Color Filter Array Demosaicking:
New Method and Performance Measures

Wenmiao Lu, *Student Member, IEEE,* and Yap-Peng Tan, *Member, IEEE*

Abstract—Single-sensor digital cameras capture imagery by covering the sensor surface with a color filter array (CFA) such that each sensor pixel only samples one of three primary color values. To render a full-color image, an interpolation process, commonly referred to as CFA demosaicking, is required to estimate the other two missing color values at each pixel. In this paper, we present two contributions to the CFA demosaicking: a new and improved CFA demosaicking method for producing high quality color images and new image measures for quantifying the performance of demosaicking methods. The proposed demosaicking method consists of two successive steps: an interpolation step that estimates missing color values by exploiting spatial and spectral correlations among neighboring pixels, and a post-processing step that suppresses noticeable demosaicking artifacts by adaptive median filtering. Moreover, in recognition of the limitations of current image measures, we propose two types of image measures to quantify the performance of different demosaicking methods; the first type evaluates the fidelity of demosaicked images by computing the peak signal-to-noise ratio and CIELAB $\triangle E_{ab}^*$ for edge and smooth regions separately, and the second type accounts for one major demosaicking artifact—zipper effect. We gauge the proposed demosaicking method and image measures using several existing methods as benchmarks, and demonstrate their efficacy using a variety of test images.

Index Terms—Color filter array demosaicking, color interpolation, false color, image measure, zipper effect.

I. INTRODUCTION

DIGITAL cameras are becoming popular and replacing traditional film-based cameras in many applications. To reduce the cost and size, most digital cameras acquire imagery using a single-chip CCD (Charge Coupled Device) or CMOS (Complementary Metal Oxide Semiconductor) sensor whose surface is covered with a color filter array (CFA). The CFA consists of a set of spectrally selective filters that are arranged in an interleaving pattern so that each sensor pixel samples one of the three primary color values (for example, red, green and blue values). We refer to these sparsely sampled color values as CFA samples. To render a full-color image from the CFA samples, an image reconstruction process, commonly referred to as CFA demosaicking or CFA interpolation method, is required to estimate for each pixel its two missing color values.

An immense number of demosaicking methods have been proposed in the literature [1]–[15]. The simplest one is probably bilinear interpolation, which fills missing color values with weighted averages of their neighboring pixel values. Although computationally efficient and easy to implement, bilinear interpolation introduces severe demosaicking artifacts and smears sharp edges. To obtain more visually pleasing results, many adaptive CFA demosaicking methods have been proposed to exploit the spectral and spatial correlations among neighboring pixels.

In [1], Cok presents a simple spectral correlation between different color planes: within a local image region, the ratios between the red and green values are highly similar, so are the ratios between the blue and green values. Based on this observation, several schemes (e.g., [1], [2], [13]) have been devised to estimate missing color values with the aid of other color planes. Besides color ratios, many methods also make use of inter-channel color differences (red-green and blue-green) [1]–[4]. However, these methods normally do not perform satisfactorily around sharp edges and fine details, where the assumed spectral correlation does not necessarily hold. In [15], Gunturk *et al.* have recently proposed an effective scheme to exploit spectral correlation by alternately projecting the estimates of the missing color values onto constraint sets based on original CFA samples and prior knowledge of spectral correlation.

As many undesired demosaicking artifacts are due to improper fusing of neighboring pixel values, a number of CFA demosaicking methods (e.g., [2], [5]–[10]) first analyze the spatial structure of a local image neighborhood and then select suitable interpolation schemes or neighboring pixels to estimate the missing color values. In general, by exploiting this spatial correlation among neighboring pixels, these methods aim to perform color interpolation along image edges, rather than across them. For example, several edge classifiers are proposed in [6], [7], [10] to identify the best directions for interpolating the missing color values.

By taking the advantages of both types of demosaicking methods mentioned above, hybrid demosaicking methods are capable of generating demosaicked images with sharper reconstructed edges and less visible demosaicking artifacts. In [13], Kimmel uses an edge-directed scheme to estimate each missing color value by iteratively interpolating its four surrounding color values or color ratios weighted by some edge indicators; an inverse diffusion process is then used to remove artifacts by smoothing the demosaicked results. In a recent related effort, Li and Orchard [14] propose to use a new edge-directed scheme

Fig. 1.   Schematic diagram of the Bayer color filter array pattern.

CFA samples

Interpolation Step

Green plane interpolation

Fully-populated green plane, CFA sampled red and blue planes

Red and blue plane interpolation

Full-color demosaicked image

Fig. 2.   Flowchart of the interpolation step.

to interpolate inter-channel color differences by exploiting the geometric duality between the covariances of low-resolution CFA samples and high-resolution demosaicked images.

In this paper, we propose a new and improved hybrid CFA demosaicking method. It consists of two successive steps: an interpolation step to render full-color images and a post-processing step to suppress visible demosaicking artifacts. Although the proposed method can be generalized to other CFA patterns, we limit our discussion in this paper to the Bayer CFA pattern [16] because of its popularity. Fig. 1 shows a schematic diagram of the Bayer CFA pattern, where R, G and B denote the pixels having only red, green and blue color values, respectively.

The interpolation step of our proposed method fills in missing color values progressively; the green plane is the first to be fully populated, and then it is used to assist the red and blue plane interpolation. The three color planes are interpolated in the same manner as follows. To interpolate a missing value of one color plane, we first obtain its estimates from four different interpolation directions by exploiting the spectral correlation and then combine these estimates with proper weights. The weights are determined based on the spatial correlation among pixels along the respective interpolation directions to reflect the odds that the associated estimates are accurate.

The proposed post-processing step is designed to suppress visible artifacts residing in the demosaicked images obtained from the interpolation step. We extend the Freeman's median filtering method [3] by lifting the constraint of keeping the original CFA-sampled values intact. This extension can reduce more effectively such demosaicking artifacts as "false colors" and "zipper effect." Furthermore, to avoid tempering the good estimates obtained in the interpolation step, the post-processing step is only selectively applied to image regions around sharp edges and fine details, which are prone to demosaicking artifacts.

In another major contribution of the paper, we propose several image measures to quantify the performance of demosaicking methods, which include modified peak signal-to-noise ratio (PSNR) and CIELAB $\triangle E_{ab}^*$ as well as a new objective measure for one prominent demosaicking artifact—zipper effect [8]. The experimental results show that the proposed
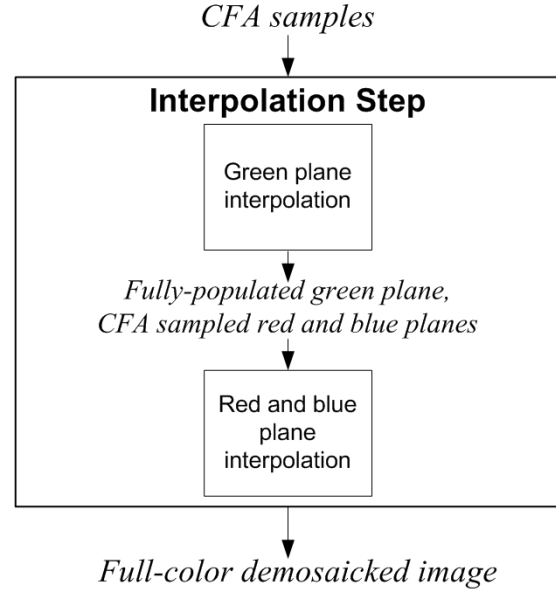
image measures are well correlated with the human perceptual evaluation of demosaicked results.

The remainder of this paper is organized as follows. Sections II and III are devoted to the interpolation step and the post-processing step of the proposed demosaicking method, respectively. Section IV describes the proposed image measures for quantifying the performance of different demosaicking methods. The experimental results are presented with comparison to that of other existing methods in Section V, and the paper is concluded in Section VI.

II. INTERPOLATION STEP

We describe in this section the first step of our proposed demosaicking method, an interpolation step that progressively renders a full-color image from CFA samples. Fig. 2 depicts the flowchart of this interpolation step; the green plane is the first to be interpolated and, once fully populated, used to assist the subsequent red and blue plane interpolation. Although processed in a sequential order, the three color planes are interpolated in the same manner. Specifically, every missing color value is interpolated by properly combining the estimates obtained from its four interpolation directions, which are defined according to the four nearest CFA samples of the same color.

The estimate from one particular interpolation direction is obtained by exploiting the spectral correlation among the neighboring pixels along that direction. The spectral correlation refers to the assumption that the differences between the green and red/blue values within a local neighborhood are well correlated with constant offsets [1], [2], [17]. However, this spectral correlation is defined for the color values at each individual pixel, which are not fully available from the CFA samples. To make use of this spectral correlation, we further assume that the rate of change of neighboring pixel values along an interpolation direction is also a constant. Clearly, all these assumptions fail to
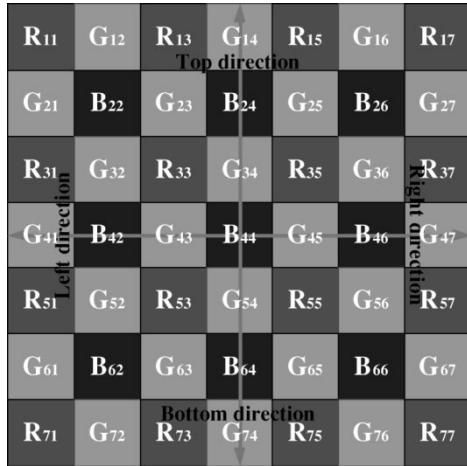
Fig. 3. A 7×7 window where the green value of the central pixel is to be estimated.

hold in the presence of sharp edges and fine details, where color values experience abrupt changes.

The abrupt changes of color values indicate low spatial correlation among neighboring pixels. Intuitively, the higher the spatial correlation among pixels along an interpolation direction, the more accurate the estimate of a missing color value can be obtained from that direction. Therefore, to interpolate one missing color value, we combine the estimates from its four interpolation directions by assigning them with weights that measure the spatial correlations among the neighboring pixels along the corresponding interpolation directions.

In Sections II-A and B, we will describe the interpolation of each color plane in detail. Note that although the missing color values at each pixel will be progressively interpolated, we shall denote pixels as red, green and blue pixels according to the color of their original CFA samples.

A. Green Plane Interpolation

We start with interpolating the green plane because it contains twice as many samples as the red or blue plane. Thus, the green plane possesses most spatial information of the image to be demosaicked and has great influence on the perceptual quality of the image. Furthermore, once fully interpolated, the green plane can assist the subsequent red and blue plane interpolation by making full and direct use of the spectral correlation.

Fig. 3 shows a 7×7 window of CFA samples, where the indices i and j of each color value (R_{ij}, G_{ij} or B_{ij}) denote its row and column locations, respectively. Consider the central pixel whose green value G_{44} is to be estimated from the four interpolation directions corresponding to its four nearest green pixels, G_{34}, G_{43}, G_{45} and G_{54}, at the top, left, right and bottom locations, respectively. The missing green value G_{44} is estimated by

$$\hat{G}_{44} = \frac{\alpha_{34}\tilde{G}_{34} + \alpha_{43}\tilde{G}_{43} + \alpha_{45}\tilde{G}_{45} + \alpha_{54}\tilde{G}_{54}}{\alpha_{34} + \alpha_{43} + \alpha_{45} + \alpha_{54}} \quad (1)$$

where $\tilde{G}_{ij}$ is a color-adjusted green value at pixel (i, j) and α_{ij} is its associated weight. To exploit the spectral correlation, the four neighboring green pixel values, G_{34}, G_{43}, G_{45} and G_{54},

are adjusted with the help of the blue pixels along the respective interpolation directions, given by

$$\tilde{G}_{34} = G_{34} + \frac{(B_{44} - B_{24})}{2}$$
$$\tilde{G}_{43} = G_{43} + \frac{(B_{44} - B_{42})}{2}$$
$$\tilde{G}_{45} = G_{45} + \frac{(B_{44} - B_{46})}{2}$$
$$\tilde{G}_{54} = G_{54} + \frac{(B_{44} - B_{64})}{2}. \quad (2)$$

These color values are adjusted based on two assumptions: 1) the green and blue/red pixel values are well correlated with constant offsets; and 2) the rate of change of neighboring pixel values along an interpolation direction is also a constant. We now show how these two assumptions lead to (2).

