

Image Enhancement in Embedded Devices for Internet of Things

Gwanggil Jeon¹, Kitsuchart Pasupa^{2,*}, Marco Anisetti³, and Awais Ahmad⁴

¹Department of Embedded Systems Engineering, Incheon National University, Incheon 22012, Korea

²Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand

³Dipartimento di Informatica (DI), Università degli Studi di Milano, Via Celoria 18, Milano (MI) 20133, Italy

⁴Department of Computer Science, Bahria University, Islamabad, Pakistan

gjeon@inu.ac.kr, *kitsuchart@it.kmitl.ac.th, marco.anisetti@unimi.it, aahmad.marwat@gmail.com

Abstract

This paper proposes a new color interpolation method which can be used in embedded devices for IoT system. In this work, we use regression approach for generating and designing filters to restore color image. The filters are designed with four sizes, 5x5 training filter, 7x7 training filter, 9x9 training filter, and 11x11 training filter. The obtained filters are tested in 25 LC dataset to assess the performance. Experimental results inform that the proposed filters provide outstanding performance when they are compared with conventional methods. As compared with the other methods, the proposed filters produce the best average interpolation performance both objectively and visually.

Keywords: *Regression method, embedded device, IoT, filter design, Bayer pattern, demosaicking.*

1. Introduction

The embedded system consumes low power which is designed to perform unique tasks such as sensors, simple calculator, and remote controls [1,2]. In general, embedded apparatus is a function of a bigger device where it executes particular mission of the device. The embedded system is also used in various forms of Internet of Things (IoT) solutions, particularly for industrial purposes [3-6]. The IoT is one of

well-studied research topics due to the rapid development in information and communication techniques. The IoT is the spreading of internet extension into embedded devices. The electronics with IoT are able to transfer data and act reciprocally with others over the network, and they can be distantly supervised. Possible embedded devices for IoT can include wireless sensor networks, control systems, automation, and video surveillance. In this work, we study image enhancement method in embedded devices for IoT.

The IoT system concentrates upon human visual system which is known as perfect solution for the viewers. The color filter array (CFA) is a design made with small tiles of color filters which is located before the image sensor. In a commercial manner, the most frequently used CFA formation is the Bayer CFA (Fig. 1) [7]. This CFA mosaic arranges RGB color filters on a square grid of photo sensors. Currently, this order of CFA is widely accepted in most single chip digital image sensors, especially for digital video devices to reproduce a color image from CFA. It has 2×2 sub-block, which has the diagonal sensing pieces, one red (R) pixel, two green (G) pixels, and one blue (B) pixel in a pair. There are double G pixels as R or B pixels, considering to the human visual system.

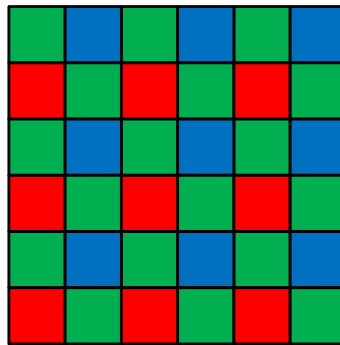


Fig. 1. Bayer pattern CFA example.

As every sensor pixel is located after the CFA, the result of an image is a single channel signal. Therefore, each signal has raw intensity of one of three colors. Thus, we apply a demosaicking method to restore full primary color channels. Here is an example of mosaicking: color channel decomposition and demosaicking. Figure 2(a) shows the original image, while Figs. 2(b), 2(c), and 2(d) show R, G, B color channels. Figure 2(e) shows CFA image. Figure 3 shows demosaicking pipeline. The desired demosaicking method must take advantage of high visual quality images with low complexity. To accomplish this task, many researchers have been studied various demosaicking methods [8-15]. Most existent demosaicking algorithms usually concentrate upon how to precisely discover the minimum variation direction in the mosaicked CFA image. The demosaicking, super-resolution, and denoising require similar task, which have been researched separately last few decades.

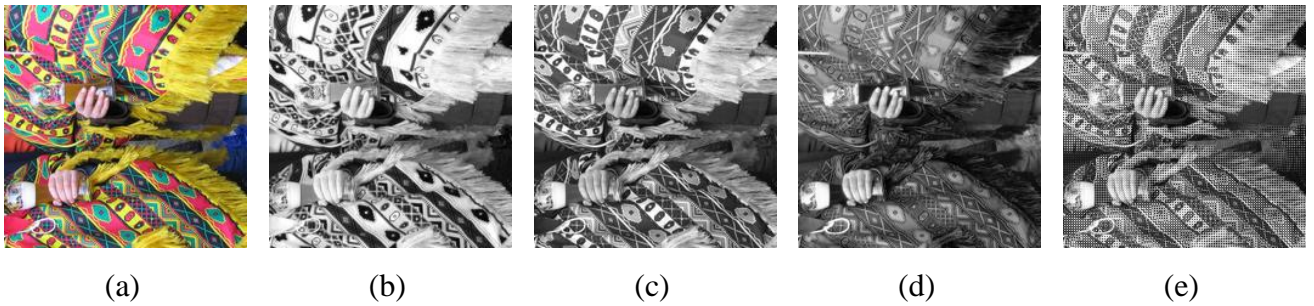


Fig. 2. (a) Original image, (b) red channel image, (c) green channel image, (d) blue channel image, and (e) CFA image.

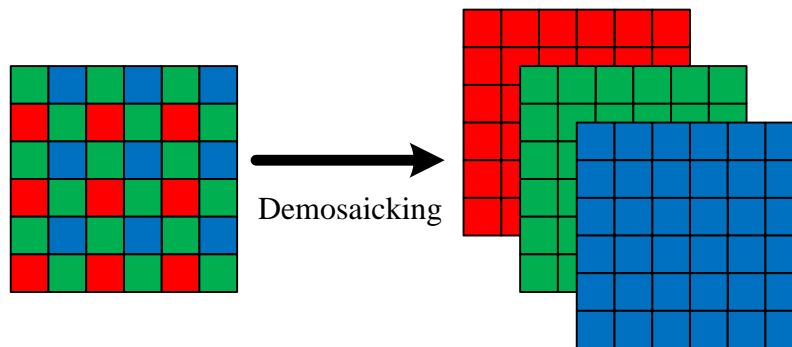


Fig. 3. Demosaicking pipeline.

However, these methods are sub-optimum and normally arrives to difference calculation, such as image artifacts between original and the estimates. In other words, color error (orange or purple color artifact) effects are considered missing information of super resolution or denoising methods [16-18]. Images can be classified with flat region and detail region. The region with details near to the resolution restraint of the digital sensor may cause of an artifact issue to the restored image. This effect is called Moiré effect, where reiterated patterns are shown. Therefore, restored images do not look practical. One of unwanted artifacts in color demosaicking is that false coloring [19-21] artifact. In general, false coloring artifact reveals in detail region with an edge area, where sudden color shifts are shown as a result of wrong restoration. Another well-known demosaicking artifact is zippering effect. This effect is yet another negative effect of demosaicking, often shown near edge region. The zippering artifact normally takes places when the interpolation approach returns the average result over an edge, particularly in the red and blue channels, returning edge blur effect.

In this paper, we propose a new image interpolation method which can be used in embedded devices for IoT system. This paper is organized as follows. In Section 2, a filter-based image processing is introduced. Then, the proposed method is explained in Section 3. Simulation results respect to traditional methods by objective (circular peak signal-to-noise ratio, circular mean square error, S-CIELAB) and subjective metrics (visual quality, feature similarity index) are explained in Section 4. Finally, conclusion remarks are provided in Section 5.

