An Improved Color Image Demosaicking Algorithm

Shousheng Luo
School of Mathematical Sciences, Peking University, Beijing 100871, China
Haomin Zhou
School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332, USA
Tie Zhou
School of Mathematical Sciences, Peking University, Beijing 100871, China

***Abstract*—Color image demosaicking is an important issue in color image acquisition procedures. Most of the existing demosaicking methods reduce the artifacts at the price of the high computational cost posterior processing. In this paper, we propose a more effective demosaicking algorithm. The proposed method improves the demosaicked image's qualities only by interpolation, rather than posterior processing. In order to avoid the propagation of errors occurred during the green channel reconstruction, we interpolate the red and blue channels by the original samples to the greatest extent. The experimental results show that the proposed algorithm yields better demosaicking images as compared with a number of typical demosaicking algorithms. Furthermore, the computational cost of the proposed method is lower.**

I. Introduction

In order to reduce cost and size, most digital cameras use a single sensor (CCD or CMOS) whose surface is covered by a color filter array (CFA), such as Bayer pattern CFA [1] shown in Figure 1, so that each sensor samples one of the three primary color values. To render a full-color image, the missing color values have to be estimated from neighboring available ones. This process is usually called color image demosaicking (or demosaicing), because the original samples are usually arranged in a mosaic pattern. The goal of demosaicking is to reconstruct an image so that it can be as close to the original full-resolution color image as possible. Meanwhile the computational cost needs to be kept low for the reason of being cost effective in practical applications.

Since all color channels have very similar characteristics in natural images, such as texture and edge location, there is a strong local inter-channel correlation [3] [6] [10]. This inter-channel correlation is generally exploited by assumption that the differences between two values in two color channels are likely to be constant within a local image region [10] [11] [12]. And a number of demosaicking methods based on this assumption were proposed. A comprehensive survey of demosaicking method is available in [9]. Most of the algorithms usually consist of three steps. Step 1, green channel is interpolated by utilizing the neighboring samples. Step 2, the red and blue channels are interpolated by the benefit of the estimated green channel and the inter-channel correlation. Obviously, the accuracy of step 1 is critical to the overall performance because of notorious error propagation. Step 3, a postprocessing is followed to suppress the artifacts caused by interpolation error.

In view of the importance of green channel estimate, significant efforts have been devoted to improve the accuracy of green channel interpolation [2]- [8], [10]- [20]. The common strategy for green channel estimate is that two or more values are computed by first or second order interpolation, then the weighted average or one of them is taken as the estimate based on various criteria [10] [15] [16] [18].

Traditionally, based on the interpolated full-resolution green channel, the red and blue channels are estimated [10] [13] [18]. Specifically, color-differences of between red and blue channels to the green one, instead of the red and blue channels themselves, are interpolated based on the full-resolution green channel and the down-sampled red and blue channels; then the green channel is simply added back to the interpolated color differences for recovering red and blue channels. Despite the popularity of this method, it has a fundamental weakness of error propagation - any errors rendered during the interpolation of the green channel would inevitably propagate to the red and blue channels.

In order to suppress the annoying artifacts caused by interpolation error, many researchers resort to postprocessing. Most of the postprocessing methods smooth color differences between two channels of the demosaicked image to enhance the inter-channel correlation [5] [6] [11] [20]. Recently, regularization based on non-local technique was applied to demosaicked image spatially [14] [15]. Researchers have pointed out that the previous kind of method fails in images with low inter-channel correlation and high saturation, even results in annoying artifacts [9] [15]. Although the non-local regularization method could partially overcome the faults of the previous method, it substantially increases the computational cost, which can hardly be satisfied by the consumer portable cameras. We have to find the best tradeoff between image quality and computational cost.