Consider the top interpolation direction of G_{44}. The first assumption suggests that the following relationship exists among the pixels along that direction

$$\bar{G}_{44} - B_{44} = G_{34} - \bar{B}_{34} = \bar{G}_{24} - B_{24} \quad (3)$$

where $\bar{G}_{24}$, $\bar{G}_{44}$ and $\bar{B}_{34}$ denote the missing color values at the respective pixel locations.

Since $\bar{B}_{34}$ is not available from the CFA samples, we can only use B_{24} and B_{44} to assist the estimation of $\bar{G}_{44}$, and rewrite the above relationship as

$$\bar{G}_{44} - \bar{G}_{24} = B_{44} - B_{24}. \quad (4)$$

The second assumption gives the following relationship for the neighboring green pixel values along the top interpolation direction

$$\bar{G}_{44} - G_{34} = G_{34} - \bar{G}_{24}. \quad (5)$$

Combining (4) and (5), we have

$$B_{44} - B_{24} = \bar{G}_{44} - \bar{G}_{24}$$
$$= (\bar{G}_{44} - G_{34}) + (G_{34} - \bar{G}_{24})$$
$$= 2(\bar{G}_{44} - G_{34}) \quad (6)$$

which states that $\bar{G}_{44} = G_{34} + (B_{44} - B_{24})/2$. Therefore, the color-adjusted green value $\tilde{G}_{34}$ is in fact the estimate of $\bar{G}_{44}$ from the top interpolation direction.

To properly combine the estimates from the four interpolation directions, we assign each estimate with an appropriate weight α, which is the reciprocal of the gradient magnitude along the respective interpolation direction. As the two aforementioned assumptions fail to hold in the presence of sharp edges and fine details, the weights in fact reflect the odds that the associated estimates are accurate.

Let us illustrate how the gradient magnitude is computed by using the top interpolation direction of G_{44} as an example. Referring to Fig. 4, the gradient magnitude along the top interpolation direction is computed by applying a Sobel operator [18] and a 1-D gradient operator on the Group A and Group B pixels, respectively. As a result, the gradient magnitude of this interpolation direction is obtained as $|G_{54} - G_{34}| + |G_{34} - G_{14}| + |B_{44} - B_{24}| + |G_{43} - G_{23}|/2 + |G_{45} - G_{25}|/2$. Since a large
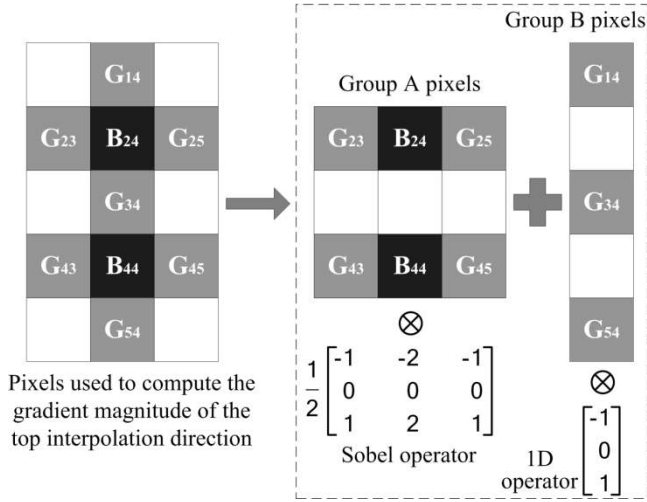
Fig. 4.   Illustration of computing the gradient magnitude of an interpolation direction.

Fig. 5.   A 5×5 window where the red value of the central pixel is to be estimated.

gradient magnitude implies low spatial correlation among pixels along the respective interpolation direction, proper weights of the four color-adjusted green values are given by (7), shown at the bottom of the page, where 1 in the denominator is included to avoid division by zero.

Once the green plane is fully populated, it will be used to assist the subsequent red and blue interpolation, as detailed in Section II-B.

B. Red and Blue Plane Interpolation

Since the red and blue planes are more sparsely sampled than the green plane, their interpolation evolves over two sub-steps: 1) interpolating the missing red values at blue pixels and vice versa, and then 2) filling the rest of the missing red and blue values at green pixels. As the same procedure is used to interpolate the missing red and blue values, only the red plane interpolation will be described.

Referring to Fig. 5, the red value R_{33} of the pixel at the center of a 5×5 window is to be estimated. Similar to the green plane interpolation, the missing red value R_{33} is estimated by

$$\hat{R}_{33} = \frac{\alpha_{22}\tilde{R}_{22} + \alpha_{24}\tilde{R}_{24} + \alpha_{42}\tilde{R}_{42} + \alpha_{44}\tilde{R}_{44}}{\alpha_{22} + \alpha_{24} + \alpha_{42} + \alpha_{44}} \qquad (8)$$

where $\tilde{R}_{ij}$ is a color-adjusted red value at pixel (i,j) and α_{ij} is its associated weight. Note that for the sake of clarity, not
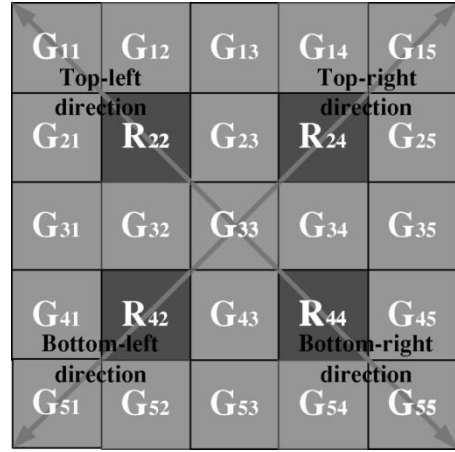
all the available color values are shown in Fig. 5; it should be understood that at this step the green values are available at all pixels and the blue values exist at pixels whose row index i and column index j are odd integers.

Following a derivation similar to that for the green plane interpolation, the four neighboring red values, R_{22}, R_{24}, R_{42} and R_{44}, are adjusted with the help of the green pixel values along the respective interpolation directions as follows:

$$\begin{aligned} \tilde{R}_{22} &= R_{22} + (G_{33} - G_{22}) \\ \tilde{R}_{24} &= R_{24} + (G_{33} - G_{24}) \\ \tilde{R}_{42} &= R_{42} + (G_{33} - G_{42}) \\ \tilde{R}_{44} &= R_{44} + (G_{33} - G_{44}). \end{aligned} \qquad (9)$$

Based on the fully populated green plane, the weights associated with these color-adjusted red values are given by (10), shown at the bottom of the next page.

Finally, we proceed to fill in the rest of the missing red values at green pixels. Referring to Fig. 6, the red value R_{33} of the pixel at the center of the 5×5 local window is similarly estimated as follows:

$$\hat{R}_{33} = \frac{\alpha_{23}\tilde{R}_{23} + \alpha_{32}\tilde{R}_{32} + \alpha_{34}\tilde{R}_{34} + \alpha_{43}\tilde{R}_{43}}{\alpha_{23} + \alpha_{32} + \alpha_{34} + \alpha_{43}} \qquad (11)$$

where $\tilde{R}_{ij}$ is a color-adjusted red value at pixel (i,j) and α_{ij} is its associated weight. Note again that not all the available color values are shown in Fig. 6.

$$\alpha_{34} = \frac{1}{1 + |G_{54} - G_{34}| + |G_{34} - G_{14}| + |B_{44} - B_{24}| + \left|\frac{G_{43} - G_{23}}{2}\right| + \left|\frac{G_{45} - G_{25}}{2}\right|}$$

$$\alpha_{43} = \frac{1}{1 + |G_{45} - G_{43}| + |G_{43} - G_{41}| + |B_{44} - B_{42}| + \left|\frac{G_{34} - G_{32}}{2}\right| + \left|\frac{G_{54} - G_{52}}{2}\right|}$$

$$\alpha_{45} = \frac{1}{1 + |G_{43} - G_{45}| + |G_{45} - G_{47}| + |B_{44} - B_{46}| + \left|\frac{G_{34} - G_{36}}{2}\right| + \left|\frac{G_{54} - G_{56}}{2}\right|}$$

$$\alpha_{54} = \frac{1}{1 + |G_{34} - G_{54}| + |G_{54} - G_{74}| + |B_{44} - B_{64}| + \left|\frac{G_{43} - G_{63}}{2}\right| + \left|\frac{G_{45} - G_{65}}{2}\right|} \qquad (7)$$

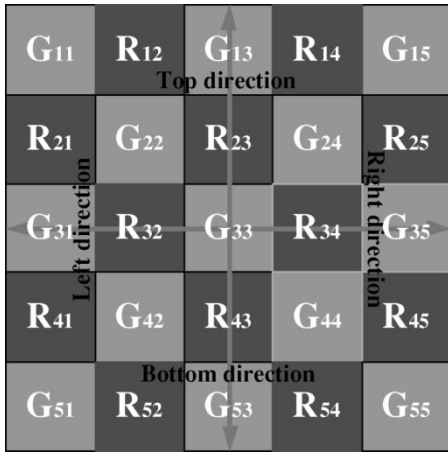Fig. 6. A 5×5 window where the red value of the central pixel is to be estimated.

The four neighboring red values are adjusted with the help of the associated green values as follows:

$$
\begin{aligned}
\tilde{R}_{23} &= R_{23} + (G_{33} - G_{23}) \\
\tilde{R}_{32} &= R_{32} + (G_{33} - G_{32}) \\
\tilde{R}_{34} &= R_{34} + (G_{33} - G_{34}) \\
\tilde{R}_{43} &= R_{43} + (G_{33} - G_{43}).
\end{aligned} \tag{12}
$$

The weights of these color-adjusted red values are also computed based on the fully populated green plane, as given by (13) shown at the bottom of the page.

A full-color image can be obtained after applying all the interpolation processes described above.

III. POST-PROCESSING STEP

The proposed post-processing step aims to suppress visible artifacts residing in the initial demosaicked images obtained from the aforementioned interpolation step. There are two main types of demosaicking artifacts, namely false colors and zipper effect. False colors are those artifacts corresponding to noticeable color errors as compared to the original, nonmosaicked image. One example is shown in Fig. 7(a), where the left hand is a full-color image and the right hand is its demosaicked image with false colors around the numbers region. The zipper effect refers to abrupt or unnatural changes of color differences between neighboring pixels, manifesting as an "on-off" pattern. One example is shown in Fig. 7(b), where the left hand is a full-color image and the right hand is its demosaicked image with the zipper effect around the fence region.

