International Journal of Soft Computing 10 (2): 110-117, 2015
ISSN: 1816-9503

Color Filter Array Interpolation by Decision Based Edge Oriented Directional Filter in Digital Camera

[1]R. Niruban, [3]T. Sree Renga Raja and [2]M.N. Sinthiya
[1]Department of Electronics and Communication Engineering, Sathyabama University, Chennai, India
[2]M.E. Communication Systems, Anand Institute of Higher Technology, Chennai, India
[3]Department of Electrical and Electronics Engineering, Anna University (BIT Campus), Tiruchirapalli, India

Abstract: Digital cameras usually uses single sensor covered with Color Filter Array (CFA) to capture the image which reduces the cost and size of the camera. The CFA samples only single color value at each pixel location. To render the full color image, the missing two color components at each pixel is to be estimated. The process of estimating the missing component is called as demosaicing or color filter array interpolation. The proposed research uses the edge oriented filter named as edge strength filter which uses the edge information accurately. The missing components are interpolated by making the hard decision with the edge strength map. The reconstructed image still remains in color artifacts to reduce this problem, the refinement process is implemented. The low pass filter is used in refinement process to separate the low and high frequency components further interpolation technique is applied. The proposed technique produces the high quality image with less computational complexity.

Key words: Color artifacts, color filter array, demosaicing, edge strength filter, technique

INTRODUCTION

A digital imaging device such as digital cameras is become more popular nowadays. In last decades the CCD/CMOS (Charge Coupled Device/Complementary Metal Oxide Semiconductor) uses three sensors to capture the primary colors red, green and blue separately but the size and cost of the camera are high. To reduce the cost and size, now a days, digital cameras uses single chip CCD or CMOS sensors that consist of single image sensor covered with Color Filter Array (CFA) shown in Fig. 1. The CFA is an array pattern consists of a set of spectrally selective filters arranged in interleaved pattern consist of primary colors. Because of this arrangement each sensor samples only one color component at each pixel among the three primary color components such as red, green and blue. Since, there is only one color element available at each pixels (either green, red or blue), the missing two colors to be estimated from the adjacent pixels. This process is called as demosaicing or CFA interpolation.

Several CFA patterns are existing in digital camera, some of them are Bayer CFA pattern, Lukac pattern, Yamanka pattern, pseudo-random pattern (Lukac and Plataniotis, 2005). The reconstruction performance of a demosaicing process is closely related to the human visual system. The human visual system is more sensitive

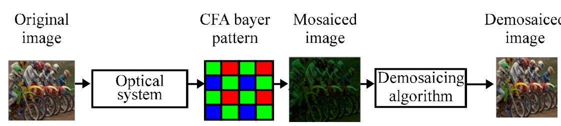

Fig. 1: A single sensor digital camera structure

to luminance than the chrominance, so that it is essential to accurately recover the luminance information and green component contributes about 72% of total luminance power (Kehtarnavaz *et al*., 2003). Because of this reason, Bayer CFA (Bayer, 1976) pattern is mostly used in demosaicing process, since Bayer pattern consist of 50% of green component and remaining 50% are equally shared by red and blue component. The CFA Bayer pattern is shown in Fig. 2.

There are different interpolation techniques applied in demosaicing. The earliest proposed interpolation method for images namely nearest neighbor interpolation, linear interpolation (Losson and Yang, 2010) cubic spline interpolation, bilinear interpolation (Gonzalez *et al*., 2004) are not able to provide good performance. Nearest neighbor technique produces color errors, mainly along edges. Linear interpolation introduces the false color and zipper effect which results in poor quality of image. Cubic interpolation also results in zipper effect and false color. These techniques are failed to reduce blurring effect and

Corresponding Author: R. Niruban, Department of Electronics and Communication Engineering, Sathyabama University, Chennai, India

zipper effect along edges. In the last decades, many demosaicing algorithms have been proposed. These algorithms are classified into two categories: iterative and non-iterative. In general, the performance of Iterative algorithm is better than the Non-Iterative algorithm.

