

DIRECTIONAL INTERPOLATION OF NOISY IMAGES

Lei ZHANG^a and Xin LI^b

^aDept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China

^bLane Dept of CSEE, West Virginia University, WV, USA

ABSTRACT

Most of the existing image interpolation schemes assume that the image is noise free. This assumption is invalid in practice because noise will be corrupted in the image acquisition process. The conventional way is to denoise the image first and then interpolate the denoised image. The denoising process, however, may smooth much the image details and introduce some artifacts, which could be amplified in the interpolation process. This paper presents a directional estimation scheme to implement denoising and interpolation simultaneously. For each noisy sample, we compute multiple directional estimates of it and then fuse them for a more accurate output. The estimation parameters computed in the denoising process can be subsequently used for interpolation. Compared with the schemes that perform denoising and interpolation in tandem, the proposed method can better reproduce the image fine structures and reduce much the interpolation artifacts.

Index Terms — Interpolation, denoising, directional estimation, data fusion

1. INTRODUCTION

Most of the existing interpolation [1-2] schemes assume that the original image is noise free. This assumption, however, is invalid in practice because noise will be introduced in the image acquisition process. One conventional way for noisy image interpolation is to denoise and interpolate the image in tandem. This strategy may not be able to yield satisfying result because some artifacts (blur, block effects, etc) will be introduced in the denoising process and they will be further amplified in the following interpolation stage.

Actually, both denoising and interpolation are an estimation problem. Denoising is to estimate the pixels from the noisy measurements and interpolation is to estimate a missing sample from its neighbors. Therefore, one can implement denoising and interpolation simultaneously under an estimation framework.

In [6], an algorithm of joint denoising and color demosaicking (which is a special case of interpolation) was proposed by using the total least square (TLS) technique. In

[7], Zhang *et al* developed a directional filtering and wavelet based joint denoising and demosaicking scheme. To the best of the authors' knowledge, few joint denoising and interpolation algorithms on gray level images have been proposed. A well designed joint denoising and interpolation method, however, can generate much less artifacts compared with the denoising first and interpolation later schemes.

Since human visual system is sensitive to edge structures, edge preservation is crucial in both denoising and interpolation. The general principle is to denoise or interpolate along the edge direction, instead of across the edge direction [3-5]. In wavelet based denoising [8], the noisy image is decomposed into many directional sub-bands for further processing. In the interpolation method [3], Li *et al* estimated the local covariance matrix, which can reflect the local directional information, from the low-resolution image, and used it to guide interpolation. Zhang *et al* [4] computed the directional estimates of the missing samples and then fused them for interpolation. Directional filtering and estimation have proved to be very effective to preserve the edges in image processing.

This paper presents an algorithm to simultaneously estimate both the noisy and missing samples in an image. We bring the problems of denoising and interpolation into the same framework of signal estimation. For each noisy sample, we compute multiple estimates of it along different directions. Those directional estimates are then adaptively fused for a more accurate output. The directional estimation coefficients learned in the denoising process can be directly used for the interpolation of missing samples.

This paper is organized as follows. Section 2 formulates the problem. Section 3 describes detailedly the joint denoising and interpolation scheme. Section 4 presents the experimental results and Section 5 concludes the paper.

2. PROBLEM FORMULATION

In this paper, we consider image interpolation as to construct a higher resolution (HR) image, whose size is $r \cdot N \times r \cdot M$, from a lower resolution (LR) image, whose size is $N \times M$, where r is the interpolation factor. As in [3-4], for the convenience of discussion and without loss of

generality, we focus on the case of $r=2$. Refer to Fig. 1, the black dots represent the LR pixels and the white dots represent the samples to be interpolated.

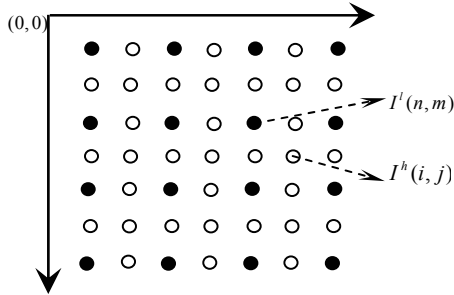


Figure 1. The low resolution image (black dot) and the missing pixels (white dot) to be interpolated.