As demosaicking artifacts often exhibit as color outliers, Freeman [3] proposed to use a median filter to process the inter-channel differences (red-green and blue-green) of demosaicked images obtained by bilinear interpolation. The rationale is that median filtering the inter-channel differences can force pixels with distinct colors to be more similar to their neighbors, thus eliminating color outliers or errors. Note that in Freeman's method, the original CFA-sampled color value at each pixel is not altered, and it is combined with median-filtered inter-channel differences to obtain the other two missing color values. In general, Freeman's method is rather effective in suppressing demosaicking artifacts, while preserving sharp edges. However, when applying this method to post-process the initial demosaicked images obtained from our proposed interpolation step, we observed that some demosaicking artifacts still remain around sharp edges and fine details. This is partly due to the fact that each pixel has two independent inter-channel

$$
\begin{aligned}
\alpha_{22} &= \frac{1}{1 + |G_{11} - G_{22}| + |G_{22} - G_{33}| + |\frac{G_{21} - G_{32}}{2}| + |\frac{G_{12} - G_{23}}{2}|} \\
\alpha_{24} &= \frac{1}{1 + |G_{15} - G_{24}| + |G_{24} - G_{33}| + |\frac{G_{14} - G_{23}}{2}| + |\frac{G_{25} - G_{34}}{2}|} \\
\alpha_{42} &= \frac{1}{1 + |G_{51} - G_{42}| + |G_{42} - G_{33}| + |\frac{G_{41} - G_{32}}{2}| + |\frac{G_{52} - G_{43}}{2}|} \\
\alpha_{44} &= \frac{1}{1 + |G_{55} - G_{44}| + |G_{44} - G_{33}| + |\frac{G_{54} - G_{43}}{2}| + |\frac{G_{45} - G_{34}}{2}|}
\end{aligned} \tag{10}
$$

$$
\begin{aligned}
\alpha_{23} &= \frac{1}{1 + |G_{33} - G_{23}| + |G_{23} - G_{13}| + |\frac{G_{32} - G_{22}}{2}| + |\frac{G_{22} - G_{12}}{2}| + |\frac{G_{34} - G_{24}}{2}| + |\frac{G_{24} - G_{14}}{2}|} \\
\alpha_{32} &= \frac{1}{1 + |G_{33} - G_{32}| + |G_{32} - G_{31}| + |\frac{G_{23} - G_{22}}{2}| + |\frac{G_{22} - G_{21}}{2}| + |\frac{G_{43} - G_{42}}{2}| + |\frac{G_{42} - G_{41}}{2}|} \\
\alpha_{34} &= \frac{1}{1 + |G_{33} - G_{34}| + |G_{34} - G_{35}| + |\frac{G_{23} - G_{24}}{2}| + |\frac{G_{24} - G_{25}}{2}| + |\frac{G_{43} - G_{44}}{2}| + |\frac{G_{44} - G_{45}}{2}|} \\
\alpha_{43} &= \frac{1}{1 + |G_{33} - G_{43}| + |G_{43} - G_{53}| + |\frac{G_{32} - G_{42}}{2}| + |\frac{G_{42} - G_{52}}{2}| + |\frac{G_{34} - G_{44}}{2}| + |\frac{G_{44} - G_{54}}{2}|}
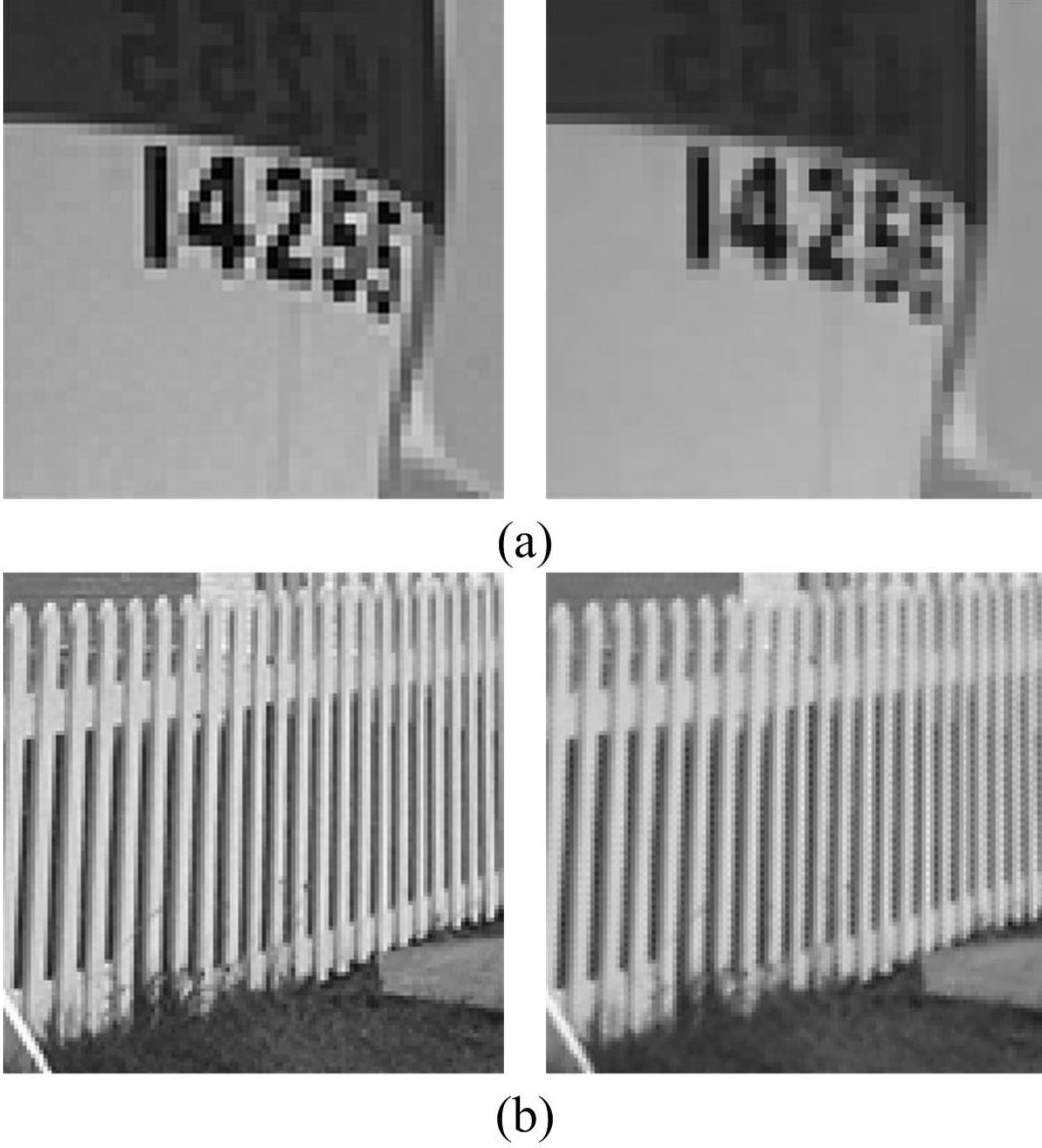\end{aligned} \tag{13}
$$

(a)

(b)

Fig. 7.   (a) Example of false colors and (b) example of zipper effect.

differences, and filtering the differences separately does not take into account the spectral correlation between color planes.

To make use of the spectral correlation, one intuitive extension of Freeman's method is to adopt a vector median filter [19] which takes as input the vector of two inter-channel differences. However, we found this extension barely shows any improvement in suppressing artifacts as compared to Freeman's method. This is because when the three color planes are separately interpolated, the estimation errors incurred in different color planes, which can be regarded as additive noise, are rather independent from each other. It has been shown in [19] that when the noise is independent in different vector components, the vector-based median filtering cannot outperform the component-wise median filtering in suppressing noise.

To incorporate median filtering with the spectral correlation for more effective suppression of demosaicking artifacts, we lift the constraint of keeping the original CFA-sampled color values intact. Furthermore, we make use of the latest processed color values to filter the subsequent pixels so that estimation errors can be effectively diffused into local neighborhoods. Specifically, we adjust the three color values at the central pixel of a local window (the window size is equal to the support of the median filter) as follows:

$$\hat{G}_{center} = \frac{(R_{center} - v_{RG}) + (B_{center} - v_{BG})}{2}$$
$$\hat{R}_{center} = \hat{G}_{center} + v_{RG}$$
$$\hat{B}_{center} = \hat{G}_{center} + v_{BG} \tag{14}$$

where $v_{RG} = median\{R_{ij} - G_{ij}|(i,j) \in \Re\}$, $v_{BG} = median\{B_{ij} - G_{ij}|(i,j) \in \Re\}$, and $\Re$ denotes the support of the median filter, which covers 5×5 pixels in our implementation. Subsequently, the color values of the central pixel are replaced by $\hat{R}_{center}, \hat{G}_{center}$ and $\hat{B}_{center}$ so that they will be involved in filtering the following pixels.

Experimental results show that this post-processing step is capable of suppressing most visually annoying artifacts. The first column of Fig. 8 shows the originals of two test image regions:
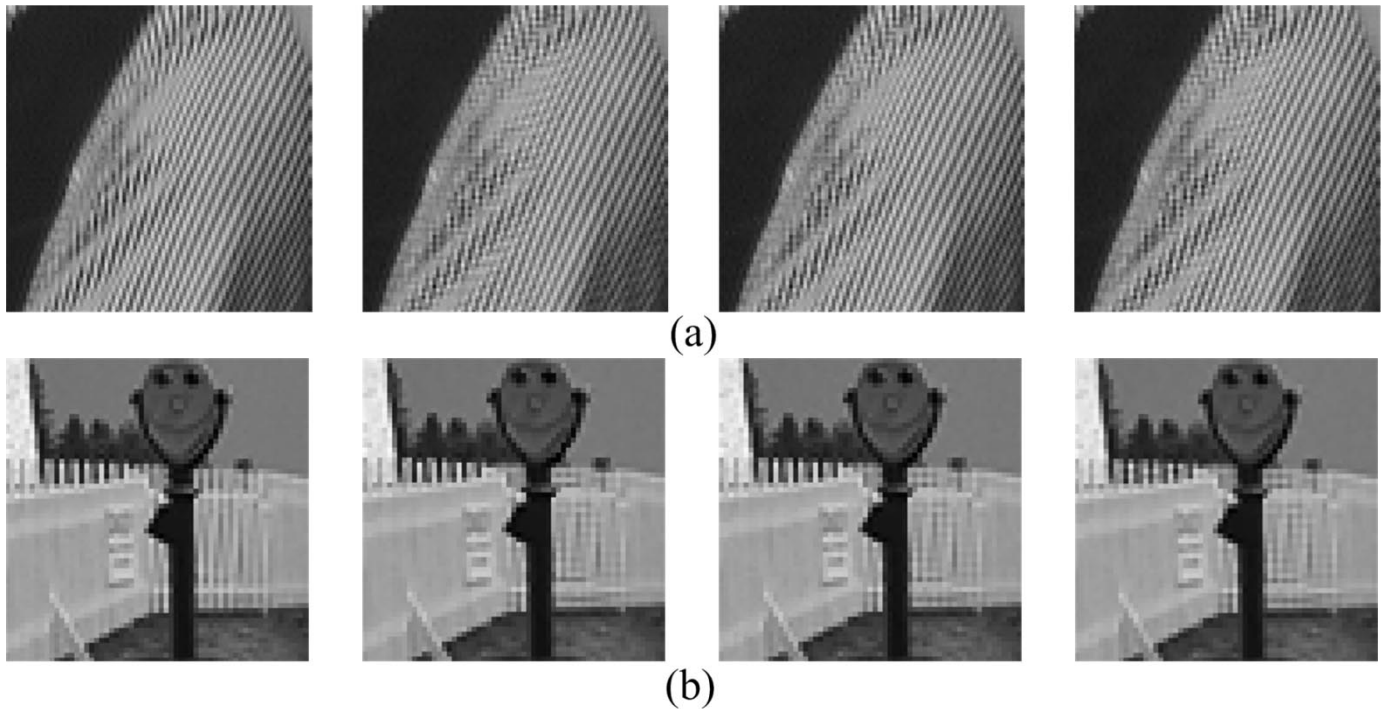
Fig. 8.   (From left to right) Nonmosaicked originals, initial demosaicked images, post-processing results using Freeman's method and our post-processing step of (a) the trousers region of Barbara image and (b) the fence region of Lighthouse image.

the trousers region in Barbara image and the fence region in Lighthouse image. (See Fig. 14 for all the test images examined in this paper.) The fine details in these two regions pose great challenges to many demosaicking methods. The second column of Fig. 8 shows the initial demosaicked images, which have some visible false colors and zipper artifacts, obtained by the proposed interpolation step. We then post-process them with Freeman's method and our post-processing step; the results are shown in the third and fourth columns of Fig. 8, respectively. It is clear that our proposed post-processing step can perform noticeably better than Freeman's method.