In Iterative algorithm like highly effective iterative demosaicing (Su, 2006) and successive approximation (Li, 2005), the Demosaicing algorithm is classified into initial stage and iterative stage. Constant color difference rule (Kimmel, 1999) is used in their iterative stage but fails to reduce the zipper effect along edges. Gunturk *et al.* (2002) uses interpolation technique with high frequency sub-band correlation. It updates the high frequency wavelet coefficients of red and blue channel with the help of green channel. Lu and Pan (2003) uses filter bank technique for directional interpolation and non-linear iterative procedure to reduce aliasing, misguidance artifacts. By Lukac *et al.* (2004) researchers used Edge Adaptive algorithm to determine pattern of line edges, it reduces the artifact along edges. In ECI (Menon *et al.*, 2007), the color difference between the luminance and

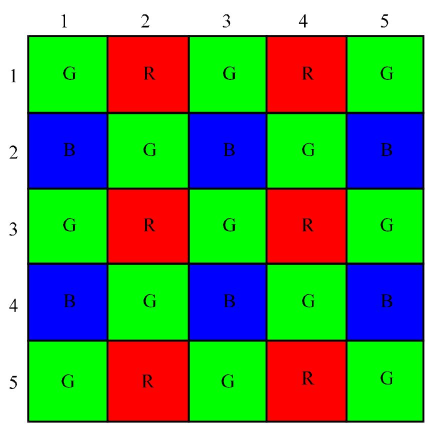

Fig. 2: Bayer pattern

chrominance component are determined. But, it introduces false color in the image. Likewise several methods are proposed but most of the algorithms fail to utilize the edge information. In this proposed research, the edge information is effectively utilized for the Interpolation algorithm.

In this proposed research, adaptive color plane interpolation, bilinear interpolation and constant color difference rules (Laroche and Prescott, 1994) were used. However, the resulting image may still contains some artifacts. To reduce such artifacts and improve image quality, refinement process is further applied.

MATERIALS AND METHODS

The proposed algorithm architecture is shown in Fig. 3. Initially, the input image is modified into Bayer CFA pattern, then the edge strength of the image is determined using the edge strength filter which is used to take the decision for further interpolation technique. The green channel interpolation is done using the vertical and horizontal cost. Based on the estimated green channel, the missing red and blue channels are estimated. Further to increase the performance, the refinement process is implemented.

Edge strength filter: Edge information is the important parameter in the image processing. Many edge detection algorithm likes Canny filter and Sobel filter are used but it detects only the presence of edge structure and it does not provides information such as sharpness, brightness and luminance about the pixels. Most of the demosaicing algorithm uses constant color difference assumption which fails across edges (Hirakawa and Parks, 2005).

If edge information are utilize effectively then the non-correlated color difference can be easily avoided and

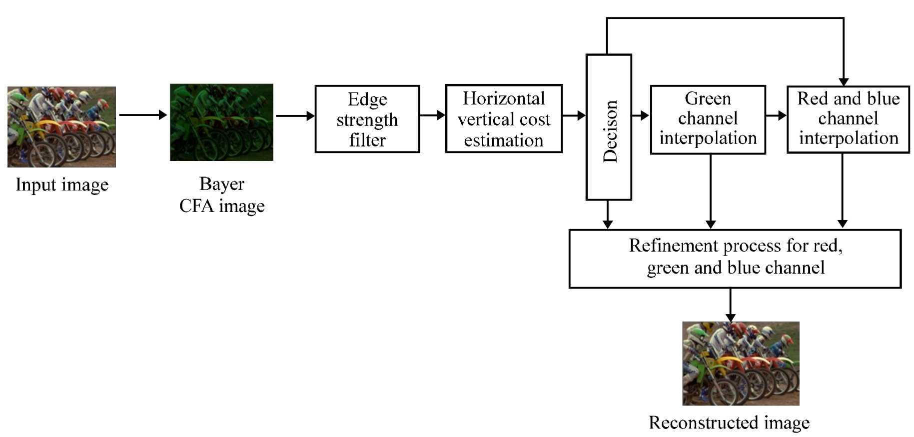

Fig. 3: Block diagram of proposed technique

this increases the demosaicing performance. For this purpose, the proposed algorithm introduces a new filter called edge strength filter which provides local and orientation free luminance transition information. For the given 5×5 image size (Fig. 2), the edge strength for pixel location G13 could be equation as:

