Four-Direction Residual Interpolation for Demosaicking

Yonghoon Kim and Jechang Jeong

***Abstract*— In this paper, we propose a four-direction residual interpolation (FDRI) method for color filter array interpolation. The proposed algorithm exploits a guided filtering process to generate the tentative image. The residual image is generated by exploiting the tentative and original images. We use an FDRI algorithm to more accurately estimate the missing pixel values; the estimated image is adaptively combined with a joint inverse gradient weight. Based on the experimental results, the proposed method provides a superior performance in terms of objective and subjective quality compared with the conventional state-of-the-art demosaicking methods.**

***Index Terms*— Bayer pattern, color filter array (CFA) interpolation, demosaicking, directional interpolation, residual interpolation (RI).**

I. Introduction

MOST image data use the RGB color format, which contains red, green, and blue data for each pixel position. A typical digital camera employs a single sensor covered with a color filter array (CFA); therefore, it captures all three color channels, but one at each pixel position. Using the obtained pixels, the missing color pixels must be interpolated to create a complete color image. This process of generating a full color image from the subsampled color image is called CFA interpolation or demosaicking.

The most common CFA pattern is the Bayer pattern [1], as shown in Fig. 1. Half of the pixels of the Bayer pattern are allocated to the green channel because it is more useful for representing the details and is less affected by aliasing. The key characteristic of the Bayer pattern is that each color pixel is aligned at a regular distance, which abets accurate interpolation using the correlation between color channels. The reconstruction process of the three color components is called CFA interpolation or demosaicking [2]. The simplest method of generating the missing pixels is to separately fill the missing pixels of each channel using a bilinear or bicubic interpolation algorithm, which is a spatially invariant interpolation method. These algorithms typically show good results in the homogenous region; however, they produce artifacts, such as aliasing or blurring, at the texture and edge structure.

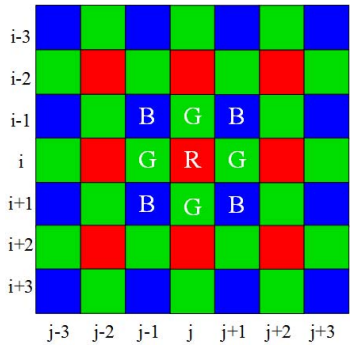


Fig. 1. Bayer CFA.

In the past few years, many demosaicking algorithms have been proposed to enhance interpolation performance and image quality. To preserve edge and texture, an edge-directed interpolation method generates the missing color components along the estimated interpolation directions by exploiting the spectral correlation of neighboring color components [3]–[12]. Zhang and Wu [4] proposed directional linear minimum mean square error estimation (DLMMSE) that uses a soft direction decision. It optimally combines color differences along horizontal and vertical directions. Paliy *et al.* [5] suggest a directional approach with scale-adaptive filtering based on local polynomial approximation (LPA). Laroche and Prescott [7], Malvar *et al.* [8], and Wu and Zhang [9] separately calculate the vertical and horizontal gradients of red and blue channels, which they use as correction terms to improve the green channel interpolation. Menon *et al.* [10] used color gradients over a local window for directional filtering with an *a posteriori* decision. In addition, a new approach has been proposed based on the finding that refinement of the green channel yields a higher quality image. Pekkucuksen and Altunbasak [11], [12] proposed an orientation-free edge strength filter (ESF) and multiscale gradient (MSG)-based CFA interpolation. Both algorithms utilize an accumulated gradient to calculate the weight for vertical and horizontal directions; moreover, they more accurately estimate edge direction and more efficiently use edge calculation [12] to noniteratively update the green channel.

Li *et al.* [13] report on the test data set. They claim that the Kodak data set [14] is not suitable for the test because its images acquired by digital cameras of an earlier period

Manuscript received October 20, 2014; revised February 27, 2015; accepted April 27, 2015. Date of publication May 1, 2015; date of current version May 3, 2016. This work was supported by the Ministry of Science, ICT and Future Planning, Korea, through the Project for Technical Development of Information Communication and Broadcasting Supervised within the Institute for Information and Communications Technology Promotion under Grant IITP 2015-B0101-15-1377. This paper was recommended by Associate Editor M. H. Hayes. *(Corresponding author: Jechang Jeong.)*

The authors are with the Department of Electronics and Computer Engineering, Hanyang University, Seoul 133-791, Korea (e-mail: charismakyh@gmail.com; jjeong@hanyang.ac.kr).

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

Digital Object Identifier 10.1109/TCSVT.2015.2428552

have different characteristics than those of today, whereas the McMaster data set contains images that are more similar to the current images. From that point, several algorithms have been proposed that focus on characteristics of the new data set [15]–[19]. Zhang *et al.* [15] proposed the local directional interpolation and nonlocal adaptive thresholding (LDI-NAT) algorithm that shows good performance on the McMaster data set but has very high complexity. Chen *et al.* [17] aim to adopt voting-based directional interpolation (VDI). In [18] and [19], residual interpolation (RI) techniques are introduced. The RI method exploits the characteristics of the guided filter [21] and shows significant performance improvement. However, these recently proposed VDI [17], RI [18], and minimized Laplacian RI (MLRI) [19] methods still produce severe demosaicking artifacts.

In this paper, we propose a four-direction RI (FDRI) algorithm, which is based on RI techniques. The remainder of this paper is structured as follows. In Section II, the concepts of RI and MLRI are given, and the proposed FDRI is introduced in Section III. The performance evaluation of the proposed algorithm is given in Section IV and Section V reports the conclusion.

II. Related Work

The proposed algorithm is based on the RI [18] and MLRI [19] algorithms. The guided filtering technique is exploited to calculate the residual data. The guided filter is a powerful edge-preserving filter and filter and has lower complexity than the bilateral filter; however, it provides better behavior near edges than the bilateral filter. Both filters require two reference images: 1) input image q and 2) guidance image I. The guided filter calculates the filtering output $\tilde{q}$ by considering the characteristics of the guidance image I, which itself can be the input image or a different image [21]. The guided filter derives the linear model from the guidance image and the target image. The linear model is given as follows:

$$q_i = a_k I_{i,} + b_k \quad \forall i \in \omega_k \tag{1}$$

where a_k and b_k denote the linear coefficients assumed to be constant in local window ω_k. To determine the coefficients, the cost function in the window is given as

$$E(a_k, b_k) = \sum_{i \in \omega_k} (a_k I_{i,} + b_k - q_i)^2 \tag{2}$$

and the solution is given as

$$\begin{aligned} a_k &= \frac{\frac{1}{|\omega|}\sum_{i\in\omega_k} I_i q_i - \mu_k \bar{q}_k}{\sigma_k^2} \\ b_k &= \bar{q}_k - a_k \mu_k \end{aligned} \tag{3}$$

where μ_k and σ_k^2 are the mean and variance of I in ω_k, $|\omega|$ is the number of pixels in ω_k, and $\bar{q}_k$ is the mean of q in ω_k. Equation (1) is not identical when it is computed in different windows. The simple solution is to average the coefficients as follows:

$$\tilde{q}_i = \bar{a}_k I_{i,} + \bar{b}_k \tag{4}$$

where $\bar{a}_k$ and $\bar{b}$, respectively, denote the average coefficients of a_k and b_k in ω_k, and $\tilde{q}_i$ is the filtered output. The guided filter can be used for denoising, dehazing, and guided feathering. Furthermore, it can be applied in the demosaicking area because it can transfer the structures of the guidance image to the filtering output. There have been several attempts to derive the color correlation in the demosaicking field using a guided filter. The first attempt was proposed in [19]. The RI exploits the guided filter to calculate the missing pixels using a different color plane as the input image q and the guidance image I; the filtering output is called the tentative image.

