International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.7 (2015), pp.339-350
http://dx.doi.org/10.14257/ijmue.2015.10.7.35

Lagrange Interpolation for Upsampling

Gwanggil Jeon

Department of Embedded Systems Engineering, Incheon National University
119 Academy-ro, Yeonsu-gu, Incheon 406-772, Korea
gjeon@incheon.ac.kr

Abstract

In this paper, we compare well known interpolation methods such as nearest neighbor, bilinear, bicubic, triangle kernel, and Lagrangian interpolation method. Reconstruction errors from above interpolation methods are compared using test image. From the simulation results, it can be found that Lagrangian method outperforms all other upsampling methods. Visual performance comparison is provided by using two LC images.

Keywords: *Lagrangian interpolation, image upsampling, performance comparison*

1. Introduction

Upsampling methods are widely exploited in image processing. For instance, image de interlacing upsamples vertically and enhances vertical resolution [1]. The demosaicking is used for color image interpolation which enhances horizontal and vertical resolution of each color components [2]. Thus the red, green and blue channels are populated and missing information is upsampled in each color channel.

The image upsampling well matches to the case where multiple inputs are anonymous [3]. In this case, the number of missing pixels to be assessed is more than the size of the existing data, estimating missing information is hard to solve [4]. For instance, if we infer missing pixels in an image by a factor of n, a pixel in the input image corresponds to n^2 unknown pixels. The super-resolution method tries to restore a high-resolution image from a low-resolution input with sub-pixel displacements [5]. Many single channel upsampling approaches have been proposed literature [6-7]. Among them, nearest neighbor, bilinear, bicubic, triangle kernel, and Lagrangian interpolation are well known and utilized for upsampling. In this paper, we compare above methods and assess each method's performance.

In this paper, we test some upsampling filters: they are nearest neighbor, bilinear, bicubic, triangle kernel, and Lagrangian methods. These filters are known as work well and relatively fast in general, however some methods are known to be outperforming the others. Section 2 introduces details of all filters. Performance comparison is simulated in Section 3 and Section 4 finishes this paper with conclusion remarks.

2. Various Image Upsampling Filters

Figure 1 shows resulted images with downsampling and upsampling processes with bilinear filter. We used '*Pont du Gard*' image as our test sequence and Figure. 1(a) shows the original image and Figures. 1(b,c) represent downsampled images with factors of 2 and 4 from Figure. 1(a), respectively. Then, Figures. 1(b,c) are upsampled by 'bilinear' method, these results are shown in Figures. 1(d,e). As can be seen in Figures, details and edges of Figure. 1(d) is more well preserved than that of Figuew. 1(e). However, this edge preservation can be further improved by using other upsampling methods.

ISSN: 1975-0080 IJMUE

(a)

(b)

(c)

(d)

(e)

Figure. 1. Downsampling and Upsampling Examples

2.1. Conventional Filters

The nearest neighbor is a method which determines intensity of missing pixel with the locally closest pixel value. Thus once the nearest neighbor method selects the value of the closest point, it ignores the other neighbor information.

Another example is bilinear filter. The bilinear interpolation method is an extension of linear interpolation for interpolating functions of two variables, a and b. If we two points (a_0 ,b_0) and (a_1 ,b_1) are known, then output value b_{out} is obtained as,

$$b_{out}=\frac{b_0\left(a_1-a_0\right)+\left(b_1-b_0\right)\left(a-a_0\right)}{a_1-a_0}. \qquad (1)$$

Other example is bicubic method. The bicubic method is an extension of cubic interpolation for upsampling data points on a two dimensional regular grid. Generally speaking, bicubic method is widely selected over other methods such as bilinear interpolation or nearest neighbor due to its fair performance. However, as bicubic method considers more pixels to calculate, sometimes slow speed of bicubic method can be issued. The weight, w, of pixel position can be explained as,

$$w(a)=\begin{cases}(c+2)\,|\,a\,|^3-(c+3)\,|\,a\,|^2+1, & |\,a\,|\le 1,\\ c\,|\,a\,|^3-5c\,|\,a\,|^2+8a\,|\,a\,|-4c, & 1<|\,a\,|<2,\\ 0, & otherwise.\end{cases} \qquad (2)$$

Another example is cubic spline method which is represented by polynomial functions. In interpolation problems, cubic spline is well adopted to polynomial interpolation tool due to its characteristics that avoids instability.

