A Color Interpolation Algorithm for Bayer Pattern Digital Cameras Based on Green Components and Color Difference Space

Lei Hua[1], Lei Xie[1,2], *IEEE Member*, Huifang Chen[1,2], *IEEE Member*
[1]Dept. of Information Science and Electronic Engineering, Zhejiang University
[2]Zhejiang Provincial Key Laboratory of Information Network Technology
No. 38, Zheda Road, Hangzhou 310027, P. R. China
E-mail: {noah135, xiel, chenhf}@zju.edu.cn

***Abstract*—The color interpolation method of Bayer CFA used in most digital cameras plays an important role in imaging system. In Bayer pattern, the green components are the twice as much as red or blue and contain more image information, which can be used to compensate the interpolations of red and blue. Due to the high correlation among the three color channels, color mutation is prevented by the fixed color difference. Based on these principles, the proposed interpolation algorithm estimates green component firstly with adapted color plane interpolation algorithm, then computes the color difference image (*R-G* and *B-G*) to interpolate the full resolution difference image with the compensation of interpolated *G* and edge adaptive method. Edge blurring problem is addressed by using edge judgment to estimate *RGB*. Color distortion is reduced using color difference interpolation procedure. Experimental results show that the proposed algorithm outperforms existing typical interpolation methods in terms of image quality and peak signal-to-noise ratio (*PSNR*), and also computational complexity is low.**

Keywords – color interpolation; color filter array; Bayer pattern; digital cameras

I. INTRODUCTION

For cost and size considerations, most digital cameras and industrial cameras, which are increasingly being applied to daily life and production in the past twenty years, use a single image sensor to receive images. Since three color components (*RGB*) are needed to represent a full color pixel, surface of image sensor is covered with color filter array (CFA). Due to good sensitivity of color signals and color restoration features, Bayer pattern CFA [1] is the most commonly used in image sensors, shown in Fig. 1. Each sensitive point in Bayer pattern CFA only allows one color component to pass through, so each pixel sample only has gray value of one color component. To render a full color image, two missing colors are needed to be estimated from surrounding colors for each sample dot, referred to as color interpolation.

In the imaging technology of digital cameras, color interpolation in Bayer pattern CFA plays an important role to reconstruct pleasing image. An early interpolation method, as the hardware conditions of early system, prefers to use simple, fast speed algorithms, such as nearest neighbor, bilinear [2] and bicubic interpolation [3]. These algorithms calculate simple and fast, but they have many drawbacks, interpolated image quality is very poor, such as fringe effects and false colors. Especially many artifacts are produced in the edge region.

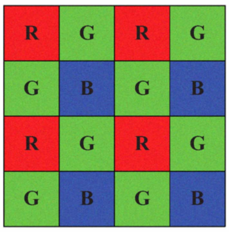

Figure 1. Bayer pattern CFA

In view of edge blurring, image details and texture, edge adaptive techniques [4] are proposed to prevent interpolation artifacts in edge lines. These algorithms implement color interpolation following the edge direction gradients to obtain good performance in the edge region. To prevent from detecting wrong edge directions, surrounding sample dots are proposed to compensate the edge adaptive algorithm.

In view of the correlation between color components and the importance of *G* component, *R* and *B* components are proposed to be estimated based on interpolated *G* component and color difference space [5] to prevent color mutation and improve the color smoothness.

By directing the algorithmic design to image texture and color distortion explicitly, the proposed interpolation algorithm outperforms existing typical color interpolation algorithms in terms of image quality and peak signal-to-noise ratio, and also computational complexity is low.

II. PROBLEM STATEMENT

Green samples are the twice as much as red or blue and contain much more image information in Bayer pattern CFA, shown in Fig. 1. Moreover, human eyes are more sensitive to green compared to red or blue, and also more sensitive to luminance compared to chrominance. In basic equations to convert from *RGB* to *YUV* color space, green component contributes nearly sixty percent in equation *Y*, expressed as

$$Y = 0.299R + 0.587G + 0.114B\,. \tag{1}$$

In view of these, since the interpolation of *G* component directly affects image visual effects, edge lines, details and texture, reconstruction *G* component is the most important.

