

Edge Sensing Demosaicing Using Adaptive Weighted Interpolation

¹Dev R. Newlin and ²Elwin Chandra Monie

¹Department of Electronics and Communication Engineering,
Faculty of Engineering, Noorul Islam College of Engineering, Kumaracoil, India

²Department of Electronics and Communication Engineering,
Faculty of Engineering, R.M.K. College of Engineering, Kavaraipettai, Chennai

Received 2012-05-31, Revised 2013-01-20; Accepted 2013-05-15

ABSTRACT

Demosaicing is a process of obtaining a full color image by interpolating the missing colors of an image captured from a digital still and video cameras that use a single-sensor array. The edges are not handled properly in the existing demosaicking methods. This study provides an effective demosaicing algorithm applying an adaptive weighted-edge interpolation to handle green pixels. Due to the introduction of adaptive threshold based on the high frequency information for the classification of edges and textures, the interpolation of edges is performed excellently. Adaptive weights depending on spatial and spectral correlation are assigned for interpolation on both edges and smooth region. A refinement scheme based on enhanced effective color interpolation is used to improve the red and blue interpolation to obtain better quality image. As compared with the latest demosaicing algorithms, the proposed algorithm produces the best average demosaicing performance both objectively and subjectively. This approach handles the edges in a proper way and reconstructs the color image with best peak signal to noise ratio.

Keywords: Demosaicing, Bayer CFA, Adaptive Weighted Interpolation

1. INTRODUCTION

In order to reduce the hardware cost, many digital still cameras (DSC) use a single sensor equipped with a color filter array (CFA) to capture any one of the three primary color components, R (Red), G (Green), or B (Blue), on each pixel location. Among the various suggested CFAs, the Bayer CFA pattern (Bayer, 1976) is the most prevalent one, where G pixels occupy half of all and R and B pixels share the others. A representation of a full-color image needs all the information from the three colors on each pixel location. As a result, the missing two colors on each pixel location should be interpolated back to get a full-color image. The process of interpolating the missing colors is called as demosaicing or color interpolation whose main objective aims to reconstruct the missing

colors as closely to the original ones as possible while keeping the computational complexity as low as possible.

It is identified that most of the existing techniques (Menon *et al.*, 2007; Chung and Chan, 2006; Su, 2006; Gunturk *et al.*, 2002; Li, 2005; Zen *et al.*, 1998) have not considered the level of smoothness of smoother region during interpolation. Poor quality on smoother region in demosaiced image has resulted due to this problem. Addressing the issue of improving the quality on smoother region was another focus of this research study. A weighted technique to address this problem is proposed on Demosaicing using Adaptive weighted interpolation to improve the quality on smoother region.

Demosaicing using Adaptive weighted interpolation (Hore and Ziou, 2011), in which interpolation for each missing green component in the CFA, the algorithm

Corresponding Author: Dev. R. Newlin, Department of Electronics and Communication Engineering, Faculty of Engineering, Noorul Islam College of Engineering, Kumaracoil, India

performs a gradient test, to identify edge direction and then carries out an interpolation along the direction of a smaller gradient to determine the missing green component. For smooth region, the variance of color differences in vertical, horizontal and diagonal directions can be used as a supplementary criterion to determine the interpolation direction for the green components. Other color planes are estimated based on the interpolated value of the green plane.

Due to the introduction of adaptive threshold based on the high frequency information for the classification of edges and textures, the interpolation of edges is performed excellently. Adaptive weights depending on spatial and spectral correlation are assigned for interpolation on both edges as well as smooth region. A refinement scheme based on enhanced effective color interpolation is used to improve the red and blue interpolation to make better quality image. As compared with the latest demosaicing algorithms, the proposed algorithm produces the best average demosaicing performance objectively and subjectively.

The study describes how to demosaic an image captured through a single sensor array camera and simultaneously produce a better image at various edge levels.

2. MATERIALS AND METHODS

2.1. Edge Sensing Demosaicking Using Adaptive Weighted Interpolation

The directional gradients for 5x5 windows are computed first, sharp and smooth regions are effectively classified using an adaptive technique that uses spatial and spectral correlation of respective local window. To interpolate the green component at each red and blue pixel positions on sharp region, adaptive weights are derived using the local window characteristics. On smooth region, variances of color differences are computed horizontally, vertically and block-wise to find the suitable interpolation direction in-order to give best performance. The green component on smooth region is also estimated by applying the weighting technique. Once the green plane is interpolated, then red and blue components can be interpolated accurately using the green component available or interpolated at the corresponding position with the color difference plane.