2.2. Lagrange Interpolation Method

Let us assume we have a set of k points on a Cartesian plane.

$$\left[a_0,b_0\right],\left[a_1,b_1\right],...,\left[a_k,b_k\right]. \qquad (3)$$

The interpolation polynomial in the Lagrange form can be stated as,

$$L(a) := \sum_{j=0}^{k} b_j l_j(a), \tag{4}$$

of Lagrange basis polynomials

$$l_j(a) := \prod_{\substack{0 \le m \le k \\ m \ne j}} \frac{a - a_m}{a_j - a_m}. \tag{5}$$

Finally $L(a_i)$ can be calculated as,

$$\begin{aligned} L(a_i) &= \sum_{j=0}^{k} b_j l_j(a_i) \\ &= \sum_{j=0}^{k} b_j \delta_{ji} = b_i. \end{aligned} \tag{6}$$

2.3. Implementation

A process of Lagrange interpolation calculation is shown in Fig. 2, where six (a,b) point sets are provided as shown in Eq. (7):

```
a=[-9 -4 -1 5 7 9];
b=[ 3 -2 1 4 8 7];
```
(7)

The polynomial interpolation travels through all six control points. Each scaled basis polynomial travels through its corresponding control point. Figure 2 shows an example of Lagrange interpolation where the cubic interpolation polynomial is drawn in solid black, pixels (-9,3), (-4,-2), (-1,1), (5,4), (7,8), and (9,7) are drawn in red, green, blue, magenta, yellow, and cyan color.

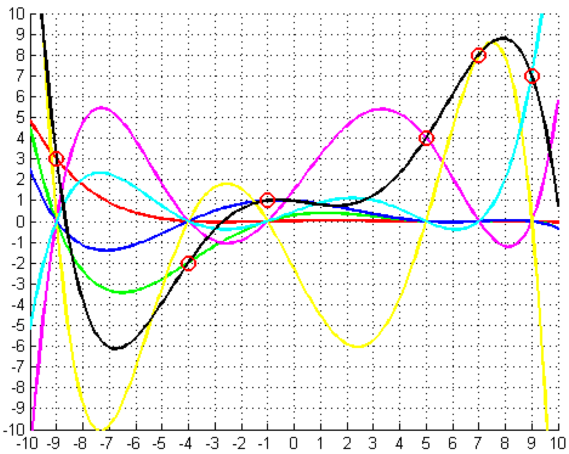


Figure 2. Lagrange Interpolation Example #1

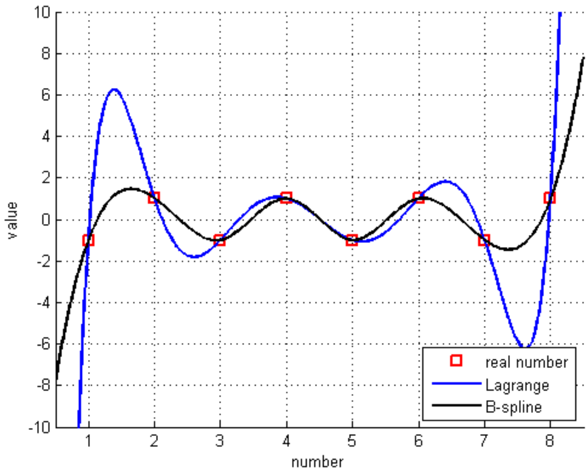


Figure 3. Lagrange Interpolation Example #2

Figure 3 shows an another example of Lagrange interpolation. We used eight points (1,-1), (2,-1), (3,-1), (4,-1), (5,-1), (6,-1), (7,-1), and (8,-1). Lagrange interpolation results are drawn in Fig. 3, where basis-spline is drawn in solid-black and Lagrange interpolation is drawn in blue. Square symbols in red display the existing values.

