False Color Suppression in Demosaiced Color Images

Jayanta Mukherje
Department of Computer Science & Engg.
IndianInstituteofTechnology
Kharagpur, INDIA-721302
jay@cse.iitkgp.ernet.in

Manfred K. Lan
Institute for Human--Machine Communication
TechnicalUniversityofMunich,
D-80290, Munich, Germany
lang@ei.mmk.tum.de

S. K. Mitra
Department of Electrical & Computer Engg.
UniversityofCalifornia
Santa Barbara, 93106, USA
mitra@ece.ucsb.edu

Abstract

In a single-chip digital color camera, a color filter array (CFA) is used to obtain sampled spectral components (red, green and blue) in an interleaved fashion. A color demosaicing operation is then carried out to determine the missing spectral components at every location. One of the problems in color demosaicing is that many of the interpolated images are affected by colored artifacts near the edges creating false colors. The problem is more severe if the edges are achromatic. In this paper we propose the use of median filtering for suppressing this phenomenon. We have considered a set of existing interpolation algorithms and presented their performances in interpolating mosaiced patterns. Next we have carried out median filtering of the chrominance components of the demosaiced images. In each case, the post-processing has remarkably improved the quality of the reconstructions. The observations are verified by both quantitative measures for expressing the quality of reconstruction as well as by visual examinations of the reconstructed images.

1. Introduction

For the last two decades, considerable attention has been drawn to the problem of color demosaicing. Color demosaicing is required for the images captured by a single-sensor camera using a color filter array (CFA). In the captured image (also called as *mosaiced pattern*) only single spectral components are available. There are different checkered patterns used for the purpose of filtering a single spectral component. One of the most popular ones is the widely used Bayer pattern shown in Figure 1.

As can be seen from Figure 1, at every pixel location of the image only one of the three color components (red (R), green (G) or blue (B)) is available. Hence for a full color image, it is necessary to compute the other two missing color components at every pixel location. The operation for obtaining these missing color pixel values from the sampled color pixel data is commonly known as *color interpolation* or *color demosaicing*.

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

Figure 1: The Bayer pattern.

One of the major concerns of this interpolation task is to keep the hardware cost as well as the computation time as small as possible. This is to make the digital color imaging (for both still and video images) cost-effective and technologically viable. Hence the methods usually employed in practice are based on low storage requirement (storing two or three rows of the captured image during scanning and processing) and simple computations (such as averaging [1], [23], copying of neighboring pixels [20], [21], [24], convolving with a given mask [8], [16], decisions based on simple comparisons [10], [5], [3] etc.). But the challenge is to get the image quality as best as possible with the available resources at hand.

Usually, in good interpolation techniques one of the following two principles are followed:

1. Interpolate with the pixels lying with the low gradient directions [10],[5], and

2. Use the homogeneity of the cross-ratios of different spectral components around a small neighborhood [1],[4].

Other approaches combining the above two principles have also been advanced. Kimmel [15] proposed one such algorithm by weighting the cross-ratios with the gradient information around the neighborhood of a pixel. In a similar approach Hur and Kang [11] also used the gradient information for computing weighted averages of the cross

ratios and recovering the spectral component from it. In [17] also, relatively faster algorithms combining the above two principles are designed for interpolating color images. The approaches are adopted for processing with two rows or three rows of the mosaiced patterns. Accordingly, they are called as Two-Line Algorithm (2L) and Three-Line Algorithm (3L).

One of the problems associated with color interpolation is the appearances of false colors or colored artifacts near the edges. The problem gets more severe if the edges are achromatic (of grey shades). A typical example is demonstrated in Figure 2. The original image is shown in Figure 2(a). A mosaiced pattern following the Bayer CFA is generated from it and then interpolated using the bilinear interpolation algorithm. The interpolated image is shown in Figure 2(b). One could see that the reconstructed image is severely affected by colored artifacts in the walls of the house and also in the white fencing.

