method performs worse than NARM. But it consumes much less time compared to NARM. In our experiment, the average run time of the proposed method for interpolating the test images is 4.19 s, which of NARM is 286.15 s.

From Table 1 and Figs. 4–7, we can see that the proposed method achieves visually pleasant interpolated results with high PSNR values. Note that both DIDW and the proposed method

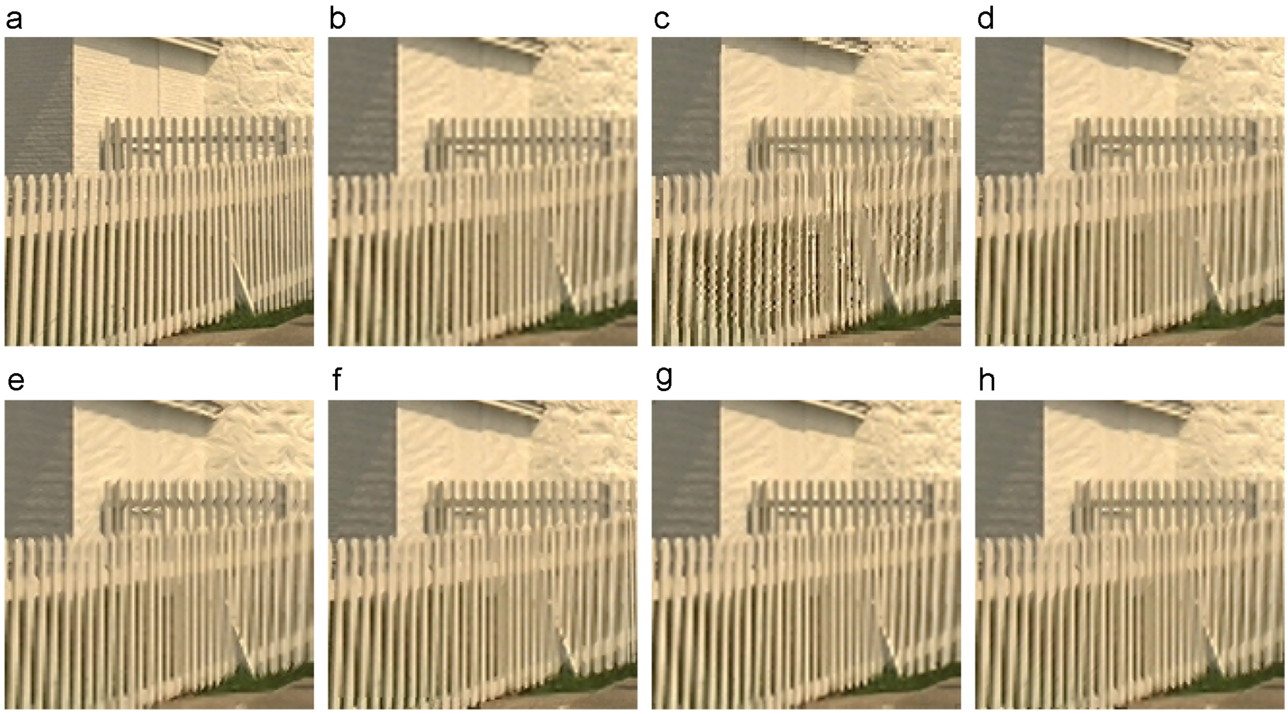

Fig. 7. Reconstructed HR images of Fence by different interpolation methods. (a) Original image, (b) bicubic, (c) NEDI [7], (d) LMMSE [8], (e) C2xinterp [20], (f) DIDW [9], (g) NARM [10], and (h) Proposed method.

utilize intensity distances to calculate the interpolation weights. However, our algorithm produces higher quality reconstruction images than DIDW. This clearly demonstrates the superiority of the ordinary Kriging interpolation.

5. Conclusion

In this paper, we proposed a new interpolation method for single image SR, which simultaneously considers the intensity distances and geometrical structure of the pixel data. The windowed ordinary Kriging interpolation technique and the intensity distances were incorporated to implement the algorithm. Compared to conventional interpolation methods, the proposed algorithm has the advantage of preserving edges. The simulations demonstrated that our method obtained qualitatively competitive interpolation images with high values of PSNR and SSIM. Note that the calculated intensity distances can be used to detect edge and textured regions of the image. In future work, we would like to apply the proposed method selectively in these edge regions, and process smooth regions using bicubic interpolation to speed up the algorithm.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant no. 61401368. The authors would like to thank authors of Refs. [7,8,10,13] for providing their programs online.

References

[1] R.Y. Tsai, T.S. Huang, Multiframe image restoration and registration, in: T.S. Huang (Ed.), Advances in Computer Vision and Image Processing, JAI Press Inc., London, 1984.
[2] S.C. Park, M.K. Park, M.G. Kang, IEEE Signal Process. Mag. 20 (2003) 21.
[3] M. Elad, A. Feuer, IEEE Trans. Image Process. 6 (1997) 1646.
[4] S. Farsiu, M.D. Robinson, M. Elad, P. Milanfar, IEEE Trans. Image Process. 13 (2004) 1327.
[5] H. Zhang, Y. Zhang, H. Li, T. Huang, IEEE Trans. Image Process. 21 (2012) 4054.
[6] H.S. Hou, H.C. Andrews, IEEE Trans. Signal Process. 26 (1978) 508.
[7] X. Li, M. Orchard, IEEE Trans. Image Process. 10 (2001) 1521.
[8] L. Zhang, X. Wu, IEEE Trans. Image Process. 15 (2006) 2226.
[9] M. Jing, J. Wu, Opt. Commun. 286 (2013) 111.
[10] W. Dong, L. Zhang, R. Lukac, G. Shi, IEEE Trans. Image Process. 22 (2013) 1382.
[11] Y. Romano, M. Protter, M. Elad, IEEE Trans. Image Process. 23 (2014) 3085.
[12] J. Yang, J. Wright, T. Huang, Y. Ma, IEEE Trans. Image Process. 19 (2010) 2861.
[13] W.T. Freeman, T.R. Jones, E.Z. Pasztor, IEEE Comput. Graph. Appl. 22 (2002) 56.
[14] H. Chang, D.-Y. Yeung, Y. Xiong, Super-resolution through neighbor embedding, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2004, pp. 275–282.
[15] P.M.MatherM. Koch, Computer Processing of Remotely-Sensed Images: An Introduction, 4th ed., Wiley-Blackwell, WestSussex, 2011.
[16] N.A.C. Cressie, Statistics for Spatial Data, John Wiley and Sons, New York, 1993.
[17] S. Chandra, M. Petrou, R. Piroddi, Pattern Recognit. Image Anal. 3523 (2005) 183.
[18] K.F. Riley, M.P. Hobson, S.J. Bence, Mathematical Methods for Physics and Engineering, Cambridge University Press, Cambridge, 2010.
[19] S.G. Mallat, A Wavelet Tour of Signal Processing, Academic Press, New York, 1998.
[20] P. Getreuer, Contour stencils for edge-adaptive image interpolation, in: SPIE Visual Communications and Image Processing, San Jose, CA, vol. 7257, 2009, p. 349.
[21] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, IEEE Trans. Image Process. 13 (2004) 600.