$$ES_{G_{13}} = \frac{|G_7\text{-}G_{19}|}{2} + \frac{|G_9\text{-}G_{17}|}{2} + |B_8\text{-}B_{18}| + |R_{12}\text{-}R_{14}| \quad (1)$$

where, $ES_{G_{13}}$ represents the edge strength at pixel location $P_{G_{13}}$. Likewise by applying the edge strength for all available pixels in the image, edge strength map of the input can be obtained. Since, mosaicked image has only one color at each pixel, it does not have complete luminance information about that pixel. Because of this reason, the edge strength filter can be applicable for only to the mosaicked image.

Edge strength at each pixel can be determined by considering surrounding pixels. For instance, for the green centre pixel case, the diagonal difference will come from the green channel and the rest from the red and blue channel. Thus, the edge strength map is obtained by calculating edge strength at each pixel which is used for determination horizontal and vertical cost.

Green channel interpolation: The green channel in the Bayer pattern contains more information because it is measured at a high sampling rate than the red and blue channel. Since, the green channel consists of more information, it is necessary to determine the green channel accurately. Various interpolation techniques are introduced to determine the missing green component. Edge directed interpolation (Adams and Hamilton, 1996) detects the local spatial information in the neighboring pixels. Su (2006) introduced weighted edge interpolation to determine green channel which is based on the gradient and weight. The effective color interpolation (Pei and Tam, 2003) method determines the image spectral correlation using bilinear interpolation and it uses the average of neighboring color difference value to estimate the missing green component. In the proposed algorithm, for green channel interpolation, hard decision is made by using an edge strength filter. For this purpose the horizontal and vertical difference cost are determined from the edge strength filter. The green channel interpolation is done either horizontally or vertically, depends upon the difference cost which is equation as:

$$\begin{aligned} HC_{i,j} &= \sum_{i=1}^{m}\left(\sum_{j=1}^{n-1}(ES_{i,j} - ES_{i,j+1})\right) \\ VC_{i,j} &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}(ES_{i,j} - ES_{i+1,j})\right) \end{aligned} \quad (2)$$

Where:

$ES_{i,j}$ = The edge strength filter outputs
$HC_{i,j}$ and $VC_{i,j}$ = The horizontal and vertical difference cost, respectively at pixel location (i, j)

This cost estimation is also used in red and blue plane interpolation and in the refinement process.

The horizontal difference cost estimation is done horizontally (considering pixels in same row) and vertical difference cost estimation is done vertically (considering pixels in same column). The target pixel is marked horizontal if the horizontal difference cost is less than the vertical difference cost and vice versa. After all the pixels are marked, green channel interpolation is done based on the final direction label. The green channel interpolation in red location (in pixel location (3, 3)) is given as:

$$\begin{aligned} \tilde{G}_{3,3} &= R_{3,3} + \frac{\tilde{G}^H_{3,3}\text{-}R_{3,3}}{2} + \frac{\tilde{G}^H_{3,2}\text{-}R_{3,2}}{4} + \frac{\tilde{G}^H_{3,4}\text{-}R_{3,4}}{4} \quad \text{if } HC_{i,j} < VC_{i,j} \\ \tilde{G}_{3,3} &= R_{3,3} + \frac{\tilde{G}^V_{3,3}\text{-}R_{3,3}}{2} + \frac{\tilde{G}^V_{2,3}\text{-}R_{2,3}}{4} + \frac{\tilde{G}^V_{4,3}\text{-}R_{4,3}}{4} \quad \text{if } VC_{i,j} < HC_{i,j} \end{aligned} \quad (3)$$

