Heterogeneity-Projection Hard-Decision Color Interpolation Using Spectral-Spatial Correlation

Chi-Yi Tsai and Kai-Tai Song, *Associate Member, IEEE*

***Abstract*—This paper presents a novel heterogeneity-projection hard-decision (HPHD) color interpolation procedure for reproduction of Bayer mosaic images. The proposed algorithm aims to estimate the optimal interpolation direction and perform hard-decision interpolation, in which each pixel only needs to be interpolated once. A new heterogeneity-projection scheme based on a novel spectral-spatial correlation concept is proposed to estimate the best interpolation direction directly from the original mosaic image. Using the proposed heterogeneity-projection scheme, a hard-decision rule can be decided before performing the interpolation. The advantage of this scheme is that it provides an efficient way for decision-based algorithms to generate improved results using fewer computations. Compared with three recently reported demosaicing techniques, Gunturk's, Lu's, and Li's methods, the proposed HPHD outperforms all of them in both PSNR values and S-CIELAB ΔE^*_{ab} measures by utilizing 25 natural images from Kodak PhotoCD.**

***Index Terms*—Adaptive filtering, color artifacts, color filter array (CFA) demosaicing, color reproduction, digital cameras, image representation.**

I. Introduction

DIGITAL color images from single-chip digital still cameras are obtained by interpolating the output from a color filter array (CFA). The CFA consists of a set of spectrally selective filters that are arranged in an interleaved pattern so that each sensor pixel samples one of three primary color components. These sparsely sampled color values are termed mosaic images. To render a full-color image from a mosaic image, an image reconstruction process, commonly known as CFA interpolation or CFA demosaicing, is required to estimate for each pixel its two missing color values. The simplest demosaicing methods apply well-known interpolation techniques, such as nearest-neighbor replication, bilinear interpolation, and cubic spline interpolation, to each color channel separately. However, these single-channel algorithms usually introduce severe color artifacts and blurs around sharp edges [1]. These drawbacks motivate the need of more advanced algorithms for improving demosaicing performance. An excellent review on advanced demosaicing algorithms can be found in [2].

Manuscript received June 27, 2005; revised June 27, 2006. This work was supported by the National Science Council of Taiwan, R.O.C., under Grant NSC 92-2213-E-009-007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Reiner Lenz.

The authors are with the Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: chiyi.ece91@nctu.edu.tw; u9112824@cn.nctu.edu.tw; ktsong@mail.nctu.edu.tw).

Color versions of Figs. 1, 2, 5–7, and 9–11 are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2006.884943

In recent years, there have been investigations on more sophisticated demosaicing algorithms. In [3], Lu and Tan presented an improved hybrid CFA demosaicing method that consists of two successive steps: an interpolation step to render full-color images and a postprocessing step to suppress visible demosaicing artifacts. Muresan and Parks proposed an improved edge-directed demosaicing algorithm based on optimal recovery interpolation of grayscale images [4]. They first utilized a grayscale image interpolation algorithm based on optimal recovery estimation theory to interpolate the green plane. The red/blue channels were interpolated using interchannel color difference adaptive filtering. These two demosaicing algorithms in general produce high quality visual results, especially in reconstructing sharp or well-defined edges of the image. However, in fine details or textured regions, where edges tend to be short and in different directions, these algorithms introduce undesirable errors and give degraded performance.

Meanwhile, two iterative demosaicing techniques were proposed by Gunturk *et al.* [5] and Li [6], respectively. In [5], a projection-onto-convex-set (POCS) technique was presented to estimate the missing color values in red and blue channels using alternating projection scheme based on high interchannel correlation. In [6], Li formulated the CFA demosaicing as a problem of reconstructing correlated signals from decimated versions and proposed a successive approximation strategy by adopting color difference interpolation iteratively. Although these iterative demosaicing algorithms perform well in textured regions and reveal low computational complexity, they cannot produce satisfactorily high quality visual results in well-defined edges of the image.

Another recent demosaicing approach divides the demosaicing procedure into interpolation stage and decision stage [7]–[10]. In the interpolation stage, horizontally and vertically interpolated images are produced respectively. In the latter decision stage, a soft-decision method, in which the interpolation must be performed before the decision procedure, was employed for choosing the pixels interpolated in the direction with fewer artifacts. Because the decision stage is essential for these demosaicing approaches, we refer them as *decision-based demosaicing algorithm*. For the decision stage, Hirakawa *et al.* proposed a homogeneity metric to measure the misguidance level of color artifacts presented in interpolated images [7]. Based on this measurement, the interpolation decision is made by choosing the region with larger homogeneity values. In [8], Wu *et al.* adopted the Fisher's linear discriminant technique to determine the optimal interpolation direction in a local window. In [9], Grossmann and Eldar utilized the YIQ color space as a tool to select the reconstructed regions with a smoother chrominance component. Recently, Omer and Werman pro-

posed an enhanced decision-based demosaicing algorithm that combines the decision process with the standard demosaicing algorithm such as edge-directed scheme [17] to improve its performance in places the standard algorithm tends to fail [10]. The decision-based demosaicing algorithm performs well not only in textured regions, but also in well-defined edges of the image. However, the main drawback of these demosaicing algorithms is that they are not efficient in the interpolation stage because each pixel needs to be interpolated at least twice, one in horizontal direction and the other in vertical direction, for the next soft-decision procedure. This drawback also greatly increases the computing efforts in the latter decision stage. Therefore, it is still a challenge in CFA demosaicing design to develop an efficient color interpolation method with high performance in both textured and edge regions.

In this paper, a novel heterogeneity-projection hard-decision (HPHD) color interpolation algorithm is proposed for color reproduction from Bayer mosaic images. The proposed algorithm aims to estimate the optimal interpolation direction before performing color interpolation. Because the decision stage is performed before the interpolation stage (termed as hard-decision interpolation), each pixel only needs to be interpolated once. To do so, a new heterogeneity-projection scheme based on a novel spectral-spatial correlation concept is proposed to estimate the best interpolation direction directly from the original Bayer mosaic image. Using the proposed heterogeneity-projection scheme, a hard-decision rule can be decided before performing color interpolation. The advantage of the proposed demosaicing algorithm is threefold. First, the proposed heterogeneity-projection scheme can combine with existent decision-based demosaicing algorithms. More specifically, the proposed heterogeneity-projection scheme can adopt into the decision step of existent decision-based demosaicing algorithms. Second, each pixel only has to be interpolated once. Therefore, the proposed algorithm is much more efficient than other decision-based schemes. Finally, the proposed demosaicing algorithm performs well not only in textured regions, but also in well-defined edges of the image.

The rest of this paper is organized as follows. In Section II, the spectral-spatial correlation concept will be described. Section III presents the proposed heterogeneity-projection scheme based on the spectral-spatial correlation. Section IV describes the proposed HPHD interpolation algorithm. In Section V, experimental results and computational complexity of the proposed method are discussed. The demosaicing results of the proposed method are compared with those from other existing methods. Section VI summarizes the contributions of this work. In Appendix A, an experiment of tweaking parameters is presented to find the local optimal parameters for the proposed method.

II. Spectral-Spatial Correlation

Fig. 1 shows the most used CFA pattern, the Bayer pattern [11], where R, G, and B denote, respectively, the pixels having only red, green and blue color values. We limit our discussion to the Bayer pattern in this paper. In the following, image spectral and spatial correlations are first introduced. A novel spectral-spatial correlation is then derived based on these correlations.

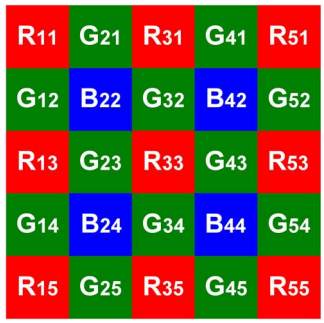

Fig. 1. Bayer CFA pattern (Bayer pattern).

A. Spectral and Spatial Correlations

Many existing demosaicing methods are developed using image spectral and/or spatial correlation. The concept of spectral correlation is based on the assumption that the color difference signals are locally constant in chrominance smooth areas [12]. Let $[R \quad G \quad B]$ denote three color planes of a nature color image, the concept of spectral correlation leads to the following assumption.

A1) The color differences between green and red/blue channels satisfy the following conditions:

$$R(x,y) = G(x,y) + A_{rg}(x,y) \text{ and}$$
$$B(x,y) = G(x,y) + A_{bg}(x,y)$$

where $A_{rg}(x,y)$ and $A_{bg}(x,y)$ are piecewise constant within the boundary of a given object. □

The spatial correlation reflects the fact that within a homogeneous image region, neighboring pixels share similar color values [13]. In other words, the difference between neighboring pixel values along an edge direction in spatial domain is a constant. Thus, we have the following assumption based on the concept of spatial correlation [3].

A2) The rate of change of neighboring pixel values along an edge direction is a constant. □

To illustrate this, let us consider the interpolation of R_{33} in Fig. 1. Suppose that the pixel R_{33} is located on a horizontal edge. Based on A1), the neighboring pixels of R_{33} along the horizontal direction have the following relationship between green and red/blue pixel values:

$$A_{rg}(x-1,y) = A_{rg}(x,y) = A_{rg}(x+1,y) \text{ and}$$
$$A_{bg}(x-1,y) = A_{bg}(x,y) = A_{bg}(x+1,y). \quad (1)$$

So, we have

$$\overline{R}_{23} - G_{23} = R_{33} - \overline{G}_{33} = \bar{R}_{43} - G_{43} \text{ and}$$
$$\overline{B}_{23} - G_{23} = \overline{B}_{33} - \overline{G}_{33} = \overline{B}_{43} - G_{43}. \quad (2)$$

The assumption A2) gives the following relationship on horizontal edges

$$\overline{R}_{23} - R_{33} = R_{33} - \overline{R}_{43} \equiv dR_h$$
$$G_{23} - \overline{G}_{33} = \overline{G}_{33} - G_{43} \equiv dG_h \text{ and}$$
$$\overline{B}_{23} - \overline{B}_{33} = \overline{B}_{33} - \overline{B}_{43} \equiv dB_h \quad (3)$$

where $\overline{G}_{33}$, $\overline{R}_{23}$, $\overline{R}_{43}$, $\overline{B}_{23}$, $\overline{B}_{33}$, and $\overline{B}_{43}$ denote the missing color values at the respective pixel locations. dG_h, dR_h, and dB_h are constants.

