RESIDUAL INTERPOLATION FOR COLOR IMAGE DEMOSAICKING

Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi

Tokyo Institute of Technology

ABSTRACT

A color difference interpolation technique is widely used for color image demosaicking. In this paper, we propose *residual* interpolation as an alternative to the color difference interpolation, where the residual is a difference between an observed and a tentatively estimated pixel value. We incorporate the proposed residual interpolation into the gradient based threshold free (GBTF) algorithm, which is one of current state-of-the-art demosaicking algorithms. Experimental results demonstrate that our proposed demosaicking algorithm using the residual interpolation can give state-of-the-art performance for the 30 images of Kodak and IMAX datasets.

Index Terms— Interpolation, demosaicking, color filter array (CFA), residual.

1. INTRODUCTION

A single sensor technology with a color filter array (CFA) is widely used in the digital camera industry [1]. In a single sensor camera with the CFA, only one pixel value among RGB values is recorded at each pixel and other pixel values are interpolated by a in-camera process called demosaicking [2, 3]. The development of a high-performance demosaicking algorithm plays a crucial role to acquire high-quality color images.

The most popular and widely used CFA is the Bayer CFA (Fig. 1) [4]. Researches on the demosaicking algorithm for the Bayer CFA have a long history [2, 3]. Most of Bayer demosaicking algorithms first interpolate G pixel values since G pixels have the double sampling density compared with R and B pixels. Then, color differences (R-G or B-G) are calculated at the R and B pixels and color difference interpolation is performed. Finally, the interpolated G image is added to the interpolated color difference images to acquire R and B images. The reason why the color differences are used is based on the observation that all color bands have very similar image structures such as textures and edges [5, 6]. From this observation, the color difference images can be assumed approximately flat within the boundary of a captured object, which makes the interpolation process easy.

In this paper, we propose a novel demosaicking algorithm using *residual* interpolation. Instead of using color differences, we generate tentative estimates of R and B images ($\check{R}$ and $\check{B}$) and calculate residuals which are the differences between the observed and the tentatively estimated R and B pixel values ($R - \check{R}$ and $B - \check{B}$). The motivation we use the residuals instead of the color differences is that if we can accurately generate the tentative estimates, the residuals should be flatter than the standard color differences, which leads to better results for color image demosaicking. We generate such accurate tentative estimates by using the guided filter [7], which is a recently-proposed powerful edge-preserving filter. After calculating the residuals, we interpolate the residuals instead of the color differences. We incorporate the residual interpolation into the gradient based threshold free (GBTF) algorithm [8], which is one of state-of-the-art Bayer demosaicking algorithms using the color difference interpolation. Experimental results demonstrate that our proposed demosaicking algorithm using the residual interpolation can give state-of-the-art performance for the 30 images of Kodak and IMAX datasets.

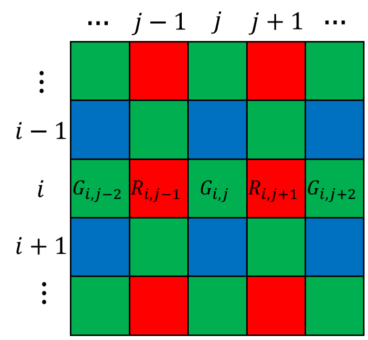


Fig. 1. Bayer CFA.

2. PROPOSED RESIDUAL INTERPOLATION

We first describe the basic processing pipeline of the proposed residual interpolation. Taking the interpolation of R pixel values as an example, we compare the proposed residual interpolation with the standard color difference interpolation in Fig. 2.

Fig. 2 (a) shows the interpolation process of the R pixel values by using the standard color difference interpolation. First, the G image is generated by an interpolation algorithm. Then, the color differences (R-G) are calculated at the R pix-

978-1-4799-2341-0/13/$31.00 ©2013 IEEE

els and the color difference interpolation is performed. Finally, the G image is added to the interpolated color difference image to acquire the interpolated R image.