In view of the above discussion, we aim to improve the interpolation accuracy and suppress the artifacts by modifying the interpolation steps, rather than the postprocessing.In terms of green channel interpolation, we compute the six values of color difference of between green channel and red(blue) channel by inter-channel and intra-channel correlation. Then we take the weighted average of them as the estimate of the color difference, and the estimate of green value is obtained by adding the red(blue) pixel value. In order to avoid the error propagation of the estimated green channel, we interpolate the

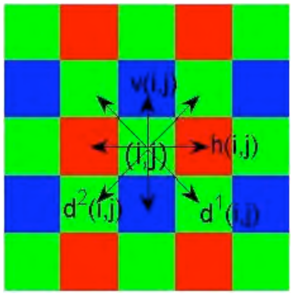

Fig. 1. Bayer pattern.

red and blue channels by the original samples as much as we can. Specifically, we first estimate red and blue values at green pixels by the weighted average of several directional interpolations which are computed by neighboring original samples. Then, the red(blue) values at blue(red) pixels are estimated by the available values including the estimated and original values.

This paper is organized as follows : Section II is devoted to the proposed demosaicking method in details. Some experimental results are demonstrated in Section III to present the effectiveness of the proposed method. Conclusions are stated in Section IV.

II. The Proposed Demosaicking Algorithm

For the sake of reference, we use (i, j) to denote the spatial coordinate, and use C and $\hat{C}$ to denote the original and reconstructed values. We assume that the original samples are R at $(i = odd, j = even)$, G at $(i = odd, j = odd)$ and $(i = even, j = even)$, and B at $(i = even, j = odd)$, where R, G and B are the abbreviations of $red, green$ and blue. Moreover, we use $v_{(i,j)} = |C_{(i+1,j)} - C_{(i-1,j)}|, h_{(i,j)} = |C_{(i,j+1)} - C_{(i,j-1)}|, d^1_{(i,j)} = |C_{(i-1,j-1)} - C_{(i+1,j+1)}|, d^2_{(i,j)} = |C_{(i-1,j+1)} - C_{(i+1,j-1)}|$ to measure the local gradient along vertical, horizontal and two oblique directions at location (i, j)(see Figure 1).

A. Interpolation of green Channel

Instead of estimating the missing green value directly, we first compute color difference between the green channel and the red(or blue) one at pixel (i, j) along four directions: up(u), down(d), left(l) and right(r) by using inter-channel correlation.

$$\begin{aligned} GC^u &= G_{(i-1,j)} - \frac{1}{2}(C_{(i-2,j)} + C_{(i,j)}) \\ GC^d &= G_{(i+1,j)} - \frac{1}{2}(C_{(i+2,j)} + C_{(i,j)}) \\ GC^l &= G_{(i,j-1)} - \frac{1}{2}(C_{(i,j-2)} + C_{(i,j)}) \\ GC^r &= G_{(i,j+1)} - \frac{1}{2}(C_{(i,j+2)} + C_{(i,j)}) \end{aligned}$$

In [15], Zhang et al took the weighted sum of the above four values as the color difference estimate of $GC_{(i,j)} = G_{(i,j)} - C_{(i,j)} (C = R$ or $B)$. Here, we propose a new strategy to estimate $GC_{(i,j)}$. Besides the above four values, we compute another two values along horizontal(h) and vertical(v) directions which make use of intra-channel correlation.

$$\begin{aligned} GC^h &= \frac{1}{2}(G_{(i,j+1)} + G_{(i,j-1)}) - C_{(i,j)} \\ GC^v &= \frac{1}{2}(G_{(i+1,j)} + G_{(i-1,j)}) - C_{(i,j)} \end{aligned}$$

The value of $GC_{(i,j)}$ is computed by the weighted average of the above six values

$$\begin{aligned} \hat{GC}_{(i,j)} &= \frac{1}{S}(w_u GC^u + w_d GC^d + w_l GC^l \\ &\quad + w_r GC^r + w_h GC^h + w_v GC^v) \end{aligned} \quad (1)$$

where $S = w_u + w_d + w_l + w_r + w_h + w_v$. Then the value of $\hat{G}_{(i,j)}$ can be obtained by $\hat{G}_{(i,j)} = \hat{GC}_{(i,j)} + C_{(i,j)}$. Obviously, it is critical to compute the weights of (1) properly. Intuitively, these weights should relate to the local gradient(smoothness). There are a number of formulae to compute the weights [15] [18]. Here, we use the following formulae.

