International Journal of Engineering & Technology, 8 (1.4) (2019) 73-81

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Training-based Image Upsampling using Kernels

Seungjong Kim

Department of Computer Science & Information Systems,
Hanyang Women's University, Korea
**Corresponding author E-mail: jkim@hywoman.ac.kr*

Abstract

This paper proposes an issue of upsampling which populates missing pixel components at the zero-padded images. This process is called image upsampling (or image reconstruction). In this paper, a kernel design approach is studied in detail, which is obtained based on least-squares method by exploiting training set to achieve desired upsampling performance. To meet the complexity requirement, tradeoff approach is considered during the kernel design. Performance of the proposed method is compared with that of other conventional up-sampling methods such as nearest neighbor, bilinear interpolation, and bicubic interpolation, in terms of objective and subjective quality metrics. Simulation results inform that the proposed method provides outstanding performance when it is compared with conventional methods.

Keywords: *Image upscaling, least squares method, kernel design, image training.*

1. Introduction

The image upsampling in display devices is the doubling of separate pixels in horizontal or vertical dimensions which can be presented in certain size device [1,2]. The resolution of given images is generally represented with two parameters: the width in horizontal direction and the height in vertical direction, where the unit is pixel. For example, if a device has 1920×1080 size image, this tells that width size is 1920-pixel and the height size is 1080-pixel. Therefore, image upsampling is generally used to show pixel dimensions. The image up-sampling is used to enlarge or sharpen particular images in a window. Or, sometimes it smoothers rough details in images which have been revealed by a unique upsampling method.

Many single channel (black and white) upsampling methods have been proposed in literature [3-5]. Among them, the nearest neighbor, the bilinear interpolation, and the bicubic interpolation are widely known and exploited for upsampling execution. Therefore, improving a given low resolution (LR) image to high-quality high resolution (HR) resolution image is an important task. There are many applications of image upsampling, which cover deinterlacing, demosaicking, super resolution, etc.

To build an HR image using a LR one, image upsampling method is required. However, image upsampling method may bring unwanted artefacts, and this may cause inferior effect for spectators. Several researches with different level of performance and complexity have been conducted where effective image upsampling methods were introduced [6]. To enlarge an image with factor of 2, some researchers have been taking advantage of nearest neighbor method, bilinear method, and bicubic method. However, nearest neighbor implemented results provide blocking or staircase artefact, which is not acceptable for commercial purpose devices. On the other hand, the bilinear and bicubic methods provide blurring artefact which provide annoying effects for spectators [7]. To alleviate this issue, the purpose of most upsampling methods is to present an upsampling method which reduces blocking artefacts while giving pleasant visual effect [8-17].

There are several applications such as high-definition television (HDTV) and satellite images, where image upsampling is widely used. With the increase digital cameras and displays, users may want to see different size of images on their devices. To do so, digital zooming technique is needed which has a function in grasping details in a given image. As HDTV industry marketplace is growing, researchers are promptly fascinated developing upsampling methods for watching conventional low resolution videos on devices. Not just only HDTV industry, image upsampling techniques are widely needed in astronomical or medical image processing fields.

This paper investigates kernel design method which minimizes the mean-squared error over a training image. Then, the desired trained filter is obtained, which is used for image upsampling during image reconstruction.

This paper is arranged as follows. Conventional image upsampling methods are explained in Section 2. These methods include nearest neighbor method, bilinear method, and bicubic method. In Section 3, the kernel design approach and the proposed kernel-based upsampling method is explained. Simulation results are presented in Section 4, where three objective performances and visual performance are provided and compared. Finally, conclusion remarks are made in Section 5.

2. Conventional upsampling methods

The image upsampling is considered as a concept of image restoration in the view of Nyquist sampling theorem (NST) which is an essential bridge between continuous time signal and the discrete time signal. Based on this Nyquist sampling theorem, sampling frequency

Copyright © 2019 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

must be twice higher than the highest frequency of given signal. Otherwise, image information can be lost during the restoration. There have been several methods to upsample images. The most well-known upsampling methods are nearest neighbor method, the bilinear interpolation method, and the bicubic interpolation method. Among them, the nearest neighbor method is the simplest approach which populates intensity of missing pixel with the locally nearest pixel brightness. Once the nearest neighbour method chooses the intensity of the closest location, it disregards other neighbor information.

The nearest-neighbor method is one of the simplest methods which increase image size by duplicating existing pixels. This method substitutes for every pixel with twice (triple, multiple, and depending on factor *N*) pixels of the given pixel. In general, the resulting images contain detail preserved original pixels, however generated pixels show serious staircase artefact.

