Demosaicing of Color Filter Array Captured Images Using Gradient Edge Detection Masks and Adaptive Heterogeneity-Projection

Kuo-Liang Chung, *Senior Member, IEEE*, Wei-Jen Yang, Wen-Ming Yan, and Chung-Chou Wang

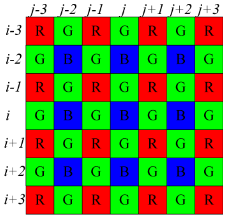
***Abstract*—Without demosaicing processing, this paper first proposes a new approach to extract more accurate gradient/edge information on mosaic images directly. Next, based on spectral-spatial correlation, a novel adaptive heterogeneity-projection with proper mask size for each pixel is presented. Combining the extracted gradient/edge information and the adaptive heterogeneity-projection values, a new edge-sensing demosaicing algorithm is presented. Based on 24 popular testing images, experimental results demonstrated that our proposed high-quality demosaicing algorithm has the best image quality performance when compared with several recently published algorithms.**

***Index Terms*—Adaptive heterogeneity projection, color filter array (CFA), color peak signal-to-noise ratio (CPSNR), demosaicing algorithm, digital cameras, luminance estimation, mosaic images, sobel operator.**

I. Introduction

RECENTLY, digital cameras have become more and more popular in consumer electronics market. In order to economize the hardware cost, instead of using three sensors, most digital cameras capture a color image with a signal sensor imaging pipeline [32] based on the well-known Bayer CFA [3], where each pixel in the captured image has only one measured color and this kind of images is called mosaic images. Fig. 1 depicts the Bayer CFA structure. Because G (green) color channel is the most important factor to determine the luminance of the color image, half of the pixels in Bayer CFA structure are assigned to G channel. R (red) and B (blue) color channels, which share the other half pixels in the Bayer CFA structure, are considered as the chrominance signals.

In order to recover the full color image from the input mosaic image, the demosaicing process is used to estimate the other two color channels for each pixel [15], [20]. Bilinear interpolation (BI) [38] is the simplest demosaicing algorithm in which the unknown two color channels of each pixel are obtained by averaging its proper adjacent pixels. In [8], Cok presented a smooth hue transition-based demosaicing algorithm. In [12], Freeman presented a median filter-based demosaicing algorithm. Based on the single color gradient, Laroche and Prescott [23] presented an edge preservation-based demosaicing algorithm. Based on the gradient evaluation and a specific threshold, Hibbard [17] presented a threshold-based demosaicing algorithm. Based on the adaptive color plane, Adams and Hamilton [1] presented an efficient demosaicing algorithm to modify the previous algorithms [17], [23]. In [21], Kimmel presented a color difference-based demosaicing algorithm by using the template matching technique. In [14], Gunturk *et al.* presented an efficient demosaicing algorithm by using alternating projections. Based on steerable wavelet decomposition, Hel-Or and Keren [16] presented an efficient demosaicing algorithm by using the directional smoothing technique. Based on color correlation concept, Pei and Tam [37] presented an efficient demosaicing algorithm. In [26], Lu and Tan presented an efficient demosaicing algorithm based on exploiting spatial and spectral correlations among adjacent pixels and they also presented a quality measure. Based on primary-consistent soft-decision framework, Wu and Zhang [45] presented a multiple estimation to estimate unknown color values according to different edge directions. In [29], Lukac and Plataniotis presented an efficient demosaicing algorithm by using normalized color-ratios. Based on optimal recovery interpolation of grayscale images, Muresan and Parks [36] presented an improved edge-directed demosaicing algorithm. In [18], Hirakawa and Parks presented an adaptive homogeneity-directed demosaicing algorithm. Based on projection-onto-convex-set approach, Li [25] presented the first iterative demosaicing algorithm. In [9], Dubois presented a novel demosaicing algorithm based on frequency-domain representation. In [22], Lai and Liaw presented an modified mean-removed



	j-3	j-2	j-1	j	j+1	j+2	j+3
i-3	R	G	R	G	R	G	R
i-2	G	B	G	B	G	B	G
i-1	R	G	R	G	R	G	R
i	G	B	G	B	G	B	G
i+1	R	G	R	G	R	G	R
i+2	G	B	G	B	G	B	G
i+3	R	G	R	G	R	G	R

Fig. 1. Bayer CFA structure.

Manuscript received July 31, 2007; revised July 08, 2008. Current version published November 12, 2008. This work was upported in part by the National Science Council of Taiwan, R.O.C., under Contracts NSC95-2221-E-011-153, NSC96-2221-E-011-102-MY3, and NSC96-2219-E-001-001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Gabriel Marcu.

K.-L. Chung is with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology No. 43, Taipei, Taiwan 10672, R.O.C. (e-mail: k.l.chung@mail.ntust.edu.tw).

W.-J. Yang, W.-M. Yan, and C.-C. Wang are with the Department of Computer Science and Information Engineering, National Taiwan University No. 1, Taipei, Taiwan 10617, R.O.C.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2008.2005561

1057-7149/$25.00 © 2008 IEEE

vector quantization algorithm to improve the image quality performance of the previous algorithm by Pei and Tam [37]. Based on the linear minimum mean square-error estimation technique, Zhang and Wu [46] presented a new color demosaicing algorithm. Based on concepts of edge-sensing and correlation-correction, Lukac *et al.* [34] presented a new efficient CFA interpolation framework. Su [40] presented an improved iterative demosaicing algorithm using weighted-edge and color-difference interpolations. Based on the spatial correlation and the edge-directional information of the neighboring pixels, Lee *et al.* [24] presented a weighted edge-sensing demosaicing algorithm. In [5], Chung and Chan presented an adaptive demosaicing algorithm by using the variances of color differences along horizontal and vertical edge directions. Currently, based on the Nth-order directional finite derivative of spectral-special correlation, empirically $N = 11$, Tsai and Song [44] presented a lined-based demosaicing algorithm and their algorithm has the best image quality performance in average when compared with all other existing demosaicing algorithms. In addition, the edge-sensing concept used and the obtained edge-directional information in these developed demosaicing algorithms are very helpful to handle the digital zooming issue on CFA model [6], [28], [30], [31], [47].

After examining most of all these previously published demosaicing algorithms and registered patents, we find that the quality of demosaiced images is heavily dependent on the extracted gradient/edge information from input mosaic images, but usually the extracted gradient/edge information on mosaic images is not so accurate. Since each pixel in the mosaic image only has one color channel, the previous color edge detectors [4], [10], [11], [41]–[43], [48] cannot work well on mosaic images directly. The motivations of this research are threefold: 1) developing a new approach to extract more accurate gradient/edge information on mosaic images directly; 2) developing a new approach to determine the adaptive mask size for each pixel in the heterogeneity-projection; 3) developing a new high-quality edge-sensing demosaicing algorithm based on the more accurate gradient/edge information and the adaptive heterogeneity-projection values. The above three motivations lead to the three main contributions of this paper.

In this paper, without demosaicing processing, a new approach to extract more accurate gradient/edge information on mosaic images directly is first proposed. Next, based on spectral-spatial correlation [44], a novel adaptive heterogeneity-projection with proper mask size for each pixel is presented. Combining the extracted gradient/edge information and the adaptive heterogeneity-projection values, we present a new high-quality edge-sensing demosaicing algorithm. Based on 24 popular testing images, experimental results demonstrated that our proposed new demosaicing algorithm has the best quality performance when compared with current published seven demosaicing algorithms by Pei and Tam [37], Lu and Tan [26], Lukac and Plataniotis [29], Dubois [9], Lukac *et al.* [34], Chung and Chan [5], and Tsai and Song [44].

The remainder of this paper is organized as follows. In Section II, a new approach to extract more accurate gradient/edge information on mosaic images is presented. In Section III, a novel adaptive heterogeneity-projection with proper mask size for each pixel is first presented and then combining the extracted gradient/edge information and the adaptive heterogeneity-projection values, our proposed new edge-sensing demosaicing algorithm is presented. In Section IV, some experimental results are carried out to illustrate the quality advantage of our proposed demosaicing algorithm when compared with several recently published algorithms. Finally, some conclusions are addressed in Section V.

c	b	c
b	a	b
c	b	c

Fig. 2. The 3 × 3 single symmetric convolution mask.

II. New Approach to Extract More Accurate Gradient Information on Mosaic Images

This section presents a new approach to extract more accurate gradient information on mosaic images directly. In what follows, the luminance estimation technique [2] for mosaic images is first introduced. Then, combining the luminance estimation technique and Sobel operator [13], our proposed new approach to extract more gradient information on mosaic images is presented. The extracted gradient information will be used in our proposed edge-sensing demosaicing algorithm described in Section III and it could lead to a high-quality advantage.

A. Luminance Estimation Technique for Mosaic Images

In this subsection, the luminance estimation technique for mosaic images is introduced. In the discussions hereafter, the luminance is defined as $L = (1/4)(R + 2G + B)$ and the luminance of the pixel located at position (i, j) is denoted by $L(i, j)$; the R, G, and B color pixels located at position (i, j) of the mosaic image are denoted by $I^r_{\text{mo}}(i, j)$, $I^g_{\text{mo}}(i, j)$, and $I^b_{\text{mo}}(i, j)$, respectively; for a demosaiced full color image, suppose the R, G, and B color values of the pixel at position (i, j) are denoted by $I^r_{\text{dm}}(i, j)$, $I^g_{\text{dm}}(i, j)$, and $I^b_{\text{dm}}(i, j)$, respectively, where i denotes the vertical axis and j denotes the horizontal axis (see Fig. 1).