To improve the RI estimation performance, the use of MLRI is proposed. The motivation for using MLRI is that bilinear interpolation can provide better interpolation performance for images that have smaller Laplacian energies. To calculate coefficient a, Laplacian filtered input q^{Lap} and guidance image I^{Lap} are exploited, and b is calculated using original q and I. Therefore, (3) is rewritten as

$$\begin{aligned} a_k &= \frac{\frac{1}{|\omega|}\sum_{i\in\omega_k} I_i^{\text{Lap}} q_i^{\text{Lap}} - \mu_k^{\text{Lap}} \bar{q}_k^{\text{Lap}}}{\left(\sigma_k^{\text{Lap}}\right)^2} \\ b_k &= \bar{q}_k - a_k \mu_k \end{aligned} \tag{5}$$

where μ_k^{Lap} and $(\sigma_k^{\text{Lap}})^2$ are the mean and variance of I^{Lap} in ω_k, $|\omega|$ is the number of pixels in ω_k, and $\bar{q}_k^{\text{Lap}}$ is the mean of q^{Lap} in ω_k.

The outline of the RI and MLRI processes are given in Fig. 2. As shown in Fig. 2(a) and (b), to obtain the guided filtered G image, original G image and interpolated R image are required because R pixels do not exist at G pixel location. To interpolate R pixels at G pixel position, two-tap bilinear filter ([1/2 1/2]) is used. Using the output tentative image, the missing pixel is estimated by exploiting the difference, which is residual, between the originally captured G value and tentative estimates $\tilde{G}$. If $\tilde{G}^H$ is the horizontal tentative estimate of the G pixel at the R pixel position, then the target G pixel value is estimated as follows:

$$G_{i,j}^H = \tilde{G}_{i,j}^H + \frac{\left(G_{i,j-1} - \tilde{G}_{i,j-1}^H\right) + \left(G_{i,j+1} - \tilde{G}_{i,j+1}^H\right)}{2} \tag{6}$$

where G^H denotes the horizontal estimate of G. In the vertical case, it is given as

$$G_{i,j}^V = \tilde{G}_{i,j}^V + \frac{\left(G_{i,j-1} - \tilde{G}_{i,j-1}^V\right) + \left(G_{i,j+1} - \tilde{G}_{i,j+1}^V\right)}{2} \tag{7}$$

where G^V denotes the vertical estimates of G, and $\tilde{G}^V$ is the vertical tentative estimate of G. Using the residuals of vertical and horizontal neighbors, the target value is estimated; these directional estimates are adaptively combined using the directional weight. This method provides significant objective quality improvement; nevertheless, subjective quality remains insufficient. To solve this problem, we change the structure of the RI process and it will be discussed in the following section.

III. Proposed Algorithm

The proposed FDRI algorithm exploits the FDRI algorithm for the missing G pixel. The proposed algorithm requires only

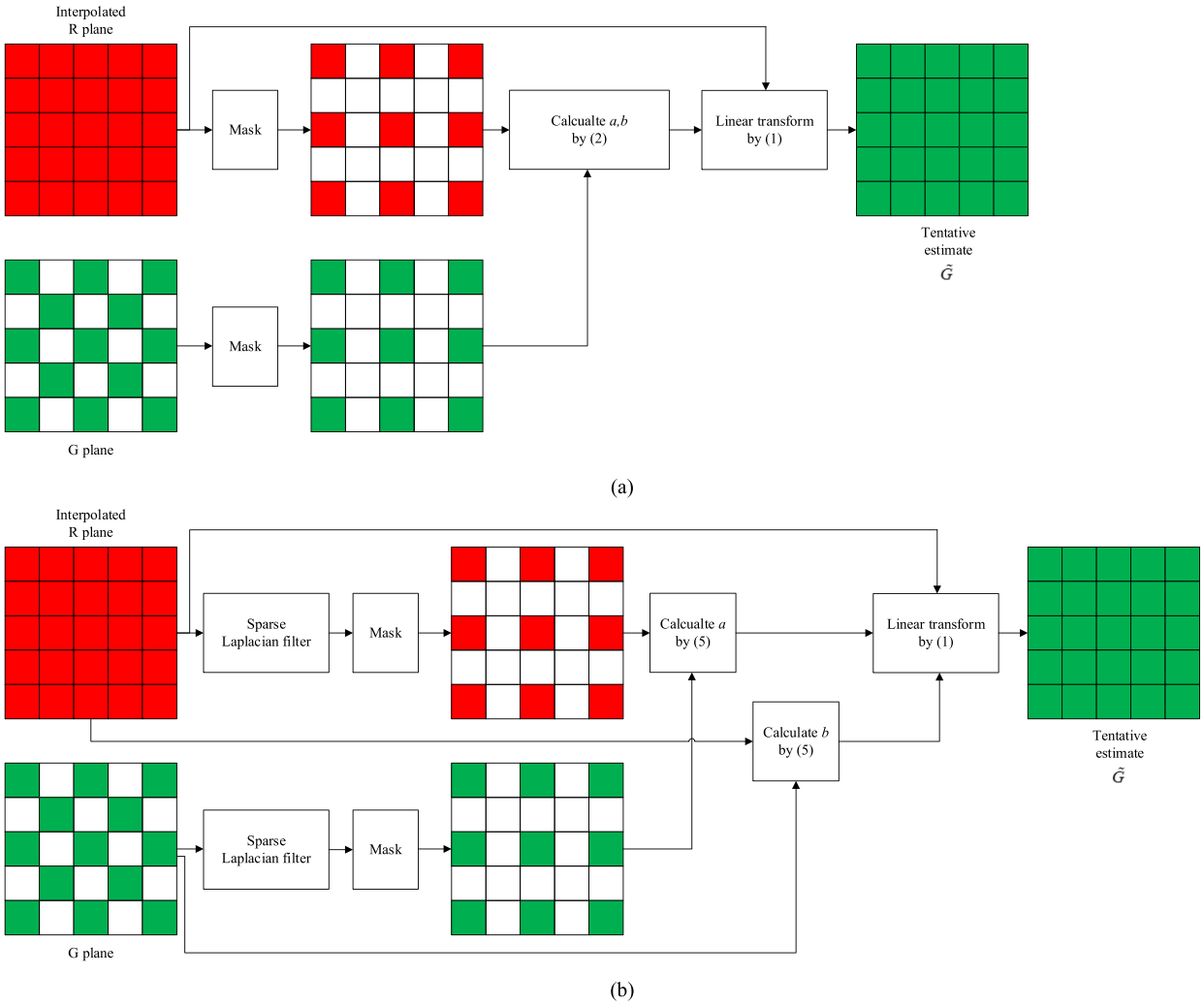


Fig. 2. Outline of the guided upsampling process of (a) RI and (b) MLRI.

a tentative G plane, whereas RI and MLRI require tentative G, R, and B planes for the G channel interpolation. The proposed algorithm interpolates the G plane using an estimated G and originally captured G plane; however, RI and MLRI utilize the vertical and horizontal color differences of G-R and G-B for directional estimation. The guided filtering processes of the proposed algorithm, RI, and MLIR are shown in Fig. 3. As shown in Fig. 3, the guided filtering value of the proposed algorithm is half of what RI and MLRI require for G channel interpolation.