A refinement step is used finally to generate more consistent values for red and blue components as close as to original, which is based on exploiting the image spectral correlation on either green and red or green and blue color planes.

2.2. Computation of Directional Gradients

To classify the sharp and smooth regions on the mosaic image and to interpolate green component, the directional gradients L^H (horizontal) and L^V (vertical) are computed for each red pixel position. There is a variety of classifiers used in demosaicing algorithms. The simplest first order spatial classifier is the neighborhood difference in pixel values of green plane as $|G_{i,j-1} - G_{i,j+1}|$. The most commonly used classifier in the demosaicing algorithms is:

$$|G_{i,j-1} - G_{i,j+1}| + |2R_{i,j} - R_{i,j-2} - R_{i,j+2}|$$

which uses both spatial and spectral correlations. These classifiers are not good enough to classify a natural image into sharp and smooth regions.

To compute the directional gradients L^H and L^V at blue pixel position, Equation 1 and 2 should be modified by replacing R by B and B by R:

$$L^H = \left| \begin{aligned} & \sum_{n=\pm 2} \left[\sum_{m=0,\pm 2} R_{i+m,j+n} + \sum_{m=\pm 1} G_{i+m,j+n} \right] \\ & + \sum_{n=\pm 1} \left[\sum_{m=0,\pm 2} G_{i+m,j+n} + \sum_{m=\pm 1} B_{i+m,j+n} \right] \\ & - 4 \left[\sum_{m=0,\pm 2} R_{i+m,j} + \sum_{m=\pm 1} G_{i+m,j} \right] \end{aligned} \right| \tag{1}$$

$$L^V = \left| \begin{aligned} & \sum_{m=\pm 2} \left[\sum_{n=0,\pm 2} R_{i+m,j+n} + \sum_{n=\pm 1} G_{i+m,j+n} \right] \\ & + \sum_{m=\pm 1} \left[\sum_{n=0,\pm 2} G_{i+m,j+n} + \sum_{n=\pm 1} B_{i+m,j+n} \right] \\ & - 4 \left[\sum_{n=0,\pm 2} R_{i,j+n} + \sum_{n=\pm 1} G_{i,j+n} \right] \end{aligned} \right| \tag{2}$$

2.3. Interpolation of Green Plane for Sharp Edge Blocks

The leading edge direction is determined by computing the ratio of the above two directional gradients and as in Equation 3:

$$e = \max(L^H | L^V, |L^V | L^H) \tag{3}$$

To classify the regions, an edge threshold is derived based on the edge details of mosaic image. The edge threshold (ethresh) is calculated as the ratio of number of 5x5 blocks to number of 5x5 blocks for which $e \in \{1, 2\}$ in the image. If the e (edge value) of a 5x5 block is greater than the ethresh (edge threshold i.e., $(e > \text{ethresh})$), then the block is defined as a sharp edge block. If horizontal gradient value and vertical gradient value of a block are same, then the block is defined as perfectly smooth block and if $e < \text{ethresh}$ for a block, then that block is classified as smooth block.

2.4. Determination of Weights for Sharp Edge Blocks

Weights for sharp edge blocks (W_{sharp}) are determined by local neighborhood as a ratio of color differences. For a reference neighborhood of 3x3 Bayer mosaic image with red pixel at center as shown in Fig. 1. weights are estimated using Equation 4:

$$W_{\text{sharp}} = \begin{cases} \frac{(R_{i,j} - G_{\text{left}})(R_{i,j} - G_{\text{right}})}{(R_{i,j} - G_{\text{up}})(R_{i,j} - G_{\text{down}})} & \text{if } L^H < L^V \\ \frac{(R_{i,j} - G_{\text{up}})(R_{i,j} - G_{\text{down}})}{(R_{i,j} - G_{\text{left}})(R_{i,j} - G_{\text{right}})} & \text{if } L^V < L^H \end{cases} \tag{4}$$