Fig. 2 (b) shows the interpolation process of the R pixel values by using the proposed residual interpolation. First, the G image is generated by an interpolation algorithm, which is the same as the color difference interpolation. Then, we generate the tentative estimate of the R image ($\check{\mathrm{R}}$) and calculate the residuals between the observed and the tentatively estimated R pixel values ($\mathrm{R} - \check{\mathrm{R}}$) at the R pixels. After that, we interpolate the residuals instead of the color differences. We can expect that the residuals become flatter than the color differences by accurately generating the tentative estimate of the R image, which makes the interpolation process easy. Finally, the tentative estimate of the R image is added to the interpolated residual image to acquire the interpolated R image.

In our proposed demosaicking algorithm, we generate the tentative estimate by upsampling the observed R pixel values by using the guided filter [7]. The guided filter can accurately upsample input sparse data by using the guide image, which is used as a reference to exploit image structures, in the similar manner as the joint bilateral upsampling [9]. For each local patch, the output of the guided filter is represented by a linear transform of the guide image, therefore, the image structures of the guide image are preserved. We use the interpolated G image as the guide image in the same manner as in [10]. In this paper, we call this upsampling process the guided upsampling.

3. PROPOSED DEMOSAICKING ALGORITHM

We develop our proposed demosaicking algorithm by incorporating the proposed residual interpolation into the GBTF algorithm [8]. The GBTF algorithm first interpolates the G pixel values. Then, the R and B pixel values are interpolated by using the color difference interpolation. For the interpolation of the R and B pixel values, we simply replace the color difference interpolation with the proposed residual interpolation as described in the previous section. We use bilinear interpolation for the residual interpolation. In the following, we propose the interpolation process of the G pixel values based on the GBTF algorithm.

The interpolation process of the G pixel values by the GBTF algorithm consists of three steps: (i) The Hamilton and Adams' interpolation formula [5] is applied in the horizontal and vertical directions to estimate the G pixel values at the R and B pixels and the R or B pixel values at the G pixels. As a result, the horizontally and vertically estimated R, G, and B pixel values are generated. (ii) The horizontal and vertical color differences (G-R or G-B) are calculated at each pixel. Then, the horizontal and vertical color differences are smoothed and combined into the final color difference estimate. (iii) The G pixel values at the R and B pixels are interpolated by adding the observed R or B pixel values to the final color difference estimates.

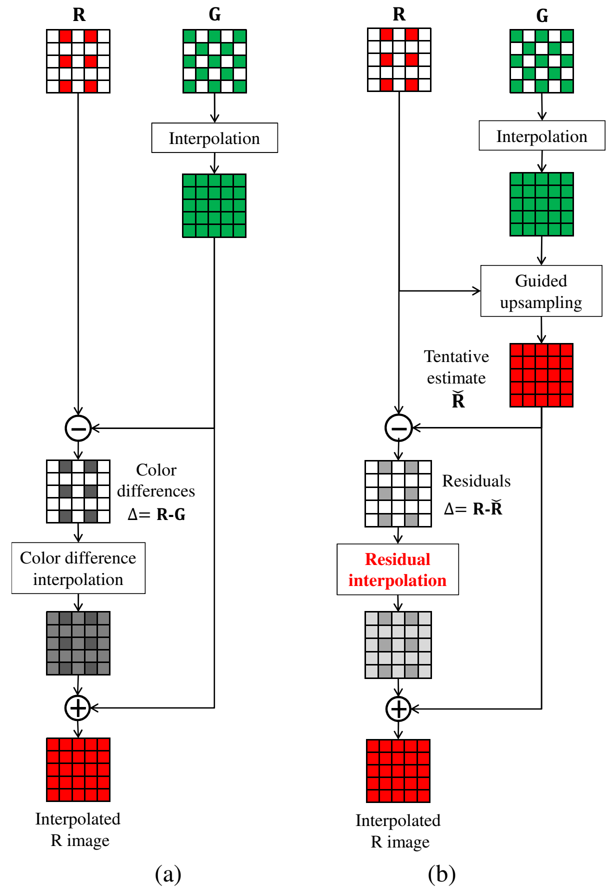


Fig. 2. The interpolation of R pixel values (a) by using the standard color difference interpolation, and (b) by using our proposed residual interpolation.

The Hamilton and Adams' interpolation formula in the step (i) can be interpreted as a linear color difference interpolation. We replace the linear color difference interpolation with the proposed residual interpolation. To simplify the explanation, in the following, we focus on the estimation of the R pixel values at the G pixels in the horizontal direction. The B pixel values at the G pixels are estimated in the same manner as the R pixel values. And also, we apply the same process in the vertical direction.