To improve the guided filtering process, we use a different Laplacian filter with MLRI as shown in Table I. We determined that the 5×5 Laplacian filter, which considers diagonal neighbors, gives better estimation performance than the 1-D directional Laplacian filter used in MLRI.

A. Four Directional Weights Using Joint Gradient

In this paper, we use four directional interpolations. To more accurately combine the directional estimates, we derive the gradient from two intra-channels: 1) the original Bayer pattern plane (Z) and 2) the horizontally and vertically interpolated Bayer planes (Z^H, Z^V). The directionally interpolated Bayer plane is given as follows:

$$\begin{aligned} Z^H &= Z * f \\ Z^V &= Z * f^T \end{aligned} \tag{8}$$

where $f = [1/4,\ 1/2,\ -1/2,\ 1/2,\ 1/4]$, which is Hamilton and Adams' interpolation filter [20]. The directional gradients are given as

$$\begin{aligned} \Delta^H(i,j) &= |Z_{i,j-1} - Z_{i,j+1}| + \left|Z^H_{i,j+1} - Z^H_{i,j-1}\right| \\ \Delta^V(i,j) &= |Z_{i-1,j} - Z_{i+1,j}| + \left|Z^V_{i-1,j} - Z^V_{i+1,j}\right| \end{aligned} \tag{9}$$

where Δ^H and Δ^V represent the joint gradients of horizontal and vertical directions, respectively. In the Bayer pattern plane, only one gradient can be calculated at a pixel position. However, using the interpolated Bayer plane, we can obtain a gradient of a different color plane at the same pixel position. The directional weights are made by inversing the local sum of gradients. Using (9), the four directional weights are calculated as

$$w^N_{i,j} = \sum_{a=-1}^{1} \sum_{b=-1}^{1} \Delta^V(i-1+a, j+b)$$

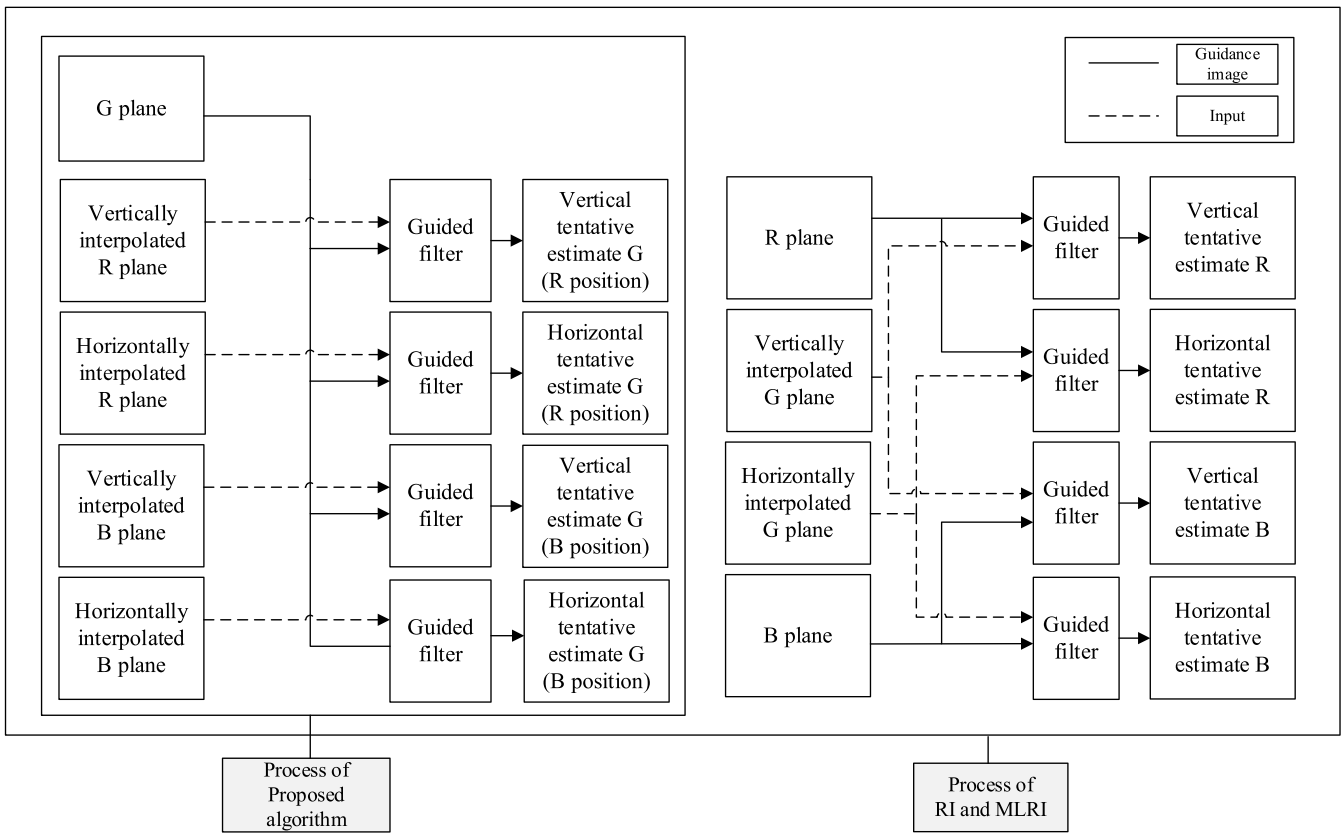


Fig. 3. Guided filtering process required for G plan interpolation of the proposed algorithm, RI, and MLRI.

$$w^S_{i,j} = \sum_{a=-1}^{1}\sum_{b=-1}^{1} \Delta^V(i+1+a, j+b)$$
$$w^E_{i,j} = \sum_{a=-1}^{1}\sum_{b=-1}^{1} \Delta^H(i+a, j-1+b)$$
$$w^W_{i,j} = \sum_{a=-1}^{1}\sum_{b=-1}^{1} \Delta^H(i+a, j+1+b) \tag{10}$$

where $w^N_{i,j}$, $w^S_{i,j}$, $w^E_{i,j}$, and $w^W_{i,j}$ denote the weights of north, south, east, and west, respectively.