For a reference neighborhood of 3x3 Bayer mosaic image with blue pixel at center as shown in Fig. 2, weights are estimated using Equation 5:

$$W_{\text{sharp}} = \begin{cases} \frac{(B_{i,j} - G_{\text{left}})(B_{i,j} - G_{\text{right}})}{(R_{i,j} - G_{\text{up}})(R_{i,j} - G_{\text{down}})} & \text{if } L^H < L^V \\ \frac{(B_{i,j} - G_{\text{up}})(B_{i,j} - G_{\text{down}})}{(B_{i,j} - G_{\text{left}})(B_{i,j} - G_{\text{right}})} & \text{if } L^V < L^H \end{cases} \tag{5}$$

2.5. Interpolation of Green Plane for Sharp Edge Blocks

For sharp edge blocks, green components at red pixel positions are estimated exploiting the neighborhood correlation by weight of sharp transition (W_{sharp}) as in Equation 6:

$$g_{i,j} W_{\text{sharp}} = \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i,j-2} - R_{i,j+2}}{4} \right] / 4 \right) + (1 - W_{\text{sharp}}) \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i-2,j} - R_{i+2,j}}{4} \right] / 4 \right) \text{ if } L^H < L^V \tag{6}$$

$$g_{i,j} = W_{\text{sharp}} \left(\left[\frac{G_{i-1,j} + G_{i+1,j}}{2} + \frac{2R_{i,j} - R_{i-2,j} - R_{i+2,j}}{4} \right] / 4 \right) + (1 - W_{\text{sharp}}) \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i,j-2} - R_{i,j+2}}{4} \right] / 4 \right) \text{ if } L^V < L^H$$

Similarly green components at blue pixel positions for sharp edge blocks are estimated as in Equation 7:

$$g_{i,j} W_{\text{sharp}} = \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2B_{i,j} - B_{i,j-2} - B_{i,j+2}}{4} \right] / 4 \right) + (1 - W_{\text{sharp}}) \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2B_{i,j} - B_{i-2,j} - B_{i+2,j}}{4} \right] / 4 \right) \text{ if } L^H < L^V \tag{7}$$

$$g_{i,j} = W_{\text{sharp}} \left(\left[\frac{G_{i-1,j} + G_{i+1,j}}{2} + \frac{2B_{i,j} - B_{i-2,j} - B_{i+2,j}}{4} \right] / 4 \right) + (1 - W_{\text{sharp}}) \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2B_{i,j} - B_{i,j-2} - B_{i,j+2}}{4} \right] / 4 \right) \text{ if } L^V < L^H$$

2.6. Determination of Variances of Color Differences for Smooth Edge Blocks

The variances of the color differences for smooth edge blocks are computed at each red and blue pixel positions using Equation 8, along horizontal, vertical and diagonal directions and are represented as, $H\sigma_{i,j}^2, V\sigma_{i,j}^2$ and $B\sigma_{i,j}^2$ respectively.

2.7. Interpolation of Green Plane for Smooth Edge Blocks

The missing green value can be estimated for smooth region by using an experimentally found weight (W_{smooth}) constant throughout the smooth regions as in Equation 8-10. The weight value ($W_{\text{smooth}} = 0.87$) is used in this method, which has given the performance close to optimal.

If $F\sigma_{i,j}^2 = \min(H\sigma_{i,j}^2, V\sigma_{i,j}^2, B\sigma_{i,j}^2)$ then:

$$g_{i,j} W_{\text{sharp}} = \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i,j-2} - R_{i,j+2}}{4} \right] / 4 \right) + (1 - W_{\text{sharp}}) \left(\left[\frac{G_{i,j-1} + G_{i,j+1}}{2} + \frac{2R_{i,j} - R_{i-2,j} - R_{i+2,j}}{4} \right] / 4 \right) \tag{8}$$