Refer to Fig. 1, the pixels in the LR image are denoted as $I^l(n, m)$, $n=1,2,\dots,N$; $m=1,2,\dots,M$. The pixels in the HR image are denoted as $I^h(i, j)$, $i=1,2,\dots,2N$; $j=1,2,\dots,2M$. At the black dot positions, $I^h(2n-1, 2m-1) = I^l(n, m)$. In real applications, the available LR data are noise corrupted. We assume the noise is Gaussian additive:

$$I_v^l(n, m) = I^l(n, m) + v(n, m) \quad (1)$$

where noise v is zero mean with variance σ^2 . The conventional way to enlarge the noisy image is to denoise first and interpolate later. However, denoising will smooth the image and the following interpolation process may amplify many artifacts introduced in the denoising process. Therefore, new interpolation techniques for noisy images need to be developed.

We first filter the noisy LR image I_v^l using a Gaussian low pass filter to obtain the smoothed image Γ^l . Most noise is removed in Γ^l and we simply use the bicubic interpolator to interpolate Γ^l . Noise and much edge information is included in $H_v^l = I_v^l - \Gamma^l$. The final interpolation result of I_v^l is obtained by adding the interpolated images of Γ^l and H_v^l . Next we focus on how to interpolate H_v^l . For the convenience of expression, we still use symbol I_v^l in the following development.

This paper is to develop a joint denoising and interpolation scheme by modeling both two problems under the framework of signal estimation. For a noisy pixel $I_v^l(n, m)$, we can take its estimate, denoted by $\hat{I}^l(n, m)$, as the weighted average of $I_v^l(n, m)$ and its 8 nearest neighbors. For a missing HR sample $I^h(i, j)$, we can estimate it as the weighted average of its LR neighbors. Thus both denoising and interpolation are a problem of estimation and how to calculate the weights is the key issue in joint denoising and interpolation.

3. JOINT DENOISING AND INTERPOLATION

The whole joint denoising and interpolation algorithm can be divided into two stages. In the first stage, the weights to denoise the noisy LR pixels are calculated. In the second stage, the weights for interpolation will be directly obtained from the denoising weights and thus interpolation can be simultaneously accomplished.

3.1. Denoising by Directional Estimation

Let's first consider how to estimate an LR pixel $I^l(n, m)$. We use a 3×3 window here, i.e. $I^l(n, m)$ will be estimated from $I_v^l(n, m)$ and its 8 nearest neighbors. Certainly the eight weights can be computed together by using some optimal estimation techniques. However, the computation complexity may be very high. As we will see in Section 3.2, to accommodate the interpolation simultaneously, we divide the estimation of $I^l(n, m)$ into several sub-problems.

We divide the nine LR samples within the 3×3 window centered at location $I^l(n, m)$ into three groups: the four diagonal neighbors of $I_v^l(n, m)$, the four horizontal and vertical neighbors of $I_v^l(n, m)$, and the sample $I_v^l(n, m)$ itself. Each group can give an estimate of $I^l(n, m)$. The three estimates can then be fused for a better estimation.

Let's first discuss how to estimate $I^l(n, m)$ using the four diagonal neighbors. For the convenience of expression, we denote by s_0 the desired noiseless pixel $I^l(n, m)$, by s_0^v the noisy measurement $I_v^l(n, m)$ and by s_1^v, s_2^v, s_3^v , and s_4^v the four diagonal noisy neighbors of $I^l(n, m)$. The estimate of s_0 , denoted by \hat{s}_0 , is calculated as:

$$\hat{s}_0 = \sum_{i=1}^4 a_i s_i^v \quad (2)$$

Denote by $\tilde{s} = \hat{s}_0 - s_0$ the estimation error of \hat{s}_0 . The objective is then to minimize $f = E[\tilde{s}^2]$. Differentiating f with respect to a_i and letting them be zero, we obtain the optimal weights a_i as follows

$$\bar{a} = \mathbf{R}^{-1} \bar{r} \quad (3)$$

where $\bar{a} = [a_1 \ a_2 \ a_3 \ a_4]^T$, $\mathbf{R} = E[\bar{s}_v \bar{s}_v^T]$, $\bar{r} = E[s_0 \bar{s}_v]$ and $\bar{s}_v = [s_1^v \ s_2^v \ s_3^v \ s_4^v]^T$.

To calculate \bar{a} , we need to know \mathbf{R} and \bar{r} . They can be estimated adaptively by using a window centered at s_0^v , i.e. $I_v^l(n, m)$. Suppose the size of the window is $W \times W$. For each pixel within the inner window of size $(W-2) \times (W-2)$, there are four diagonal neighbors of it. Denote by \bar{s}_0^v the column vector containing all the measurements inside the

inner block and by \bar{S}_0 its unknown noiseless counterpart. Then we can form four sample vectors, denoted by \bar{S}_1^v , \bar{S}_2^v , \bar{S}_3^v and \bar{S}_4^v , containing the four sets of diagonal neighbors of \bar{S}_0^v , respectively. Let $\mathbf{S} = [\bar{S}_1^v \ \bar{S}_2^v \ \bar{S}_3^v \ \bar{S}_4^v]$. The covariance matrix \mathbf{R} can then be estimated by