2. Filter-based Image Processing

Image filtering is a tool for improving a given low quality image. For instance, one can use a filter to an image to enhance specific characteristic or quality or can decrease other effects. Image filters are mostly adopted to control either the low frequency or high frequency in an image to make specific effects. The filtering approaches are conducted in spatial domain or in frequency domain. The image filter is also known as a kernel, which looks a convolution matrix or image mask. In general, image mask is with a form of small matrix. Equation (1) shows the general convolution expression.

$$\begin{aligned} f_R(i, j) &= h * f_O(i, j) \\ &= \sum_{u=-r_u}^{r_u} \sum_{v=-r_v}^{r_v} h(u, v) f_O(i-u, j-v), \end{aligned} \quad (1)$$

where $f_R(i, j)$ is the filtered result image, $f_O(i, j)$ is the original image, and h is the filter kernel. Kernels are normally small matrix of numbers. The range of all components of the filter kernel is assumed as $-r_u \leq i \leq r_u$ in horizontal direction and $-r_v \leq j \leq r_v$ in vertical direction, respectively. Conditional upon the component values, a kernel can execute a large spectrum of effects.

A convolution procedure, which is denoted by $*$, impacts each component of the image to its regionally adjacent pixels regard by the value of kernel. The convolution is conducted by multiplying pixel's and its neighboring pixels value by a matrix. For instance, if we have two images and each image size are $(2n+1)$ -by- $(2n+1)$ pixels, then the first image is assumed as kernel and the second image is considered as image to be filtered. To perform convolution, the kernel is firstly overturned both the rows and columns directions, then multiplying locally comparable elements and adding. The component at a center position (n, n) of the filtered image is the weighted average of all elements of the image matrix.

One of primary goals of the image filter is edge preserving smoothing. This is a representative effect that removes irregularities or roughness noise or textures while keeping sharp details. There are many image filters such as the bilateral filter, median filter, and guided filter. For instance, image sharpening is needed for medical imaging or satellite imaging, where the details are primary depicts and therefore sharpness should be conserved, and the original image must not be distorted. Recently proposed new filter method is guided filter. This filter is derived from a local linear model, which creates the filtering result in light of the content of a guidance image, which is either input image or other different image. The guided filter can act as an edge-preserving smoothing filter such as the bilateral filter. Figure 4 shows frequency response of three representative filters: identical filter, low pass filter, and high pass filter.

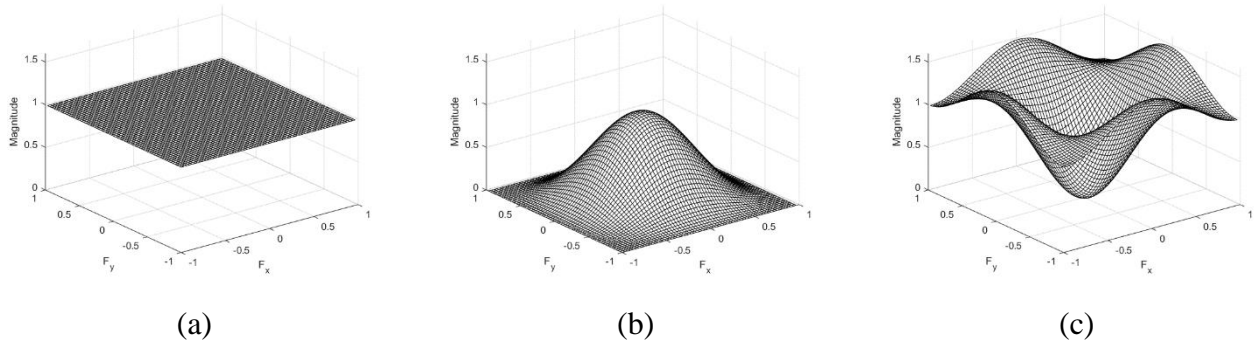


Fig. 4. Frequency response of three representative filters: (a) identical filter, (b) low pass filter, and (c) high pass filter.

3. Proposed Method

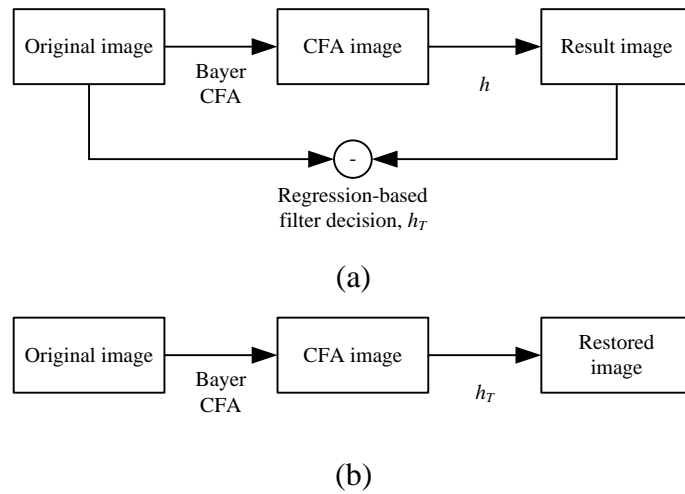


Fig. 5. (a) Block diagram of the training procedure, (b) block diagram of the demosaicking procedure.

Two block diagrams are shown in Fig. 5. The first figure shows the training process, and the second figure shows the color interpolation process. The regression method is a model to information which is one of frequently used subjects of a large part of statistical mission in image processing. The least squares method plays an important role due to its modest characteristics, and this method is comparatively simple to calculate, but still it works well under the supposition of random data. In this work, we use both of least squares regression and quantile regression for generating filters to restore color image. The quantile regression is a regression method which is used in statistics. In contrast to the least squares regression, quantile regression tries to compute conditional median of the response variable. This point is different from the least squares regression where conditional mean is computed. Thus, the quantile regression is the branch of linear regression and in this work, we use both of least squares and quantile regression for color demosaicking.

The least squares regression model is a linear model when it includes a linear union of the parameters. The basic form of least squares regression is,

$$\min_x \|Cx - d\|^2, \quad (2)$$

where we explain $\|x\|^2 = \sum_i x_i^2 = x^T x$ is the normal Euclidean norm and x^T is the x 's transpose. This problem has a singular answer given that C is independent columns, and therefore, $C^T C$ can be invertible. Then,

$$\begin{aligned} \nabla \|Cx - d\|^2 &= \nabla (Cx - d)^T (Cx - d) \\ &= 2C^T (Cx - d). \end{aligned} \quad (3)$$

After equating Eq. (3) to zero, it can be rearranged as,

$$C^T Cx = C^T d. \quad (4)$$

This normal formula has a singular answer for x as $C^T C$ is invertible. Then, we can find that the least squares estimate x is calculated by

$$x = (C^T C)^{-1} C^T d. \quad (5)$$

This formula can be further studied and applied in demosaicking. The role of least squares method is to minimize difference between the original data and the restored data. The filter coefficients are computed during training process by least squares methods. Let us consider three images: original image (denoted as im_O), low resolution CFA image (denoted as im_{CFA}), and the restored image (denoted as im_R). It is noted that image size of im_{CFA} is smaller than other images, and also it has only one color channel. Therefore, each pixel came from one of three color components. Then the result image im_R can be computed by,

$$im_R = h_T * im_{CFA}, \quad (6)$$

where h_T is the filter we obtained during training process. Then, filter h_T can be computed as

$$h_T = \arg \min_h \Pi(im_O - im_R) \quad (7)$$

where h is anonymous candidate filter, parameter $\Pi(im_O - im_R)$ is the cost function that calculate difference between im_O and im_R .