	G_{up}	
G_{left}	$R_{i,j}$	G_{right}
	G_{down}	

Fig. 1. Reference neighborhood of 3x3 red centered Bayer mosaic image

	G_{up}	
G_{left}	$B_{i,j}$	G_{right}
	G_{down}	

Fig. 2. Reference neighborhood of 3x3 blue centered Bayer mosaic image

if $V\sigma_{i,j}^2 = \min(H\sigma_{i,j}^2, V\sigma_{i,j}^2, B\sigma_{i,j}^2)$ then:

$$g_{i,j} W_{sharp} = \left(\begin{array}{l} \left[\frac{G_{i-1,j} + G_{i+1,j}}{2} \right. \\ \left. + \left[\frac{2R_{i,j} - R_{i-2,j} - R_{i+2,j}}{4} \right] \right] / 4 \end{array} \right)^+ \quad (9)$$

$$(1 - W_{sharp}) \left(\begin{array}{l} \left[\frac{G_{i,j-1} + G_{i,j+1}}{2} \right. \\ \left. + \left[\frac{2R_{i,j} - R_{i,j-2} - R_{i,j+2}}{4} \right] \right] / 4 \end{array} \right)$$

if $B\sigma_{i,j}^2 = \min(H\sigma_{i,j}^2, V\sigma_{i,j}^2, B\sigma_{i,j}^2)$ then:

$$g_{i,j} = \left(\begin{array}{l} \left[\frac{G_{i-1,j} + G_{i+1,j} + G_{i,j-1} + G_{i,j+1}}{4} \right. \\ \left. + \left[\frac{4R_{i,j} - R_{i-2,j} - R_{i+2,j} + R_{i,j-2} - R_{i,j+2}}{8} \right] \right] / 8 \end{array} \right) \quad (10)$$

2.8. Interpolation of Red and Blue Components at Green Pixel Positions

The missing red and blue values at green CFA sampling positions are estimated using the available and interpolated green plane along with the information on the plane itself as Equation 11:

$$C_{i,j} = G_{i,j} + (C_{i,j-1} - \tilde{G}_{i,j-1} + C_{i,j+1} + \tilde{G}_{i,j+1}) / 2 \quad (11)$$

where, C is R(red) or B(blue) color plane component and is the interpolated green plane component.

2.9. Interpolation of Red Components at Blue Pixel Position and Blue Components at Red Pixel Position

The missing red components at blue pixel positions and blue components at red pixel positions are estimated as in Equation 12:

$$C_{i,j} = \tilde{G}_{i,j} + \frac{\sum_{m=\pm 1} \sum_{n=\pm 1} (C_{i+m,j+n} - \tilde{G}_{i+m,j+n})}{4} \quad (12)$$

where, C is R (red) or B (blue) color plane component and is the interpolated green plane component.

2.10. Refinement on Red and Blue Interpolation

To reduce the error on red and blue estimated values, a refinement scheme is used based on spectral and spatial correlation. This refinement step is focused on generating more consistent and close to optimal values for red and blue pixels using the values interpolated in initial step.

The weighted color difference value is calculated exploiting the image spectral correlation on two color planes around a pixel under consideration. The weights along the four adjacent directions for the reference Fig. 3 are estimated as in Equation 13:

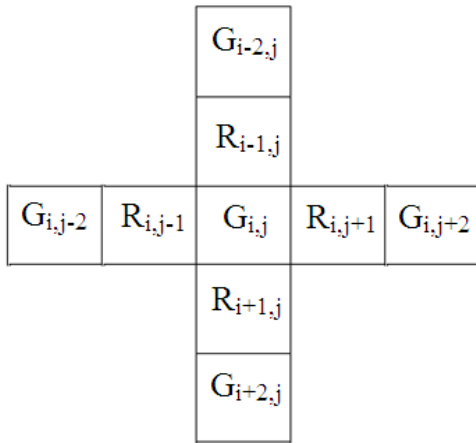


Fig. 3. Bayer mosaic image with vertical and horizontal neighborhoods

$$\beta_{i-1,j} = \frac{1}{1 + (|G_{i-2,j} - G_{i,j}| + |R_{i-1,j} - R_{i+1,j}|)}$$

$$\beta_{i+1,j} = \frac{1}{1 + (|G_{i+2,j} - G_{i,j}| + |R_{i-1,j} - R_{i+1,j}|)}$$

$$\beta_{i,j-1} = \frac{1}{1 + (|G_{i,j-2} - G_{i,j}| + |R_{i,j-1} - R_{i,j+1}|)}$$

$$\beta_{i,j+1} = \frac{1}{1 + (|G_{i,j+2} - G_{i,j}| + |R_{i,j-1} - R_{i,j+1}|)}$$
(13)

