Optical Engineering

SPIEDigitalLibrary.org/oe

Multidirectional edge-directed interpolation with region division for natural images

Yujin Yun
Jonghyun Bae
Jaeseok Kim

SPIE

Multidirectional edge-directed interpolation with region division for natural images

Yujin Yun, Jonghyun Bae, and Jaeseok Kim
Yonsei University, Department of Electrical and Electronic
Engineering, 50 Yonsei-ro, Seodaemun-gu, Seoul,
Republic of Korea
E-mail: yujin.yun@yonsei.ac.kr

Abstract. A multidirectional edge-directed interpolation algorithm that features a region division method is proposed. In the proposed method, an interpolation pixel is newly modeled as a weighted sum of 12 neighboring pixels representing 12 different directions. Each weight is estimated by Wiener filter theory using geometric duality. The proposed method for dividing the interpolation region reduces the heavy computational complexity of the proposed model. Analyzing edge continuities, the model divides an image into three regions, and only strong edge regions are interpolated. Simulation results show that several directional edges are restored clearly in a subjective test, with fair performance in an objective test. [DOI: 10.1117/1.OE.51.4.040503]

Subject terms: image interpolation; Wiener filter theory; canny edge detector; demosaicing.

Paper 111346L received Nov. 9, 2011; revised manuscript received Feb. 1, 2012; accepted for publication Feb. 13, 2012; published online Apr. 11, 2012.

1 Introduction

In digital image processing, image resolution enhancement is in great demand. Medical images, surveillance images, and old photographs require an accurate image enlargement scheme for reconstructing missing information. Interpolation is such a process of estimating the missing pixels of high-resolution images from low-resolution images.

Linear interpolation is simple and fast, but not suitable for high-quality image restoration. For sophisticated applications, nonlinear interpolation methods such as edge-directed interpolation (EDI) are preferable. The human vision system is particularly responsive to a sudden change of pixel intensities. Hence, the quality of the entire image can be improved by restoring edges with less degradation. New EDI (NEDI)[1] is a representative EDI that uses Wiener filter theory. NEDI restores high-quality images; however, it suffers from blocking artifacts because of the deficiency of edge directionality information and the geometric duality assumption.[1] While various modifications of NEDI in Refs. 2 and 3 show some improvements, restored images are still degraded easily.

In this letter, we present a multidirectional EDI method. A new interpolation algorithm uses 12 directional neighboring pixels to restore various directional edges. To reduce high computational complexity, a method for dividing the interpolation region is also proposed.

0091-3286/2012/$25.00 © 2012 SPIE

2 Proposed Algorithm

2.1 *Multidirectional Interpolation Model*

Consider an interpolation of a low-resolution image X with a size of $H \times W$ into a high-resolution image Y with a size of $2H \times 2W$, that is, $Y_{2i,2j} = X_{i,j}$. For unknown pixel $Y_{2i+1,2j+1}$, a new interpolation model is proposed:

$$Y_{2i+1,2j+1} = \sum_{k=0}^{3}\sum_{l=0}^{3} \alpha_{4k+l} Y_{2(i+k-1),2(j+l-1)}, \tag{1}$$

where α indicates an interpolation coefficient and $\alpha_{0,3,12,15} = 0$. Figure 1(a) shows the visual concept of Eq. (1). Twelve neighboring pixels participate in the interpolation of $Y_{2i+1,2j+1}$. Unlike NEDI, the proposed model uses a sufficiently large number of directional pixels for precise estimation. Since this new interpolation model is expanded to all directions for an estimation window at Sec. 2.3, the geometric duality mismatch problem of NEDI is also solved.

With the proposed interpolation model, the computational complexity of the algorithm increases dramatically because of the large dimensional matrix multiplications in the coefficient estimation process. Also, using additional neighboring pixels may duplicate details in the short- and random-edge regions. These difficulties can be resolved by the interpolation region division method proposed below.

2.2 *Interpolation Region Division Method*

Depending on image features, a natural image can be divided into even (or non-edge) regions, short-edge regions, and long-edge regions. Following are details of the four steps.

1. *Divide the even regions.* Even regions and edge regions are divided first. The variances of 4×4 neighborhood pixels are inspected. If the variance does not exceed a certain threshold, TH_e, it is considered an even, or non-edge, region.
2. *Obtain an edge map.* An edge map of a target image can be obtained with the canny edge detector, which is one of the most prominent edge detectors. An edge map is interpolated by the nearest interpolation for a high-resolution edge map.
3. *Analyze edge features.* From the beginning of an edge map, all edge pixels are examined. The detailed procedure is visualized in Fig. 1(b). First, a starting edge pixel is selected. If there are edge pixels among eight neighbors, those edge pixels are considered to be connected and are marked with a certain number. Another eight neighbors of the connected edge pixels with the starting pixel are inspected for more connectivity. If there are no more edge pixels around its neighbors, another starting edge pixel is selected for another connected edges. After the entire image has been analyzed, a connected edge map is obtained. Connected edges are marked with the same number starting from 1, and non-edge pixels remain as 0.
4. *Divide a connected edge map into two regions.* The number of connected edges is calculated in the $M \times M$ coefficient estimation window described in Sec. 2.3. The connected edge number threshold $\mathrm{TH}_{\mathrm{edge}}$ is set from 0 to 5 depending on the image features. If the