$$\begin{aligned} w_h &= \frac{1}{\rho + H_{(i,j)} + \frac{1}{2}(h_{(i,j-1)} + h_{(i,j+1)})} \\ w_v &= \frac{1}{\rho + H(i,j) + \frac{1}{2}(v_{(i-1,j)} + v_{(i+1,j)})} \\ w_u &= \frac{1}{\rho + |GC^u - GC^v| + V_{(i-1,j)}} \\ w_d &= \frac{1}{\rho + |GC^d - GC^v| + V_{(i+1,j)}} \\ w_l &= \frac{1}{\rho + |GC^l - GC^h| + H_{(i,j-1)}} \\ w_r &= \frac{1}{\rho + |GC^r - GC^h| + V_{(i,j+1)}} \end{aligned}$$

where $V_{(i,j)} = v_{(i,j)} + \frac{1}{2}(v_{(i,j-1)} + v_{(i,j+1)}), H_{(i,j)} = h_{(i,j)} + \frac{1}{2}(h_{(i-1,j)} + h_{(i+1,j)})$, $\rho > 0$ is a user-specified small positive number to avoid the denominator being zero. In this paper, we let $\rho = 0.1$.

B. Interpolation of red / blue Channel

The interpolation of red and blue channels includes two substeps: 1) filling the missing red and blue values at green pixels, 2) interpolating the missing red values at blue pixels and vice versa. In order to reduce error propagation, we interpolate the red and blue channels by using original samples as far as possible. Since the samples of red and blue channels are symmetric, in the following we only discuss the interpolation of red channel. We first focus on the estimate of the red missing values at green pixels. Here, we only give the concrete interpolation formulae for the red values at $(i = odd, j = odd)$, and the interpolation formulae at $(i = even, j = even)$ can be obtained similarly by a 90° rotation. We first compute seven color differences of red channel to the green one at (i, j)

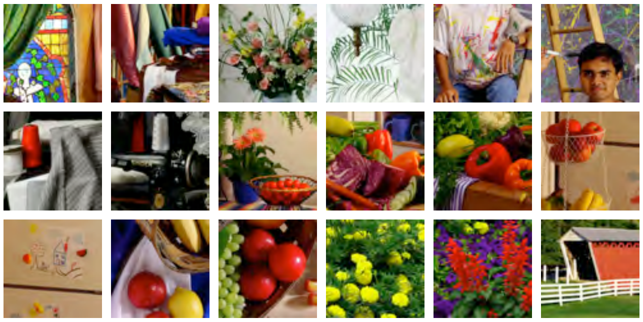

Fig. 2. Test images (image 1 to image 18, enumerated from left to right and top to bottom).

$$
\begin{aligned}
RG^{lu} &= \frac{1}{2}(R_{(i-2,j-1)} + R_{(i,j-1)}) - G_{(i-1,j-1)} \\
RG^{ld} &= \frac{1}{2}(R_{(i+2,j-1)} + R_{(i,j-1)}) - G_{(i+1,j-1)} \\
RG^{rd} &= \frac{1}{2}(R_{(i+2,j+1)} + R_{(i,j+1)}) - G_{(i+1,j+1)} \\
RG^{ru} &= \frac{1}{2}(R_{(i-2,j+1)} + R_{(i,j+1)}) - G_{(i-1,j+1)} \\
RG^{hr} &= R_{(i,j+1)} - \frac{1}{2}(G_{(i,j)} + G_{(i,j+2)}) \\
RG^{hl} &= R_{(i,j-1)} - \frac{1}{2}(G_{(i,j)} + G_{(i,j-2)}) \\
RG^{hc} &= R_{(i,j)} - \frac{1}{2}(G_{(i,j-1)} + G_{(i,j+1)})
\end{aligned}
$$