The color difference values at the four surrounding locations are estimated as in Equation 14:

$$\tilde{K}_{R(i+m,j+n)} = G_{i+m,j+n} - \tilde{R}_{i+m,j+n}$$

$$\tilde{K}_{B(i+m,j+n)} = G_{i+m,j+n} - \tilde{B}_{i+m,j+n}$$
(14)

The color difference value at the center location is estimated as in Equation 15 by assigning the weights of four adjacent color difference values:

$$\tilde{K}_{R(i,j)} = \frac{\tilde{K}_{R(i-1,j)}\beta_{i-1,j} + \tilde{K}_{R(i+1,j)}\beta_{i+1,j} + \tilde{K}_{R(i,j-1)}\beta_{i,j-1} + \tilde{K}_{R(i,j+1)}\beta_{i,j+1}}{\beta_{i-1,j} + \beta_{i+1,j} + \beta_{i,j-1} + \beta_{i,j+1}}$$
(15)

Similarly the color difference value is computed. Using the weighted color difference values, the red and blue pixel values are refined as in Equation 16:

$$\tilde{R}_{i,j} = \tilde{G}_{i,j} - \tilde{K}_{R(i,j)}$$

$$\tilde{B}_{i,j} = \tilde{G}_{i,j} - \tilde{K}_{B(i,j)}$$
(16)

Thus the weighted color difference value has generated more consistent and optimal demosaicing values.

Most of the demosaicing techniques use a hard threshold to classify the regions and hard interpolation solely along the edge direction that results poor interpolation on edges. Also a hard decision is used for the interpolation of smooth region that results poor image quality on smooth regions. These problems are considered while deriving this Adaptive Demosaicing.

In this method, to identify the edge details an adaptive threshold is proposed. Due to the introduction of adaptive threshold based on the high frequency information for the classification of edges and textures, the interpolation of edges can be performed superbly.

Adaptive weights depending on spatial and spectral correlation are assigned for interpolation on both edges and smooth region. A refinement scheme based on enhanced effective color interpolation is used to improve the red and blue interpolation to make better quality image. Due to the introduction of adaptive threshold and weights, this demosaicing technique can achieve high performance.

3. RESULTS

Simulation was carried out to evaluate the performance of the proposed algorithm.

The 24 digital color images shown in **Fig. 4** were used to generate a set of testing images. The Color Peak Signal-to-Noise Ratio (CPSNR) and Mean Square Error (MSE) were used as measures to quantify the performance of the demosaicing methods.

Table 1-3 and Fig. 5 shows the performance of various algorithms like Bilinear Interpolation (BI) (Sakamoto *et al.*, 1998), Effective Color Interpolation (ECI) (Pei and Tam, 2003), Enhanced ECI (EECI) (Chang and Tam, 2004) and Demosaicing using Variance of Color Difference (DVCD) (Chung and Chan, 2006). The proposed algorithm provided the best performance among the evaluated algorithm.



Fig. 4. Set of 24 bit tested images (images are numbered from 1 to 24 in the order of left-to-right and top-to-bottom)

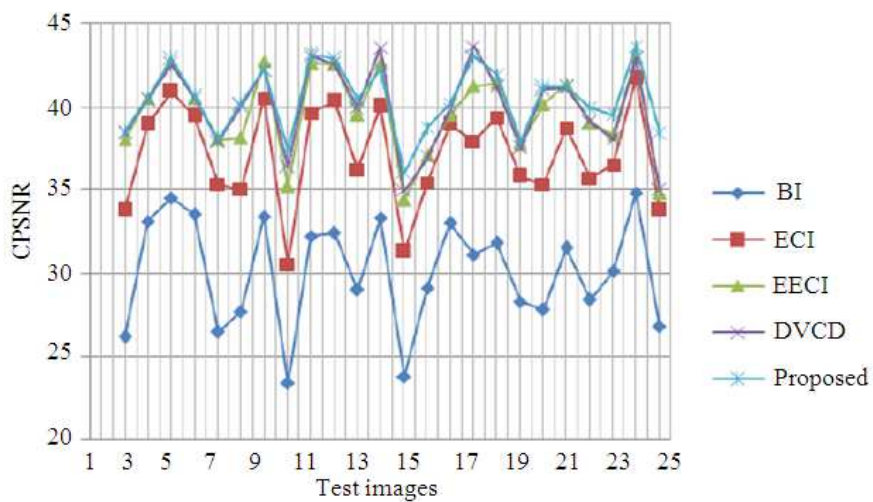


Fig. 5. CPSNR performance comparison of different demosaicing techniques

Table 1. MSE performance comparison of different demosaicing techniques with Proposed for red plane

