Demosaicing: Heterogeneity-Projection Hard-Decision Adaptive Interpolation Using Spectral-Spatial Correlation

Chi-Yi Tsai and Kai-Tai Song
Department of Electrical and Control Engineering, National Chiao Tung University
1001 Ta Hsueh Road, Hsinchu 300, Taiwan
E-mail: chiyi.ece91g@nctu.edu.tw; ktsong@mail.nctu.edu.tw

ABSTRACT

A novel heterogeneity-projection hard-decision adaptive interpolation (HPHD-AI) algorithm is proposed in this paper for color reproduction from Bayer mosaic images. The proposed algorithm aims to estimate the optimal interpolation direction and perform hard-decision interpolation, in which the decision is made before interpolation. To do so, a new heterogeneity-projection scheme based on spectral-spatial correlation is proposed to decide the best interpolation direction from the original mosaic image directly. Exploiting the proposed heterogeneity-projection scheme, a hard-decision rule can be designed easily to perform the interpolation. We have compared this technique with three recently proposed demosaicing techniques: Lu's, Gunturk's and Li's methods, by utilizing twenty-five natural images from Kodak PhotoCD. The experimental results show that HPHD-AI outperforms all of them in both PSNR values and S-CIELab ΔE^*_{ab} measures.

Keywords: Color reproduction, CFA demosaicing, color artifacts, adaptive filtering, digital cameras.

1. INTRODUCTION

Digital color images from single-chip digital still cameras are obtained by interpolating the output from a color filter array (CFA), in which each sensor pixel only samples one of three primary color components. These sparsely sampled color values are termed mosaic images. A full-color image is reproduced from a mosaic image by estimating two missing color values for each pixel. This image reconstruction process is commonly known as CFA interpolation or CFA demosaicing. The simplest CFA demosaicing methods apply well-known interpolation techniques to each color channel separately such as bilinear interpolation and cubic spline interpolation. However, these single-channel algorithms usually introduce severe color artifacts and blurs around sharp edges [1]. These drawbacks motivate the need of more specialized algorithms for advanced demosaicing performance. An excellent literature survey on advanced demosaicing algorithms can be found in [2].

In recent years, there have been researches on more sophisticated demosaicing algorithms. In [3], Lu and Tan presented an improved hybrid CFA demosaicing method that consists of interpolation and post-processing steps to render full-color images and suppress visible demosaicing artifacts. In [4], the authors utilized a projection-onto-convex-set (POCS) technique to estimate the missing color values in red and blue channels using alternating projection scheme based on high inter-channel correlation. In [5], Li proposed a successive approximation demosaicing strategy by adopting color difference interpolation iteratively. Another recent demosaicing approach, termed as *decision-based demosaicing algorithm*, divides the demosaicing procedure into interpolation step and decision step [6, 7]. In the interpolation step, they produce respectively horizontally interpolated and vertically interpolated images. In decision step, a soft decision method was adopted for choosing the pixels interpolated in the direction with fewer artifacts. For the decision step, Hirakawa *et al* proposed the color image homogeneity metric to measure the level of misguidance color artifacts presented in these two images [6]. Based on this measurement, the interpolation decision is made by choosing the region with larger homogeneity map values. In [7], Wu *et al* adopted the Fisher's linear discriminant technique to determine the optimal interpolation direction under two hypotheses, one for horizontal structure and the other for vertical structure, in a local window. 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 decision-based demosaicing algorithms is that they are not efficient in the interpolation step because each pixel has to interpolate twice,

Digital Photography II, edited by Nitin Sampat, Jeffrey M. DiCarlo, Russel A. Martin,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6069, 606906, © 2006 SPIE-IS&T · 0277-786X/06/$15

one in horizontal direction and the other in vertical direction, before applying the soft decision method. Therefore, to develop an efficient color interpolation algorithm with high performance in both textured and edge regions is still a challenge in CFA demosaicing research.

In this paper, a novel heterogeneity-projection hard-decision adaptive interpolation (HPHD-AI) method is proposed for color reproduction from Bayer mosaic images. The proposed algorithm aims to decide the optimal interpolation direction before performing interpolation. To do so, a new heterogeneity-projection scheme based on spectral-spatial correlation is proposed to estimate the best interpolation direction from the original Bayer mosaic images directly. Based on the proposed heterogeneity-projection scheme, a hard-decision rule can be designed easily to perform the interpolation. The advantage of the proposed demosaicing algorithm is threefold. First, the proposed heterogeneity-projection scheme can be combined with existent decision-based demosaicing algorithms. Second, the decision is made before interpolation and thus each pixel only has to interpolate once in the interpolation step. Finally, the proposed demosaicing algorithm also performs well not only in textured regions, but also in well-defined edges of the image.

2. SPECTRAL-SPATIAL CORRELATION

Fig. 1 shows the most used CFA pattern, the Bayer pattern [8], where R, G and B denote, respectively, the pixels having only red, green and blue color values. We limit our discussion in this paper to the Bayer pattern because it is popular. In the following, we will introduce a novel spectral-spatial correlation based on two popular image correlations: spectral and spatial correlations.