B. Spectral-Spatial Correlation (SSC)

A significant characteristic of Bayer pattern is that for each pixel, the surrounding pixels are one of the primary components in different channels. It is then interesting to investigate the relationship between neighboring pixels in different color channels. Consider the following situation: On a horizontal edge, two green pixels surround a red pixel on horizontal direction. Take the difference between the center red pixel and right green pixel, we then have

$$\begin{aligned} &R(x,y)-G(x+1,y)\\ &\quad=\left[R(x,y)-\overline{G}(x,y)\right]+\left[\overline{G}(x,y)-G(x+1,y)\right] \end{aligned} \tag{4}$$

where $\overline{G}(x,y)$ denotes the missing green value at center red pixel location. Recall assumptions A1) and A2), expression (4) becomes such that

$$S_{rg}^{h(x,x+1)}\equiv R(x,y)-G(x+1,y)=A_{rg}(x,y)+dG_h. \tag{5}$$

Similarly, the difference between a blue pixel and its right green pixel is given by

$$S_{bg}^{h(x,x+1)}\equiv B(x,y)-G(x+1,y)=A_{bg}(x,y)+dG_h. \tag{6}$$

The same results also can be obtained along vertical direction on a vertical edge such that

$$\begin{aligned} S_{rg}^{v(y,y+1)}&\equiv R(x,y)-G(x,y+1)=A_{rg}(x,y)+dG_v \text{ and}\\ S_{bg}^{v(y,y+1)}&\equiv B(x,y)-G(x,y+1)=A_{bg}(x,y)+dG_v. \end{aligned} \tag{7}$$

Expressions (5)–(7) show that the difference between surrounding pixels in different color channels is equal to the summation of spectral and spatial correlations. We refer these relationships (5)–(7) as *spectral-spatial correlation* (SSC). SSC has two important characteristics. First, SSC can be easily and directly calculated from the original Bayer mosaic image. Second, SSC inherits the characteristics of spectral and spatial correlations. In other words, SSC is also piecewise constant within the boundary of a given object or along an edge direction. Therefore, we have the following assumption based on these observations:

A3) The SSC defined in (5)–(7) within the boundary of a given object or along an edge direction is also piecewise constant. □

Assumption A3) is a significant clue for us to find the directional smooth regions in Bayer mosaic images directly before performing the interpolation. In Section III, we will present the method of heterogeneity-projection based on A3).

III. Heterogeneity-Projection for Bayer Mosaic Images

The proposed heterogeneity-projection scheme transfers the original Bayer mosaic image directly into horizontal and vertical heterogeneity maps, respectively. Using these two heterogeneity maps, the interpolation direction can be determined easily by choosing the smallest heterogeneity values.

A. Heterogeneity-Projection

Assumption A3) implies that the nth-order directional finite derivative of SSC along an edge direction tends toward a small value. For example, consider a red pixel $R(x,y)$ locates on a horizontal edge, the SSC values of $R(x,y)$ and its neighboring pixels along horizontal direction can be found such that

$$\begin{aligned} S_{rg}^{h(x,x+1)}&=A_{rg}(x,y)+dG_h,\\ S_{gr}^{h(x+1,x+2)}&=-A_{rg}(x+2,y)+dG_h \end{aligned} \tag{8}$$

where $S_{gr}^{h(x+1,x+2)}\equiv G(x+1,y)-R(x+2,y)$. Based on the basic definition of the first-order derivative of a 1-D discrete function, the first-order horizontal derivative of SSC are given by [14]

$$\begin{aligned} dS_{rg}^{h(x,x+3)}&\equiv S_{rg}^{h(x,x+1)}-S_{rg}^{h(x+2,x+3)}\\ &=A_{rg}(x,y)-A_{rg}(x+2,y),\\ dS_{gr}^{h(x+1,x+4)}&\equiv S_{gr}^{h(x+1,x+2)}-S_{gr}^{h(x+3,x+4)}\\ &=A_{rg}(x+4,y)-A_{rg}(x+2,y). \end{aligned} \tag{9}$$

Recall A1) and A3), one can see that $dS_{rg}^{h(x,x+3)}$ and $dS_{gr}^{h(x+1,x+4)}$ both will approach to zero along this horizontal edge. Because the higher-order derivative of a discrete function is a linear combination of the first-order ones, it implies the higher-order horizontal derivative of SSC will also approach to zero along the horizontal edge. Thus, we have the following assumption.

A4) If pixels locate on a directional edge, then the corresponding nth-order directional finite derivative of SSC along the edge direction approaches to zero. □

Assumption A4) poses a question that how the nth-order directional derivative of SSC can be directly calculated from Bayer mosaic image. To resolve this problem, a heterogeneity-projection scheme is developed to transfer the row data of Bayer mosaic image directly into nth-order directional derivative of SSC. Note that the value of nth-order directional derivative of SSC is defined as *heterogeneity measure*, because it leads to a small value within a directional smooth region.

Denote $RG_{1\times N}=[\,R_1\quad G_2\quad R_3\quad \ldots]_{1\times N}$ as a row data of Bayer mosaic image, N is the presetting window size, and H_h is the corresponding horizontal heterogeneity value. To calculate the horizontal heterogeneity value H_h from $RG_{1\times N}$, we propose the following steps. First, the row data $RG_{1\times N}$ is transferred into a $1\times(N-3)$ vector of first-order horizontal finite derivative of SSC using a linear transformation such that

$$\begin{aligned} dS_{1\times(N-3)}&=\left[\,dS_{rg}^{h(1,4)}\quad dS_{gr}^{h(2,5)}\quad dS_{rg}^{h(3,6)}\quad \cdots\right]_{1\times(N-3)}\\ &=RG_{1\times N}T_{N\times(N-3)}^{1} \end{aligned} \tag{10}$$

where $T_{N\times(N-3)}^{1}=[\,1\quad -1\quad -1\quad 1\,]^T\otimes \mathrm{eye}(N-3)$, $\otimes$ denotes the 2-D convolution operator and $\mathrm{eye}(M)$ denotes a $M\times M$ identity matrix. Second, because the higher-order derivative of a discrete function is derived by the linear combination of

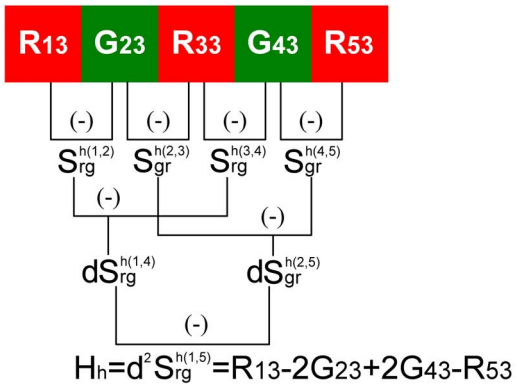

Fig. 2. Concept of horizontal heterogeneity-projection from a 1 × 5 row data of a Bayer mosaic image.

its first-order ones, the horizontal heterogeneity value H_h, the $(N-3)$th-order horizontal derivative of SSC, is obtained such that [14]

$$\begin{aligned} H_h &\equiv d^{N-3}S_{rg}^{h(1,N)} = d^{N-4}S_{rg}^{h(1,N-1)} - d^{N-4}S_{gr}^{h(2,N)} \\ &= d^{N-5}S_{rg}^{h(1,N-2)} - 2d^{N-5}S_{gr}^{h(2,N-1)} + d^{N-5}S_{rg}^{h(3,N)} \\ &\vdots \\ &= w_1 dS_{rg}^{h(1,4)} + w_2 dS_{gr}^{h(2,5)} + w_3 dS_{rg}^{h(3,6)} + \cdots \\ &= dS_{1\times(N-3)}T^2_{(N-3)\times 1} \end{aligned} \tag{11}$$

where $T^2_{(N-3)\times 1} = \prod_{i=1}^{N-4}[1 \quad -1]^T \otimes \mathrm{eye}(N-3-i)$ is a $(N-3)\times 1$ coefficient vector which transfers vector $dS_{1\times(N-3)}$ into the $(N-3)$-order derivative value through Euclidean inner product [15]. Next, substituting (10) into (11) yields

$$H_h = RG_{1\times N}T^1_{N\times(N-3)}T^2_{(N-3)\times 1} = RG_{1\times N}P_{N\times 1} \tag{12}$$

where $P_{N\times 1} = T^1_{N\times(N-3)}T^2_{(N-3)\times 1}$ is a $N \times 1$ vector and referred as *heterogeneity vector*. Expression (12) shows that the horizontal heterogeneity value H_h is the projection of the row data of Bayer mosaic image onto the heterogeneity vector $P_{N\times 1}$. Thus, (12) is termed as *horizontal heterogeneity-projection*. Fig. 2 illustrates an example of horizontal heterogeneity-projection from a 1 × 5 row data of Bayer mosaic image. Using (12), the heterogeneity vector $P_{N\times 1}$ is obtained as

$$P_{5\times 1} = T^1_{5\times 2}T^2_{2\times 1} = [1 \quad -2 \quad 0 \quad 2 \quad -1]^T.$$

The horizontal heterogeneity value H_h of R_{33} is then given by

$$\begin{aligned} H_h &= d^2 S_{rg}^{h(1,5)} \\ &= RG_{1\times 5}P_{5\times 1} \\ &= R_{13} - 2G_{23} + 2G_{43} - R_{53}. \end{aligned}$$

Similarly, the vertical heterogeneity value H_v is the projection of Bayer mosaic image's column data onto the heterogeneity vector $P_{N\times 1}$ such that

$$H_v = RG^T_{N\times 1}P_{N\times 1} \tag{13}$$

where $RG_{N\times 1} = [R_1 \quad G_2 \quad R_3 \quad \cdots]^T_{N\times 1}$ is a column data of Bayer mosaic image. Finally, based on (12) and (13), the horizontal and vertical heterogeneity maps, H_{h_map} and H_{v_map} are obtained, respectively, by

$$\begin{aligned} H_{h\_\mathrm{map}} &= \left|Bayer \otimes P^T_{N\times 1}\right| \text{ and} \\ H_{v\_\mathrm{map}} &= \left|Bayer \otimes P_{N\times 1}\right| \end{aligned} \tag{14}$$

where $Bayer$ denotes the original Bayer mosaic image. One can see from (14) that the horizontal and vertical heterogeneity maps are derived directly from the Bayer mosaic image via horizontal and vertical heterogeneity-projection, respectively.