Where $\tilde{G}^H, \tilde{R}^H, \tilde{G}^V, \tilde{R}^V$ are the directional estimation. The green channel interpolation in blue location can be determined in similar way be replacing R by B. The directional estimations can be determined using the adaptive color interpolation. This algorithm is used to determine the missing red and blue component in green pixel and vice versa. The directional estimation is calculated as follow:

$$\begin{aligned} \tilde{G}^H_{3,4} &= \frac{G_{3,3} + G_{3,5}}{2} + \frac{2 \times R_{3,4} - R_{3,2} - R_{3,6}}{4} \\ \tilde{G}^V_{3,4} &= \frac{G_{2,4} + G_{4,4}}{2} + \frac{2 \times R_{3,4} - R_{1,4} - R_{5,4}}{4} \\ \tilde{R}^H_{3,3} &= \frac{R_{3,2} + R_{3,4}}{2} + \frac{2 \times G_{3,3} - G_{3,1} - G_{3,5}}{4} \end{aligned} \quad (4)$$

Green channel interpolation at blue pixel location can be performed by replacing R with B in above equations. The estimated green channel is used to determine the missing red and blue channel.

Red and blue channel interpolation: After the green component has been reconstructed, next step is to interpolate the missing red and blue components. For the reconstruction of red and blue component, Bayer CFA, interpolated green channel and the horizontal and vertical difference costs are used. In (Menon *et al*., 2007; Kuno *et al*., 1999; Kimmel, 1999; Cok, 1987) color difference, B-G and R-G are used for estimation of red and blue components instead of using R and B directly. Generally in most papers, red and blue interpolation algorithm uses constant color difference rule and bilinear interpolation, sometimes with a small adjustment for the reconstruction of red in blue component and vice versa. For the reconstruction of red and blue component in green pixel, bilinear interpolation technique is used because in most of the approaches the error is higher in the interpolated red and blue pixels in green component. This technique yields good performance with low computational complexity and cost.

For missing red and blue estimations at green pixels, bilinear interpolation over color difference estimation technique is used. The missing red in green pixel are formulated as:

$$\begin{aligned}\tilde{R}_{3,3} &= G_{3,3} + \frac{(R_{3,2} - \tilde{G}_{3,2}) + (R_{3,4} - \tilde{G}_{3,4})}{2} \\ \tilde{R}_{4,4} &= G_{4,4} + \frac{(R_{3,4} - \tilde{G}_{3,4}) + (R_{5,4} - \tilde{G}_{5,4})}{2}\end{aligned} \tag{5}$$

Where, $\tilde{G}$ is the estimated green pixel value and the blue channel interpolation at green pixel is performed by replacing R by B in above equations. The pixel consideration of missing red at green pixel is shown in Fig. 4. The considering pixels are denoted in colors.

Based on the estimated red and blue component, vertical and horizontal difference cost estimation, the blue channel interpolation in red component and red channel interpolation in blue component are done. The missing blue component in red location are estimated as:

If $HC_{3,4} < VC_{3,4}$

Then,

$$\tilde{B}_{3,4} = R_{3,4} + \frac{(\tilde{B}_{2,4} - \tilde{R}_{2,4}) + (\tilde{B}_{4,4} - \tilde{R}_{4,4})}{2} \tag{6}$$

Else,

$$\tilde{B}_{3,4} = R_{3,4} + \frac{(\tilde{B}_{3,3} - \tilde{R}_{3,3}) + (\tilde{B}_{3,5} - \tilde{R}_{3,5})}{2}$$

Where:

HC and VC = The horizontal and vertical difference cost estimated from the edge strength map

$\tilde{R}$ and $\tilde{B}$ = The estimated red and blue component in green pixels

For the interpolation of red value in blue pixels, the same strategy is applied. By the end of these steps, separate full red, green and blue planes are reconstructed. By combining these planes a full color image can be obtained. But, still the reconstructed image contains some artifacts. To reduce the artifact and improve the image quality further refinement process is implemented.

Refinement process: The refinement process in demosaicing is done after the determination of missing components. It is implemented to reduce the artifacts and to increase the performance. The refinement process in this proposed technique is carried out with the estimated color values and the difference cost. In the proposed algorithm for refinement process the filter design approximation is implemented. The color artifact generally remains in high frequency component. In order to reduce this artifact, high band inter-channel correlation is determined in three color plane which separates the high and low frequency component. The low frequency components are unchanged because it is less correlated and high frequency components are replaced with Bayer known components. For example, the green value is decomposed as:

$$G_{i,j} = G^L_{i,j} + G^H_{i,j} \tag{7}$$

Where, G^L and G^H represents the low frequency and high frequency components, respectively. Similarly, red and blue values are corrected as:

$$\begin{aligned}R_{i,j} &= R^L_{i,j} + G^H_{i,j} \\ B_{i,j} &= B^L_{i,j} + G^H_{i,j}\end{aligned} \tag{8}$$