The bilinear and bicubic methods are more complicated than the nearest-neighbor interpolation method. These methods restore images by interpolating existing intensity values, presenting a continuous change into the output even where the original pixels have discrete changes. The resulting images of these methods show continuous tone images, and therefore blur effects can be found. Thus, edge sharpness is lost and details can be reduced. The bicubic method provides slightly better performance than that of bilinear by paying increased computational complexity. The bilinear interpolation method is a branch of linear interpolation for populating functions of two variables, *m* and *n*. If two pixels (m_0, n_0) and (m_1, n_1) are known, then output intensity n_{out} is computed as,

$$n_{out} = \frac{n_0(m_1 - m_0) + (n_1 - n_0)(m - m_0)}{m_1 - m_0}. \tag{1}$$

If we select a coordinate system where the four locations at (0,0), (0,1), (1,0), and (1,1), then the interpolation function can be simplified as,

$$f(x,y) = f(0,0)(1-x)(1-y) + f(1,0)x(1-y) + f(0,1)(1-x)y + f(1,1)xy. \tag{2}$$

This equation can be further simplified as,

$$f(x,y) = \begin{bmatrix} 1-x & x \end{bmatrix} \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix}. \tag{3}$$

The third conventional method is the bicubic interpolation method. The bicubic method is a branch of cubic interpolation for populating data locations on a two-dimensional regular grid. In general, the bicubic method is well adopted over commercial image and video devices because other competitors such as bilinear interpolation or nearest neighbor do not provide comparable performance. However, the bicubic method is more complex than that of bilinear method. To implement bicubic method, weight parameter, $\omega(x)$, is needed. The weight parameter is computed as,

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

In equation (4), the parameter *c* is normally set to -1/2 or -3/4. It is noted that $\omega(0)=1$ and $\omega(k)=0$ for all non-zero integers *k*.

3. Proposed method

In contrast to the conventional linear interpolation method, the least squares filter-based upsampling method is elastic approach which can be utilized in several applications of image processing. This method is a form of equational regression estimation which restore the missing pixels of the most successful fit for a given low resolution image, granting a subjective exhibition of the relationship between the pixels in an image. Each pixel in an image is neighbour of the relationship between an existing pixel and the missing pixel.

In this work, we used the least squares approach which yields the universal solution for the estimation of the line of appropriate fit among the images pixels being dealt with. The basic purpose of least squares approach is to generate a line which optimize the upsampling system by minimizing the sum of differences (also known as error) obtained by the results of the collaborated equations. The lines can be linear or squared, and the resulting errors are computed by observing differences between the original signal and the estimated signal.

The least squares method is a form of versatile filter which is used to minimize errors. The desired filter is obtained by training process with the form of kernel coefficients that tells showing the least squares of the error between original and the restored images. In general, this error is obtained as the difference between the desired and the original images. It is assumed that $x_O(i,j)$ is the original image, which we restore by kernel filtering given LR image $x_I(i,j)$ with the linear and shift-invariant kernel using unit-sample reply $x'_O = x_I * h_{DK}$, where h_{DK} is desired kernel acquired by training process. If we assume the remainder between x'_O and x_O is a stationary random field, then we can calculate h_{DK} as,

$$h_{DK} = \arg\min_h COST\left[\left\{x_O(i,j) - (x_I * h)(i,j)\right\}^2\right]. \tag{5}$$

Now, the desired kernel is computed as equation (6), and Φ is represented by equation (7).

$$h_{DK} = \arg\min_h \sum_{\kappa=1}^{U} \sum_{(i,j)\in B^{(\kappa)}} \left\{x_O^{(\kappa)}(i,j) - \Phi\right\}^2 \tag{6}$$

$$\Phi = \sum_{(m,n)\in L} h(m,n) x_I^{(\kappa)}(i-m, j-n) \tag{7}$$

We segmented the training process into U sub-images, where κ^{th} sub-image is described on the partial block $B^{(\kappa)}$. Then, the desired filter h_{DK} is a two dimensional finite impulse response filter with area of support L. Finally, equation (6) is represented in matrix form as equation (8).

$$\mathbf{h}_{DK} = \arg\min_{\mathbf{h}} \sum_{\kappa=1}^{U} \left\| \mathbf{R}^{(\kappa)} \mathbf{h} - \mathbf{x}_O^{(\kappa)} \right\|^2 \tag{8}$$