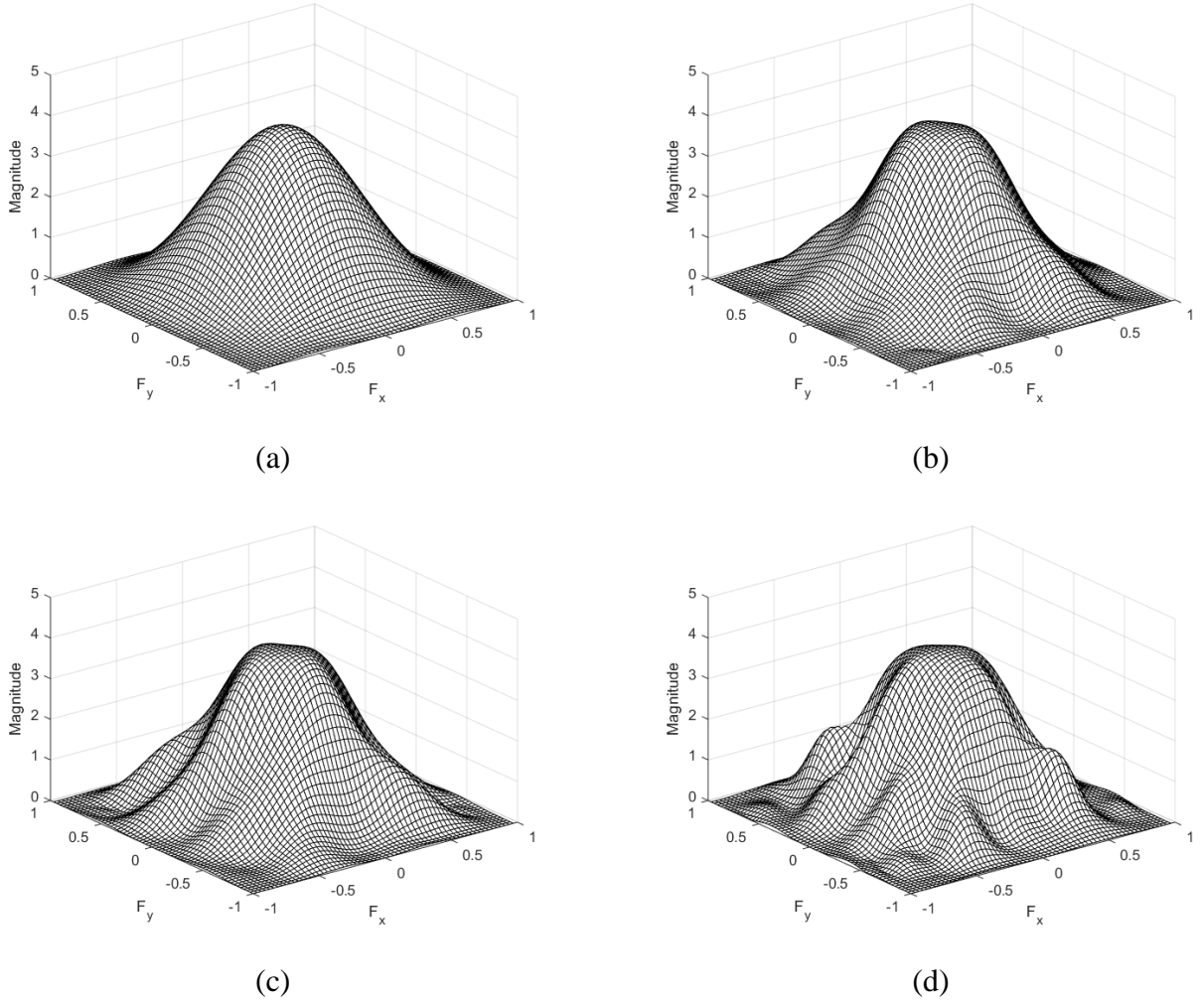


Fig. 6. Frequency responses of the four trained filters: (a) $h_{T(5 \times 5)}$, (b) $h_{T(7 \times 7)}$, (c) $h_{T(9 \times 9)}$, and (d) $h_{T(11 \times 11)}$.

In this work, we trained the filters with four scenarios: 5×5 training filter, 7×7 training filter, 9×9 training filter, and 11×11 training filter. In literature, larger filter returns better image quality while complexity becomes higher. Figures 6 and 7 show frequency responses of four trained filters ($h_{T(5 \times 5)}$, $h_{T(7 \times 7)}$, $h_{T(9 \times 9)}$, $h_{T(11 \times 11)}$) and their corresponding coefficient values. Figure 8(a) shows frequency response of $h_{T(21 \times 21)}$. Figures 8(b-d) show frequency response difference between $h_{T(21 \times 21)}$ and $h_{T(5 \times 5)}$, $h_{T(7 \times 7)}$, $h_{T(9 \times 9)}$, $h_{T(11 \times 11)}$. To display Figs. 6 and 8, we used a MATLAB command $F = \text{freqz2}(h_{2D}, r, c)$, where h_{2D} is a two-dimensional spatial filter and F is its two-dimensional corresponding frequency domain filter. Parameters r and c are row number and columns number, respectively. For instance, $h_{T(5 \times 5)}$ is obtained when $r=5$ and $c=5$. In general, filtering in spatial domain requires less computations than that of filtering in frequency domain. However, as filter size increases, due to the increment of multiplication and addition, filtering in frequency domain becomes more efficient. The frequency response is the measurable capacity of the spectrum of a result image in reaction to a stimulus, and is utilized to describe the activities of the system. It has two measurements: magnitude and phase of the image as a function of frequency, in

comparison to the input image. Figure 6(a) shows frequency response of $h_{T(5 \times 5)}$. The magnitude of all filters ($h_{T(5 \times 5)}$, $h_{T(7 \times 7)}$, $h_{T(9 \times 9)}$, $h_{T(11 \times 11)}$) is 4.0 when $(u,v)=(0,0)$. This implies that the filters return result images with four times brighter than the given CFA image. However, when $(u,v) \neq (0,0)$, frequency responses are different in each filters. For example, frequency response of $h_{T(11 \times 11)}$ contains more low-frequency information than that of $h_{T(5 \times 5)}$, while it contains less high-frequency information. In addition, it can be found that the frequency response of $h_{T(11 \times 11)}$ has more details out of all trained filters. This implies that training with larger filter can investigate detailed frequency response.

-0.0000	-0.0047	0.0000	0.0036	-0.0000
0.0100	0.2561	0.4861	0.2410	0.0009
-0.0000	0.4879	1.0000	0.5122	-0.0000
0.0059	0.2188	0.4843	0.2867	0.0141
0.0000	-0.0138	-0.0000	0.0160	0.0000

(a)

-0.0262	-0.0359	-0.0002	-0.0039	-0.0100	-0.0268	-0.0258
0.0025	-0.0000	0.0017	0.0000	0.0193	-0.0000	-0.0109
0.0047	0.0403	0.2812	0.4938	0.2698	0.0287	0.0025
-0.0289	-0.0000	0.5062	1.0000	0.5365	-0.0000	-0.0388
0.0141	0.0356	0.2475	0.4893	0.3078	0.0437	-0.0010
-0.0125	0.0000	0.0011	-0.0000	0.0285	0.0000	-0.0046
-0.0361	-0.0322	-0.0155	0.0043	0.0187	-0.0375	-0.0320

(b)

0.0000	-0.0284	-0.0000	0.0328	-0.0000	-0.0039	-0.0000	-0.0046	0.0000
0.0167	-0.0262	-0.0456	-0.0002	-0.0045	-0.0100	-0.0283	-0.0258	0.0013
-0.0000	0.0159	0.0000	-0.0110	0.0000	0.0187	0.0000	-0.0079	-0.0000
-0.0042	0.0047	0.0416	0.2812	0.4937	0.2698	0.0331	0.0025	-0.0077
0.0000	-0.0241	-0.0000	0.5034	1.0000	0.5315	-0.0000	-0.0341	0.0000
-0.0056	0.0141	0.0389	0.2475	0.4900	0.3078	0.0425	-0.0010	0.0008
-0.0000	-0.0110	0.0000	0.0052	-0.0000	0.0102	0.0000	0.0078	-0.0000
-0.0042	-0.0361	-0.0298	-0.0155	0.0042	0.0187	-0.0425	-0.0320	0.0090
0.0000	-0.0017	-0.0000	-0.0147	0.0000	0.0441	-0.0000	-0.0282	0.0000

(c)