Many existent demosaicing methods are developed using image spectral or spatial correlation, or both. The concept of spectral correlation is based on the assumption that the color difference signals are locally constant in chrominance smooth areas [9]. The spatial correlation refers to the fact that within a homogeneous image region, neighboring pixels share similar color values [3, 10]. In other words, the difference between neighboring pixel values along an edge direction in spatial domain is a constant. Spectral and spatial correlations of a natural image describe the relationship between different color channels. However, in Bayer mosaic image, it is difficult to calculate the spectral and spatial correlations directly because each pixel only contains one primary component. This problem motivates us to find a more efficient criterion instead of spectral and spatial correlations for Bayer mosaic images.

A significant characteristic of Bayer pattern is that for each pixel, the surrounding pixels are one of the primary components in different channels. This causes us 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

$$R(x,y) - G(x+1,y) = \left[R(x,y) - \overline{G}(x,y)\right] + \left[\overline{G}(x,y) - G(x+1,y)\right], \tag{1}$$

where $\overline{G}(x,y)$ denotes the missing green value at center red pixel location. By the assumption of spectral and spatial correlations, expression (1) becomes such that

$$S_{rg}^{h(x,x+1)} \equiv R(x,y) - G(x+1,y) = A_{rg}(x,y) + dG_h . \tag{2}$$

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

$$S_{bg}^{h(x,x+1)} \equiv B(x,y) - G(x+1,y) = A_{bg}(x,y) + dG_h . \tag{3}$$

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

$$S_{rg}^{v(y,y+1)} \equiv R(x,y) - G(x,y+1) = A_{rg}(x,y) + dG_v \text{, and}$$

$$S_{bg}^{v(y,y+1)} \equiv B(x,y) - G(x,y+1) = A_{bg}(x,y) + dG_v . \tag{4}$$

where $A_{rg}(x,y)$ and $A_{bg}(x,y)$ are piecewise constant functions; dG_h and dG_v are constants. Expressions (1)-(4) tell us that the difference between surrounding pixels in different color channels is equal to the summation of spectral and spatial correlations. We refer these relationships (1)-(4) as *spectral-spatial correlations* (SSC). SSC has two important characteristics. First, SSC can be easily and directly calculated from Bayer mosaic images. 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. SSC acts as a significant clue for us to find the directional smooth regions in Bayer mosaic images directly before performing the interpolation. In the following section, we will present the proposed heterogeneity-projection based on these observations.

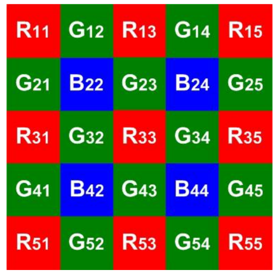


Fig. 1: Bayer color filter array pattern (Bayer pattern)

3. HETEROGENEITY-PROJECTION FOR BAYER MOSAIC IMAGES

The aim of this section is to derive the heterogeneity-projection formulation based on SSC. The proposed heterogeneity-projection scheme can directly transfer the original Bayer mosaic image into horizontal and vertical heterogeneity maps. Using these two heterogeneity maps, the decision of interpolation direction can be determined easily by choosing the smallest heterogeneity values.

A. Heterogeneity-Projection

Because SSC is piecewise constant along an edge direction, the nth-order directional finite derivative of SSC along the edge direction tends toward a small value. For example, let's consider the interpolation of R_{33} in Fig. 1. Suppose that the pixel R_{33} is located on a horizontal edge. The SSC values of pixel R_{33} and its neighboring pixels along horizontal direction can be found such that

$$S_{rg}^{h(1,2)} = A_{rg}(1,3) + dG_h\,,\quad S_{rg}^{h(3,4)} = A_{rg}(3,3) + dG_h\,,$$
$$S_{gr}^{h(2,3)} = -A_{rg}(2,3) + dG_h\,,\quad S_{gr}^{h(4,5)} = -A_{rg}(4,3) + dG_h\,, \tag{5}$$

where $S_{gr}^{h(x,x+1)} \equiv G(x,y) - R(x+1,y)$. Define the first-order horizontal finite derivative of SSC such that

$$dS_{rg}^{h(1,4)} \equiv S_{rg}^{h(1,2)} - S_{rg}^{h(3,4)} = A_{rg}(1,3) - A_{rg}(3,3)\text{, and}$$
$$dS_{gr}^{h(2,5)} \equiv S_{gr}^{h(2,3)} - S_{gr}^{h(4,5)} = A_{rg}(4,3) - A_{rg}(2,3)\,. \tag{6}$$

Because $A_{rg}(x,y)$ is piecewise constant function, $dS_{rg}^{h(1,4)}$ and $dS_{rg}^{h(2,5)}$ both will approach to zero along this horizontal edge. Consequently, the second-order horizontal finite derivative of SSC

$$d^2 S_{rg}^{h(1,5)} \equiv dS_{rg}^{h(1,4)} - dS_{gr}^{h(2,5)} = A_{rg}(1,3) + A_{rg}(2,3) - A_{rg}(3,3) - A_{rg}(4,3)$$

will also tend toward zero along the horizontal edge. This observation poses a question that how the nth-order directional finite derivative of SSC can be directly calculated from a Bayer mosaic image. To resolve this problem, a heterogeneity-projection scheme has been developed to transfer row data of a Bayer mosaic image into nth-order directional finite derivative of SSC directly. Note that we refer the value of nth-order directional finite derivative of SSC as *heterogeneity value* because it leads to a small value within a directional smooth region.