Although effective in eliminating most demosaicking artifacts, the proposed post-processing method, like many other median filtering schemes, may de-saturate the color of demosaicked images if applied indiscriminately. To strike a good balance between suppression of demosaicking artifacts and preservation of color fidelity, the post-processing step is only selectively applied around image regions that are prone to demosaicking artifacts. We detect these artifact-prone regions as follows.

1) Convolve the green plan, denoted as g, of the initial demosaicked image with the following discrete Laplacian operator

$$ h = \frac{1}{11} \begin{bmatrix} 1 & 9 & 1 \\ 9 & -40 & 9 \\ 1 & 9 & 1 \end{bmatrix} \qquad (15) $$

which is a second-order derivative operator for detecting image edges and details.

2) Threshold the output of convolution $f = g * h$ to produce a binary edge map, given by

$$ m(i,j) = \begin{cases} 1, & \text{if } |f(i,j)| > T \\ 0, & \text{otherwise} \end{cases} \qquad (16) $$

where T is the threshold for identifying pixels prone to demosaicking artifacts (i.e., those pixels corresponding to $m(i,j) = 1$). In our implementation, we found that setting T equal to 15 gives satisfactory identification of artifact-prone regions. Fig. 9 shows the edge map m of the initial demosaicked Lighthouse image obtained by the procedures described above. The post-processing step will only be applied to those pixels (highlighted in white) that are identified to be prone to demosaicking artifacts.

IV. IMAGE MEASURES FOR DEMOSAICKING PERFORMANCE

While a large number of demosaicking methods have been proposed in the literature, there is, however, a lack of image measures which can effectively quantify the performance of these methods. Ideally, from a psychological standpoint, effective image measures should be in accordance with perceptual judgements made by human observers. Such image measures could help simplify the development, improvement, and evaluation of different demosaicking methods, as they would rely much less on subjective test of image preference, which is not only time consuming but also dependent on the experimental settings and the subjects involved.

It is generally agreed that current image measures cannot effectively quantify the perceptual quality of images obtained from a restoration process [14], [20], [21]. This limitation holds for CFA demosaicking and is supported by the recent experiments conducted by Philippe et al. on the subjective preference
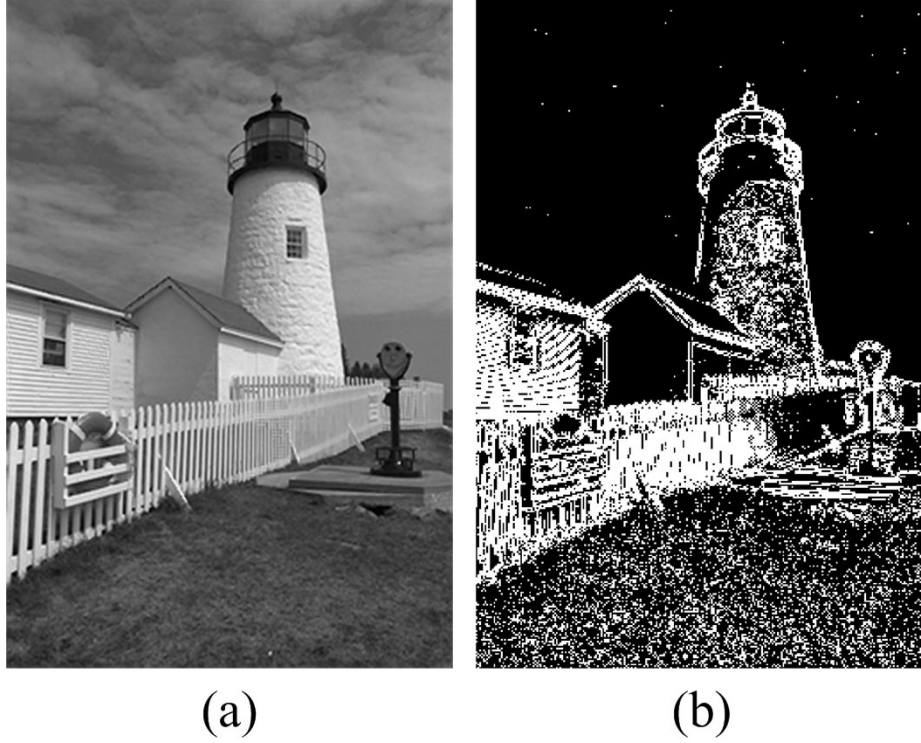
Fig. 9. (a) Initial demosaicked Lighthouse image and (b) its artifact-prone regions detected by the Laplacian operator.

of demosaicked results [22]. Their experiments reveal that when evaluated using the current image measures, some demosaicked results that are very different from the nonmosaicked originals are the most preferred. They also discover that many most-preferred results are sharpened by the demosaicking method under examination. Only when the image sharpness is removed as a salient factor, the nonmosaicked originals are the most preferred.

As it remains unclear what factors are conducive to visually-pleasing images, we will treat demosaicking as a reconstruction process. In our view, since subjective image preference could vary from individual to individual, the images should be first reproduced as accurately as possible before any enhancement process, such as image sharpening, is applied. Therefore, we propose to quantify the performance of a demosaicking method from two aspects: 1) to measure the image fidelity of its demosaicked results as compared to the nonmosaicked originals and 2) to measure the amount of artifacts residing in the demosaicked results, in particular, the zipper effect.

A. Measures for Image Fidelity

Peak signal-to-noise ratio (PSNR) and CIELAB $\triangle E_{ab}^*$ are two common measures for assessing the image fidelity of the outputs of many image processing tasks, such as image restoration, enhancement, and compression. The PSNR in decibels (dB) for a color plane c of a processed image computes the mean squared difference between the processed image and its original image, given by

$$PSNR = 10 \times log_{10} \left\{ \frac{255^2}{\frac{1}{N} \sum_{1 \leq n \leq N} \|O_c(n) - R_c(n)\|^2} \right\} \quad (17)$$

where N is the total number of image pixels, $O_c(n)$ is the color value (red, green, or blue) of the nth pixel in color plane c of the original image, and $R_c(n)$ is the corresponding color value in the reconstructed image. Note that the PSNR computed in RGB color space does not equate with human perception of color difference. The CIELAB $\triangle E_{ab}^*$, which measures the Euclidean distance between the original and the reconstructed images in CIELAB color space (a more perceptually uniform color space recommended by the CIE [23]), can better capture the color difference perceived by human observers. (Readers are referred to [23] and [24] for details of converting a color image from RGB color space to CIELAB color space.) The CIELAB $\triangle E_{ab}^*$ is given by

$$\triangle E_{ab}^* = \frac{1}{N} \sum_{1 \leq n \leq N} \|O(n)_{Lab} - R(n)_{Lab}\| \quad (18)$$

where $O(n)_{Lab}$ and $R(n)_{Lab}$ are the CIELAB color values of the nth pixel in the original and reconstructed images, respectively. According to [25], the $\triangle E_{ab}^*$ greater than 2.3 indicates the color difference is visible. When the $\triangle E_{ab}^*$ is greater than 10, the reconstructed image is so different from the original that color comparison is not worthwhile.

Nevertheless, the PSNR and $\triangle E_{ab}^*$ are not considered effective especially when used as global measures of image fidelity [14], [22]. Most demosaicking methods perform reasonably well in smooth regions; however, they normally result in different amounts of artifacts and color errors around edge regions. Both PSNR and $\triangle E_{ab}^*$, when computed for the entire image, will likely disguise estimation errors incurred in edge regions with those in smooth regions. One example is shown in Fig. 10, where the two demosaicked images have almost

Fig. 10.   Two demosaicked images with almost the same PSNR and $\triangle E_{ab}^*$ values but different amounts of demosaicking artifacts perceived in smooth and edge regions.

Fig. 11.   Nonmosaicked original and its detected edge regions.

TABLE I
PSNR (IN DECIBELS) AND $\triangle E_{ab}^*$ VALUES COMPUTED FOR THE EDGE AND SMOOTH REGIONS OF THE TWO DEMOSAICKED IMAGES IN FIG. 10

	Left-hand demosaicked image			Right-hand demosaicked image		
	Edge regions	Smooth regions	Overall	Edge regions	Smooth regions	Overall
PSNR	28.65	33.52	29.58	31.62	29.32	30.32
$\triangle E_{ab}^*$	6.42	0.25	2.48	3.54	1.95	2.78

the same PSNR and $\triangle E_{ab}^*$ when computed for the entire image—29.58 dB (PSNR) and 2.48 $\triangle E_{ab}^*$ for the left-hand image, and 30.32 dB (PSNR) and 2.78 $\triangle E_{ab}^*$ for the right-hand image. However, it is visually clear that these two demosaicked images have different amounts and types of demosaicking artifacts around their edge and smooth regions. This example demonstrates the need for better image measures to assess the fidelity of demosaicked images.

To evaluate the fidelity of demosaicked images of different methods, we compute the PSNR and $\triangle E_{ab}^*$ for edge and smooth regions separately. For fair comparison, the edge and smooth regions are identified from the same original, nonmosaicked image using the following procedure.

1) Perform edge detection on the grayscale of the nonmosaicked original to obtain an edge map.
2) Label the edge regions by dilating the edge map with the following 3×3 structuring element [18]

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \tag{19}$$

3) Label the rest of the image as the smooth regions.

Note that in our implementation, we have used the edge detection function (with the Sobel method and the default threshold) and the dilation function from MATLAB 6.5 [26]. Fig. 11 shows
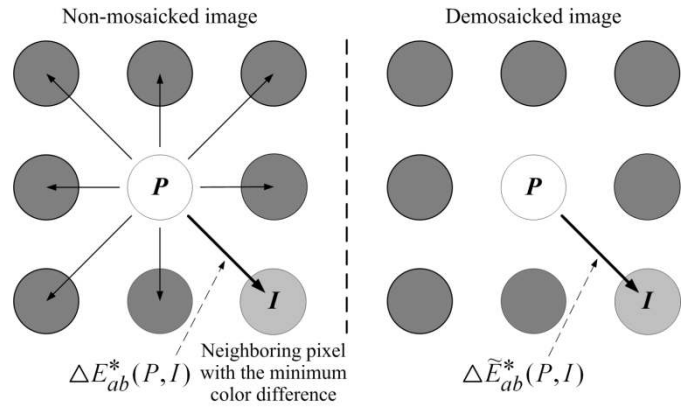
Fig. 12.   Illustration of computing the change in color difference between a pixel under consideration, P, and its eight neighboring pixels in the original nonmosaicked image and the demosaicked image, respectively.

the edge regions (highlighted in white) detected in the nonmosaicked original, from which the two demosaicked images shown in Fig. 10 are derived. As can be seen from Table I, the PSNR and $\triangle E_{ab}^*$ computed for the edge and smooth regions correlate well with the observation that the demosaicking artifacts in the left-hand image mainly exist around the edge regions, and that in the right-hand image mainly in the smooth regions. Therefore, evaluating the PSNR and $\triangle E_{ab}^*$ for the edge

Fig. 13.   (From left to right) Nonmosaicked original and its demosaicked results with 91.14%, 78.39%, 58.74%, 11.22%, and 5.43% pixels having the zipper effect.