3. Simulation Results

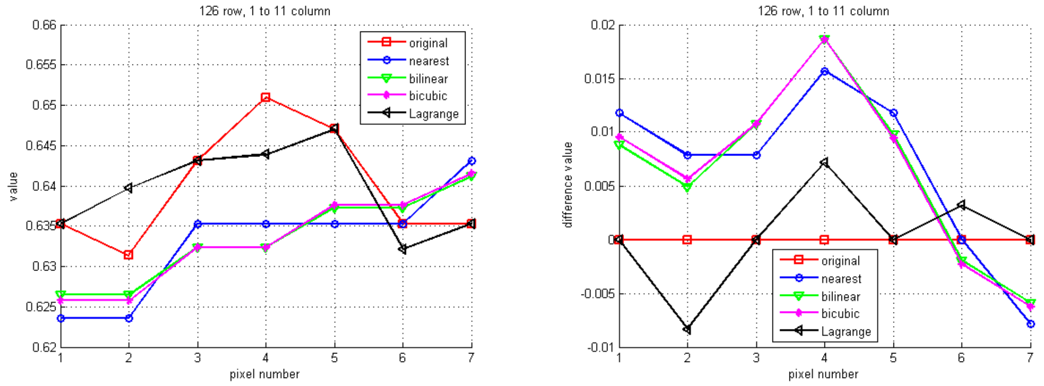
This section compares objective and subjective performance on LC dataset. Figure 4 shows 25 selected images of 150 LC dataset. Among them, images #14 and #18 were used for visual comparison. The size of LC dataset is 720×540 pixels. For objective performance comparison, we used 'Cameraman.tif' image provided in Matlab.

Figure 4. 25 Selected LC Dataset

3.1. Objective Performance Comparison

To compare objective performance, we adopted 'Cameraman' image. To carefully assess each method's performance, we selected three image areas: (a) 126th row and 1-11th, (b) 126th row and 101-111th, and (c) 126th row and 201-211th.

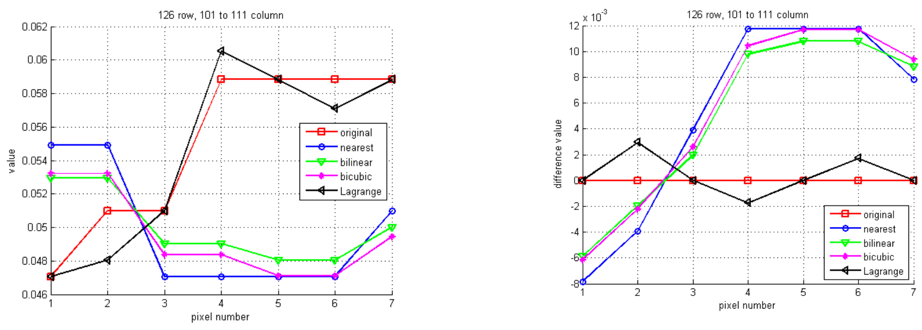
Figure 5 shows objective performance comparison on ‘Cameraman’ image at 126th row and 1 to 11th columns for four methods with original image. The used methods were nearest neighbor method, bilinear, bicubic, and Lagrange method. Figure 5(a) shows each method’s reconstructed intensities and Figure. 5(b) shows intensity differences between original and each used method. Note that intensity ‘0’ and ‘1’ in double represent ‘0’ and ‘255’ in uint8, respectively. From Figure. 5, one can find Lagrange method gives the best performance (the most similar intensities with the original intensities). The bilinear and bicubic methods displayed similar performance and nearest neighbor method was the poorest.



(a) (b)

Figure 5. Objective Performance Comparison on ‘*Cameraman*’ Image at 126th Row and 1-11th Columns for Various Methods: (a) Intensities, (b) Intensity Differences between Original and Each Method

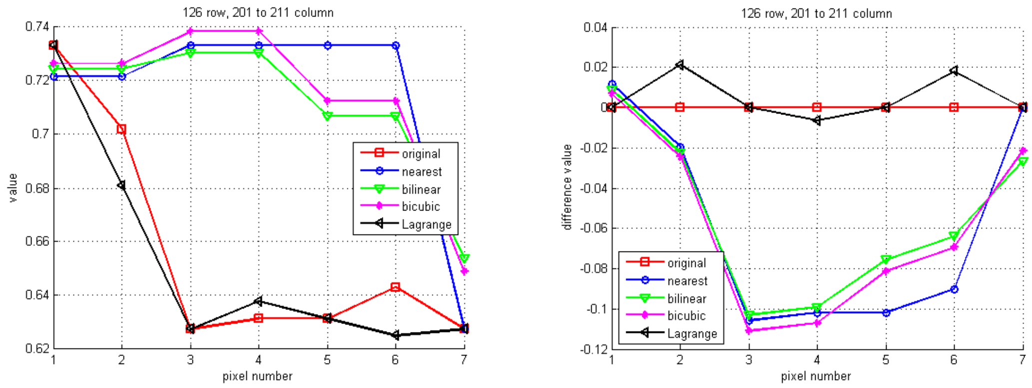
Figure 6 shows results on ‘Cameraman’ image at 126th row and 101 to 111th columns for four methods. As one can see, Lagrange method provided the most similar results with the original pixels. Interestingly, bilinear filter outperformed bicubic filter for these pixels. The nearest neighbor method showed the worst results for these pixels.