(a) Origina (b) BI

Figure 2: Examples of false color edges in the reconstructed image

There are also a few efforts in suppressing the false colors near the edges. Earlier Kimmel [15] had adopted an iterative cross-ratio adjustment policy for mutually correcting the spectral components. Similar policy has been also adopted by Hur and Kang [11]. They have also used the color edge information for localizing these corrections. Kimmel [15] also carried out an inverse diffusion process for the further enhancement of the images. There are also efforts in using Markov Random Field processing [6], [18] for enhancing the image quality. However, all these processes are computationally intensive and also expensive for hardware realizations. Besides there are also efforts [22],[19] in using different types of color filter arrays (CFAs) for suppressing false colored edges. In a recent work Gunturk et. al. [9] have exploited interchannel correlation of different spectral components for reconstructing missing pixels through an iterative adjustment. They have performed subband decomposition of different color planes and used the higher subbands of those planes whose true pixel values are available (e.g. CFA values at red and blue masks) during the synthesis of non CFA green pixels. Similarly, red and blue pixel values are reconstructed by synthesising their low-low subband with higher subbands of interpolated green values. The process is repeated for a number of times by projecting back 'true' pixel values of red (blue) in the synthesised image. The algorithm (denoted by AP in the Table 1) has provided interpolated images of excellent quality and subsequently false colored artifacts are also greatly reduced by this process.

In this work we have used median filtering for suppressing false colors in the demosaiced images. Median filtering [12] is a well known operation for removing shot or impulsive noises in an image. It has the good edge preserving characteristics. In fact some of the interpolation algorithms [7], [14] have used median filtering for estimating the missing spectral components. In this contribution we use it as a post-processing operation for improving the image quality and also suppressing false colored artifacts at the end. Interestingly, in our methods median filtering is applied globally at every pixel of the image. There is no requirement of localizing the false color occurrences. This makes our computation much simpler and faster than the existing approaches for false color removal. One may note also that in hardware realization median filtering does not require any floating point operation. Only comparators are necessary for the computation. This makes it an attractive proposition for VLSI implementation.

2. Color Demosaicing Algorithms

A variety of different algorithms have been advanced for color demosaicing. Brief discussions on different approaches are available in [18], and [9]. In Table 1 we present a summary of comparative performances of different techniques. In these comparisons various features are considered such as quality of reconstruction of images, quality of edge reconstruction, speed of computation and buffer requirement. The algorithms are mostly identified by acronyms as used in [18]. For evaluating quality of reconstructions, we have used measures such as Composite-Peak-Signal-to-Noise-Ratio (CPSNR) and Peak-Edge-Intensity-to-Noise-Ratio (PEINR) defined as follows.

Let $I_s(x,y), s = 1,2,3$, be the spectral components of a benchmark image of size $M \times N$ and $I_s^{'}(x,y), s = 1,2,3$, be the respective reconstructed spectral components. Then CPSNR is defined as:

Table 1: Summary of relative performances

Algorithms	Overall Image Reconstruction	Edge Reconstruction	Speed	Number of rows to be stored in the buffer
NN [20], [21]	Poor	Blurred	Fast	2
2L [17]	Good	Fair	Medium	2
BI (Bilinear)	Fair	Blurred	Fast	3
ARBH [1]	Fair	Fair	Fast	3
ECI [10], [5]	Fair	Good	Fast	3
3L [17]	Good	Good	Medium	3
EDCRAC [15]	Excellent	Good	Slow	3
LCEC [13]	Excellent	Good	Fast	5
VNGD [2]	Excellent	Good	Slow	5
AP [9]	Excellent	Excellent	Slow	5

$$CPSNR = 20\log_{10}\left(\frac{255}{\sqrt{\frac{\Sigma_{\forall s}\Sigma_{\forall x}\Sigma_{\forall y}(I_s(x,y)-I'_s(x,y))^2}{3MN}}}\right) \quad (1)$$