Denote $RG_{1\times N} = [R_1 \quad G_2 \quad R_3 \quad \cdots]_{1\times N}$ as row data of a 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 have 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

$$[dS_{rg}^{h(1,4)} \quad dS_{gr}^{h(2,5)} \quad dS_{rg}^{h(3,6)} \quad \cdots]_{1\times(N-3)} = RG_{1\times N} T_{N\times(N-3)}^{1}\,, \tag{7}$$

where $T_{N\times(N-3)}^{1} = \begin{bmatrix}1 & -1 & -1 & 1\end{bmatrix}^T \otimes eye(N-3)$, $\otimes$ denotes the 2D convolution operator and $eye(M)$ denotes a $M\times M$ identity matrix. Second, the horizontal heterogeneity value H_h can be calculated using Euclidean inner product [11]

$$H_h = [dS_{rg}^{h(1,4)} \quad dS_{gr}^{h(2,5)} \quad dS_{rg}^{h(3,6)} \quad \cdots]_{1\times(N-3)} T_{(N-3)\times 1}^{2}\,, \tag{8}$$

where $T_{(N-3)\times 1}^{2} = \prod_{i=1}^{N-4} \begin{bmatrix}1 & -1\end{bmatrix}^T \otimes eye(N-3-i)$ is a $(N-3)\times 1$ vector. Next, substituting (7) into (8) yields

$$H_h = RG_{1\times N} T^1_{N\times(N-3)} T^2_{(N-3)\times 1} = RG_{1\times N} P_{N\times 1}\,, \tag{9}$$

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 (9) 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 expression (9) is termed as *horizontal heterogeneity-projection* of Bayer mosaic image's row data. 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

$$H_v = RG^T_{N\times 1} P_{N\times 1}\,, \tag{10}$$

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 (9) and (10), the horizontal and vertical heterogeneity maps, H_{h_map} and H_{v_map} can be obtained, respectively by

$$H_{h\_map} = \left|Bayer \otimes P^T_{N\times 1}\right|, \text{ and } H_{v\_map} = \left|Bayer \otimes P_{N\times 1}\right|, \tag{11}$$

where $Bayer$ denotes the original Bayer mosaic image. We see from (11) 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

The directional heterogeneity-projection along an edge direction leads to a small heterogeneity value; however, it may also obtain a small heterogeneity value when the directional heterogeneity-projection performs along a wrong edge direction. This problem will cause a wrong decision in the interpolation step. In order to overcome this problem, a directional adaptive filter whose behavior changes based on statistical characteristics of the image inside a local window is designed to reduce the estimation error in horizontal and vertical heterogeneity maps.

The proposed directional adaptive filter is divided into horizontal and vertical adaptive filters. For horizontal heterogeneity map, only the horizontal adaptive filter is applied to it without the vertical one. The concept of directional adaptive filter is to perform adaptive filtering based on statistical measures of surrounding pixels along one direction. The simplest statistical measures are the mean and variance in a local window [12]. For instance, consider the horizontal adaptive filtering of a pixel H_h on the horizontal heterogeneity map; the adaptively filtered pixel H_h^* is obtained by

$$H_h^* = \overline{H}_h^L + \frac{\delta H_h^L}{\delta H_h^L + \delta H_h^R}(\overline{H}_h^R - \overline{H}_h^L)\,. \tag{12}$$

where H_h^L and H_h^R, respectively, denote the left and right neighboring pixels of H_h; $(\overline{H}_h^L, \delta H_h^L)$ and $(\overline{H}_h^R, \delta H_h^R)$ are the local mean and variance of H_h^L and H_h^R, respectively. Similarly, the vertical adaptive filter for the pixel H_v on the vertical heterogeneity map is given by

$$H_v^* = \overline{H}_v^U + \frac{\delta H_v^U}{\delta H_v^U + \delta H_v^D}(\overline{H}_v^D - \overline{H}_v^U)\,, \tag{13}$$

where H_h^U and H_h^D, respectively, denote the up and down neighboring pixels of H_h; $(\overline{H}_v^U, \delta H_v^U)$ and $(\overline{H}_v^D, \delta H_v^D)$ are the local mean and variance of H_v^U and H_v^D, respectively. After adopting, respectively the horizontal and vertical adaptive filters presented above into horizontal and vertical heterogeneity maps, the filtered horizontal and vertical heterogeneity maps $H^*_{h\_map}$ and $H^*_{v\_map}$ are obtained.

4. HARD-DECISION ADAPTIVE INTERPOLATION

When the horizontal and vertical heterogeneity maps are obtained, a hard-decision rule is employed for color interpolation. First, we define three subsets in the image such that

$$\begin{aligned}
\Omega_h &\equiv \left\{(x,y) \mid H^*_{h_map}(x,y) < \alpha H^*_{v_map}(x,y)\right\}, \\
\Omega_v &\equiv \left\{(x,y) \mid H^*_{v_map}(x,y) < \alpha H^*_{h_map}(x,y)\right\}, \\
\Omega_s &\equiv \left\{(x,y) \mid (x,y) \notin \Omega_h, (x,y) \notin \Omega_v\right\},
\end{aligned} \tag{14}$$

where Ω_h, Ω_v, and Ω_s denote the horizontal, vertical, and smooth subsets, respectively. α is a positive constant satisfying $0 \le \alpha \le 1$. The parameter α in (14) controls the size of a 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. On the contrary, for $\alpha = 1$, the image only contains horizontal and vertical subsets but without smooth subset.