-0.0065	-0.0115	-0.0159	0.0145	0.0315	0.0026	-0.0025	0.0033	0.0097	0.0046	-0.0101
0.0141	0.0000	-0.0346	-0.0000	0.0325	-0.0000	-0.0044	0.0000	-0.0019	0.0000	-0.0082
0.0206	0.0202	-0.0326	-0.0526	-0.0174	-0.0073	-0.0121	-0.0325	-0.0445	-0.0036	0.0324
-0.0129	-0.0000	0.0198	0.0000	-0.0107	0.0000	0.0190	-0.0000	-0.0139	-0.0000	0.0130
-0.0027	-0.0033	0.0121	0.0410	0.2828	0.4941	0.2681	0.0341	0.0176	-0.0073	-0.0357
-0.0024	0.0000	-0.0218	0.0000	0.5032	1.0000	0.5308	-0.0000	-0.0337	0.0000	0.0017
-0.0272	-0.0052	0.0234	0.0392	0.2477	0.4906	0.3074	0.0415	0.0031	0.0010	0.0037
0.0103	-0.0000	-0.0154	0.0000	0.0054	-0.0000	0.0109	0.0000	0.0117	-0.0000	-0.0124
0.0295	-0.0089	-0.0541	-0.0300	-0.0064	0.0047	-0.0020	-0.0457	-0.0349	0.0101	0.0147
-0.0069	0.0000	0.0006	-0.0000	-0.0149	0.0000	0.0443	-0.0000	-0.0307	0.0000	0.0074
-0.0117	0.0085	0.0176	-0.0027	-0.0256	-0.0037	0.0408	0.0074	-0.0178	-0.0035	-0.0032

(d)

Fig. 7. Filter coefficients of the four trained filters: (a) $h_{T(5 \times 5)}$, (b) $h_{T(7 \times 7)}$, (c) $h_{T(9 \times 9)}$, and (d) $h_{T(11 \times 11)}$.

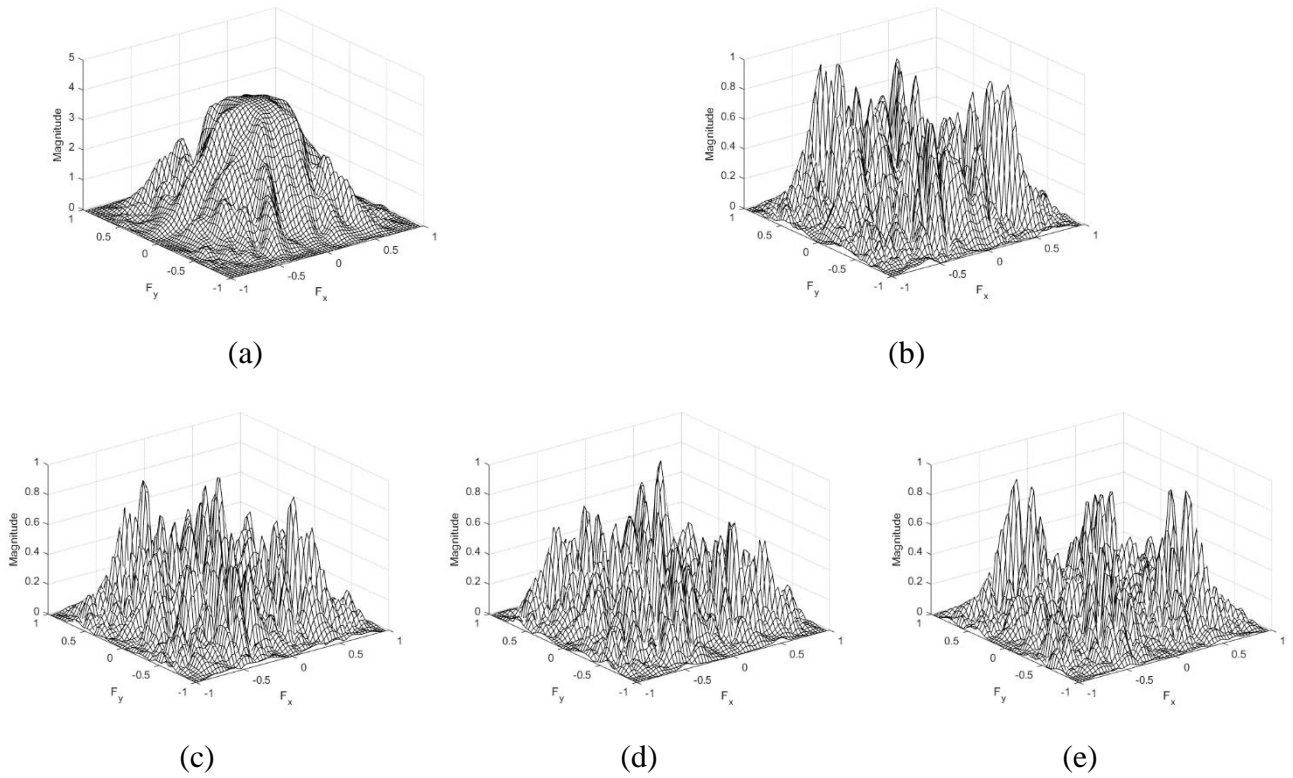


Fig. 8. (a) Frequency response of $h_{T(21 \times 21)}$, (a) frequency response difference between $h_{T(5 \times 5)}$ and $h_{T(21 \times 21)}$, (b) frequency response difference between $h_{T(7 \times 7)}$ and $h_{T(21 \times 21)}$, (c) frequency response difference between $h_{T(9 \times 9)}$ and $h_{T(21 \times 21)}$, (d) frequency response difference between $h_{T(11 \times 11)}$ and $h_{T(21 \times 21)}$.

4. Simulation Results

Experiments were carried out to assess the performance of the different methods. The peak signal-to-noise ratio (PSNR) was utilized as a weigh to assess the performance of the interpolation approaches.



Fig. 9. Test LC dataset: (a) color images, (b) luminance images.

In this Section, we use 25 selected images from LC dataset [22] for evaluating the PSNR of each approach. These images have a resolution of 720x540 as shown in Figure 9: Fig. 9(a) shows color image and Fig. 9(b) shows luminance image. The benchmark methods are tested on luminance channel of each images. The proposed approach with different filter sizes are compared with NN (nearest neighbor), BI (bilinear), and BC (bicubic) methods. The PSNR metric is used to compute the quality of the restored images. PSNR and MSE formulas are shown in Eqs. (8) and (9),

$$PSNR(dB) = 10 \log_{10} \left\{ \frac{P^2}{MSE} \right\}, \quad (8)$$

$$MSE = \frac{1}{RC} \sum_{r=1}^R \sum_{c=1}^C \{im_O(r, c) - im_R(r, c)\}^2, \quad (9)$$

where parameters R and C are row and column sizes of images, and the parameter P is peak intensity vale: 255. Parameters im_O and im_R are original and restored images, and parameters r and c are pixel positions in row and column direction.

Table 1. The PSNR (dB) of luminance images for the LC dataset.

#	NN	BI	BC	Prop _{5x5}	Prop _{7x7}	Prop _{9x9}	Prop _{11x11}
1	24.5162	26.8147	28.6883	30.5516	30.6746	30.6893	30.7069
2	19.5635	21.4334	23.1805	25.5192	25.6687	25.6716	25.8009
3	22.3868	24.6338	26.3641	27.9451	28.0423	28.0352	28.0336
4	23.1235	24.9644	27.3361	29.7436	30.1142	30.1139	30.1378
5	22.4494	24.7633	26.1862	28.1825	28.1977	28.2253	28.2628
6	21.8849	24.0402	25.6234	27.4888	27.5448	27.5442	27.5443
7	19.6397	21.7184	23.1927	25.0890	25.1337	25.1416	25.1776
8	21.4684	23.6493	25.4840	27.3553	27.5013	27.4922	27.4932
9	19.7429	21.8927	23.5134	25.3257	25.4255	25.4324	25.4357
10	22.7774	25.3999	27.0782	28.8335	28.8981	28.8979	28.9044
11	27.2968	30.5902	33.8354	36.0015	36.7899	36.7907	36.8481
12	21.1630	23.2548	25.1128	26.9718	27.1221	27.1400	27.1560
13	21.9439	24.3830	26.2881	28.0971	28.2554	28.2530	28.2596
14	18.5101	20.2997	22.2045	24.4239	24.6962	24.7660	24.8312
15	28.0765	30.8183	32.9797	34.9221	35.2286	35.2036	35.2347
16	27.5039	30.9485	33.7741	35.3642	35.8417	35.8267	35.8380
17	23.5153	26.2174	28.4555	30.0976	30.3696	30.3542	30.3681