B. Directional Adaptive Filtering For Error Reduction

Assumption A4) states that the directional heterogeneity-projection along an edge direction leads to a small heterogeneity value. However, a small heterogeneity measure does not imply the directional heterogeneity-projection along a correct edge direction. This problem will induce estimation error in the initial estimated heterogeneity maps. In order to reduce the estimation error, a directional adaptive filter, whose behavior changes based on the statistical characteristics inside a local window, is proposed to reduce the estimation error and estimate the optimal heterogeneity maps. Moreover, since each heterogeneity measure in the initial heterogeneity maps is static, this estimation problem is equivalent to a static estimation problem, in which the estimation errors are modeled as the zero mean Gaussian noises with nonzero variance. According to [16], the minimum mean square-error (MMSE) solution of the static estimation problem can be obtained using a predictor-corrector filter. Therefore, the design of the proposed directional adaptive filter adopts the structure of predictor-corrector filter to obtain the MMSE estimates. The interested reader is referred to [16] for more technical details.

The proposed directional adaptive filter is divided into horizontal and vertical adaptive filters. For the horizontal heterogeneity map, only the horizontal adaptive filter is applied to it. Fig. 3(a) illustrates the concept of horizontal adaptive filter. In Fig. 3(a), the center pixel H_h is to be adaptively filtered along the horizontal direction based on statistical measures of surrounding pixels H_h^R and H_h^L. The simplest statistical measures of H_h^R and H_h^L are their mean and variance in a local window [14]. For instance, if a 1 × 3 rectangular window defines the window size, the local mean and variance of H_h^R and H_h^L are, respectively, given by

$$\begin{aligned} \overline{H}_h^R &= \frac{(H_h + H_h^R + H_h^{RR})}{3} \\ \delta H_h^R &= \frac{\left[\left(\overline{H}_h^R - H_h\right)^2 + \left(\overline{H}_h^R - H_h^R\right)^2 + \left(\overline{H}_h^R - H_h^{RR}\right)^2\right]}{3} \end{aligned} \tag{15}$$

$$\begin{aligned} \overline{H}_h^L &= \frac{(H_h + H_h^L + H_h^{LL})}{3} \\ \delta H_h^L &= \frac{\left[\left(\overline{H}_h^L - H_h\right)^2 + \left(\overline{H}_h^L - H_h^L\right)^2 + \left(\overline{H}_h^L - H_h^{LL}\right)^2\right]}{3}. \end{aligned} \tag{16}$$

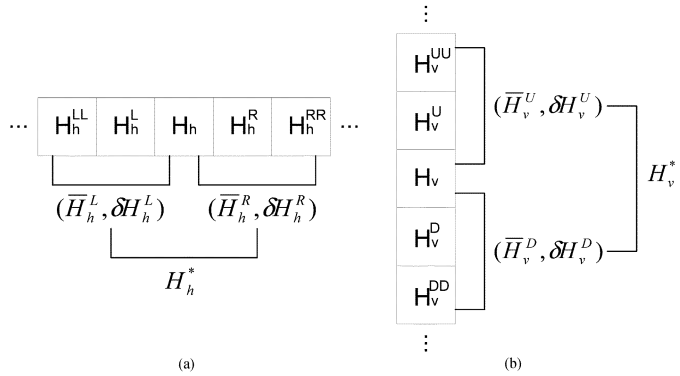

Fig. 3. Concept of (a) horizontal and (b) vertical adaptive filtering using a rectangular window.

Using (15) and (16), the adaptively filtered pixel H_h^* is obtained as follows:

$$H_h^* = \overline{H}_h^L + \frac{\delta H_h^L}{\delta H_h^L + \delta H_h^R}\left(\overline{H}_h^R - \overline{H}_h^L\right). \tag{17}$$

In (17), the local mean $\overline{H}_h^L$ is the predictor term with an associated error variance δH_h^L, and the local mean $\overline{H}_h^R$ is the corresponding corrector term with error variance δH_h^R. Therefore, (17) provides the MMSE estimate of the horizontal heterogeneity measure in a local window. Fig. 3(b) illustrates an example of vertical adaptive filter for vertical heterogeneity map. Using the same procedure discussed above, the adaptively filtered pixel H_v^* is obtained as follows:

$$H_v^* = \overline{H}_v^U + \frac{\delta H_v^U}{\delta H_v^U + \delta H_v^D}\left(\overline{H}_v^D - \overline{H}_v^U\right) \tag{18}$$

where $\left(\overline{H}_v^U, \delta H_v^U\right)$ and $\left(\overline{H}_v^D, \delta H_v^D\right)$ are the local mean and variance of H_v^U and H_v^D. Similarly, (18) also provides the MMSE estimate of the vertical heterogeneity measure in a local window. After adopting the horizontal and vertical adaptive filters presented above into horizontal and vertical heterogeneity maps, respectively, the MMSE estimates of horizontal and vertical heterogeneity maps $H_{h\_\mathrm{map}}^*$ and $H_{v\_\mathrm{map}}^*$ are obtained.

IV. Hard-Decision Color Interpolation

With the horizontal and vertical heterogeneity maps, a hard-decision rule is applied for color interpolation. First, we classify three subsets in the image such that

$$\begin{aligned}
\Omega_h &\equiv \left\{(x,y)|H_{h_\mathrm{map}}^*(x,y) < \alpha H_{v_\mathrm{map}}^*(x,y)\right\} \\
\Omega_v &\equiv \left\{(x,y)|H_{v_\mathrm{map}}^*(x,y) < \alpha H_{h_\mathrm{map}}^*(x,y)\right\} \\
\Omega_s &\equiv \{(x,y)|(x,y) \notin \Omega_h, (x,y) \notin \Omega_v\}
\end{aligned} \tag{19}$$

where Ω_h, Ω_v, and Ω_s denote the horizontal, vertical, and smooth subsets, respectively. α is a positive constant satisfying $0 \le \alpha \le 1$. Second, based on (19), the concept of hard-decision rule for interpolation is obtained in (20), shown at the bottom of the page.

In the following discussion, a color interpolation method is developed based on the hard-decision rule (20).

Remark 1: The parameter α in (19) determines the size of smooth subset in the image. A small (large) α leads to a large (small) smooth subset in the image. For example, if $\alpha = 0$, the image only contains smooth subset without horizontal and vertical subsets. Based on (20), the interpolation of image only adopts the weight averaging of neighboring pixels on each missing color channel [3], [13], [17]. On the contrary, for $\alpha = 1$, the image only contains horizontal and vertical subsets but without smooth subset. The interpolation of image only adopts horizontal and vertical interpolations on each missing color channel [7], [8]. Therefore, for $0 < \alpha < 1$, the hard-decision rule (20) is characterized by weight averaging and directional interpolating.

A. Hard-Decision Adaptive Interpolation

We first interpolate green channel because the green plane possesses most spatial information of the image. Each missing green value G_{miss} is to be estimated from its four surrounding green pixels by (21), shown at the bottom of the page, where $\hat{G}_{\{\mathrm{up,right,down,left}\}}$ denote the color-adjusted green values of four surrounding green pixels, and $e_{\{\mathrm{up,right,down,left}\}}$ denote the corresponding edge indicators. In our method, the following

$$\begin{aligned}
&if\ (x,y) \in \Omega_h \\
&\quad \text{Perform } \textit{horizontal} \text{ interpolation on each missing color channel} \\
&elseif\ (x,y) \in \Omega_v \\
&\quad \text{Perform } \textit{vertical} \text{ interpolation on each missing color channel} \\
&else \\
&\quad \text{Perform } \textit{weight averaging} \text{ of neighboring pixels on each missing color channel}
\end{aligned} \tag{20}$$

$$G_{\mathrm{miss}} = \frac{e_{\mathrm{up}}\hat{G}_{\mathrm{up}} + e_{\mathrm{right}}\hat{G}_{\mathrm{right}} + e_{\mathrm{down}}\hat{G}_{\mathrm{down}} + e_{\mathrm{left}}\hat{G}_{\mathrm{left}}}{e_{\mathrm{up}} + e_{\mathrm{right}} + e_{\mathrm{down}} + e_{\mathrm{left}}} \tag{21}$$

modification on edge indicators is adopted according to the hard-decision rule (20), such that

$$\begin{aligned} &\text{if} \quad (x,y) \quad \in \Omega_h \\ &(e_{\text{up}}, e_{\text{down}}) = (0,0) \\ &\text{else if } (x,y) \in \Omega_v \\ &(e_{\text{right}}, e_{\text{left}}) = (0,0). \end{aligned} \tag{22}$$

Therefore, the hard-decision adaptive interpolation for green channel is summarized in (23), shown at the bottom of the page.

Remark 2: The color-adjusted green value is the green value adjusted with the help of the surrounding red/blue pixels along the respective interpolation directions. The derivation of color-adjusted value is based on the assumptions of spectral correlation A1) and spatial correlation A2) discussed in Section II-A. Interested reader can refer [3] for detailed derivations of color-adjusted values. In this paper, the formulation of each surrounding color-adjusted green value in (23) adopts the results in [3], while the corresponding edge indicator can be referred in [3], [13], and [17]. In the remainder of this paper, the color-adjusted value of each color pixel and the corresponding edge-indicator are determined by adopting the procedure presented in [3].

When the green channel has been fully recovered, it can be used in the interpolation of red and blue channels. The interpolation procedure of red and blue channels consists of two substeps: 1) interpolating the missing red/blue values at blue/red pixels, and 2) interpolating the rest of the missing red/blue values at green pixels. In our method, we only apply the hard-decision rule (20) to the substep 2) because there is not enough information to perform horizontal and vertical interpolations in substep 1). Since the same procedure is applied to interpolate the red and blue channels, only the red channel interpolation is presented.