Since CPSNR measures do not always reflect the quality of the images in terms of edge reconstructions, we have also used another measure, PEINR (Peak-Edge-Intensity-to-Noise-Ratio) for reflecting how edges are recovered in the interpolated images. For defining PEINR we have used the binary edge map of an image, which is computed from the gradient image of the sampled array. In an edge map $e(x,y)$ of an image, if the value at a pixel location is 1 it shows the presence of edge pixels and otherwise the value is 0. Then, PEINR is defined as:

$$PEINR = 20\log_{10}\left(\frac{255}{\sqrt{\frac{\Sigma_{\forall s}\Sigma_{\forall x}\Sigma_{\forall y}e(x,y)\times(I_s(x,y)-I'_s(x,y))^2}{3\Sigma_{\forall x}\Sigma_{\forall y}e(x,y)}}}\right) \quad (2)$$

For some of the representative techniques as referred in Table 1, CPSNR and PEINR values for some typical images are shown in Table 2 and Table 3 respectively. In the tables, highest CPSNR and PEINR values (among these techniques) are highlighted in bold figures. One may note that the alternating projection technique[9] (denoted by AP) outperforms other techniques in terms of quality of reconstruction (as well as edge reconstructions) of images. However a computationally faster technique such as LCEC[13] also provides good image reconstructons. Later it will be shown that LCEC coupled with median filtering as a post processing measure improves the quality of reconstruction to a great extent and provides image qualities as good as obtained using AP[9].

3. Median Filtering for Improving the Image Quality

We presume that false color appears due to the impulsive noises (on account of estimation errors) present in the

Table 2: **CPSNR for Different Interpolation Techniques**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	27.85	31.59	32.21	34.08	**36.47**
Window	26.84	32.69	33.09	33.62	**35.78**
Pepper	25.45	25.04	27.24	26.94	**28.13**
Lighthouse	25.09	30.40	30.48	31.46	**34.62**
Sail	27.14	31.20	31.01	31.81	**32.52**

Table 3: **PEINR for Different Interpolation Techniques**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	12.91	13.57	13.70	13.79	**29.02**
Window	15.17	24.18	24.09	21.13	**29.69**
Pepper	11.51	11.77	12.10	12.13	**16.80**
Lighthouse	19.61	17.35	**20.53**	20.04	19.57
Sail	18.33	23.74	24.60	22.95	**27.95**

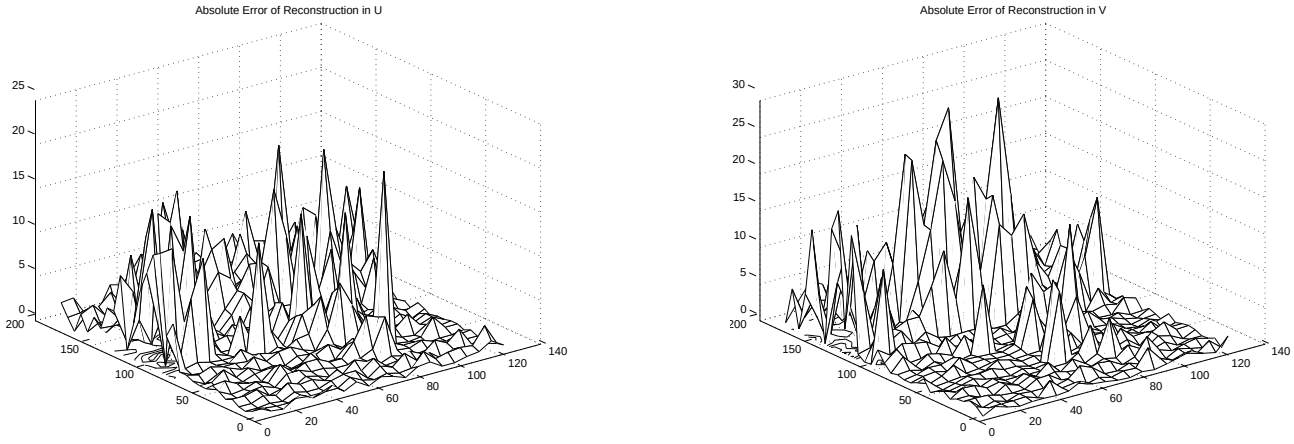
chrominance components of the interpolated images. Hence we have separated the chrominance components of the interpolated image from its luminance component and subjected them to the median filtering operations. For this we have converted the interpolated image from the RGB space to the YUV space. Here Y represents the luminance or the achromatic component. On the other hand, U and V represent the chrominance components. This helps in tackling the appearances of false colors near the achromatic edges of the color image. We have adopted a very simple strategy for suppressing the false colors. We model false colors as noisy samples of the U and V components. A typical case is presented in Figure 3 , where errors of reconstructions of U and V components are shown. Visibly they occur in isolated regions with sharp discontinuities. Hence, we modeled them as 'salt and pepper' noise. We have reduced this noise in our work using median filtering [12]. Finally, as the CFA pixel values of the color channels are also modifed in this process, we restore them back once again. The algorithm is briefly described below.

