One-Dimensional Grey Polynomial Interpolators for Image Enlargement

Cheng-Hsiung Hsieh, Ren-Hsien Huang and Ting-Yu Feng
Department of Computer Science and Information Engineering
Chaoyang University of Technology
Wufong, Taiwan 41349, ROC
E-mail: chhsieh@cyut.edu.tw

Abstract

In this paper, one-dimensional (1-D) grey polynomial interpolators (GPIs) for image enlargement are proposed. Note that (i) the randomness inherent in image data affects the performance of polynomial interpolators (PIs) in image enlargement, and (ii) that the preprocessing scheme in grey systems, the first-order accumulated generating operation (1-AGO), is able to reduce randomness in data. In this paper, 1-D grey polynomial interpolators are developed for image enlargement where 1-AGO is used to preprocess image data. To improve the performance of GPIs further, an α *filter is applied to smoothen the interpolated pixels. Examples are given to verify the proposed 1-D GPIs. Simulation results indicate that the 1-D GPIs have better performance than the 1-D PIs in terms of PSNR. Besides, the 1-D GPIs are compared with well-known 2-D polynomial interpolators in image enlargement, i.e., the nearest-neighbor interpolator, the bilinear interpolator, and the bicubic interpolator. Interesting enough, the 1-D 2-order GPIs performs little better than the 2-D bicubic interpolator in PSNR on average, with equally well visual quality.*

Index Terms—*Polynomial Interpolation, Grey First Order Accumulated Generating Operation, Image Enlargement*

1. Introduction

Image enlargement is a technique to generate a higher resolution image from a lower resolution one. Some applications of image enlargement are videoconference [1], medical imaging [2], and digital photographs [3-4]. From the viewpoint of digital signal processing, image enlargement can be considered as a re-sampling problem [5]. To resample a digital signal, an interpolator is required. Popular interpolators for image enlargement are the nearest neighbor interpolator [6-10], the bilinear interpolator [6, 11], and the bicubic interpolator [12-13]. These interpolators are 2-D polynomials of zero-, first-, and third-order, respectively. In general, a polynomial interpolator with higher order has better performance. Recently, many researchers have put their efforts on edges of interpolated images. Several edge-oriented algorithms for image interpolation have been proposed [14-21]. However, to our knowledge up to present no studies have been done on the effect of a preprocessing scheme on the performance of polynomial interpolators. In this paper, a preprocessing scheme in grey systems [22] is employed to preprocess image data for 1-D polynomial interpolators. The effect of preprocessing on the performance of these interpolators is investigated in the paper. This paper is organized as follows. In Section 2, 1-D polynomial interpolators are briefly reviewed. Next, the proposed 1-D grey polynomial interpolators are described in Section 3. Then examples are provided to verify the proposed interpolators and to compare with popular 2-D polynomial interpolators in Section 4 where discussions are given as well. Finally, conclusion is made in Section 5.

2. Review of 1-D Polynomial Interpolators

In this section, 1-D polynomial interpolators (PIs) are briefly reviewed. For details, one may consult [22]. Given $x(k)$, the implementation steps for PIs are described as follows.

Step 1. Assume $x(k)$ is a L-order polynomial of the form

$$x(k) = c_L k^L + c_{L-1} k^{L-1} + \cdots + c_1 k + c_0 \qquad (1)$$

Step 2. Substitute $1 \le k \le L+1$ into (1) whose result is

$$\boldsymbol{x} = \boldsymbol{V}\boldsymbol{c} \qquad (2)$$

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE

where elements of $\boldsymbol{x}$, $\boldsymbol{c}$, and $\boldsymbol{V}$ are $x(k)$, c_k, and $v_{kj} = k^{L-j}$ for $0 \le j \le L$, $1 \le k \le L+1$, respectively.

Step 3. Find the interpolated value, $\hat{x}(k+1/M)$, as

$$\hat{x}(k+1/M) = c_L(k+1/M)^L + \cdots + c_1(k+1/M) + c_0 \quad (3)$$

where c_k are the coefficients found in (2).

Step 4. Obtain the final interpolated value as

$$\hat{x}(k+1/M) = \hat{x}(k+1/M) \times M \quad (4)$$

where M denotes a magnification factor and is assumed an integer without the loss of generality.