The low frequency components are selected using a low pass filter and high frequency components are calculated by subtracting the low frequency values. The refinement of green pixel is done with the FIR low pass filter and the frequency is value for FIR filter is given as:

$$F = \begin{bmatrix}\frac{1}{3} & \frac{1}{3} & \frac{1}{3}\end{bmatrix} \tag{9}$$

The difference cost is also assigned for refinement process. The high frequency and low frequency components are obtained by applying filter and then the green high frequency values are replaced with red and blue high frequency values. Thus, the green component is reconstructed. The algorithm for green channel refinement, the color difference estimation is calculated and it is formulated as:

$$\begin{aligned}R_{less} &= R - G \\ B_{less} &= B - G \\ G_{less} &= R - B\end{aligned} \tag{10}$$

For the green channel refinement at red pixel, the filter is applied to the color difference estimation and it is given as:

If $HC_{i,j} < VC_{i,j}$

Then,

$$\hat{G}(i,j) = F \times R_{less}(i, j-1 : j+1) \tag{11}$$

Else,

$$\hat{G}(i,j) = F \times R_{less}(i-1 : i+1, j)$$

Similarly, the green channel refinement in blue channel is done by replacing R by B. The estimated red channel is further refined by the color difference interpolation and by considering neighboring pixels. The red channel refinement in green pixel is formulated as:

$$\hat{R}_{i,j} = G_{i,j} + R'_{i,j} \tag{12}$$

Where:

$$R'_{i,j} = \frac{R_{less}(i-1,j) + R_{less}(i+1,j)}{2} \tag{13}$$

The refinement of estimated blue component in green pixel can be done in similar way. Finally, the refinement of blue in red pixel and red in blue pixel are implemented as follow:

If $HC_{i,j} < VC_{i,j}$

Then,

$$\hat{R}(i,j) = F \times G_{less}(i, j-1 : j+1) \tag{14}$$

Else,

$$\hat{R}(i,j) = F \times G_{less}(i-1 : i+1, j)$$

The blue channel refinement at red pixel is done by replacing R as B in above equation. At the end of this algorithm, the refined separate red, green and blue color planes are obtained and by combining the three planes, the full color image with high performance can obtained. The refinement process improves the quality of the image and reduces the color artifacts.

RESULTS AND DISCUSSION

The proposed algorithm reduces the color artifacts and it has less computational complexity. This proposed research is experimentally tested using the programming language MATLAB. For this purpose the 24 Kodak test images (Su and Kao, 2009) are considered which are in TIFF format with size 512×768 which are shown in Fig. 4.

The performance of the proposed method is compared with four existing methods such as bilinear interpolation (Lukac and Plataniotis, 2004) Edge Based algorithm (Pekkucuksen and Altunbasak, 2012), sub-band synthesis (Chang and Tan, 2004) and wavelet with spatial refinement (Zhang and Wu, 2005). And the performances are measured by the factors CPSNR and PSNR. Table 1 shows the CPSNR comparison of the proposed algorithm with three different existing algorithms. The bilinear interpolation is the oldest technique which introduces the color artifacts usually along the edges (Muresan and Parks, 2005). The edge based interpolation reduces the false color at the edge but the quality of image is low. The Wavelet Based algorithm uses the refinement process to reduce the zipper effect and color artifacts but some error remains in the image.