Fig. 14.   Test images: Barbara, Window, Brickhouse, Bike, Lighthouse, Sails, and Starburst (from left-to-right and top-to-bottom).

and smooth regions separately can provide more effective and informative measures for the fidelity of demosaicked images.

B. Measure for Zipper Effect

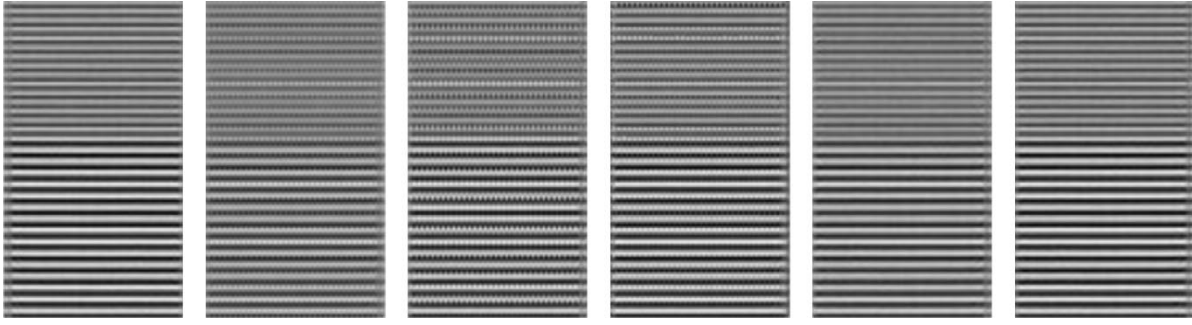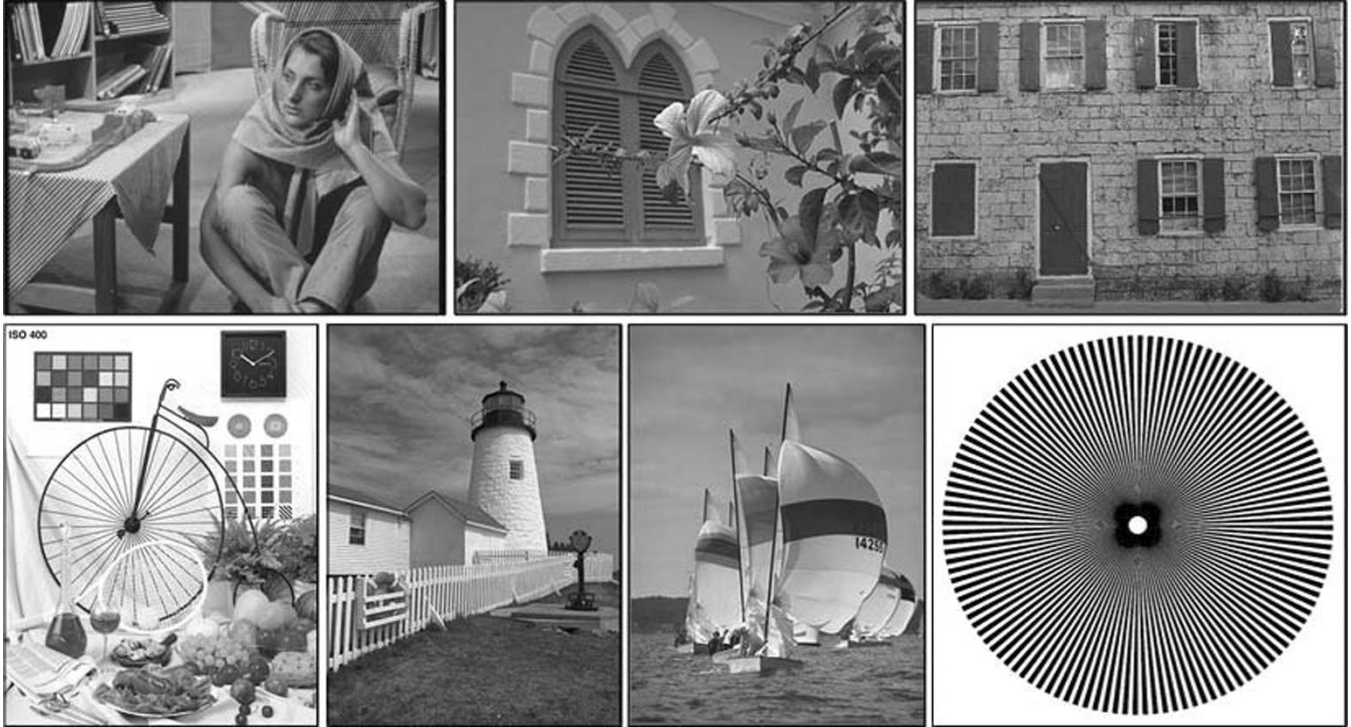While $\triangle E_{ab}^*$ accounts for false colors (i.e., visible color differences as compared to the nonmosaicked original), another objective measure is designed to quantify the zipper effect that results from improper fusing of neighboring pixel values (see Fig. 7(b) for the example of zipper effect). The zipper effect mainly manifests as "on-off" image patterns artificially created in smooth regions near edges. For an individual pixel, the zipper effect is in fact an increase in color difference with respect to its most similar neighboring pixel. Based on this observation, we first compute the change in color difference between each pixel and its most similar neighboring pixel as follows.

1) For a pixel P in the original, nonmosaicked image, we compute the color differences to its eight nearest neigh-

bors, as shown in Fig. 12. We identify its neighboring pixel I that has the minimum color difference as follows:

$$I = \min_{i \in \aleph} \triangle E_{ab}^*(P, i) \qquad (20)$$

where $\aleph$ denotes the set of eight neighboring pixels.

2) We then compute the CIELAB $\triangle E_{ab}^*$ color difference between the same pair of pixels, P and I, in the demosaicked image (referring to Fig. 12), and denote it as $\triangle \tilde{E}_{ab}^*(P, I)$.

3) The change in the $\triangle E_{ab}^*$ color difference between pixels P and I is given by

$$\psi = \triangle \tilde{E}_{ab}^*(P, I) - \triangle E_{ab}^*(P, I). \qquad (21)$$

When $|\psi| > \delta$ (in our implementation δ is set to 2.3 based on the results reported in [25]), the pixel P in the demosaicked image is considered to have noticeable change in color difference with respect to its most similar neighboring pixel I. Depending on the sign of the associated ψ, the pixel P could have

Fig. 15.   (a) Nonmosaicked original from Barbara image and demosaicked results obtained from (b) bilinear interpolation, (c) Freeman's method, (d) Hamilton's method, (e) Gunturk's method, and (f) the proposed method.

Fig. 16.   (a) Nonmosaicked original from Starburst image and demosaicked results obtained from (b) bilinear interpolation, (c) Freeman's method, (d) Hamilton's method, (e) Gunturk's method, and (f) the proposed method.

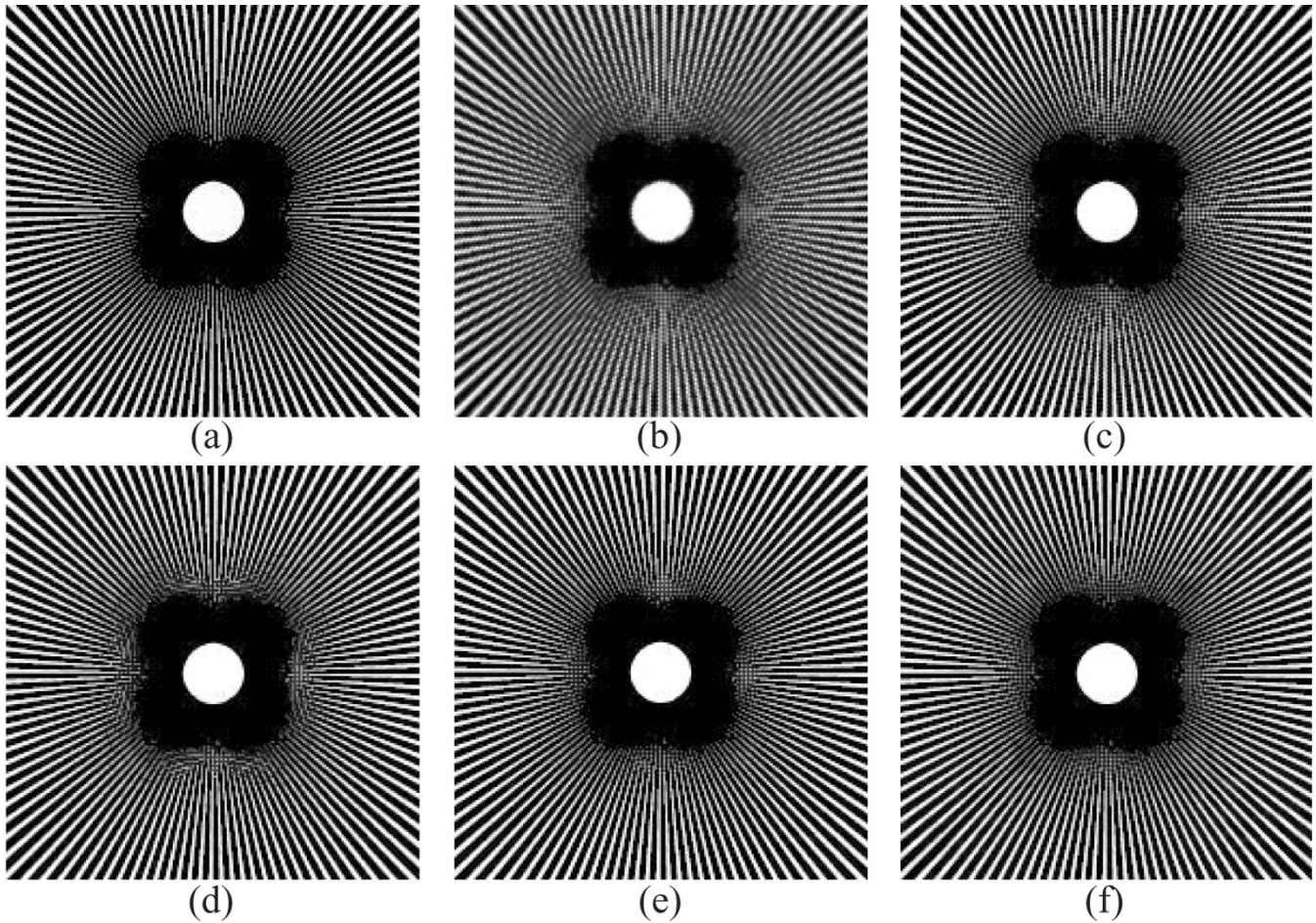reduced or increased color difference with respect to the pixel I. Specifically, $\psi < -\delta$ indicates that the pixel P has reduced contrast compared to the pixel I, while $\psi > \delta$ states that the zipper effect exists at the pixel P.
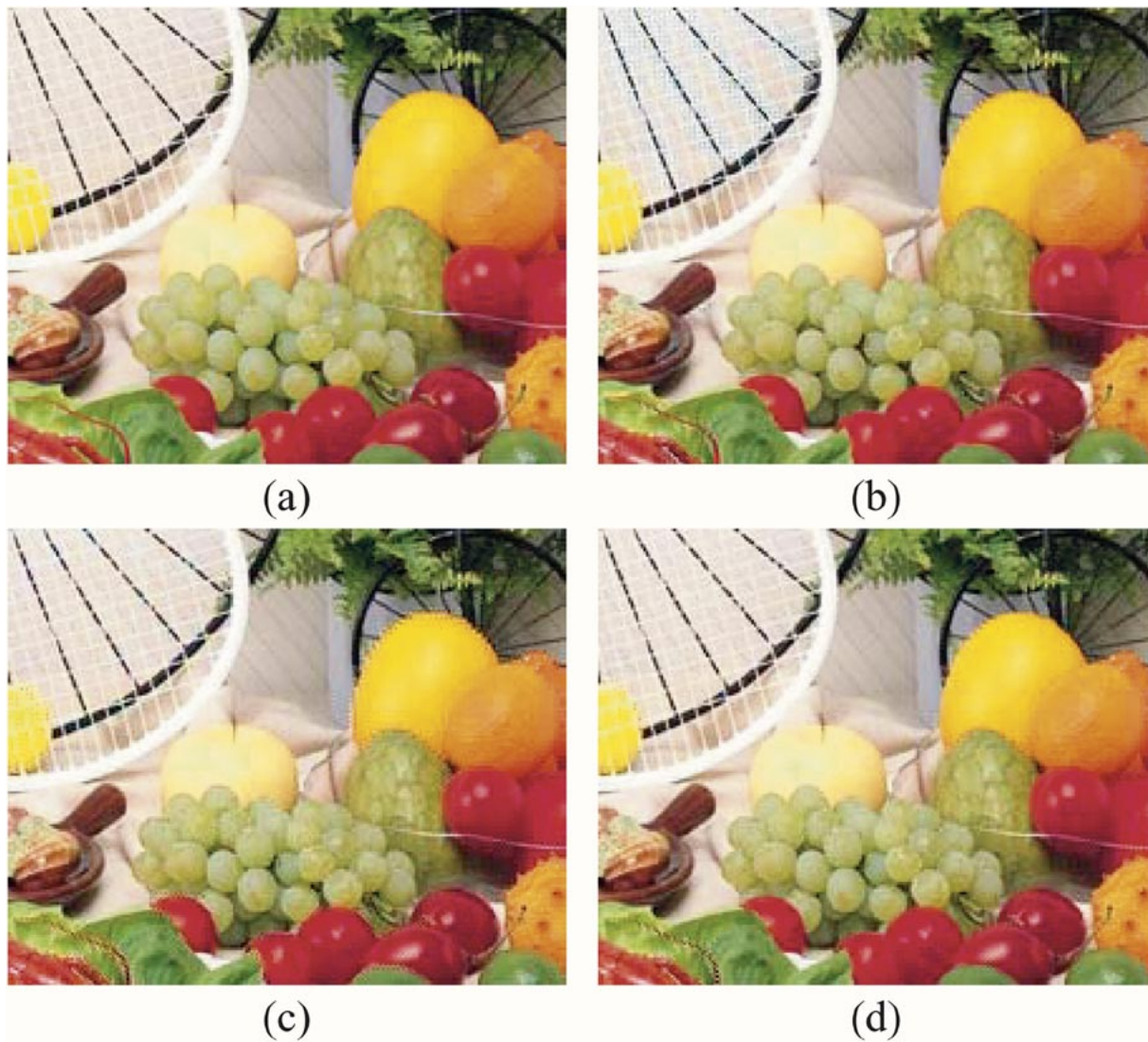
Fig. 17.   (a) Nonmosaicked original from Bike image and demosaicked results obtained from (b) Li's method, (c) Gunturk's method, and (d) the proposed method.