Algorithm *False_Color_Suppression_using_Median_Filtering*
Input: Demosaiced color image I in RGB color space using any conventional color interpolation algorithm and mask-size of the median filtering, say, $m \times m$.
Output: Post-processed image I_m.
Begin

1. Convert I from RGB color space to YUV color space. Let us call the components as Y, U and V, where Y represents the luminance component and U and V represent the chrominance components.

2. Apply median filtering to U using a mask of size $m \times m$. Let the filtered component be called as U_m.

3. Apply median filtering to V using a mask of size $m \times m$. Let the filtered component be called as V_m.



(a) U (b) V

Figure 3: Error of reconstruction in U and V components using LCEC technique

4. Convert Y, U_m and V_m to the image I_m in the RGB color space.

5. True pixels values of the respective spectral components in the CFA are projected back to the interpolated image.

End *False_Color_Suppression_using_Median_Filtering*

We present here in Tables 4 and 5 the respective gains in CPSNR and PEINR values in each method using median filtering with a 3×3 mask. One may observe that in many cases there are significant improvements in the CPSNR and PEINR values. For the first case, the gains in the CPSNR values range from 0.11 dB to 3.07 dB. With respect to the PEINR values there are substantial gains in many cases. The gains are as high as more than 5 dB. Another interesting aspect is that substantial gains in CPSNR values are also obtained even for a good demosaicing technique (yielding higher CPSNR values in usual circumstances) like the LCEC method (see Table 2). A typical comparitive result is presented in Figure 4. One may observe that false colored edges are significantly reduced by using median filtering. The reduction of spikes in the error plane of the chromatic components are also abserved in Figure 5. It may be noted that there is also improvement in the quality of reconstructions by AP, though the degree is smaller than others. This also indicates the good false color suppression feature of this algorithm.

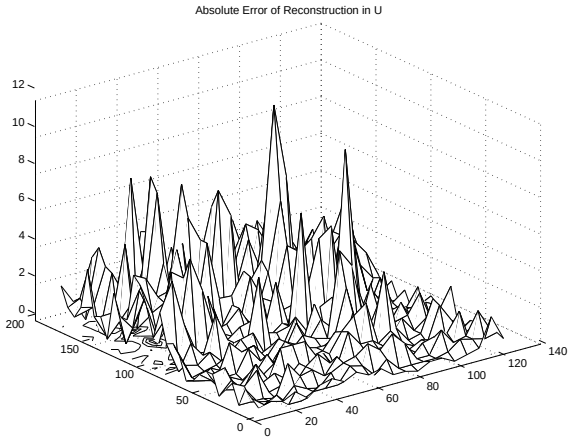
3.1. Varying the mask-sizes of the median filtering operations

We have also experimented with different mask-sizes for the median filtering operations. It has been *empirically* observed that in most cases, use of a mask of size 5×5 yields

(a) (b)