$$\mathbf{R} \approx \mathbf{S}^T \mathbf{S} / (W - 2)^2 \quad (4)$$

The element of \bar{r} , $\bar{r}(i) = E[s_0 s_i^v]$, can be estimated as $\bar{r}(i) \approx \bar{S}_0^T \bar{S}_i^v / (W - 2)^2$. However, \bar{S}_0 is not available in practice. Fortunately, since the noise v is white additive and uncorrelated with I^l , we have $\bar{r}(i) = E[s_0 s_i^v] = E[s_0^v s_i^v]$. Thus \bar{r} can be estimated by

$$\bar{r} \approx (\bar{S}_0^v)^T \mathbf{S} / (W - 2)^2 \quad (5)$$

Now both \mathbf{R} and \bar{r} are estimated and then the weights α_i can be calculated using (3). The estimate \hat{s}_0 is subsequently obtained using (2). The variance of estimation error of \hat{s}_0 can be calculated as

$$e = E[(\hat{s}_0 - s_0)^2] = E[s_0^2] - \bar{r}^T \mathbf{R}^{-1} \bar{r} \quad (6)$$

where the variance of s_0 can be estimated as:

$$E[s_0^2] \approx (\bar{S}_0^v)^T \bar{S}_0^v / (W - 2)^2 - \sigma^2 \quad (7)$$

Similar to what described above, the four horizontal and vertical neighbors of s_0 will also yield an estimate of it. At last, s_0^v will give the third estimate of s_0 . The three estimates can be fused for a more accurate estimate of s_0 .

3.2. Fusion of Directional Estimates

The optimal fusion of the three estimates of s_0 may cost much computation. To reduce the fusion complexity, here we use a simple linear weighting scheme.

Denote by \hat{s}_0^i and e_i , $i = 1, 2, 3$, the three estimates and the variances of estimation error that are calculated by (2) and (6), respectively. We fuse \hat{s}_0^i as

$$\hat{s}_0^f = \sum_{i=1}^3 w_i \hat{s}_0^i \quad (8)$$

where the weights $\sum_{i=1}^3 w_i = 1$. Intuitively, if the variance e_i is high, then the corresponding weight w_i on \hat{s}_0^i should be low. Therefore, we set the weights as

$$w_i = \xi_i / \sum_{i=1}^3 \xi_i \quad (9)$$

where $\xi_i = 1/e_i$.

3.3. Simultaneous Interpolation

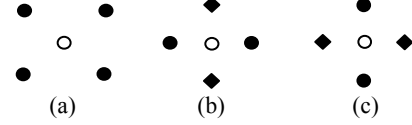


Figure 2. Simultaneously interpolation of the missing HR pixels. (a) Diagonal, (b) horizontal and (c) vertical pixels.

As shown in Fig. 2, the missing HR pixels to be interpolated can be divided into three groups: the diagonal pixels, the horizontal pixels and the vertical pixels. For a missing diagonal HR pixel, denoted by I_d^h , it has four diagonal denoised LR neighbors, denoted by I_k^l , $k=1,2,3,4$, from which the missing HR pixel I_d^h can be estimated by using a linear combination of them:

$$\hat{I}_d^h = \sum_{k=1}^4 \alpha_k I_k^l \quad (10)$$

For each I_k^l , we have a set of diagonal denoising weights \bar{a}_k , which are computed to offer an optimal diagonal estimate of I_k^l . Those weights can be used to determine $\bar{\alpha} = [\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4]^T$ because $\bar{\alpha}$ also aims to offer a good diagonal estimate of I_d^h . Therefore, we let $\bar{\alpha}$

be the average of the four sets of weights: $\bar{\alpha} = \sum_{k=1}^4 \bar{a}_k$.

After interpolating the diagonal HR pixels, the horizontal/vertical HR pixels can then be interpolated. Referring to Figs. 2 (b) and (c), a horizontal/vertical HR pixel has two denoised LR neighbors and two interpolated diagonal HR neighbors (diamond dots). Similar to that of the diagonal HR pixels, the horizontal/vertical HR pixels are estimated by weighting the four neighbors. The weights are obtained by averaging the two sets of denoising weights associated with the two horizontal/vertical LR neighbors.

4. EXPERIMENTAL RESULTS

Experiments were performed to verify the proposed joint denoising and interpolation algorithm. For comparison, we first denoise the LR image using the sophisticated wavelet based scheme [8] and then interpolate them using the state-of-the-art interpolation schemes [3-4]. The standard test images *Barbara* and *Peppers* (512×512) were employed in the experiments. The original images were downsampled to 256×256 and added Gaussian white noise, from which the clean and zoomed images are to be interpolated.