Let R^b_{miss} denote a missing red value at a blue pixel. It is estimated from its four neighboring red pixels by (24), shown at the bottom of the page, where $\hat{R}_{\{\text{up-right,down-right,down-left,up-left}\}}$ denote the color-adjusted red values of four neighboring red pixels, and $e_{\{\text{up-right,down-right,down-left,up-left}\}}$ denote the corresponding edge indicators. For example, in Fig. 1, the missing red value R^b_{22} at blue pixel B_{22} is estimated by

$$R^b_{22} = \frac{e_{31}\hat{R}_{31} + e_{33}\hat{R}_{33} + e_{13}\hat{R}_{13} + e_{11}\hat{R}_{11}}{e_{31} + e_{33} + e_{13} + e_{11}}. \tag{25}$$

Subsequently, the rest of the missing red values at green pixels are estimated using the same procedure performed for the green channel. Each missing red value at a green pixel R^g_{miss} can be estimated from its four surrounding red pixels by the hard-decision adaptive interpolation in (26), shown at the bottom of the page, where $\hat{R}_{\{\text{up,right,down,left}\}}$ denote the color-adjusted red values of four surrounding red pixels, and $e_{\{\text{up,right,down,left}\}}$ are the corresponding edge indicators. Finally, a full-color image can be obtained by applying the same interpolation steps described above on each missing blue value.

Remark 3: Although adaptive interpolation can provide more pleasing results, it also increases the computational load and the amount of memory transactions compared with linear interpolation [12]. In order to reduce the computational cost in the color interpolation step, we can still use linear interpolation instead of the adaptive interpolation. More specifically, for linear interpolation, the edge indicators $e_{\{\text{up,right,down,left}\}}$ in (13) and (26) are simplified such that

$$(e_{\text{up}}, e_{\text{down}}, e_{\text{right}}, e_{\text{left}}) = \begin{cases} (0,0,1,1), & \text{if } (x,y) \in \Omega_h \\ (1,1,0,0), & \text{if } (x,y) \in \Omega_v \\ (1,1,1,1), & \text{if } (x,y) \in \Omega_s \end{cases}. \tag{27}$$

$$G_{\text{miss}} = \begin{cases} \frac{e_{\text{right}}\hat{G}_{\text{right}} + e_{\text{left}}\hat{G}_{\text{left}}}{e_{\text{right}} + e_{\text{left}}}, & \text{if } (x,y) \in \Omega_h \\ \frac{e_{\text{up}}\hat{G}_{\text{up}} + e_{\text{down}}\hat{G}_{\text{down}}}{e_{\text{up}} + e_{\text{down}}}, & \text{if } (x,y) \in \Omega_v \\ \frac{e_{\text{up}}\hat{G}_{\text{up}} + e_{\text{right}}\hat{G}_{\text{right}} + e_{\text{down}}\hat{G}_{\text{down}} + e_{\text{left}}\hat{G}_{\text{left}}}{e_{\text{up}} + e_{\text{right}} + e_{\text{down}} + e_{\text{left}}}, & \text{if } (x,y) \in \Omega_s \end{cases} \tag{23}$$

$$R^b_{\text{miss}} = \frac{e_{\text{up-right}}\hat{R}_{\text{up-right}} + e_{\text{down-right}}\hat{R}_{\text{down-right}} + e_{\text{down-left}}\hat{R}_{\text{down-left}} + e_{\text{up-left}}\hat{R}_{\text{up-left}}}{e_{\text{up-right}} + e_{\text{down-right}} + e_{\text{down-left}} + e_{\text{up-left}}} \tag{24}$$

$$R^g_{\text{miss}} = \begin{cases} \frac{e_{\text{right}}\hat{R}_{\text{right}} + e_{\text{left}}\hat{R}_{\text{left}}}{e_{\text{right}} + e_{\text{left}}}, & \text{if } (x,y) \in \Omega_h \\ \frac{e_{\text{up}}\hat{R}_{\text{up}} + e_{\text{down}}\hat{R}_{\text{down}}}{e_{\text{up}} + e_{\text{down}}}, & \text{if } (x,y) \in \Omega_v \\ \frac{e_{\text{up}}\hat{R}_{\text{up}} + e_{\text{right}}\hat{R}_{\text{right}} + e_{\text{down}}\hat{R}_{\text{down}} + e_{\text{left}}\hat{R}_{\text{left}}}{e_{\text{up}} + e_{\text{right}} + e_{\text{down}} + e_{\text{left}}}, & \text{if } (x,y) \in \Omega_s \end{cases} \tag{26}$$

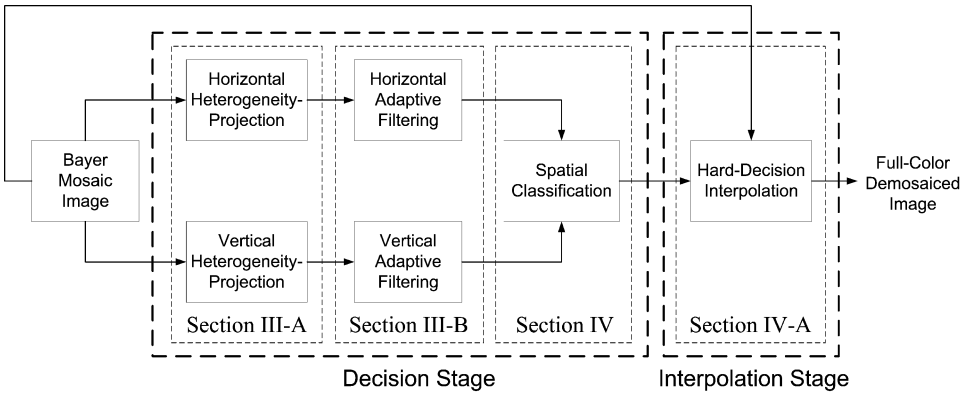

Fig. 4. Flowchart of the proposed HPHD color interpolation algorithm.

And the edge indicators $e_{\{\mathrm{up-right},\mathrm{down-right},\mathrm{down-left},\mathrm{up-left}\}}$ in (24) are fixed such that

$$(e_{\mathrm{up-right}}, e_{\mathrm{down-right}}, e_{\mathrm{down-left}}, e_{\mathrm{up-left}}) = (1,1,1,1). \tag{28}$$

The advantage of linear interpolation is that it not only can skip the calculation of edge indicators, but also use bit-shift instead of division to reduce the computation time. Therefore, compared with adaptive interpolation, the computational cost of linear interpolation will be greatly reduced.

B. Complete HPHD Color Interpolation Algorithm

We summarize the proposed HPHD color interpolation algorithm as follows.

1) Initialization: Set window size N to calculate the heterogeneity vector $P_{N\times 1}$ by $T^1_{N\times(N-3)}$ and $T^2_{(N-3)\times 1}$ defined in (10) and (11), respectively; set parameter α for spatial classification.
2) Decision Stage.
   a) Heterogeneity-projection: Calculate the horizontal and vertical heterogeneity maps, H_{h_map} and H_{v_map}, from original Bayer mosaic image by (14).
   b) Directional adaptive filtering: Filter the horizontal and vertical heterogeneity maps by directional adaptive filters (17) and (18), respectively.
   c) Spatial classification: Use parameter α and the two filtered heterogeneity maps to classify the image into three subsets Ω_h, Ω_v, and Ω_s by (19).
3) Interpolation Stage.
   a) Interpolate G channel at R and B pixels by interpolation rule (23).
   b) Interpolate R channel at B pixels by interpolation rule (24) and the B channel similarly.
   c) Interpolate R channel at G pixels by interpolation rule (26) and the B channel similarly.

Fig. 4 illustrates the flowchart of the proposed HPHD color interpolation algorithm. The main difference between the proposed algorithm and the existent decision-based schemes is that the decision stage is performed before the interpolation stage in this design, thanks to the heterogeneity-projection. This advantage contributes not only to improving the quality of demosaicing result, but also to reducing the computational complexity of the decision stage. In Section V, a comparative study of experimental results and analysis of computational complexity will be discussed to demonstrate the performance of the proposed method.

C. Example Study

Fig. 5 illustrates the execution steps of the proposed algorithm by using an example. The Kodak small Lighthouse image (384×256) is downsampled into a Bayer mosaiced image as shown in Fig. 5(a). In this picture, the fence regions usually challenge the performance of a demosaicing procedure. Fig. 5(b) and (c), respectively, are the horizontal heterogeneity map H_{h_map} and vertical heterogeneity map H_{v_map} obtained from (14) discussed in Section III-A ($N = 24$ in this example). Through the directional adaptive filtering discussed in Section III-B, the filtered horizontal heterogeneity map $H^*_{h\_\mathrm{map}}$ and filtered vertical heterogeneity map $H^*_{v\_\mathrm{map}}$ are obtained in Fig. 5(d)–(e), respectively. Comparing Fig. 5(d)–(e) with Fig. 5(b)–(c), one can see that the unwanted noises in both original heterogeneity maps have been removed effectively by using the directional adaptive filters. Employing two filtered heterogeneity maps, the horizontal, vertical, and smooth subsets of the image are obtained directly by (19) with $\alpha = 0.8$. Fig. 5(f) shows three decided subsets of the image, where the gray region is the horizontal subset Ω_h, the white region is the vertical subset Ω_v, and the black region is the smooth subset Ω_s. Note that Fig. 5(f) shows that the decisions in fence regions are almost all vertical. The interpolations are, thus, along the correct directions. Finally, the proposed hard-decision interpolation discussed in Section IV was applied to reconstruct the color image based on these three decided subsets. Fig. 5(g) illustrates the interpolation results. In Fig. 5(g), one can see that the fine details of interpolation such as the fence and house regions are reconstructed successfully.