In the luminance estimation technique, a 3×3 single symmetric convolution mask as shown in Fig. 2 is used to estimate the luminance of the pixel at (i, j) in the mosaic image. Within a small smooth region of the mosaic image, the color values of R, G, and B components approach three different constants, i.e., $I^r_{\text{mo}}(i, j) \cong R_c$, $I^g_{\text{mo}}(i, j) \cong G_c$, and $I^b_{\text{mo}}(i, j) \cong B_c$. The four possible cases of the 3×3 mosaic subimage are illustrated in Fig. 3(a)–(d), respectively. Because of the symmetry of the 3×3 mask in Fig. 2, only Case 1 and Case 2 need to be considered. First, each R channel of Case 1 and Case 2 is considered. After running the mask of Fig. 2 on the two 3×3 mosaic subimages of Fig. 3(a) and (b), we have $4cR_c = 2bR_c$. By the same argument, for the B channel, we have $aB_c = 2bB_c$. For normalizing the sum of the nine coefficients in the mask, we let $a + 4b + 4c = 1$. From $4c = 2b$, $a = 2b$, and $a + 4b + 4c = 1$, we have the following three equations:

$$\begin{cases} 4c = 2b \\ a = 2b \\ a + 4b + 4c = 1. \end{cases} \tag{1}$$

R_c	G_c	R_c
G_c	B_c	G_c
R_c	G_c	R_c

(a)

G_c	R_c	G_c
B_c	G_c	B_c
G_c	R_c	G_c

(b)

B_c	G_c	B_c
G_c	R_c	G_c
B_c	G_c	B_c

(c)

G_c	B_c	G_c
R_c	G_c	R_c
G_c	B_c	G_c

(d)

Fig. 3. Four possible cases of 3 × 3 mosaic subimages. (a) Case 1. (b) Case 2. (c) Case 3. (d) Case 4.

1/16	1/8	1/16
1/8	1/4	1/8
1/16	1/8	1/16

Fig. 4. The 3 × 3 luminance estimation mask.

After solving (1), it yields $a = (1/4)$, $b = (1/8)$, and $c = (1/16)$. Thus, the luminance estimation mask can be determined and it is shown in Fig. 4. After running the luminance estimation mask of Fig. 4 on the 3×3 mosaic subimage centered at position (i, j), the luminance $L(i, j)$ can be obtained by

$$L(i,j) = \frac{1}{16}\left\{\begin{array}{l} \left[\begin{array}{l} I_{\text{mo}}^{C}(i-1,j-1)+I_{\text{mo}}^{C}(i-1,j+1) \\ +I_{\text{mo}}^{C}(i+1,j-1)+I_{\text{mo}}^{C}(i+1,j+1) \end{array}\right] \\ +2\left[\begin{array}{l} I_{\text{mo}}^{C}(i-1,j)+I_{\text{mo}}^{C}(i+1,j) \\ +I_{\text{mo}}^{C}(i,j-1)+I_{\text{mo}}^{C}(i,j+1) \end{array}\right] \\ +4I_{\text{mo}}^{C}(i,j) \end{array}\right\} \tag{2}$$

where

$$I_{\text{mo}}^{C}(m,n) = \begin{cases} I_{\text{mo}}^{r}(m,n), & \text{if } m \in \text{ odd and } n \in \text{ even} \\ I_{\text{mo}}^{g}(m,n), & \text{if } (m+n) \in \text{ even} \\ I_{\text{mo}}^{b}(m,n), & \text{if } m \in \text{ even and } n \in \text{ odd} \end{cases}$$

In the next subsection, the proposed approach to extract more accurate gradient information on mosaic images by combining the Sobel operator and the luminance estimation technique will be presented.

B. Proposed Approach to Extract More Accurate Gradient Information on Mosaic Images

In this subsection, combining the Sobel operator [13] and the luminance estimation technique mentioned above, our proposed new approach to extract more accurate gradient information on mosaic images is presented.

-1	0	1
-2	0	2
-1	0	1

(a)

-1	-2	-1
0	0	0
1	2	1

(b)

0	1	2
-1	0	1
-2	-1	0

(c)

2	1	0
1	0	-1
0	-1	-2

(d)

Fig. 5. The 3 × 3 Sobel operator. (a) The horizontal mask. (b) The vertical mask. (c) The $(\pi/4)$-diagonal mask. (d) The $(-\pi/4)$-diagonal mask.

Before presenting the proposed new approach, for completeness, how to use the Sobel operator to extract the gradient information on the luminance map is first introduced. Fig. 5 illustrates the four masks used in the 3×3 Sobel operator for the luminance map.

Given a luminance map, it is known that the luminance of the pixel located at position (i, j) is denoted by $L(i, j)$. After running Sobel horizontal, vertical, $(\pi/4)$-diagonal, and $(-\pi/4)$-diagonal masks as shown in Fig. 5 on the 3×3 luminance submap centered at position (i, j), the horizontal response $\Delta I_{\text{dm}}^{H}(i,j)$, the vertical response $\Delta I_{\text{dm}}^{V}(i,j)$, the $(\pi/4)$-diagonal response $\Delta I_{\text{dm}}^{(\pi/4)}(i,j)$, and the $(-\pi/4)$-diagonal response $\Delta I_{\text{dm}}^{(-\pi/4)}(i,j)$ can be calculated by (3), shown at the bottom of the page.

In order to make Sobel operator workable on mosaic images to extract more accurate gradient information, the luminance estimation technique could be embedded into the Sobel operator. Combining (2) and (3), our proposed new approach to extract more accurate gradient information on mosaic images directly can be followed (detailed derivations are shown in Appendix I). According to Appendix I, the derived Sobel-and luminance estimation-based (SL-based) quad-mask can be obtained. For saving computational effort, the coefficients in the derived quad-mask can be normalized to integers, and the four normalized SL-based quad-masks are shown in Fig. 6.

After running the above four SL-based mask on the 5×5 mosaic subimage centered at position (i, j), the horizontal response $\Delta I_{\text{dm}}^{H}(i,j)$, the vertical response $\Delta I_{\text{dm}}^{V}(i,j)$, the $(\pi/4)$-diagonal response $\Delta I_{\text{dm}}^{(\pi/4)}(i,j)$, and the $(-\pi/4)$-diagonal response $\Delta I_{\text{dm}}^{(-\pi/4)}(i,j)$ can be obtained. In addition, experimental results show that the proposed approach to extract gradient information on mosaic images directly has better

$$\begin{aligned} \Delta I_{\text{dm}}^{H}(i,j) &= \left\{\begin{array}{l} [L(i-1,j+1)+2L(i,j+1)+L(i+1,j+1)] \\ -[L(i-1,j-1)+2L(i,j-1)+L(i+1,j-1)] \end{array}\right\} \\ \Delta I_{\text{dm}}^{V}(i,j) &= \left\{\begin{array}{l} [L(i+1,j-1)+2L(i+1,j)+L(i+1,j+1)] \\ -[L(i-1,j-1)+2L(i-1,j)+L(i-1,j+1)] \end{array}\right\} \\ \Delta I_{\text{dm}}^{\frac{\pi}{4}}(i,j) &= \left\{\begin{array}{l} [L(i-1,j)+2L(i-1,j+1)+L(i,j+1)] \\ -[L(i+1,j)+2L(i+1,j-1)+L(i,j-1)] \end{array}\right\} \\ \Delta I_{\text{dm}}^{\frac{-\pi}{4}}(i,j) &= \left\{\begin{array}{l} [L(i-1,j)+2L(i-1,j-1)+L(i,j-1)] \\ -[L(i+1,j)+2L(i+1,j+1)+L(i,j+1)] \end{array}\right\} \end{aligned} \tag{3}$$

-1	-2		2	1
-4	-8		8	4
-6	-12		12	6
-4	-8		8	4
-1	-2		2	1

(a)

-1	-4	-6	-4	-1
-2	-8	-12	-8	-2
2	8	12	8	2
1	4	6	4	1

(b)

	1	4	5	2
-1		8	14	5
-4	-8		8	4
-5	-14	-8		1
-2	-5	-4	-1	

(c)

2	5	4	1	
5	14	8		-1
4	8		-8	-4
1		-8	-14	-5
	-1	-4	-5	-2

(d)

Fig. 6. Four SL-based masks. (a) The horizontal SL-based mask. (b) The vertical SL-based mask. (c) The $(\pi/4)$-diagonal SL-based mask. (d) The $(-\pi/4)$-diagonal SL-based mask.

performance and the average execution-time improvement ratio is 17.8% $(= (0.360s - 0.296s)/(0.360s)) \times 100\%)$ when compared with the indirect approach: first apply the bilinear demosaicing process to the input mosaic image; next convert the demosaiced full color image to the luminance map, and then the Sobel edge detector is run on the obtained luminance map [35].

Note that besides our proposed approach and the indirect approach mentioned above, we now briefly introduce the third approach in [7]: first, plug the bilinear interpolation technique [38] into the Sobel operator to derive two quad-mask for the G channel, four quad-mask for R channel, and the same four mask-pairs for B channel. Next, run the above proper mask-pairs on the 5×5 mosaic subimage to compute the gradient response of the corresponding color channel. Further, let the four gradient responses of R, G, and B channels are denoted by $\Delta I_{\mathrm{dm}}^{d,r}$, $\Delta I_{\mathrm{dm}}^{d,g}$, and $\Delta I_{\mathrm{dm}}^{d,b}$ for $d \in \{V, H, (\pi/4), (-\pi/4)\}$, respectively. Finally, combine R, G, and B gradient responses to obtain the integrated gradient responses $\Delta I_{\mathrm{dm}}^{d}$ by using the equation: $\Delta I_{\mathrm{dm}}^{d} = 0.299\Delta I_{\mathrm{dm}}^{d,r} + 0.587\Delta I_{\mathrm{dm}}^{d,g} + 0.114\Delta I_{\mathrm{dm}}^{d,b}$ for $d \in \{V, H, (\pi/4), (-\pi/4)\}$. For convenience, the demosaicing algorithm based on the third approach is called the Sobel- and interpolation-based (SI-based) demosaicing algorithm. Experimental results show that the average image quality performance, color peak signal-to-noise ratio (CPSNR) and ΔE_{ab}^{*}, of the SI-based demosaicing algorithm could be improved by $0.027\%(= \quad (40.200\ \mathrm{dB} - 40.189\ \mathrm{dB}/40.200\ \mathrm{dB}) \times 100\%)$ and $0.135\%(= \quad (1.2606 - 1.2589/1.2606) \times 100\%)$, respectively, when compared with the proposed demosaicing algorithm in this paper. However, the average time performance of the SI-based demosaicing algorithm would be degraded $8.3\%(= \ (1.573s - 1.442s/1.573s) \times 100\%)$. Since the average image quality performance improvement of the SI-based demosaicing algorithm over our proposed algorithm is negligible but the degraded time performance is moderate, thus, our proposed SL-based operator described in this section is the better choice for handling the edge-sensing process which will be used in our proposed demosaicing algorithm described in next section.