(a) (b)

Figure 6. Objective Performance Comparison on *Cameraman* Image at 126th Row and 101-111th Columns for Carious Methods: (a) Intensities, (b) Intensity Differences between Original and Each Method

Finally, Figure 7 shows results on ‘Cameraman’ image at 126th row and 201 to 211th columns for four methods. Still, the Lagrange method produced the best results, followed by bilinear, bicubic, and nearest neighbor methods.



(a) (b)

Figure 7. Objective Performance Comparison on *Cameraman* Image at 126th Row and 201-211th Columns for Various Methods: (a) Intensities, (b) Intensity Differences between Original and each Method

Table 1. Reconstructed Intensities of 126th Row, 3rd to 9th Columns

Methods	3	4	5	6	7	8	9
Original	0.6353	0.6314	0.6431	0.6510	0.6471	0.6353	0.6353
Nearest	0.6235	0.6235	0.6353	0.6353	0.6353	0.6353	0.6431
Bilinear	0.6265	0.6265	0.6324	0.6324	0.6373	0.6373	0.6412
Bicubic	0.6257	0.6257	0.6324	0.6324	0.6376	0.6376	0.6415
Lagrange	0.6353	0.6397	0.6431	0.6439	0.6471	0.6321	0.6353

Table 2. Intensity Difference between Original and the Various Methods of Table 1

Methods	3	4	5	6	7	8	9	abs. diff.
Nearest	0.0118	0.0079	0.0078	0.0157	0.0118	0.0000	-0.0078	0.0090
Bilinear	0.0088	0.0049	0.0107	0.0186	0.0098	-0.0020	-0.0059	0.0087
Bicubic	0.0096	0.0057	0.0107	0.0186	0.0095	-0.0023	-0.0062	0.0089
Lagrange	0.0000	-0.0083	0.0000	0.0071	0.0000	0.0032	0.0000	0.0027

Table 3. Reconstructed Intensities of 126th Row, 103rd to 109th Columns

Methods	3	4	5	6	7	8	9
Original	0.0471	0.0510	0.0510	0.0588	0.0588	0.0588	0.0588
Nearest	0.0549	0.0549	0.0471	0.0471	0.0471	0.0471	0.0510
Bilinear	0.0529	0.0529	0.0490	0.0490	0.0480	0.0480	0.0500
Bicubic	0.0532	0.0532	0.0484	0.0484	0.0471	0.0471	0.0494
Lagrange	0.0471	0.0480	0.0510	0.0605	0.0588	0.0571	0.0588

Table 4. Intensity Difference between Original and the Various Methods of Table 3

Methods	3	4	5	6	7	8	9	abs. diff.
Nearest	-0.0078	-0.0039	0.0039	0.0117	0.0117	0.0117	0.0078	0.0084
Bilinear	-0.0058	-0.0019	0.0020	0.0098	0.0108	0.0108	0.0088	0.0071
Bicubic	-0.0061	-0.0022	0.0026	0.0104	0.0117	0.0117	0.0094	0.0077
Lagrange	0.0000	0.0030	0.0000	-0.0017	0.0000	0.0017	0.0000	0.0009

Figures 5-7 can be tabulated in Tables 1-3. Odd numbers of tables are intensity values while even numbers of Tables are intensity differences between original and various methods. The last columns of even numbered Tables are absolution difference results. It means, larger 'abs. diff.' indicates bigger variance and smaller 'abs. diff.' represents small variance.