Image/ method	BI	ECI	EECI	DVCD	Proposed
1	191.48	31.05	12.98	10.42	10.32
2	42.08	13.43	10.28	6.74	9.44
3	29.38	5.78	4.12	3.97	3.96
4	35.66	13.34	11.02	6.63	7.45
5	173.03	20.38	11.30	10.95	11.86
6	139.03	22.29	11.38	5.92	7.00
7	35.99	6.06	4.03	3.90	4.39
8	363.17	62.68	23.78	15.91	13.25
9	45.41	8.85	4.48	3.43	3.41
10	42.57	7.52	4.84	4.17	4.11
11	99.11	17.83	9.75	6.87	7.57
12	35.17	7.78	4.34	3.49	4.43
13	319.23	46.79	24.50	22.34	17.55
14	99.57	21.63	15.96	15.02	10.89
15	51.06	15.14	13.07	7.16	9.00
16	60.69	11.54	5.72	3.08	3.05
17	44.68	7.93	5.12	5.23	4.71
18	120.54	20.95	15.53	12.48	11.59
19	136.49	22.92	8.30	5.31	6.36
20	53.97	8.32	4.94	5.47	4.96
21	112.23	17.79	8.69	8.33	7.40
22	67.47	14.86	11.12	10.42	7.66
23	23.45	5.36	4.15	3.85	3.30
24	148.98	23.78	22.34	20.42	10.79
Average	102.93	18.08	10.49	8.40	7.69

Table 2. MSE performance comparison of different demosaicing techniques with Proposed for green plane

Image/ method	BI	ECI	EECI	DVCD	Proposed
1	71.63	17.71	5.61	7.20	7.18
2	15.39	4.84	2.70	4.37	2.53
3	12.48	3.18	1.83	3.19	1.80
4	14.46	3.95	2.55	4.38	4.13
5	76.05	13.46	5.25	9.38	8.07
6	51.06	13.04	4.94	6.74	4.73
7	14.66	3.95	1.90	3.49	2.44
8	118.34	31.92	8.92	12.40	8.90
9	17.42	4.74	2.16	3.03	2.14
10	18.89	4.04	2.05	2.70	1.95
11	39.01	10.17	3.95	6.26	3.93
12	13.52	3.74	1.78	1.64	2.52
13	145.92	35.41	13.43	18.84	12.17
14	40.94	10.62	5.69	10.20	5.66
15	18.93	5.04	3.52	6.17	3.23
16	21.88	6.30	2.54	2.35	2.50
17	22.86	6.00	3.00	4.28	2.98
18	57.43	14.43	7.40	9.16	7.25
19	43.56	12.31	3.66	4.93	3.36
20	22.65	5.72	2.51	4.42	3.39
21	45.62	11.38	4.22	7.03	4.03
22	30.14	8.89	5.19	10.17	5.06
23	10.35	2.69	1.78	2.26	1.64
24	75.01	18.16	11.30	17.80	6.50
Average	41.59	10.49	4.49	6.77	4.49

Table 3. MSE performance comparison of different demosaicing techniques with Proposed for blue plane