The proposed measure that quantifies the zipper effect residing in a demosaicked image is the percentage of pixels for which $\psi > \delta$. For the sake of completeness, we also examine the percentage of pixels having reduced contrast as compared to their neighboring pixels. Fig. 13 shows a nonmosaicked image and its several demosaicked results having different amounts of the zipper effect. It can be seen that the proposed measure is well correlated with the amount of the zipper effect that our eyes can perceive. For example, the second from the right demosaicked image in Fig. 13, which is obtained using a threshold-based variable gradient demosaicking method [10], has a lot of false colors as compared to the original, but it maintains good smoothness along the edges (i.e., little amount of the zipper effect). This observation is well captured by the proposed measure indicating only 11.22% pixels suffer from the zipper effect, which is the second lowest in the series of images compared in Fig. 13.

V. EXPERIMENTAL RESULTS

We demonstrate in this section the efficacy of our proposed method by comparing it with six existing demo-saicking methods—bilinear interpolation, Freeman's method [3], Hamilton's method [7], Kimmel's method [13], Li's method [14], and Gunturk's method [15]. Among the methods to be compared, bilinear interpolation is the simplest and also the most common reference for performance comparison. As reported in [22], [27] on comparing different demosaicking methods, both Freeman and Hamilton's methods are capable of producing more visually pleasing results than many others in terms of sharper reconstructed edges and less demosaicking artifacts. This is also supported by our experiments in comparing them with a number of other methods, including [1], [2], [5], [6], [9]–[12]. Kimmel's method is another one capable of producing good demosaicked results; it is similar to Freeman's method and our proposed method in the sense that all consist of two successive steps: a reconstruction step and an enhancement step. Li and Gunturk's methods are two recently proposed methods that can obtain superior results.

Two sets of test images are used in the experiments reported in this paper. Test Set I includes two standard test images (Barbara and Bike), four natural images from Kodak
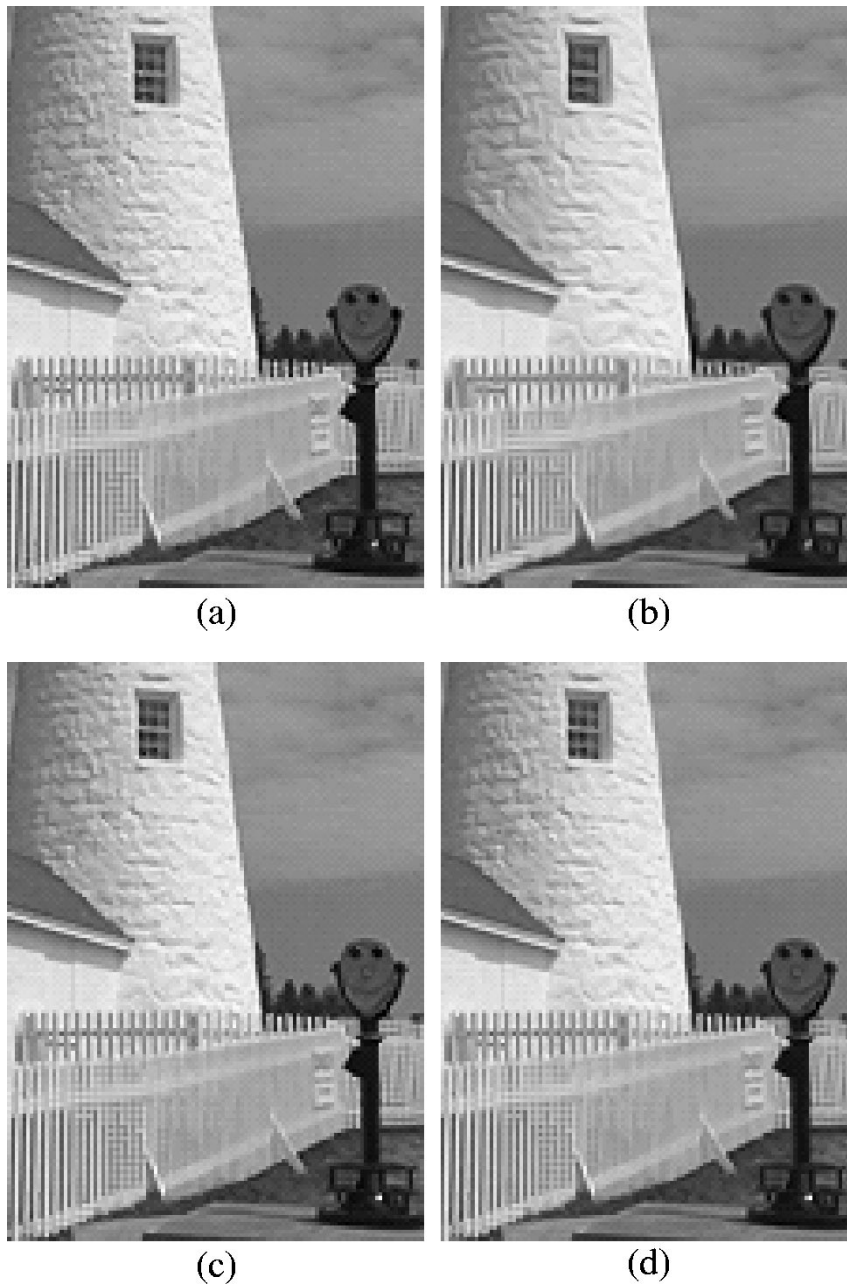
Fig. 18.   Demosaicked SmallLighthouse image of (a) the first step and (b) the second step of Kimmel's method, as well as that of (c) the interpolation step and (b) the post-processing step of our proposed method.

PhotoCD (Lighthouse, Window, Sails, and Brickhouse), and one synthetic images (Starburst). The Barbara and Bike test images contain many sharp edges and fine textures, while the four natural images have been used as test images for several demosaicking methods [10], [11], [13], [15]. The synthetic image is used to test the ability of demosaicking methods in handling edges of various orientations and spatial resolutions. Test Set I is used for comparing our proposed method with bilinear interpolation, Freeman, Hamilton, Li, and Gunturk's methods. For Gunturk's method, we make use of one-level (1-L) decomposition with eight projection iterations in our experiments as it generally performs better than two-level (2-L) decomposition [15]. Test Set II comprises the down-sampled versions of the Lighthouse, Window, and Sails test images,

which are used by Kimmel in [13] and will be referred to here-after as the SmallLighthouse, SmallWindow, and SmallSails images. Because Kimmel's method depends on several preset parameters and our implementation of the method may not be exactly the same as his, we shall compare our results of Test Set II directly with the demosaicked results made available by Kimmel at [28]. All the test images are shown in Fig. 14. They are sampled by the Bayer CFA pattern and then reconstructed using the demosaicking methods under comparison in RGB color space.