III. Proposed New Edge-Sensing Demosaicing Algorithm

Based on the accurate gradient information obtained in last section, this section presents our proposed new high-quality edge-sensing demosaicing algorithm. Our proposed novel adaptive heterogeneity-projection with proper mask size for each pixel is first presented in Section III-A to extract more accurate horizontal and vertical heterogeneity-projection value. Based on the two obtained heterogeneity-projection values and those obtained gradient information mentioned in Section II, the proposed new edge-sensing interpolation estimation is presented. The proposed interpolation estimation consists of two steps, the first step for G channel and the second step for R and B channels.

A. Novel Adaptive Heterogeneity-Projection for Mosaic Images

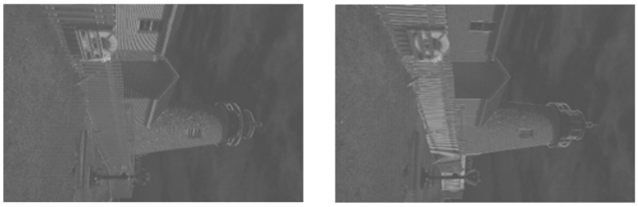
In this subsection, a novel adaptive heterogeneity-projection with proper mask size for each pixel is presented. Given an original mosaic image I_{mo}, its horizontal heterogeneity-projection map $\mathrm{HP}_{\mathrm{H-map}}$ and vertical heterogeneity-projection map $\mathrm{HP}_{\mathrm{V-map}}$ can be obtained by running the following two 1-D Laplacian operations [44]

$$\begin{aligned} \mathrm{HP}_{\mathrm{H-map}} &= |I_{\mathrm{mo}} * (\mathbf{F}^{N\times(N-3)}\mathbf{T}^{(N-3)\times 1})^{t}| \\ \mathrm{HP}_{V-map} &= |I_{\mathrm{mo}} * (\mathbf{F}^{N\times(N-3)}\mathbf{T}^{(N-3)\times 1})| \end{aligned} \tag{4}$$

where N (=11 empirically [44]) denotes the vector length (or the 1-D mask size); $\mathbf{F}^{N\times(N-3)} = [1\ -1\ -1\ 1]^{t} * \mathbf{I}^{(N-3)\times(N-3)}$ denotes a $N \times (N-3)$ coefficient matrix; $\mathbf{T}^{(N-3)\times 1} = \prod_{x=1}^{N-4}[1 \ \ -1]^{t} * \mathbf{I}^{(N-3-x)\times(N-3-x)}$ denotes a $(N-3) \times 1$ coefficient vector; $\mathbf{I}^{M\times M}$ denotes an $M \times M$ identity matrix; the symbol "$*$" denotes the 2-D convolution operator; $|\cdot|$ denotes the absolute value operator and the operator "t" denotes the transpose operator. In what follows, our approach can determine the suitable value of N adaptively.

Let us examine some cases of the current pixel on the mosaic image. If the surrounding region of the current pixel is homogenous, the two responses by (4) are almost the same whether a large mask size or small mask size is adopted. If there is one tiny horizontal edge passing through the current pixel, a small mask size for $\mathrm{HP}_{V-\mathrm{map}}$ is enough rather than a large mask size. It still holds for the tiny vertical edge corresponding to $\mathrm{HP}_{H-\mathrm{map}}$. In [44], the adopted mask size is fixed and is set to $N = 11$. According to the above discussion, we now present our proposed adaptive heterogeneity-projection with proper mask size for each pixel such that the used mask size is as small as possible and the computed responses are more accurate than those computed by the mask with size 11. Experimental results reveal that our proposed adaptive heterogeneity-projection approach under proper horizontal and vertical mask sizes has the computation-saving and more accurate advantages.

We utilize the horizontal spectral-spatial correlation (SSC) [44] between the current pixel at location (i, j) and its neighboring pixel at location $(i, j+1)$ to determine the proper horizontal mask size $N_H(i, j)$, and the proper vertical mask size $N_V(i, j)$ is determined by the vertical SSC between the current pixel and its neighboring pixel at location $(i+1, j)$. For a horizontal SSC map, the horizontal SSC value at location (i, j) can be obtained by using the following rule, shown in the first equation at the bottom of the next page. Similarly, the vertical SSC value at location (i, j) can be obtained by using the following rule, shown in the second equation at the bottom of the next page. The horizontal SSC map and the vertical SSC map of the mosaic Lighthouse image are illustrated in Fig. 7(a) and (b),



(a) (b)

Fig. 7. Two SSC maps of the mosaic Lighthouse image. (a) The horizontal SSC map. (b) The vertical SSC map (in order to show the images more clear, the gray value 80 is used to represent the value 0).

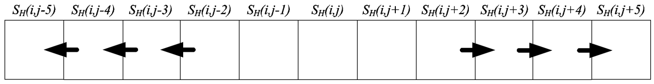


Fig. 8. Depiction of our proposed approach to determine the proper horizontal mask size $N_H(i,j)$.

respectively, and it is observed that the SSC values are locally constant in homogeneous regions.

Because the determination of the proper horizontal mask size $N_H(i,j)$ is the same as that for the vertical mask size $N_V(i,j)$, we thus only focus on the determination of horizontal mask size. Fig. 8 depicts our proposed approach to determine the proper horizontal mask size $N_H(i,j)$. Centered at location (i,j), Fig. 8 illustrates a 1×11 row data extracted from the horizontal SSC map. The procedure to determine the proper horizontal mask size $N_H(i,j)$ consists of the following three steps.

Step 1) Initially, we compute the two SSC values $S_H(i,j-2)$ and $S_H(i,j+2)$, and temporarily set $N_H(i,j) = 5, x_{\text{left}} = j-2$, and $x_{\text{right}} = j+2$.

Step 2) If the condition $N_H(i,j) = 11$ holds, output $N_H(i,j)$ and stop. Here, the maximal horizontal mask size is bounded by eleven. Otherwise, go to Step 3.

Step 3) Examine whether the neighboring horizontal SSC values are locally chosen by using the following testing condition:

$$\text{Max}\left(DS_H^{\text{left}}, DS_H^{\text{right}}\right) < T_h$$

where $DS_H^{\text{left}} = |S_H(i,x_{\text{left}}) - S_H(i,x_{\text{left}}-1)| + |S_H(i,x_{\text{left}}) - S_H(i,x_{\text{left}}+1)|, DS_H^{\text{right}} = |S_H(i,x_{\text{right}}) - S_H(i,x_{\text{right}}-1)| + |S_H(i,x_{\text{right}}) - S_H(i,x_{\text{right}}+1)|$, and determination of the threshold T_h is discussed in Appendix II. If the above testing condition holds, output the value of $N_H(i,j)$ as the proper horizontal mask size and stop. Otherwise, perform the operation $N_H(i,j) = N_H(i,j)+2, x_{\text{left}} = x_{\text{left}} - 1$, and $x_{\text{right}} = x_{\text{right}} + 1$, and then go to Step 2.

The above three-step procedure can be easily applied to determine the proper horizontal mask size. In order to normalize the masks for different sizes, the normal factor $1/Q_{(N)}$ is used to normalize the coefficients of the mask. In other words, instead of using $\mathbf{F}^{N\times(N-3)}\mathbf{T}^{(N-3)\times 1}$, we use $1/Q_{(N)}(\mathbf{F}^{N\times(N-3)}\mathbf{T}^{(N-3)\times 1})$ to obtain the normalized heterogeneity-projection map. The value of $Q_{(N)}$ is defined as the sum of the positive coefficients in the mask. For example, if $N=5$, the mask $\mathbf{F}^{5\times(5-3)}\mathbf{T}^{(N-3)\times 1} = [1\ -2\ 0\ 2\ -1]^t$, can be normalized to $F^{5\times 2}T^{2\times 1}/Q_{(5)} = [1\ -2\ 0\ 2\ -1]^t/3$.

Finally, in order to reduce the estimation error, we use the local mean to tune the heterogeneity-projection maps. For $\text{HP}_{\text{H-map}}$, the horizontal heterogeneity-projection value at location (i,j) is denoted by $\text{HP}_H(i,j)$ and for $\text{HP}_{V\text{-map}}, \text{HP}_V(i,j)$ denotes the vertical heterogeneity-projection value at location (i,j). The tuned horizontal heterogeneity-projection value $\text{HP}'_H(i,j)$ and the tuned vertical heterogeneity-projection value $\text{HP}'_V(i,j)$ can be computed by the following operations:

$$\text{HP}'_H(i,j) = \frac{1}{6}\sum_{k=-2}^{2} \delta_k \text{HP}_H(i,j+k)$$

$$\text{HP}'_V(i,j) = \frac{1}{6}\sum_{k=-2}^{2} \delta_k \text{HP}_V(i+k,j)$$

where $\delta_k = 2$ if $k=0$; $\delta_k = 1$, otherwise.

