# Low-complexity color demosaicing algorithm based on integrated gradients

**King-Hong Chung**
**Yuk-Hee Chan**
The Hong Kong Polytechnic University
Centre for Multimedia Signal Processing
Department of Electronic and Information Engineering
Hong Kong
E-mail: enyhchan@polyu.edu.hk

**Abstract.** *Color demosaicing is critical for digital cameras, because it converts a Bayer sensor mosaic output to a full color image, which determines the output image quality of the camera. In this work, an efficient decision-based demosaicing method is presented. This method exploits a new edge-sensing measure called integrated gradient (IG) to effectively extract gradient information in both color intensity and color difference domains simultaneously. This measure is reliable and supports full resolution, which allows one to interpolate the missing samples along an appropriate direction and hence directly improves the demosaicing performance. By sharing it in different demosaicing stages to guide the interpolation of various color planes, it guarantees the consistency of the interpolation direction in different color channels and saves the effort required to repeatedly extract gradient information from intermediate interpolation results at different stages. An IG-based green plane enhancement is also proposed to further improve the method's efficiency. Simulation results confirm that the proposed demosaicing method outperforms up-to-date demosaicing methods in terms of output quality at a complexity of around 80 arithmetic operations per pixel. © 2010 SPIE and IS&T.* [DOI: 10.1117/1.3432484]

## 1 Introduction

Most digital cameras use a single charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) image sensor to capture a scene. To reduce the cost, a Bayer color filter array (CFA),[1] as shown in Fig. 1, is coated over the sensor such that only one of the three primary colors (R, G, and B) is sampled at each pixel location. The acquired mosaic image (Bayer image) is then converted to a full color image by interpolating the two missing color samples of a pixel. This process is called demosaicing or color interpolation. It is critical to a digital camera because it determines the image quality of the camera output.

Early demosaicing methods such as bilinear interpolation[2] apply interpolation techniques for grayscale images to each color channel separately. These simple methods generally introduce severe color artifacts such as blurring, false color, and zipper effect around edges. To provide better demosaicing performance, many advanced demosaicing methods[3–17] have been proposed. Most of them are heuristic in a way that they do not interpolate the missing samples by solving a mathematically defined optimization problem. Methods reported in Refs. 4–11 and Refs. 12–17 are, respectively, typical examples of heuristic and nonheuristic demosaicing methods.

Recently, the well-known directional Laplacian filter[7] has been proven to be a good approximation of the optimal interpolator for Bayer images in one dimension.[18] Accordingly, many demosaicing methods try to orient the filter horizontally, vertically, or sometimes diagonally to provide better local interpolation results with fewer artifacts. As a consequence, a decision has to be made to determine the filter orientation. This can be done based on the color intensity gradient[7] or the smooth property of the image's color difference (green-to-red and green-to-blue) signals.[18–22] In particular, Hirakawa and Parks select the directions with the least misguidance level of color artifacts presented in images,[18] while Wu and Zhang adopt Fisher's discriminant technique to determine the optimal direction that preserves the highest correlation of color gradients.[19] In Ref. 20, Chung and Chan make the decisions by comparing the variances of color differences along window axes. In contrast, Menon, Andriani, and Calvagno[21] compute the local horizontal and vertical color difference gradients to determine the interpolation direction, whereas Tsai and Song proposed a hard-decision scheme that selects the direction by comparing the horizontal and vertical heterogeneity values of the pixels.[22]

It is interesting to find that, though early demosaicing methods[7,9,11,23,24] try to extract gradient information from color intensity (CI) domains to guide the interpolation in corresponding color planes, recent demosaicing methods[19,21,22] generally put their focus on the color differ-

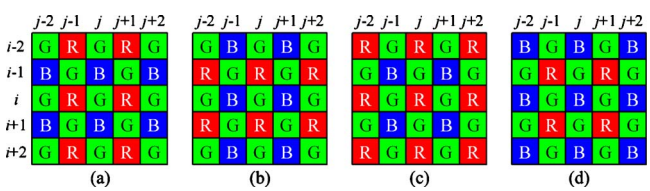


**Fig. 1** Four 5×5 regions of Bayer CFA pattern having their centers at (a) and (b) green, (c) red, and (d) blue Bayer samples.

Paper 09104SSR received Jun. 11, 2009; revised manuscript received Mar. 19, 2010; accepted for publication Apr. 12, 2010; published online Jun. 1, 2010.

1017-9909/2010/19(2)/021104/15/$25.00 © 2010 SPIE and IS&T.

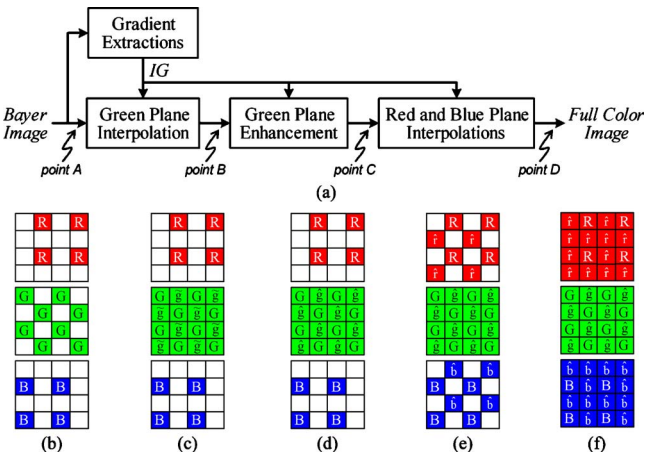


**Fig. 2** The spatial arrangement of the proposed demosaicing method's intermediate results: (a) work flow of the method, (b) raw sensor output Bayer image at point A, (c) green plane interpolation result at point B, (d) green plane enhancement result at point C, (e) intermediate interpolation result during red and blue plane interpolations, and (f) final interpolation result at point D.

ence (CD) domains and exploit the interchannel spectral correlation to guide the interpolation. In either approach, the information extracted from the color difference and the color intensity domains is not properly balanced. This bias may result in misleading information that guides one to interpolate samples along a wrong direction.

Another observation is that, to improve the output quality, some demosaicing methods such as Refs. 9 and 11 re-extract gradient information from intermediate interpolation results obtained at different stages. This adapt-to-new-information approach is great but it does not always work properly. For example, in texture areas where edges are locally ambiguous along the horizontal and vertical directions, the gradient information extracted at different stages can be mutually contradictory and makes the situation confusing. Besides, gradient estimation is generally computationally expensive and hence repeated estimation increases the complexity a lot.

Color artifacts in high frequency areas are commonly found in the outputs of various demosaicing methods. The most common solution for this problem is to introduce a postprocessing step[10,25] to suppress the artifacts after demosaicing. Nevertheless, as all missing samples have to be processed again in the postprocessing stage, it seems to be inefficient from a system point of view. Besides, as the interpolation of the red and blue planes is carried out before the enhancement of the green plane, one cannot make use of the enhanced green plane as a "better" reference to interpolate the red and blue planes. This also lowers the efficiency.

Triggered by the aforementioned observations, we put our effort toward tackling the corresponding problems. In this work, an efficient decision-based demosaicing method is proposed to reconstruct color images from Bayer images. It aims at producing high quality output images at a low computation cost. Figure 2 shows a flow diagram of the proposed demosaicing method. In this method, a new directional edge-sensing parameter called integrated gradient (IG), which extracts gradient information in both color intensity (CI) and color difference (CD) domains simultaneously, is defined for being shared in various stages throughout the demosaicing process to interpolate the color channels. This IG is not only used as an edge detector to determine the interpolation direction when interpolating a green sample, but is also used to adjust the coefficients of two spatial-variant interpolators when estimating the missing red and blue samples. In addition, a green plane enhancement that works with the IG is introduced to further improve the method's performance. Simulation results confirmed that the proposed demosaicing method provides superior output quality at a comparatively low computation cost.

The remainder of this work is organized as follows. In Sec. 2, gradients in various domains are introduced and their uses are discussed. They form basic elements to construct the proposed IG. The details are discussed in Sec. 3. Section 4 presents the details of the proposed demosaicing method. Section 5 provides some experimental results of the proposed method for comparison study, while Section 6 shows the computational complexity of the proposed method. Finally, a conclusion is given in Section 7.

## 2 Gradients for Edge Detection

Since our human visual system is sensitive to edge structures, many demosaicing methods try to avoid doing interpolation across edges. To achieve the goal, gradients are estimated in various directions at each pixel to guide the interpolation along an edge.

The CI gradient is commonly used to identify edges based on the fact that the intensity gradient perpendicular to the edge direction is large.[3,24] Since R, G, and B samples in a Bayer CFA image are interlaced, the horizontal and vertical CI gradients at a particular pixel $(i,j)$ are, respectively, approximated by

$$\bar{\delta}^H(i,j) = |Z(i,j-1) - Z(i,j+1)|, \tag{1}$$

and

$$\bar{\delta}^V(i,j) = |Z(i-1,j) - Z(i+1,j)|, \tag{2}$$

in general,[3] where $Z(m,n)$ denotes the valid Bayer sample at pixel $(m,n)$. The approximation result can be very misleading in regions containing fine details or textures, as the approximation is based on two samples not involving $(i,j)$. Accordingly, interpolation may be done along a wrong direction.

Supplementary information can be used to have a better decision in a texture region. In practice, the color difference signals are locally constant along an edge or a line.[22] In consequence, the CD gradient along the corresponding direction should be close to zero. Accordingly, one can make a hypothesis test to determine if a pixel should be interpolated along a particular direction.