It should be pointed out that randomness in data generally affects the performance of PIs. Consequently, the performance may be improved by reducing the randomness in data. Based on the idea, grey polynomial interpolators are proposed in the following section.

3. The Proposed Grey Polynomial Interpolators

In this section, grey polynomial interpolators (GPIs), modified from the 1-D PIs, are introduced. Since the 1-AGO is able to reduce randomness in data, it is applied to preprocess data in the GPIs. The 1-AGO preprocessed data is then put into the PIs as described in Section II. Next, interpolated values are found through the inverse of 1-AGO, i.e., the first-order inverse accumulated generating operation (1-IAGO). Finally, an α filter is applied to interpolated values to complete the interpolation process. Assume a color image $\boldsymbol{O}$ has the YC_bC_r format and is down sampled by a factor $M \times M$. The down sampled image is denoted as $\boldsymbol{O}_M$. Since YC_bC_r components are processed separately in the GPIs, thus only Y component is considered in the following. For C_bC_r components, the steps can be applied equally well. The implementation steps for the GPIs are given as follows.

Step 1. On a row-by-row base, the Y component in $\boldsymbol{O}_M$ is segmented into (L+1)-point subsets denoted as $\{x(k),\ 1 \le k \le L+1\}$.

Step 2. Preprocess $x(k)$ by the 1-AGO as

$$x^{(1)}(k) = \sum_{i=1}^{k} x(i) \quad (5)$$

for $1 \le k \le L+1$.

Step 3. With $x^{(1)}(k)$, interpolated pixels $\hat{x}^{(1)}(k+1/M)$ are found as in (3).

Step 4. By the 1-IAGO, find the $\hat{x}(k+1/M)$ as

$$\hat{x}(k+1/M) = [\hat{x}^{(1)}(k+1/M) - \hat{x}^{(1)}(k)] \times M \quad (6)$$

Step 5. By an α filter, the interpolated pixel $\hat{x}(k+1/M)$ is modified as

$$\hat{x}(k+1/M) = \alpha\hat{x}(k) + (1-\alpha)\hat{x}(k+1/M) \quad (7)$$

where $0 \le \alpha \le 1$.

Step 6. Similarly, on a column-by-column base, Steps 1 to 5 are applied to find interpolated pixels.

Note that in Step 6 some pixels are not available. In the case, the interpolated pixels are used in the process of interpolation. The block diagram for the GPIs is depicted in Figure 1.

To show that the 1-AGO is able to reduce randomness in data, an example is depicted in Figure 2. It is easy to see the preprocessed data by the 1-AGO is smoother than the original data. Consequently, better interpolation can be expected when the 1-AGO preprocessed data is put into PIs.

4. Simulation Results and Discussion

In this section, examples are provided to verify the proposed 1-D GPIs. The simulations involve two parts. First, the GPIs are compared with their corresponding PIs. Second, the GPIs are compared with the well-known 2-D polynomial-based interpolators in image enlargement, i.e., the nearest-neighbor interpolator (NNI), the bilinear interpolator (BLI) and the bicubic interpolator (BCI). In the simulations, 512×512 color images Lena, Peppers, Baboon of YC_bC_r format are down sampled to 256×256 images, that is, $M = 2$. Then the three images are up sampled back to the size 512×512 by 1-D PIs, 1-D GPIs, the NNI, the BLI, and the BCI, respectively.

In the first part of simulations, the down sampled images are enlarged by 1-D PIs and GPIs with different orders. The results are given in Table 1 where PI(L) denotes the 1-D PI with order L and GPI(L,α) denotes the GPI of order L with parameter α. For example, GPI(2,0.5) stands for the 2-order GPI with $\alpha = 0.5$. Table 2 shows the differences of PSNRs, ΔPSNRs, between PIs and the corresponding GPIs. The results indicate that the preprocessing scheme 1-AGO improves the performance of PIs from 0.06 dB up to 1.39 dB in the given examples. The result is consistent with the idea that the interpolation performance of PIs can be improved by reducing the randomness in data. Besides, one interesting result is that the GPI(1,0.5) has better performance than all the PIs in Table 1 for all images. It suggests that the 1-AGO preprocessing scheme is able to reduce the order required in the PIs when a similar PSNR is desired.