Second, based on (14), the concept of hard-decision rule for interpolation is obtained

$$\begin{aligned} & \textit{if } (x,y) \in \Omega_h \\ & \quad \text{Perform } \textit{horizontal} \text{ interpolation on each missing color channel.} \\ & \textit{elseif } (x,y) \in \Omega_v \\ & \quad \text{Perform } \textit{vertical} \text{ interpolation on each missing color channel.} \\ & \textit{else} \\ & \quad \text{Perform } \textit{weight averaging} \text{ of neighboring pixels on each missing color channel.} \end{aligned} \tag{15}$$

The color interpolation method is performed based on the hard-decision rule (15). 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 the following expression

$$G_{miss} = \frac{e_{up}\hat{G}_{up} + e_{right}\hat{G}_{right} + e_{down}\hat{G}_{down} + e_{left}\hat{G}_{left}}{e_{up} + e_{right} + e_{down} + e_{left}}, \tag{16}$$

where $\hat{G}_{\{up,right,down,left\}}$ denote the color-adjusted green values of four surrounding green pixels, and $e_{\{up,right,down,left\}}$ denote the corresponding edge indicators. However, in our method, the following modification on edge indicators is adopted according to the hard-decision rule (15) such that

$$\begin{aligned} & \textit{if } (x,y) \in \Omega_h \\ & \quad (e_{up}, e_{down}) = (0,0). \\ & \textit{elseif } (x,y) \in \Omega_v \\ & \quad (e_{right}, e_{left}) = (0,0). \end{aligned} \tag{17}$$

In other words, the hard-decision *adaptive* interpolation for green channel is summarized as follows

$$G_{miss} = \begin{cases} \dfrac{e_{right}\hat{G}_{right} + e_{left}\hat{G}_{left}}{e_{right} + e_{left}}, & \textit{if } (x,y) \in \Omega_h \\ \dfrac{e_{up}\hat{G}_{up} + e_{down}\hat{G}_{down}}{e_{up} + e_{down}}, & \textit{if } (x,y) \in \Omega_v \\ \dfrac{e_{up}\hat{G}_{up} + e_{right}\hat{G}_{right} + e_{down}\hat{G}_{down} + e_{left}\hat{G}_{left}}{e_{up} + e_{right} + e_{down} + e_{left}}, & \textit{if } (x,y) \in \Omega_s \end{cases} \tag{18}$$

Note that the formulation of each surrounding color-adjusted green value in (18) adopts the approach proposed in [3] while the corresponding edge indicator can be referred from among the references [3], [10], and [13]. Hereafter, the color-adjusted value of each color pixel and the corresponding edge-indicator are determined as in [3].

When the green channel has been fully recovered, it can be used to assist the interpolation of red and blue channels. The interpolation procedure for red and blue channels consists of two sub-steps: 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 (15) to the sub-step 2) because there is not enough information to perform horizontal and vertical interpolations in sub-step 1). Since the same procedure is utilized to interpolate the red and blue channels, only the red channel interpolation will be presented.

Let R^b_{miss} denote a missing red value at a blue pixel. It is estimated from its four neighboring red pixels by the following formulation

$$R^b_{miss} = \frac{e_{up-right}\hat{R}_{up-right} + e_{down-right}\hat{R}_{down-right} + e_{down-left}\hat{R}_{down-left} + e_{up-left}\hat{R}_{up-left}}{e_{up-right} + e_{down-right} + e_{down-left} + e_{up-left}}, \tag{19}$$

where $\hat{R}_{\{up-right,down-right,down-left,up-left\}}$ denote the color-adjusted red values of four neighboring red pixels, and $e_{\{up-right,down-right,down-left,up-left\}}$ denote the corresponding edge indicators. Subsequently, the rest of the missing red values at green pixels will be proceeded. As the same procedure is performed in green channel, each missing red value at a green pixel R^g_{miss} can be estimated from its four surrounding red pixels by the following hard-decision adaptive interpolation

$$R^g_{miss} = \begin{cases} \dfrac{e_{right}\hat{R}_{right} + e_{left}\hat{R}_{left}}{e_{right} + e_{left}}, \ if \ \ (x,y) \in \Omega_h \\ \dfrac{e_{up}\hat{R}_{up} + e_{down}\hat{R}_{down}}{e_{up} + e_{down}}, \ if \ \ (x,y) \in \Omega_v \\ \dfrac{e_{up}\hat{R}_{up} + e_{right}\hat{R}_{right} + e_{down}\hat{R}_{down} + e_{left}\hat{R}_{left}}{e_{up} + e_{right} + e_{down} + e_{left}}, \ if \ \ (x,y) \in \Omega_s \end{cases} \tag{20}$$

where $\hat{R}_{\{up,right,down,left\}}$ denote the color-adjusted red values of four surrounding red pixels, and $e_{\{up,right,down,left\}}$ are the corresponding edge indicators. Finally, a full-color image can be obtained after applying the same interpolation processes described above on each missing blue value.