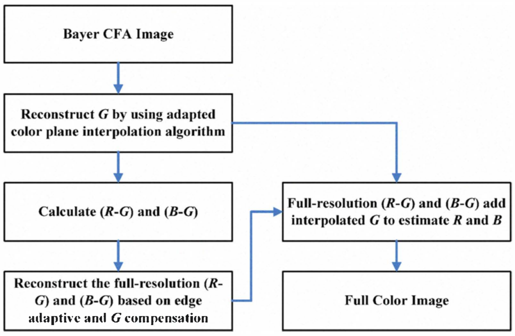

Figure 2. Overview of proposed algorithm

The proposed interpolation algorithm renders full-resolution green component firstly which is used to compensate the reconstruction of red and blue components.

In addition, colors are relevant in different color planes, namely color ratio rule. According to the model of Mondriaan color image, pixel gray value of each color plane can be considered as the projection of the light source direction $\hat{l}$ onto the real object surface normal $\hat{N}(x)$, multiplied by the albedo ρ_i, $i \in [R,G,B]$. The three color components can be written as

$$I_R(x) = \rho_R(x) < \hat{N}(x), \hat{l} >. \tag{2}$$

$$I_G(x) = \rho_G(x) < \hat{N}(x), \hat{l} >. \tag{3}$$

$$I_B(x) = \rho_B(x) < \hat{N}(x), \hat{l} >. \tag{4}$$

This means that the different color values capture the change in different materials via ρ_i that multiplies the normalized shading image $\hat{I}(x) =< \hat{N}(x), \hat{l} >$. The albedo is the same with a given object in the image, e.g. $\rho_i(x) = c_i$, where c_i is a constant. The following constant ratio holds:

$$\frac{I_i(x)}{I_j(x)} = \frac{\rho_i(x)\hat{I}(x)}{\rho_j(x)\hat{I}(x)} = \frac{\rho_i(x)}{\rho_j(x)} = \frac{c_i}{c_j} = \text{constant}. \tag{5}$$

Although the equation (5) is not correct in image edge and detail, the color correlation is quite effective in the relatively smooth location [6].

From (5), if $I_i(x)/I_j(x)$ is calculated using logarithmic operation in a small region, color difference rule is obtained, which can be expressed as $I_i - I_j = \text{constant}$. Since color difference rule doesn't have non-linear division operation, the proposed method reconstructs *R* and *B* based on color difference rule to prevent color mutation and improve the color smoothness.

III. Proposed Color Interpolation Algorithm

In this section, this paper presents the proposed interpolation algorithm for Bayer pattern digital cameras based on *G* component and color difference space in details.

There are many classical interpolation algorithms for Bayer pattern digital cameras, such as bilinear, edge adaptive [7] and adapted color plane interpolation [8], and so on. From the image quality, adapted color plane interpolation is more precise in edge estimation, reduces image edge blurring and color distortion effectively. Since reconstruction of *G* component plays an important role in terms of image details and texture, this paper proposes using adapted color plane interpolation algorithm to render *G* component. Moreover, *R* and *B* components are rendered based on color difference rule and *G* component compensation to prevent color mutation and improve the color smoothness.

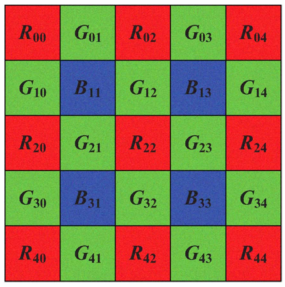

Figure 4. 5x5 reference samples of Bayer CFA

The proposed algorithm consists of four steps: the first step is reconstructing *G* component by using adapted color plane interpolation algorithm, the second step is calculating the differences between *R* and *G* component and between *B* and *G* component (*R-G* and *B-G*), and the third step is rendering full-resolution color difference channels [9] with edge adaptive and reconstructed *G* component, the last step is adding interpolated *G* to interpolated color difference channels to render *R* and *B* components. The overview of proposed algorithm is illustrated in Fig. 2. The strategy for reconstruction of *R*, *G* and *B* is described in Fig. 3.

A. Reconstruction of Green Color Component

G component is reconstructed by using adapted color plane interpolation at the first step. In this algorithm, pixel values of *R* and *B* are used to compensate for determining the gradients of *G* component in the 5x5 reference samples of Bayer CFA, shown in Fig. 4.