In particular, one can assume that the pixel, say $(i,j)$, is on a horizontal line and, based on the assumption, estimate its color difference as

$$C_d^H(i,j) = Z(i,j) - \frac{1}{2}[Z(i,j-1) + Z(i,j+1)]. \quad (3)$$

Note that the estimation can be highly accurate when the assumption is valid, as averaging the available Bayer samples of $(i,j)$'s neighboring pixels along the line provides a good estimate of the corresponding missing color sample in $(i,j)$. Similarly, $C_d^H(i,j\pm1)$ can be estimated under the same assumption, and the horizontal CD gradient can then be estimated as

$$\tilde{\delta}^H(i,j) = \frac{1}{2}\sum_{l=\pm1} |C_d^H(i,j) + C_d^H(i,j+l)|. \quad (4)$$

Note that, due to the interlaced color sampling arrangement in a Bayer CFA image, $|C_d^H(i,j)+C_d^H(i,j\pm1)|$ actually provides the absolute difference between two adjacent pixels in the same color difference plane. For example, for the local region shown in Fig. 1(c), $C_d^H(i,j)$ is the red-to-green difference, while $C_d^H(i,j\pm1)$ is the green-to-red difference. Hence, $C_d^H(i,j)+C_d^H(i,j+l)$ for $l=\pm1$ are estimates of the gradient in the red-to-green difference plane already. $\tilde{\delta}^H(i,j)$ is a strengthened estimate that is the mean absolute value of the two estimates. The absolute value nature of the items prevents them from canceling each other in averaging. Obviously, a large value of $\tilde{\delta}^H(i,j)$ means that the assumption is invalid.

Similarly, one can make another assumption that $(i,j)$ is on a vertical line and, based on the new assumption, estimate the color difference and the vertical CD gradient respectively as

$$C_d^V(i,j) = Z(i,j) - \frac{1}{2}[Z(i-1,j) + Z(i+1,j)]$$

and

$$\tilde{\delta}^V(i,j) = \frac{1}{2}\sum_{l=\pm1} |C_d^V(i,j) + C_d^V(i+l,j)|.$$

By comparing it with the aforementioned horizontal CD gradient, one can tell which assumption is more likely to be valid and determine the interpolation direction accordingly.

Both CI and CD gradients can be used to guide the interpolation direction, and they make different contributions. In Sec. 3, gradients in different color intensities and color difference domains are combined to form an integrated gradient for better guidance performance.

## 3 Extraction of Integrated Gradient

The proposed demosaicing method uses a measure called the integrated gradient (IG) to guide the interpolation. This IG is a combination of the gradients in both the CI and CD domains, which provides more information for one to reach a better decision in selecting the interpolation direction. This section presents the definition of this IG and its rationale.

Before we define IG, let us first define an intermediate measure called the weaker integrated gradient (WIG). It is called this because, compared with IG, it extracts information from fewer CI and CD planes and hence provides less information for edge detection.

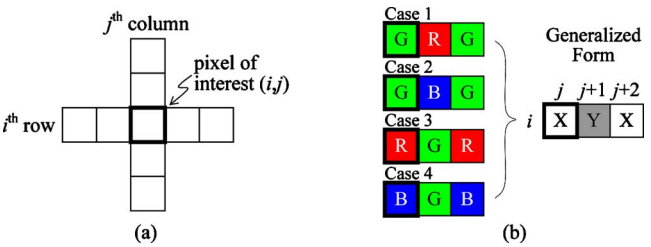


**Fig. 3** (a) A cross template for computing WIGs. (b) All possible Bayer patterns covered for computing the eastbound WIG of a pixel and their generalized form.

Figure 3(a) shows a cross template for extracting Bayer samples to calculate the WIG of the pixel located at the template center. The template has four extensions. The Bayer samples covered in a particular extension of the template (including the center) are used to compute the WIG along the corresponding direction. No matter which possible $5\times5$ Bayer pattern is concerned (see Fig. 1), after rotating the Bayer sample patterns covered by the northbound, the westbound, and the southbound extensions of the cross template by 90, 180, and 270 deg, respectively, they are all in the same standard pattern form shown in Fig. 3(b) as those patterns covered by the eastbound extension. Hence, as long as the eastbound WIG is defined, the westbound, northbound, and southbound WIGs can also be defined in the same manner. To save effort, here we just define the eastbound WIG.

In the generalized form shown in Fig. 3(b), $X$ and $Y$ denote the colors of the corresponding Bayer samples in a concerned pattern. Let $\bar{\delta}_X^E(i,j)$ and $\tilde{\delta}_{XY}^E(i,j)$ be respectively the eastbound CI and CD gradients of pixel $(i,j)$. In the notations, superscript $E$ and subscripts $XY$ and $X$ respectively denote the direction of the WIG, the involved CD plane, and the involved CI plane.

The eastbound CI gradient of pixel $(i,j)$, $\bar{\delta}_X^E(i,j)$, measures the extent of eastbound CI change and is defined on the color channel that contains the Bayer sample of pixel $(i,j)$. Specifically, we have

$$\bar{\delta}_X^E(i,j) = \bar{\delta}^H(i,j+1) = |X(i,j) - X(i,j+2)|. \quad (5)$$

Note $X(i,j)$ is now the known Bayer sample at position $(i,j)$. The eastbound CI gradient is used to identify edges, and hence only the gradient magnitude is concerned. There is likely an edge if $\bar{\delta}_X^E(i,j)$ is large.

The eastbound CD gradient $\tilde{\delta}_{XY}^E(i,j)$ is introduced to provide supplementary edge detection information by evaluating the CD change of two successive pixels along the same eastbound direction. In formulation, it is defined as

$$\tilde{\delta}^E_{XY}(i,j) = \frac{1}{2}\sum_{l=0,1} |d_{XY}(i,j+l) - d_{XY}(i,j+1+l)|, \quad (6)$$

where $d_{XY}(m,n)$ represents the $X$-$Y$ CD value at position $(m,n)$. $d_{XY}(m,n)$ has to be estimated based on the neighboring Bayer samples. When it is estimated as

$$d_{XY}(m,n) = \begin{cases} C^H_d(m,n) & \text{if } (m,n) \text{ contains} \\ & \text{a } X \text{ Bayer sample} \\ -C^H_d(m,n) & \text{if } (m,n) \text{ contains} \\ & \text{a } Y \text{ Bayer sample} \end{cases}, \quad (7)$$

Eq. (6) can be rewritten as $\tilde{\delta}^E_{XY}(i,j) = \tilde{\delta}^H(i,j+1)$. In our proposed approach, this estimate is further low-pass filtered with a three-point averaging filter to remove the potential high frequency noise before being used as $d_{XY}(m,n)$. The details are discussed in Sec. 3.1.

With the eastbound CI and CD gradients respectively defined in Eqs. (5) and (6), the eastbound WIG for the pixel $(i,j)$ shown in Fig. 3(b) is then defined as

$$\delta^E_{XY}(i,j) = \bar{\delta}^E_X(i,j) + \alpha\tilde{\delta}^E_{XY}(i,j), \quad (8)$$

where $\alpha$ is a weighting factor used to control the contribution of the two gradients. The determination of its value is discussed later.

The WIG of pixel $(i,j)$ only provides the edge information extracted from the $X$ plane and the $X$-$Y$ plane. It might happen that, over the edge to be detected, there is only a sharp change in the $Y$ plane or another CD plane. In that case, WIG fails to detect the edge. To solve this problem, more CI or CD planes should be included in the detection. One of the possible solutions is to combine the eastbound WIGs of pixels $(i-1,j)$, $(i,j)$ and $(i+1,j)$ to form the eastbound IG of pixel $(i,j)$.

As an example, for the case shown in Fig. 1(b), the eastbound IG of $(i,j)$ can be defined as

$$\begin{aligned}\Delta^E(i,j) &= 2\delta^E_{GB}(i,j) + \delta^E_{RG}(i-1,j) + \delta^E_{RG}(i+1,j) \\ &= [2\bar{\delta}^E_G(i,j) + \bar{\delta}^E_R(i-1,j) + \bar{\delta}^E_R(i+1,j)] + \alpha[2\tilde{\delta}^E_{GB}(i,j) \\ &\quad + \tilde{\delta}^E_{RG}(i-1,j) + \tilde{\delta}^E_{RG}(i+1,j)]. \end{aligned} \quad (9)$$

Note that $\delta^E_{GB}(i,j)$ is weighted by 2 to balance the contribution of the CD (CI) gradients extracted from different CD (CI) planes to $\Delta^E(i,j)$. By so doing, CD gradients from R-G and G-B planes have equal votes in Eq. (9). Similarly, CI gradients from R and G planes also have equal votes.

However, this definition of IG makes IGs along different directions incompatible. Though the Bayer patterns shown in Figs. 1(c) and 1(d) are symmetric in all four $(E,W,S,N)$ directions, the ones shown in Figs. 1(a) and 1(b) are not. To obtain the northbound IG of the pixel $(i,j)$ shown in Fig. 1(b), one can rotate Fig. 1(b) clockwise by 90 deg and then compute the eastbound IG of the rotated version with Eq. (9). In other words, we have

$$\begin{aligned}\Delta^N(i,j) &= 2\delta^N_{GR}(i,j) + \delta^N_{BG}(i,j-1) + \delta^N_{BG}(i,j+1) \\ &= [2\bar{\delta}^N_G(i,j) + \bar{\delta}^N_B(i,j-1) + \bar{\delta}^N_B(i,j+1)] \\ &\quad + \alpha[2\tilde{\delta}^N_{GR}(i,j) + \tilde{\delta}^N_{BG}(i,j-1) + \tilde{\delta}^N_{BG}(i,j+1)], \end{aligned} \quad (10)$$

for the case shown in Fig. 1(b). From Eqs. (9) and (10), one can see that $\Delta^E(i,j)$ carries CI gradient information from the R plane but not the B plane, while $\Delta^N(i,j)$ does the opposite. Hence, they are not compatible and it does not make sense to compare them.

By considering the aforementioned compatibility constraint, we regulate the definition of IG to eliminate the items that cause the incompatibility and, accordingly, modify Eqs. (9) and (10) as

$$\begin{aligned}\Delta^E(i,j) &= \bar{\delta}^E_G(i,j) + \alpha[2\tilde{\delta}^E_{GB}(i,j) + \tilde{\delta}^E_{RG}(i-1,j) \\ &\quad + \tilde{\delta}^E_{RG}(i+1,j)], \end{aligned} \quad (11)$$

and

$$\begin{aligned}\Delta^N(i,j) &= \bar{\delta}^N_G(i,j) + \alpha[2\tilde{\delta}^N_{GR}(i,j) \\ &\quad + \tilde{\delta}^N_{BG}(i,j-1) + \tilde{\delta}^N_{BG}(i,j+1)], \end{aligned} \quad (12)$$

for the case shown in Fig. 1(b). Note that $\bar{\delta}^E_G(i,j)$ and $\bar{\delta}^N_G(i,j)$'s original scaling factor of 2 is also eliminated in Eqs. (11) and (12). However, this can be compensated for by adjusting weighting factor $\alpha$ at the end.