5. COMPARATIVE STUDY ON EXPERIMENTAL RESULTS

Fig. 2 shows twenty-five Kodak photographic images employed in the experiments for demonstrating the demosaicing performance. According to [14], the CFA operations in digital camera pipeline usually introduce a demosaiced image post-processing framework to provide more pleasing color output. Therefore, the experiments also apply a post-processing framework to complete the comparisons. Fig. 3 illustrates the flowchart of the experiment, which contains interpolation and post-processing steps. In interpolation step, the demosaiced results of the proposed HPHD adaptive interpolation (HPHD-AI) method are compared with those using three recent published methods: Lu's [3], Gunturk's [4], and Li's [5] methods. For Gunturk's method, we make use of one-level (1-L) decomposition with eight projection iterations in the experiment. For Li's method, the universal threshold value ($\delta_l = \delta_h = 4$), suggested threshold value ($\delta_l = 4, \delta_h = 0.05$), and maximum iteration number $iter = 20$ are chosen in the experiment. For the proposed method, the presetting window size and positive constant are chosen as $N = 9$ and $\alpha = 0.8$, respectively. All test images are down-sampled to obtain the Bayer pattern (as shown in Fig. 1) and then reconstructed using the demosaicing methods under comparison in RGB color space.

To evaluate the quality of the demosaiced images, two performance measures are adopted in the experiments: PSNR metric and S-CIELab ΔE^*_{ab} metric [3, 15]. The PSNR (in dB) metric in this paper is defined as

$$PSNR(dB) = 10\log_{10}\left\{255^2\left(\frac{1}{MN}\sum_{1\le v\le M}\sum_{1\le u\le N}\left\|\overline{O}(u,v) - \overline{D}(u,v)\right\|^2\right)^{-1}\right\}, \tag{21}$$

where M , N are the total column and row number of the image; $\overline{O}(u,v)$ is the color vector at the (u,v) th position of the original color image; $\overline{D}(u,v)$ is the corresponding color vector in the demosaiced color image. Note that, for a demosaiced image, high fidelity implies high PSNR and small S-CIELab ΔE^*_{ab} measures.

A. Quantitative Comparison Using PSNR and S-CIELab Measures

Table I records the PSNR values and S-CIELab ΔE^*_{ab} measures of the demosaiced results obtained by the proposed interpolation method together with these by other methods for comparison. The bold-type font denotes the highest PSNR and smallest ΔE^*_{ab} values across each row. From Table I, it can be seen that HPHD-AI method generates improved demosaiced fidelity in most of the test images in the interpolation step. Moreover, in the post-processing step, HPHD-AI method not only has significant improvement, but also obtains the best demosaiced results in most of the test images compared with other methods.

B. Visual Comparison

The demosaiced results shown in Figs. 4-5 evaluate the performance of the proposed HPHD-AI method in edge regions and fine textures. Figs. 4(a)-5(a) show the zoom-in of the test image No. 16 and 20, respectively. These scenes contain many long edges and fine detail regions such as fine fiber patterns (Fig. 4) and picket fences (Fig. 5). These features can effectively challenge the performance of demosaicing methods. Figs. 4(b)-5(b) and 4(c)-5(c) are,

Fig. 2: Test images used in the experiment.

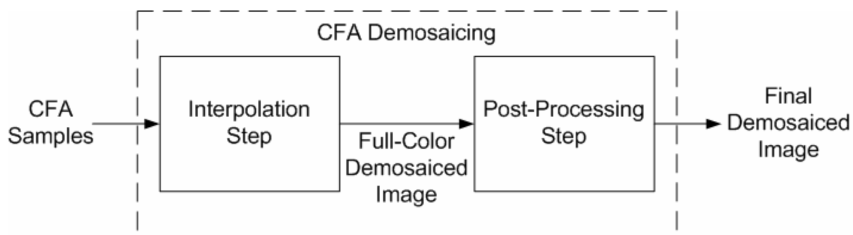


Fig. 3: Flowchart of the experiment. In the interpolation step, we compare the performance of Lu's, Gunturk's, Li's and proposed HPHD adaptive interpolation (HPHD-AI) methods. In post-processing step, Lu's post-processing method is adopted into each demosaicing method.