To further illustrate the performance, we tweak parameter α and compare the demosaicing results with original image ($N = 24$ is fixed). Fig. 6(a) is the zoom-in of the original fence regions. Fig. 6(b) is the zoom-in of demosaicing result with parameter $\alpha = 0$. One can see that the demosaiced image contains many color artifacts due to the inaccurate smooth interpolation. Fig. 6(c) and (d) show the demosaicing results with parameter $\alpha = 0.5$ and $\alpha = 0.8$, respectively. It is clear that the proposed

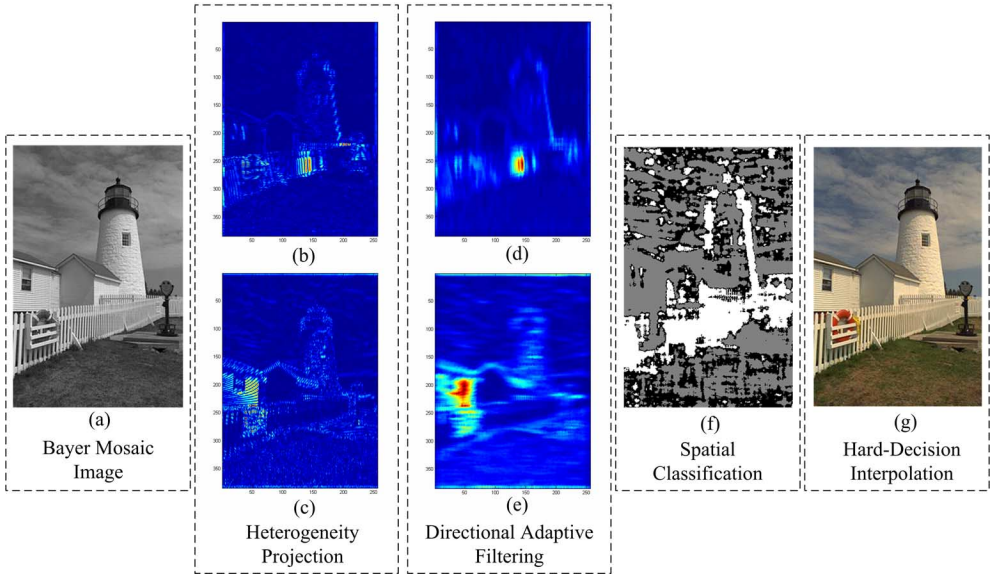

Fig. 5. Illustration of execution steps of the proposed HPHD color interpolation algorithm. (a) Original Bayer mosaic image of small Lighthouse image (384 $\times$ 256). (b) Horizontal heterogeneity map H_{h_map} $(N = 24)$. (c) Vertical heterogeneity map H_{v_map}. (d) Filtered horizontal heterogeneity map $H^{*}_{h\_\mathrm{map}}$. (e) Filtered vertical heterogeneity map $H^{*}_{v\_\mathrm{map}}$. (f) Three decided subsets in the image $(\alpha = 0.8)$. The gray region is the horizontal subset Ω_h, the white region is the vertical subset Ω_v, and the black region is the smooth subset Ω_s. (g) Interpolation result using the proposed hard-decision adaptive interpolation presented in Section IV-A.

Fig. 6. (a) Zoom-in of the original Lighthouse image in the fence region. Zoom-in of the demosaicing results with parameters $N = 24$; (b) $\alpha = 0$; (c) $\alpha = 0.5$; (d) $\alpha = 0.8$.

hard-decision interpolation method reduces the color artifacts efficiently. Visually compare Fig. 6(d) with Fig. 6(a), one can see that most detail features have been reconstructed correctly.

V. Experimental Results

In the experiments, 25 Kodak photographic images as shown in Fig. 7 were employed for demonstrating the demosaicing performance. According to [18], the CFA operations in a digital-camera pipeline usually include a demosaiced image postprocessing framework to provide more visually pleasing color output. Therefore, we introduce the post processing framework in the experiments to complete the comparisons. Fig. 8 illustrates the flowchart of the experiment, which contains interpolation and postprocessing steps. In the interpolation step, the demosaiced results of the proposed method, HPHD linear interpolation (HPHD-LI) and HPHD adaptive interpolation (HPHD-AI) methods, are compared with those using bilinear interpolation and three recently published methods: Lu's [3], Gunturk's [5], and Li's [6] methods. The above schemes are chosen due to their high citation rate in peer-reviewed literature [2]–[8], [13] and represent the state of the technology of CFA demosaicing. For Gunturk's method, we make use of one-level (1-L) decomposition with eight projection iterations in the experiments. For Li's method, the universal threshold value $\delta_1 = \delta_h = 4$ and maximum iteration number $\mathrm{iter} = 20$ are chosen in the experiments. For the proposed method, an experiment of tweaking parameters (N, α) presented in Appendix A was set to find the local optimal parameters for these 25 test images. The local optimal parameters were given by $(N, \alpha) = (11, 0.6)$, which were chosen in the experiments. Subsequently, Lu's postprocessing method was adopted as the postprocessing procedure for each demosaicing method. The demosaiced results in each step were compared accordingly. As shown in Fig. 1, all test images were down-sampled to obtain the Bayer pattern and then reconstructed using the demosaicing methods under comparison in RGB color space.

Two performance measures were adopted in the experiments: PSNR and S-CIELAB ΔE^{*}_{ab} metric [3], [6], [19] to evaluate

Fig. 7. Test images selected from Kodak PhotoCD used in the experiment.

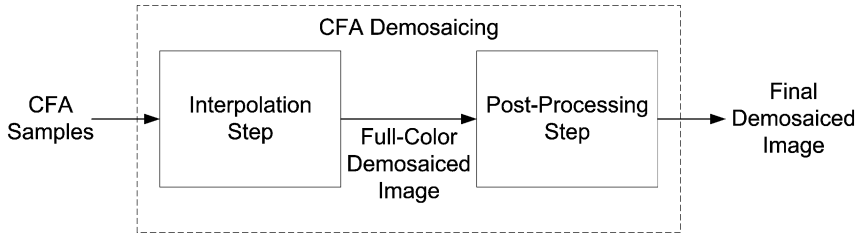

Fig. 8. Flowchart of the experiment. In the interpolation step, we compare the performance of bilinear, Lu's, Gunturk's, Li's, and the proposed HPHD-AI methods. In the postprocessing step, Lu's postprocessing method is adopted into each demosaicing method.

the quality of the demosaiced images. The PSNR (in decibels) metric in this paper is defined as

$$\mathrm{PSNR}(O,D) = 10\log_{10}\left\{255^2\left(\frac{1}{UV}\times\sum_{1\le y\le V}\sum_{1\le x\le U}\|O(x,y)-D(x,y)\|^2\right)^{-1}\right\} \quad (29)$$

where U, V are the total column and row number of the image, $O(x,y)$ is the color vector at the (x,y) th position of the original color image, and $D(x,y)$ is the corresponding color vector in the demosaiced color image. Note that, for a demosaiced image, high fidelity implies large PSNR and small S-CIELAB ΔE^*_{ab} measures.

A. Quantitative Comparison

Table I records the PSNR values and S-CIELAB ΔE^*_{ab} measures of the demosaiced results obtained by the proposed interpolation method together with those from other methods for comparison. In each step, the bold font denotes the largest PSNR and smallest ΔE^*_{ab} values across each row. Moreover, since Gunturk's and Li's methods are iterative and others are noniterative, we categorized these methods into iterative and noniterative groups for more detailed comparisons. From Table I, one can see that Li's and HPHD-AI methods provide improved demosaiced fidelity in most of the test images in the interpolation step. However, when one compares the average PSNR and ΔE^*_{ab} measures in the interpolation step, HPHD-AI generates the highest fidelity demosaiced images, followed by the Gunturk's or other methods.

In the postprocessing step, Table I indicates an interesting phenomenon that all noniterative methods have significant improvement compared with iterative ones, especially the bilinear interpolation (BI). On average, the improvement of BI can add-up the PSNR and reduce ΔE^*_{ab} of the interpolation results by 5.8031 dB and 1.4345 units, respectively. The other noniterative methods also have noticeable improvement on average. In contrast, the iterative methods, e.g., Gunturk's and Li's methods, only have modest improvement through the postprocessing step on average. These observations also can be seen in [18], where the postprocessing step provides the most significant improvement with BI and the smallest improvement with Gunturk's method. Therefore, the experimental results presented in Table I as well as [18] pose a question why postprocessing is more beneficial to the interpolation results of noniterative approaches compared to that of iterative ones. The main reasons are as follows.

Many color interpolation schemes, especially the simple ones such as BI or HPHD-LI, usually induce visible artifacts due