The estimate of $RG_{(i,j)} = R_{(i,j)} - G_{(i,j)}$ is computed by the weighted sum of the above seven values

$$
\begin{aligned}
\hat{RG}_{(i,j)} = \frac{1}{S}(&w_{lu}RG^{lu} + w_{ld}RG^{ld} + w_{rd}RG^{rd} + \\
&w_{ru}RG^{ru} + w_{hr}RG^{hr} + w_{hl}RG^{hl} + w_{hc}RG^{hc}) \qquad (2)
\end{aligned}
$$

Where $S = w_{lu} + w_{ld} + w_{rd} + w_{ru} + w_{hl} + w_{hr} + w_{vl} + w_{vr} + w_{hc}$. We can obtain $R_{(i,j)}$ by $\hat{R}_{(i,j)} = \hat{RG}_{(i,j)} + G_{(i,j)}$. The weights of (2) are computed by the following formulae

$$
\begin{aligned}
w_{lu} &= \frac{1}{\rho + V(i-1, i-1) + |G_{(i,j)} - G_{(i-1,j-1)}|} \\
w_{ld} &= \frac{1}{\rho + V(i+1, i-1) + |G_{(i,j)} - G_{(i+1,j-1)}|} \\
w_{ru} &= \frac{1}{\rho + V(i-1, j+1) + |G_{(i,j)} - G_{(i-1,j+1)}|} \\
w_{rd} &= \frac{1}{\rho + V(i+1, j+1) + |G_{(i,j)} - G_{(i+1,j+1)}|} \\
w_{hr} &= \frac{1}{\rho + H(i, j+1) + |R_{(i,j+1)} - R_{(i,j-1)}|} \\
w_{hl} &= \frac{1}{\rho + H(i, j-1) + |R_{(i,j+1)} - R_{(i,j-1)}|} \\
w_{hc} &= \frac{1}{\rho + H(i,j) + \frac{1}{2}(h_{(i,j+1)} + h_{(i,j-1)})}
\end{aligned}
$$

Next we compute the red value at blue pixel ($i = even, j = odd$) by available values. Similarly, we compute the color difference $\hat{RG}_{(i,j)}$ between the red channel and the green one at blue pixel by

$$
\begin{aligned}
\hat{RG}_{(i,j)} = \frac{1}{S}(&w_{lu}\hat{RG}^{lu} + w_u\hat{RG}^{u} + w_{ru}\hat{RG}^{ru} + w_l\hat{RG}^{l} \\
&+ w_r\hat{RG}^{r} + w_{ld}\hat{RG}^{ld} + w_d\hat{RG}^{d} + w_{rd}\hat{RG}^{rd}) \quad (3)
\end{aligned}
$$

where $S = w_{lu} + w_u + w_u + w_{ru} + w_{ru} + w_l + w_r + w_{ld} + w_d + w_{rd}$, $RG^{lu} = R_{(i-1,j-1)} - \hat{G}_{(i-1,j-1)}, RG^u = \hat{R}_{(i-1,j)} - G_{(i-1,j)}, RG^{ru} = R_{(i-1,j+1)} - \hat{G}_{(i-1,j+1)}, RG^l = \hat{R}_{(i,j-1)} - G_{(i,j-1)}, RG^r = \hat{R}_{(i,j+1)} - G_{(i,j+1)}, RG^{ld} = R_{(i+1,j-1)} - \hat{G}_{(i+1,j-1)}, RG^d = \hat{R}_{(i+1,j)} - G_{(i+1,j)}, RG^{rd} = R_{(i+1,j+1)} - \hat{G}_{(i+1,j+1)}$, and the weights of (3) are computed by the following formulae.