After performing the above adaptive heterogeneity-projection for mosaic images, the values $\text{HP}'_H(i,j)$ and $\text{HP}'_V(i,j)$ can be obtained for each mosaic pixel. In next two subsections, the six responses $\text{HP}'_H(i,j), \text{HP}'_V(i,j), \Delta I_{\text{dm}}^H(i,j), \Delta I_{\text{dm}}^V(i,j), \Delta I_{\text{dm}}^{\pi/4}(i,j)$, and $\Delta I_{\text{dm}}^{-\pi/4}(i,j)$ will be used in our proposed new edge-sensing interpolation estimation for demosaicing mosaic images.

$$S_H(i,j) = \begin{cases} |I_{\text{mo}}^g(i,j) - I_{\text{mo}}^r(i,j+1)|, & \text{if the current pixel is G and } i \in \text{odd} \\ |I_{\text{mo}}^g(i,j) - I_{\text{mo}}^b(i,j+1)|, & \text{if the current pixel is G and } i \in \text{even} \\ |I_{\text{mo}}^r(i,j) - I_{\text{mo}}^g(i,j+1)|, & \text{if the current pixel is R.} \\ |I_{\text{mo}}^b(i,j) - I_{\text{mo}}^g(i,j+1)|, & \text{if the current pixel is B} \end{cases}$$

$$S_V(i,j) = \begin{cases} |I_{\text{mo}}^g(i,j) - I_{\text{mo}}^r(i+1,j)| & \text{if the current pixel is green and } j \in \text{even.} \\ |I_{\text{mo}}^g(i,j) - I_{\text{mo}}^b(i+1,j)| & \text{if the current pixel is green and } j \in \text{odd.} \\ |I_{\text{mo}}^r(i,j) - I_{\text{mo}}^g(i+1,j)| & \text{if the current pixel is red.} \\ |I_{\text{mo}}^b(i,j) - I_{\text{mo}}^g(i+1,j)| & \text{if the current pixel is blue} \end{cases}$$

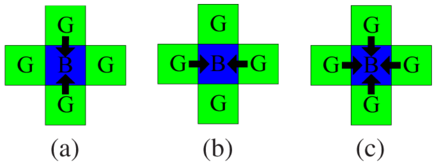


Fig. 9. Data dependence of our proposed interpolation estimation for G channel. (a) Horizontal variation (vertical edge). (b) Vertical variation (horizontal edge). (c) The other variations.

B. Edge-Sensing Interpolation Estimation for G Channel

In this subsection, we present the proposed high-quality edge-sensing interpolation estimation for *G* channel. For exposition, let us take Fig. 1 to explain how to estimate the *G* channel value located at the center position of Fig. 1. Before performing the interpolation estimation for *G* channel, assume the gradient information of the current mosaic pixel at position (i,j) and the eight neighboring pixels at positions $\Omega_n = \{(x,y)|(x,y) = (i\pm 1,j),(i\pm 2,j),(i,j\pm 1),(i,j\pm 2)\}$ have been extracted by using our proposed method described in last section, and the extracted nine horizontal gradient magnitudes and nine vertical gradient magnitudes are denoted by $\Delta I^H_{\text{dm}}(*,*)$ and $\Delta I^V_{\text{dm}}(*,*)$, respectively.

According to the tuned horizontal heterogeneity-projection value $\text{HP}'_H(i,j)$ and the tuned vertical heterogeneity-projection value $\text{HP}'_V(i,j)$ of the current mosaic pixel at position (i,j), the interpolation estimation scheme in our proposed edge-sensing demosaicing algorithm considers three cases, namely 1) horizontal variation as shown in Fig. 9(a), 2) vertical variation as shown in Fig. 9(b), and 3) the other variations as shown in Fig. 9(c). The arrows in Fig. 9 denote the data dependence.

In addition, in order to estimate $I^g_{\text{dm}}(i,j)$ more accurately from its four neighbors, four proper weights in terms of gradient information are assigned to corresponding four spectral-correlation terms in the interpolation estimation. Considering the neighboring pixel located at location $(i-1,j)$, if there is a horizontal edge passing through it, i.e., the vertical gradient magnitude of the pixel at location $(i-1,j)$ is large, the color difference assumption [26], [37] reveals that the green component of this pixel makes less contribution to the estimation of green component for the current pixel at location (i,j); otherwise, the green component of this pixel makes more contribution to the estimation of green component for the current pixel. On the other hand, if the gradient magnitudes of the pixels at location $(i-2,j)$ and (i,j) are large, the pixel at location $(i-1,j)$ is located in a nonhomogeneous region and we claim that the pixel at location $(i-1,j)$ makes less green contribution to the estimation of green component for the current mosaic pixel at location (i,j); otherwise, the pixel at location $(i-1,j)$ is located in a homogeneous region and we claim that the pixel at location $(i-1,j)$ makes more green contribution to the estimation of green component for the current mosaic pixel. Combining the above analysis of gradient and direction effects, the weight of pixel at location $(i-1,j)$ can be given by $w_g(i-1,j) = 1/\{1+\beta[\Delta I^V_{\text{dm}}(i-2,j)+2\{\Delta I^V_{\text{dm}}(i-1,j)+\Delta I^V_{\text{dm}}(i,j)]\}$. Following the similar discussion, the weights of the four neighbors of the current pixel are expressed by $w_g(i-1,j) = 1/\{1+\beta[\sum_{k=0}^{2}\delta'_k\Delta I^V_{\text{dm}}(i-k,j)]\}, w_g(i+1,j) = 1/\{1+\beta[\sum_{k=0}^{2}\delta'_k\Delta I^V_{\text{dm}}(i+k,j)]\}, w_g(i,j-1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta'_k\Delta I^H_{\text{dm}}(i,j-k)]\}$, and $w_g(i,j+1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta'_k\Delta I^H_{\text{dm}}(i,j+k)]\}$ where $\delta'_k = 2$ if $k = 1$; $\delta'_k = 1$, otherwise. The determination of the parameter β is discussed in Appendix II.

According to the above description, the value of $I^g_{\text{dm}}(i,j)$ of the current pixel at location (i,j) can be estimated by the rules, shown in the equation at the bottom of the page, where for $(x_1,y_1) \in \{(i\pm 1,j)\}, D_g(x_1,y_1) = I^g_{\text{mo}}(x_1,y_1) - I^b_{\text{mo}}(x_1+1,y_1)+I^b_{\text{mo}}(x_1-1,y_1)/2$; for $(x_2,y_2) \in \{(i,j\pm 1)\}, D_g(x_2,y_2) = I^g_{\text{mo}}(x_2,y_2) - I^b_{\text{mo}}(x_2,y_2+1)+I^b_{\text{mo}}(x_2,y_2-1)/2$; the determination of the parameter α is discussed in Appendix II.

After performing the above edge-sensing interpolation estimation for *G* channel, the *G* channel of the demosaiced image is fully populated. In next subsection, the fully populated *G* channel of the image will be used to assist the interpolation of *R* and *B* channels.

C. Edge-Sensing Interpolation Estimation for R and B Channels

Because the number of R pixels or B pixels is less than G in the mosaic image, the interpolation estimation for R and B channels should be partitioned into two steps: 1) estimating the red values at blue pixels, and vice versa; 2) then, recovering missing red and blue values at green pixels. Because the interpolation estimation for R channel is the same as it for B channel, we thus only present it for R channel. In our proposed interpolation estimation for R and B channels, the fully populated G channel is used to assist the interpolation of R and B channels. For convenience, we still use Fig. 1 to explain how to estimate the R channel value for the current pixel at position (i,j).

Similar to the interpolation estimation for G channel, assume the gradient information of the current mosaic pixel at position (i,j) and the eight pixels at positions $\Omega'_n = \{(x,y)|(x,y) = (i\pm 1,j\pm 1),(i\pm 2,j\pm 2)\}$, respectively, have been extracted by using our proposed method. By the same argument, in order

$$I^g_{\text{dm}}(i,j) = I^b_{\text{mo}}(i,j) + \frac{\sum_{(x,y)\in\Omega_g} w_g(x,y)D_g(x,y)}{\sum_{(x,y)\in\Omega_g} w_g(x,y)}$$

$$\Omega_g = \begin{cases} \{(i\pm 1,j)\}, \text{ if } \text{HP}'_V(i,j) < \alpha\text{HP}'_H(i,j) \text{ (horizontal variation)} \\ \{(i,j\pm 1)\}, \text{ if } \text{HP}'_H(i,j) < \alpha\text{HP}'_V(i,j) \text{ (vertical variation)} \\ \{(i\pm 1,j),(i,j\pm 1)\}, \text{ otherwise (other variations)} \end{cases}$$

to estimate $I_{\rm dm}^{r}(i,j)$ more accurately from its four red neighbors, four proper weights in terms of gradient/direction information are assigned to the corresponding four spectral-correlation terms in the interpolation estimation. For estimating $I_{\rm dm}^{r}(i,j)$, we consider four diagonal variations of the mosaic pixel at position (i,j) to determine the four weights which will be used in the proposed edge-sensing estimation of $I_{\rm dm}^{r}(i,j)$. Considering the neighboring pixel at location $(i-1,j-1)$, if there is a $\pi/4$-diagonal edge passing through it, i.e., the $-\pi/4$-diagonal gradient magnitude of the pixel at location $(i-1,j-1)$ is large, and then it claims that the red component of this pixel makes less contribution to the estimation of red component for the current pixel at location (i,j); otherwise, it claims that the red component of this pixel makes more contribution to the estimation of red component for the current pixel. On the contrary, if the gradient magnitudes of the pixel at location $(i-1,j-1)$ and $(i-2,j-2)$ are large, i.e., the pixel at location $(i-1,j-1)$ is inside nonhomogeneous region, the color difference assumption indicates that the red component of this pixel makes less contribution to the estimation of red component for the current pixel at location (i,j); otherwise, the red component of this pixel makes more contribution to the estimation of red component for the current pixel. Following the above analysis, the weights of the four diagonal red neighbors of the current pixel can be expressed by $w_r(i-1,j-1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{-\pi/4}(i-k,j-k)]\}, w_r(i-1,j+1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{\pi/4}(i-k,j+k)]\}, w_r(i+1,j-1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{\pi/4}(i+k,j-k)]\}$, and $w_r(i+1,j+1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{-\pi/4}(i+k,j+k)]\}$ where $\delta_k' = 2$ if $k=1$; $\delta_k' = 1$, otherwise. Based on the four weights obtained above and the color difference concept, the demosaiced full red color for the blue pixel in the mosaic image, $I_{\rm dm}^{r}(i,j)$, can be estimated by

$$I_{\rm dm}^{r}(i,j) = I_{\rm dm}^{g}(i,j) + \frac{\sum_{(x,y)\in\Omega_r} w_r(x,y)D_r(x,y)}{\sum_{(x,y)\in\Omega_r} w_r(x,y)}$$

where $\Omega_r = \{(x,y)|(i\pm 1, j\pm 1)\}$; for $(x,y)\in\{(i\pm 1,j\pm 1)\}, D_r(x,y) = I_{\rm mo}^{r}(x,y) - I_{\rm dm}^{g}(x,y)$.