B. Green Channel Interpolation

Using the process represented in Fig. 2(b), we obtain the vertical and horizontal tentative images. Using the tentative images, we estimate the G value in four directions. The proposed FDRI is given as follows:

$$\begin{cases} G^N_{i,j} = \tilde{G}^V_{i,j} - \left(\tilde{G}^V_{i-1,j} - G_{i-1,j}\right) \\ G^S_{i,j} = \tilde{G}^V_{i,j} - \left(\tilde{G}^V_{i+1,j} - G_{i+1,j}\right) \\ G^E_{i,j} = \tilde{G}^H_{i,j} - \left(\tilde{G}^H_{i,j-1} - G_{i,j-1}\right) \\ G^W_{i,j} = \tilde{G}^H_{i,j} - \left(\tilde{G}^H_{i,j+1} - G_{i,j+1}\right) \end{cases} \tag{11}$$

where $G^N_{i,j}$, $G^S_{i,j}$, $G^E_{i,j}$, and $G^W_{i,j}$ denote the directional estimates of G in the north, south, east, and west directions, respectively. These values and the original G values are adaptively combined using the weight in (10) as

$$\begin{aligned} G' = \alpha \times \left(w^N_{i,j}G^N_{i,j} + w^S_{i,j}G^S_{i,j} + w^E_{i,j}G^E_{i,j}\right. \\ \left. + w^W_{i,j}G^W_{i,j}\right) + (1-\alpha) \\ \times\left(w^N_{i,j}G_{i-1,j} + w^S_{i,j}G_{i+1,j}\right. \\ \left. + w^E_{i,j}G_{i,j-1} + w^W_{i,j}G_{i,j+1}\right) \end{aligned} \tag{12}$$

where G' represents the reconstructed G pixel and α is experimentally determined as 0.7. We expected this combination to

TABLE I
SPARSE LAPLACIAN FILTER EXPLOITED IN MLRI AND THE PROPOSED ALGORITHM

	MLRI	Proposed algorithm
Green channel interpolation (horizontal)	$\begin{pmatrix}1 & 0 & -2 & 0 & 1\end{pmatrix}$	$\begin{pmatrix}1 & 0 & 1 & 0 & 1\\ 0 & 0 & 0 & 0 & 0\\ 1 & 0 & -8 & 0 & 1\\ 0 & 0 & 0 & 0 & 0\\ 1 & 0 & 1 & 0 & 1\end{pmatrix}$
Green channel interpolation (vertical)	$\begin{pmatrix}1\\0\\-2\\0\\1\end{pmatrix}$	
Red and blue channel interpolation	$\begin{pmatrix}0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 1 & 0 & -4 & 0 & 1\\ 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 0 & 1 & 0 & 1\\ 0 & 0 & 0 & 0 & 0\\ 1 & 0 & -8 & 0 & 1\\ 0 & 0 & 0 & 0 & 0\\ 1 & 0 & 1 & 0 & 1\end{pmatrix}$

TABLE II
CPSNR COMPARISON WITH 18 MCMASTER IMAGES (DECIBELS)

No.	ESF	DLMMSE	MSG	LPA	EDAEP	VDI	LDI-NAT	RI	MLRI	FDRI
1	26.38	26.98	27.05	26.82	27.60	28.01	29.01	29.12	28.97	**29.38**
2	33.48	33.68	33.67	33.84	33.99	34.18	35.01	35.00	35.08	**35.16**
3	32.56	32.59	32.93	32.47	32.07	32.64	32.56	33.75	**33.86**	33.54
4	34.97	34.32	35.49	34.94	34.36	36.00	35.94	37.86	37.64	**38.06**
5	30.64	31.27	31.12	31.39	32.10	32.63	34.09	33.91	33.99	**34.83**
6	32.57	33.84	33.56	34.37	35.03	35.63	37.87	38.29	38.26	**38.91**
7	39.10	38.64	39.17	**39.20**	36.22	36.03	35.96	36.95	37.46	35.85
8	37.85	37.45	37.61	37.88	37.12	37.41	37.46	36.96	36.94	**38.17**
9	34.39	34.41	34.69	35.07	35.23	35.95	36.91	35.92	36.44	**37.54**
10	35.78	36.34	36.47	37.02	37.01	37.28	38.72	38.15	38.62	**39.02**
11	36.61	37.25	37.28	37.72	37.83	37.98	39.48	39.43	39.91	**40.02**
12	36.16	36.60	36.80	36.97	37.16	37.09	38.89	39.61	39.64	**39.95**
13	38.67	38.79	38.83	39.29	39.34	39.40	40.78	40.27	40.51	**41.01**
14	37.21	37.23	37.13	37.54	37.65	37.32	38.67	38.92	38.76	**39.32**
15	37.01	37.27	37.19	37.66	37.76	37.86	38.93	38.37	38.91	**39.45**
16	29.24	30.46	30.18	29.48	31.39	31.40	33.52	35.17	35.09	**35.42**
17	28.57	29.31	29.30	29.27	30.59	31.15	32.83	32.44	32.58	**33.76**
18	33.69	33.92	34.10	33.96	34.05	34.23	35.00	**36.52**	36.12	36.21
average	34.16	34.46	34.59	34.72	34.81	35.12	36.20	36.48	36.60	**36.98**

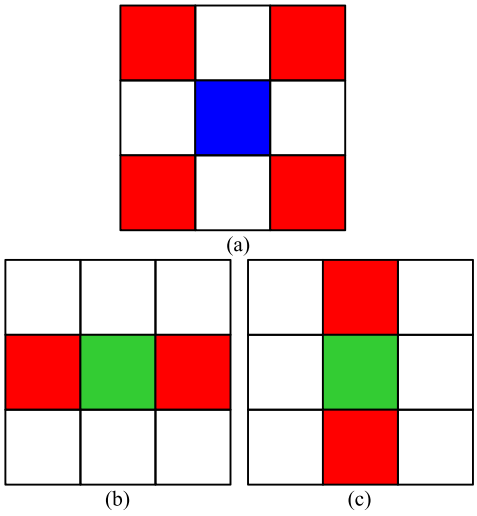


Fig. 4. Three different available reference pixels based on the pixel position for red pixel interpolation. (a) Diagonal. (b) Horizontal. (c) Vertical.

Fig. 5. 18 McMaster test images.

be more accurate than the combination without the original pixel because the original G pixel can correct the misguided estimates.

C. Red and Blue Channel Interpolation

After the G pixels are fully reconstructed, the R and B planes are interpolated using a similar method to the G plane interpolation. For the R plane, the reconstructed G plane can be the input image, and the R plane serves as the guidance image. At this stage, we do not apply a directional weight. The residual is given as follows:

$$\Gamma(i, j) = \tilde{R}(i, j) - R(i, j) \tag{13}$$

where $\tilde{R}$ is the tentative estimates of R, and Γ denotes the residual. These residual data can be combined based on the pixel position, as shown in Fig. 4. At the B pixel position in Fig. 4(a), only diagonal references are available; therefore, the interpolation process of R is given as

$$R'(i, j) = \tilde{R}(i, j) - \frac{\Gamma(i-1, j-1)+\Gamma(i-1, j+1)+\Gamma(i+1, j-1)+\Gamma(i+1, j+1)}{4} \tag{14}$$

where R' is the reconstructed R pixel. The interpolation processes of R at the G pixel position are calculated as

$$R'(i, j) = \tilde{R}(i, j) - \frac{\Gamma(i, j-1) + \Gamma(i, j+1)}{2} \tag{15}$$

$$R'(i, j) = \tilde{R}(i, j) - \frac{\Gamma(i-1, j) + \Gamma(i+1, j)}{2}. \tag{16}$$

Equation (15) is exploited for the horizontal case in Fig. 4(b) and (16) is exploited for the vertical case in Fig. 4(c). The interpolation process for the missing B pixels can be achieved in the same method by exchanging the roles of R and B, as explained above. After the interpolation process, we obtain the full three color images.