$$
\begin{aligned}
w_{lu} &= \frac{1}{\rho + |G_{(i,j)} - G_{(i-1,j-1)}| + d^1_{(i-1,j-1)} + d^1_{(i,j)}} \\
w_u &= \frac{1}{\rho + |G_{(i,j)} - G_{(i-1,j)}| + v_{(i,j)} + v_{(i-1,j)}} \\
w_{ru} &= \frac{1}{\rho + |G_{(i,j)} - G_{(i-1,j+1)}| + d^2_{(i-1,j+1)} + d^2_{(i,j)}} \\
w_l &= \frac{1}{\rho + |G_{(i,j)} - G_{(i,j-1)}| + h_{(i,j)} + h_{(i,j-1)}} \\
w_r &= \frac{1}{\rho + |G_{(i,j)} - G_{(i,j+1)}| + h_{(i,j)} + h_{(i,j+1)}} \\
w_{ld} &= \frac{1}{\rho + |G_{(i,j)} - G_{(i+1,j-1)}| + d^2_{(i+1,j-1)} + d^2_{(i,j)}} \\
w_d &= \frac{1}{\rho + |G_{(i,j)} - G_{(i+1,j)}| + v_{(i,j)} + v_{(i+1,j)}} \\
w_{rd} &= \frac{1}{\rho + |G_{(i,j)} - G_{(i+1,j+1)}| + d^1_{(i+1,j+1)} + d^1_{(i,j)}}
\end{aligned}
$$

TABLE I
CPSNR PERFORMANCE COMPARISON OF VARIOUS DEMOSAICKING METHODS.

Img	[10]	[11]	[12]	[13]	[14]	[15]	Proposed
1	28.17	24.96	27.03	26.93	27.54	28.91	**29.12**
2	33.92	32.12	33.70	33.63	34.12	**35.00**	34.95
3	30.51	31.35	32.65	32.20	32.28	32.45	**32.64**
4	32.92	33.57	34.40	34.14	34.22	35.76	**36.74**
5	32.81	29.09	31.35	31.27	32.02	33.86	**34.04**
6	36.41	31.01	33.80	33.75	35.08	**37.97**	37.79
7	33.93	37.55	**38.67**	37.25	36.90	35.83	35.47
8	35.95	36.84	37.52	**37.68**	37.17	37.43	37.46
9	35.92	32.62	34.38	34.40	35.63	37.11	**37.56**
10	37.49	35.18	36.38	36.44	37.32	**38.63**	38.58
11	38.45	36.21	37.24	37.00	38.16	39.58	**39.61**
12	37.30	36.04	36.63	36.53	37.81	**38.85**	38.39
13	39.70	37.95	38.80	38.56	39.80	40.75	**40.96**
14	37.70	36.53	37.27	37.10	37.74	38.63	**38.79**
15	38.19	36.56	37.26	37.03	37.90	38.95	**39.22**
16	32.71	28.42	30.44	30.11	31.17	33.64	**33.97**
17	31.96	26.83	29.35	29.41	30.54	32.74	**33.02**
18	33.87	32.51	33.86	33.63	34.13	35.20	**35.74**
Avg	34.88	33.07	34.49	34.28	34.97	36.18	**36.34**

III. Experimental Results

We evaluate the performance of our algorithm by comparing it with six representative algorithms in the literature [10]- [15].

In our experiments, we use the McMaster dataset established by Zhang [21] as tested images. In image demosaicking field, the Kodak color image dataset was widely used as a standard dataset in the past. Because the inter-channel correlation of the Kodak images is very high, which is very different from other natural images [9], researchers started to use the McMaster dataset as the tested images [14] [15].

Without the postprocessing, the computational cost of our proposed method is lower than most of the methods in the literature. Let M and N denote the height and width of the image. The computational cost of the proposed demosaicking method is about $60MN$ operation. The computational cost of the self-similarity driven (SSD) method proposed by Buades et al. [14] is about $2109MN$, and the computational cost of the non-local adaptive thresholding(NAT) method proposed by Zhang et al. in [15] is much higher than the SSD method.