After describing how to estimate demosaiced full red colors for those blue pixels in the mosaic image, we now introduce how to estimate full red colors for those green pixels. Fig. 10 illustrates the pattern of R channel. Referring to Fig. 10, the full red color for the green pixel, $I_{\rm dm}^{r}(i,j)$, can be estimated by the rules, shown at the bottom of the page, where for $(x,y)\in\{(i\pm 1,j),(i,j\pm 1)\}$, we perform $D_r(x,y) = I_{\rm mo}^{r}(x,y) - I_{\rm dm}^{g}(x,y)$. If $k=1$, we set $\delta_k' = 2$; otherwise, we set $\delta_k' = 1$. We further perform $w_r(i-1,j) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{V}(i-k,j)]\}, w_r(i+1,j) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{V}(i+k,j)]\}, w_r(i,j-1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{H}(i,j-k)]\}$, and $w_r(i,j+1) = 1/\{1+\beta[\sum_{k=0}^{2}\delta_k'\Delta I_{\rm dm}^{H}(i,j+k)]\}$.

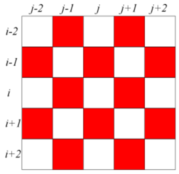


Fig. 10. Patterns of the R channel.

Fig. 11. Twenty-four testing images from Kodak PhotoCD [49].

After presenting our proposed new edge-sensing demosaicing algorithm, experimental results in next section will illustrate the quality advantage of our proposed new edge-sensing demosaicing algorithm.

IV. Experimental Results

In this section, based on 24 popular testing mosaic images, some experimental results are demonstrated to show that our proposed new demosaicing algorithm has better image quality performance when compared with the previous seven algorithms by Pei and Tam [37], Lu and Tan [26], Lukac and Plataniotis [29], Dubois [9], Lukac *et al.* [34], Chung and Chan [5], and Tsai and Song [44], respectively. The concerned algorithms are implemented on the IBM compatible computer with Intel Core 2 Duo CPU 1.6 GHz and 1-GB RAM. The operating system used is MS-Windows XP and the program developing environment is Borland C++ Builder 6.0. Our program has been uploaded in [50].

Fig. 11 illustrates the 24 testing images from Kodak PhotoCD [49]. In our experiments, the 24 testing images shown in Fig. 11 are first down-sampled to obtain the mosaic images. Furthermore, the boundaries of the image are dealt with using the mirroring method. Here, we adopt two objective image quality measures, CPSNR and S-CIELAB ΔE_{ab}^{*} metric [19], [26], and one subjective image quality measure, color artifacts, to justify the

$$I_{\rm dm}^{r}(i,j) = I_{\rm dm}^{g}(i,j) + \frac{\sum_{(x,y)\in\Omega_r'} w_r(x,y)D_r(x,y)}{\sum_{(x,y)\in\Omega_r'} w_r(x,y)}$$

$$\Omega_r' = \begin{cases} \{(i\pm 1,j)\}, \text{ if } {\rm HP}_V'(i,j) < \alpha{\rm HP}_H'(i,j) \text{ (horizontal variation)} \\ \{(i,j\pm 1)\}, \text{ if } {\rm HP}_H'(i,j) < \alpha{\rm HP}_V'(i,j) \text{ (vertical variation)} \\ \{(i\pm 1,j),(i,j\pm 1)\}, \text{ otherwise (other variations)} \end{cases}$$

TABLE I
CPSNR QUALITY COMPARISON FOR 24 TESTING IMAGES

Method	[37]	[26]	[29]	[9]	[34]	[5]	[44]	Proposed
Image01	36.119	39.134	37.647	38.017	37.414	37.995	39.128	**39.879**
Image02	38.155	38.737	39.701	39.319	**40.012**	39.028	39.553	39.790
Image03	40.769	41.092	41.964	41.693	**42.393**	41.492	42.084	42.314
Image04	39.402	39.466	40.451	40.515	**40.786**	39.735	40.042	40.147
Image05	36.795	35.830	36.983	37.875	37.778	36.962	37.779	**37.955**
Image06	37.160	37.838	38.066	39.959	37.445	39.435	40.292	**40.754**
Image07	40.516	40.713	41.899	41.973	**42.478**	40.89	41.51	41.668
Image08	32.578	34.805	35.487	35.272	34.572	35.723	37.067	**37.558**
Image09	40.743	41.363	42.024	42.174	42.320	42.030	42.683	**42.914**
Image10	41.338	41.026	41.582	42.166	42.188	41.495	41.979	**42.323**
Image11	38.278	38.406	39.367	39.805	39.139	39.331	40.160	**40.556**
Image12	40.755	41.819	42.415	42.977	42.274	42.669	43.300	**43.541**
Image13	34.358	33.139	34.160	35.127	34.292	34.651	35.595	**36.224**
Image14	35.425	35.013	36.556	36.035	**37.066**	35.747	36.199	36.244
Image15	38.293	38.461	39.264	39.265	39.245	38.808	39.008	**39.326**
Image16	39.907	41.697	41.547	43.705	40.452	43.293	43.926	**44.314**
Image17	41.068	40.587	40.991	41.706	41.585	41.244	41.752	**42.050**
Image18	37.020	35.747	36.668	37.266	37.089	36.688	37.262	**37.683**
Image19	37.284	39.940	40.160	40.416	39.335	40.399	41.310	**41.752**
Image20	40.075	40.266	41.006	40.386	41.175	40.345	41.263	**41.580**
Image21	37.886	38.236	38.736	38.857	38.735	38.517	39.711	**40.165**
Image22	37.514	37.521	38.219	38.223	**38.518**	37.650	38.139	38.445
Image23	41.055	41.219	42.007	42.193	**42.928**	41.526	41.776	42.069
Image24	34.986	33.569	34.389	**35.338**	34.89	34.778	34.779	35.291
Average	38.228	38.491	39.220	39.594	39.338	39.185	39.846	**40.189**

better quality performance of our proposed novel demosaicing algorithm. The CPSNR of a color image with size X × Y is defined by

$$\mathrm{CPSNR} = 10\log_{10}\frac{255^2}{\mathrm{MSE}}$$

$$\mathrm{MSE} = \frac{1}{3XY}\sum_{i=0}^{X-1}\sum_{j=0}^{Y-1}\sum_{c\in C}\left[I_{\mathrm{ori}}^{c}(i,j) - I_{\mathrm{dm}}^{c}(i,j)\right]^2$$

where $C = \{r, g, b\}; I_{\mathrm{ori}}^{r}(i,j), I_{\mathrm{ori}}^{g}(i,j)$, and $I_{\mathrm{ori}}^{b}(i,j)$ denote the three color components of the pixel at location (i,j) in the original full color image; $I_{\mathrm{dm}}^{r}(i,j), I_{\mathrm{dm}}^{g}(i,j)$, and $I_{\mathrm{dm}}^{b}(i,j)$ denote the three color components of the pixel at location (i,j) in the demosaiced image. The S-CIELAB ΔE_{ab}^{*} of a color image with size X × Y is defined by

$$\Delta E_{ab}^{*} = \frac{1}{XY}$$

$$\times \textstyle\sum_{i=0}^{X-1}\sum_{j=0}^{Y-1}\left\{\sqrt{\sum_{c\in\Psi}\left[\mathrm{LAB}_{\mathrm{ori}}^{c}(i,j) - \mathrm{LAB}_{\mathrm{dm}}^{c}(i,j)\right]^2}\right\}$$

where $\Psi = \{L, a, b\}; \mathrm{LAB}_{\mathrm{ori}}^{L}(i,j), \mathrm{LAB}_{\mathrm{ori}}^{a}(i,j)$, and $\mathrm{LAB}_{\mathrm{ori}}^{b}(i,j)$ denote the three CIELAB color components of the pixel at location (i,j) in the original full color image; $\mathrm{LAB}_{\mathrm{dm}}^{L}(i,j), \mathrm{LAB}_{\mathrm{dm}}^{a}(i,j)$, and $\mathrm{LAB}_{\mathrm{dm}}^{b}(i,j)$ denote the three CIELAB color components of the pixel at location (i,j) in the demosaiced image.

For fairness, among the eight concerned demosaicing algorithms, the three existing algorithms [5], [9], [34] still apply their own refinement schemes; the other four existing algorithms [26], [29], [37], [44] and our proposed algorithm utilize the postprocessing approach by Lukac *et al.* [27] to enhance the demosaiced image quality. Based on 24 testing mosaic images, Tables I and II demonstrate the demosaiced image quality comparison in terms of CPSNR and S-CIELAB ΔE_{ab}^{*} among our proposed algorithm and the other seven demosaicing algorithms, respectively. In Tables I and II, the entries with the best CPSNR or the least ΔE_{ab}^{*} are highlighted by bold black. It is observed that in average, our proposed demosaicing algorithm has the best demosaiced image quality in terms of CPSNR and ΔE_{ab}^{*}.