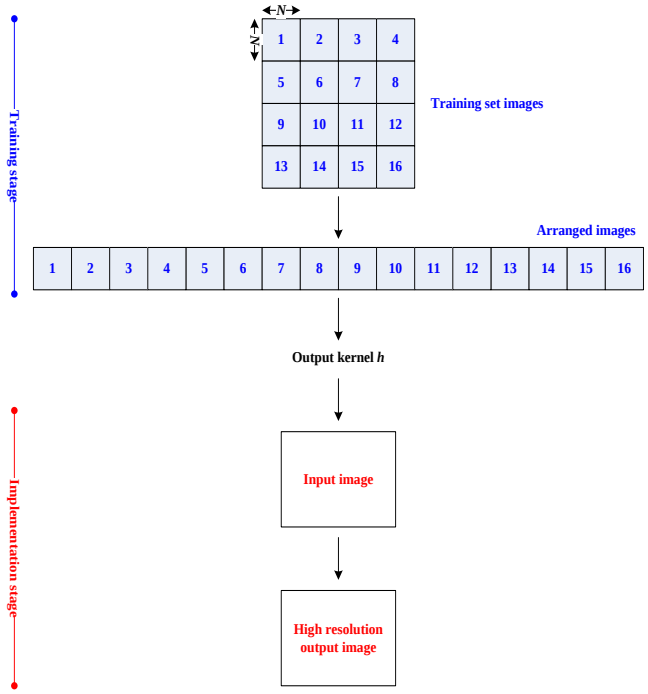
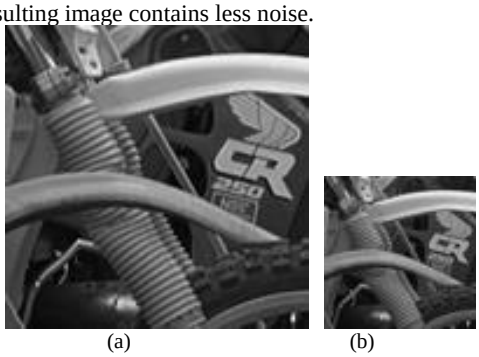


Fig. 1: Block diagram of the proposed method

Finally, h_{DK} can be straight-forwardly computed with the standard least-squares form as equation (9). It is noted that $\mathbf{R}^{(\kappa)}$ is computed as $B_N * L_N$.

$$\mathbf{h}_{DK} = \left[\sum_{\kappa=1}^{U} \left(\mathbf{R}^{(\kappa)} \right)^T \mathbf{R}^{(\kappa)} \right]^{-1} \left[\sum_{\kappa=1}^{U} \left(\mathbf{R}^{(\kappa)} \right)^T \mathbf{x}_O^{(\kappa)} \right] \tag{9}$$

Figure 1 shows the block diagram of the proposed method. As can be seen in this figure, the proposed method has two stages: (Stage 1) *Training stage* and (Stage 2) *Implementation stage*. In the *Training stage*, desired kernel is computed using least-squares method. In the *Implementation stage*, HR image is achieved by using input LR image with kernel h. It is noted that we considered block size is 8×8 (N=8). Figure 2 shows the upsampling process in detail. Figure 2(a) is the original image, while Figure 2(b) is the downsampled LR image. Figures 2(c) and 2(d) are zero-padded image and the restored image. Figure 2(e) is the frequency response coefficients of kernel h. It is noted that this kernel coefficients were obtained by using single Kodak #5 image. Figure 3 shows frequency response of bilinear interpolation kernel and the proposed kernel. The proposed kernel well preserves low frequency information, and well removes high frequency spectrum to avoid aliasing effect. As can be seen in Figure 3(b), cut-off frequency of the proposed kernel is closer to the centre, which prevents blurred image and therefore the resulting image contains less noise.



(a) (b)



(c)

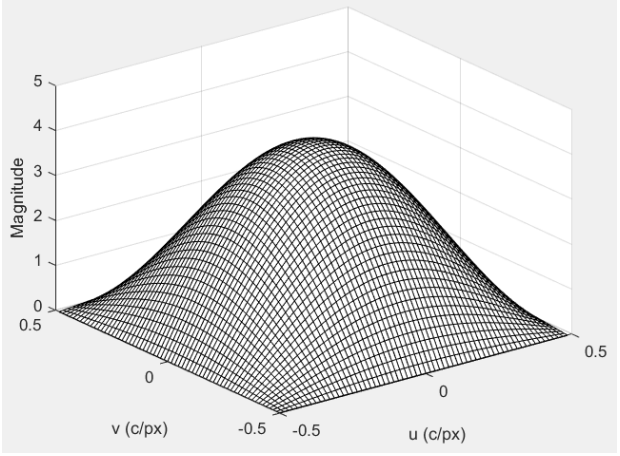


(d)