Image/ method	BI	ECI	EECI	DVCD	Proposed
1	189.72	29.86	11.73	10.32	10.07
2	37.77	7.55	4.67	6.68	5.26
3	26.92	6.40	4.42	3.82	4.40
4	33.28	5.45	3.95	6.39	5.48
5	166.38	21.83	12.95	11.32	11.08
6	129.75	25.18	12.62	6.84	6.77
7	35.90	6.98	4.50	4.37	4.82
8	367.37	67.01	23.45	16.39	13.49
9	47.33	7.61	4.54	3.32	4.05
10	48.99	7.42	4.93	4.21	4.06
11	95.31	16.00	7.48	6.82	6.34
12	37.85	7.50	4.15	3.57	4.90
13	325.92	56.12	30.84	21.94	19.63
14	89.77	19.87	13.40	14.62	9.60
15	38.65	7.62	6.01	7.10	6.98
16	59.45	12.05	5.97	3.13	4.18
17	53.35	9.00	6.20	5.28	5.00
18	135.24	21.44	14.19	13.04	12.74
19	131.25	21.73	7.36	5.27	5.03
20	56.77	11.62	7.21	5.26	6.31
21	114.85	20.71	10.89	8.64	8.34
22	77.82	17.34	11.35	10.34	8.96
23	25.18	4.42	3.63	3.91	3.58
24	192.36	41.60	32.67	23.46	10.69
Average	104.88	18.85	10.38	8.59	7.57

4. DISCUSSION

Different results prove the effectiveness of the study developed. We note that the proposed algorithm is developed based on the fact that the interpolation direction for each missing green sample is critical to the final demosaicing result.

Here due to the introduction of the adaptive weights based on edge levels this demosaicking achieves better performance.

5. CONCLUSION

In Edge Sensing Demosaicing using Adaptive weighted interpolation, an adaptive threshold exploiting the correlation among neighborhood pixels is employed for efficient classification of sharp edges and smooth region. To preserve the high frequency information, an adaptive weighted interpolation scheme is proposed. Also, to improve the performance of demosaicing technique on smooth region, an adaptive weighted interpolation scheme is employed. Simulation results show that this proposed algorithm is able to produce a

subjectively and objectively better demosaicing results as compared with a number of advanced algorithms widely used in the industry.

There are good scopes for future improvements in the proposed demosaicing techniques. For practical implementation, the computational complexities of the adaptive algorithms can be further reduced. An adaptive weighted interpolation by avoiding the classifiers can be used to achieve the demosaiced output without color artifacts at lower computational complexity.

6. REFERENCES

- Bayer, B.E., 1976. Color imaging array. U.S. Patent 3 971 065.
- Chang, L. and Y.P. Tam, 2004. Effective use of spatial and spectral correlations for color filter array demosaicking. *IEEE Trans. Consumer Elect.*, 50: 355-365. DOI: 10.1109/TCE.2004.1277885
- Chung, K.H. and Y.H. Chan, 2006. Color demosaicing using variance of color differences. *IEEE Trans. Image Process.*, 15: 2944-2955. DOI: 10.1109/TIP.2006.877521
- Gunturk, B.K., Y. Altunbasak and R. Mersereau, 2002. Color plane interpolation using alternating projections. *IEEE Trans. Image Process.*, 11: 997-1013. DOI: 10.1109/TIP.2002.801121
- Hore, A. and D. Ziou, 2011. An edge-sensing generic demosaicing algorithm with application to image resampling. *IEEE Trans. Image Process.*, 20: 3136-3150. DOI: 10.1109/TIP.2011.2159229
- Li, X., 2005. Demosaicing by successive approximation. *IEEE Trans. Image Process.*, 14: 370-379. DOI: 10.1109/TIP.2004.840683
- Menon, D., S. Andriani and G. Calvagno, 2007. Demosaicing with directional filtering and a posteriori decision. *IEEE Trans. Image Process.*, 16: 132-141. PMID: 17283772
- Pei, S.C. and I.K. Tam, 2003. Effective color interpolation in CCD color filter arrays using signal correlation. *IEEE Trans. Circ. Syst. Video Technol.*, 13: 503-513. DOI: 10.1109/TCSVT.2003.813422
- Sakamoto, T., C. Nakanishi and T. Hase, 1998. Software pixel interpolation for digital still cameras suitable for a 32-bit MCU. *IEEE Trans. Consumer Elect.*, 44: 1342-1352. DOI: 10.1109/30.735836
- Su, C.Y., 2006. Highly effective iterative demosaicing using weighted-edge and color-difference interpolations. *IEEE Trans. Consumer Elect.*, 52: 639-645.
- Zen, H., T. Koizumi, H. Yamamoto and I. Kimura, 1998. A new digital signal processor for progressive scan CCD. *IEEE Trans. Consumer Elect.*, 44: 289-295. DOI: 10.1109/30.681940