TABLE III
S-CIELAB COMPARISON WITH 18 McMASTER IMAGES

No.	ESF	DLMMSE	MSG	LPA	EDAEP	VDI	LDI-NAT	RI	MLRI	FDRI
1	3.784	3.204	3.170	3.414	2.903	2.620	2.495	2.486	2.517	**2.381**
2	1.453	1.315	1.252	1.267	1.246	1.144	1.087	1.016	1.035	**0.987**
3	2.076	2.083	1.825	2.103	2.290	1.893	1.854	1.592	1.621	**1.468**
4	1.461	1.443	1.086	1.169	1.484	1.113	1.000	**0.661**	0.781	0.723
5	1.771	1.615	1.551	1.518	1.526	1.374	1.222	1.158	1.165	**1.100**
6	1.857	1.511	1.502	1.410	1.278	1.127	0.965	0.898	0.910	**0.832**
7	0.944	0.941	**0.897**	0.877	1.228	1.199	1.173	1.119	1.076	1.229
8	0.659	0.631	0.589	0.584	0.723	0.613	0.586	0.578	0.575	**0.565**
9	1.396	1.377	1.203	1.199	1.305	1.058	0.964	0.976	1.006	**0.871**
10	1.192	1.083	1.025	0.968	0.999	0.932	0.837	0.838	0.795	**0.768**
11	0.907	0.793	0.783	0.745	0.725	0.675	0.604	0.624	0.611	**0.582**
12	1.257	1.089	1.081	1.057	0.940	0.956	0.824	0.746	0.731	**0.727**
13	0.739	0.711	0.692	0.667	0.683	0.650	0.582	0.577	0.579	**0.559**
14	0.860	0.797	0.784	0.758	0.789	0.742	0.691	0.664	0.680	**0.644**
15	0.913	0.855	0.836	0.807	0.819	0.773	0.719	0.745	0.719	**0.676**
16	2.839	2.064	2.317	2.594	1.673	1.499	1.391	1.365	1.383	**1.365**
17	3.090	2.826	2.577	2.757	2.290	1.960	1.826	1.797	1.827	**1.521**
18	1.601	1.510	1.460	1.506	1.451	1.287	1.234	**1.063**	1.168	1.138
average	1.600	1.436	1.368	1.411	1.353	1.201	1.114	1.050	1.065	**1.008**

TABLE IV
FSIMc COMPARISON WITH 18 McMASTER IMAGES

No.	ESF	DLMMSE	MSG	LPA	EDAEP	VDI	LDI-NAT	RI	MLRI	FDRI
1	0.9906	0.9938	0.9924	0.9918	0.9927	0.9935	**0.9956**	0.9952	0.9954	0.9955
2	0.9959	0.9970	0.9964	0.9966	0.9966	0.9968	0.9977	0.9976	0.9976	**0.9977**
3	0.9973	0.9986	0.9978	0.9975	0.9970	0.9975	0.9986	0.9989	**0.9990**	0.9988
4	0.9985	0.9992	0.9988	0.9986	0.9982	0.9987	0.9993	0.9995	0.9996	**0.9996**
5	0.9960	0.9974	0.9967	0.9969	0.9971	0.9974	0.9986	0.9986	0.9986	**0.9987**
6	0.9971	0.9986	0.9980	0.9983	0.9984	0.9985	**0.9993**	0.9991	0.9992	0.9992
7	0.9987	**0.9990**	0.9989	0.9988	0.9978	0.9978	0.9984	0.9988	0.9989	0.9985
8	0.9978	0.9981	0.9979	0.9981	0.9979	0.9981	0.9985	0.9976	0.9976	**0.9987**
9	0.9968	0.9975	0.9973	0.9976	0.9976	0.9978	0.9986	0.9976	0.9981	**0.9986**
10	0.9981	0.9984	0.9985	0.9986	0.9984	0.9985	0.9991	0.9988	0.9990	**0.9991**
11	0.9980	0.9984	0.9984	0.9985	0.9984	0.9984	**0.9991**	0.9986	0.9989	0.9990
12	0.9981	0.9985	0.9984	0.9985	0.9984	0.9983	0.9990	0.9991	0.9991	**0.9992**
13	0.9977	0.9982	0.9979	0.9982	0.9983	0.9983	0.9989	0.9989	0.9989	**0.9991**
14	0.9977	0.9983	0.9980	0.9982	0.9983	0.9985	0.9989	0.9988	0.9988	**0.9990**
15	0.9977	0.9980	0.9980	0.9981	0.9980	0.9981	0.9985	0.9979	0.9984	**0.9985**
16	0.9955	0.9979	0.9969	0.9961	0.9977	0.9977	**0.9987**	0.9986	0.9985	0.9986
17	0.9937	0.9959	0.9952	0.9949	0.9958	0.9958	**0.9979**	0.9966	0.9973	0.9975
18	0.9976	0.9983	0.9979	0.9978	0.9977	0.9980	0.9985	**0.9989**	0.9988	0.9988
average	0.9968	0.9980	0.9974	0.9974	0.9975	0.9977	0.9985	0.9983	0.9984	**0.9986**

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed algorithm, we tested it on 18 McMaster data set images [22] with a resolution of 500 × 500, as listed in Fig. 5. We chose the McMaster data set because the images have lower spectral correlations and are similar to nature images captured by color sensors. The widely used Kodak data set is outdated because the images are scanned from film-based photographs and their characteristics do not match those of the current images [13]. Therefore, the Kodak data set was not suitable for a fair demosaicking performance comparison. The proposed FDRI algorithm was compared with ESF [11], DLMMSE [8], MSG [12], LPA [5], effective demosaicking algorithm based on edge property (EDAEP) [16], VDI [17], LDI-NAT [15], RI [18], and MLRI [19].

To simulate the proposed algorithm, we conducted experiments using MATLAB with an Intel Core i7-4770k and 3.5-GHz CPU processer. To avoid the boundary effects, every calculation excluded the border of 10 pixels around the image. The numerical results are summarized in Table II for each of the 18 test images. The results are in ascending order of average CPSNR value which denotes average PSNR value of three color channel; the highest CPSNR value is in bold. The S-CIELAB results, which show perceptual color fidelity, are given in Table III; its lowest value is likewise in bold. It measures how close the reconstructed color is to the original color when viewed by a human observer [23]. To further evaluate the performance of the proposed method, we adopted another measure, the feature similarity index for image quality assessment (FSIMc) [24]. This model considers image distortion, such as gradient correlation, luminance distortion, and contrast distortion; when the value of FSIMc is close to 1, the reconstructed image is similar to the original image. In Table IV, the values closest to 1 are marked in bold.