8	0.0080	0.0000	(0.0436)	0.0000	0.0760	0.0000	(0.0176)	0.0000
7	0.0093	0.0120	(0.0328)	0.0235	0.3575	0.5406	0.2744	0.0128
6	0.0058	0.0000	(0.0504)	(0.0000)	0.5332	1.0000	0.5355	(0.0000)
5	0.0002	(0.0055)	(0.0264)	0.0135	0.2744	0.5452	0.3612	0.0211
4	(0.0010)	(0.0000)	(0.0040)	0.0000	(0.0218)	0.0000	0.0782	(0.0000)
3	0.0031	0.0024	(0.0078)	(0.0204)	(0.0416)	(0.0616)	(0.0100)	0.0016
2	(0.0006)	0.0000	(0.0028)	(0.0000)	0.0121	(0.0000)	(0.0209)	0.0000
1	(0.0068)	(0.0055)	0.0093	0.0154	0.0118	(0.0016)	(0.0336)	(0.0015)
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**

(e)

Fig. 2: Upsampling process: (a) Original image, (b) downsampled image with factor of 2, (c) zero-padded image, (d) resulting image by upsampling method, and (e) 8×8 kernel coefficients trained by Kodak #5 image. It is noted that the numbers in bracket are negative.



(a)

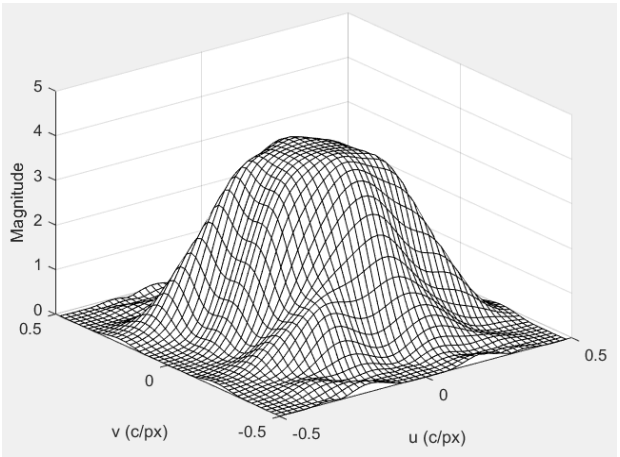


Fig. 3: Examples of frequency responses: (a) bilinear interpolation kernel, (b) proposed kernel.

4. Experimental results

This section describes objective and subjective performance on Kodak dataset [18]. Figure 4 shows 4 images of Kodak dataset. Each image is either 768×512 or 512×768 in size, but we listed them as sideways images. To show the superiority of the proposed method, we used four test images: #1, #8, #19, and #24, in visual performance comparison.

The original Kodak images are downsampled by factor of 2, then upsampled using various methods. We used two benchmark methods: bilinear and bicubic methods. In Figures 5-8, the original images and interpolated images using bilinear, bicubic, and proposed method are shown.