Next, we adopt the subjective image visual measure, color artifacts, to demonstrate the quality advantage of our proposed demosaicing algorithm. After performing the demosaicing processing, some degree of color artifacts may happen on edges or textures of the demosaiced image. We first take the magnified subimages cut from the testing image No. 19 as shown in Fig. 12 to compare the visual effect among the concerned eight algorithms. Fig. 12(a)–(i) illustrates the nine magnified subimages cut from the original testing image No. 19, the demosaiced image obtained by Pei and Tam's demosaicing algorithm, the one obtained by Lu and Tan's demosaicing algorithm, the one obtained by Lukac and Plataniotis' algorithm, the one obtained by Dubois' demosaicing algorithm, the one obtained by Lukac *et al.*'s algorithm, the one obtained by Chung and Chan's algorithm, the one obtained by Tsai and Song's demosaicing algorithm, and the one obtained by our proposed demosaicing algorithm, respectively. Comparing the visual effect between the original full color image and the one in Fig. 12(b)–(i), it is observed that our proposed demosaicing algorithm creates less color artifacts when compared with the other seven demosaicing algorithms. Then, we take the magnified subimages cut from the testing image No. 8 to depict the visual comparison. Fig. 13(a)–(i) illustrates the magnified subimages cut from the original full color testing image No. 8 and the eight demosaiced images. From visual comparison, it is observed that our proposed demosaicing algorithm produces less color artifacts when compared with the other seven demosaicing algorithms.

Furthermore, the average execution-time of the eight concerned demosaicing algorithms for the 24 testing mosaic images

TABLE II
S-CIELAB ΔE^*_{ab} QUALITY COMPARISON FOR 24 TESTING IMAGES

Method	[37]	[26]	[29]	[9]	[34]	[5]	[44]	Proposed
Image01	1.7773	1.4103	1.6251	1.5592	1.5948	1.5276	1.3680	**1.2739**
Image02	1.7874	1.6021	1.5501	1.7603	1.5337	1.6490	1.5584	**1.5327**
Image03	1.0050	0.9238	0.9311	0.9465	**0.8761**	0.9562	0.8897	0.8939
Image04	1.2817	1.1987	1.1788	1.2177	**1.1507**	1.2702	1.2119	1.1950
Image05	2.2565	2.1473	2.2298	2.0353	1.9759	2.2026	1.9745	**1.9432**
Image06	1.3383	1.1518	1.2577	1.0926	1.2867	1.0883	1.0131	**0.9750**
Image07	1.2586	1.1609	1.1060	1.0755	**1.0107**	1.2080	1.1252	1.1213
Image08	2.1716	1.6341	1.7945	1.7774	1.8203	1.7101	1.4984	**1.4329**
Image09	0.8860	0.8208	0.8419	0.8078	0.7935	0.8205	0.7796	**0.7736**
Image10	0.8452	0.8195	0.8558	0.8170	0.7911	0.8475	0.8005	**0.7902**
Image11	1.5881	1.3496	1.4200	1.3677	1.3768	1.4086	1.2814	**1.2508**
Image12	0.7489	0.6641	0.6823	0.6569	0.6663	0.6649	0.6261	**0.6202**
Image13	2.3943	2.2976	2.5256	2.4274	2.4413	2.4568	2.2369	**2.0849**
Image14	2.0050	1.8286	1.7552	1.8755	**1.6659**	1.8928	1.7252	1.7157
Image15	1.4960	1.3949	1.3969	1.4358	**1.3592**	1.4729	1.4038	1.3785
Image16	1.1752	0.9422	1.0296	0.8828	1.0934	0.8922	0.8472	**0.8306**
Image17	1.4391	1.3341	1.3937	1.3296	**1.2998**	1.3934	1.3269	1.3160
Image18	2.3071	2.3321	2.3421	2.2634	2.2438	2.4106	2.2796	**2.2193**
Image19	1.4089	1.1726	1.2415	1.2176	1.2403	1.2341	1.1383	**1.0986**
Image20	1.0446	0.9674	0.9880	1.0351	0.9467	1.0340	0.9456	**0.9229**
Image21	1.3317	1.2030	1.2934	1.2943	1.2460	1.3000	1.1681	**1.1197**
Image22	1.4972	1.4718	1.4323	**1.3643**	1.4096	1.5187	1.4541	1.4219
Image23	1.0353	0.9821	0.9564	0.9263	**0.9045**	1.0153	0.9854	0.9741
Image24	1.4904	1.5017	1.5346	1.4323	1.4187	1.5077	1.4201	**1.3697**
Average	1.4821	1.3463	1.3901	1.3583	1.3392	1.3951	1.2941	**1.2606**

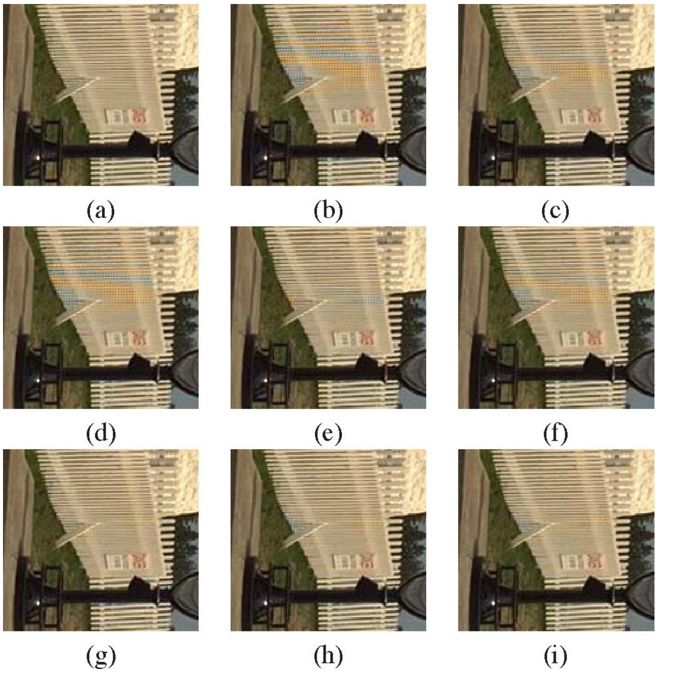


Fig. 12. Magnified subimages cut from the testing image No. 19. (a) Original full color image and the demosaiced images obtained from (b) Pei and Tam's algorithm. (c) Lu and Tan's algorithm. (d) Lukac and Plataniotis' algorithm. (e) Dubois' algorithm. (f) Lukac *et al.*'s algorithm. (g) Chung and Chan's algorithm. (h) Tsai and Song's algorithm. (i) Our proposed algorithm.

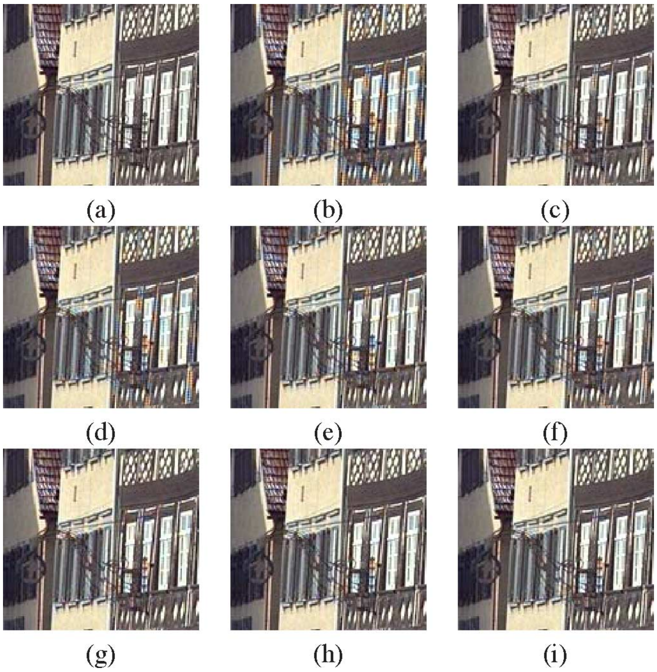


Fig. 13. Magnified subimages cut from the testing image No. 8. (a) Original full color image and the demosaiced images obtained from (b) Pei and Tam's algorithm. (c) Lu and Tan's algorithm. (d) Lukac and Plataniotis' algorithm. (e) Dubois' algorithm. (f) Lukac *et al.*'s algorithm. (g) Chung and Chan's algorithm. (h) Tsai and Song's algorithm. (i) Our proposed algorithm.

are shown in Table III. It is observed that the average execution-time of our proposed demosaicing algorithm is moderate when compared with the other seven algorithms. However, our proposed algorithm has the best demosaiced image quality performance among the eight algorithms.

Finally, in order to demonstrate the advantage of our proposed adaptive heterogeneity-projection, the mask-use ratio $\Re_{(N)}$ for demosaicing a mosaic image with size $\mathrm{X} \times \mathrm{Y}$ is defined as $\Re_{(N)} = T_{(N)}/XY \times 100\%$ where $T_{(N)}$ denotes the number of times which the mask with size N is adopted during the demosaicing processing. Based on 24 testing mosaic images, in average, the four mask-use ratios for N in $\{5, 7, 9, 11\}$ are $\Re_{(5)} = 28.58\%$, $\Re_{(7)} = 5.30\%$, $\Re_{(9)} = 4.28\%$, and

TABLE III
AVERAGE EXECUTION-TIME OF EIGHT DEMOSAICING ALGORITHMS FOR 24 TESTING MOSAIC IMAGES

Method	[37]	[26]	[29]	[9]	[34]	[5]	[44]	Proposed
Time(s)	0.607	1.704	1.303	1.406	1.492	1.052	1.469	1.442

$\Re_{(11)} = 61.84\%$, respectively. According to the above statistical data, $N = 11$ could be set for about 62% cases, but for the other 38% cases, smaller mask sizes could be more suitable for obtaining the heterogeneity-projection values.

V. CONCLUSION

Without demosaicing processing, this paper first proposes a new approach to extract more accurate gradient/edge information on mosaic images directly. Next, based on spectral-spatial correlation [44], a novel adaptive heterogeneity-projection with proper mask size for each pixel is presented. Combining the extracted gradient/edge information and the adaptive heterogeneity-projection values, a new edge-sensing demosaicing algorithm is presented. Some experimental results have been carried out to demonstrate the quality advantage in terms of CPSNR and S-CIELAB ΔE^*_{ab} of our proposed new demosaicing algorithm when compared with several recently published algorithms.