In Sections V-A and B, we will first present some demosaicked images generated by the different methods for visual comparison. To quantify the improvement made by the proposed method, we then tabulate the performance scores of each
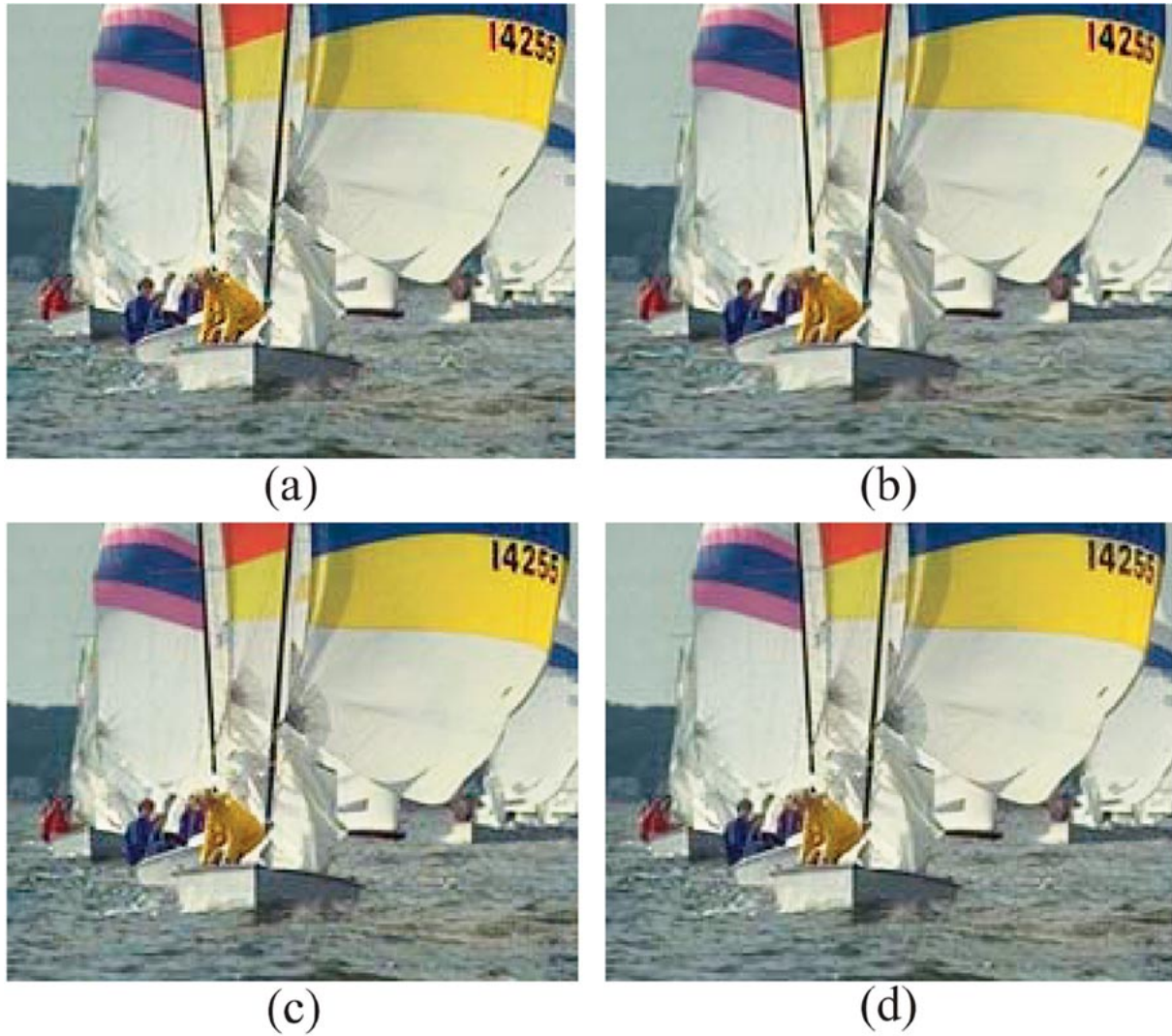
Fig. 19. Demosaicked SmallSails image of (a) the first step and (b) the second step of Kimmel's method, as well as that of (c) the interpolation step and (b) the post-processing step of our proposed method.

method based on the proposed image measures described in Section IV.

A. Visual Comparison

Figs. 15–17 show the sample demosaicked results produced by our proposed method and the other existing methods under comparison. It can be observed that our proposed method incurs much fewer false colors in high-frequency image regions, for example, the scarf in the Barbara image (Fig. 15), the central part of the Starburst image (Fig. 16), and the white racket in the Bike image (Fig. 17). Furthermore, compared to Gunturk's method, our proposed method can reconstruct sharp edges with little zipper effect, which can be seen by examining the boundaries of the fruits in the Bike image.

Figs. 18 and 19 present the demosaicked results of the SmallLighthouse and SmallSails images produced by the first and second steps of Kimmel's method and our proposed method, respectively. Compared to Kimmel's results, our demosaicked results have fewer artifacts around the fence region (Fig. 18) and the pole region (Fig. 19). Moreover, the second step of Kimmel's method oversmooths fine image details, such as the

lighthouse's texture (Fig. 18) and the water's surface (Fig. 19). In comparison, our post-processing step can well preserve these image details.

B. Quantitative Comparison Based on the Proposed Image Measures

The PSNR values computed for the edge and smooth regions of the demosaicked test images are listed in Table II. The table shows that our proposed interpolation step can reconstruct more accurate red, green and blue planes in most images than the other methods under comparison. Gunturk's method is the only method that can produce very comparable PSNR results. By comparing the PSNR values for the edge and smooth regions separately, we can see the improvement made by our proposed method around the edge regions, which are image contents crucial to the visual image quality. Because the post-processing step is only applied to artifact-prone regions (mainly the edge regions), the PSNR values obtained from the smooth regions of the initial and the post-processed demosaicked images are about the same. In some cases, as our post-processing step alters the original CFA-sampled color values, it leads to smaller

TABLE II
PROPOSED PSNR (dB) MEASURES FOR EDGE AND SMOOTH REGIONS OF DEMOSAICKED IMAGES. THE RESULTS OF THE RED, GREEN, AND BLUE PLANES FOR EACH IMAGE ARE SHOWN IN THE TOP, MIDDLE, AND BOTTOM ROWS, RESPECTIVELY. (A) RESULTS OF TEST SET I. (B) RESULTS OF TEST SET II

Image	Bilinear		Freeman		Hamilton		Li		Gunturk		Proposed Intp.		Post Processing	
	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth
Bike	18.6	29.4	24.8	32.1	26.3	34.8	26.6	30.3	28.2	33.4	29.4	35.7	29.5	35.7
	22.3	32.9	30.3	36.1	27.5	36.1	28.3	32.4	30.7	36.1	30.6	37.1	30.9	37.8
	18.6	28.6	24.6	30.9	25.9	33.1	26.2	29.4	27.9	32.4	28.9	33.8	29.4	33.8
Barbara	22.7	33.3	27.5	35.8	27.8	36.3	26.8	27.2	30.0	35.0	30.3	37.3	30.4	37.2
	27.1	37.9	34.3	41.4	30.2	39.8	30.5	30.1	34.6	38.7	33.3	41.6	34.3	41.6
	22.9	33.2	27.3	35.6	27.7	36.2	27.1	27.1	30.0	34.5	29.8	36.9	30.1	36.8
Window	28.1	40.2	35.1	44.3	37.0	44.6	35.1	42.0	39.1	44.1	38.6	45.9	38.0	45.7
	31.9	44.2	40.5	48.2	38.3	46.9	37.7	44.7	40.5	45.7	41.2	48.7	40.3	48.1
	28.2	39.8	34.9	43.7	36.8	43.7	35.6	41.9	36.7	41.9	38.2	45.1	37.6	44.8
Lighthouse	23.6	33.2	28.8	36.4	34.1	40.2	34.0	39.6	37.3	42.0	36.5	41.7	36.3	41.9
	28.6	37.9	36.8	42.2	35.6	42.2	36.3	42.2	40.5	45.1	38.3	43.9	38.6	44.3
	23.8	32.9	29.3	36.5	34.6	40.2	35.0	40.3	37.0	41.7	36.4	41.5	37.1	41.9
Sails	26.6	39.1	32.3	42.3	35.8	42.3	35.9	42.4	38.1	43.7	37.9	44.1	37.3	44.1
	30.9	42.5	39.3	46.5	37.7	44.5	37.9	44.8	40.7	46.4	40.1	46.1	39.1	45.9
	26.5	38.7	32.4	42.9	36.6	42.7	36.5	42.7	37.3	43.4	36.8	43.9	37.1	43.9
Brickhouse	23.0	29.6	28.5	34.8	31.7	35.2	32.4	35.5	36.0	38.6	34.2	37.9	36.0	39.3
	27.4	33.4	35.1	40.0	33.0	37.0	34.4	38.0	38.7	41.6	36.0	39.6	40.1	43.3
	23.0	30.0	28.6	35.2	31.7	35.5	32.8	36.3	35.6	38.6	34.5	38.2	36.4	39.7
Starburst	14.3	25.0	20.9	28.9	26.6	30.3	27.0	31.6	25.8	30.8	30.3	33.2	31.7	34.9
	17.9	29.3	28.3	34.6	28.9	32.2	29.5	33.8	28.4	33.4	30.1	34.3	32.2	35.9
	14.3	25.1	21.0	28.9	26.7	30.3	24.7	29.2	24.1	29.1	30.4	33.1	31.7	34.9

(a)

Image	Kimmel's Step 1		Kimmel's Step 2		Proposed Intp.		Post Processing	
	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth
SmallWindows	29.45	37.27	30.86	38.86	35.39	42.62	34.72	42.34
	31.59	40.24	32.24	41.29	37.34	45.98	36.09	44.81
	30.11	40.04	29.81	39.71	34.62	42.03	33.95	41.54
SmallLighthouse	30.19	37.61	29.14	36.11	32.54	37.55	32.98	38.08
	34.01	41.35	31.21	38.52	34.45	40.54	35.21	40.82
	28.92	35.16	28.29	34.97	32.86	37.94	33.96	38.60
SmallSails	31.43	41.33	30.73	40.99	34.54	43.00	33.86	42.87
	35.59	45.34	33.42	43.80	36.75	46.25	35.95	46.02
	31.92	43.97	30.92	42.04	33.69	43.69	33.85	43.70

(b)

PSNR values, especially for the edge regions. Such degradation in PSNR, however, is relatively imperceptible as compared to the demosaicking artifacts removed by the post-processing step. Therefore, evaluating demosaicking performance only based on the modified PSNR may not be sufficient to quantify the improvement made by different demosaicking methods. This is supported by the visual comparison given above and the results of other proposed image measures to be presented below.

When resorting to the $\triangle E_{ab}^*$ computed for the edge and smooth regions of Test Set I [see Table III(a)], the performance of the demosaicking methods under comparison is in the following order: the proposed post-processing step (the best), the proposed interpolation step or Gunturk's method, Li's method, Hamilton's method, Freeman's method, and bilinear interpolation (the worst). The $\triangle E_{ab}^*$ results computed for Test Set II also show that the proposed method outperforms Kimmel's method in both edge and smooth regions. For five out of the seven test images in Test Set I and two out of the three test images in Test Set II [see Table III(b)], the post-processing step further reduces the $\triangle E_{ab}^*$ for the edge regions of the initial demosaicked results obtained by the proposed interpolation method. One common characteristic of these test images is that they have many sharp edges and fine details. This is what we expected, as the post-processing step is designed to suppress visible demosaicking artifacts around edge regions.