Due to the absolute value nature of $\tilde{\delta}^E_{XY}(i,j)$ [see Eq. (6)] and the fact that $d_{XY}(i,j) = -d_{YX}(i,j)$, we have $\tilde{\delta}^E_{XY}(i,j) = \tilde{\delta}^E_{YX}(i,j)$. In other words, both $\Delta^E(i,j)$ and $\Delta^N(i,j)$ now carry the information extracted from the same G, G-R, and G-B planes, and hence they are compatible. As a matter of fact, with this regulated definition, all four IGs of any particular pixel are compatible.

To maintain this compatibility, though theoretically a full color image of three color components contains three CI and three CD planes, we do not further include more CI and CD planes in the definition of IG. This final definition of IG is used in the proposed demosaicing method.

Similarly, the eastbound IGs of pixel $(i,j)$ in the cases shown in Figs. 1(a), 1(c), and 1(d) are, respectively, defined as

$$\begin{aligned}\Delta^E(i,j) &= \bar{\delta}^E_G(i,j) + \alpha[2\tilde{\delta}^E_{GR}(i,j) + \tilde{\delta}^E_{BG}(i-1,j) \\ &\quad + \tilde{\delta}^E_{BG}(i+1,j)], \end{aligned} \quad (13)$$

for the case shown in Fig. 1(a);

$$\begin{aligned}\Delta^E(i,j) &= \bar{\delta}^E_R(i,j) + \alpha[2\tilde{\delta}^E_{RG}(i,j) + \tilde{\delta}^E_{GB}(i-1,j) \\ &\quad + \tilde{\delta}^E_{GB}(i+1,j)], \end{aligned} \quad (14)$$

for the case shown in Fig. 1(c); and

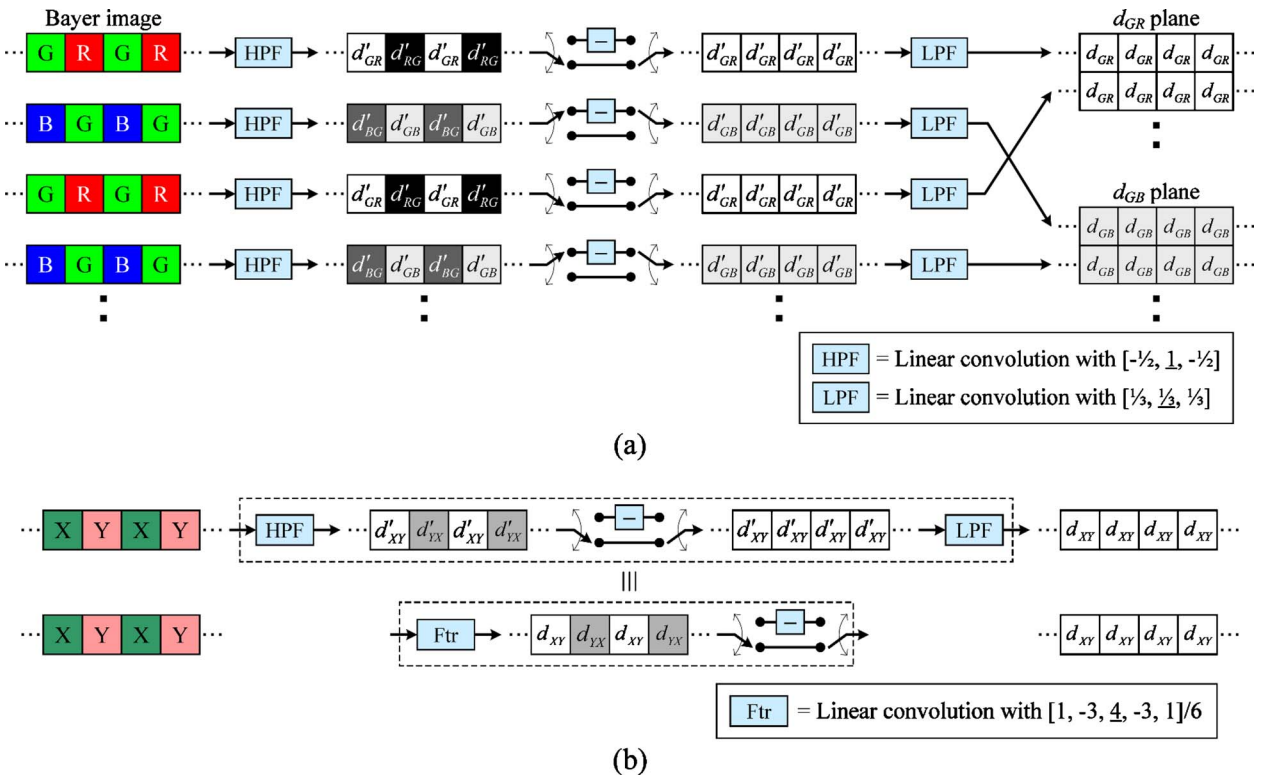


**Fig. 4** Estimation of color difference values for computing eastbound or westbound IGs: (a) realization procedure and (b) alternative implementations of the core module.

$$\Delta^E(i,j) = \tilde{\delta}^E_B(i,j) + \alpha[2\tilde{\delta}^E_{BG}(i,j) + \tilde{\delta}^E_{GR}(i-1,j) + \tilde{\delta}^E_{GR}(i+1,j)], \quad (15)$$

for the case shown in Fig. 1(d). The northbound, westbound, and southbound IGs of a pixel can be defined similarly as before. As a matter of fact, they can be determined by rotating the Bayer image clockwise by 90, 180, and 270 deg, respectively, and then computing the eastbound IGs of the rotated versions.

As a final remark, we note that, due to the absolute value nature of each component in the definition of IG, we have $\Delta^E(i,j)=\Delta^W(i,j+2)$ and $\Delta^S(i,j)=\Delta^N(i+2,j)$. By making use of this property, one can save an amount of computation effort.

### 3.1 *Color Difference Estimation for Computing Integrated Gradients*

When computing the proposed IGs for pixel $(i,j)$, $d_{GB}(i,j)$, $d_{BG}(i,j)$, $d_{GR}(i,j)$, and $d_{RG}(i,j)$ are required, and their estimation depends on which IG of the pixel is evaluated.

Figure 4(a) shows how to generate the required G-R and G-B planes for computing eastbound/westbound IGs. In a Bayer image, the sensor pattern repeats every other row and every other column. For each row, the Bayer sample sequence is convolved with $[-0.5,1,-0.5]$ to provide $C^H_d(i,j)$. The output can be considered as a sequence of alternate preliminary estimates $d'_{GR}$ and $d'_{RG}$ ($d'_{BG}$ and $d'_{GB}$). Since we have $d_{XY}(i,j)=-d_{YX}(i,j)$ in theory, by negating every second estimate in a row, we have a row of $d'_{GR}$ ($d'_{GB}$). A three-point averaging filter is then applied to remove the potential high frequency noise in each row, which produces the final CD estimates $d_{GR}$ ($d_{GB}$). The G-R (G-B) plane is constructed with all odd (even) rows.

To get the required G-R and G-B planes for computing the northbound or southbound IG, one can follow the same procedures as presented in Fig. 4(a) after rotating the Bayer image by 90 deg.

Figure 5 shows the CD planes of a testing image for computing IGs of different directions. The vertical (horizontal) resolution supported by the CD planes shown in

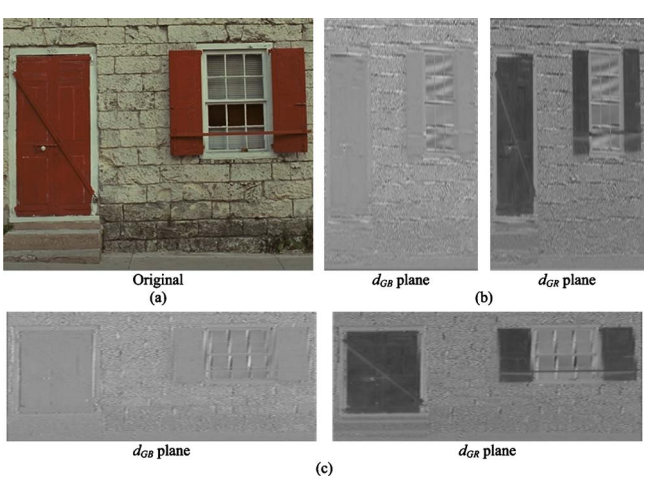


**Fig. 5** (a) Part of a testing image. (b) Color difference planes for computing northbound or southbound IGs. (c) Color difference planes for computing eastbound or westbound IGs.

**Fig. 6** Set of full-color training images used for parameter study. (Color online only.)

Fig. 5(b) [Fig. 5(c)] is identical to that of the original image. One can see that some edges can only be detected in a particular CD plane but not the other one. That explains why IG instead of WIG is used in our detection process.

### 3.2 *Realization in Practice*

Figure 4(a) presents the idea how CDs are estimated step by step. These steps can be combined to make their realization much easier by making use of the following facts. First, the functions of the two approaches of implementation shown in Fig. 4(b) are identical, so one can replace the upper approach with the lower one in the realization. Second, the negation process in the lower approach can be skipped because $|d_{XY}(i,j)+d_{YX}(i,j+1)|$, the absolute value of the sum of two successive output values of the filter, equals $|d_{XY}(i,j)-d_{XY}(i,j+1)|$. It is already the information required for computing $\tilde{\delta}^E_{XY}(i,j)$ and hence the corresponding IG.