18	21.6008	23.6969	25.4007	27.4296	27.5105	<i>27.5123</i>	27.5234
19	26.5491	28.6233	30.8788	32.7353	33.0488	<i>33.0372</i>	<i>33.0425</i>
20	31.0519	34.5164	37.9329	39.7979	<i>40.5964</i>	<i>40.5955</i>	40.6617
21	23.5665	25.9684	27.6972	29.3772	29.4781	<i>29.5533</i>	29.5938
22	18.6425	20.8644	22.4046	23.9807	24.0403	<i>24.0274</i>	<i>24.0322</i>
23	30.5505	33.3685	36.0685	37.6621	38.1072	<i>38.1013</i>	<i>38.1022</i>
24	22.8805	24.8144	26.1558	27.7181	27.7234	<i>27.7222</i>	<i>27.7221</i>
25	23.4107	25.4521	26.5695	28.2982	<i>28.3310</i>	<i>28.3295</i>	28.3334
Avg.	23.3526	25.7251	27.6962	29.5565	29.7736	<i>29.7783</i>	29.8018

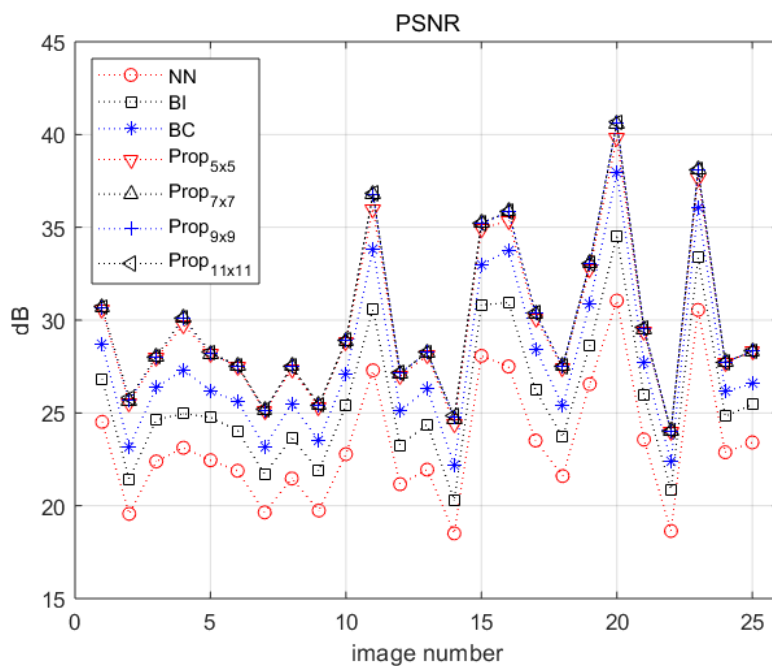


Fig. 10. Plot of PSNR results for seven approaches.

Table 1 shows the PSNR values of different methods to luminance channel. The reason we only investigated gray channel is that, luminance information of YCbCr is the most important channel for human visual system. The numbers with the highest PSNR are displayed in **bold**. In addition, the numbers with the second highest PSNR are displayed in *italic*. From Table 1, it can be found that the bigger filters provide better performance. For example, average PSNR result of $h_{T(11 \times 11)}$ was 29.8018 which was the best filter, followed by $h_{T(9 \times 9)}$ with 29.7783. The margins between the best filter and the rest filters are, 6.4492 (NN), 4.0767 (BI), 2.1056 (BC), 0.2453 ($h_{T(5 \times 5)}$), 0.0282 ($h_{T(7 \times 7)}$), and 0.0235 ($h_{T(9 \times 9)}$). The $h_{T(11 \times 11)}$ filter showed the best performance for most images, however for some images (nine images: #3, #6, #8, #16, #17, #19, #22, #23, #24), $h_{T(7 \times 7)}$ showed the best performance. It is noted that $h_{T(9 \times 9)}$ did not show the best performance for any of images, however in average performance-wise, $h_{T(9 \times 9)}$ outperformed

$h_{T(7 \times 7)}$. The proposed method produced the best average PSNR performance among the tested methods. Figure 10 shows the plot of PSNR results for seven approaches.

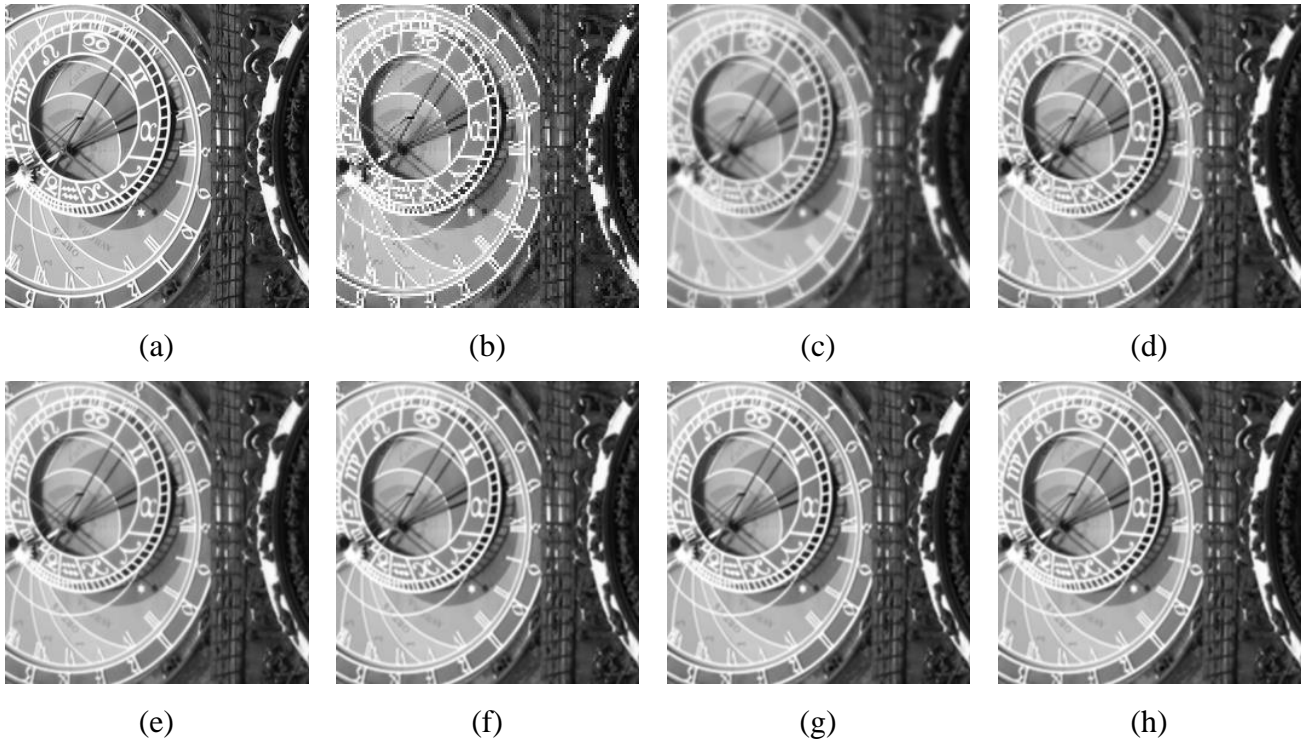


Fig. 11. #08 LC image: (a) Zoomed original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5 \times 5)}$, (f) result with $h_{T(7 \times 7)}$, (g) result with $h_{T(9 \times 9)}$, and (h) result with $h_{T(11 \times 11)}$.