As shown by information in Table II, the proposed algorithm outperformed the other algorithms on 15 out of 18 images. Its

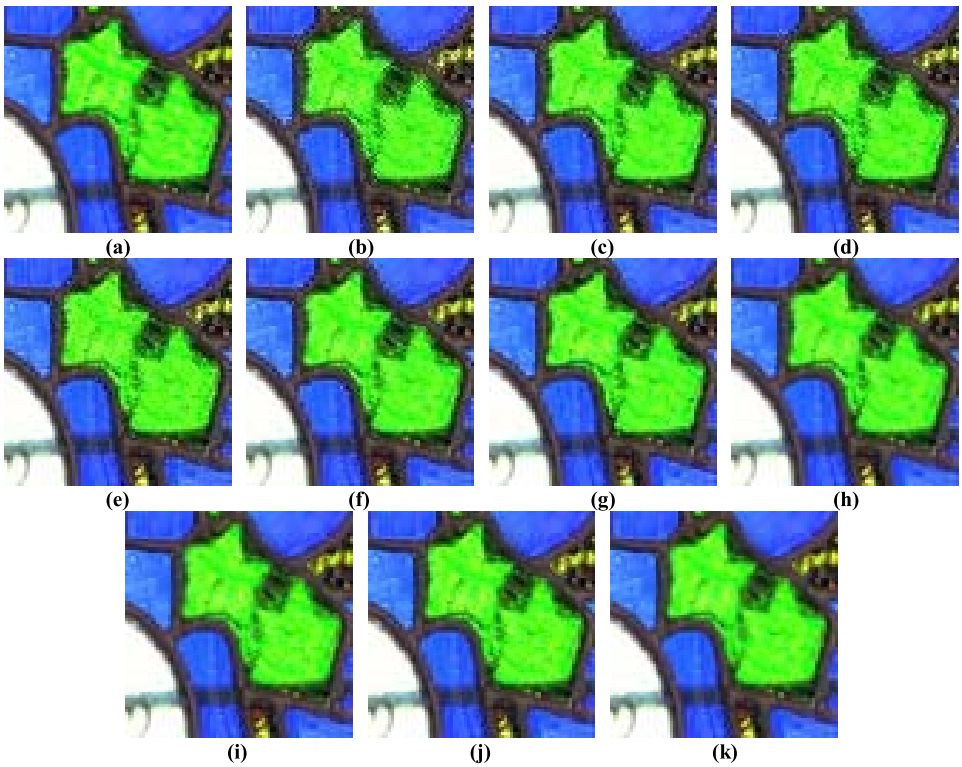


Fig. 6. Demosaicking results of a cropped region of Image 1. (a) Original image. (b) ESF. (c) DLMMSE. (d) MSG. (e) LPA. (f) EDAEP. (g) VDI. (h) LDI-NAT. (i) RI. (j) MLRI. (k) FDRI.

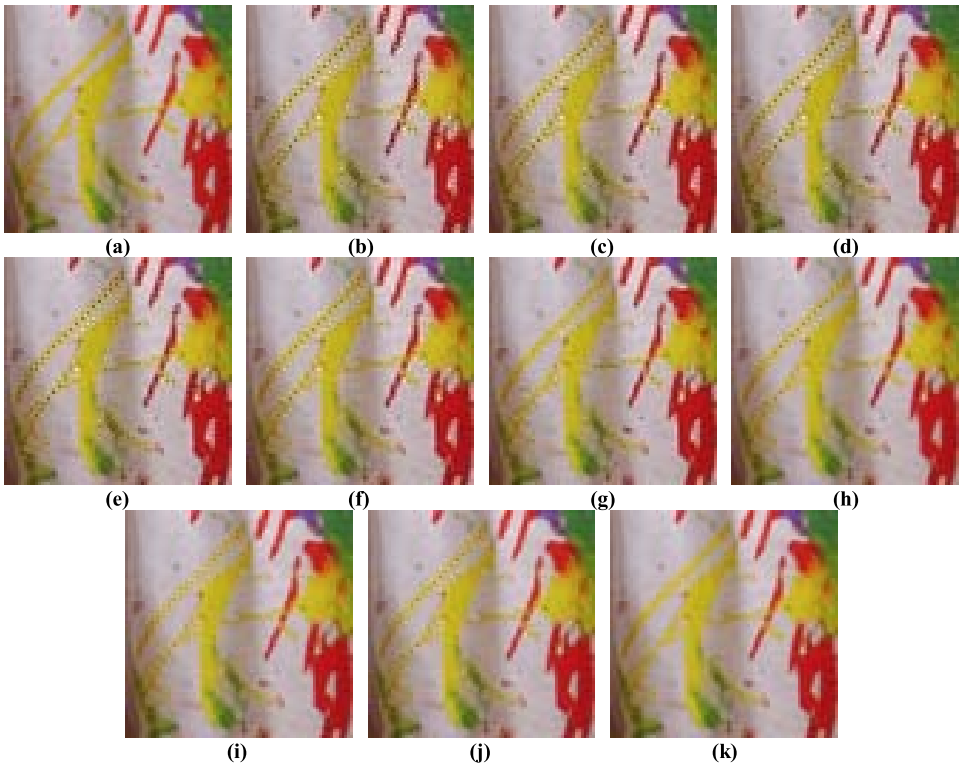


Fig. 7. Demosaicking results of a cropped region of Image 5. (a) Original image. (b) ESF. (c) DLMMSE. (d) MSG. (e) LPA. (f) EDAEP. (g) VDI. (h) LDI-NAT. (i) RI. (j) MLRI. (k) FDRI.

average CPSNR was higher than the closest method (MLRI) by 0.38 dB and RI by 0.50 dB. The S-CIELAB measure likewise showed similar results, as presented in Table III. The proposed method outperformed the other methods on 15 out of 18 images. It can be seen that MLRI gave a lower performance than RI in terms of the S-CIELAB measure despite it having higher average CPSNR performance. However, the proposed algorithm, which uses the same guided filtering method as

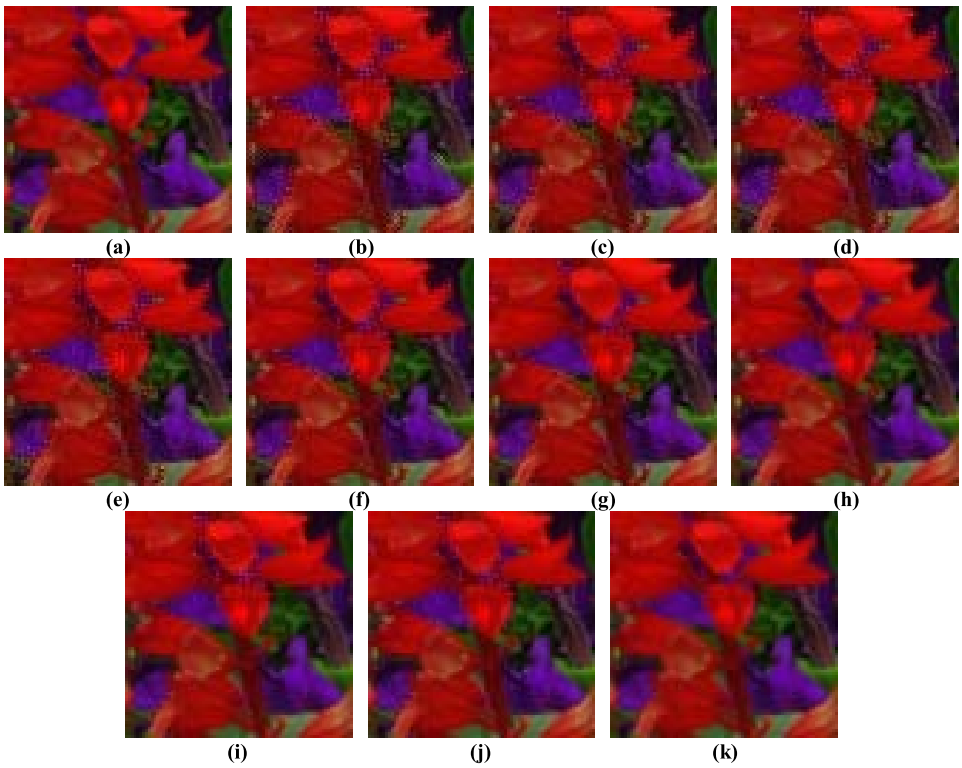


Fig. 8. Demosaicking results of a cropped region of Image 17. (a) Original image. (b) ESF. (c) DLMMSE. (d) MSG. (e) LPA. (f) EDAEP. (g) VDI. (h) LDI-NAT. (i) RI. (j) MLRI. (k) FDRI.