Table 5. Reconstructed Intensities of 126th Row, 203rd to 209th Columns

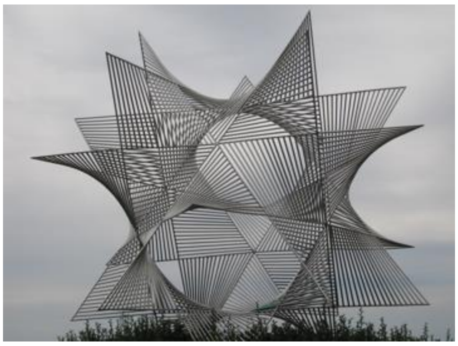
Methods	3	4	5	6	7	8	9
Original	0.7333	0.7020	0.6275	0.6314	0.6314	0.6431	0.6275
Nearest	0.7216	0.7216	0.7333	0.7333	0.7333	0.7333	0.6275
Bilinear	0.7245	0.7245	0.7304	0.7304	0.7069	0.7069	0.6539
Bicubic	0.7264	0.7264	0.7384	0.7384	0.7126	0.7126	0.6490
Lagrange	0.7333	0.6809	0.6275	0.6377	0.6314	0.6250	0.6275

Table 6. Intensity Difference between Original and the Various Methods of Table 5

Methods	3	4	5	6	7	8	9	abs. diff.
Nearest	0.0117	-0.0196	-0.1058	-0.1019	-0.1019	-0.0902	0.0000	0.0616
Bilinear	0.0088	-0.0225	-0.1029	-0.0990	-0.0755	-0.0638	-0.0264	0.0570
Bicubic	0.0069	-0.0244	-0.1109	-0.1070	-0.0812	-0.0695	-0.0215	0.0602
Lagrange	0.0000	0.0211	0.0000	-0.0063	0.0000	0.0181	0.0000	0.0065

3.2. Visual Performance Comparison

Subjective performance comparison is tested using two LC images of LC dataset: #14 and #18 images. Figure 8 shows both original images.



(a)

(b)

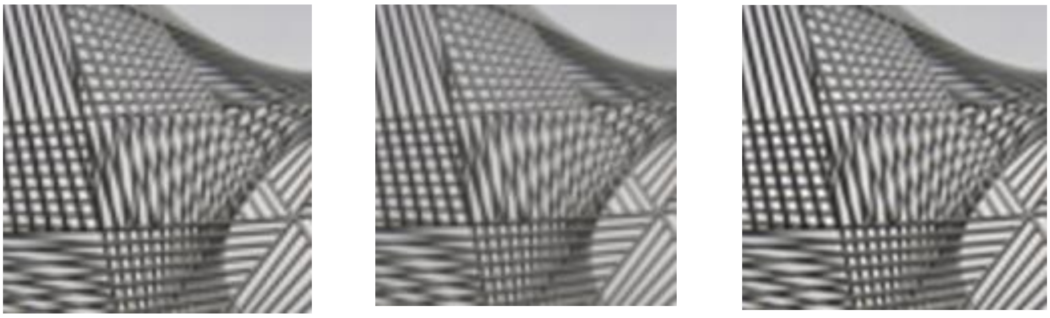
Figure 8. Test LC Dataset: (a) #14 LC Image, (b) #18 LC Image.

The original images are downsampled by factor of 2, and upsampled using various methods. Figures 9(a) and 10(a) show zoomed results of various methods. Original images are shown in Figures. 9(a) and 10(a), and results of nearest neighbor, bilinear, bicubic, triangle filter, and Lagrange interpolation are displayed in Figures. 9(b-f) and 10(b-f).

Restored images produced various artifacts. For instance, Figures. 9(b) and 10(b) showed staircase artifact and 'yellow and orange' color-artifacts were shown at high frequency area. Figures 9(c,e) and 10(c,e) showed blurred results. Although Figures. 9(d) and 10(d) produced blurred images, but edges are more preserved than Figures. 9(c,e) and 10(c,e). In contrast, Figures. 9(f) and 10(f) provided the best visual results where details were well preserved.

Figures 11 and 12 show the difference images between original and produced reconstructed ones. These figures clearly indicate that Lagrange method produced the least artifacts and provide the best results out of all compared methods.