Performance measure: The performance of the demosaicing is measured using different factor such as CPSNR (Color Peak Signal to Noise Ratio), PSNR (Peak Signal to Noise Ratio) and ΔE^*_{ab}. In image processing pipeline, most of the techniques uses CPSNR and PSNR

Fig. 4: 24 Kodak test images used for experiment

Table 1: CPSNR comparisons of proposed algorithm with existing algorithm

Images	Bilinear interpolation	Edge based interpolation	Subband synthesis	Wavelet with spatial refinement	Proposed algorithm
1	30.7730	31.2458	39.5224	40.6942	43.2781
2	36.6873	37.0425	39.5198	41.3527	45.7990
3	37.6775	38.0647	43.3218	43.9651	47.9754
4	37.4213	37.8047	41.4224	43.4641	46.1247
5	31.2157	31.8233	38.6369	40.5972	43.0823
6	32.1414	32.5964	40.2550	41.9364	44.6445
7	37.1797	37.6152	42.1956	44.6154	47.1192
8	28.2879	28.7457	36.0158	38.9305	42.7918
9	36.4896	36.9292	43.2609	45.2354	48.6827
10	36.3952	36.8716	43.5337	45.2596	48.4436
11	33.6645	34.1372	41.0060	42.9933	44.3100
12	36.9695	37.3912	43.5063	46.4256	50.8701
13	28.6113	29.1087	37.2893	37.3747	41.3609
14	33.4690	33.9611	36.7377	38.7378	43.8398
15	35.4486	35.8213	39.8739	42.0606	45.9462
16	35.2366	35.6156	43.2864	44.6285	44.9031
17	36.6810	37.2463	43.6912	44.4829	47.2387
18	32.5620	33.0611	38.9911	40.0813	43.3361
19	32.7829	33.2797	40.5543	43.9703	45.8365
20	34.6422	34.9845	41.2988	42.7135	47.2216
21	33.0627	33.5363	40.9836	41.8151	45.0262
22	34.9277	35.3907	39.4920	40.9682	45.0060
23	38.3544	38.7374	42.7698	40.6942	48.4513
24	31.4110	31.8746	37.6036	41.3527	44.8848

Table 2: PSNR comparisons of proposed algorithm with existing algorithm

	Bilinear			Edge based			Subband synthesis			Wavelet with spatial refinement			Proposed algorithm		
Image	R	G	B	R	G	B	R	G	B	R	G	B	R	G	B
1	29.741	34.286	29.678	29.741	39.442	29.678	38.881	40.591	39.272	39.811	42.617	40.154	42.7830	44.1777	40.8723
2	35.022	40.797	36.100	35.022	44.854	36.100	36.524	43.533	41.665	39.147	44.034	42.327	42.7328	41.4116	39.5293
3	35.942	41.591	37.252	35.942	45.912	37.252	42.825	45.853	42.113	43.217	46.289	43.078	44.1311	45.9738	40.9862
4	37.106	41.126	35.663	37.106	45.051	35.663	37.928	46.396	44.927	40.751	46.799	45.211	40.6203	42.8221	41.1771
5	30.126	34.082	30.430	30.126	40.240	30.430	37.853	41.500	37.547	39.959	42.717	39.705	37.9554	40.0960	37.3456
6	30.803	35.699	31.355	30.803	40.606	31.355	39.616	41.921	39.609	41.342	44.180	40.943	39.9064	42.8070	38.1181
7	35.514	41.092	36.662	35.514	46.389	36.662	42.159	45.984	40.226	44.156	47.043	43.420	42.9876	44.8800	40.3448
8	27.104	32.181	27.220	27.104	37.959	27.220	34.998	37.820	35.697	37.959	41.183	38.307	37.6731	39.6122	37.1484
9	36.102	40.438	34.717	36.102	45.917	34.717	42.785	45.225	42.298	44.552	46.919	44.624	42.9979	46.1785	43.0589
10	36.272	40.017	34.528	36.272	45.509	34.528	41.980	46.187	43.416	44.351	47.456	44.598	42.4608	45.7168	43.4385
11	32.269	37.035	33.016	32.269	41.880	33.016	39.313	43.338	41.279	41.738	45.071	42.801	38.8781	41.7316	38.6279
12	35.266	41.164	36.404	35.266	36.404	36.404	42.376	46.138	42.872	45.619	48.716	45.612	45.4448	48.5239	45.0743
13	27.650	31.312	27.764	27.650	35.342	27.764	37.443	38.584	36.173	37.128	39.053	36.363	36.8073	38.1165	35.3016
14	32.123	36.675	32.828	32.123	41.524	32.828	34.990	40.838	36.262	37.767	41.174	38.021	39.3696	40.9527	37.5489
15	34.216	39.315	34.439	34.216	43.306	34.439	37.339	43.381	41.055	39.927	44.670	42.936	40.5561	42.7442	40.5677
16	33.815	39.277	34.378	33.815	43.662	34.378	43.143	44.836	42.260	43.632	46.928	44.008	39.6464	41.8206	39.3261
17	36.205	39.248	35.440	36.205	43.949	35.440	43.318	45.433	42.754	43.964	46.035	43.786	42.1666	43.9587	41.6077
18	31.906	35.128	31.478	31.906	38.980	31.478	37.865	41.464	38.428	39.039	41.776	39.860	38.1413	40.0974	37.7917
19	31.592	36.521	31.772	31.592	42.845	31.772	39.476	42.313	40.337	42.768	45.944	43.773	40.8361	42.7492	40.0367
20	33.838	39.032	33.080	33.838	43.287	33.080	41.613	44.279	39.349	42.673	45.111	41.206	42.4339	44.1826	41.2278
21	31.892	36.279	32.221	31.892	40.838	32.221	40.720	42.780	39.923	41.446	43.773	40.762	40.2222	41.9771	39.0499
22	33.866	38.111	33.983	33.866	42.446	33.983	38.562	42.531	38.472	39.964	43.059	40.467	40.6767	42.1036	38.6207
23	36.114	42.556	38.598	36.114	47.324	38.598	41.226	46.550	42.145	42.076	47.010	44.159	43.1748	46.5210	42.3552
24	31.062	34.104	30.016	31.062	37.717	30.016	37.292	40.023	36.291	37.664	40.261	36.797	40.2503	42.2425	38.5954