TABLE I
PSNR (DECIBELS) AND ΔE^*_{ab} MEASURES OF DEMOSAICED IMAGES IN THE INTERPOLATION AND POSTPROCESSING STEPS

Step	Interpolation Step						Post-Processing Step					
Group	Iterative		Non-Iterative				Iterative		Non-Iterative			
Method	Gun.[5]	Li [6]	Bilinear	Lu [3]	HPHD-LI	HPHD-AI	Gun.[5]	Li [6]	Bilinear	Lu [3]	HPHD-LI	HPHD-AI
1	29.3765	28.4957	24.8843	31.0257	30.5902	**31.3390**	29.2676	28.3192	28.8878	30.7940	30.5619	**31.1022**
	1.7666	1.8899	2.7289	1.5357	1.6579	**1.5096**	1.7845	1.9083	1.9067	1.5466	1.6239	**1.5126**
2	33.2296	**33.6676**	21.5611	31.6889	31.3683	31.8131	33.6595	33.9846	27.3176	33.8433	34.0348	**34.3242**
	1.5972	**1.5396**	5.1618	1.7135	1.7700	1.6990	1.5445	1.4974	2.8440	1.4668	1.4397	**1.4078**
3	34.7577	35.2213	28.3682	35.7152	35.4998	**36.0111**	34.6331	35.0579	33.2804	35.7232	36.0086	**36.2513**
	1.6598	1.5958	3.1280	1.4910	1.5541	**1.4591**	1.6721	1.6008	1.9659	1.4943	1.4923	**1.4324**
4	36.6168	36.3808	29.7242	37.3966	37.3608	**37.9582**	36.7206	36.3960	35.3870	38.0096	37.9466	**38.4163**
	0.9774	0.9766	1.7728	0.9094	0.9143	**0.8766**	0.9635	0.9615	1.0873	0.8576	0.8596	**0.8272**
5	34.9839	34.8997	28.8694	35.4482	34.9540	**35.5235**	34.9657	34.7714	34.4611	36.1356	35.6365	**36.2585**
	1.3508	1.3260	2.5128	**1.3020**	1.3565	1.3036	1.3075	1.3213	1.4752	1.1861	1.2306	**1.1759**
6	32.6411	31.8126	21.8873	32.7081	31.8449	**32.7811**	32.6069	31.6062	29.3927	33.7802	33.4118	**33.9482**
	2.1864	2.3790	5.7055	**2.0318**	2.3001	2.0417	2.1709	2.3857	2.8618	1.8551	1.9444	**1.8263**
7	**34.0239**	33.8198	23.0206	32.4465	33.4723	33.9217	34.3593	34.5397	28.6949	34.0965	35.6736	**35.8956**
	1.2157	1.2266	3.6536	1.2998	1.2130	**1.1590**	1.1896	1.1590	2.0249	1.1592	1.0243	**0.9972**
8	36.8763	36.7725	28.7405	37.9098	36.8668	**37.9233**	36.6670	36.4265	35.3671	38.1854	37.5005	**38.3097**
	1.1338	1.1444	2.1614	0.9885	1.0789	**0.9830**	1.1576	1.1699	1.2509	0.9694	1.0442	**0.9597**
9	30.8332	**31.2495**	18.7578	29.7212	29.2916	30.4949	31.1581	31.4196	23.9221	31.3071	31.3034	**32.2654**
	1.7679	**1.7192**	5.9022	1.8327	1.9456	1.7327	1.7214	1.6968	3.4143	1.6277	1.6376	**1.5092**
10	36.7662	**37.2501**	27.5750	36.8133	37.0465	37.1882	37.0662	37.2927	33.3472	37.8106	**38.3269**	38.2854
	0.8925	**0.8255**	1.8783	0.8758	0.8258	0.8524	0.8491	0.8226	1.1263	0.7919	**0.7465**	0.7650
11	36.7975	**37.0956**	27.6351	36.8098	36.5621	36.9169	37.0497	37.0952	34.1833	37.5213	37.7538	**37.7553**
	0.8954	**0.8286**	1.8566	0.8715	0.8716	0.8710	0.8536	0.8263	1.0627	0.7926	0.7828	**0.7823**
12	**34.5407**	34.4102	24.3568	33.8725	33.6367	34.2984	34.6820	34.7541	30.3304	35.2610	35.4407	**35.8845**
	1.4748	1.4275	3.8112	1.4666	1.5243	**1.4177**	1.4288	1.3622	2.1498	1.3140	1.3067	**1.2434**
13	37.8205	37.7569	28.7032	37.3884	37.7280	**38.1802**	37.9377	37.8628	34.1471	38.3279	38.9659	**39.2314**
	0.6731	0.6760	1.5011	0.6695	0.6527	**0.6377**	0.6665	0.6610	0.8955	0.6267	0.5971	**0.5875**
14	29.7386	**30.4264**	19.0903	27.8600	27.5973	28.0143	30.2466	**30.8242**	25.6167	30.2549	30.3762	30.6383
	2.5595	**2.4457**	7.1986	2.4652	2.9558	2.8073	2.8077	2.3680	3.7864	2.3619	2.3839	**2.3102**
15	30.8370	29.6090	24.4266	32.4833	31.7108	**32.6772**	30.6644	29.3860	30.1370	32.6128	31.9835	**32.9406**
	1.9406	2.1114	4.1290	1.7491	1.8908	**1.7195**	1.9284	2.1159	2.3280	1.6518	1.7319	**1.5988**
16	34.4301	34.3050	28.2748	34.4161	34.5631	**34.6991**	34.3523	34.2067	33.7516	34.9354	35.1211	**35.1327**
	1.4764	1.4804	2.5666	1.3868	1.4168	**1.3735**	1.4682	1.4704	1.5735	1.3388	1.3436	**1.3179**
17	37.3602	37.0917	26.5250	35.6650	37.1741	**37.6200**	37.6885	37.8239	31.8934	37.2329	39.1877	**39.4594**
	0.9964	1.0009	2.8594	1.0971	0.9836	**0.9556**	0.9740	0.9477	1.6581	0.9865	0.8509	**0.8337**
18	36.2947	**36.4685**	27.2295	35.7449	35.4640	35.8602	36.5932	36.6429	33.7658	36.8960	36.9038	**37.0233**
	1.4628	**1.3340**	2.8610	1.4857	1.5146	1.4874	1.3572	1.3100	1.6517	1.3056	1.3098	**1.2994**
19	**32.3393**	32.3295	23.3178	31.6767	31.3516	31.6487	32.5119	32.2416	29.5746	32.9921	32.8655	**33.0483**
	2.3592	2.3903	5.0440	**2.2879**	2.3566	2.3067	2.3137	2.4326	2.8095	**2.0898**	2.1323	2.0900
20	34.9738	**35.2707**	23.1302	34.5020	34.6201	35.1984	35.2671	35.5570	28.4598	35.7424	36.1790	**36.6174**
	1.3061	**1.2493**	3.6480	1.3409	1.3254	1.3024	1.2452	1.2094	2.0810	1.1902	1.1547	**1.1439**
21	35.7991	35.7714	26.8448	**35.8899**	35.1749	35.8708	36.0108	35.9894	33.1566	36.8055	36.5337	**36.9362**
	1.0396	1.0294	2.1996	**1.0016**	1.0650	1.0052	1.0077	0.9971	1.2790	0.9230	0.9458	**0.9154**
22	**34.0980**	33.8535	23.7154	33.0809	32.5035	33.1734	34.3656	34.4198	29.6633	34.6893	34.4820	**34.9650**
	1.3142	1.3468	3.4864	1.3691	1.4679	1.3721	1.2900	1.2757	1.9425	1.2138	1.2435	**1.1885**
23	32.8830	32.9540	25.5756	33.5303	32.9888	**33.5531**	32.8127	32.8188	30.7133	33.7291	33.4516	**33.8199**
	1.5024	1.5250	2.9269	**1.3922**	1.4540	1.3935	1.5307	1.5364	1.7699	1.3651	1.4050	**1.3562**
24	37.0203	37.0820	30.3178	38.0689	37.9213	**38.2043**	36.9022	36.9148	36.3310	38.1993	38.2472	**38.4723**
	0.9664	0.9820	1.4579	0.8977	0.9063	**0.8951**	0.9867	0.9956	0.9942	0.8945	0.8976	**0.8848**
25	29.8870	**30.0755**	21.9902	29.4449	29.7334	29.8290	30.0602	30.0909	27.8193	30.0984	**30.7241**	30.4261
	1.4933	1.5055	3.6050	1.4432	1.5206	**1.4329**	1.4858	1.5123	1.9570	1.3599	1.3714	**1.3366**
Avg.	34.1970	34.1628	25.3809	34.0523	33.8810	**34.4279**	34.3300	34.2577	31.1840	34.9994	35.1448	**35.4963**
	1.4403	1.4382	3.3503	1.4099	1.4609	**1.3838**	1.4145	1.4217	1.9158	1.2948	1.3000	**1.2521**
Add-up in Average							0.1330	0.0949	**5.8031**	0.9471	1.2638	1.0684
							-0.0258	-0.0165	**-1.4345**	-0.1151	-0.1609	-0.1317

to the nonsmooth local color ratios and color differences (red-green and blue-green). The function of current postprocessing schemes is to correct the interpolated color values by enforcing the local color ratio rule [17], [18] and color difference rule [3] of initial demosaiced image. Similarly, the principle of iterative demosaicing approaches [5], [6] is to iteratively update the initial interpolation result by fitting the local color difference rule. For example, according to [8], the idea of Gunturk's iterative method is equivalent to the filtering of down sampled color difference images of the initial interpolated image by a 5×5 2-D low-pass filter for reducing the high frequency energy of reconstructed color difference images without changing original mosaic samples. In [6], Li utilized the Hamilton-Adams' method [20] and BI to get initial estimates of missing green and red/blue samples, respectively. The following iterative procedure is equivalent to linear low-pass filtering of the color difference image until the reconstructed results converge to a smooth one. In other words, the iterative demosaicing approaches can be regarded as an initial interpolation combined with a meta-algorithm that performs iterative linear low-pass filtering of color difference images to enforce the local color difference rule on initial interpolated image, which is also the main purpose of the latter postprocessing step. Therefore, postprocessing only provides modest improvement for iterative approaches.

Summarizing the above discussion on the experimental results, we have the following conclusions.

1) For iterative approaches, postprocessing only provides the modest improvement due to both have the same purpose of enforcing the local color difference rule on the initial demosaiced image.

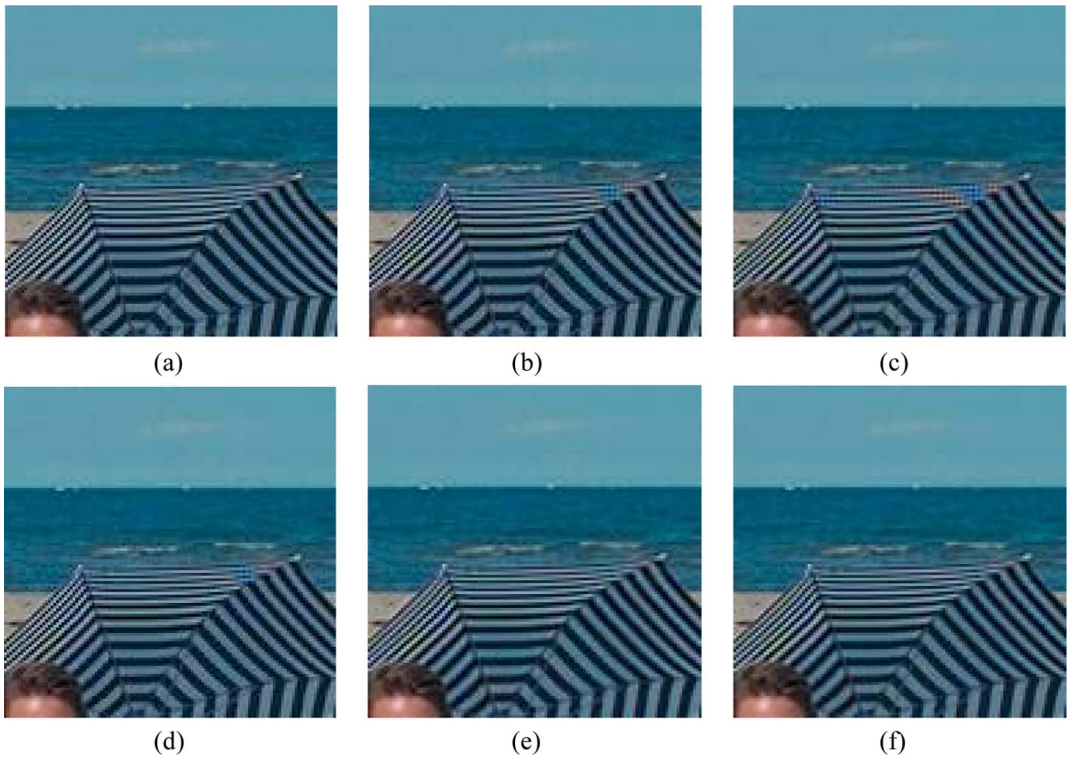

Fig. 9. Zoom-in demosaicing results of test image No. 1. (a) Original picture; Demosaiced result in interpolation step. (b) Gunturk's method. (c) Lu's method. (d) Li's method. (e) HPHD-LI method. (f) HPHD-AI method.