Table 1. PSNR results (dB) of the denoised and interpolated images by different schemes.

Methods	[8]+[3]	[8]+[4]	Proposed
Barbara	22.02	23.70	22.89
Peppers	30.26	30.23	29.48



Figure 3. (a) Original image *Barbara*; denoised and zoomed images by (b) [8]+[3]; (c) [8]+[4]; and (d) the proposed scheme.

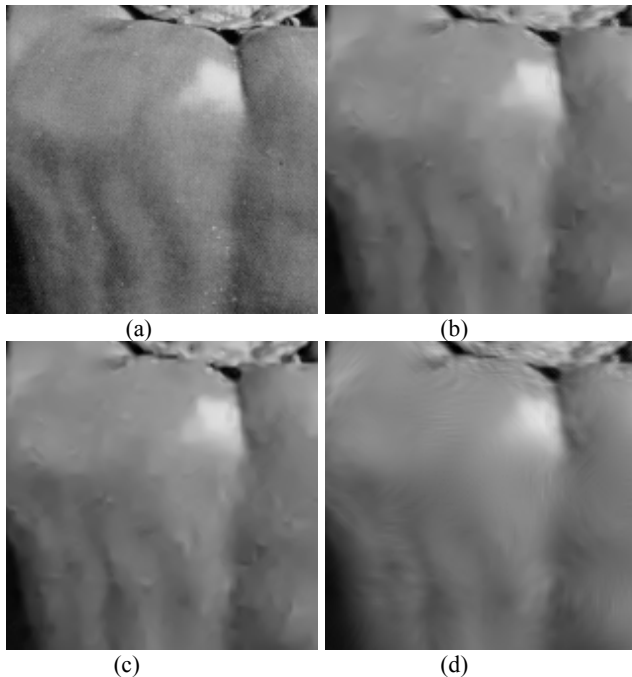


Figure 4. (a) Original image *Peppers*; denoised and zoomed images by (b) [8]+[3]; (c) [8]+[4]; and (d) the proposed scheme.

Fig. 3 shows cropped output images by different methods on the noisy *Barbara* image (PSNR=24.56dB) and Fig. 4 shows the cropped interpolation results on the noisy *Peppers* image (PSNR=24.66dB). Table 1 lists the PSNR

results by the three schemes. Although the PSNR results by the proposed method are not higher than the other two schemes, we can see that the recovered images are much more visually pleasing. Many block effects and other noise caused artifacts appeared in other schemes are removed by the proposed joint denoising and interpolation scheme.

5. CONCLUSION

This paper presented a novel interpolation scheme for noisy images. Unlike the conventional schemes that perform denoising first and interpolation later, the proposed method treats both denoising and interpolation as an estimation problem and implements them under a unified framework. In addition, directional estimation and fusion were used for edge preservation. The experimental results validate that the proposed scheme can better suppress the many noise-caused artifacts in the enlarged image while preserving the image fine structures.

6. REFERENCES

- [1] T. M. Lehmann, Claudia Gönner and K. Spitzer, "Survey: interpolation methods in medical image processing," *IEEE Trans. Medical Imaging*, vol. 18, pp. 1049-1075, Nov. 1999.
- [2] E. Meijering, "A chronology of interpolation: from ancient astronomy to modern signal and image processing," *Proc. Of The IEEE*, vol. 90, pp. 319-342, Mar. 2002.
- [3] Xin Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Processing*, vol. 10, pp. 1521-1527, Oct. 2001.
- [4] Lei Zhang and X. Wu, "An edge guided image interpolation algorithm via directional filtering and data fusion," *IEEE Trans. on Image Processing*, vol. 15, pp. 2226-2238, Aug. 2006.
- [5] D. D. Muresan, "Fast Edge Directed Polynomial Interpolation," *Proceedings of International Conference on Image Processing*, vol. 2, pp. 990-993, 2005.
- [6] K. Hiraoka and T. W. Parks, "Joint demosaicking and denoising," *IEEE Trans. on Image Processing*, vol. 15, no. 8, pp. 2146 – 2157, Aug. 2006.
- [7] Lei Zhang, X. Wu and D. Zhang, "Color reproduction from noisy CFA data of single sensor digital cameras," *IEEE Trans. Image Processing*, vol. 16, no. 9, pp. 2184-2197, Sept. 2007.
- [8] J. Portilla, V. Strela, M. J. Wainwright and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. on Image Processing*, vol. 12, pp. 1338 – 1351, Nov. 2003.