In the second part of simulations, the GPIs are compared with the 2-D interpolators NNI, BLI, and

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE

BCI. With the given examples, the resulted PSNRs for these 2-D interpolators recorded in Table 3. The ΔPSNRs between the GPIs and the NNI, BLI, BCI are given in Table 4. In Table 3, it indicates that the proposed 1-D GPI(1,0.5) has similar performance to the 2-D BLI in terms of PSNR. And better performance are achieved for the GPI(3,0.5) when compared with the BCI. One more interesting result is that the GPI(2,0.5) has a lightly better PSNR than the BCI on average. As expected, the 1-AGO is able to improve the interpolation performance of PIs. When compared with 2-D interpolators, the preprocessing scheme 1-AGO is able to reduce the order even the dimensionality is different. In other words, the proposed GPIs works with lower order and dimensionality having similar or better interpolation performance compared with the popular 2-D interpolators, in terms of the PSNR. It also can be said that the results imply the proposed GPIs with lower computational complexity than the 2-D interpolators achieve similar or better interpolation performance.

It is well-known that the objective assessment PSNR may not consistent with the visual quality of interpolated images. To compare the visual quality, parts of interpolated Lena and Peppers images obtained from the BLI, BCI, and the GPI(1,0.5), GPI(2,0.5), GPI(3,0.5) are shown in Figures 3 and 4 where the corresponding original images are given as well. From Figures 3 and 4, it is shown that the interpolated images obtained from the GPIs have good visual quality and equally well when compared with the 2-D interpolators.

5. Conclusion

In this paper, we apply the grey preprocessing scheme 1-AGO to 1-D PIs and a class of grey polynomial interpolators (GPI) for image enlargement is proposed. Four stages are involved in the GPIs: (i) to reduce the randomness in image data by the 1-AGO, (ii) to model the 1-AGO converted data by 1-D PIs, (iii) to find interpolated pixels through the 1-IAGO, and (iv) to modify the interpolated pixels by an α filter. Three color images are used to justify the proposed GPIs where comparisons with PIs, the NNI, the BLI, and the BCI are made as well. The simulation results indicate the GPIs outperform their corresponding interpolators. Interesting enough, the GPI(2,0.5) is shown having little better performance than the BCI on average, even the GPI(2,0.5) has lower computational complexity than the BCI. The results imply that the preprocessing scheme 1-AGO is able to improve the interpolation performance of PIs and to reduce the order required as well. It seems also true when compared with the 2-D NNI, BLI, and BCI even the proposed GPIs are 1-D PIs.

Acknowledgment

This work is supported by the National Science Council of the Republic of China under grant NSC 95-2221-E-324-040.

References

[1] Mei-Juan Chen, Chin-Hui Huang, and Wen-Li Lee, "A Fast Edge-Oriented Algorithm for Image Interpolation," *Image and Vision Computing*. Vol. 23, No.9, pp.791-798, Sep. 2005.

[2] T. M. Lehmann, C. Gönner, and K. Spitzer, "Survey: Interpolation Methods in Medical Image Processing," *IEEE Trans. on Medical Imaging*, Vol. 18, No. 11, pp. 1049-1075, Nov. 1999.

[3] M. Unser, "Splines: A Perfect Fit for Signal and Image Processing," *IEEE Signal Processing*, No. 11, pp. 22-38, Nov. 1999.

[4] E. Meijering, "A Chronology of Interpolation: From Ancient Astronomy to Modern Signal and Image Processing," *Proc. IEEE*, Vol. 90, No. 3, pp. 319-342, Mar. 2002.

[5] R.W. Schafer and L.R. Rabiner, "A Digital Signal Processing Approach to Interpolation," *Proc. IEEE*, Vol. 61, No. 6, pp. 692-702, June 1973.

[6] P. Thevenaz, T. Blu, and M. Unser, "Interpolation Revisited," *IEEE Trans. on Medical Imaging*, Vol. 19, No.7, pp. 739 -758, July 2000.

[7] D. P. Mitchell and A. N. Netravali, "Reconstruction Filters in Computer Graphics," *Comput. Graph.*, Vol. 22, pp. 221-288, Aug. 1988,.

[8] W.K. Pratt, *Digital Image Processing*, Wiley, 1978.

[9] B. Lee, "Performance Comparison of Various Interpolation Methods for Color Filter Arrays," *International Symposium on Industrial Electronics*, Vol. 1, pp. 232-236, June 2001.