(Adams, 1995) calculation to measure their performance. To determine the Color Peak Signal Noise Ratio (CPSNR), initially mean square error value is to be determined. It is calculated by taking the sum of squared difference between the input image and the reconstructed image and then dividing it by the total size of the image.

Table 1 shows the comparison of CPSNR values of proposed technique with different existing interpolation techniques. The performance of proposed technique is high when compared to the performance of existing algorithms.

The equation for calculating the CPSNR is given as follow:

$$I_{CPSNR} = 10\log_{10}\frac{255^2}{I_{CMSE}} \quad (15)$$

where, I_{CMSE} is a color mean square error value of the image. And its expression is given as:

$$I_{CMSE} = \sum_{i,j}\frac{[I_{in}(i,j) - I_{recons}(i,j)]^2}{3\times M\times N} \quad (16)$$

Where:
$I_{in}(i, j)$ and
$I_{recons}(i, j)$ = The original image and reconstructed image value, respectively at each pixels
M and N = The width and height of the image

The image with higher CPSNR indicates the high quality image. And the mean square error values are low for the high quality image. The decreased mean square error images have higher performance. The Peak Signal to Noise Ratio (PSNR) calculation is formulated as:

$$I_{PSNR} = 10\log_{10}\frac{255^2}{I_{MSE}} \quad (17)$$

where, I_{MSE} is a mean square error value of the image. And its expression is given as:

$$I_{MSE} = \sum_{i,j}\frac{[I_{in}(i, j) - I_{recons}(i, j)]^2}{M \times N} \quad (18)$$

Table 2 shows the comparison of PSNR values of proposed technique with different existing interpolation techniques. The performance of proposed technique is high when compared to the performance of existing algorithms.

CONCLUSION

This study provides the edge based interpolation technique with refinement process. The color artifacts and false colors are some common artifacts in the Demosaicing algorithm. The proposed technique produces the low complexity, high quality image with reduced artifacts. The edge strength filter is used in the proposed algorithm which utilizes the edge information for Interpolation algorithm. To reduce the color artifacts, the refinement process is implemented which uses the FIR low pass filter to separate the low and high frequency component. The errors in the high frequency components are replaced by Bayer known components. Thus, the performance of the Proposed algorithm is increased. Further the future research will focus with improvement in the Demosaicing algorithm and the edge strength can be used in other image processing problem, it can reduce the artifacts and other problems.