(a) (b) (c)



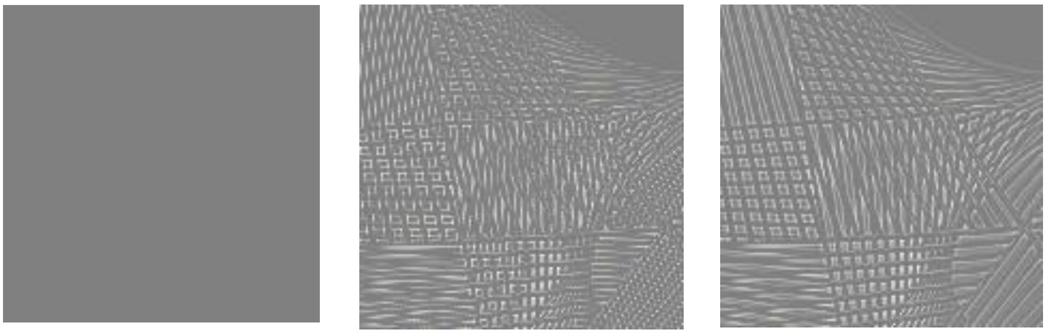
(d) (e) (f)

Figure 9. Restored Images: (a) Original #14 LC Image, (b) Nearest Neighbor Method, (c) Bilinear Method, (d) Bicubic Method, (e) Triangle Filter, and (f) Lagrange Method

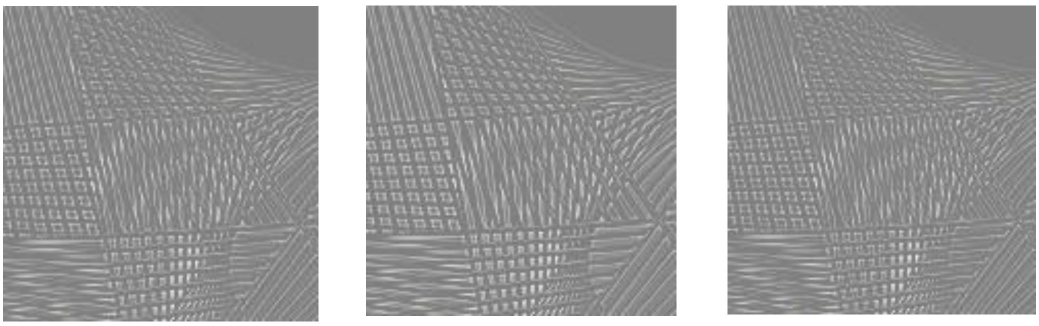
(a) (b) (c)

(d) (e) (f)

Figure 10. Restored Images: (a) Original #18 LC Image, (b) Nearest Neighbor Method, (c) Bilinear Method, (d) Bicubic Method, (e) Triangle Filter, and (f) Lagrange Method



(a) (b) (c)



(d) (e) (f)

Figure 11. Difference between Original and Restored Images: (a) Original #14 LC Image, (b) Nearest Neighbor Method, (c) Bilinear Method, (d) Bicubic Method, (e) Triangle Filter, and (f) Lagrange Method