At red sample dot, R_{22} in Fig. 4 is taken for example. In order to interpolate the missing green pixel G_{22}, horizontal and vertical gradients of G_{22} are calculated with surrounding *R* and *G* components firstly, defined as

$$\Delta H = |-R_{20} + 2R_{22} - R_{24}| + |G_{21} - G_{23}|. \tag{6}$$

$$\Delta V = |-R_{02} + 2R_{22} - R_{42}| + |G_{12} - G_{32}|. \tag{7}$$

By comparing with the two gradients, the missing green pixel G_{22} can be estimated with surrounding green pixels and compensation of corresponding direction red pixels, as

$$G_{22} = \begin{cases} \dfrac{G_{21}+G_{23}}{2} + \dfrac{(-R_{20}+2R_{22}-R_{24})}{4} \ldots if\ \Delta H < \Delta V, \\ \dfrac{G_{12}+G_{32}}{2} + \dfrac{(-R_{02}+2R_{22}-R_{42})}{4} \ldots if\ \Delta H > \Delta V, \\ \dfrac{G_{21}+G_{23}+G_{12}+G_{32}}{4} \\ + \dfrac{(-R_{02}-R_{20}+4R_{22}-R_{24}-R_{42})}{8} . if\ \Delta H = \Delta V. \end{cases} \tag{8}$$

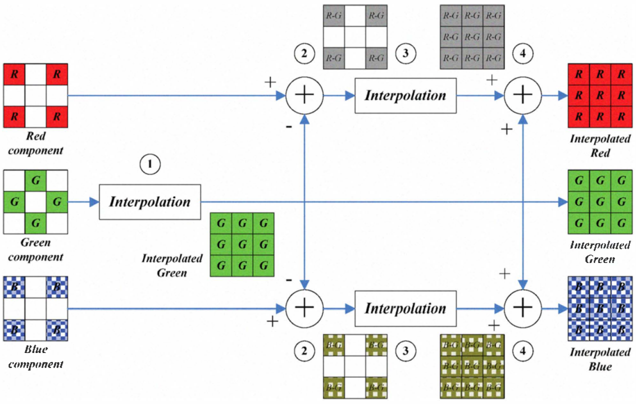

Figure 3. The strategy for reconstruction of *R*, *G* and *B*

At blue sample dot, the gradient of *G* component is determined by surrounding *B* and *G* components. The interpolation process of *G* component at this dot is the same as that at red sample dot.

Once missing pixels of *G* component are reconstructed, *R* and *B* components can be reconstructed by using important information of reconstructed green component.

B. Calculating Color Differences

According to the reconstructed full-resolution *G* channel, corresponding color differences (*R-G* and *B-G*) can be estimated at red pixel sample dot and blue pixel sample dot respectively. The result is shown in Fig. 5.

C. Reconstruction of Full-Resolution Color Differences

To produce visually more pleasing image and reduce color distortion more effectively, full-resolution color difference channels should be rendered based on edge adaptive and reconstructed green component.

At *R-G* dot, R_{22}-G_{22} in Fig. 5 is taken for example to estimate the missing color difference B_{22}-G_{22}. $H_{b\text{-}g}$ and $V_{b\text{-}g}$ figure southeast and southwest direction respectively at color difference dot *B-G*. Therefore, $H_{b\text{-}g}$ and $V_{b\text{-}g}$ of B_{22}-G_{22} should be calculated firstly, defined as

$$H_{b-g} = \left|(B_{11}-G_{11})-(B_{33}-G_{33})\right| + \left|-G_{11}+2G_{22}-G_{33}\right|. \tag{9}$$

$$V_{b-g} = \left|(B_{13}-G_{13})-(B_{31}-G_{31})\right| + \left|-G_{13}+2G_{22}-G_{31}\right|. \tag{10}$$

By comparing with the two gradients, the missing color difference B_{22}-G_{22} can be estimated with reconstructed green pixels and surrounding color differences *B-G*, as

$$B_{22}-G_{22} = \begin{cases} \dfrac{(B_{11}-G_{11})+(B_{33}-G_{33})}{2} + \dfrac{(-G_{11}+2G_{22}-G_{33})}{4} \quad .. H_{b-g} < V_{b-g} \\ \dfrac{(B_{13}-G_{13})+(B_{31}-G_{31})}{2} + \dfrac{(-G_{13}+2G_{22}-G_{31})}{4} \quad .. H_{b-g} > V_{b-g} \\ \dfrac{(B_{11}-G_{11})+(B_{13}-G_{13})+(B_{31}-G_{31})+(B_{33}-G_{33})}{4} \\ + \dfrac{(-G_{11}-G_{13}+4G_{22}-G_{31}-G_{33})}{8} \quad .... H_{b-g} = V_{b-g} \end{cases} \tag{11}$$

At *B-G* dot, *R-G* can be estimated by using surrounding *R-G* and reconstructed *G* component. The calculation process is the same as the reconstruction of *B-G*.