Besides combining the bilinear interpolation and the edge detection, it is an interesting research issue to replace the bilinear solution with another more accurate method, e.g., nonlinear solution, to have better gain of both edge-detection and demosaicing performance.

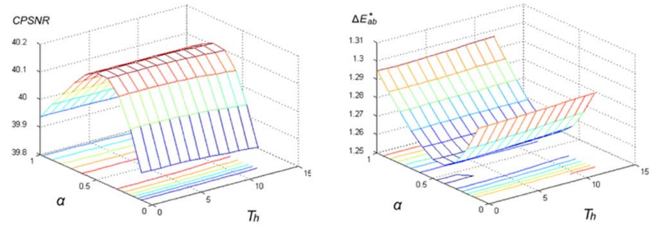
APPENDIX
DERIVATION OF THE SL-BASED QUAD-MASK

Combining (2) and (3), the four masks used to extract more accurate color gradient information on mosaic images directly can be obtained by the following derivation:

$$
\begin{aligned}
&\Delta I^H_{\mathrm{dm}}(i,j)\\
&= \left\{ \begin{array}{c} [L(i-1,j+1)+2L(i,j+1)+L(i+1,j+1)] \\ -[L(i-1,j-1)+2L(i,j-1)+L(i+1,j-1)] \end{array} \right\}\\
&= \frac{1}{16}\left\{ \begin{array}{c}
\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j+2)+I^C_{\mathrm{mo}}(i+2,j+2) \\ -I^C_{\mathrm{mo}}(i-2,j-2)-I^C_{\mathrm{mo}}(i+2,j-2) \end{array}\right] \\
+2\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j+1)+I^C_{\mathrm{mo}}(i+2,j+1) \\ -I^C_{\mathrm{mo}}(i-2,j-1)-I^C_{\mathrm{mo}}(i+2,j-1) \end{array}\right] \\
+4\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-1,j+2)+I^C_{\mathrm{mo}}(i+1,j+2) \\ -I^C_{\mathrm{mo}}(i-1,j-2)-I^C_{\mathrm{mo}}(i+1,j-2) \end{array}\right] \\
+6\times[I^C_{\mathrm{mo}}(i,j+2)-I^C_{\mathrm{mo}}(i,j-2)] \\
+8\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-1,j+1)+I^C_{\mathrm{mo}}(i+1,j+1) \\ -I^C_{\mathrm{mo}}(i-1,j-1)-I^C_{\mathrm{mo}}(i+1,j-1) \end{array}\right] \\
+12\times[I^C_{\mathrm{mo}}(i,j+1)-I^C_{\mathrm{mo}}(i,j-1)]
\end{array}\right\}\\
&\Delta I^V_{\mathrm{dm}}(i,j)\\
&= \left\{ \begin{array}{c} [L(i+1,j-1)+2L(i+1,j)+L(i+1,j+1)] \\ -[L(i-1,j-1)+2L(i-1,j)+L(i-1,j+1)] \end{array} \right\}\\
&= \frac{1}{16}\left\{ \begin{array}{c}
\left[\begin{array}{c} I^C_{\mathrm{mo}}(i+2,j-2)+I^C_{\mathrm{mo}}(i+2,j+2) \\ -I^C_{\mathrm{mo}}(i-2,j-2)-I^C_{\mathrm{mo}}(i-2,j+2) \end{array}\right] \\
+2\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i+1,j-2)+I^C_{\mathrm{mo}}(i+1,j+2) \\ -I^C_{\mathrm{mo}}(i-1,j-2)-I^C_{\mathrm{mo}}(i-1,j+2) \end{array}\right] \\
+4\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i+2,j-1)+I^C_{\mathrm{mo}}(i+2,j+1) \\ -I^C_{\mathrm{mo}}(i-2,j-1)-I^C_{\mathrm{mo}}(i-2,j+1) \end{array}\right] \\
+6\times[I^C_{\mathrm{mo}}(i+2,j)-I^C_{\mathrm{mo}}(i-2,j)] \\
+8\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i+1,j-1)+I^C_{\mathrm{mo}}(i+1,j+1) \\ -I^C_{\mathrm{mo}}(i-1,j-1)-I^C_{\mathrm{mo}}(i-1,j+1) \end{array}\right] \\
+12\times[I^C_{\mathrm{mo}}(i+1,j)-I^C_{\mathrm{mo}}(i-1,j)]
\end{array}\right\}\\
&\Delta I^{\frac{\pi}{4}}_{\mathrm{dm}}(i,j)\\
&= \left\{ \begin{array}{c} [L(i-1,j)+2L(i-1,j+1)+L(i,j+1)] \\ -[L(i+1,j)+2L(i+1,j-1)+L(i,j-1)] \end{array} \right\}\\
&= \frac{1}{16}\left\{ \begin{array}{c}
\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j-1)+I^C_{\mathrm{mo}}(i+1,j+2) \\ -I^C_{\mathrm{mo}}(i-1,j-2)-I^C_{\mathrm{mo}}(i+2,j+1) \end{array}\right] \\
+2\times[I^C_{\mathrm{mo}}(i-2,j+2)-I^C_{\mathrm{mo}}(i+2,j-2)] \\
+4\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j)+I^C_{\mathrm{mo}}(i,j+2) \\ -I^C_{\mathrm{mo}}(i+2,j)-I^C_{\mathrm{mo}}(i,j-2) \end{array}\right] \\
+5\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j+1)+I^C_{\mathrm{mo}}(i-1,j+2) \\ -I^C_{\mathrm{mo}}(i+2,j-1)-I^C_{\mathrm{mo}}(i+1,j-2) \end{array}\right] \\
+8\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-1,j)+I^C_{\mathrm{mo}}(i,j+1) \\ -I^C_{\mathrm{mo}}(i+1,j)-I^C_{\mathrm{mo}}(i,j-1) \end{array}\right] \\
+14\times[I^C_{\mathrm{mo}}(i-1,j+1)-I^C_{\mathrm{mo}}(i+1,j-1)]
\end{array}\right\}\\
&\Delta I^{\frac{-\pi}{4}}_{\mathrm{dm}}(i,j)\\
&= \left\{ \begin{array}{c} [L(i-1,j)+2L(i-1,j-1)+L(i,j-1)] \\ -[L(i+1,j)+2L(i+1,j+1)+L(i,j+1)] \end{array} \right\}\\
&= \frac{1}{16}\left\{ \begin{array}{c}
\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j+1)+I^C_{\mathrm{mo}}(i+1,j-2) \\ -I^C_{\mathrm{mo}}(i-1,j+2)-I^C_{\mathrm{mo}}(i+2,j-1) \end{array}\right] \\
+2\times[I^C_{\mathrm{mo}}(i-2,j-2)-I^C_{\mathrm{mo}}(i+2,j+2)] \\
+4\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j)+I^C_{\mathrm{mo}}(i,j-2) \\ -I^C_{\mathrm{mo}}(i+2,j)-I^C_{\mathrm{mo}}(i,j+2) \end{array}\right] \\
+5\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-2,j-1)+I^C_{\mathrm{mo}}(i-1,j-2) \\ -I^C_{\mathrm{mo}}(i+2,j+1)-I^C_{\mathrm{mo}}(i+1,j+2) \end{array}\right] \\
+8\times\left[\begin{array}{c} I^C_{\mathrm{mo}}(i-1,j)+I^C_{\mathrm{mo}}(i,j-1) \\ -I^C_{\mathrm{mo}}(i+1,j)-I^C_{\mathrm{mo}}(i,j+1) \end{array}\right] \\
+14\times[I^C_{\mathrm{mo}}(i-1,j-1)-I^C_{\mathrm{mo}}(i+1,j+1)]
\end{array}\right\}.
\end{aligned}
$$

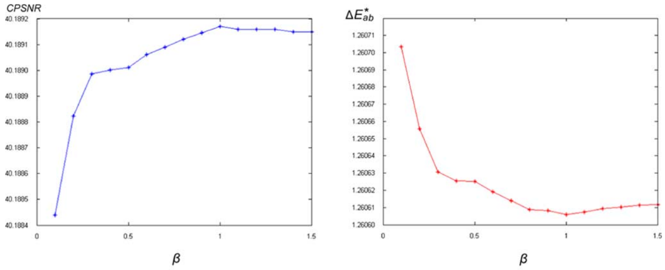
APPENDIX
DETERMINATION OF THRESHOLD T_h AND PARAMETERS β AND α

Since the image quality performance of our proposed demosaicing algorithm is influenced by the threshold T_h and parameters β and α, how to select the best choices of T_h, β and α should be discussed. When $\beta \geq 1$, our experiments reveal that the CPSNR surfaces and the S-CIELAB ΔE^*_{ab} surfaces are quite similar to Fig. 14(a) and (b), respectively. Based on the peak of Fig. 14(a) and the valley of Fig. 14(b), the best choices of T_h and α are selected as $T_h = 7$ and $\alpha = 0.5$. After determining the



(a) (b)

Fig. 14. Two demosaiced image quality surfaces in terms of T_h and α for $\beta \geq 1$. (a) The CPSNR surface. (b) The S-CIELAB ΔE^*_{ab} surface.



(a) (b)

Fig. 15. Two demosaiced image quality curves in terms of β for $T_h = 7$ and $\alpha = 0.5$. (a) The CPSNR curve. (b) The S-CIELAB ΔE^*_{ab} curve.

threshold $T_h = 7$ and the parameter $\alpha = 0.5$, by the same argument, the best choice of the parameter β is selected as $\beta = 1$ based on the peak of Fig. 15(a) and the valley of Fig. 15(b).

References

[1] J. E. Adams and J. F. Hamilton, "Adaptive Color Plan Interpolation in Single Sensor Color Electric Camera," U.S. Patent 5 506 619, 1996.

[2] D. Alleysson, S. Susstrunk, and J. Herault, "Linear demosaicing inspired by the human visual system," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 439–449, Apr., 2005.

[3] B. E. Bayer, "Color Imaging Array," U.S. Patent 3 971 065, 1976.