Several insights into the existing demosaicking methods can be obtained by carefully examining the $\triangle E_{ab}^*$ listed in Table III. Comparing Hamilton's method with Freeman's method, we can see that the former mainly outperforms the latter around edge regions, while showing little difference in smooth regions. This is mainly due to the fact that Hamilton's method employs several edge classifiers to identify the 'best' interpolation direction so that the missing color values are estimated with adaptation to the edges present in a local neighborhood. On the other hand,

TABLE III
PROPOSED $\triangle E_{ab}^*$ MEASURES FOR EDGE AND SMOOTH REGIONS OF DEMOSAICKED IMAGES. (A) RESULTS OF TEST SET I. (B) RESULTS OF TEST SET II

Image	Bilinear		Freeman		Hamilton		Li		Gunturk		Proposed Intp.		Post Processing	
	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth
Bike	16.50	2.69	8.19	1.99	7.07	1.88	6.90	2.74	5.81	2.20	4.82	1.62	4.63	1.59
Barbara	11.66	2.79	6.84	2.29	6.54	2.31	6.64	3.88	5.60	2.77	5.48	2.11	5.41	2.13
Window	5.42	1.48	2.64	1.05	2.31	1.09	2.55	1.23	2.06	1.23	1.94	0.96	2.10	0.98
Lighthouse	9.74	2.63	5.05	1.78	3.49	1.74	3.24	1.66	2.45	1.46	2.78	1.52	2.68	1.50
Sails	7.11	1.92	3.65	1.31	2.66	1.49	2.66	1.35	2.23	1.25	2.37	1.29	2.40	1.29
Brickhouse	12.73	5.14	6.49	2.91	4.97	2.94	4.48	2.75	3.15	2.10	3.62	2.17	2.98	1.85
Starburst	24.92	2.20	8.41	1.28	4.49	1.17	4.39	1.40	2.56	1.16	3.03	0.86	1.41	0.62

(a)

Image	Kimmel's Step 1		Kimmel's Step 2		Proposed Intp.		Post Processing	
	Edge	Smooth	Edge	Smooth	Edge	Smooth	Edge	Smooth
SmallWindows	4.71	1.65	4.84	1.70	2.73	1.15	3.12	1.20
SmallLighthouse	4.62	2.08	5.07	2.21	3.65	1.89	3.27	1.81
SmallSails	3.70	1.31	3.95	1.32	3.34	1.22	3.32	1.22

(b)

TABLE IV
PERCENTAGE (%) OF PIXELS WITH THE ZIPPER EFFECT (ZE) AND WITH REDUCED CONTRAST (RC) TO THEIR NEIGHBORS IN DEMOSAICKED IMAGES. (A) RESULTS OF TEST SET I. (B) RESULTS OF TEST SET II

Image	Bilinear		Freeman		Hamilton		Li		Gunturk		Proposed Intp.		Post Processing	
	ZE	RC	ZE	RC	ZE	RC	ZE	RC	ZE	RC	ZE	RC	ZE	RC
Bike	41.6	0.5	24.7	1.2	22.3	1.2	27.3	1.6	17.7	2.2	12.3	1.9	10.2	2.9
Barbara	45.1	1.0	26.9	3.2	23.4	4.1	23.4	4.4	15.6	7.6	15.7	5.7	13.2	8.1
Window	24.9	0.2	9.8	0.5	7.9	0.4	8.6	0.7	5.4	0.8	3.9	0.5	3.8	0.7
Lighthouse	46.5	0.2	24.0	0.8	19.9	0.8	14.7	1.2	6.8	1.4	10.5	1.3	8.7	1.8
Sails	33.2	0.1	12.3	0.3	10.9	0.3	6.2	0.4	5.7	0.6	4.9	0.4	4.3	0.6
Brickhouse	70.0	0.2	44.1	0.8	39.5	1.0	28.2	1.4	12.5	1.8	20.1	1.6	10.4	1.9
Starburst	50.3	0.1	29.2	0.1	22.8	1.0	21.3	0.8	12.5	0.9	14.0	0.7	9.5	1.4

(a)

Image	Kimmel's Step 1		Kimmel's Step 2		Proposed Intp.		Post Processing	
	ZE	RC	ZE	RC	ZE	RC	ZE	RC
SmallWindow	16.2	1.3	13.8	1.8	8.1	1.5	7.5	1.3
SmallLighthouse	15.7	1.5	14.9	2.6	14.4	1.5	11.7	2.5
SmallSails	7.6	0.9	6.4	1.4	7.5	0.7	6.1	1.3

(b)

the performance of Freeman's method relies much on the size of the median filter's support, for which it is difficult to find an optimal value suitable for different image contents. For the Kimmel's method, its second step increases the $\triangle E_{ab}^*$ color differences for all the demosaicked images obtained by its first reconstruction step. This complies with the result of the above visual comparison: the second step of Kimmel's method suppresses demosaicking artifacts at the expense of fine image details.

Table IV lists the percentages of pixels with the zipper effect (ZE) and that with reduced color contrast (RC) with respect to their neighboring pixels. The table shows that for most of the test images in Test Set I, our proposed post-processing step consistently generates demosaicked images with the lowest amount of the zipper effect, followed by the proposed interpolation step or Gunturk's method, Li's method, Hamilton's method, Freeman's method and bilinear interpolation; for Test Set II, our proposed post-processing step outperforms Kimmel's method in suppressing the zipper effect. Note that while Gunturk's method is capable of producing demosaicked results with PSNR and $\triangle E_{ab}^*$ comparable to that obtained by our proposed method, it results in more severe zipper effect along sharp edges,

which can be seen around the fruits in the Bike image (Fig. 17). Such degradation in visual quality is well predicted by our proposed measure for the zipper effect.

It should be noted that the percentages of pixels with reduced contrast obtained by our proposed method (as well as Li and Gunturk's methods) are generally larger than that of bilinear interpolation, Freeman, and Hamilton's methods. However, when compared to the pixels where the zipper effect is suppressed by the proposed method, the pixels with reduced contrast not only make up a small percentage but also randomly scatter in the demosaicked images. Hence, these pixels have little adverse effect on the perceptual quality of the demosaicked images generated by our proposed method.

VI. CONCLUSION

In this paper, we have presented a new CFA demosaicking method that consists of two successive steps: an interpolation step that fills in missing color values in a progressive fashion by exploiting the spectral and spatial correlations among neighboring pixels, and a post-processing step that incorporates spectral correlation with median filtering of inter-channel differences to suppress demosaicking artifacts. To preserve the color fidelity of demosaicked images, the post-processing step is selectively applied to artifact-prone regions identified by a discrete Laplacian operator. When applied to the CFA samples of a variety of test images, the proposed method is able to generate demosaicked images with better perceptual quality compared to that produced by other existing methods.

Several image measures are also proposed to quantify the performance of different demosaicking methods. Considering CFA demosaicking as a reconstruction process, we make use of original, nonmosaicked images as baselines to compare the results of different demosaicked methods. For each test image, we compute the PSNR and CIELAB $\triangle E_{ab}^*$ values for the edge and smooth regions, separately. A new objective measure has also been devised to explicitly account for the zipper effect in demosaicked images. Experimental results show that the proposed image measures can effectively quantify the performance of different demosaicking methods.

ACKNOWLEDGMENT

REFERENCES

[1] D. Cok, "Signal Processing Method and Apparatus for Producing Interpolated Chrominance Values In a Sampled Color Image Signal," U.S. patent 4 642 678, 1987.
[2] J. Adams, "Interactions between color plane interpolation and other image processing functions in electronic photography," *Proc. SPIE*, vol. 2416, pp. 144–151, 1995.
[3] T. W. Freeman, "Median Filter for Reconstructing Missing Color Samples," U.S. Patent 4 724 395, 1988.
[4] J. Weldy, "Optimized design for a single-sensor color electronic camera system," *Proc. SPIE*, vol. 1071, pp. 300–307, 1990.
[5] H. Hibbard, "Apparatus and Method for Adaptively Interpolating a Full Color Image Utilizing Luminance Gradients," U.S. Patent 5 382 976, 1996.
[6] C. Laroche and M. Prescott, "Apparatus and Method for Adaptively Interpolating a Full Color Image Utilizing Chrominance Gradients," U.S. Patent 5 373 322, 1994.
[7] J. Hamilton and J. Adams, "Adaptive Color Plane Interpolation in Single Sensor Color Electronic Camera," U.S. Patent 5 629 734, 1997.
[8] J. Adams, "Design of practical color filter array interpolation algorithms for digital cameras," *Proc. SPIE*, vol. 3028, pp. 117–125, Feb. 1997.
[9] D. Cok, "Signal Processing Method and Apparatus for Sampled Image Signals," U.S. Patent 4 630 307, 1986.
[10] E. Chang, S. Cheung, and D. Pan, "Color filter array recovery using a threshold-based variable number of gradients," *Proc. SPIE*, vol. 3650, pp. 36–43, 1999.
[11] M. Gupta and T. Chen, "Vector color filter array demosaicing," *Proc. SPIE*, vol. 4306, 2001.
[12] S. C. Pei and I. K. Tam, "Effective color interpolation in CCD color filter array using signal correlation," in *IEEE Int. Conf Image Processing*, vol. 3, 2000.
[13] R. Kimmel, "Demosaicing: image reconstruction from color CCD samples," *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 1221–1228, 1999.
[14] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Processing*, vol. 10, no. 10, 2001.
[15] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Processing*, vol. 11, no. 9, 2002.
[16] B. Bayer, "Color Imaging Array," U.S. Patent 3 971 065, 1976.
[17] J. Adams, K. Parulski, and K. Spaulding, "Color processing in digital cameras," *Proc. IEEE*, vol. 18, no. 6, pp. 20–30, 1998.
[18] *MATLAB Version 6.5, Image Processing Toolbox User's Guide*, Math-Works, Inc., Natick, MA, 2002.
[19] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proc. IEEE*, vol. 78, no. 4, pp. 678–689, 1990.
[20] Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, 2002.
[21] B. Girod, "What's wrong with mean-squared error," in *Digital Images and Human Vision*, A. B. Watson, Ed.   Cambridge, MA: MIT Press, 1993.
[22] L. Philippe, X. Zhang, P. Delahunt, and D. Brainard, "Perceptual assessment of demosaicing algorithm performance," *Proc. IEEE*, vol. 90, no. 1, pp. 123–132, 2002.
[23] "Recommendations on Uniform Color Spaces, Color Difference Equations, Psychometric Color Terms," C. I. E, Supplement no. 2 to CIE publication no. 15(E.-1 31) 1971/(TC-1.3), 1978.
[24] M. D. Fairchild, *Color Appearance Models*.   Reading, MA: Addison Wesley, 1997.
[25] M. Mahy, E. Van, and O. A., "Evaluation of uniform color spaces developed after the adoption of CIELAB and CIELUV," *Color Res. Applicat.*, vol. 19, no. 2, pp. 105–121, 1994.
[26] A. Jain, *Fundamentals of Digital Image Processing*.   Englewood Cliffs, NJ: Prentice-Hall, 1989.
[27] R. Rajeev, E. Wesley, L. Griff, and S. William, "Demosaicking methods for Bayer color arrays," *J. Electron. Imag.*, 2002.
[28] [Online] Available: http://www.cs.technion.ac.il/~ ron/Demosaic/.

Wenmiao Lu (S'99) received the B.Eng. and M.Eng. degrees in electrical engineering from Nanyang Technological University, Singapore, in 1999 and 2003, respectively. He is currently a graduate student with the department of electrical engineering, Stanford University, CA.

His current research interests include image and video processing, pattern recognition, computer vision, and color imaging.

Yap-Peng Tan (M'97) received the B.S. degree in electrical engineering from National Taiwan University, Taiwan, R.O.C., in 1993, and the M.A. and Ph.D. degrees in electrical engineering form Princeton University, Princeton, NJ, in 1995 and 1997, respectively.

He was the recipient of an IBM Graduate Fellowship from IBM T. J. Watson Research Center, Yorktown Heights, NY, from 1995 to 1997, and was with Intel and at Sharp Labs of America from 1997 to 1999. Since November 1999, he has been a faculty member of Nanyang Technological University, Singapore. His current research interests include image and video processing, content-based multimedia analysis, computer vision, and pattern recognition. He is the principal inventor on five U.S. patents in the areas of image and video processing.