2) On the contrary, postprocessing for noniterative approaches, especially simple linear interpolation schemes such as BI or HPHD-LI schemes, provides significant improvement due to its enforcing on the smoothness of local color ratios and color differences.
3) Because the proposed HPHD-AI scheme is noniterative and provides the best interpolation results in interpolation step, it also has great improvement and obtains the best results after the postprocessing step.

B. Visual Comparison

Figs. 9(a) and 10(a) show the zoom-in of test images No. 1 and 20, respectively. Both scenes contain many fine detail features, such as fine fiber patterns (Fig. 9) and picket fences (Fig. 10), and can effectively challenge the performance of demosaicing methods. Figs. 9(b) and 10(b), 9(c) and 10(c), 9(d) and 10(d), 9(e) and 10(e), and 9(f) and 10(f) are, respectively, the demosaiced results obtained from Gunturk's, Lu's, Li's, HPHD-LI, and HPHD-AI methods in the interpolation step. From visual comparison, one can see that the Gunturk's, Lu's, and Li's methods induce more color artifacts in edge and textured regions than HPHD-LI or HPHD-AI does. These experimental results validate that the proposed HPHD interpolation method performs satisfactorily not only in textured regions, but also in well-defined edges. Due to space limitations, more discussions and visual comparisons are available online [21].

Further, as can be seen in Figs. 9 and 10, HPHD-LI gives almost the same demosaiced results in edge and textured regions as HPHD-AI does. Hence, HPHD-LI can be used instead of HPHD-AI in practical applications for HPHD-LI not only saves a great amount of computational cost, but also gives comparable visual results as HPHD-AI.

C. Computational Complexity

The calculation performed in reconstructing one color pixel in each stage of the proposed algorithm is listed in Table II, where N and α denote the parameter of window size and spatial classification, respectively. For two directional heterogeneity-projections (H.P.), (12) and (13) require a total of $2N-2$ additions, $2N$ multiplications and 2 absolute conversions for each color pixel. In the directional adaptive filtering (D.A.F) stage, a fixed 1×9 rectangular window was used to compute the local mean and variance by (15) and (16). Thus, the total calculation of (17) and (18) needs 106 additions and 48 multiplications. In the hard-decision interpolation (H.D.I), the total calculation of interpolation with $\alpha=0$ and $\alpha=1$ requires the maximum and minimum computation for each color pixel, respectively. Therefore, if $0<\alpha<1$, the total computational load of interpolation will be between that with $\alpha=1$ and $\alpha=0$.

Note that, for other existent decision-based demosaicing methods, the latter decision stage usually requires much more computation compared with the interpolation stage. Moreover, if the interpolation stage includes a smooth interpolation step, the calculation of decision stage will increase greatly, because it will need to evaluate three interpolation results for each color pixel. In contrast, the calculation of the proposed hard-decision method depends only on the parameter N of window size. The evaluation of horizontal, vertical and smooth interpolations depends on the parameter α and only needs at most 3 compare

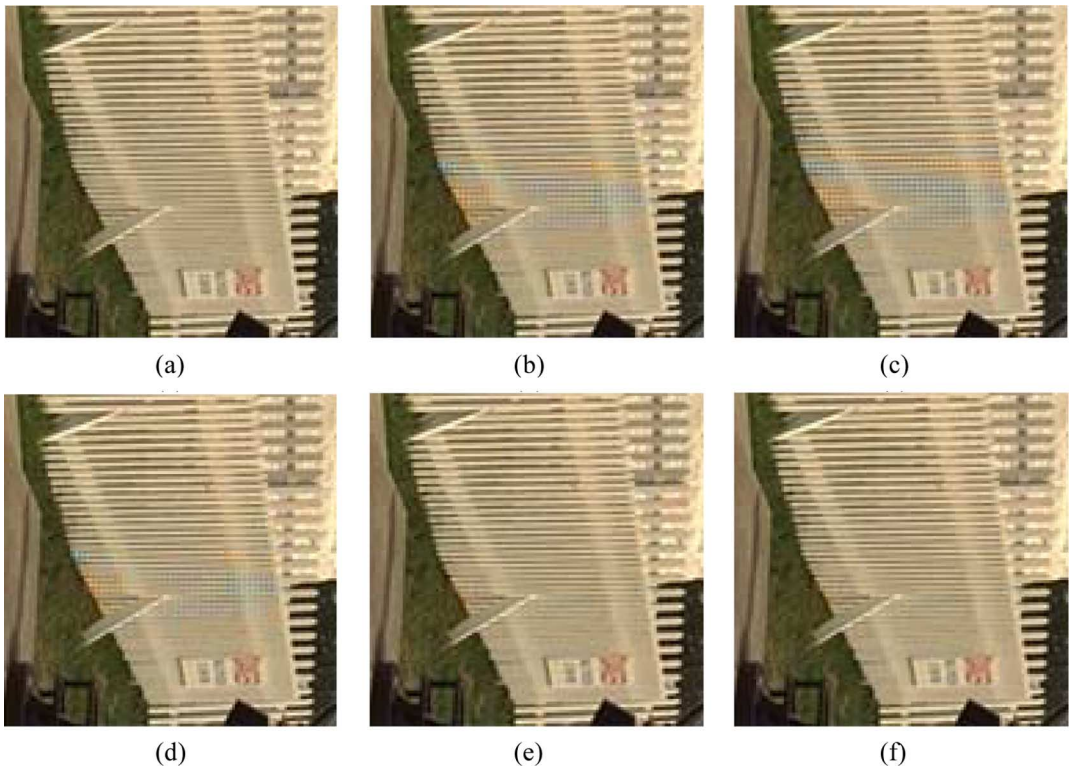

Fig. 10. Zoom-in demosaicing results of test image No. 20. (a) Original picture; Demosaiced result in interpolation step. (b) Gunturk's method. (c) Lu's method. (d) Li's method. (e) HPHD-LI method. (f) HPHD-AI method.

TABLE II
CALCULATIONS PERFORMED FOR RECONSTRUCTING ONE COLOR PIXEL

		HPHD-AI				HPHD-LI			
Stage		Addition	Multiplication	Absolute	Bit-shift	Addition	Multiplication	Absolute	Bit-shift
H.P.		$2N-2$	$2N$	2	0	$2N-2$	$2N$	2	0
D.A.F.		106	48	0	0	106	48	0	0
H.D.I.	$\alpha=1$	50.5	11	18.5	11	10.5	0	0	3.5
	$\alpha=0$	81	16	29	18	17.5	0	0	6
Total	$\alpha=1$	$2N+154.5$	$2N+59$	20.5	11	$2N+114.5$	$2N+48$	2	3.5
	$\alpha=0$	$2N+185$	$2N+64$	31	18	$2N+121.5$	$2N+48$	2	6

operations for each color pixel. Therefore, the proposed method provides an efficient solution for decision-based demosacing.

Note that the software implementations (MATLAB source codes) of the proposed HPHD-AI and HPHD-LI methods along with the 25 test images are also available online [21].

VI. CONCLUSION

A novel hard-decision color interpolation procedure has been developed based on the spectral-spatial correlation of a mosaiced image. The proposed HPHD interpolation method effectively reconstructs fine detail features in both edge and texture regions of demosaiced images. One merit of the proposed algorithm is that it can combine with many existing image interpolation methods such as decision-based algorithm (set $\alpha = 1$), edge-directed interpolation, adaptive interpolation, linear interpolation, etc., to obtain improved performance. Moreover, the proposed heterogeneity-projection scheme provides an efficient method for decision-based algorithms to make accurate direction-selection before performing interpolation. The performance of HPHD method has been compared with three renowned demosaicing methods. Experimental results show that HPHD method not only outperforms all of them in PSNR (in decibels) and S-CIELAB ΔE^*_{ab} measures, but also gives superior demosaiced fidelities in visual comparison.