At green sample dot, G_{32} in Fig. 5 is taken for example to estimate the missing color differences R_{32}-G_{32} and B_{32}-G_{32}, as

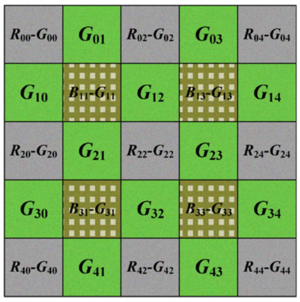

Figure 5. The result of corresponding color differences

$$R_{32}-G_{32} = \frac{(R_{22}-G_{22})+(R_{42}-G_{42})}{2} + \frac{(-G_{22}+2G_{32}-G_{42})}{4}. \tag{12}$$

$$B_{32}-G_{32} = \frac{(B_{31}-G_{31})+(B_{33}-G_{33})}{2} + \frac{(-G_{31}+2G_{32}-G_{33})}{4}. \tag{13}$$

By using the same method, full-resolution color difference channels *R-G* and *B-G* can be reconstructed.

D. Reconstruction of Red and Blue color Component

In the last step, interpolated *G* component is added to the reconstructed full-resolution color differences *R-G* and *B-G* in accordance with the corresponding position to obtain the interpolated *R* and *B* components.

IV. Simulation Results and Discussion

The proposed algorithm is tested using image lenna, lighthouse and Circular Zone Plate (512x512) defined as

$$f(x,y) = 128\sin(c_x x^2 + c_y y^2) + 128, \tag{14}$$
$$c_x = \pi/512, c_y = \pi/512, -256 \le x < 256, -256 \le y < 256.$$

In the simulation, we compare the proposed algorithm with six different methods: (1) *RGB* all use bilinear, (2) *G* uses edge adaptive and *RB* use bilinear, (3) *G* uses bilinear and *RB* use color difference bilinear, (4) *G* uses edge adaptive and *RB* use color difference bilinear, (5) *G* uses adapted color plane and *RB* use color difference bilinear, (6) *G* uses adapted color plane and *RB* use color difference edge adaptive.

From Fig. 6-8, we observe that as using method (1), although calculation is simple due to image edge isn't distinguished, rendered image quality is poor, including image edge blurring, texture performing inaccurately and color distorting seriously. As using method (2), image texture and edge perform better compared with method (1), illuminating that *G* estimation is important to render image edge and texture. As using method (3), image color improves greatly compared with method (1) and (2), illuminating that interpolation based on color differences can improve color effect. As using method (4), image texture, edge and color are all improved, color distortion is also reduced, especially in the low frequency area. However, there are some defective pixels due to sometimes detecting wrong edge directions. As using method (5), reconstructed image performs ideally in terms of texture, edge and color in the low frequency area, and also modifies the defective pixels. As using method (6), color difference