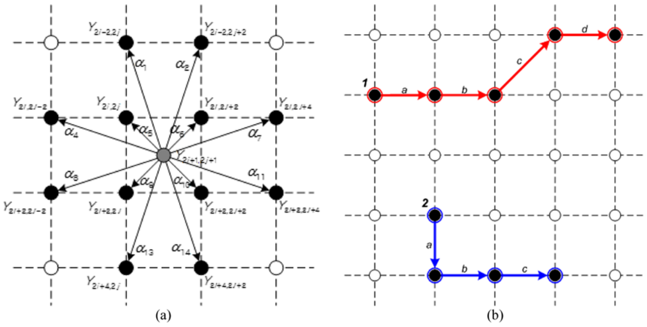

Fig. 1 (a) Interpolation model for $Y_{2i+1,2j+1}$ (Ref. 4). (b) Step 3 of interpolation region division method. Black pixels are edge pixels and white pixels are non-edge pixels. The upper red pixels are marked with 1 if there are other edge pixels around the eight neighborhoods. Starting from the leftmost edge pixel, red pixels are marked from *a* to *d* in sequence. The lower blue pixels are marked with 2 from *a* to *c* in the same manner.

number of connected edges is more than $\mathrm{TH}_{\mathrm{edge}}$, these regions are considered short-edge regions. Other edge regions are considered long-edge regions.

For even (non-edge) regions, linear interpolation is sufficient since the pixels have similar intensities. For short-edge regions, NEDI is appropriate since the edges are short enough to ignore precise directions. Long edges are the major edges of an image. If the pixels are interpolated with slightly different directions, block artifacts are easily noted. Accordingly, the proposed method is appropriate for long-edge regions. By interpolating an image with three methods, computational complexity can be greatly reduced over the use of the proposed model only.

In the proposed method, interpolation coefficients are estimated by Wiener filter theory, which is described in the following section.

2.3 *Interpolation Coefficient Estimation*

Interpolation coefficient vector $\vec{\alpha}$ in Eq. (1) is estimated by Wiener filter theory.[1] For the optimal minimum mean-square error (MMSE) condition, $\vec{\alpha}$ can be found as

$$\vec{\alpha} = R^{-1}\vec{r}, \tag{2}$$

where R and $\vec{r}$ indicate the auto and cross covariance matrix at the $M \times M$ local window. Since $Y_{2i+1,2j+1}$ is not available, $\vec{r}$ cannot be obtained directly from a high-resolution image. Statistically, a low-resolution image has geometric duality with a high-resolution image in a small local block. Accordingly, R and $\vec{r}$ can be calculated from a low-resolution image.

With the classical covariance method, Eq. (2) can be written as

$$\vec{\alpha} = (C^T C)^{-1}(C^T \vec{y}), \tag{3}$$

where $\vec{y}$ is a $M^2 \times 1$ center pixel vector, and C is a $M^2 \times 12$ matrix of 12 neighbors for $\vec{y}$. $Y_{2i+1,2j+1}$ can be calculated from Eq. (1) by substituting $\vec{\alpha}$ with Eq. (3).[1] $Y_{2i,2j+1}$ and $Y_{2i+1,2j}$ can be calculated in the same manner except that an interpolation window is slanted 45 deg.

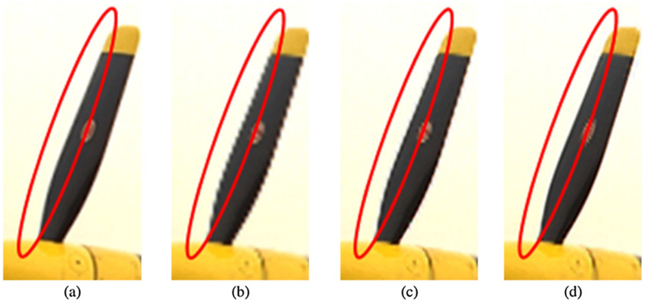

Fig. 2 Zoomed-in portions of the (a) original kodim20 image and the interpolated images from (b) bilinear, (c) NEDI, and (d) proposed method.

2.4 *Demosaicing of Color Filter Array Images*

The proposed interpolation can be applied to demosaicing problems. Because of the cost and the size of suitable digital cameras, an image obtained from a charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) sensor is sampled by color filter array (CFA). Reconstructing a full-resolution color image from CFA samples is called the demosaicing problem.