[4] S. C. Cheng and T. L. Wu, "Subpixel edge detection of color images by principal axis analysis and moment-preserving principle," *Pattern Recognit.*, vol. 38, no. 4, pp. 527–537, 2005.

[5] K. H. Chung and Y. H. Chan, "Color demosaicing using variance of color differences.," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2944–2955, Oct. 2006.

[6] K. H. Chung and Y. H. Chan, "A low-complexity joint color demosaicking and zooming algorithm for digital camera," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1705–1715, Jul. 2007.

[7] K. L. Chung, W. J. Yang, W. M. Yan, and C. C. Wang, "Efficient high-quality demosaicing algorithm for color filter array," Dept. Comput. Sci. Inf. Eng., National Taiwan Univ. Science and Technology, Tech. Rep. NTUST-CSIE-07-04, 2007 [Online]. Available: http://www.csie.ntust.edu.tw/treport/tr_04.pdf

[8] D. R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," U.S. Patent 4 642 678, 1987.

[9] E. Dubois, "Frequency-domain methods for demosaicking of bayer-sampled color images," *IEEE Signal Process. Lett.*, vol. 12, no. 12, pp. 847–850, Dec. 2005.

[10] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and W. G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1454–1466, Oct. 2001.

[11] J. Fan, W. G. Aref, M. S. Hacid, and A. K. Elmagarmid, "An Improved automatic isotropic color edge detection technique," *Pattern Recognit. Lett.*, vol. 22, no. 13, pp. 1419–1429, 2001.

[12] W. T. Freeman, "Median Filter for Reconstructing Missing Color Samples," U.S. Patent 4 724 395, 1988.

[13] R. Gonzalez and R. Woods, *Digital Image Processing*. New York: Addison Wesley, 1992.

[14] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 997–1013, Nov. 2002.

[15] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: Color filter array interpolation," *IEEE Signal Process. Mag.*, vol. 22, no. 1, pp. 44–54, Jan. 2005.

[16] Y. Hel-Or and D. Keren, "Image demosaicking method utilizing directional smoothing," U.S. Patent 6 404 918, 2002.

[17] H. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients," U.S. Patent 5 382 976, 1995.

[18] K. Hirakawa and T. W. Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 360–369, Mar. 2005.

[19] R. W. G. Hunt, *Measuring Colour*, 2nd ed. Chichester, U.K.: Ellis Horwood, 1995.

[20] D. Keren and M. Osadchy, "Restoring subsampled color images," *Mach. Vis. Appl.*, vol. 11, no. 4, pp. 197–202, 1999.

[21] R. Kimmel, "Demosaicing: Image reconstruction from color CCD samples," *IEEE Trans. Image Process.*, vol. 8, no. 9, pp. 1221–1228, Sep. 1999.

[22] J. Z. C. Lai and Y. C. Liaw, "Improvement of interpolated color filter array image using modified mean-removed vector quantization," *Pattern Recognit. Lett.*, vol. 26, no. 8, pp. 1047–1058, 2005.

[23] C. A. Laroche and M. A. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent 5 373 322, Dec. 1994.

[24] W. Lee, S. Lee, and J. Kim, "Cost-effective color filter array demosaicing using spatial correlation," *IEEE Trans. Consum. Electron.*, vol. 52, no. 2, pp. 547–554, Feb. 2006.

[25] X. Li, "Demosaicing by successive approximation," *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 370–379, Mar. 2005.

[26] W. Lu and Y. P. Tan, "Color filter array demosaicking: New method and performance measures," *IEEE Trans. Image Process.*, vol. 12, no. 10, pp. 1194–1210, Oct. 2003.

[27] R. Lukac, K. Martin, and K. N. Plataniotis, "Demosaicked image post-processing using local color ratios," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 6, pp. 914–920, Jun. 2004.

[28] R. Lukac, K. Martin, and K. N. Plataniotis, "Digital camera zooming based on unified CFA image processing steps," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 15–24, Jan. 2004.

[29] R. Lukac and K. N. Plataniotis, "Normalized color-ratio modeling for CFA interpolation," *IEEE Trans. Consum. Electron.*, vol. 50, no. 2, pp. 737–745, Feb. 2004.

[30] R. Lukac and K. N. Plataniotis, "Digital zooming for color filter array based image sensors," *Real-Time Image*, vol. 11, no. 2, pp. 129–138, 2005.

[31] R. Lukac, K. N. Plataniotis, and D. Hatzinakos, "Color image zooming on the Bayer pattern," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 11, pp. 1475–1492, Nov. 2005.

[32] R. Lukac and K. N. Plataniotis, "Color filter arrays: Design and performance analysis," *IEEE Trans. Consum. Electron.*, vol. 51, no. 4, pp. 1260–1267, Apr. 2005.

[33] R. Lukac and K. N. Plataniotis, "A robust, cost-effective postprocessor for enhancing demosaicked camera images," *Real-Time Image*, vol. 11, no. 2, pp. 139–150, 2005.

[34] R. Lukac, K. N. Plataniotis, D. Hatzinakos, and M. Aleksic, "A new CFA interpolation framework," *Signal Process.*, vol. 86, no. 7, pp. 1559–1579, 2006.

[35] R. Lukac and K. N. Plataniotis, , P. W. Hawkes, Ed., "A Taxonomy of Color Image Filtering and Enhancement Solutions," in *Advances in Imaging and Electron Physics*. New York: Elsevier/Academic, 2006, vol. 140, pp. 187–264.

[36] D. D. Muresan and T. W. Parks, "Demosaicing using optimal recovery," *IEEE Trans. Image Process.*, vol. 14, no. 2, pp. 267–278, Feb. 2005.

[37] S. C. Pei and I. K. Tam, "Effective color interpolation in CCD color filter arrays using signal correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 6, pp. 503–513, Jun. 2003.

[38] T. Sakamoto, C. Nakanishi, and T. Hase, "Software pixel interpolation for digital still camera suitable for a 32-bit MCU," *IEEE Trans. Consum. Electron.*, vol. 44, no. 4, pp. 1342–1352, Apr. 1998.

[39] J. Scharcanski and A. N. Venetsanopoulos, "Edge detection of color images using directional operators," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 397–401, Feb. 1997.

[40] C. Y. Su, "Highly effective iterative demosaicing using weighted-edge and color-difference interpolations," *IEEE Trans. Consum. Electron.*, vol. 52, no. 2, pp. 639–645, Feb. 2006.

[41] C. Theoharatos, G. Economou, and S. Fotopoulos, "Color edge detection using the minimal spanning tree," *Pattern Recognit.*, vol. 38, no. 4, pp. 603–606, 2005.

[42] P. J. Toivanen, J. Ansamaki, J. P. S. Parkkinen, and J. Mielikainen, "Edge detection in multispectral images using the selforganizing map," *Pattern Recognit. Lett.*, vol. 24, no. 16, pp. 2987–2993, 2003.

[43] P. E. Trahanias and A. N. Venetsanopoulos, "Color edge detection using order statistics," *IEEE Trans. Image Process.*, vol. 2, no. 2, pp. 259–264, Feb. 1993.

[44] C. Y. Tsai and K. T. Song, "Heterogeneity-projection hard-decision color interpolation using spectral-spatial correlation," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 78–91, Jan. 2007.

[45] X. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicing for digital cameras (patent pending)," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1263–1274, Sep. 2004.

[46] L. Zhang and X. Wu, "Color demosaicking via directional linear minimum mean square-error interpolation," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2167–2178, Dec. 2005.

[47] L. Zhang and D. Zhang, "A joint demosaicking-zooming scheme for single chip digital color cameras," *Comput. Vis. Image Understand.*, vol. 107, no. 1–2, pp. 14–25, 2007.

[48] S. Y. Zhu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Comprehensive analysis of edge detection in color image processing," *Opt. Eng.*, vol. 38, no. 4, pp. 612–625, 1999.

[49] [Online]. Available: http://www.site.uottawa.ca/~edubois/demosaicking/

[50] [Online]. Available: http://140.118.175.164/WJYang/paper/Demosaic/

Kuo-Liang Chung (SM'01) received the B.S., M.S., and Ph.D. degrees in computer science and information engineering from the National Taiwan University, Taiwan, R.O.C., in 1982, 1984, and 1990, respectively.

He was a visiting scholar at the University of Washington, Seattle, in the summer of 1999. He was the Head of the Department of Computer Science and Information Engineering at the National Taiwan University of Science and Technology from 2003 to 2006. He is now a Professor. He was the Executive Editor of the *Journal of the Chinese Institute of Engineers* from 1996 to 1998.

Dr. Chung received the Distinguished Engineering Professor Award from Chinese Institute of Engineers in 2001; the Distinguished Research Award (2004 to 2007) from the National Science Council, Taiwan; and the best paper award from the Society of Computer Vision, Graphics, and Image Processing (Taiwan) in 2007. He is a fellow of IET. He has authored and coauthored over 130 publications in books and international well-known journals. His research interests include image/video compression, image/video processing, pattern recognition, artificial intelligence, algorithms, and multimedia applications.

Wei-Jen Yang received the B.S. degree in computer science and information engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C., in 2004. He is currently pursuing the Ph.D. degree in computer science and information engineering at the National Taiwan University, Taipei.

His research interests include color image processing, digital camera image processing, image/video compression, computer vision, pattern recognition, and algorithms.

Wen-Ming Yan received the B.S. and M.S. degrees in mathematics from the National Taiwan University, Taipei, Taiwan, R.O.C.

Now, he is an Associate Professor of computer science and information engineering at the National Taiwan University. His research interests include scientific computation, image compression, computer vision, numerical linear algebra, image processing, coding theory, and algorithms.

Chung-Chou Wang received the B.S. degree in mathematics from the National Central University, Taoyuan, Taiwan, R.O.C., in 2006, and the M.S. degree in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 2008.

His research interests include digital camera image processing, computer vision, numerical linear algebra, pattern recognition, and algorithms.