Based on these facts, one can first convolve each row of the Bayer image with filter kernel $F=[1,-3,\underline{4},-3,1]*[1,\underline{1}]=[1,-2,1,\underline{1},-2,1]$, where the $*$ sign denotes convolution and the underscore marks the position of the pixel of interest, and combine the filter outputs of all rows to form a plane of $s(i,j)$, which equals $6[d_{XY}(i,j)+d_{YX}(i,j+1)]$ if $(i,j)$ carries an $X$ Bayer sample, or $6[d_{YX}(i,j)+d_{XY}(i,j+1)]$ if $(i,j)$ carries a $Y$ Bayer sample. $\tilde{\delta}^E_{XY}(i,j)$ can then be obtained with

$$\tilde{\delta}^E_{XY}(i,j)=\frac{1}{12}[|s(i,j)|+|s(i,j+1)|]. \tag{16}$$

### 3.3 *Determination of $\alpha$*

Parameter $\alpha$ is a weighting factor used to control the contribution of the CI and CD gradients to IG. An empirical study was carried out to investigate its impact to the demosaicing performance. The performance was measured by the average color peak signal-to-noise ratio (CPSNR) over a set of testing images shown in Fig. 6, and the CPSNR of a testing image is given by

$$\text{CPSNR}=10\log_{10}\frac{255^2}{\frac{1}{3HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\|I_o(i,j)-I_r(i,j)\|^2}, \tag{17}$$

where $I_o(i,j)$ and $I_r(i,j)$ respectively denote the three-element color vectors of pixel $(i,j)$ in the original and reconstructed images of size $H\times W$ each.

It was found that the optimum in terms of CPSNR happened at around $\alpha=1$ and, within the range of $1<\alpha<5$, the CPSNR variation of the results achieved by the proposed demosaicing method was less than 0.1 dB. By considering this, the value of $\alpha$ is selected to be $3/2$, such that the computation of $\alpha\tilde{\delta}^E_{XY}(i,j)$ using the approach presented in Sec. 3.2 only involves shift and add operations. This saves realization effort.

## 4 Proposed Demosaicing Method

Figure 2 briefly illustrates the work flow of the proposed demosaicing method. After IG extraction, the proposed method interpolates the green plane first. The resultant green plane is then enhanced to provide a reference for the subsequent red and blue plane interpolations. The extracted IGs are used in various stages of the proposed method to improve the interpolation efficiency. The details of these stages are described in this section.

For the sake of reference, hereafter a pixel at location $(i,j)$ in the Bayer image is represented by either $[R(i,j),g(i,j),b(i,j)]$, $[r(i,j),G(i,j),b(i,j)]$, or $[r(i,j),g(i,j),B(i,j)]$, where $R(i,j)$, $G(i,j)$, and $B(i,j)$ denote the known red, green, and blue Bayer samples, and $r(i,j)$, $g(i,j)$, and $b(i,j)$ denote the unknown samples of corresponding color channels in the image. The final estimates of $r(i,j)$, $g(i,j)$, and $b(i,j)$ are denoted as $\hat{r}(i,j)$, $\hat{g}(i,j)$, and $\hat{b}(i,j)$, respectively, for clear presentation.

### 4.1 *Green Plane Interpolation*

As far as a pixel that does not have a green Bayer sample is concerned, the pattern of its local region must be in the form shown in either Fig. 1(c) or 1(d). Without losing generality, the former pattern is discussed in this work. For the pattern shown in Fig. 1(d), one can exchange the red samples with the corresponding blue samples and then perform the interpolation in the same way.

For the case shown in Fig. 1(c), the missing green sample of pixel $(i,j)$ can be interpolated with one of the following Laplacian interpolation filters as proposed in Ref. 7.

$$g^H_{i,j}=\frac{G(i,j-1)+G(i,j+1)}{2}+\frac{2R(i,j)-R(i,j-2)-R(i,j+2)}{4}, \tag{18}$$

$$g^V_{i,j}=\frac{G(i-1,j)+G(i+1,j)}{2}+\frac{2R(i,j)-R(i-2,j)-R(i+2,j)}{4}, \tag{19}$$

$$g^D_{i,j}=\frac{g^H_{i,j}+g^V_{i,j}}{2}, \tag{20}$$

where $g^H_{i,j}$, $g^V_{i,j}$, and $g^D_{i,j}$ are, respectively, the estimates obtained with the corresponding horizontal, vertical, and nondirectional interpolators. The interpolator defined in Eq. (20) is nondirectional in a way that $g^D_{i,j}$ is actually the av-

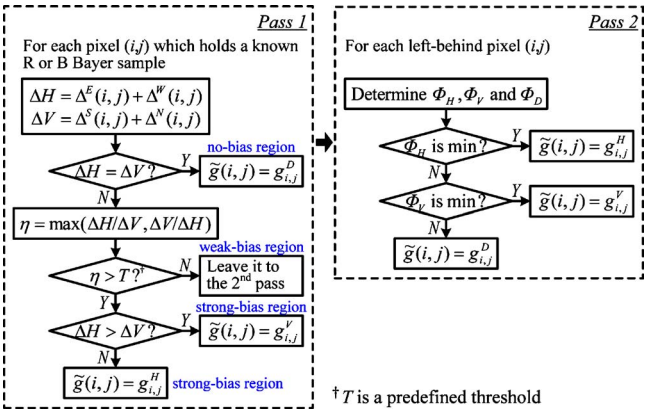


**Fig. 7** Procedures for interpolating a missing green sample in the proposed demosaicing method.

erage of $g^H_{i,j}$ and $g^V_{i,j}$. It favors neither horizontal nor vertical interpolation in such a case.

The selection of the interpolators is critical to the demosaicing performance. In the proposed demosaicing method, the high performance two-step estimation scheme proposed in Ref. 26 is modified in three aspects to estimate the missing green samples at a reduced complexity. First, instead of the parameters $L^H$ and $L^V$ used in Ref. 26, IGs are utilized to determine the interpolation direction of the missing green sample to improve the performance. Second, pixels that obviously should not be interpolated horizontally or vertically are processed in the first pass rather than in the second pass to save computation effort. Third, in the second pass, the analysis on the variance of local CD values is simplified without sacrificing its reliability. Figure 7 summarizes the procedures of the proposed two-step estimation scheme. This scheme produces a preliminary green estimate denoted as $\tilde{g}(i,j)$ for each pixel $(i,j)$ without a G Bayer sample.

The first pass of the estimation scheme raster scans the Bayer image to classify the local region of each pixel that does not carry a G Bayer sample. The classification is based on $\eta = \max(\Delta H/\Delta V, \Delta V/\Delta H)$, where $\Delta H = \Delta^E(i,j) + \Delta^W(i,j)$ and $\Delta V = \Delta^N(i,j) + \Delta^S(i,j)$. When $\eta$ equals 1 (i.e., $\Delta H = \Delta V$), obviously the interpolation should not be done along the horizontal or vertical direction. When $\eta$ is larger than a predefined threshold $T$, the difference between $\Delta H$ and $\Delta V$ is significant enough to provide a clear message that a directional interpolation should be done. Otherwise, there is no clear suggestion, and additional information is required to make the selection. Accordingly, based on the extent of preference for using a horizontal or a vertical interpolator, one can classify the region in which the pixel of interest is to be a no-bias, strong-bias, or a weak-bias region. For any pixel $(i,j)$ that is not in a weak-bias region, $\tilde{g}(i,j)$ is determined in pass 1 as specified in Fig. 7. As for those pixels in weak-bias regions, more information is required to make a decision and hence they are handled in pass 2.

Extra CD information is exploited in pass 2 to decide the interpolation direction. Though one may use the CD estimates already obtained in IG extraction to save the effort, a re-estimation using the green estimates obtained in pass 1 is performed in our realization, as these green estimates are generally more accurate and hence can guide one to a better decision.

Assume that the green sample of pixel $(i,j)$ shown in Fig. 1(c) cannot be determined in pass 1. In pass 2, the CD values of pixel $(m,n) \in \{(i, j \pm 2t), (i \pm 2t, j) | t = 0, 1, 2, \ldots L\}$ are re-estimated to estimate: 1. the CD variation in the region, and 2. the CD variation along the horizontal and vertical axes passing pixel $(m,n)$, where $L$ determines the number of pixels involved in the estimation. Let $\rho^H_{GR}(m,n)$, $\rho^V_{GR}(m,n)$, and $\rho^D_{GR}(m,n)$ be three green-to-red CD estimates of pixel $(m,n)$ obtained under three different conditions as

$$\rho^k_{GR}(m,n) = \begin{cases} \tilde{g}(m,n) - R(m,n) & \text{if } \tilde{g}(m,n) \text{ was obtained in pass 1} \\ g^k_{m,n} - R(m,n) & \text{otherwise} \end{cases} \quad \text{for } k \in \{H,V,D\}. \tag{21}$$

Three parameters are then computed with these estimates as follows.

$$\Phi_H = \sum_{t=-L}^{L} |\rho^H_{GR}(i,j) - \rho^H_{GR}(i,j+2t)|,$$

$$\Phi_V = \sum_{t=-L}^{L} |\rho^V_{GR}(i,j) - \rho^V_{GR}(i+2t,j)|,$$

and

$$\Phi_D = \frac{1}{2} \sum_{t=-L}^{L} [|\rho^D_{GR}(i,j) - \rho^D_{GR}(i,j+2t)| + |\rho^D_{GR}(i,j) - \rho^D_{GR}(i+2t,j)|]. \tag{22}$$

In particular, parameters $\Phi_H$ and $\Phi_V$ are respectively used to reflect the extent of CD variation along the horizontal and vertical axes. Parameter $\Phi_D$ is an estimate of the extent of CD variation in a local region under the assumption that the nondirectional interpolator, defined as in Eq. (20), is used. It serves as a reference to determine whether a nondirectional interpolator should be used.

CDs are generally locally constant along an edge. If a

directional interpolation is used, the right directional interpolator should provide an interpolation result, the CD variation of which is minimum along the corresponding direction. Accordingly, the missing green sample of pixel $(i,j)$ can be estimated as

$$\tilde{g}(i,j) = g_{i,j}^{Z}, \quad \text{where } z = \arg\min_{k\in\{H,V,D\}}(\Phi_k). \tag{23}$$

For the case that pixel $(i,j)$ contains a blue Bayer sample as shown in Fig. 1(d), one can interchange the roles of red and blue samples and then follow the same procedures as mentioned before to determine its green estimate $\tilde{g}(i,j)$.

At the end of pass 2, a complete demosaiced green plane is obtained. Figure 2(c) shows the spatial arrangement of the available color samples in the processing image after green plane interpolation.

An empirical study was carried out to investigate the impact of the threshold $T$ and the parameter $L$ to the demosaicing performance of the proposed method. The demosaicing performance under various combinations of $L \in \{1,2,3,4\}$ and $T \in \{1.0, 1.1, \ldots, 5.0\}$ were evaluated, and the set of testing images shown in Fig. 6 were used in the evaluation. Based on this study result, the settings of $T=1.7$ and $L=3$ are selected. All the simulation results reported in this work are obtained with these settings unless other arrangements are specified.