Color peak signal-noise ratio(CPSNR) evaluation(see Table I) is usually used to measure the deviation between two color images in color image processing. Here, we used CPSNR to measure the demosaicking quality and assess various demosaicking methods. CPSNR is defined as

$$CPSNR \quad = \quad 10log_{10}(\frac{255^2}{CMSE}) \tag{4}$$

where $CMSE = \frac{1}{3MN}\sum\limits_{p=r,g,b}\sum\limits_{i,j}(\hat{u}(i,j,p) - u(i,j,p))^2$, and $\hat{u}$ is the demosaicked image and u the original.

Table I tabulates the CPSNR performance of different demosaicing algorithms(the highest CPSNR value in each row is highlighted in bold). It can be observed that our algorithm achieves the best CPSNR performance in most of the images and the best mean performance among the tested algorithms.

To assess the visual quality of different demosaicking algorithms, in Figures 3 and 4 we present two challenging regions for demosaicking cropped from image 1 and image 16 respectively, and the results obtained by several typical demosaikcing methods. It is clear that the proposed method and the NAT method yield much better results than the other methods. And in terms of the tested images, we can hardly find distinct visual difference between the NAT method and the proposed method. However, the computational cost of our method is much lower than the NAT method. Generally speaking, the proposed method is much better than the other methods.

IV. Conclusion

In this paper, we presented a powerful and efficient demosaicking method. We efficiently balanced the inter-channel correlation and the intra-channel correlation in our demosaicking algorithm. The proposed algorithm is able to suppress the error propagation in demosaicking procedure and preserve much more image edges. Compared with a number of typical methods in the literature, the proposed algorithm is able to produce better demosaicking results, and reduce the artifacts. Furthermore, the computational cost of the proposed method is much lower.