From Bayer CFA samples, green pixels are initially interpolated by the proposed method. Since red (R), green (G), and blue (B) planes are highly correlated, the interpolation process for R and B uses their color difference planes to avoid color misregistration problems.[5] Color difference planes D_R and D_B,

$$D_R = G - R \qquad \text{and} \qquad D_B = G - B, \tag{4}$$

are interpolated with the proposed algorithm, and R and B pixels can be reconstructed by Eq. (5).

$$R = G + D_R \qquad \text{and} \qquad B = G + D_B \tag{5}$$

3 Simulation Results

The proposed algorithm has been implemented with MATLAB 7.1. TH_e is set to be 8, and $\mathrm{TH}_{\mathrm{edge}}$ is set to be 3, the default. Twenty-four color images from a Kodak PhotoCD are used for the tests. Each image is downsampled directly from an original image and interpolated with three methods: bilinear, NEDI, and the proposed. Since other modifications of NEDI can be applied to the proposed method in

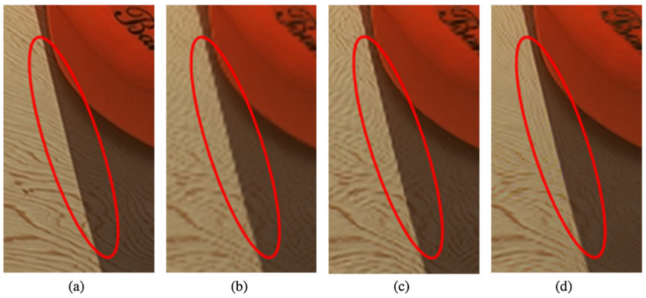

Fig. 3 Zoomed-in portions of the (a) original kodim03 image and the interpolated images from (b) bilinear, (c) NEDI, and (d) proposed method.

Table 1 (a) PSNR(dB) values of interpolated images. (b) SSIM values of interpolated images.

Image	Bilinear	NEDI	Proposed
(a)			
kodim03	30.15	31.07	32.19
kodim06	24.57	25.38	27.45
kodim07	28.56	30.76	31.27
kodim09	28.29	29.95	30.76
kodim10	28.33	30.13	31.69
kodim11	26.15	27.34	27.86
kodim20	27.23	28.03	30.73
kodim21	25.48	26.70	27.93
Average	27.76	28.99	30.75
(b)			
kodim03	0.89	0.92	0.91
kodim06	0.72	0.79	0.79
kodim07	0.90	0.94	0.94
kodim09	0.86	0.90	0.90
kodim10	0.85	0.90	0.90
kodim11	0.77	0.82	0.82
kodim20	0.86	0.90	0.90
kodim21	0.80	0.85	0.85
Average	0.86	0.90	0.90

the same way, only NEDI is selected for representative comparison.

Zoomed-in portions of interpolated images are presented in Figs. 2 and 3. Interpolated images from bilinear (b) and NEDI (c) show blocking artifacts along edges. On the other hand, the interpolated image from the proposed method (d) shows clear edges just like the original figure. In Fig. 3, the fine wood grain is restored in the right direction in (d), but not in (b) and (c). In Table 1, the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) index[6] are compared. SSIM quantifies the degradation of the structural information in an image. Our algorithm also shows competitive performance in the objective tests.

The proposed algorithm has disadvantages in computational time. A user can adjust threshold values for reducing computational complexity at the expense of performance. However, process time is not a critical issue when restored image quality is a major concern.

4 Conclusion

This letter presents a new idea for EDI. Twelve neighboring pixels are used for interpolation in order to reflect 12 directionalities. Also proposed is an interpolation region division method that resolves concerns related to the computational complexity and performance. Depending on the edge continuity, an image can be divided into three regions: even (non-edge) region, short-edge region, and long-edge region. Only long-edge regions are interpolated by the proposed model. Simulation results show that our method restores multidirectional edges clearly with fair performance in objective tests.

Acknowledgments

This work was supported by the Mid-career Researcher Program through an NRF grant funded by the MEST (2011-0027515), and partially supported by the Ministry of Knowledge Economy (MKE, Korea) and IDEC/IDEC Platform Center (IPC) at Hanyang University.

References

1. X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.* **10**(10), 1521–1527 (2001).
2. W.-S. Tam, C.-W. Kok, and W.-C. Siu, "Modified edge-directed interpolation for images," *J. Electron. Imag.* **19**, 013011 (2010).
3. N. Asuni and A. Giachetti, "Accuracy improvements and artifacts removal in edge based image interpolation," in *Proc. 3rd Int. Conf. Computer Vision Theory and Applications (VISAPP'08)*, Funchal, Madeira, Portugal (2008).
4. Y. Yun, J. Bae, and J. Kim, "Adaptive multidirectional edge directed interpolation for selected edge regions," *TENCON2011-2011 IEEE Region 10 Conf.*, Bali, pp. 385–388 (Nov 2011).
5. B. K. Gunturk et al., "Demosaicking: color filter array interpolation," *IEEE Signal Proc. Mag.* **22**(1), 44–54 (2005).
6. Z. Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**(4), 600–612 (2004).