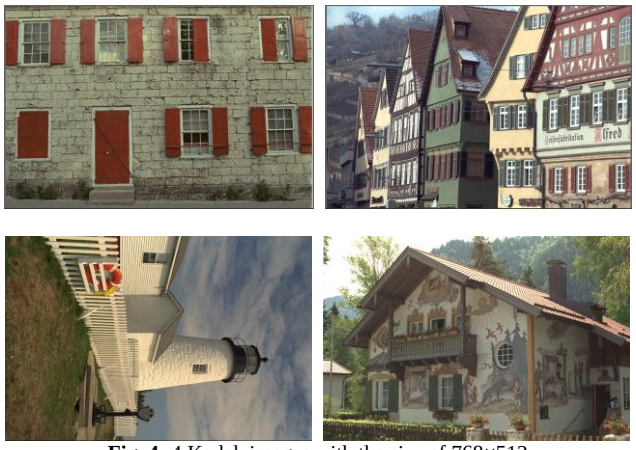


Fig. 4: 4 Kodak images with the size of 768×512

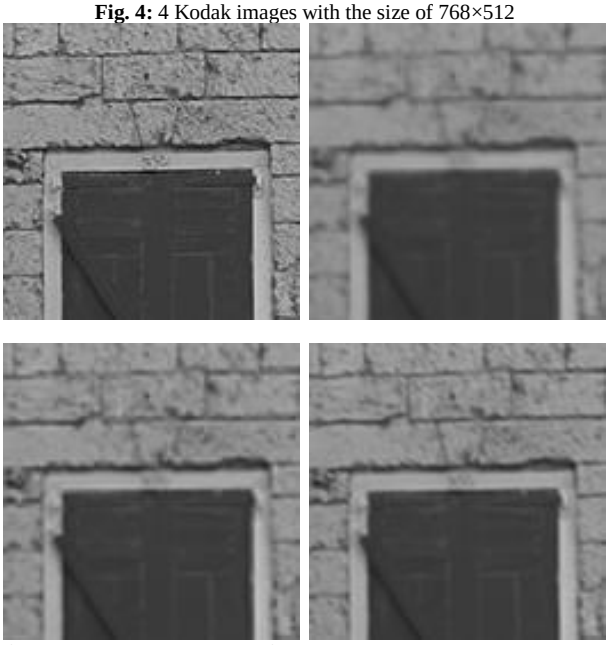


Fig. 5: Visual performance comparison of the Kodak #1 image. Top left: original image, top right: bilinear, bottom left: bicubic, and bottom right: the proposed method.

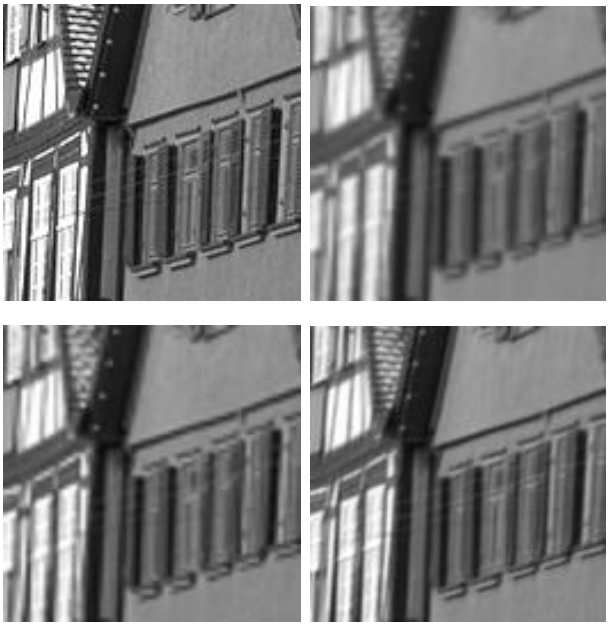


Fig. 6: Visual performance comparison of the Kodak #8 image. Top left: original image, top right: bilinear, bottom left: bicubic, and bottom right: the proposed method.

4.1. Visual performance comparison

Figure 5(a) shows the Kodak #1 original image. Figures 5(b)-5(c) are HR images by bilinear and bicubic methods. Figure 5(d) shows the resulting image by the proposed kernel. The subjective quality can be assessed with regard to reconstruction of edges, structures, and assorted kinds of geometric lists such as corners, diagonals, and delicate patterns. The nearest neighbour method returned zipper artefact and therefore restored images were not well connected. The bilinear method returned totally blurred image, and even door gate was not well restored. Although the bicubic method provides relatively comparable results, the proposed method well restored door gate and brick. Figure 6 are obtained from the Kodak #8 image. Figure 6(b) shows blurred image and therefore image details were removed during the restoration. Figures 6(c) and 6(d) are comparable, however the proposed method presents pleasant image in the area of window and roof.

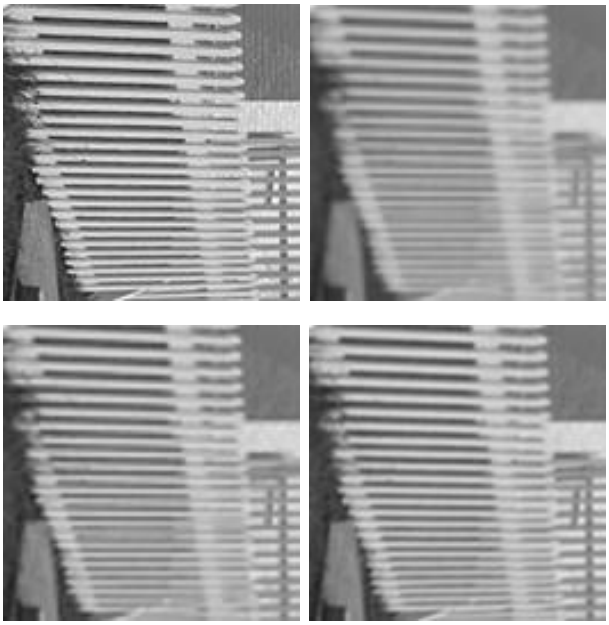


Fig. 7: Visual performance comparison of the Kodak #19 image. Top left: original image, top right: bilinear, bottom left: bicubic, and bottom right: the proposed method.