### 4.2 *Green Plane Enhancement*

With the fully populated green plane, the proposed demosaicing method then enhances the demosaiced green samples prior to the interpolation of the red and blue planes. Since the high frequency proportion of a CD signal is generally weak,[6] the proposed enhancement scheme is carried out in the CD domain to produce more accurate green estimates.

Without losing generality, the case of enhancing the demosaiced green sample at a red Bayer sample position, as depicted in Fig. 1(c), is described in this work. As for the case shown in Fig. 1(d), one can exchange the role of the red and blue samples and perform the same treatment used in this case to achieve the goal.

After green plane interpolation, $\tilde{g}(i,j)$ is available for all pixels without green Bayer samples, and it is the "best" green estimate so far. Based on the idea that a better temporary green estimate can be used to derive a better CD estimate and hence a better interpolation result, in green plane enhancement the CD estimate of a pixel is re-evaluated with $\tilde{g}(i,j)$. For example, for the pixel $(i,j)$ shown in Fig. 1(c), its G-R CD is re-evaluated as

$$\bar{d}_{GR}(i,j) = \tilde{g}(i,j) - R(i,j), \tag{24}$$

for $(i,j)$ carrying R Bayer sample. After re-evaluating the G-R CDs of all pixels carrying R Bayer samples, the CD estimate of pixel $(i,j)$ is further adjusted to be $\hat{d}_{GR}(i,j)$ by fusing $\bar{d}_{GR}(i,j)$ with $\tilde{d}_{GR}(i,j)$, an interpolation result based on $\bar{d}_{GR}(i,j\pm 2)$ and $\bar{d}_{GR}(i\pm 2,j)$, as follows.

$$\hat{d}_{GR}(i,j) = \beta\bar{d}_{GR}(i,j) + (1-\beta)\tilde{d}_{GR}(i,j) \tag{25}$$

for $(i,j)$ carrying R Bayer sample, where $0 \leqslant \beta \leqslant 0.5$ is a weighting factor controlling the fusion. In formulation, $\tilde{d}_{GR}(i,j)$ is defined as

$$\tilde{d}_{GR}(i,j) = \frac{\begin{pmatrix} w^E\bar{d}_{GR}(i,j+2) + w^W\bar{d}_{GR}(i,j-2) \\ + w^S\bar{d}_{GR}(i+2,j) + w^N\bar{d}_{GR}(i-2,j) \end{pmatrix}}{w^E + w^W + w^S + w^N} \tag{26}$$

for $(i,j)$ carrying R Bayer sample, where $w^k = 1/\Delta^k(i,j)$ for $k \in \{E,W,S,N\}$. Since a large value of $\Delta^k(i,j)$ implies that there is a great change of either CD or CI in the corresponding direction, the weighting mechanism in Eq. (26) automatically directs the interpolation of $\tilde{d}_{GR}(i,j)$ along an edge when there is one.

Parameter $\beta$ can be determined off-line by linear regression. In our study, a set of training images covering the testing images shown in Fig. 6 were used in the training process. For the simulation results reported in this work, $\beta$ is selected to be 0.33. As a matter of fact, the demosaicing performance of the proposed method is not sensitive to $\beta$. In our simulation study, the variation of the CPSNR performance is less than 0.055 dB on average when $\beta$ varies from 0.2 to 0.45.

With the adjusted $\hat{d}_{GR}(i,j)$, the demosaiced green sample is updated to be

$$\hat{g}(i,j) = \hat{d}_{GR}(i,j) + R(i,j) \tag{27}$$

for $(i,j)$ carrying R Bayer sample. Similar procedures can be performed with the same parameter $\beta$ to enhance the demosaiced green samples at pixels carrying blue Bayer samples. At the end of the enhancement, all $\tilde{g}(i,j)$ are updated and finalized to be $\hat{g}(i,j)$, as shown in Fig. 2(d).

### 4.3 *Red Plane and Blue Plane Interpolations*

The red plane interpolation is presented first here. During green plane enhancement, the $\hat{d}_{GR}(i,j)$ for pixels carrying R Bayer samples are evaluated with Eq. (25) and they are ready to form a partial G-R plane. The complete G-R plane is interpolated with this partial plane through two steps.

In step 1, the $\hat{d}_{GR}(i,j)$ for pixels carrying B Bayer samples are interpolated. Consider the pixel $(i,j)$ shown in Fig. 1(d). Its four diagonal neighbors carry R Bayer samples, and hence $\hat{d}_{GR}(i\pm 1,j\pm 1)$ are known. Accordingly, $\hat{d}_{GR}(i,j)$ can be interpolated with

$$\hat{d}_{GR}(i,j) = \frac{\begin{pmatrix} w^{NW}\hat{d}_{GR}(i-1,j-1) + w^{NE}\hat{d}_{GR}(i-1,j+1) \\ + w^{SE}\hat{d}_{GR}(i+1,j+1) + w^{SW}\hat{d}_{GR}(i+1,j-1) \end{pmatrix}}{w^{NW} + w^{NE} + w^{SE} + w^{SW}} \tag{28}$$

for $(i,j)$ carrying B Bayer sample, where

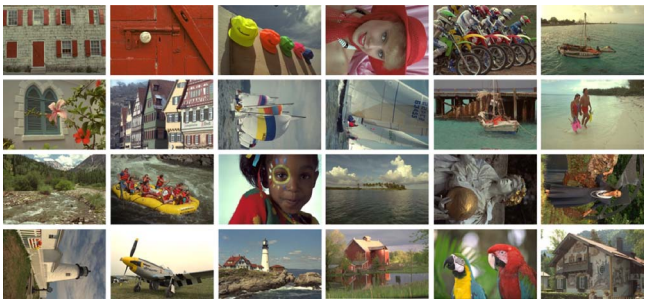

**Fig. 8** 24 full color testing images, referring to images 1 to 24 in a raster scanning sequence. (Color online only.)

$$w^{k_1k_2} = \frac{1}{\Delta^{k_1}(i,j) + \Delta^{k_2}(i,j)} \quad \text{for } k_l \in \{S,N\} \text{ and } k_2 \in \{E,W\}. \tag{29}$$

Theoretically, the weight $w^{k_1k_2}$ should be derived from the two-norm of two perpendicular IGs [$\Delta^{k_1}(i,j)$ and $\Delta^{k_2}(i,j)$]. However, our empirical study shows that the improvement over one-norm is insignificant for almost all the testing images.

To complete the G-R plane, the G-R CDs for pixels carrying G Bayer samples are interpolated in step 2. For the pixel $(i,j)$ shown in Fig. 1(a) or 1(b), $\hat{d}_{GR}(i,j)$ is interpolated with

$$\hat{d}_{GR}(i,j) = \frac{\begin{pmatrix} w^E \hat{d}_{GR}(i,j+1) + w^W \hat{d}_{GR}(i,j-1) \\ + w^S \hat{d}_{GR}(i+1,j) + w^N \hat{d}_{GR}(i-1,j) \end{pmatrix}}{w^E + w^W + w^S + w^N} \tag{30}$$

for $(i,j)$ carrying G Bayer sample, where $w^k = 1/\Delta^k(i,j)$ for $k \in \{E,W,S,N\}$.

With the complete G-R plane and the enhanced green plane, all missing red samples can be estimated by

$$\hat{r}(i,j) = \begin{cases} G(i,j) - \hat{d}_{GR}(i,j) & \text{if } G(i,j) \text{ exists} \\ \hat{g}(i,j) - \hat{d}_{GR}(i,j) & \text{if } B(i,j) \text{ exists} \end{cases}. \tag{31}$$

Blue plane interpolation can be achieved in the same way by exchanging the role of red and blue samples. The interpolations of the two color planes are independent and can be carried out in parallel. Figure 2(e) shows the spatial arrangement of the intermediate red and blue planes obtained after step 1, and Fig. 2(f) shows the final reconstructed full color image.

## 5 Simulation Results

Simulations were carried out to evaluate the performance of the proposed demosaicing method. Some other state of the art demosaicing methods such as enhanced effective color interpolation (EECI),[11] alternating projections (AP),[12] adaptive filtering (AF),[15] directional linear mean square error (DLMSE),[16] adaptive homogeneity-directed demosaicing algorithm (AHDA),[18] primary-consistent soft decision (PCSD),[19] variance of color differences (VCD),[20] directional filtering and a-posteriori decision (DFPD),[21] and heterogeneity-projection hard decision (HPHD) (with adaptive interpolation)[22] were also evaluated for comparison. 24 digital full color images from the Kodak database, as shown in Fig. 8, were used in the simulations. These full color images were first subsampled according to the Bayer pattern to form a set of testing Bayer images. They were then reconstructed to full color images by the evaluated demosaicing methods. Whenever there is a postprocessing scheme recommended by the authors of a particular demosaicing method to enhance its demosaicing results, the scheme was performed in the simulation as suggested.

Two performance measures, CPSNR and S-CIELAB,[27] were exploited to measure the quality of the demosaiced images. Tables 1 and 2 list the CPSNR and S-CIELAB measures of various demosaicing methods. As mentioned, recommended postprocessing was performed to improve the demosaicing performance of HPHD, AHDA, DFPD, and VCD. One can see from the table that the proposed method outperforms the other demosaicing methods for the majority of the testing images. For images that contain many fine structures, such as images 1, 6, 9, and 13, the proposed method provides significant performance improvement over the others. As an example, for image 1, in which the fine stone structure is almost everywhere, the proposed method provides a CPSNR of 39.96 dB, which is around 1 dB higher than the second best CPSNR performance provided by other evaluated methods in the simulation. Consistent performance improvement can also be found when the S-CIELAB measure is used. These results demonstrate that the proposed demosaicing method is robust to the input when recovering a full color image from a Bayer image.

Figure 9 shows part of the demosaicing results of image 19 for visual comparison. It can be observed that the proposed demosaicing method produces an outstanding perceptual result compared with the other evaluated methods. Especially in the area near the boundary of the fence and the grass, the proposed method preserves the fine fence details and produces a result with almost invisible color artifacts. Some other sophisticated methods such as HPHD and VCD can also produce comparable results.