Figure 4: Reconstructed images: (a) LCEC and (b) LCEC with median filtering with a mask-size of 3x3

the maximum CPSNR. Further increases in the sizes cause reduction in the CPSNR values. In Tables 6 and 7, we present the CPSNR and PEINR values obtained after using median filtering with a 5×5 mask. In the tables absolute values of the CPSNR and PEINR values (instead of gains) are shown in order to highlight the maximum (mostly) what could be achieved by these post-processings for different conventional techniques. Interestingly in this case, the LCEC method sometimes outperforms AP. Though mask sizes higher than 5×5 yield relatively lower CPSNR values, it has been observed that in many cases false colored edges are better suppressed by them. A typical example is shown here in Figure 6. The reconstructed images (from



(a) U

(b) V

Figure 5: Error of reconstruction in U and V components using LCEC technique with median filtering

Table 4: **Gains in CPSNR for Different Interpolation Techniques after median filtering of U and V components (mask-size = 3x3)**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	2.96	2.24	3.07	2.49	1.26
Window	2.79	1.61	2.48	0.62	0.48
Pepper	1.80	2.42	1.34	1.05	0.11
Lighthouse	2.54	1.84	2.86	0.46	0.79
Sail	2.33	0.80	1.50	0.58	0.30

Table 5: **Gains in PEINR for Different Interpolation Techniques after median filtering of U and V components (mask-size = 3x3)**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	4.36	5.44	5.7	5.42	0.65
Window	2.58	0.24	2.16	1.71	-0.12
Pepper	3.13	3.66	3.72	3.74	0.48
Lighthouse	2.15	3.43	2.55	2.10	0.92
Sail	1.51	1.51	2.26	1.09	0.14

(a) 7x (b) 9x9

Figure 6: Reconstructed images (by LCEC) after median filtering by larger masks

the Lighthouse) with varying mask sizes using the LCEC method are demonstrated in Figure 6. One can observe that false colored edges are greatly reduced by the use of larger masks. One may note however, there is a risk of blurring the edges while using larger masks. It also increases the computational cost.

4. Conclusion

One of the problems in color demosaicing is that many of the interpolated images are affected by colored artifacts near the edges. The problem is more severe if the edges are achromatic. In this paper we have demonstrated the use of median filtering for suppressing this phenomenon. We have considered a set of existing interpolation algorithms and presented their performances in interpolating mosaiced patterns. Next we have carried out median filtering of the chrominance components of the demosaiced images. These post-processings have remarkably improved the quality of the reconstructions. The observations are verified by both quantitative measures for expressing the quality of reconstruction as well as by visual examinations of the reconstructed images. The post-processing operations could suppress the false colored edges to a great extent.

Table 6: **CPSNR for Different Interpolation Techniques after median filtering (mask-size = 5x5) of U and V components**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	31.89	34.49	36.37	37.16	**38.00**
Window	30.58	34.56	**36.03**	34.67	35.98
Pepper	27.57	27.75	**28.74**	28.01	28.19
Lighthouse	28.76	32.91	34.27	32.96	**35.66**
Sail	30.33	32.53	**33.09**	32.62	32.90

Table 7: **PEINR for Different Interpolation Techniques after median filtering (mask-size = 5x5)of U and V components**

Images	BI (dB)	EDCRAC (dB)	LCEC (dB)	VNGD (dB)	AP (dB)
Statue	18.70	19.36	20.52	20.55	**29.39**
Window	18.13	24.95	27.11	23.15	**29.40**
Pepper	15.66	16.04	16.44	16.58	**17.66**
Lighthouse	22.19	21.24	**23.21**	22.43	20.71
Sail	21.06	25.54	27.30	24.79	**28.08**

Acknowledgement

The first author acknowledges the Alexander Von Humboldt foundation, Germany for awarding Humboldt Fellowship, under which the work was carried out.

References

[1] J. E. Adams. Interactions between color plane interpolation and other image processing functions in electronic photography. *Proc. of SPIE*, 2416:144–151, 1995.