Fig. 8: Visual performance comparison of the Kodak #24 image. Top left: original image, top right: bilinear, bottom left: bicubic, and bottom right: the proposed method.

Figure 7 was adopted from the Kodak #19 image, especially; picket fence region was used for comparison. Figures 7(b) and 7(c) are resulting images of FIR filters. One of main drawbacks of FIR filters is that it produces blurred images and therefore its image details are lost. Although the proposed method does not fully restore the original image, but it looks similar to the original image compare to any other methods.

Figure 8 shows resulting images of the Kodak #24 image. As one can see, the proposed method outperformed all other conventional methods and even preserved flowers. However, the other methods produced many visible artefacts and false pattern along the edges of fence and window. Figures 9 and 10 show difference images between the original image and the HR images obtained by benchmark methods and the proposed kernel. It can be easily seen that the difference images by the proposed method is the least compared to the other methods. It can be seen that bicubic interpolations provides better results than nearest neighbor and bilinear interpolation. However, the proposed method provides the best visual performance out of all compared methods. It is obvious that the resulting image obtained by nearest neighbor has a lot of data scrambling, and therefore errors are added in the results. On the other hand, bilinear and bicubic result-

ing images have powerful reduction of high frequencies information, and therefore image edges became less. It can be concluded that the bilinear and bicubic mthods perform awfully in high frequencies area.

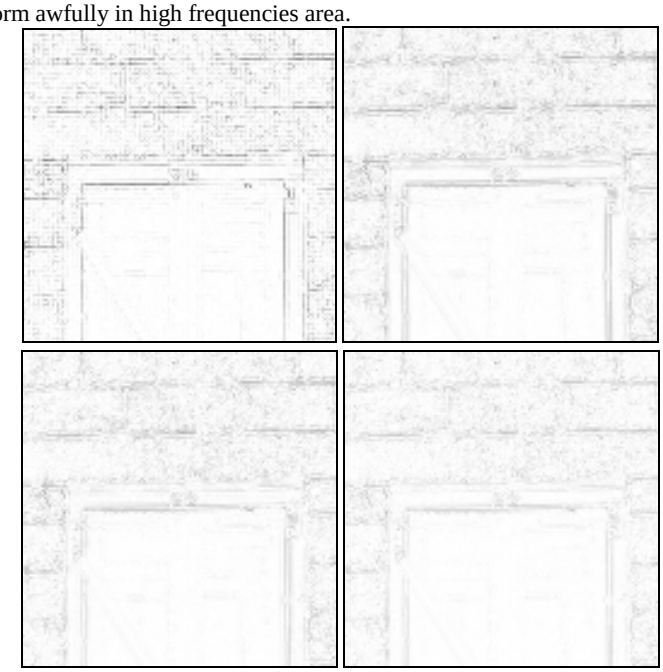


Fig. 9: Difference images from the original Kodak #1 image. Top left: nearest neighbour, top right: bilinear interpolation, bottom left: bicubic interpolation, and bottom right: the proposed method.



Fig. 10: Difference images from the original Kodak #8 image. Top left: nearest neighbour, top right: bilinear interpolation, bottom left: bicubic interpolation, and bottom right: the proposed method.

4.2. Objective performance comparison

Simulating the objective performance at higher resizing factor (in this work, resizing factor is 2) extremely decrease the image quality. The term peak signal-to-noise ratio (PSNR) is often used as a quality measurement between the original and a reconstructed image. The higher PSNR is the better quality of the reconstructed image. The MSE represents the cumulative squared error between the reconstructed and the original image, whereas PSNR represents a measure of the peak error. The lower value of MSE is the lower error. Table 1 and Table 2 show objective performance comparison in two metrics: MSE and PSNR. The proposed method outperforms other benchmark methods by 6.002 dB (nearest neighbor), 3.870 dB (bilinear), and 2.169 dB (bicubic). From the provided MSE and PSNR results of the various interpolation methods, it is implied that the proposed method performed better in a quantitative way. It is noted that some of the boundary pixels (in this simulation, we assumed 11 pixels at the boundary) are scarified before measuring the objective performance.