Figure 10 shows part of the demosaicing results of image 22 from which one can see the superiority of the proposed method to HPHD and VCD. In Figs. 10(h) and 10(i), severe zipper artifacts can be observed at the upper boundary of the wall and in the middle area of the wall. In contrast to this, they are significantly reduced in Fig. 10(j).

It is comparatively easy for a demosaicing method to produce a good demosaicing performance in smooth regions. The strength of a method is usually reflected by its performance in handling edge regions. In Table 3, the performance of HPHD, DLMSE, VCD, and the proposed method in regions of different nature is separately measured. The region classification is based on the criterion suggested in Ref. 10. The superiority of the proposed method is more visible in edge regions.

The IG proposed in this work plays a significant role in the proposed method. First, it provides information for one to select appropriate interpolators in green plane interpolation. Second, it guides the interpolation in the G-B and G-R planes such that final estimates of the missing color components can be derived indirectly in green plane enhancement and red (blue) plane interpolation. In the following

**Table 1** Performance of different demosaicing methods for CPSNR (in decibels).

| Image | AP Ref. 12 | AF Ref. 15 | EECI Ref. 11 | DLMSE Ref. 16 | AHDA Ref. 18 | PCSD Ref. 19 | VCD Ref. 20 | DEPD Ref. 21 | HPHD Ref. 22 | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 37.70 | 37.44 | 37.99 | 38.41 | 35.17 | 36.32 | 38.53 | 36.56 | 39.00 | **39.96** |
| 2 | 39.57 | 40.63 | 40.49 | 48.85 | 39.34 | 39.98 | 40.43 | 40.40 | 40.96 | **40.99** |
| 3 | 41.45 | 42.52 | 42.64 | 42.56 | 41.52 | 41.82 | 42.54 | 41.95 | 43.01 | **43.26** |
| 4 | 40.03 | 40.42 | 40.51 | 40.44 | 38.87 | 39.52 | 40.50 | 39.79 | **40.89** | 40.56 |
| 5 | 37.46 | 37.91 | 38.04 | 37.98 | 35.70 | 37.15 | 37.89 | 37.18 | **38.61** | 38.31 |
| 6 | 38.50 | 37.87 | 38.10 | 40.11 | 37.55 | 38.72 | 40.03 | 38.93 | 40.53 | **41.00** |
| 7 | 41.77 | 42.83 | 42.73 | 42.32 | 40.87 | 41.51 | 42.15 | 41.80 | **43.01** | 42.64 |
| 8 | 35.08 | 35.10 | 35.20 | 35.97 | 33.80 | 34.39 | 36.41 | 34.94 | 36.94 | **37.35** |
| 9 | 41.72 | 42.62 | 42.58 | 42.98 | 41.10 | 42.03 | 43.04 | 42.27 | 42.90 | **43.42** |
| 10 | 42.02 | 42.61 | 42.52 | 42.56 | 40.77 | 41.74 | 42.51 | 42.02 | 42.56 | **42.83** |
| 11 | 39.14 | 39.17 | 39.46 | 39.94 | 37.48 | 38.52 | 39.86 | 38.83 | 40.51 | **40.66** |
| 12 | 42.51 | 42.60 | 42.63 | 43.38 | 41.81 | 42.63 | 43.45 | 42.84 | 43.88 | **44.13** |
| 13 | 34.30 | 33.66 | 34.38 | 34.71 | 31.41 | 32.64 | 34.90 | 32.81 | 35.32 | **36.03** |
| 14 | 35.60 | 36.93 | 37.13 | 36.79 | 35.50 | 35.69 | 36.88 | 36.36 | **37.48** | 37.10 |
| 15 | 39.35 | 39.78 | 39.49 | 39.80 | 38.02 | 38.93 | 39.78 | 39.20 | 39.81 | **39.84** |
| 16 | 41.76 | 40.97 | 41.16 | 43.67 | 41.37 | 42.55 | 43.64 | 42.72 | 44.08 | **44.47** |
| 17 | 41.11 | 41.14 | 41.36 | 41.58 | 39.25 | 40.40 | 41.21 | 40.37 | 41.60 | **41.77** |
| 18 | 37.45 | 37.38 | 37.73 | 37.75 | 35.20 | 36.23 | 37.49 | 36.40 | **38.02** | 37.96 |
| 19 | 39.46 | 40.01 | 40.13 | 40.98 | 38.44 | 39.48 | 41.00 | 39.74 | 41.35 | **41.79** |
| 20 | 40.66 | 41.08 | 41.33 | 41.21 | 39.23 | 40.02 | 41.07 | 40.05 | 41.68 | **41.71** |
| 21 | 38.66 | 38.55 | 38.96 | 39.03 | 36.56 | 37.27 | 39.12 | 37.47 | 39.60 | **39.99** |
| 22 | 37.55 | 38.32 | 38.28 | 38.29 | 36.46 | 37.13 | 37.97 | 37.52 | 38.43 | **38.48** |
| 23 | 41.88 | 42.99 | 42.91 | 43.16 | 41.88 | 42.21 | 42.89 | 42.50 | 43.10 | **43.20** |
| 24 | 34.78 | 34.88 | 34.82 | **35.56** | 33.42 | 34.38 | 35.04 | 34.55 | 35.25 | 35.39 |
| Avg. | 39.15 | 39.48 | 39.61 | 40.00 | 37.95 | 38.80 | 39.93 | 39.05 | 40.36 | **40.54** |

part of this section, we show how IG contributes to the interpolation of the three color planes, and how the IG-based green plane enhancement contributes to the overall demosaicing performance of the proposed method.

Table 4 shows the performance of various decision-based demosaicing methods in the interpolation of the G plane. Specifically, the first row shows their performance in selecting the optimal interpolator among the three defined in Eqs. (18)–(20) to interpolate a missing green sample. Here, the optimal interpolator means the one that provides the best green estimate in terms of MSE. Based on the selected interpolators, missing G samples were interpolated directly to obtain the G plane. No enhancement scheme or adaptive scheme is used in any algorithms. As the MSE penalty for a miss varies from pixel to pixel and from interpolator to interpolator, a method that can achieve a higher hit rate may not be able to provide a G plane of lower PSNR. However, one can see that the hit rate of our proposed IG-based selection scheme is higher than the others, and the PSNR of its resultant green plane is the highest. This reveals that IG can effectively detect the edge directions and contribute to the interpolation of the green plane.

To study the contribution of the IG-based G plane enhancement step, three different approaches were imple-

**Table 2** Performance of different demosaicing methods for S-CIELAB.

| Image | AP Ref. 12 | AF Ref. 15 | EECI Ref. 11 | DLMSE Ref. 16 | AHDA Ref. 18 | PCSD Ref. 19 | VCD Ref. 20 | DEPD Ref. 21 | HPHD Ref. 22 | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.6460 | 1.6328 | 1.5060 | 1.4552 | 1.6140 | 1.7422 | 1.4630 | 1.8101 | 1.4301 | **1.2747** |
| 2 | 1.6639 | 1.5218 | 1.4880 | 1.4262 | 1.5726 | 1.5951 | 1.4985 | 1.5866 | 1.4445 | **1.4019** |
| 3 | 0.9509 | 0.8828 | 0.8608 | 0.8699 | 0.9296 | 0.9325 | 0.8802 | 0.9334 | 0.8444 | **0.8270** |
| 4 | 1.2863 | 1.2195 | 1.2099 | 1.1836 | 1.3196 | 1.3367 | 1.2156 | 1.3479 | 1.1990 | **1.1737** |
| 5 | 2.1668 | 1.9992 | 1.9431 | 1.9906 | 2.2155 | 2.1340 | 2.0313 | 2.2147 | **1.8585** | 1.8815 |
| 6 | 1.2447 | 1.2805 | 1.2137 | 1.0472 | 1.1348 | 1.1426 | 1.0627 | 1.1871 | 1.0164 | **0.9682** |
| 7 | 1.1062 | 0.9704 | 0.9742 | 1.0194 | 1.1285 | 1.0871 | 1.0329 | 1.0715 | **0.9643** | 0.9827 |
| 8 | 1.8668 | 1.7597 | 1.6898 | 1.6003 | 1.7256 | 1.8194 | 1.5865 | 1.8824 | 1.5243 | **1.4305** |
| 9 | 0.8411 | 0.7566 | 0.7734 | 0.7408 | 0.8179 | 0.7873 | 0.7418 | 0.7930 | 0.7743 | **0.7274** |
| 10 | 0.8263 | 0.7597 | 0.7661 | 0.7623 | 0.8407 | 0.8134 | 0.7727 | 0.8241 | 0.7853 | **0.7519** |
| 11 | 1.4742 | 1.4182 | 1.3413 | 1.3053 | 1.4240 | 1.4657 | 1.3302 | 1.4986 | 1.2759 | **1.2041** |
| 12 | 0.6765 | 0.6497 | 0.6404 | 0.6071 | 0.6554 | 0.6532 | 0.6143 | 0.6605 | 0.5951 | **0.5782** |
| 13 | 2.5824 | 2.7044 | 2.4381 | 2.4334 | 2.7473 | 2.9799 | 2.4453 | 3.0769 | 2.3441 | **2.1993** |
| 14 | 1.9443 | 1.7917 | 1.7172 | 1.7151 | 1.8681 | 1.9254 | 1.7484 | 1.9172 | **1.6292** | 1.6392 |
| 15 | 1.4286 | 1.3302 | 1.3230 | 1.3121 | 1.4597 | 1.4622 | 1.3372 | 1.4475 | 1.3287 | **1.2963** |
| 16 | 1.0303 | 1.0776 | 1.0347 | 0.8515 | 0.9192 | 0.9249 | 0.8710 | 0.9567 | 0.8411 | **0.8107** |
| 17 | 1.3292 | 1.2973 | 1.2791 | **1.2585** | 1.3983 | 1.3991 | 1.3076 | 1.4382 | 1.3023 | 1.2590 |
| 18 | 2.1841 | 2.0626 | 2.0695 | 2.0821 | 2.3892 | 2.3639 | 2.1318 | 2.3462 | **2.0613** | 2.0837 |
| 19 | 1.2877 | 1.1971 | 1.1858 | 1.1142 | 1.2715 | 1.2928 | 1.1439 | 1.3143 | 1.1420 | **1.0715** |
| 20 | 1.0030 | 0.9563 | 0.9277 | 0.9451 | 1.0326 | 1.0506 | 0.9633 | 1.0796 | 0.9158 | **0.8974** |
| 21 | 1.3281 | 1.3302 | 1.2413 | 1.2510 | 1.3612 | 1.4471 | 1.2616 | 1.4995 | 1.2028 | **1.1567** |
| 22 | 1.4973 | **1.3690** | 1.3949 | 1.3974 | 1.5942 | 1.5594 | 1.4495 | 1.5298 | 1.3698 | 1.3977 |
| 23 | 0.9500 | 0.8773 | 0.8931 | **0.8737** | 0.9630 | 0.9647 | 0.9183 | 0.9438 | 0.8911 | 0.8869 |
| 24 | 1.4378 | 1.3669 | 1.3350 | 1.3300 | 1.4876 | 1.4697 | 1.3628 | 1.5032 | 1.3165 | **1.3074** |
| Avg. | 1.4064 | 1.3421 | 1.3019 | 1.2738 | 1.4113 | 1.4312 | 1.2988 | 1.4526 | 1.2524 | **1.2170** |