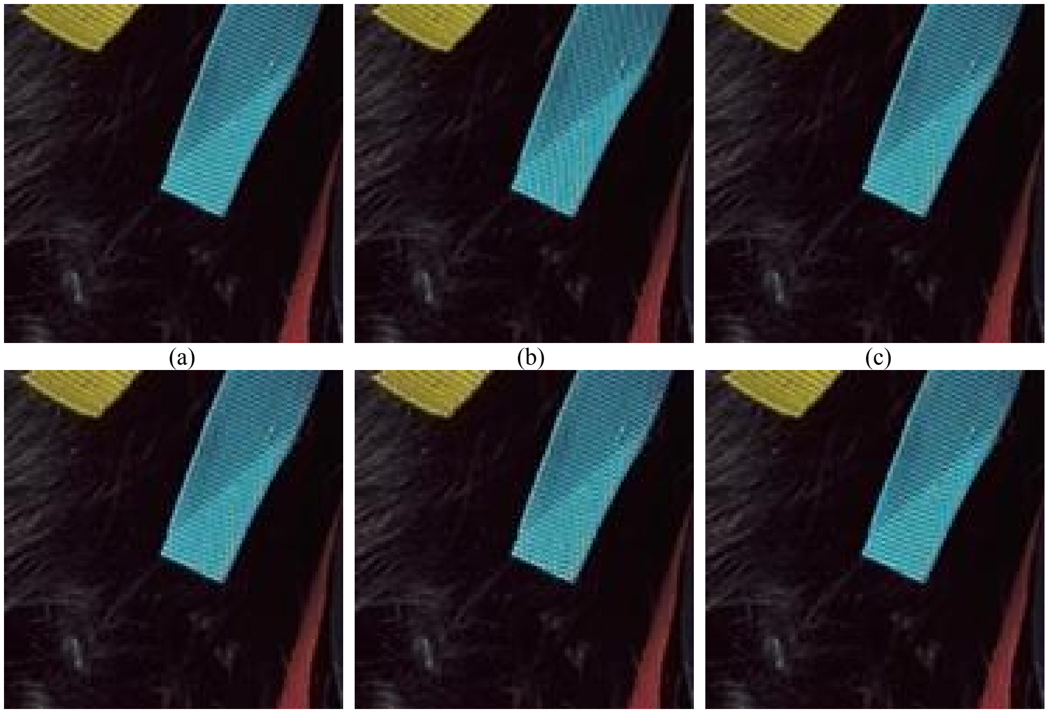
respectively, the demosaiced results obtained from Lu's and Gunturk's methods. Figs. 4(d)-5(d) and 4(e)-5(e) are the demosaiced results obtained from Li's method with the universal threshold value (UTV) and suggested threshold value (STV), respectively. Figs. 4(f)-5(f) are the demosaiced results obtained from HPHD-AI methods. From visual comparison, we observe that the Lu's, Gunturk's and Li's methods induce more color artifacts in edge and textured regions than HPHD-AI do. Therefore, these experimental results validate that proposed HPHD-AI method performs satisfactorily not only in textured regions, but also in well-defined edges of the image.

6. CONCLUSIONS

A novel heterogeneity-projection hard-decision adaptive interpolation (HPHD-AI) algorithm has been developed based on spectral-spatial correlation. The proposed HPHD-AI method effectively reconstructs the fine detail features in both edge and texture regions of demosaiced images. One merit of the proposed algorithm is that it can be combined with many existing image interpolation methods such as decision-based algorithm (set $\alpha = 1$), edge-directed interpolation, adaptive interpolation, linear interpolation, etc. The performance of HPHD-AI method has been compared with three recent published demosaicing methods. Experimental results show that HPHD-AI method not only outperforms all of them in PSNR (dB) and ΔE^*_{ab} measures, but also gives superior demosaiced fidelities in visual comparison with other methods.

Table I: Performance comparison among recent proposed methods:
PSNR (dB) and S-CIELab ΔE^*_{ab} measures of demosaiced images in the interpolation and post-processing steps.