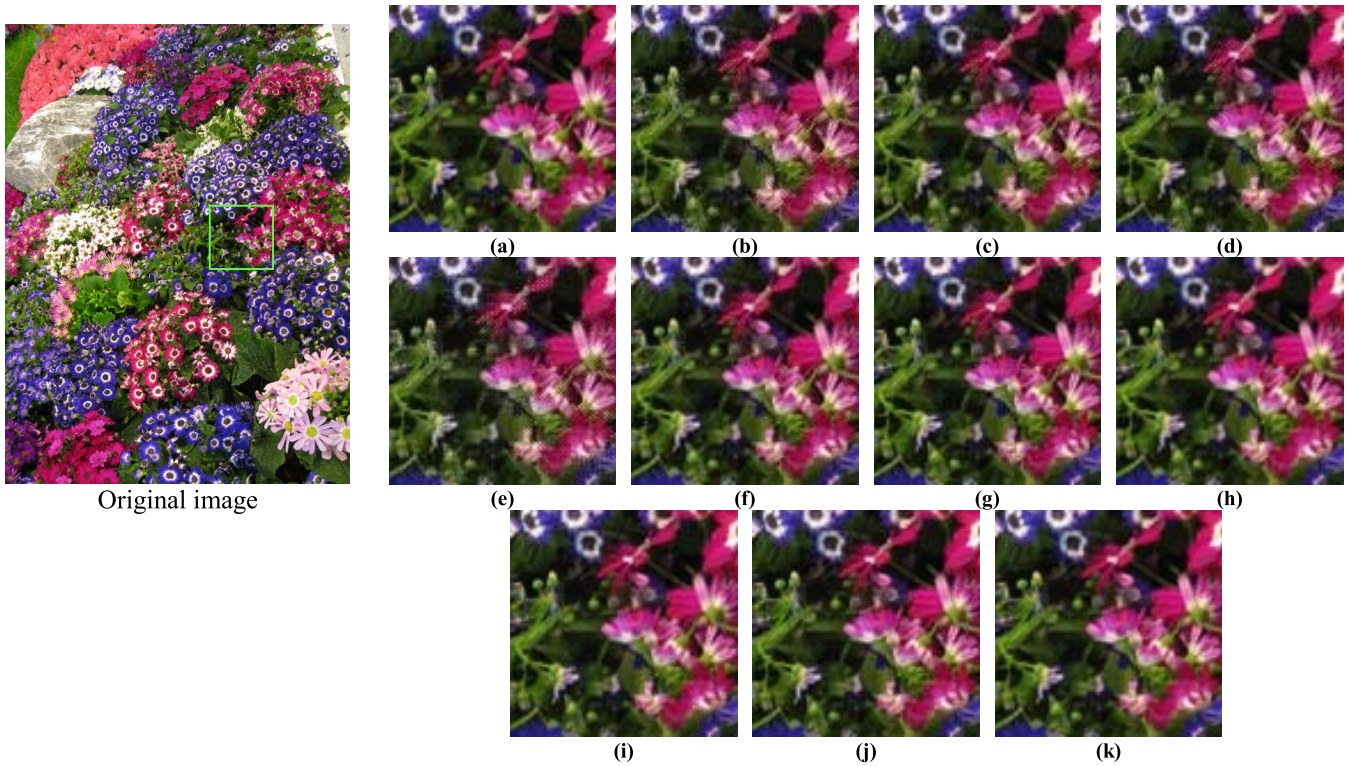


Fig. 9. Demosaicking results of a cropped region. (a) Original image. (b) ESF (25.18 dB). (c) DLMMSE (26.52 dB). (d) MSG (26.28 dB). (e) LPA (24.97 dB). (f) EDAEP (27.29 dB). (g) VDI (27.47 dB). (h) LDI-NAT (28.38 dB). (i) RI (28.50 dB). (j) MLRI (28.24 dB). (k) FDRI (28.71 dB).

MLRI, was superior to RI. This indicates that the proposed four-direction residual approach is effective and robust for a demosaicking application.

To further evaluate the performance of the proposed method, we computed the FSIMc values for each of the compared methods using the McMaster image data set. The results

Fig. 10. Demosaicking results of a cropped region. (a) Original image. (b) ESF (21.34 dB). (c) DLMMSE (22.39 dB). (d) MSG (22.36 dB). (e) LPA (22.71 dB). (f) EDAEP (23.33 dB). (g) VDI (23.57 dB). (h) LDI-NAT (24.36 dB). (i) RI (24.43 dB). (j) MLRI (24.11 dB). (k) FDRI (24.69 dB).

are outlined in Table IV. In terms of FSIMc, the proposed FDRI demonstrated the best performance, and LDI-NAT gave the second best performance. In terms of individual images, FDRI outperformed the other algorithms on 10 out of 18 images and LDI-NAT outperformed the other methods on 5 out of 18 images.

To compare subjective quality, Figs. 6–8 show the partially zoomed-in view images: images 1, 5, and 17. As shown in Fig. 6, the proposed algorithm greatly improved interpolation performance. The shirt region of Image 5 is shown in Fig. 7; the proposed algorithm perfectly removed the false coloring artifact. Our strategy of exploiting joint weight and flexible interpolation techniques has the advantage of removing demosaicking artifacts. Fig. 8 additionally shows that the proposed algorithm outperformed the other algorithms, and LDI-NAT gave a fair performance. From the subjective results, ESF, LPA, and MSG, which were focused on the Kodak data set, showed significant quality degradation. VDI, RI, and MLRI gave good CPSNR results; however, they showed relatively lower subjective image quality. Aside from FDRI, the other algorithms showed various demosaicking artifacts. It is evident that the proposed FDRI strategy successfully worked and produced clean images without demosaicking artifacts.

For further evaluation, we tested demosaicking algorithms on two images included in *LC* images that are captured using Cannon digital camera [25]. The subjective results are shown in Figs. 9 and 10. In Fig. 9, the proposed FDRI gives the best results without any demosaicking artifacts. ESF, DLMMSE, MSG, and LPA give the severe demosaicking artifact, and EDAEP, VDI, LDI-NAT, RI and MLRI show relatively small artifacts but they are still noticeable. As shown in Fig. 10, most demosaicking algorithms show severe artifacts at the boundary and they cannot interpolate the diagonal edge properly; however, FDRI shows a few demosaicking artifacts and clearly generate diagonal edge.

TABLE V
COMPLEXITY COMPARISON OF RI, MLRI, AND FDRI

	RI	MLRI	FDRI
Guided filtering (*G* channel interpolation)	8	8	4
Guided filtering (*R* and *B* channel interpolation)	2	2	2
Laplacian filtering	0	18	8
Residual calculation	10	10	6

TABLE VI
AVERAGE COMPUTATIONAL TIME COMPARISON ON McMASTER IMAGE (SECONDS)

ESF	DLMMSE	MSG	LPA	EDAEP	VDI	LDI-NAT	RI	MLRI	FDRI
5.9839	8.5878	5.5430	0.6528	0.2455	3.3062	519.8159	1.1582	1.5514	0.8125

The numbers of required processes of RI, MLRI, and FDRI are shown in Table V. For guided filtering, residual calculating, and Laplacian filtering, the numbers of required processes per one image are listed. We represented only filtering and residual calculation processes because the operations of other processes, such as weight calculating and combining directional estimates, are very similar for RI, MLRI, and FDRI. Guided filtering was the most time-consuming process. FDRI required half the number of guided filtering process compared with RI and MLRI, and also have smaller operations for

residual calculation for the G channel interpolations. Guided filtering was a more time-intensive process than Laplacian filtering and residual calculating.

We compared computational complexity using CPU processing time. Each methods original source code from the respective authors was used to calculate the CPU processing time under the same test condition. As shown in Table VI, the proposed algorithm took 0.81 s per image; MLRI and RI, respectively, took 1.55 and 1.15 s per image. FDRI reduced the guided filtering and residual calculation processes. However, it required an additional Laplacian filtering process; therefore, FDRI showed lower complexity than MLRI and still consumed lower computation time than RI. LDI-NAT consumed more than 8 min in the same condition on account of the nonlocal mean method used to refine the interpolation. VDI took an average of 3.31 s per image. Based on the results, the proposed algorithm demonstrated a relatively low complexity burden.

V. Conclusion

In this paper, we proposed an FDRI method, which improves RI by exploiting four directions and adopts a 5×5 Laplacian filter. From the experimental results, the proposed G plane interpolation yielded a high-performance gain; the developed method was applied to the McMaster data set and LC images with great objective results. These results showed that the FDRI strategy can be very effective on images captured by a digital sensor. In addition, the subjective results demonstrated that FDRI successfully removed demosaicking artifacts, such as false coloring, and produced clean images.

References

[1] B. E. Bayer, "Color imaging array," U.S. Patent 3 971 065, Jul. 20, 1976.

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

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

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

[5] D. Paliy, V. Katkovnik, R. Bilcu, S. Alenius, and K. Egiazarian, "Spatially adaptive color filter array interpolation for noiseless and noisy data," *Int. J. Imag. Syst. Technol.*, vol. 17, no. 3, pp. 105–122, 2007.

[6] Z. Dengwen, S. Xiaoliu, and D. Weiming, "Colour demosaicking with directional filtering and weighting," *IET Image Process.*, vol. 6, no. 8, pp. 1084–1092, Nov. 2012.

[7] C. A. Laroche and M. A. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent 5 373 322, Dec. 13, 1994.

[8] H. S. Malvar, L.-W. He, and R. Cutler, "High-quality linear interpolation for demosaicing of Bayer-patterned color images," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2004, pp. 485–488.

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

[10] D. Menon, S. Andriani, and G. Calvagno, "Demosaicing with directional filtering and *a posteriori* decision," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 132–141, Jan. 2007.

[11] I. Pekkucuksen and Y. Altunbasak, "Edge strength filter based color filter array interpolation," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 393–397, Jan. 2012.

[12] I. Pekkucuksen and Y. Altunbasak, "Multiscale gradients-based color filter array interpolation," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 157–165, Jan. 2013.

[13] X. Li, B. Gunturk, and L. Zhang, "Image demosaicing: A systematic survey," in *Proc. Electron. Imaging* 2008, pp. 68221J-1–68221J-15.

[14] *The Kodak Color Image Dataset*. [Online]. Available: http://r0k.us/graphics/kodak/

[15] L. Zhang, A. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *J. Electron. Imag.*, vol. 20, no. 2, p. 023016, 2011.

[16] W.-J. Chen and P.-Y. Chang, "Effective demosaicking algorithm based on edge property for color filter arrays," *Digital Signal Process.*, vol. 22, no. 1, pp. 163–169, Jan. 2012.

[17] X. Chen, G. Jeon, and J. Jeong, "Voting-based directional interpolation method and its application to still color image demosaicking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 2, pp. 255–262, Feb. 2014.

[18] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, "Residual interpolation for color image demosaicking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2013, pp. 2304–2308.

[19] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, "Minimized-Laplacian residual interpolation for color image demosaicking," *Proc. IS&T/SPIE Electron. Imaging*, pp. 90230L-1–90230L-8, 2014.

[20] J. F. Hamilton and J. E. Adams, "Adaptive color plan interpolation in single sensor color electronic camera," U.S. Patent 5 629 734, May 13, 1997.

[21] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Computer Vision*. Berlin, Germany: Springer-Verlag, 2010, pp. 1–14.

[22] X. Zhang. (1998). S-CIELAB: A spatial extension to the CIE L*a*b* DeltaE color difference metric. Stanford Univ., Stanford, CA, USA. [Online]. Available: http://white.stanford.edu/~brian/scielab/scielab.html

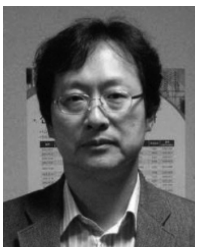
[23] *The McMaster Color Image Dataset*. [Online]. Available: http://www4.comp.polyu.edu.hk/~cslzhang/CDM_Dataset.htm, accessed 2014.

[24] L. Zhang, D. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.

[25] Laurent Condat Image Dataset. (2010). [Online]. Available: http://www.gipsa-lab.grenoble-inp.fr/~laurent.condat/imagebase.html

Yonghoon Kim received the B.S. degree from the Department of Electronics and Computer Engineering, Hanyang University, Seoul, Korea, in 2009, where he is currently working toward the Ph.D. degree in electronic and computer engineering.

His research interests include motion estimation, intra prediction of video compression in H.264 and High Efficiency Video Coding, and image processing, including demosaicing and denoising.



Jechang Jeong received the B.S. degree in electronics engineering from Seoul National University, Seoul, Korea, in 1980; the M.S. degree in electrical engineering from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1982; and the Ph.D. degree in electrical engineering from University of Michigan, Ann Arbor, MI, USA, in 1990.

He was with Korean Broadcasting System, Seoul, from 1982 to 1986, where he helped to develop teletext systems. From 1990 to 1991, he was a Post-Doctoral Research Associate with University of Michigan, where he helped to develop various signal-processing algorithms. From 1991 to 1995, he was with Samsung Electronics Company, Seoul, where he was involved in the development of HDTV, digital broadcasting receivers, and other multimedia systems. Since 1995, he has conducted research with Hanyang University, Seoul. He has authored numerous technical papers. His research interests include digital signal processing, digital communication, and image and audio compression for HDTV and multimedia applications.

Dr. Jeong was a recipient of the Scientist of the Month Award from the Ministry of Science and Technology of Korea in 1998, the IEEE Chester Sall Award in 2007, and *ETRI Journal* Paper Award in 2008. He was also honored with a government commendation from the Ministry of Information and Communication of Korea in 1998.