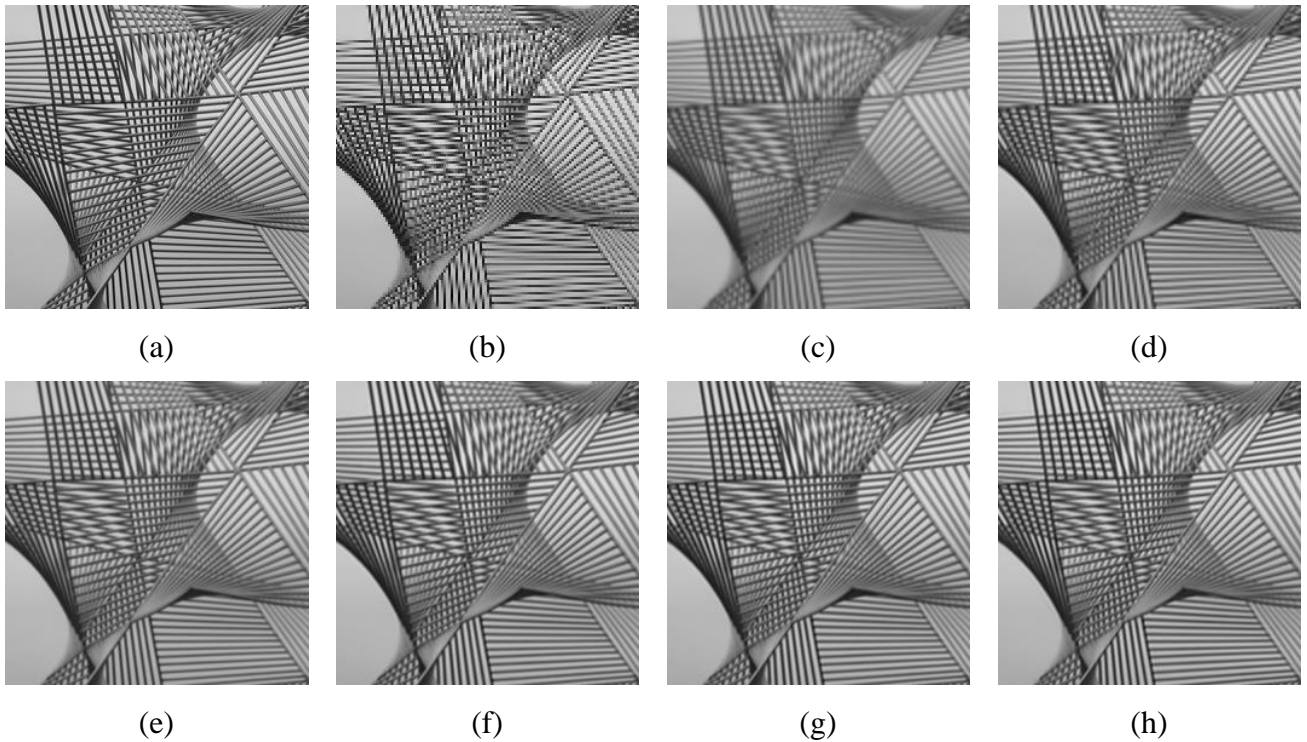


Fig. 12. #14 LC image: (a) Zoomed original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5 \times 5)}$, (f) result with $h_{T(7 \times 7)}$, (g) result with $h_{T(9 \times 9)}$, and (h) result with $h_{T(11 \times 11)}$.

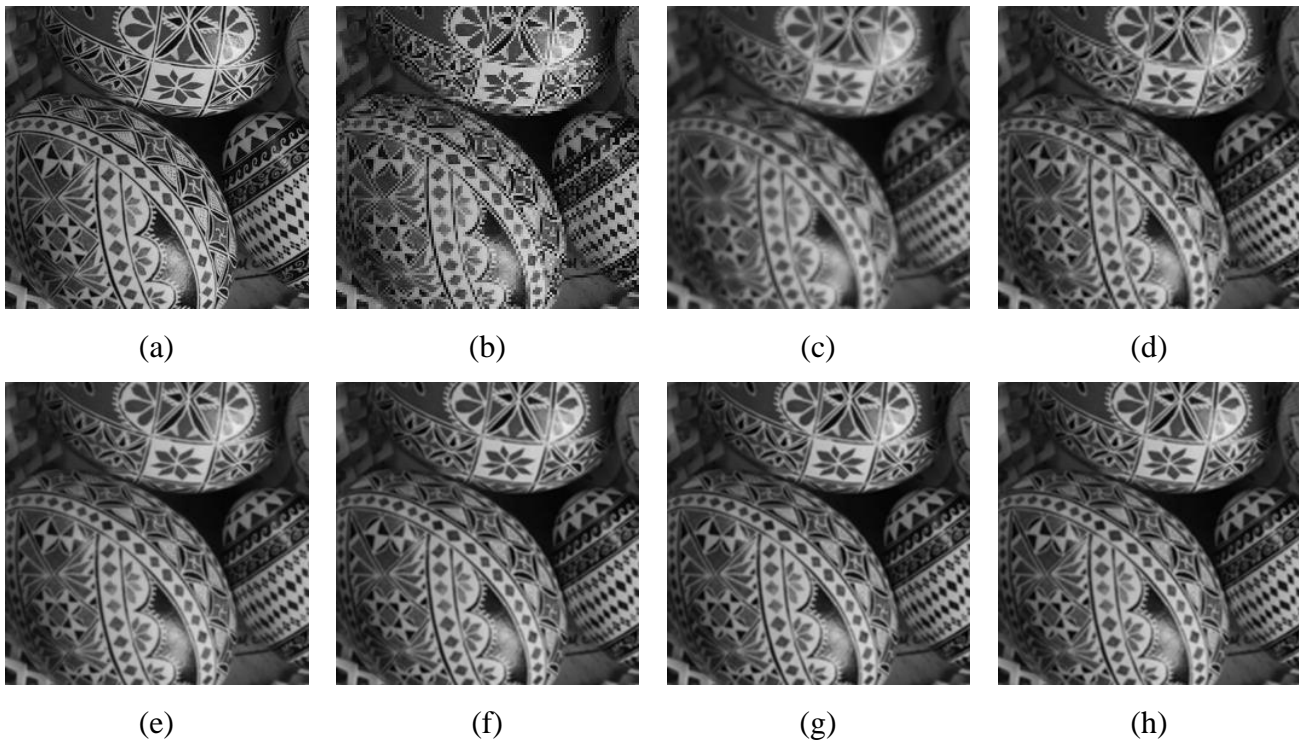


Fig. 13. #18 LC image: (a) Zoomed original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5 \times 5)}$, (f) result with $h_{T(7 \times 7)}$, (g) result with $h_{T(9 \times 9)}$, and (h) result with $h_{T(11 \times 11)}$.

It is known that the PSNR results are not the best way to evaluate the performance of the proposed method. Therefore, we used subjective quality comparison in each image. It was found that the proposed method provided fine detail patterns well. Figures 11-13 show the visual comparison of the restored images. Figure 11(a) shows original image of #08 LC image. Seven benchmark images are displayed in Figs. 11(b-h), they are NN result, BI result, BC result, result with $h_{T(5 \times 5)}$, result with $h_{T(7 \times 7)}$, result with $h_{T(9 \times 9)}$, and result with $h_{T(11 \times 11)}$. It can be found that the proposed method can conserve fine texture details and, therefore, yields less visual artifacts. Figures 12(a) and 13(a) show original images of #14 and #18 LC images. Their tested demosaicking results are shown in Figs. 12(b-h) and 13(b-h), respectively. In all test images, the proposed $h_{T(11 \times 11)}$ filter provided the best visual quality.