[10] R. Lukac and K.N. Plataniotis, "Normalized Color-Ratio Modeling for CFA Interpolation," *IEEE Trans. on Consumer Electron*, Vol. 50, No. 2, pp. 737-745, May 2004.

[11] Y.T. Tsai, W.G. Hsu, S.Y. Tseng, and S.L. Cheng, "Optimized Image Processing Algorithms for a Single Sensor Camera," *IEEE Pacific Conference on Communications, Computers and Signal Processing*, Vol. 2, pp. 1010-1013, Aug. 1997.

[12] H. S. Hou and H. C. Andrews, "Cubic Spline for Image Interpolation and Digital Filtering," *IEEE Trans. on Speech and Signal Processing*, Vol. 26, No.9, pp. 508-517, Dec. 1978.

[13] R.G. Keys, "Cubic Convolution Interpolation for Digital Image Processing," *IEEE Trans. on Speech and Signal Process*, Vol. 29, No. 6, pp. 1153-1160, Dec. 1981.

[14] S.D. Bayraker and R.M. Mersereau, "A New Method for Directional Image Interpolation," *IEEE*

International Conference on Acoustics, Speech, and Signal Processing, Vol. 4, pp. 2383-2386, May 1995.

[15] L. Khriji and M. Gabbouj, “Directional-Vector Rational Filters for Color Image Interpolation,” *International Conference on Microelectronics*, pp. 236-240, Dec. 1998.

[16] H. Jiang and C. Moloney, “A New Direction Adaptive Scheme for Image Interpolation,” The 10th *International Conference on Image Processing*, Vol. 3, pp. 369-372, June 2002.

[17] H. Shi and R. Ward, “Canny Edge Based Image Expansion,” *International Symposium on Circuits and Systems*, Vol. 1, pp. 785-788, May 2002.

[18] A.M. Darwish, M.S. Bedair, and S.I. Shaheen, “Adaptive Resampling Algorithm for Image Zooming,” *IEE Vision Image and Signal Processing*, Vol. 144, No. 4, pp. 207-212, Aug. 1997.

[19] X. Li and M.T. Orchard, “New Edge-Directed Interpolation,” *IEEE Trans. on Image Processing*, Vol. 10, No. 10, pp. 1521-1527, Oct. 2001.

[20] Lei Zhang and Xiaolin Wu, “An Edge-Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion,” *IEEE Trans. on Image Processing*, Vol. 15, No. 8, pp. 2226-2238, Aug. 2006.

[21] Cha, Y. and Seongjai Kim, “Edge-Forming Methods for Color Image Zooming,” *IEEE Trans. on Image Processing*, Vol. 15, No. 8, pp. 2315-2323, Aug. 2006.

[22] Christopher J. Zarowski, *An Introduction to Numerical Analysis for Electrical and Computer Engineers*, Wiley, 2004.

Table 1. Comparisons of PSNR for PIs and GPIs

	Lena	Peppers	Baboon
PI(1)	29.30	28.02	21.50
PI(2)	29.11	28.00	21.20
PI(3)	28.95	27.98	20.98
GPI(1,0.5)	29.67	28.28	21.56
GPI(2,0.5)	30.21	28.62	21.63
GPI(3,0.5)	30.34	28.69	21.66

Table 2. ΔPSNRs between PIs and GPIs

	Lena	Peppers	Baboon
PI(1), GPI(1,0.5)	0.37	0.26	0.06
PI(2), GPI(2,0.5)	1.10	0.62	0.43
PI(3), GPI(3,0.5)	1.39	0.71	0.68

Table 3. PSNRs for NNI, BLI, and BCI

	Lena	Peppers	Baboon
NNI	25.74	25.34	19.01
BLI	29.75	28.35	21.57
BCI	29.90	28.63	21.60

Table 4. ΔPSNRs between GPIs and NNI, BLI, BCI

	Lena	Peppers	Baboon
NNI, GPI(1,0.5)	3.93	2.94	2.55
BLI, GPI(2,0.5)	0.46	0.27	0.06
BCI, GPI(3,0.5)	0.44	0.06	0.06