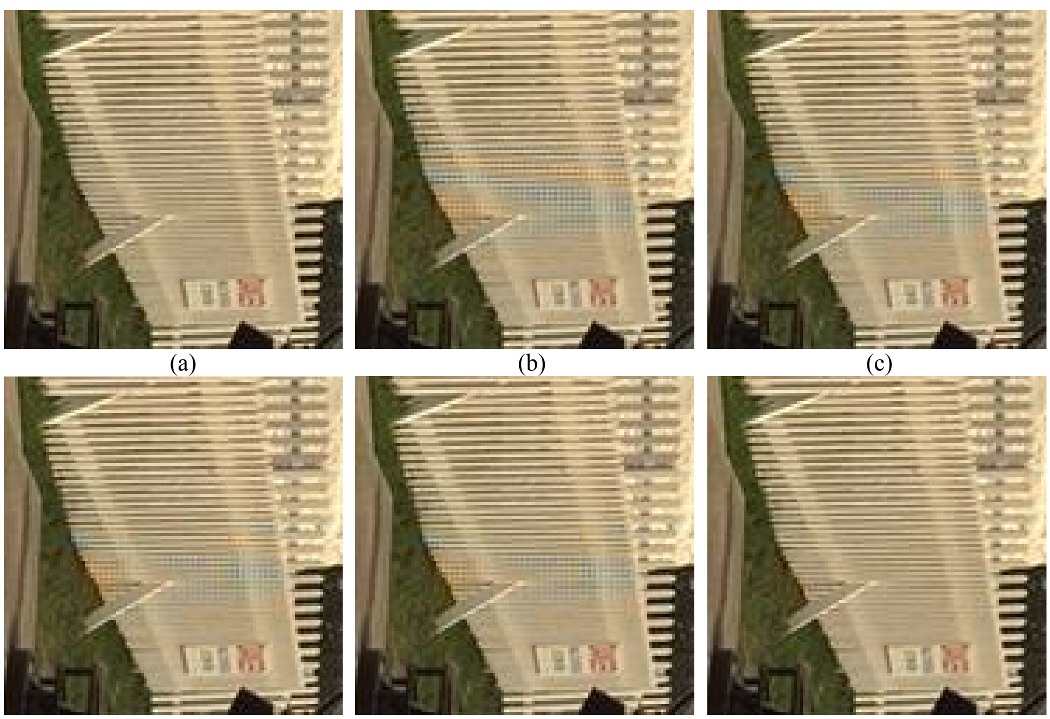
Step	Interpolation Step					Post-Processing Step				
Method	Lu [3]	Gunturk [4]	Li with UTV [5]	Li with STV [5]	HPHD-AI	Lu [3]	Gunturk [4]	Li with UTV [5]	Li with STV [5]	HPHD-AI
1	31.0257	29.3765	28.4957	27.5149	**31.2606**	30.7940	29.2676	28.3192	27.4982	**31.0342**
	1.5357	1.7666	1.8899	1.9636	**1.5202**	1.5466	1.7845	1.9083	1.9795	**1.5233**
2	31.6889	33.2296	**33.6676**	33.6663	31.4653	33.8433	33.6595	33.9846	34.0382	**34.0739**
	1.7135	1.5972	1.5396	**1.5374**	1.7480	1.4668	1.5445	1.4974	1.4871	**1.4473**
3	35.7152	34.7577	35.2213	34.9041	**35.8612**	35.7232	34.6331	35.0579	34.8693	**36.0636**
	1.4910	1.6598	1.5958	1.6084	**1.4691**	1.4943	1.6721	1.6008	1.6089	**1.4581**
4	37.3966	36.6168	36.3808	35.4652	**37.7831**	38.0096	36.7206	36.3960	35.6678	**38.2998**
	0.9094	0.9774	0.9766	0.9977	**0.8869**	0.8576	0.9635	0.9615	0.9774	**0.8385**
5	**35.4482**	34.9839	34.8997	34.6224	35.3821	36.1356	34.9657	34.7714	34.5816	**36.2018**
	1.3020	1.3508	1.3260	1.3333	1.3132	1.1861	1.3075	1.3213	1.3261	**1.1804**
6	**32.7081**	32.6411	31.8126	30.8156	32.5609	33.7802	32.6069	31.6062	30.7851	**33.8550**
	2.0318	2.1864	2.3790	2.4840	2.0839	1.8551	2.1709	2.3857	2.4808	**1.8533**
7	32.4465	**34.0239**	33.8198	33.8272	33.7754	34.0965	34.3593	34.5397	34.6761	**35.7373**
	1.2998	1.2157	1.2266	1.2288	**1.1676**	1.1592	1.1896	1.1590	1.1528	**1.0157**
8	**37.9098**	36.8763	36.7725	36.1020	37.8379	38.1854	36.6670	36.4265	35.9278	**38.2823**
	0.9885	1.1338	1.1444	1.1711	**0.9859**	0.9694	1.1576	1.1699	1.1954	**0.9626**
9	29.7212	30.8332	**31.2495**	30.8557	30.3530	31.3071	31.1581	31.4196	31.1759	**32.1709**
	1.8327	1.7679	**1.7192**	1.7774	1.7553	1.6277	1.7214	1.6968	1.7479	**1.5285**
10	36.8133	36.7662	**37.2501**	36.1575	37.1048	37.8106	37.0662	37.2927	36.3276	**38.1879**
	0.8758	0.8925	**0.8255**	0.8629	0.8562	0.7919	0.8491	0.8226	0.8594	**0.7698**
11	36.8098	36.7975	**37.0956**	36.7442	36.8842	37.5213	37.0497	37.0952	36.8198	**37.6985**
	0.8715	0.8954	**0.8286**	0.8377	0.8720	0.7926	0.8536	0.8263	0.8343	**0.7843**
12	33.8725	**34.5407**	34.4102	33.7818	34.0164	35.2610	34.6820	34.7541	34.2524	**35.6644**
	1.4666	1.4748	**1.4275**	1.4515	1.4450	1.3140	1.4288	1.3622	1.3772	**1.2713**
13	37.3884	37.8205	37.7569	37.3173	**38.0053**	38.3279	37.9377	37.8628	37.5068	**39.0792**
	0.6695	0.6731	0.6760	0.6841	**0.6463**	0.6267	0.6665	0.6610	0.6690	**0.5981**
14	27.8600	29.7386	30.4264	**30.5734**	27.7554	30.2549	30.2466	30.8242	**31.1554**	30.4845
	2.4652	2.5595	2.4457	**2.4330**	2.8844	2.3619	2.8077	2.3680	**2.3196**	2.3558
15	32.4833	30.8370	29.6090	28.4714	**32.4883**	32.6128	30.6644	29.3860	28.5487	**32.8477**
	1.7491	1.9406	2.1114	2.2016	1.7547	1.6518	1.9284	2.1159	2.1994	**1.6275**
16	34.4161	34.4301	34.3050	33.8643	**34.5715**	34.9354	34.3523	34.2067	33.8927	**35.0711**
	1.3868	1.4764	1.4804	1.4972	**1.3814**	1.3388	1.4682	1.4704	1.4846	**1.3264**
17	35.6650	37.3602	37.0917	37.0862	**37.5058**	37.2329	37.6885	37.8239	37.8704	**39.2991**
	1.0971	0.9964	1.0009	0.9993	**0.9665**	0.9865	0.9740	0.9477	0.9431	**0.8526**
18	35.7449	36.2947	**36.4685**	36.0800	35.7404	36.8960	36.5932	36.6429	36.2771	**36.9402**
	1.4857	1.4628	**1.3340**	1.3471	1.4897	1.3056	1.3572	1.3100	1.3213	**1.3029**
19	31.6767	**32.3393**	32.3295	32.0304	31.3846	**32.9921**	32.5119	32.2416	32.0507	32.9077
	2.2879	2.3592	2.3903	2.4202	2.3443	**2.0898**	2.3137	2.4326	2.4546	2.1113
20	34.5020	34.9738	**35.2707**	34.9622	34.9280	35.7424	35.2671	35.5570	35.3051	**36.4388**
	1.3409	1.3061	**1.2493**	1.2727	1.3294	1.1902	1.2452	1.2094	1.2304	**1.1637**
21	**35.8899**	35.7991	35.7714	35.2282	35.7142	36.8055	36.0108	35.9894	35.5311	**36.8352**
	1.0016	1.0396	1.0294	1.0521	1.0179	**0.9230**	1.0077	0.9971	1.0146	0.9239
22	33.0809	**34.0980**	33.8535	33.8285	32.9655	34.6893	34.3656	34.4198	34.5202	**34.8393**
	1.3691	**1.3142**	1.3468	1.3536	1.3895	1.2138	1.2900	1.2757	1.2687	**1.2031**
23	**33.5303**	32.8830	32.9540	32.4965	33.3529	33.7291	32.8127	32.8188	32.5483	**33.7452**
	1.3922	1.5024	1.5250	1.5490	1.4115	**1.3651**	1.5307	1.5364	1.5548	1.3662
24	38.0689	37.0203	37.0820	36.7586	**38.1592**	38.1993	36.9022	36.9148	36.7148	**38.4097**
	0.8977	0.9664	0.9820	0.9894	**0.8901**	0.8945	0.9867	0.9956	1.0017	**0.8834**
25	29.4449	29.8870	**30.0755**	29.8573	29.6185	30.0984	30.0602	30.0909	29.9332	**30.3096**
	1.4432	1.4933	1.5055	1.5331	1.4594	1.3599	1.4858	1.5123	1.5353	**1.3566**
Avg.	34.0523	34.1970	34.1628	33.7204	**34.2590**	34.9994	34.3300	34.2577	33.9390	**35.3791**
	1.4099	1.4403	1.4382	1.4635	**1.4027**	1.2948	1.4145	1.4217	1.4410	**1.2682**