Figures 14-16 show difference images between the original image and the restored images. Figure 14(a) shows original image of #08 LC image, while Figs. 14(b-h) show difference images NN result, BI result, BC result, result with $h_{T(5 \times 5)}$, result with $h_{T(7 \times 7)}$, result with $h_{T(9 \times 9)}$, and result with $h_{T(11 \times 11)}$. We assume that the brighter the images imply color artifacts. One can find that the difference image of $h_{T(11 \times 11)}$ is the least, therefore we can conclude $h_{T(11 \times 11)}$ provides the least errors (Figs. 14h-16h).

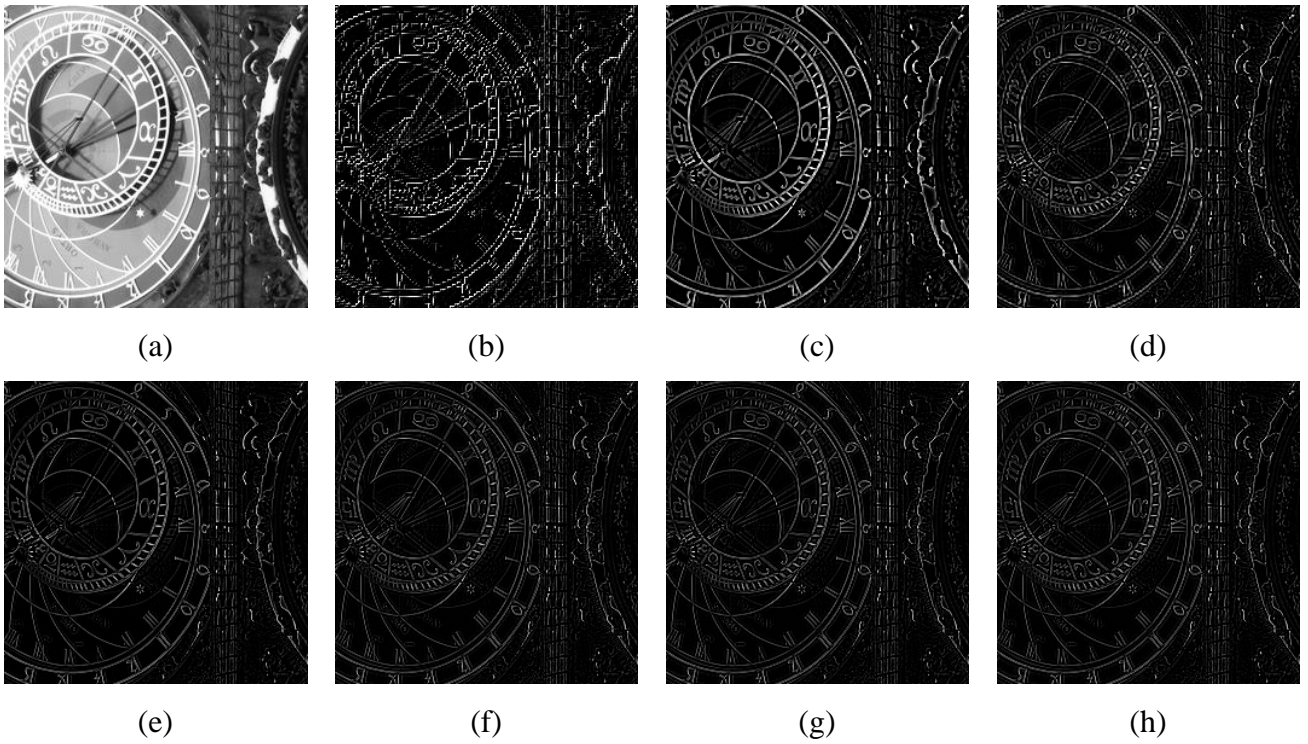


Fig. 14. Difference from original image in #08 LC: (a) Original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5x5)}$, (f) result with $h_{T(7x7)}$, (g) result with $h_{T(9x9)}$, and (h) result with $h_{T(11x11)}$.

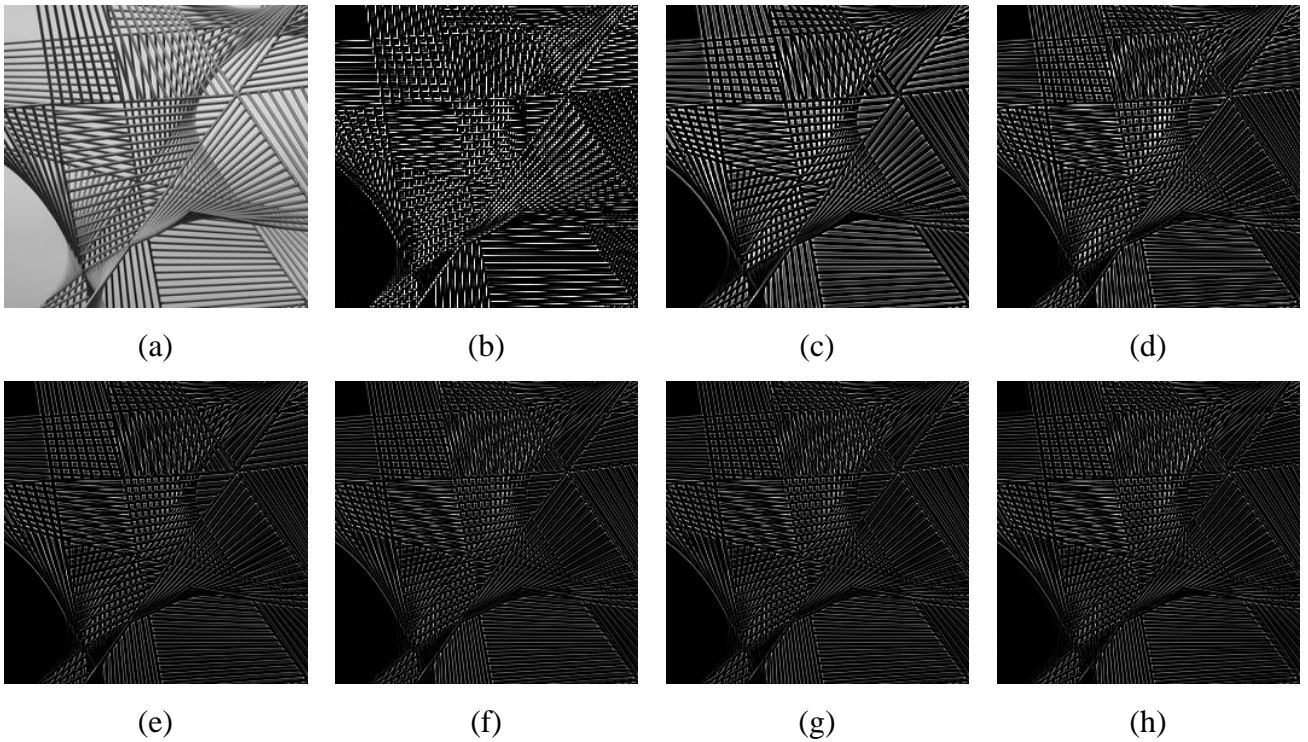


Fig. 15. Difference from original image in #14 LC: (a) Original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5x5)}$, (f) result with $h_{T(7x7)}$, (g) result with $h_{T(9x9)}$, and (h) result with $h_{T(11x11)}$.