REFERENCES

Adams, J.E. and J.F. Hamilton, 1996. Adaptive color plan interpolation in single sensor color electronic camera. US Patent, 5506619, April 1996. http://www.freepatentsonline.com/5506619.html.

Adams, J.E., 1995. Interactions between color plane interpolation and other image processing functions in electronic photography. Proc. SPIE Technol. Electron. Imag. Syst., 2416: 144-151.

Bayer, B.E., 1976. Color imaging array. US Patent, 3971065 A, July 1976. http://www.google.com/patents/US3971065.

Chang, L. and Y.P. Tan, 2004. Effective use of spatial and spectral correlations for color filter array demosaicking. IEEE Trans. Consum. Electron., 50: 355-365.

Cok, D.R., 1987. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal. US Patent, 4642678 A, February 1987. http://www.google.com/patents/US4642678.

Gonzalez, R.C., R.E. Woods and S.L. Eddins, 2004. Digital Image Processing Using Matlab. Prentice Hall, New Jersey, USA., ISBN-13: 9780982085400, pp: 609.

Gunturk, B.K., Y. Altunbasak and R.M. Mersereau, 2002. Color plane interpolation using alternating projections. IEEE Trans. Image Process., 11: 997-1013.

Hirakawa, K. and T.W. Parks, 2005. Adaptive homogeneity-directed demosaicing algorithm. IEEE Trans. Image Process., 14: 360-369.

Kehtarnavaz, N.D., H.J. Oh and Y. Yoo, 2003. Color filter array interpolation using color correlations and directional derivatives. Electron. Imag. J., 12: 621-632.

Kimmel, R., 1999. Demosaicing: Image reconstruction from color CCD samples. IEEE Trans. Image Process., 8: 1221-1228.

Kuno, T., H. Sugiura and N. Matoba, 1999. New interpolation method using discriminated color correlation for digital still cameras. IEEE Trans. Consum. Electron., 45: 259-267.

Laroche, C.A. and M.A. Prescott, 1994. Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients. US Patent, 5373322 A, December 1994. http://www.google.com/patents/US5373322.

Li, X., 2005. Demosaicing by successive Approximation. IEEE Trans. Image Process., 14: 370-379.

Losson, O.L. and M.Y. Yang, 2010. Comparison of color demosaicing methods. Adv. Imag. Electron Phys., 162: 173-265.

Lu, W. and Y.P. Tan, 2003. Color filter array demosaicking: New method and performance measures. IEEE Trans. Image Process., 12: 1194-1210.

Lukac, R. and K.N. Plataniotis, 2004. A normalized model for color-ratio based demosaicking schemes. Proceedings of the International Conference on Image Processing, Volume 3, October 24-27, 2004, Singapore, pp: 1657-1660.

Lukac, R. and K.N. Plataniotis, 2005. Color filter array: Performance analysis. IEEE Trans. Consum. Electron., 15: 1260-1267.

Lukac, R., K.N. Plataniotis, D. Hatzinakos and M. Aleksic, 2004. A novel cost effective demosaicking approach. IEEE Trans. Consum. Electron., 50: 256-261.

Menon, D., S. Andriani and G. Calvagno, 2007. Demosaicing with directional filtering and a posteriori decision. IEEE Trans. Image Process., 16: 132-141.

Muresan, D.D. and T.W. Parks, 2005. Demosaicing using optimal recovery. IEEE Trans. Image Process., 14: 267-278.

Pei, S.C. and I.K. Tam, 2003. Effective color interpolation in CCD color filter arrays using signal correlation. IEEE Trans. Circuits Syst. Video Technol., 13: 503-513.

Pekkucuksen, I. and Y. Altunbasak, 2012. Edge strength filter based color filter array interpolation. IEEE Trans. Image Process, 21: 393-397.

Su, C.Y. and W.C. Kao, 2009. Effective demosaicing using subband correlation. IEEE Trans. Consum. Electron., 55: 199-204.

Su, C.Y., 2006. Highly effective iterative demosaicing using weighted-edge and color-difference interpolations. IEEE Trans. Consum. Electron., 52: 639-646.

Zhang, D. and X. Wu, 2005. Color demosaicking via directional linear minimum mean square-error estimation. IEEE Trans. Image Process., 14: 2167-2178.