Table 1: Objective performance comparison (MSE)

Kodak image #	Nearest neighbor	Bilinear method	Bicubic method	Proposed method
1	386.88	249.13	180.14	110.12
2	72.85	47.27	34.75	22.50
3	58.83	39.95	26.70	13.52
4	83.69	47.75	30.25	18.85

5	419.12	271.06	170.89	97.95
6	277.02	172.32	127.90	78.12
7	114.11	61.40	32.58	18.56
8	777.50	461.92	331.93	191.43
9	119.12	71.16	43.52	25.85
10	102.14	66.09	43.69	20.85
11	209.20	128.46	92.29	59.57
12	80.08	48.44	33.44	20.84
13	643.73	401.21	300.93	204.62
14	227.12	138.90	92.26	57.27
15	95.92	52.70	34.65	22.84
16	118.89	73.87	56.34	34.58
17	118.10	68.25	45.05	29.40
18	261.10	160.77	110.84	74.08
19	274.45	173.09	118.64	70.39
20	133.08	77.50	51.03	31.53
21	253.95	152.33	106.56	67.52
22	151.55	95.62	66.96	43.29
23	66.59	41.53	23.41	12.55
24	325.17	207.19	149.68	95.17
Average	**223.76**	**137.83**	**96.02**	**59.22**

Table 2: Objective performance comparison (PSNR)

Kodak image #	Nearest neighbor	Bilinear method	Bicubic method	Proposed method
1	22.255	24.167	25.575	27.712
2	29.507	31.385	32.721	34.610
3	30.435	32.115	33.866	36.821
4	28.904	31.341	33.323	35.377
5	21.907	23.800	25.804	28.221
6	23.706	25.767	27.062	29.203
7	27.558	30.249	33.002	35.446
8	19.224	21.485	22.920	25.311
9	27.371	29.608	31.744	34.006
10	28.039	29.929	31.727	34.941
11	24.925	27.043	28.479	30.380
12	29.095	31.279	32.888	34.941
13	20.044	22.097	23.346	25.021
14	24.568	26.704	28.481	30.551
15	28.312	30.913	32.733	34.543
16	27.379	29.446	30.622	32.743
17	27.408	29.790	31.594	33.447
18	23.963	26.069	27.684	29.434
19	23.746	25.748	27.389	29.656
20	26.890	29.238	31.053	33.144
21	24.083	26.303	27.855	29.837
22	26.325	28.325	29.872	31.767
23	29.897	31.948	34.437	37.145
24	23.010	24.967	26.379	28.346
Average	**25.773**	**27.905**	**29.606**	**31.775**

The SSIM metric, structural similarity is a metric to measure the similarity between original image and the restored image [19]. The SSIM index can be observed as a quality measure, and can be a good tool to assess image quality. Table 3 shows objective performance results obtained by SSIM metric, where pixel intensity close to 1.0 is the pixel with well restored. The proposed method provides the most pleasant resulting images compared to all other methods and outperformed other methods by margins of 0.13061, 0.10863, and 0.04818, respectively.

Table 3: Objective performance comparison (SSIM)

Kodak image #	Nearest neighbor	Bilinear method	Bicubic method	Proposed method
1	0.65538	0.64989	0.74830	0.84197
2	0.79590	0.81428	0.85890	0.90394
3	0.87004	0.87906	0.91278	0.94914
4	0.80762	0.84094	0.89059	0.92427
5	0.72566	0.73864	0.83581	0.90099
6	0.70539	0.71484	0.79230	0.87059
7	0.86743	0.90206	0.94247	0.95980
8	0.65560	0.67900	0.76580	0.85706
9	0.83627	0.86859	0.90666	0.93339
10	0.83315	0.86188	0.90406	0.93675
11	0.74118	0.76592	0.82987	0.88674
12	0.82371	0.84639	0.88648	0.92625
13	0.59585	0.59972	0.71389	0.80057
14	0.72194	0.75127	0.83074	0.89096
15	0.82369	0.85191	0.89279	0.92652

16	0.76132	0.77937	0.83373	0.89368
17	0.82876	0.86580	0.90952	0.93639
18	0.72756	0.75362	0.83588	0.88924
19	0.75334	0.77523	0.83790	0.88869
20	0.84866	0.86896	0.90448	0.93095
21	0.77811	0.80544	0.86329	0.90303
22	0.76066	0.78455	0.85080	0.89521
23	0.89887	0.92311	0.94744	0.95827
24	0.73919	0.76242	0.83908	0.88558
Average	**0.77314**	**0.79512**	**0.85557**	**0.90375**

5. Conclusion

In this paper, a least squares filter-based upsampling method which populates missing pixel at the zero-padded images was studied. Several adaptive upsampling methods have been introduced and quantitatively assessed with the proposed approach using 24 Kodak standard test images. In addition, visual performance is evaluated by assessing image artefacts. Visual and objective results proved the superiority of the proposed method. The proposed kernel outperformed conventional methods in terms of MSE, PSNR, SSIM, and visual aspect. Particularly, the proposed kernel outperformed bicubic method with 2.169 dB for 24 Kodak images.