(a) (b) (c)

(d) (e) (f)

Fig. 4: Zoom-in demosaicing results of test image No. 16. (a) Original picture; Demosaiced result in the interpolation step using (b) Lu's method, (c) Gunturk's method, (d) Li's method with UTV, (e) Li's method with STV, and (f) proposed HPHD-AI method.



(a) (b) (c)

(d) (e) (f)

Fig. 5: Zoom-in demosaicing results of test image No. 20. (a) Original picture; Demosaiced result in the interpolation step using (b) Lu's method, (c) Gunturk's method, (d) Li's method with UTV, (e) Li's method with STV, and (f) proposed HPHD-AI method.

7. ACKNOWLEDGMENT

The authors would like to thank Prof. B. K. Gunturk of Louisiana State University, USA; Prof. Yap-Peng Tan of Nanyang Technological University, Singapore; and Prof. Xin Li of West Virginia University, Morgantown, USA for providing us their CFA demosaicing programs. This work was supported by the National Science Council of Taiwan, ROC under grant NSC 92-2213-E-009-007.

8. REFERENCES

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

[2] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal Processing Magazine*, **Vol. 22**, No. 1, pp. 44-54, January 2005.

[3] W.-M. Lu and Y.-P. Tan, "Color filter array demosaicking: new method and performance measures," *IEEE transactions on Image Processing*, **Vol. 12**, No.10, pp. 1194-1210, October 2003.

[4] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE transactions on Image Processing*, **Vol. 11**, No.9, pp. 997-1013, September 2002.

[5] X. Li, "Demosaicing by successive approximation," *IEEE transactions on Image Processing*, **Vol. 14**, No.3, pp. 370-379, March 2005.

[6] K. Hirakawa and T. W. Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE transactions on Image Processing*, **Vol. 14**, No.3, pp. 360-369, March 2005.

[7] X.-L. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicking for digital cameras (patent pending)," *IEEE transactions on Image Processing*, **Vol. 13**, No.9, pp. 1263-1274, March 2004.

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

[9] S.-C. Pei and I.-K. Tam, "Effective color interpolation in CCD color filter arrays using signal correlation," *IEEE transactions on Circuits and Systems Video Technol.*, **Vol. 13**, No.6, pp. 503-513, Jun. 2003.

[10] L.-L. Chang and Y.-P. Tan, "Effective use of spatial and spectral correlations for color filter array demosaiking," *IEEE transactions on Consumer Electronics*, **Vol. 50**, No.1, pp.355-365, January 2004.

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

[12] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd edition, NJ: Prentice-Hall, 2002.

[13] R. Kimmel, "Demosaicking: Image reconstruction from color CCD samples," *IEEE transactions on Image Processing*, **Vol. 8**, No. 9, pp.1221-1228, 1999.

[14] R. Lukac, K. Martin, and K. N. Plataniotis, "Demosaicked image postprocessing using local color ratios," *IEEE Transactions on Circuits and Systems Video Technol.*, **Vol. 14**, No. 6, pp.914-920, 2004.

[15] M. Mahy, E. Van Eyckden, and O. Oosterlinck, "Evaluation of uniform color spaces developed after the adoption of CIELAB and CIELUV," *Color Res. Applicat.*, **Vol. 19**, No. 2, pp. 105-121, 1994.