[2] E. Chang, S. Cheung, and D. Pan. Color filter array recovery using a threshold-based variable number of gradients. *Proc. of SPIE*, 3650:36–43, January, 1999.

[3] D. Cok. Reconstruction of CCD images using template matching. *Proc. of IS & T's Annual Conference (ICPS)*, pages 380–385, 1994.

[4] D. Cok. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal. US Patent No. 4642678, February, 1987.

[5] D. R. Cok. Signal processing method and appratus for sampled image signals. U.S. Patent No. 4630307, December, 1986.

[6] C. H. W. et. al. Reconstruction of color images from a single-chip CCD sensor based on markov random field models. *Proc. of SPIE*, 2564:282–288, 1995.

[7] H. Z. et al. A new digital signal processor for progressive scan CCD. *IEEE Trans. on Consumer Electronics*, 44(2):289–296, 1998.

[8] R. G. K. et. al. Cubic convolution interpolation for digital image processing. *IEEE Trans on Acoustic,Speech and Signal Processing*, ASSP-29:1153–1160, 1981.

[9] B. Gunturk, Y. Altunbasak, and R. Mersereau. Color plane interpolation using alternating projections. *IEEE Trans. on Image Processing*, 11(9):997–1013, 2002.

[10] R. Hibbard. Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients. US Patent No. 5382976 , January, 1995.

[11] B. Hur and M. Kang. Edge-adaptive color interpolation algorithm for progressive scan charge-coupled device image sensors. *Optical Engineering*, 40(12):2698–2708, 2001.

[12] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1989.

[13] J. H. (Jr.) and J. A. (Jr.). Adaptive color plane interpolation in single color electronic camera. US Patent No. 5629734, May, 1997.

[14] H. Kim, J. Y. Kim, S. Hwang, I. C. Park, and C. M. Kyung. Digital signal processor with effi cient RGB interpolation and histogram accumulation. *IEEE Trans. on Consumer Electronics*, 44(4):1389–1395, 1998.

[15] R. Kimmel. Demosaicing : Image reconstruction from color CCD samples. *IEEE Trans. on Image Processing*, 8(9):1221 – 1228, Sep. 1999.

[16] D. P. Mitchell and A. N. Netravali. Reconstruction filters in computer graphics. *Computer Graphics, (SIGGRAPH'88 Proc.)*, 22(4):221–228, 1988.

[17] J. Mukherjee, M. Moore, and S.K.Mitra. Color demosaicing with constrained buffering. *Proc. of ISSPA, Kuala Lumpur*, pages 52–55, Aug. 2001.

[18] J. Mukherjee, R. Parthasarathi, and S.K.Goyal. Markov random field processing for color demosaicing. *Pattern Recognition Letters*, 22:339–351, 2001.

[19] N. Ozawa. Chrominance Moire reduction using CCM signal interpolation single-chip color video camera. *ITEJ*, 46(2):210–216, 1992.

[20] K. A. Parulski. Color filters and processing alternatives for one-chip cameras. *IEEE Trans. on Electron Devices*, ED - 32(8):1381 – 1389, Aug. 1985.

[21] T. Sakamoto, C. Nakanishi, and T. Hase. Software pixel interpolation for digital still camerasuitable for a 32-bit mcu. *IEEE Trans. on Consumer Electronics*, 44(4):1342 – 1352, Nov. 1998.

[22] H. Sugiara, K. Asakawa, and J. Fujino. False color signal reduction method for single-chip color video cameras. *IEEE Trans. on Consumer Electronics*, 40(2):100–106, 1994.

[23] X. L. Wu and et.al. Color retoration from digital camera data by pattern matching. *Proc. of SPIE*, 3018:12–17, 1997.

[24] H. Zen, T. Koizumi, H. Yamamoto, and I. Kimura. A new digital signal processor for progressive scan CCD. *IEEE Trans. on Consumer Electronics*, 44(2):289 – 295, May 1998.