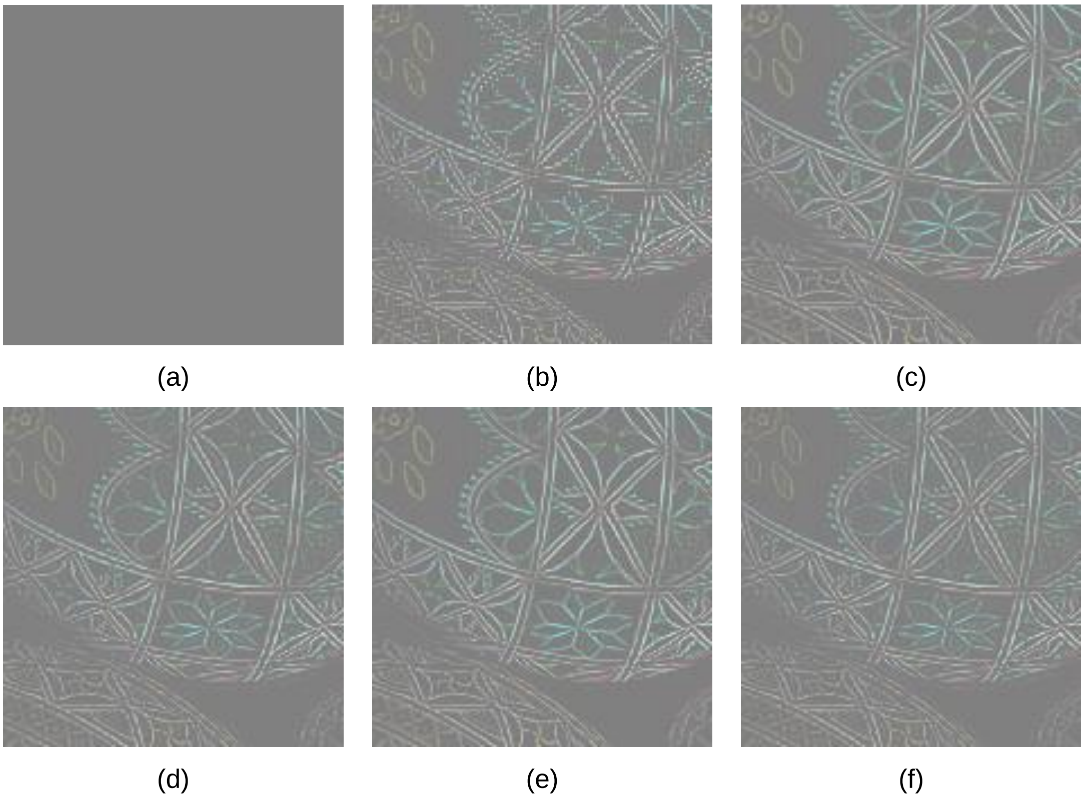


Figure 12. Difference between Original and Restored Images: (a) Original #18 LC Image, (b) Nearest Neighbor Method, (c) Bilinear Method, (d) Bicubic Method, (e) Triangle Filter, and (f) Lagrange Method

4. Conclusion

In this paper, we compared conventional interpolation filters such as nearest neighbor, bilinear, bicubic, triangle kernel, and Lagrangian interpolation method. Experimental results show that the Lagrangian method provided the best performance among all compared methods.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIP) (2014025627). This paper is a revised and expanded version of a paper entitled “Lagrange method for upsampling” presented at AST2015.

References

[1] N. Jayant and P. Noll, “*Digital Coding of Waveforms: Principles and Applications to Speech and Video*, Prentice-Hall., **(**1984**)**.
[2] S. Daly and A. Watson, “The visible differences predictor: An algorithm for the assessment of image fidelity,” *Digital Images and Human Vision*, MIT Press, **(1993)**.
[3] K. Ratakonda and N. Ahuja, “POCS based adaptive image magnification,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, **(1998)**, pp.203-207.
[4] D. Calle and A. Montanvert, “Superresolution inducing of an image,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, **(1998)**, pp.232-235.
[5] J. Allebach and P. W. Wong, “Edge-directed interpolation,” in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, **(1996)**, pp.707-710.
[6] J. W. Woods, T. S. Huang, “Two-dimensional Kalman filtersm” *Two-Dimensional Digital Signal Processing*, Springer-Verlag, vol. 42, **(1981)**, pp.155-205.

[7] J. E. Adams Jr, "Design of practical color filter array interpolation algorithms for digital cameras," in *Proc. SPIE*, vol. 3028, **(1997)**, pp.117-125.

Author

Gwanggil Jeon, he received the BS, MS, and PhD (summa cum laude) degrees in Department of Electronics and Computer Engineering from Hanyang University, Seoul, Korea, in 2003, 2005, and 2008, respectively.

From 2008 to 2009, he was with the Department of Electronics and Computer Engineering, Hanyang University, from 2009 to 2011, he was with the School of Information Technology and Engineering (SITE), University of Ottawa, as a postdoctoral fellow, and from 2011 to 2012, he was with the Graduate School of Science & Technology, Niigata University, as an assistant professor. He is currently an assistant professor with the Department of Embedded Systems Engineering, Incheon National University, Incheon, Korea. His research interests fall under the umbrella of image processing, particularly image compression, motion estimation, demosaicking, and image enhancement as well as computational intelligence such as fuzzy and rough sets theories.

He was the recipient of the IEEE Chester Sall Award in 2007 and the 2008 ETRI Journal Paper Award.