APPENDIX

Parameter Tuning of N and α : Since the value of parameters (N, α) may drastically influence demosaicing performance, and, hence, the comparison results, it is interesting to study how they affect the demosaicing results of the proposed method. In order to evaluate the demosaicing performance, we first define the following criterion

$$\mathrm{PSNR}_{\mathrm{Avg}}(N,\alpha) = \frac{1}{25}\sum_{i=1}^{25}\mathrm{PSNR}(O_i, D_i(N,\alpha)) \qquad (30)$$

where O_i and D_i indicate the i th test image and its corresponding demosaiced one by using the proposed HPHD-AI method. PSNR (in decibels) denotes the metric of peak signal-to-noise ratio defined in (29). Based on the criterion

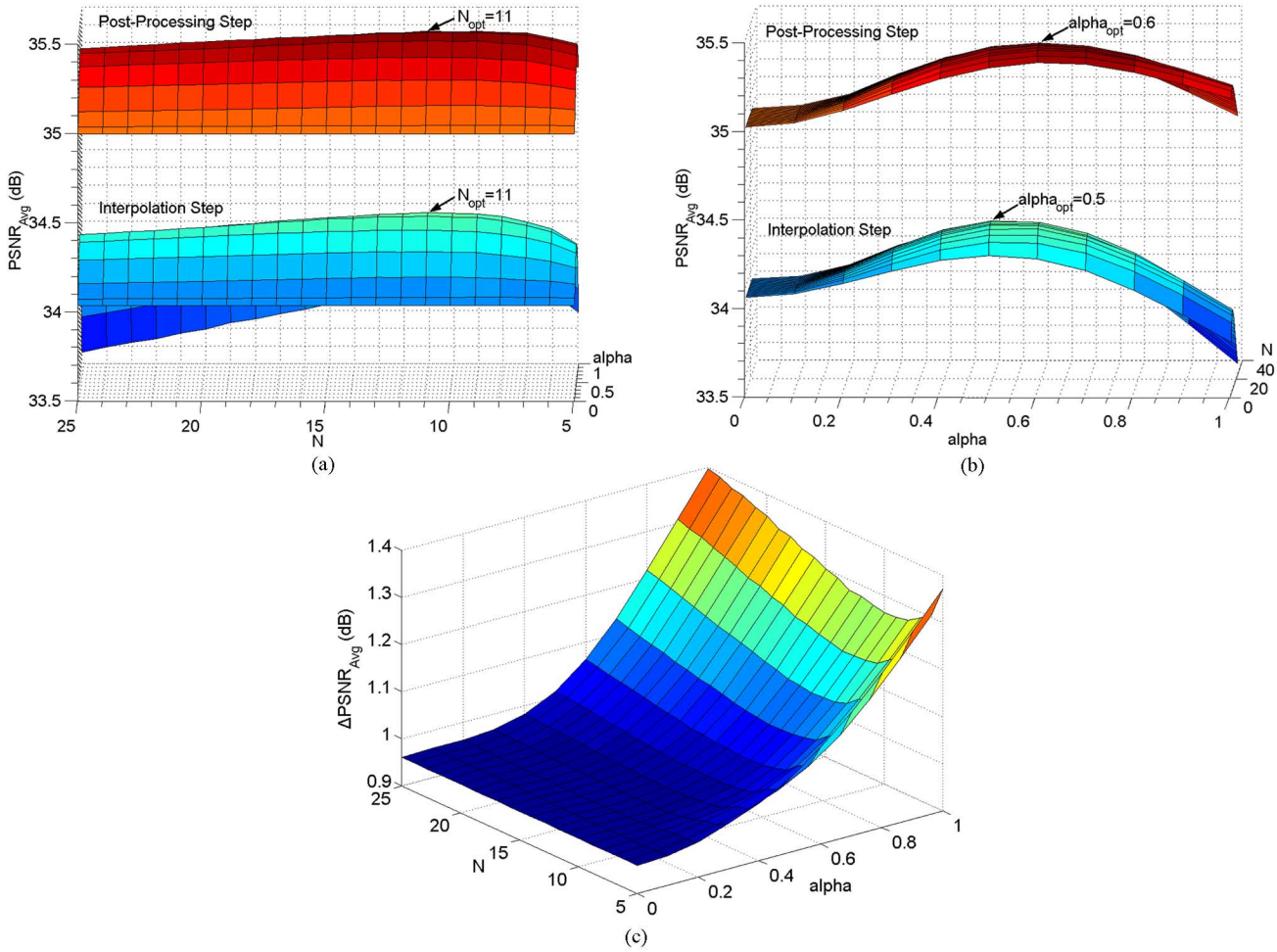

Fig. 11. Experimental results of tuning parameters in each step. (a) Evolution of $\mathrm{PSNR}_{\mathrm{Avg}}$ as the parameter N increases. (b) Evolution of N as the parameter α increases. (c) Influence of the parameters (N, α) on the performance gap $\Delta\mathrm{PSNR}_{\mathrm{Avg}}$ between postprocessing and interpolation steps.

(30), the parameter N is tweaked from 5 to 25 with interval 1, and α is tweaked from 0 to 1 with interval 0.1. Fig. 11 shows the experimental results of tweaking parameters (N, α). Fig. 11(a) and (b), respectively, represents the evolution of $\mathrm{PSNR}_{\mathrm{Avg}}$ as parameter N and α increase. In Fig. 11(a), one can see that when $\alpha = 0$ (only the smooth set under consideration), the $\mathrm{PSNR}_{\mathrm{Avg}}$ is independent from the parameter N. On the other hand, when $\alpha = 1$ (only the horizontal and vertical sets under consideration), the impact of N on $\mathrm{PSNR}_{\mathrm{Avg}}$ increases. Thus, the influence of N on $\mathrm{PSNR}_{\mathrm{Avg}}$ depends on the parameter α, especially when $\alpha = 1$. Moreover, one can see in Fig. 11(a) that the local optimal parameter N occurs at $N_{\mathrm{opt}} = 11$ in the experiment.

Fig. 11(b) shows that the parameter α has significant influence on the $\mathrm{PSNR}_{\mathrm{Avg}}$. If parameter α increases from 0 to 0.6, the $\mathrm{PSNR}_{\mathrm{Avg}}$ also increases. However, when parameter α increases from 0.6 to 1, the criterion $\mathrm{PSNR}_{\mathrm{Avg}}$ becomes decreasing. This implies the local optimal parameter α should occur in the range from 0.5 to 0.6, and the optimal interpolation result will encompass horizontal, vertical and smooth interpolations together. Since parameter $\alpha = 0.6$ obtains the maximum $\mathrm{PSNR}_{\mathrm{Avg}}$ in postprocessing step, we choose $\alpha_{\mathrm{opt}} = 0.6$ as the local optimal parameter α.

Fig. 11(c) shows the influence of the parameters (N, α) on the performance gap $\Delta\mathrm{PSNR}_{\mathrm{Avg}}$ between postprocessing and interpolation steps. It is clear that the performance gap mostly depends on the parameter α. Moreover, the maximum performance gap occurs when parameter $\alpha = 1$. This implies that the postprocessing provides significant improvement on the horizontal and vertical interpolation results. Therefore, postprocessing seems to be more beneficial to the existent soft-decision demosaicing algorithms, which only considers the horizontal and vertical interpolations.

Summarizing the tweaking parameter experiment, we have the following findings.

1) For the proposed method, the parameter α has significant influence on the demosaicing performance compared with parameter N.
2) When the interpolation only considers horizontal and vertical ones, the postprocessing provides significant improvement on the interpolation result.
3) The optimal interpolation result requires encompassing horizontal, vertical and smooth interpolations together.
4) Based on the criterion (30), the local optimal parameters $(N_{\mathrm{opt}}, \alpha_{\mathrm{opt}})$ of proposed HPHD-AI method can be found at (11,0.6).

ACKNOWLEDGMENT

The authors would like to thank Dr. B. K. Gunturk, Louisiana State University; Dr. Y.-P.Tan, Nanyang Technological University, Singapore; Dr. X. Li, West Virginia University, Morgantown; and I. Omer of Hebrew University, Israel, for providing their CFA demosaicing programs.

REFERENCES

[1] D. R. Cok, "Reconstruction of CCD images using template matching," in *Proc. IS&T Annu. Conf. ICPS*, 1994, pp. 380–385.

[2] 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.

[3] W.-M. 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.

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

[5] 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, Sep. 2002.

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

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

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

[9] L. D. Grossmann and Y. C. Eldar, "Enhancement of color images by efficient demosaicing" [Online]. Available: http://www.ee.technion.ac.il/Sites/People/YoninaEldar/Download/GY2004.pdf

[10] I. Omer and M. Werman, "Using natural image properties as demosaicing hints," in *Proc. IEEE Int. Conf. Image Processing, Singapore*, 2004, pp. 1665–1670.

[11] B. Bayer, "Color imaging array," U.S. Patent 3 971 065, 1976.

[12] 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.

[13] L.-L. Chang and Y.-P. Tan, "Effective use of spatial and spectral correlations for color filter array demosaiking," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 355–365, Feb. 2004.

[14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[15] H. Stark and Y. Yang, *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics*. New York: Wiley, 1998.

[16] P. S. Maybeck, *Stochastic Models, Estimation, and Control Volume 1*. New York: Academic, 1979.

[17] R. Kimmel, "Demosaicking: Image reconstruction from color CCD samples," *IEEE Trans. Image Process.*, vol. 8, no. 12, pp. 1221–1228, Dec. 1999.

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

[19] M. Mahy, E. V. Eyckden, and O. Oosterlinck, "Evaluation of uniform color spaces developed after the adoption of CIELAB and CIELUV," *Color Res. Appl.*, vol. 19, no. 2, pp. 105–121, 1994.

[20] J. F. Hamilton and J. E. Adams Jr, "Adaptive color plane interpolation in single sensor color electronic camera," U.S. Patent 5 629 734, 1997.

[21] Appendix document website [Online]. Available: http://isci.cn.nctu.edu.tw/video/Demo/

Chi-Yi Tsai was born in Kaohsiung, Taiwan, R.O.C., in 1978. He received the B.S. and M.S. degree in electrical engineering from the National Yunlin Technology University, Yunlin, Taiwan, in 2000 and 2002, respectively. He is currently pursuing the Ph.D. degree in electrical and control engineering at the National Chiao-Tung University, Hsinchu, Taiwan.

His research interests include image processing, visual tracking control of the mobile robot, visual servoing, and computer vision.

Kai-Tai Song (A'90) was born in Taipei, Taiwan, R.O.C., in 1957. He received the B.S. degree in power mechanical engineering from the National Tsing Hua University, Taiwan, in 1979, and the Ph.D. degree in mechanical engineering from the Katholieke Universiteit Leuven, Leuven, Belgium, in 1989.

He was with the Chung Shan Institute of Science and Technology from 1981 to 1984. Since 1989, he has been on the faculty and is currently a Professor in the Department of Electrical and Control Engineering, National Chiao-Tung University, Hsinchu, Taiwan. His areas of research interest include mobile robots, image processing, visual tracking, sensing and perception, embedded systems, intelligent system control integration, and mechatronics.

Dr. Song served as the Chairman of the IEEE Robotics and Automation Society, Taipei Chapter, from 1998 to 1999.