mented to complete the demosaicing process after the IG-based G plane interpolation. In approach 1 ($I_G$-$I_{RB}$), we move on to do the proposed IG-based R and B plane interpolations without implementing the proposed IG-based G plane enhancement step. In approach 2 ($I_G$-$I_{RB}$-$E_{RGB}$), various conventional postenhancement schemes were applied to the output of $I_G$-$I_{RB}$ to enhance all color planes. Approach 3 ($I_G$-$E_G$-$I_{RB}$) corresponds to the full version of the proposed demosaicing algorithm, in which the IG-based G plane enhancement is done before the R and B plane interpolations.

Table 5 contrasts the average performance in different scenarios. By comparing the PSNRs of different planes obtained with $I_G$-$I_{RB}$ and $I_G$-$E_G$-$I_{RB}$, one can see that the IG-based enhancement step boosts up the quality of the G plane by 2.62 dB. Subsequently, this better G plane helps to improve the quality of the interpolated R and B planes, as their interpolation relies on the enhanced G plane.

Another observation is that, from an efficiency point of view, enhancing a single G plane before R and B plane interpolations with the proposed IG-based scheme is actually better than the conventional approach that enhances all

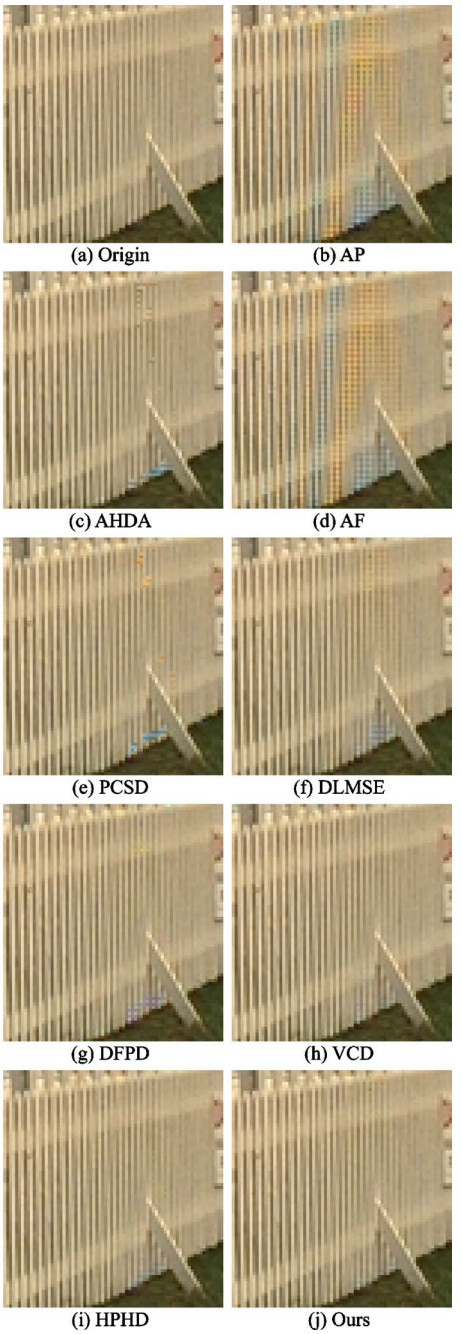


**Fig. 9** Demosaicing results of image 19.

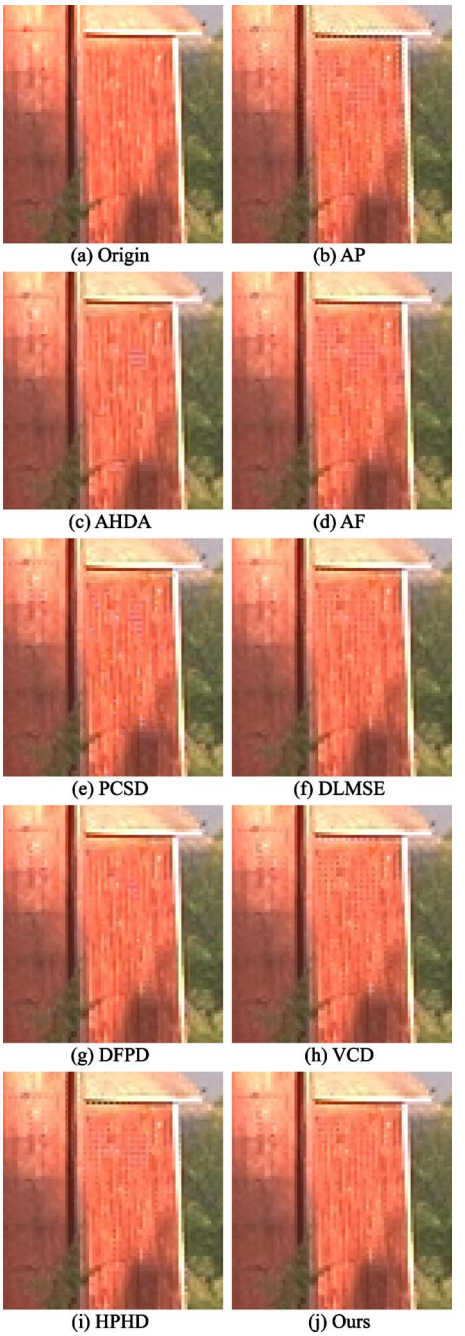


**Fig. 10** Demosaicing results of image 22.

color planes at the end. Even though the CPSNR difference between $I_G$-$I_{RB}$-$E_{RGB}$ and $I_G$-$E_G$-$I_{RB}$ is not significant apparently, one can still see that carrying out the IG-based enhancement step at an early stage can provide a better overall demosaicing performance at a much lower complexity cost.

As a final remark, we note that, even without the help of the IG-based enhancement step, the output of the proposed IG-based demosaicing algorithm (i.e., $I_G$-$I_{RB}$) can still work with various conventional postprocessing enhancement schemes such as Ref. 10, 11, and 25 to provide a demosaicing performance better than some conventional demosaicing algorithms.[12,15,18,19,21] This can be verified by comparing the CPSNRs achieved by $I_G$-$I_{RB}$-$E_{RGB}$ and the CPSNRs of the corresponding algorithms in Table 1.

## 6 Computational Complexity

This section reports the computational complexity of the proposed demosaicing method in terms of number of additions (ADD), multiplications (MUL), bit-shifts (SHT), and absolute-value-taking operations (ABS). A comparison operation is considered as an addition in our report.

Table 6 shows the number of arithmetic operations required per involved pixel in different stages of the proposed

**Table 3** Performance of various demosaicing methods in edge and smooth regions.

| | CPSNR (in dB) | | S-CIELAB | |
|---|---|---|---|---|
| | Edge | Smooth | Edge | Smooth |
| VCD[20] | 34.74 | 41.94 | 2.3671 | 1.1313 |
| DLMSE[16] | 34.78 | 42.06 | 2.3392 | 1.1042 |
| HPHD[22] | 35.05 | 42.49 | 2.3188 | 1.0842 |
| Ours | 35.37 | 42.54 | 2.1885 | 1.0652 |

demosaicing method. In this table, $\Omega_R$, $\Omega_G$, and $\Omega_B$ denote the set of pixels carrying R, G, and B Bayer samples, respectively.

The computational complexity of the other evaluated demosaicing methods is tabulated in Table 7 for comparison. The complexity figures for AP, AF, AHDA, and PCSD are directly taken from Lian *et al.*[15] For VCD,[20] DFPD,[21] and HPHD,[22] the figures are, respectively, extracted from their corresponding papers. As for DLMSE,[16] its complexity is derived based on its realization presented in Ref. 16.

From Table 7, one can see that the proposed demosaicing method requires the least computation effort compared with HPHD, DLMSE, and VCD, the demosaicing performance of which is comparable to ours. In the worst case where all red and blue Bayer samples are in weak-bias regions, the proposed method requires totally at most 93.5 operations to reconstruct a color pixel on average, which is at least 67% lower than that required by HPHD, the second best in terms of CPSNR and S-CIELAB. Comparatively speaking, the complexity of DLMSE and VCD is closer to ours, but it is still around 20 to 40% higher. On top of that, from the simulation results shown in Figs. 10(f) and 10(h), one can see some checker patterns and zipper artifacts, while the situation is better in Fig. 10(j). Although the com-

**Table 6** Number of arithmetic operations required per involved pixel in the proposed demosaicing method.

| Operational step | ADD | MUL | SHT | ABS |
|---|---|---|---|---|
| Extraction of IGs for $(i,j) \in \Omega_R \cup \Omega_G \cup \Omega_B$ | 16 | 0 | 6 | 4 |
| G plane interpolation for $(i,j) \in \Omega_R \cup \Omega_B$ | | | | |
| • If $(i,j)$ is in no-bias regions | 12 | 0 | 5 | 0 |
| • If $(i,j)$ is in strong-bias regions | 14 | 1 | 5 | 0 |
| • If $(i,j)$ is in weak-bias regions | 52 | 1 | 6 | 12 |
| G plane enhancement for $(i,j) \in \Omega_R \cup \Omega_B$ | 9 | 9 | 0 | 0 |
| Interpolating B samples for $(i,j) \in \Omega_R$ | 11 | 9 | 0 | 0 |
| Interpolating R samples for $(i,j) \in \Omega_B$ | 11 | 9 | 0 | 0 |
| Interpolating R/B samples for $(i,j) \in \Omega_G$ | 14 | 12 | 0 | 0 |