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE

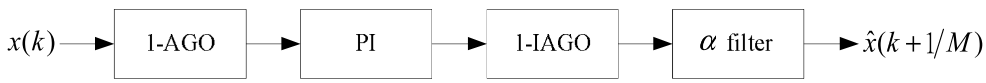

Fig. 1 The block diagram for the proposed GPIs

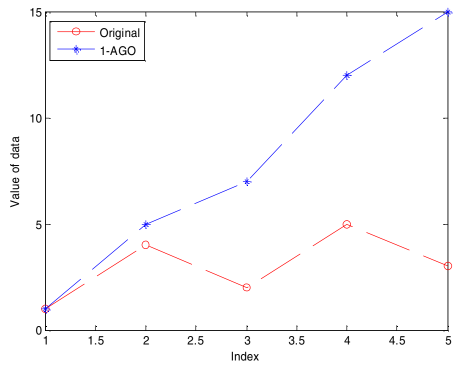

Fig. 2 An example for randomness reduction in the 1-AGO

Fig. 3 (a) Original Lena (b) Interpolated Lena by the BLI (c) Interpolated Lena by the BCI (d) Interpolated Lena by the GPI(1,0.5) (e) Interpolated Lena by the GPI(2,0.5) (f) Interpolated Lena by the GPI(3,0.5)

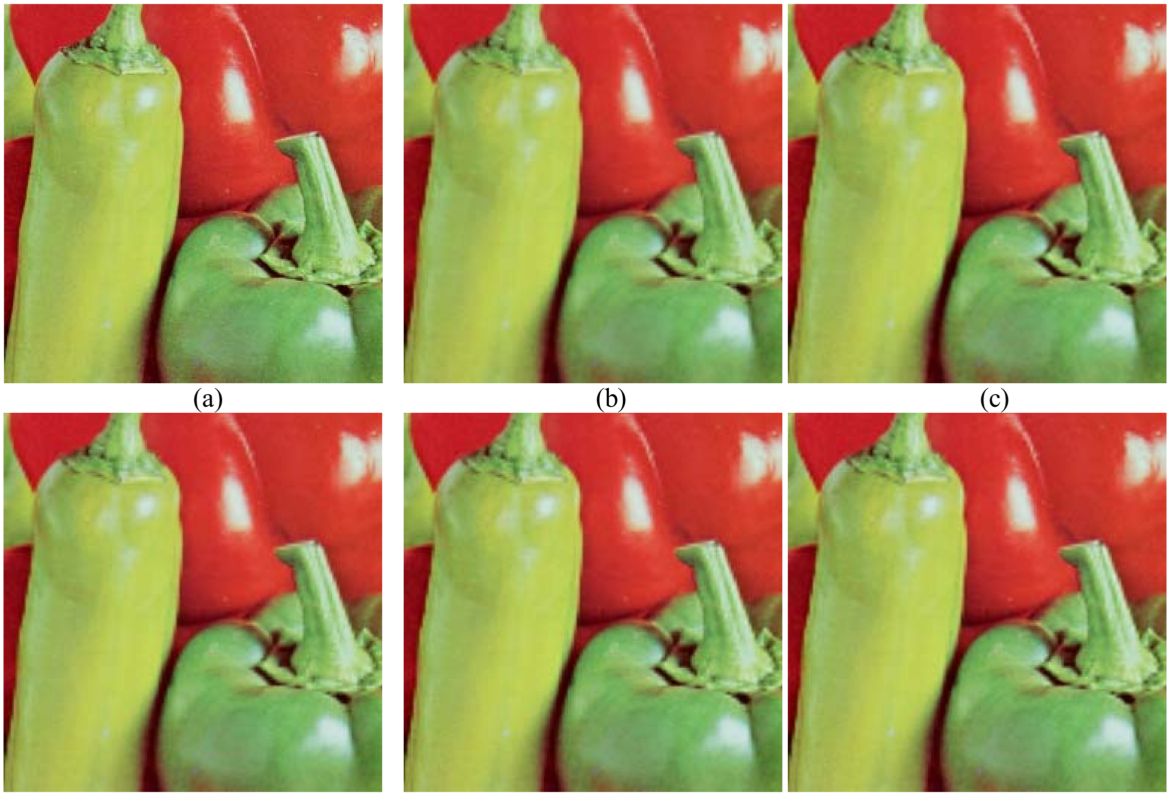

Fig. 4 (a) Original Peppers (b) Interpolated Peppers by the BLI (c) Interpolated Peppers by the BCI (d) Interpolated Peppers by the GPI(1,0.5) (e) Interpolated Peppers by the GPI(2,0.5) (f) Interpolated Peppers by the GPI(3,0.5)

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE