Advanced Directionally Weighted Demosaicing for Digital Cameras

Tae-young Jung, Siyoung Yang, Joohyun Lee, and Jechang Jeong
Department of Electronics and Computer Engineering,
Hanyang University, 17 Haengdang-dong, Seongdong-gu, Seoul, Korea

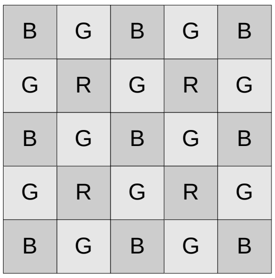
***Abstract*- Most digital camera uses the color filter array to reduce the cost and size. When we use the color filter array, however, we need to interpolate the missing color components. In this paper, we propose an advanced directionally weighted demosaicing algorithm for the color filtered images of the single-sensor digital camera. The proposed algorithm estimates the missing pixels by summing weighted color-differences in various directions. Compared to the conventional algorithms, the proposed algorithm achieves better visual quality without visual artifacts, such as aliasing and zippering.**

I. INTRODUCTION

To reduce the size and the cost, the color filter array is used in most commercially available digital camera [1]. When using the color filter array, only a single color component per pixel is obtained. For this reason, we need to interpolate the missing two components per pixel. This process is called as demosacing or color filter interpolation.

The most commonly used color filter pattern is the Bayer pattern [2] as shown in Fig. 1. The human visual system is more sensitive to the luminance (green) components than to the chrominance (red/blue) components [3]. In this reason, the density of green component is two times more than the one of red/blue component.

The demosaicing algorithms are classified by two groups. The first group interpolates the missing components separately per each color channel. Nearest neighborhood interpolation and bilinear interpolation are included in this group. However the nearest neighborhood interpolation and the bilinear interpolation shows rainbow aliasing effects in edge region. To improve the quality in edge region, the gradient based algorithms have been developed [4]. On the other hand, the other group uses the inter-channel correlation to interpolate missing pixels [5-9]. Since there are high correlations among the color channels in natural image, the algorithms in this group achieve better results than the first group, which does not consider inter-channel correlation.



B	G	B	G	B
G	R	G	R	G
B	G	B	G	B
G	R	G	R	G
B	G	B	G	B

Fig. 1. Bayer pattern.

II. PROPOSED ALGORITHM

A. Step 1: Interpolate missing green values

In most case, the missing pixels are interpolated by neighboring pixels. In this paper, we consider twelve directions to increase the edge sensitivity. The directions are illustrated in Fig. 2. Usually the missing green value on B33 is interpolated by G23, G32, G34, and G43. Additionally the proposed algorithm uses G12, G14, G21, G25, G41, G45, G52, and G54 to detect the edge more sensitively. The directions are numbered as Fig. 2 for the convenience of expression. The difference, $D_{N,n}$, between the samples in direction n of mode N is defined by

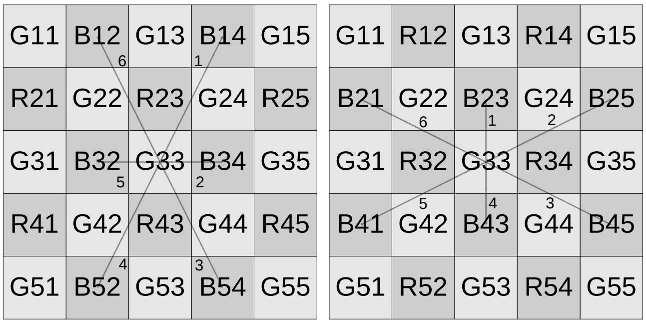
$$D_{N,n}(i,j) = \left|P(i+v_{N,n}, j+h_{N,n}) - P(i-v_{N,n}, j-h_{N,n})\right| + \left|P(i+2v_{N,n}, j+2h_{N,n}) - P(i,j)\right| \tag{1}$$

where, $P(i, j)$ denotes the sample on position (i, j). The offsets, $v_{N,n}$ and $h_{N,n}$, of nearby pixels used in mode N are listed in Table I, II, III, and IV. Since the red and blue channels are highly correlated with the green channel, color differences between red/blue and green are effective to interpolate the missing values. Like some previous algorithms, our algorithm is based on the color difference approach.

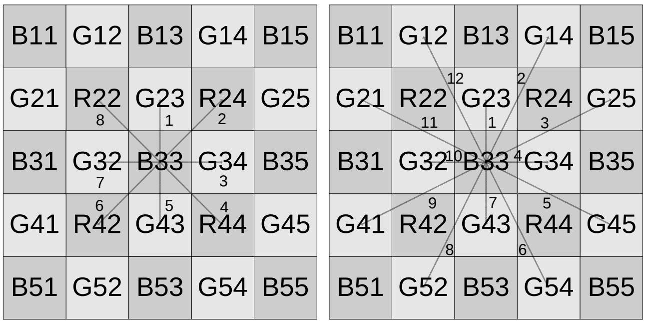
To reconstruct the missing value on (i, j), we interpolate the missing value with weighted sum of the color differences. The weight for the direction n of mode N, $w_{N,n}(i, j)$, is defined by

$$w_{N,n}(i,j) = \frac{1}{1+D_{N,n}(i,j)}. \tag{2}$$

This research was supported by Seoul Future Contents Convergence(SFCC) Cluster established by Seoul R&BD Program.



(a) 6 directions mode (vertical) (b) 6 directions mode (horizontal)



(c) 8 directions mode (d) 12 directions mode

Fig. 2. Directions of interpolation.

The color difference, $K^B_{N,n}$, between the green value G and the blue value B is calculated by subtracting the blue value from the green value in direction n of mode N. The blue values on green samples are required, however there aren't the blue samples, the averages of the nearby blue samples are used instead.

$$K^{B}_{N,n} = G(i+v_{N,n}, j+h_{N,n}) - B(i+v_{N,n}, j+h_{N,n}). \qquad (3)$$

The missing green values on the blue samples are interpolated by adding $B(i, j)$ and the weighted sum of nearby $K^B_{N,n}$ in directions described in Fig. 2(d).

$$G(i,j) = B(i,j) + \frac{\sum_{n=1}^{12} w_{12,n}(i,j)\cdot K^{B}_{12,n}(i,j)}{\sum_{n=1}^{12} w_{12,n}(i,j)}. \qquad (4)$$

Interpolating the missing green value, $G(i, j)$, on the red sample is almost same as above.

$$G(i,j) = R(i,j) + \frac{\sum_{n=1}^{12} w_{12,n}(i,j)\cdot K^{R}_{12,n}(i,j)}{\sum_{n=1}^{12} w_{12,n}(i,j)} \qquad (5)$$

where, $K^R_{N,n}(i, j) = G(i+v_{N,n}, j+h_{N,n}) - R(i+v_{N,n}, j+h_{N,n})$. $K^R_{N,n}$ is the color difference between red and green value in direction n of mode N. Similarly the red value on the green sample are not existed, we use the average of nearby red samples.

B. Step 2: Interpolate missing red/blue values on green samples

The missing red/blue values on the green samples are interpolated in this step. As shown in Fig. 2(a) and 2(b), 2 types of 6 directions are available. If the desired color are laid in vertical direction as shown in Fig. 2(a), the offsets, $v_{6v,n}$, and $h_{6v,n}$, listed in Table II are used. In the other case as shown in Fig. 2(b), the offsets, $v_{6h,n}$, and $h_{6h,n}$, listed in Table IV are used in the interpolation. The mode of directions, m, is defined by

$$m = \begin{cases} 6v, & \text{vertical direction} \\ 6h, & \text{horizontal direction} \end{cases}. \qquad (6)$$

The missing blue value on green sample is interpolated by

$$B(i,j) = G(i,j) - \frac{\sum_{n=1}^{6} w_{m,n}(i,j)\cdot K^{B}_{m,n}(i,j)}{\sum_{n=1}^{6} w_{m,n}(i,j)} \qquad (7)$$

where, $w_{N,n}$ (i, j) is obtained from (2) and $K^B_{N,n}$ is obtained from (3). The interpolation of missing blue values on the green sample is similar to (7).

C. Step 3: Interpolate missing red/blue values on blue/red samples

Since the missing red/blue values on the green samples are already interpolated, 16 directions are available in this step. The proposed algorithms consider 8 directions mode and 12 directions mode as shown in Fig. 2(c) and 2(d). The offsets, $v_{8,n}$ and $h_{8,n}$, of nearby pixels used in 8 direction mode are listed in Table III.

TABLE I
OFFSETS OF NEARBY SAMPLES (6 DIRECTIONS: HORIZINTAL DIRECTION)

n	$v_{6h,n}$	$h_{6h,n}$	n	$v_{6h,n}$	$h_{6h,n}$
1	0	-1	4	+2	-1
2	0	+1	5	-2	+1
3	-2	-1	6	+2	+1

TABLE II
OFFSETS OF NEARBY SAMPLES (6 DIRECTIONS: VERTICAL DIRECTION)

n	$v_{6v,n}$	$h_{6v,n}$	n	$v_{6v,n}$	$h_{6v,n}$
1	-1	0	4	-1	+2
2	+1	0	5	+1	-2
3	-1	-2	6	+1	+2

TABLE III
OFFSETS OF NEARBY SAMPLES (8 DIRECTION)

n	$v_{8,n}$	$h_{8,n}$	n	$v_{8,n}$	$h_{8,n}$
1	0	-1	5	0	1
2	1	-1	6	-1	1
3	1	0	7	-1	0
4	1	1	8	-1	-1

TABLE IV
OFFSETS OF NEARBY SAMPLES (12 DIRECTIONS)

n	$v_{12,n}$	$h_{12,n}$	n	$v_{12,n}$	$h_{12,n}$
1	0	-1	7	0	1
2	1	-2	8	-1	2
3	2	-1	9	-2	1
4	1	0	10	-1	0
5	2	1	11	-2	-1
6	1	2	12	-1	-2

The interpolation of missing blue value on the red sample in 8 directions mode is defined by

$$B(i,j) = G(i,j) - \frac{\sum_{n=1}^{8} w_{8,n}(i,j)\cdot K^{B}_{8,n}(i,j)}{\sum_{n=1}^{8} w_{8,n}(i,j)}. \tag{8}$$

The interpolation of missing blue value on the red sample in 12 directions mode is defined by

$$B(i,j) = G(i,j) - \frac{\sum_{n=1}^{12} w_{12,n}(i,j)\cdot K^{B}_{12,n}(i,j)}{\sum_{n=1}^{12} w_{12,n}(i,j)}. \tag{9}$$

In the interpolation of the missing red values on the blue samples, $K^R_{8,n}$ and $K^R_{12,n}$ are used instead of $K^B_{8,n}$ *and* $K^B_{12,n}$.

D. Step 4: Re-interpolate missing green values on red/blue samples

In step 1, the missing green values on the blue/red samples are interpolated with an inaccurate blue/red values, however better blue/red values are obtained in step 2 and 3, we recalculate the missing green value with the interpolated blue/red values.

III. EXPERIMENTAL RESULTS

We compared the proposed demosaicing algorithm with some conventional algorithms for various test images. For the image simulation, we used 24 images from the Kodak image database. These images are shown in Fig. 3. Table V shows the PSNR comparison for the bilinear interpolation, GB (gradient based) interpolation [4], ACPI (Adaptive Color Plane Interpolation) [4], DWCI (Directionally Weighted Color Interpolation) [7], and the proposed algorithms. In every case, the proposed algorithms achieve the highest PSNR.

Fig. 4. show the magnified details of the interpolated images and the error images. Fig. 4. (a, c, e, g, i, k, m) are first downsampled to Bayer pattern and then reconstructed to full color image. Fig 4. (b, d, f, h, j, l, n) show the difference between the interpolated image and the original image. It can be seen that the proposed algorithm makes less error in complicated edge region than the conventional algorithms.

IV. CONCLUSION

In this paper we propose the advanced directionally weighted demosaicing algorithm for Bayer color filter array. In this algorithm, we use both directional derivative and color correlation concepts. And this algorithm shows less color aliasing artifacts and zipper effects than the conventional algorithms.

V. REFERENCES

[1] J. A. Weldy, "Optimized design for a single-sensor color electronic camera system," in *Proc. SPIE*, pp. 300-307, 1988.

[2] B. E. Bayer, "Color imaging array," *U.S. Patent 3971065*, Jul. 1976.

[3] D. R. Cok, "Signal Processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," *U.S. Patent 4642678*, Feb. 1987.

[4] B. Eamanath, W.E. Synder, and G. L. Bilbro, "Demosaicking methods for Bayer color array," *Journal of Electronic Imaging*, vol. 11, no. 3, pp. 306-315, Jul. 2002.

[5] Bahadir K. Gunturk, Yucel Altunbasak, and Russell M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp 997-1013, Sept. 2002.

[6] N. Kehtarnavaz, H. Oh, and Y. Yoo, "Color filter array interpolation using correlations and directional derivatives," *Journal of Electronic Imaging*, vol. 12, no. 4, pp 621-632, Oct. 2003.

[7] Hung-An Chang and Homer Chen, "Directionally weighted color interpolation for digital cameras," in *proc. IEEE Int. Symp. on Cir. and Syst.*, vol. 6, pp. 6284-6287 23-26 May 2005.

[8] D. Darian Muresan and Thomas W. Parks, "Demosaicing using optimal recovery," *IEEE Transactions Image Processing*, vol. 14, no. 2, pp 267-278, Feb. 2005.

[9] Wonjae Lee, Seongjoo Lee, and Jaeseok Kim, "Cost-efffective color filter array demosaicing using spatial correlation,", *IEEE Trans. Consumer Electronics*, vol. 52, no. 2, pp 547-554, May 2006.

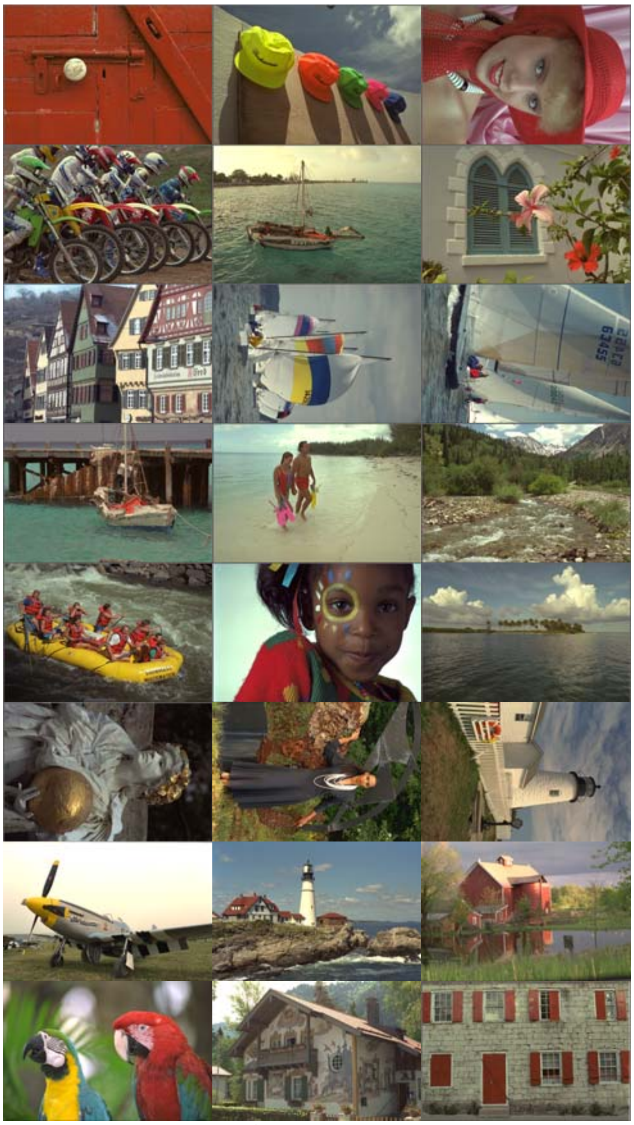


Fig. 3. Set of images used in simulation.
Images are numbered from 1 to 24, in order of left-to-right, top-to-bottom.

(a) Bilinear (b) Error of bilinear (c) GB (d) Error of GB

(e) ACPI (f) Error of ACPI (g) DWCI (linear) (h) Error of DWCI (linear)

(i) DWCI (stochastic) (j) Error of DWCI (stochastic) (k) Proposed (8 direction) (l) Error of proposed (8 direction)

(m) Proposed (12 direction) (n) Error of proposed (12 direction) (o) Original image

Fig. 4. Magnified details of interpolated images and error images.

Table V

PSNR(dB) COMPARISON

		Bilinear	GB	ACPI	DWCI: linear	DWCI: stochastic	Proposed: 8 directions	Proposed: 12 directions
1	All	25.872	28.550	31.686	37.075	37.125	37.435	37.703
	R	24.678	28.280	31.585	36.081	36.128	36.391	36.730
	G	28.546	28.891	31.904	40.356	40.409	40.860	40.860
	B	25.292	28.503	31.578	36.027	36.079	36.385	36.680
2	All	31.918	34.088	36.893	37.489	37.583	38.386	38.249
	R	30.539	32.793	36.391	36.626	36.730	37.499	37.185
	G	34.933	35.082	37.235	37.930	38.006	38.986	38.986
	B	31.399	34.759	37.100	38.059	38.160	38.834	38.812
3	All	33.058	35.044	37.801	40.382	40.455	41.162	41.002
	R	31.110	34.742	37.545	40.583	40.671	41.365	41.158
	G	37.797	35.833	38.318	41.339	41.411	42.338	42.338
	B	32.653	34.655	37.582	39.437	39.500	40.081	39.863
4	All	32.060	34.645	36.319	38.470	38.545	39.232	39.075
	R	30.890	33.408	35.244	36.534	36.605	37.191	36.864
	G	33.473	35.468	36.751	39.676	39.766	40.771	40.771
	B	32.199	35.385	37.215	40.141	40.208	40.788	40.883
5	All	26.263	28.541	31.068	35.418	35.489	36.022	35.802
	R	25.174	28.507	31.575	35.585	35.665	36.128	35.896
	G	28.142	28.558	30.593	36.885	36.956	37.712	37.712
	B	25.986	28.560	31.093	34.197	34.262	34.729	34.411
6	All	27.547	29.320	32.818	37.097	37.134	37.440	37.643
	R	25.991	29.084	32.768	36.666	36.708	36.999	37.381
	G	31.950	29.882	33.088	40.381	40.410	40.806	40.806
	B	26.734	29.045	32.612	35.555	35.590	35.879	36.014
7	All	32.305	35.244	36.525	39.688	39.772	40.524	40.250
	R	30.701	35.142	36.804	40.780	40.884	41.521	41.268
	G	35.493	35.632	36.417	40.588	40.666	41.699	41.699
	B	31.998	34.985	36.368	38.197	38.274	38.934	38.510
8	All	23.052	26.571	30.522	33.515	33.588	34.133	34.364
	R	22.294	26.413	30.473	32.673	32.749	33.320	33.626
	G	24.430	26.976	30.962	36.782	36.850	37.411	37.411
	B	22.713	26.350	30.168	32.326	32.398	32.915	33.161
9	All	31.114	34.654	37.615	40.370	40.433	41.040	40.968
	R	30.037	34.407	37.039	40.021	40.086	40.637	40.610
	G	32.259	35.277	38.066	42.320	42.395	43.272	43.272
	B	31.336	34.338	37.806	39.305	39.360	39.885	39.743
10	All	31.028	33.687	36.867	40.011	40.061	40.529	40.407
	R	30.090	33.580	36.659	39.627	39.681	40.156	39.938
	G	32.024	34.083	37.084	41.859	41.919	42.631	42.631
	B	31.187	33.426	36.868	39.027	39.069	39.409	39.313
11	All	28.809	30.996	34.052	37.867	37.923	38.393	38.404
	R	27.503	30.640	33.877	37.221	37.278	37.696	37.707
	G	31.422	31.328	34.087	39.451	39.504	40.192	40.192
	B	28.391	31.049	34.198	37.278	37.335	37.730	37.747
12	All	32.044	34.906	38.162	40.754	40.800	41.287	41.318
	R	30.154	34.490	37.806	40.612	40.656	41.012	41.087
	G	35.483	35.649	38.853	42.057	42.102	42.946	42.946
	B	32.037	34.666	37.904	39.869	39.917	40.311	40.320
13	All	23.852	25.406	28.556	34.161	34.189	34.365	34.550
	R	22.840	25.523	28.736	34.153	34.182	34.300	34.690
	G	26.423	25.481	28.655	37.337	37.362	37.606	37.606
	B	23.117	25.220	28.290	32.351	32.381	32.573	32.683
14	All	28.764	30.626	32.825	34.763	34.860	35.684	35.436
	R	27.440	30.248	32.973	34.654	34.773	35.656	35.265
	G	31.725	31.253	33.072	35.894	35.984	37.059	37.059
	B	28.203	30.441	32.454	33.957	34.041	34.664	34.393
15	All	30.230	34.040	35.514	37.106	37.178	37.818	37.699
	R	28.885	33.186	35.063	35.933	36.009	36.548	36.295
	G	31.971	34.642	36.282	37.766	37.823	38.637	38.637
	B	30.374	34.439	35.292	37.911	37.992	38.614	38.602
16	All	30.721	32.604	36.250	39.898	39.939	40.317	40.542
	R	29.000	32.254	35.876	39.384	39.434	39.815	40.208
	G	36.843	33.213	36.695	42.687	42.709	43.176	43.176
	B	29.684	32.404	36.218	38.601	38.641	38.983	39.160

17	All	31.761	34.158	36.278	40.099	40.139	40.543	40.462
	R	31.437	34.515	36.761	40.279	40.327	40.694	40.586
	G	33.533	34.208	36.002	42.184	42.220	42.786	42.786
	B	30.766	33.782	36.109	38.571	38.608	38.967	38.871
18	All	27.586	29.332	32.064	35.770	35.814	36.208	36.182
	R	27.173	29.449	32.139	35.575	35.626	36.012	35.977
	G	28.889	29.541	32.064	37.781	37.830	38.475	38.475
	B	26.943	29.024	31.990	34.549	34.585	34.870	34.840
19	All	27.671	30.870	35.136	37.955	38.030	38.640	38.762
	R	27.020	30.610	35.042	37.131	37.213	37.834	38.039
	G	29.154	31.412	35.306	41.205	41.256	41.837	41.837
	B	27.150	30.636	35.065	36.757	36.834	37.443	37.535
20	All	29.752	30.879	35.318	38.671	38.739	39.232	39.162
	R	29.071	30.345	35.828	39.393	39.478	40.026	40.040
	G	32.969	31.419	36.195	40.949	41.009	41.693	41.693
	B	28.438	30.940	34.199	36.737	36.799	37.192	37.053
21	All	28.383	30.207	33.003	37.972	38.014	38.341	38.433
	R	27.156	30.046	33.072	37.965	38.016	38.303	38.556
	G	31.785	30.554	33.072	40.676	40.712	41.237	41.237
	B	27.532	30.042	32.866	36.323	36.362	36.644	36.661
22	All	29.885	31.754	34.254	36.721	36.776	37.285	37.174
	R	29.167	31.409	34.399	37.107	37.164	37.665	37.494
	G	31.521	32.304	34.663	37.916	37.975	38.760	38.760
	B	29.338	31.600	33.752	35.497	35.547	35.912	35.785
23	All	33.467	34.995	37.573	39.141	39.214	39.773	39.575
	R	31.316	33.724	37.156	38.318	38.417	39.097	38.860
	G	37.284	35.996	38.326	40.255	40.345	41.393	41.393
	B	33.720	35.620	37.326	39.064	39.094	39.194	38.919
24	All	26.535	28.207	30.598	33.557	33.584	33.843	33.801
	R	26.323	28.766	31.392	34.001	34.033	34.365	34.236
	G	28.686	28.476	31.095	35.731	35.761	36.224	36.224
	B	25.262	27.485	29.543	31.823	31.845	31.982	31.974