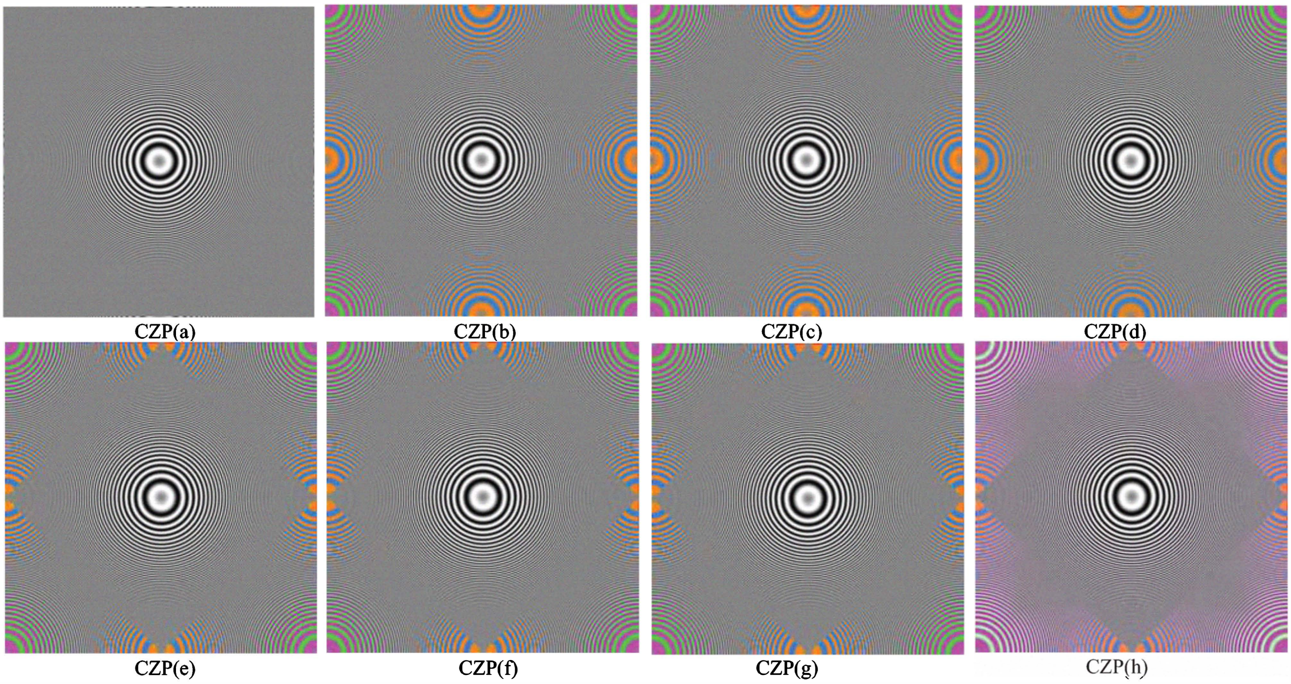

Figure 6. CZP image: (a) original, (b) method (1), (c) method (2), (d) method (3), (e) method (4), (f) method (5), (g) method (6), (h) proposed method

Figure 7. lenna images: (a) original, (b) method (1), (c) method (2), (d) method (3), (e) method (4), (f) method (5), (g) method (6), (h) proposed method

interpolation method is modified, and color distortion is slightly reduced in the high frequency area compared with method (5). As using the proposed algorithm, color distortion is reduced obviously, and image performs ideally in terms of texture, details, edge and color.

Peak signal-to-noise ratio (*PSNR*) and computation time are chosen as objective measure of performance, shown in Table 1 and Table 2. I^a and II^a represent the value of testing and interpolated image color component a ($a \in [r, g, b]$) respectively, M and N represent image width and height, *PSNR* is defined as

$$PSNR = 20\log_{10}\left[255 / \sqrt{\sum_x \sum_y (I^a(x,y) - II^a(x,y))^2 / (M \times N)}\right]. \quad (15)$$

All interpolation methods are simulated by *C* language on the computer with *AMD* Dual Core Processor, 2.71 *GHz CPU*