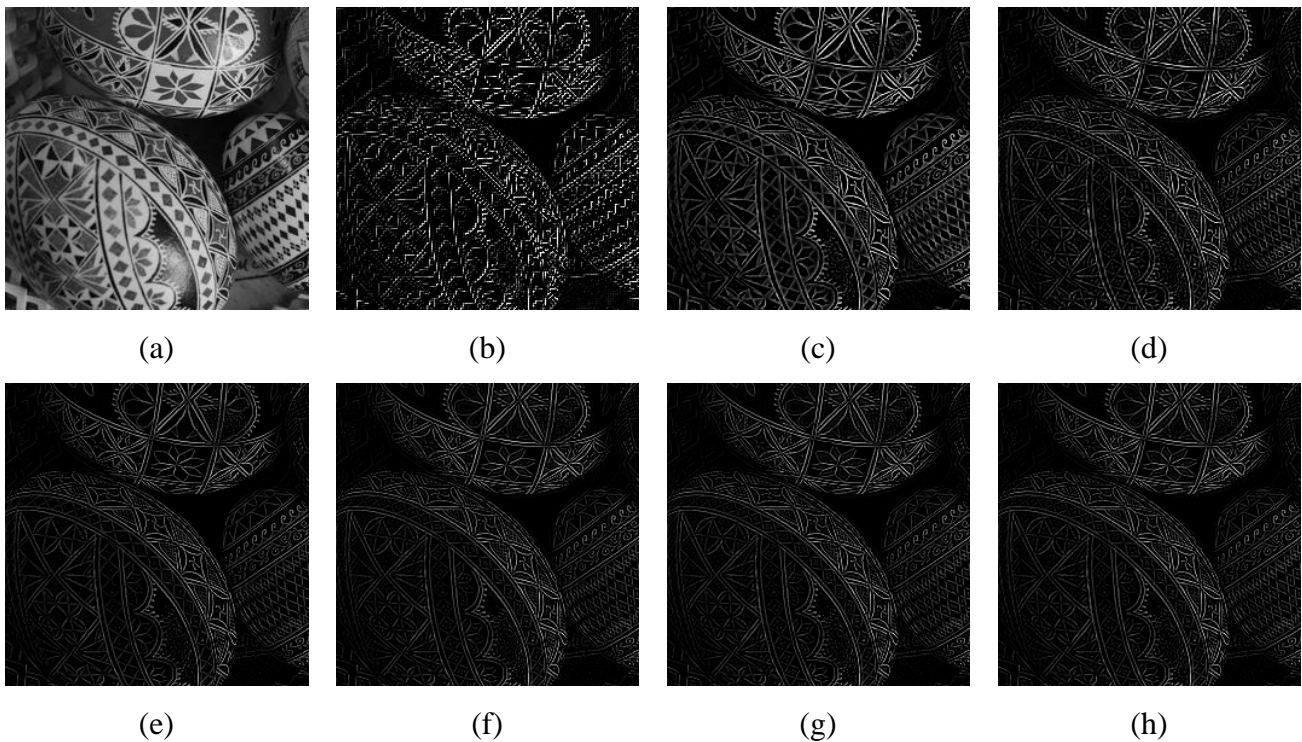


Fig. 16. Difference from original image in #18 LC: (a) Original image, (b) NN result, (c) BI result, (d) BC result, (e) result with $h_{T(5 \times 5)}$, (f) result with $h_{T(7 \times 7)}$, (g) result with $h_{T(9 \times 9)}$, and (h) result with $h_{T(11 \times 11)}$.

5. Conclusions

In this paper, a new color interpolation method was proposed. We used regression approach for designing filters to reconstruct full color image. We generated four-size filters, 5x5 training filter, 7x7 training filter, 9x9 training filter, and 11x11 training filter. These filters are tested in 25 LC dataset to evaluate the performance. Simulation results proved that the presented method outperformed conventional methods subjectively and objectively.

Acknowledgement

This work was supported by the Incheon National University (International Cooperative) Research Grant in 2016 and King Mongkut's Institute of Technology Ladkrabang.

References

- [1] G. Jeon, M. Anisetti, J. Lee, V. Bellandi, E. Damiani, and J. Jeong, "Concept of linguistic variable-based fuzzy ensemble approach: application to interlaced HDTV sequences," *IEEE Trans. Fuzzy Systems*, vol. 17, no. 6, pp. 1245-1258, Dec. 2009.
- [2] M. Babar, A. U. Rehman, F. Arif, and G. Jeon, "Energy-harvesting based on internet of things and big data analytics for smart health monitoring," *Sustainable Computing, Informatics and Systems*, vol. 20, pp. 155-164, Dec. 2018.

- [3] 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, pp. 3007-3021, 2013.
- [4] S. Cuomo, P.D. Michele, F. Piccialli, A. Galletti, and J.E. Jung, "IoT-based collaborative reputation system for associating visitors and artworks in a cultural scenario," *Expert Systems with Applications*, pp. 101-111, 2017.
- [5] A. Chianese, F. Marulli, V. Moscato, and F. Piccialli, "A smart multimedia guide for indoor contextual navigation in Cultural Heritage applications," in *Proc. International Conference on Indoor Positioning and Indoor Navigation, IPIN 2013*, 2013.
- [6] A. Chianese and F. Piccialli, "SmaCH: A framework for smart cultural heritage spaces," in *Proc. 10th International Conference on Signal-Image Technology and Internet-Based Systems, SITIS 2014*, pp. 477-484, 2015.
- [7] B. E. Bayer, "Color imaging array", U.S. Patent 3 971 065, July 1976.
- [8] J. Wu, M. Anisetti, W. Wu, E. Damiani, and G. Jeon, "Bayer demosaicking with polynomial interpolation," *IEEE Trans. Image Processing*, vol. 25, no. 11, pp. 5369-5382, Nov. 2016.
- [9] J. Wang, J. Wu, Z. Wu, M. Anisetti, and G. Jeon, "Bayesian method application for color demosaicking," *SPIE Optical Engineering*, vol. 57, no. 5, pp. 053102, 2018.
- [10] G. Jeon, M. Anisetti, L. Wang, and E. Damiani, "Locally estimated heterogeneity property and its fuzzy filter application for scanning format conversion," *Information Sciences*, vol. 354, pp. 112-130, Aug. 2016.
- [11] J. Shi, J. Wu, M. Anisetti, E. Damiani, and G. Jeon, "An interval type-2 fuzzy active contour model for auroral oval segmentation soft computing," *Soft Computing*, vol. 21, no. 9, pp. 2325-2345, May 2017.
- [12] J. Shi, Y. Lei, J. Wu, and G. Jeon, "Uncertain active contour model based on rough and fuzzy sets for auroral oval segmentation," *Information Sciences*, vol. 492, pp. 72-103, Aug. 2019.
- [13] L. Wang and G. Jeon, "Bayer pattern CFA demosaicking based on multi-directional weighted interpolation and guided filter," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2083-2087, Nov. 2015.
- [14] J. Wang, J. Wu, Z. Wu, G. Jeon, and J. Jeong, "Bilateral filtering and directional differentiation for Bayer demosaicking," *IEEE Sensors Journal*, vol. 17, no. 3, pp. 726-734, Feb. 2017.
- [15] J. Wang, J. Wu, Z. Wu, and G. Jeon, "Filter-based Bayer pattern CFA demosaicking," *Circuits, Systems and Signal Processing*, vol. 36, no. 7, pp. 2917-2940, July 2017.
- [16] J. Wu, L. Deng, and G. Jeon, "Image autoregressive interpolation model using GPU-parallel optimization," *IEEE Trans. Industrial Informatics*, vol. 14, no. 2, pp. 426-436, Feb. 2018.
- [17] 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, Nov. 2014.

-
- [18] A. Paul, S. Rho and K. Bharanitharan, "Interactive Scheduling for Mobile Multimedia Service in M2M environment" *Multimedia Tools and Application*, vol. 71, no. 1, pp. 235-246, July 2014.
- [19] A. Chianese, F. Marulli, F. Piccialli, I. Valente, "A novel challenge into multimedia cultural heritage: An integrated approach to support cultural information enrichment," in *Proc. 2013 International Conference on Signal-Image Technology and Internet-Based Systems, SITIS 2013* 6727195, pp. 217-224.
- [20] F. Amato, V. Moscato, A. Picariello, F. Piccialli, "SOS: A multimedia recommender System for Online Social networks," *Future Generation Computer Systems*, DOI: 10.1016/j.future.2017.04.028.
- [21] F. Piccialli and A. Chianese, "The internet of things supporting context-aware computing: a cultural heritage case study," *Mobile Networks and Applications*, vol. 22, no. 2, pp. 332-343, 2017.
- [22] Available: <http://www.gipsa-lab.grenoble-inp.fr/~laurent.condat/imagebase.html>