**Table 4** Performance of various decision-based demosaicing methods in G plane interpolation.

| | AHDA Ref. 18 | PCSD Ref. 19 | VCD Ref. 20 | DFPD Ref. 21 | HPHD Ref. 22 | Ours |
|---|---|---|---|---|---|---|
| Hit rate (%) | 51.42 | 49.75 | 48.11 | 50.74 | 35.55 | 52.30 |
| G Plane PSNR (dB) | 39.65 | 39.25 | 40.10 | 39.54 | 40.16 | 40.66 |

**Table 5** Average performance of various enhancement schemes. $I_G$-$E_G$-$I_{RB}$ is the equivalent at the full version of the proposed demosaicing method.

| Approach | $I_G$-$I_{RB}$ | $I_G$-$I_{RB}$-$E_{RGB}$ | | | $I_G$-$E_G$-$I_{RB}$ |
|---|---|---|---|---|---|
| Enhancement scheme | — | Ref. 11 | Ref. 10 | Ref. 25 | Ours |
| G plane PSNR (dB) | 40.68 | **43.37** | 43.02 | 42.98 | 43.30 |
| R plane PSNR (dB) | 38.67 | 39.44 | 39.32 | 38.68 | **39.55** |
| B plane PSNR (dB) | 38.86 | 39.87 | 39.62 | 39.53 | **39.94** |
| CPSNR (dB) | 39.26 | 40.49 | 40.29 | 39.93 | **40.54** |
| S-CIELAB | 1.2839 | 1.2228 | 1.2600 | 1.2839 | **1.2170** |

**Table 7** Complexity of various demosaicing methods in terms of number of arithmetic operations per pixel.

| | ADD | MUL | SHT | ABS | Total |
|---|---|---|---|---|---|
| AP | 384 | 384 | — | — | **768** |
| AF | 40.5 | 10.5 | 11 | 3 | **54** |
| DLMSE | 72.5 | 36 | 5.5 | — | **114** |
| AHDA | 184 | 12 | — | 49 | **245** |
| PCSD | 149 | 60 | 5 | 8 | **222** |
| VCD | 48 to 92 | 16.5 to 36 | 4 to 7 | 4 | **72.5** to **139** |
| DFPD | 28.5 | 0.5 | 6 | 1 | **36** |
| HPHD | 176.5 to 207 | 81 to 86 | 11 to 18 | 20.5 to 31 | **289** to **342** |
| Ours | 39 to 59 | 15 to 15.5 | 8.5 to 9 | 4 to 10 | **66.5** to **93.5** |

plexity of AF and DFPD is lower than that of the proposed method, their output quality is low, as shown in Fig. 9. All these facts reveal that the cost performance of the proposed method is high.

In our simulations, the average number of arithmetic operations is around 80 per pixel. The execution time for the proposed demosaicing method to process a Bayer image of size $768 \times 512$ on a 3.4-GHz Pentium 4 PC with 1024-MB RAM is 0.0948 s on average.

## 7 Conclusion

A new edge-sensing measure called integrated gradient is proposed. This measure effectively extracts gradient information from a Bayer image in both color intensity and color difference domains, and consequently provides reliable and complete information for one to interpolate missing samples in a Bayer image along an appropriate direction.

An efficient decision-based demosaicing method is then developed. Under the guidance of the same integrated gradients, the proposed demosaicing method interpolates different color planes in different stages. Though the method updates the green plane and the corresponding color difference planes in the course to provide better references for interpolation, computationally expensive re-estimation of local gradients based on intermediate interpolation results is avoided. It guarantees the consistency of the interpolation direction in different color channels, and saves the effort required to repeatedly extract gradient information from intermediate interpolation results at different stages.

Unlike some other demosaicing methods that carry out a postprocessing step to enhance all color planes at the end, the proposed demosaicing method enhances the green plane before the interpolation of the red and blue planes based on the integrated gradients. By so doing, it provides a better reference for one to interpolate the red and blue planes. This automatically improves the quality of the resultant red and blue planes, and hence eliminates the necessity of another enhancement step for these planes after their interpolation.

Simulation results confirm that the proposed demosaicing method outperforms up-to-date demosaicing methods in terms of output quality at a complexity of around 80 arithmetic operations per pixel.

### Acknowledgment

This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (PolyU 5130/09E), and a grant from the Center of Signal Processing (G-U413). The author would like to thank the authors of Refs. 12, 15, 16, 18, and 22 for providing programs of their demosaicing methods.

### References

1. B. E. Bayer, "Color imaging array," U.S. Patent No. 3 971 065, to Eastman Kodak Company, Rochester, NY (1976).
2. T. Sakamoto, C. Nakanishi, and T. Hase, "Software pixel interpolation for digital still cameras suitable for a 32-bit MCU," *IEEE Trans. Consum. Electron.* **44**(4), 1342–1352 (1998).
3. B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal Process. Mag.* **22**(1), 44–54 (2005).
4. D. R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," U.S. Patent No. 4 642 678, to Eastman Kodak Company, Rochester, NY (1987).
5. W. T. Freeman, "Median filter for reconstructing missing color samples," U.S. Patent No. 4 724 395, to Polaroid Corporation, Cambridge, MA (1988).
6. S. C. Pei and I. K. Tam, "Effective color interpolation in CCD color filter arrays using signal correlation," *IEEE Trans. Circuits Syst. Video Technol.* **13**(6), 503–513 (2003).
7. J. F. Hamilton and J. E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," U.S. Patent No. 5,629,734, to Eastman Kodak Company, Rochester, NY (1997).
8. R. Lukac, K. N. Plataniotis, D. Hatzinakos, and M. Aleksic, "A novel cost effective demosaicing approach," *IEEE Trans. Consum. Electron.* **50**(1), 256–261 (Feb. 2004).
9. R. Lukac and K. N. Plataniotis, "Data adaptive filters for demosaicking: a framework," *IEEE Trans. Consum. Electron.* **51**(2), 560–570 (2005).
10. W. M. Lu and Y. P. Tan, "Color filter array demosaicking: new

method and performance measures," *IEEE Trans. Image Process.* **12**(10), 1194–1210 (2003).
11. L. Chang and Y. P. Tan, "Effective use of spatial and spectral correlations for color filter array demosaicking," *IEEE Trans. Consum. Electron.* **50**(1), 355–365 (2004).
12. B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Process.* **11**(9), 997–1013 (2002).
13. X. Li, "Demosaicing by successive approximation," *IEEE Trans. Image Process.* **14**(3), 370–379 (2005).
14. D. Alleysson, S. Susstrunk, and J. Herault, "Linear demosaicing inspired by the human visual system," *IEEE Trans. Image Process.* **14**(4), 439–449 (2005).
15. N. X. Lian, L. Chang, Y. P. Tan, and V. Zagorodnov, "Adaptive filtering for color filter array demosaicking," *IEEE Trans. Image Process.* **16**(10), 2515–2525 (2007).
16. L. Zhang and X. Wu, "Color demosaicking via directional linear minimum mean square-error estimation," *IEEE Trans. Image Process.* **14**(12), 2167–2178 (2005).
17. D. D. Muresan and T. W. Parks, "Demosaicing using optimal recovery," *IEEE Trans. Image Process.* **14**(2), 267–278 (2005).
18. K. Hirakawa and T. W. Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE Trans. Image Process.* **14**(3), 360–369 (2005).
19. X. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicking for digital cameras (patent pending)," *IEEE Trans. Image Process.* **13**(9), 1263–1274 (2004).
20. K. H. Chung and Y. H. Chan, "Color demosaicing using variance of color differences," *IEEE Trans. Image Process.* **15**(10), 2944–2955 (2006).
21. D. Menon, S. Andriani, and G. Calvagno, "Demosaicing with directional filtering and a posteriori decision," *IEEE Trans. Image Process.* **16**(1), 132–141 (2007).
22. C. Y. Tsai and K. T. Song, "Heterogeneity-projection hard-decision color interpolation using spectral-spatial correlation," *IEEE Trans. Image Process.* **16**(1), 78–91 (2007).
23. C. A. Laroche and M. A. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent No. 5,373,322, to Eastman Kodak Company, Rochester, NY (1994).
24. R. H. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients," U.S. Patent No. 5,382,976, to Eastman Kodak Company, Rochester, NY (1995).
25. R. Lukac, K. Martin, and K. N. Plataniotis, "Demosaicked image postprocessing using local color ratios," *IEEE Trans. Circuits Syst. Video Technol.* **14**(6), 914–920 (2004).
26. K. H. Chung and Y. H. Chan, "A low-complexity joint color demosaicking and zooming algorithm for digital camera," *IEEE Trans. Image Process.* **16**(7), 1705–1715 (2007).
27. X. Zhang, "S-CIELAB: a spatial extension to the CIE L*a*b* DeltaE color difference metric," see http://white.stanford.edu/~brian/scielab/scielab.html.

**King-Hong Chung** received his BEng (Hons.) and PhD degrees in electronic and information engineering from the Hong Kong Polytechnic University in 2001 and 2009, respectively. He was with the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University as Research Assistant in 2001. From 2003 to 2005, he worked with COMedia Ltd, Hong Kong as electronic engineer. He then re-joined the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University, as Research Associate and postdoctoral fellow in January 2009 and August 2009, respectively. Currently, he is a postdoctoral fellow in the Communication and Network Group, Department of Electrical Engineering, National Cheng Kung University, Taiwan. His research interests include color demosaicing, super-resolution, halftoning, image compression, video coding and 3D TV processing.

**Yuk-Hee Chan** received his BSc degree in electronics from the Chinese University of Hong Kong in 1987, and his PhD degree in signal processing from Hong Kong Polytechnic University in 1992. Between 1987 and 1989, he worked as a research and development engineer at Elec and Eltek Group, Hong Kong. He joined this university in 1992 and is now an associate professor in the Department of Electronic and Information Engineering. He has published more than 120 research papers in various international journals and conferences. His research interests include image and video compression, image restoration, halftoning, demosaicking, and fast computational algorithms in digital signal processing. He is a member of IEEE. He was the Chairman of the IEEE Hong Kong Joint Chapter of CAS and COM in 2003 and 2004.