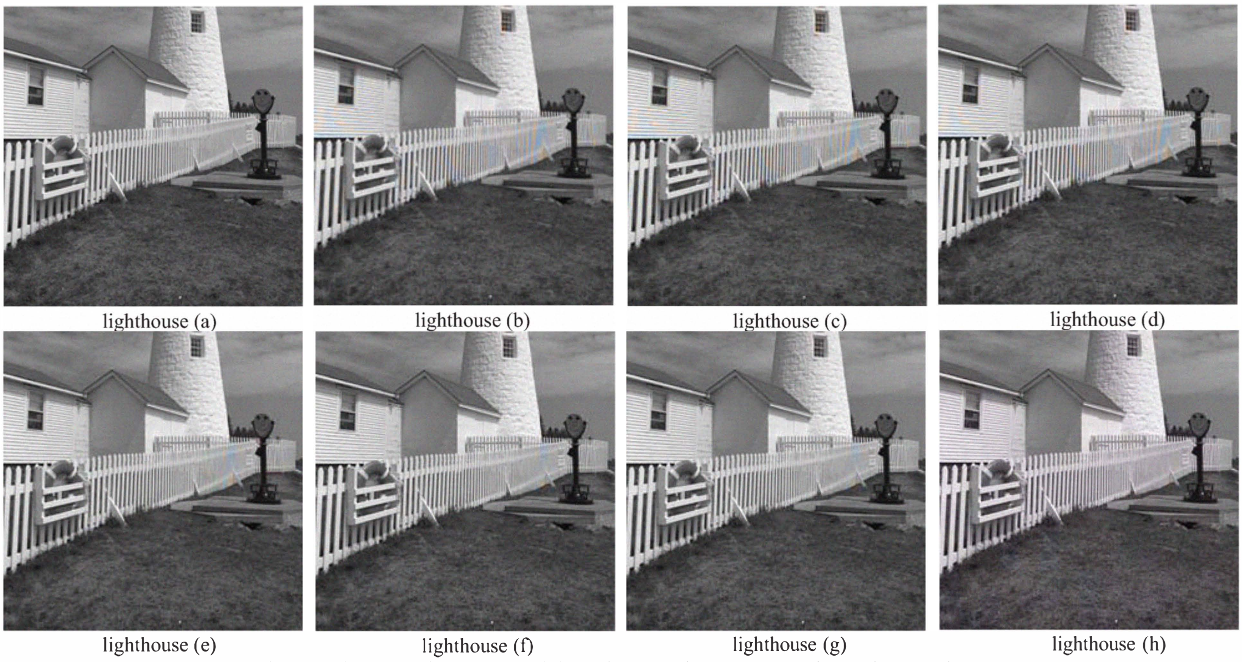

Figure 8. lighthouse images: (a) original, (b) method (1), (c) method (2), (d) method (3), (e) method (4),(f) method (5), (g) method (6), (h) proposed method

TABLE 1. PERFORMANCE COMPARISON – PSNR (DB)

method/image /component	CZP R	CZP G	CZP B	lenna R	lenna G	lenna B	lighthouse R	lighthouse G	lighthouse B
method (1)	8.74	11.05	8.74	34.96	35.48	30.94	25.48	30.34	25.36
method (2)	8.74	9.74	8.74	34.96	35.42	30.94	25.48	32.03	25.36
method (3)	11.12	11.05	11.12	34.74	35.48	32.06	29.31	30.34	29.29
method (4)	10.24	9.74	10.24	34.26	35.42	32.05	31.90	32.03	31.89
method (5)	11.15	10.98	11.15	34.24	37.16	32.97	36.57	37.24	36.53
method (6)	11.08	10.98	11.09	34.23	37.16	32.93	36.39	37.24	36.34
proposed method	12.59	10.98	12.63	35.44	37.16	33.42	36.74	37.24	36.71

TABLE 2. PERFORMANCE COMPARISON – COMPUTATION TIME (S)

method/testing image	CZP	lenna	lighthouse
method (1)	0.032	0.031	0.032
method (2)	0.047	0.046	0.046
method (3)	0.063	0.062	0.063
method (4)	0.062	0.062	0.063
method (5)	0.094	0.093	0.094
method (6)	0.078	0.078	0.078
proposed method	0.094	0.094	0.094

and 2 *GB* memory. From Table 1, our proposed algorithm improves the performance of the other methods. From Table 2, the computation time of bilinear algorithm is the shortest time, but image quality is the worst. The computation time of our proposed algorithm is not as short as that of bilinear algorithm, but interpolated image quality is the best, the algorithm complexity is low relatively.

I. CONCLUSION

In this paper, we have presented a Bayer CFA interpolation algorithm based on the combination of green component and color difference space. According to image space correlation and color correlation, *G* component is estimated firstly using adapted color plane method, and *R* and *B* components are reconstructed based on edge adaptive and interpolated *G* component. The proposed algorithm provides improved image visual quality upon the other six methods, and also performs high *PSNR* and low computation complexity.

In addition, the proposed algorithm can be improved to further reduce image color distortion, what will be considered in our future work.

REFERENCES

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

[2] J. Adams, K. Parulski and K. Spaulding, "Color processing in digital cameras," IEEE Micro, vol. 18, no. 6, pp. 20-30, Nov.-Dec. 1998.

[3] Keys, Robert. G., "Cubic Convolution Interpolation for Digital Image Processing," IEEE Transactions on Acoustic, Speech and Signal Processing, vol. 29, no. 6, pp. 1153-1160, Dec. 1981.

[4] Bo Gyu Jeong, Hyuk Su Kim, Suk Chan Kim and IL Kyu Eom, "Edge-Adaptive Demosaicking for Reducing Artifact along Line Edge," IEEE Congress on Image and Signal Processing, vol. 3, pp. 316-319, May 2008.

[5] Milos Klima and Petr Dostal, "Selected Problems of Color Image Demosaicing in Security Imaging," IEEE International Carnahan Conference on Security Technology, pp. 297-304, Oct. 2009.

[6] Ron Kimmel, "Demosaicing: Image reconstruction from color CCD samples," IEEE Transactions on Image Processing, vol. 8, no. 9, pp. 1221-1228, Sep. 1999.

[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, 1994.

[8] J. F. Hamiltion and J. E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," U.S. Patent 5, 629, 734, May 1997.

[9] Hirakawa, K. and Parks, T. W., "Adaptive Homogeneity-Directed Demosaicing Algorithm," IEEE Transaction on Image Processing, vol. 14, no. 3, pp. 360-369, Mar. 2005.