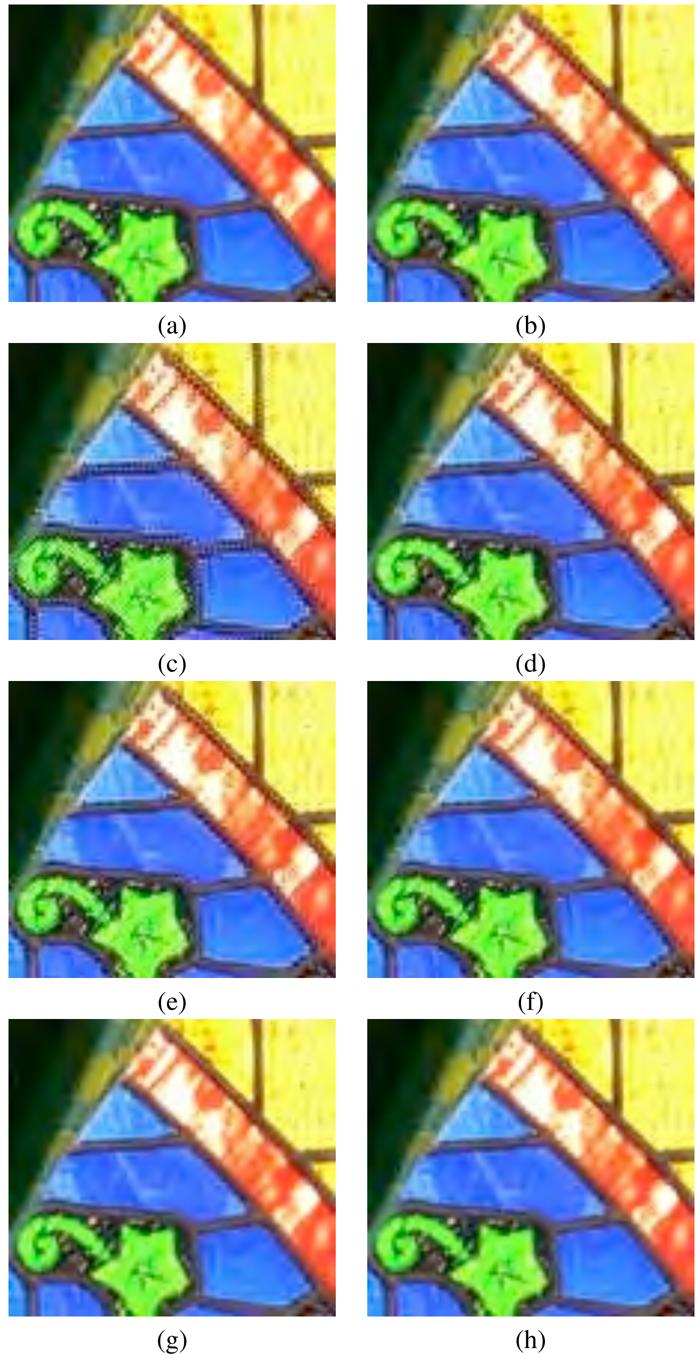

Fig. 3. Visual comparison: (a) original image 1 and demosaicked images (b) by [10];(c) by [11]; (d) by [12]; (e) by [13]; (f) [14]; (g) by [15]; and (h) by the proposed algorithm.

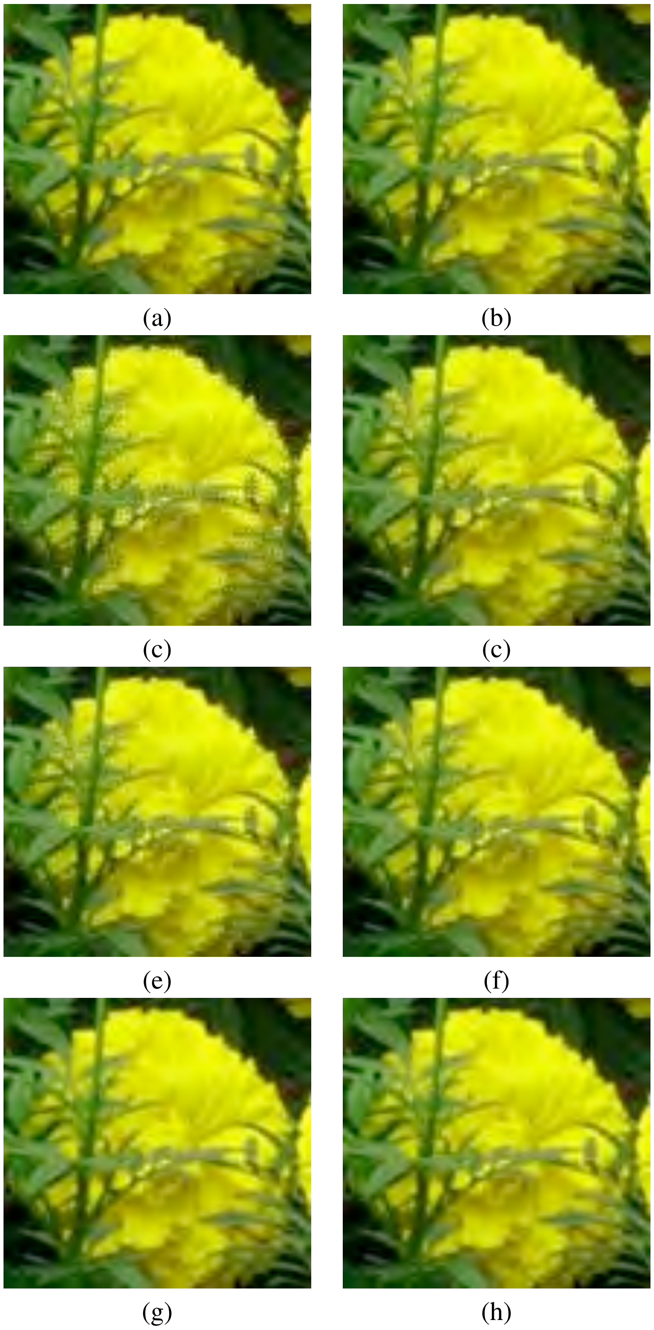

Fig. 4. Visual comparison: (a) original image 16 and demosaicked images (b) by [10];(c) by [11]; (d) by [12]; (e) by [13]; (f) [14]; (g) by [15]; and (h) by the proposed algorithm.

V. Acknowledgment

This work is supported by the National Basic Research Program of China (2011CB809105) and NSF grants of China (61121002, 10990013).

References

[1] B. E. Bayer, *Color Imaging Array*, U. S. Patent 3971065, 1976.

[2] L. Chang Y. Tan, *Hybrid color filter array demosaicking for effective artifact suppression*, J. Electron. Imaging vol.15, no.1, pp.1-17(013003), 2006.

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

[4] R.H. Hibbard, *Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients*, U. S. Patent 5382976, 1995.

[5] W. T. Freeman, *Median filter for reconstructing missing color samples*, U. S. Patent 4724395, 1988.

[6] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, *Color plane interpolation using alternating projections*, IEEE Trans. Image Process. vol.11, no.9, 997-1013, 2002.

[7] C. Pei and I. K. Tam, *Effective color interpolation in CCD color filter array using signal correlation*, IEEE Trans. Circuits Syst. Video Technol. vol.13, no.6, 503-513, 2003.

[8] K. Hirakawa and T. Parks, *Adaptive homogeneity-directed demosaicing algorithm*, IEEE Trans. Image Process. vol. 14, no. 3, pp. 360-369, 2005.

[9] X. Li, B. K. Gunturk, and L. Zhang, *Image demosaicing: a systematic survey*, Proceed. SPIE, vol. 6822, pp.68221(1-13), 2008.

[10] J. Hamilton and J. Adams, *Adaptive color plane interpolation in single sensor color electronic camera*, U. S. Patent 5629734, 1997.

[11] X. Li, *Demosaicing by successive approximation*, IEEE Trans. Image Process., vol. 14, no. 3, pp.370-379, 2005.

[12] L. Zhang and X. Wu, *Color demosaicking via directional linear minimum mean square-error estimation*, IEEE Trans. Image Process., vol 14, no. 12, pp.2167-2178, 2005.

[13] D. Menon, S. Andriani, and G. Calvagno, *Demosaicing With Directional Filtering and a posteriori Decision*, IEEE Trans. Image Process., vol. 16, no. 1, pp.132C141, 2007.

[14] A. Buades, B. Coll, J. M. Morel and C. Sbert, *Self-similarity driven color demosaicking*, IEEE Trans. Image Process. vol. 18, no. 7, pp.1192-1202, 2009.

[15] L. Zhang , X. L. Wu, A. Buades, and X. Li, *Color demosaicking by local directional interpolation and nonlocal adaptive thresholding*,J. Electron. Imaging 20(2), pp.1-15(023016), AprCJun 2011

[16] J. S. J. Li, and S. Randhawa, *Color Filter Array Demosaicking Using High-Order Interpolation Techniques With a Weighted Median Filter for Sharp Color Edge Preservation*, IEEE Trans. Iamge Process., vol. 18, no. 9, pp.1946-1957, 2009

[17] D. Paliy, V. Katkovnik, R. Bilcu, S. Alenius, and K. Egiazarian, *Spatially adaptive color filter array interpolation for noiseless and noisy data*, Int. J. Imaging Systems and Technology, vol. 17, no. 3, pp.105-122, 2007. vol. 19, no. 2, pp.105-121, 1994.

[18] W. Lu and Y. P. Tan, *Color filter array demosaicking: new method and performance measures*, IEEE Trans. Image Process., vol. 12, no. 10, pp. 1194-1210, 2003.

[19] X. Wu and N. Zhang, *Primary-consistent soft-decision color demosaicking for digital cameras* (patent pending), IEEE Trans. Image Process., vol. 13, no. 9, pp.1263-1274, 2004. *Color Demosaicking by Local Directional Interpolation and Non-local Adaptive Thresholding*, J. Electron. Imaging, vol.20, no.2,pp.1-16(023016) 2011.

[20] T. Saito and T. Komatsu, *Demosaicing approach based on extended color total-variation regularization*, in Proc. IEEE Int. Conf. Image Process., vol.1, pp.885-888, 2008.

[21] http://www4.comp.polyu.edu.hk/ cslzhang/CDM_Dataset.htm