6. Discussion

This paper proposed kernel design method which minimizes the mean-squared error over a training image. The proposed kernel is designed by least-squares method by adopting training set for the desired upsampling performance. However, the conventional interpolation method used fixed interpolation kernels for all the images, our method learned several interpolation kernels for each image structure in the training process. And in the reconstruction step, we used a weighted average of these kernels to build an adaptive interpolation kernel for each specific image. Experimental results on high resolution demonstrate that the proposed algorithm can preserve sharp edges and rich textures. In the future work, we will continue to study kernel efficiency and kernel size and performance trade-off.

Acknowledgement

This research was supported by 2018-1 Hanyang Women's University Research Fund.

Present Address

31, Achasan-ro 76-gil, Gwangjin-gu, Seoul, 04969, Republic of Korea.

References

[1] K. Jack, Video Demystified - A Handbook for the Digital Engineer, 4th ed., Elsevier, Jordan Hill, Oxford, (2005).
[2] B. Bhanu, J. Peng, T. Huang, and B. Draper, "Introduction to the special issue on learning in computer vision and pattern recognition", IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 35, no. 3, (2005), pp. 391-396.
[3] E. Maeland, "On the comparison of interpolation methods", IEEE Trans. Med. Imag., vol. 7, no. 3, (1988), pp. 213-217.
[4] J.A Parker, R.V Kenyon, and D.E Troxel, "Comparison of interpolating methods for image resampling", IEEE Trans. Med. Imag., vol. 2, no. 1, (1983), pp. 31-39.
[5] D. Bhattacharyya, A. Roy, P. Roy and T.-h. Kim, "Receiver Compatible Data Hiding in Color Image", International Journal of Advanced Science and Technology, (2009), pp. 15-24.
[6] B.V. Ramana Reddy, A. Suresh, M. Radhika Mani and V.Vijaya Kumar, "Classification of Textures Based on Features Extracted from Preprocessing Images on Random Windows", International Journal of Advanced Science and Technology, (2009), pp. 9-18.
[7] S. Haykin, Adaptive Filter Theory, (2002), Prentice Hall.
[8] D. Hwang and K. Lee, "Robust skin area detection method in color distorted images", Journal of the Korea Academia-Industrial cooperation Society, vol. 18, no. 7, (2017), pp. 350-356.
[9] W. Kim and S. Hong, "2D image numerical correction method for 2D digital image correlation", Journal of the Korea Academia-Industrial cooperation Society, vol. 18, no. 4, (2017), pp. 391-397.
[10] D. Kim, "The adopting C4.5 classification and its application for deinterlacing", Journal of the Korea Academia-Industrial cooperation Society, vol. 18, no. 1, (2017), pp. 8-14.
[11] J. S. Lim, T. T. Jeong, and J. H. Lee, "A study on image region analysis and image enhancement using detail descriptor", Journal of the Korea Academia-Industrial cooperation Society, vol. 18, no. 6, (2017), pp. 728-735.
[12] J. S. Lim, "A study on non-local image denoising method based on noise estimation", Journal of the Korea Academia-Industrial cooperation Society, vol. 18, no. 5, (2017), pp. 518-523.
[13] F. Piccialli, S. Cuomo, and P. De Michele, "A regularized MRI image reconstruction based on hessian penalty term on CPU/GPU systems", Procedia Computer Science, vol. 18, (2013), pp. 2643-2646.
[14] R. Farina, S. Cuomo, P. De Michele, and F. Piccialli, "A smart GPU implementation of an elliptic kernel for an ocean global circulation model", Applied Mathematical Sciences, vol. 7, no. 61-64, (2013), pp. 3007-3021.
[15] A. Paul, "Real-time power management for embedded M2M using intelligent learning methods", ACM Trans. Embedded Computing Systems, vol. 13, no. 5s, article no. 148, (2014).
[16] A. Paul, S. Rho, and K. Bharanitharan, "Interactive Scheduling for Mobile Multimedia Service in M2M environment", Multimedia Tools and Application, vol. 71, no. 1, (2014), pp. 235-246.
[17] A. Paul, Y.-C. Jiang, J.-F. Wang and J.-F. Yang, "Parallel reconfigurable computing based mapping algorithm for motion estimation in advance video coding", ACM Trans. Embedded Computing Systems, vol. 11, no. S2, article no. 40, (2012).
[18] Available: http://r0k.us/graphics/kodak/
[19] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity", IEEE Trans. Image Processing, vol. 13, no. 4, (2004), pp. 600-612.