The Hamilton and Adams' interpolation formula for the R pixel value in the horizontal direction can be expressed as:

$$\hat{R}^H_{i,j} = (R_{i,j-1} + R_{i,j+1})/2 + (2\times G_{i,j} - G_{i,j-2} - G_{i,j+2})/4, \quad (1)$$

where the suffix i, j represents the target pixel, $\hat{R}^H_{i,j}$ is the horizontally estimated R pixel value at the G pixel. This interpolation formula can be interpreted as the linear color difference interpolation as:

$$\hat{R}^H_{i,j} - G_{i,j} = (R_{i,j-1} - \tilde{G}^H_{i,j-1})/2 + (R_{i,j+1} - \tilde{G}^H_{i,j+1})/2, \quad (2)$$

where $\tilde{G}^H$ is the horizontally estimated G pixel value at the R pixel calculated as:

$$\tilde{G}^H_{i,j-1} = (G_{i,j-2} + G_{i,j})/2, \quad \tilde{G}^H_{i,j+1} = (G_{i,j} + G_{i,j+2})/2. \tag{3}$$

In the proposed algorithm, we use the tentative estimates instead of these estimated G pixel values. In the step (i) of the GBTF, the interpolation process is performed in one dimensional image at every row and column. First, we apply the linear interpolation to the G pixel values to generate the one dimensional guide image. Then, the guided upsampling is applied to the R pixel values to obtain the tentative estimate of the one dimensional R image. After that, the residuals are calculated and the residual interpolation is performed. Finally, we estimate the R pixel values at the G pixels by adding the tentative estimate to the interpolated one dimensional residual image.

The resultant R pixel value can be expressed as:

$$\hat{R}^H_{i,j} = (R_{i,j-1} - \check{R}^H_{i,j-1})/2 + (R_{i,j+1} - \check{R}^H_{i,j+1})/2 + \check{R}^H_{i,j}. \tag{4}$$

where $\check{R}^H$ is the tentative estimate of the R pixel value. The G pixel values at the R pixels are estimated in the same manner, where the tentative estimate of the one dimensional G image is generated.

After the above step (i), the step (ii) and (iii) of the GBTF algorithm are applied. Although the proposed residual interpolation also can be incorporated into the step (ii), we will discuss it in future work since the incorporation of the residual interpolation into the step (ii) is not straightforward.

In the GBTF algorithm, a simple averaging filter, $f = [11111]/5$, is used for smoothing the directional color differences in the step (ii). In contrast, we introduce a Gaussian weighted averaging filter instead of the simple averaging filter to improve the performance. We empirically use 1 for the standard deviation of the Gaussian weight.

4. EXPERIMENTS

The proposed algorithm [1] is evaluated with two standard color image datasets, the IMAX dataset and the Kodak dataset used in [3]. The IMAX dataset consists of 18 images and the image size is 500×500. The IMAX images are cropped from original 2310×1814 high-resolution images. The Kodak dataset consists of 12 images and the image size is 768×512. We compare the proposed algorithm with five state-of-the-art algorithms, adaptive homogeneity-directed (AHD) [11], directional linear minimum mean square-error estimation (DLMMSE) [12], gradient based threshold free (GBTF) [8], local polynomial approximation (LPA) [13], and local directional interpolation and non-local adaptive thresholding (LDI-NAT) [14] algorithms.

[1]The source code of our proposed demosaicking algorithm can be downloaded from http://www.ok.ctrl.titech.ac.jp/res/DM/RI.html.

Table 1. The average PSNRs and CPSNRs of IMAX 18 images, where the bold font represents the best performance.

method	PSNR			CPSNR
	R	G	B	
AHD [11]	33.00	36.98	32.16	33.49
DLMMSE [12]	34.03	37.99	33.04	34.47
GBTF [8]	33.48	36.59	32.71	33.89
LPA [13]	34.36	37.88	33.30	34.72
LDI-NAT [14]	**36.28**	39.76	34.39	36.20
Proposed	36.09	**40.01**	**35.38**	**36.49**

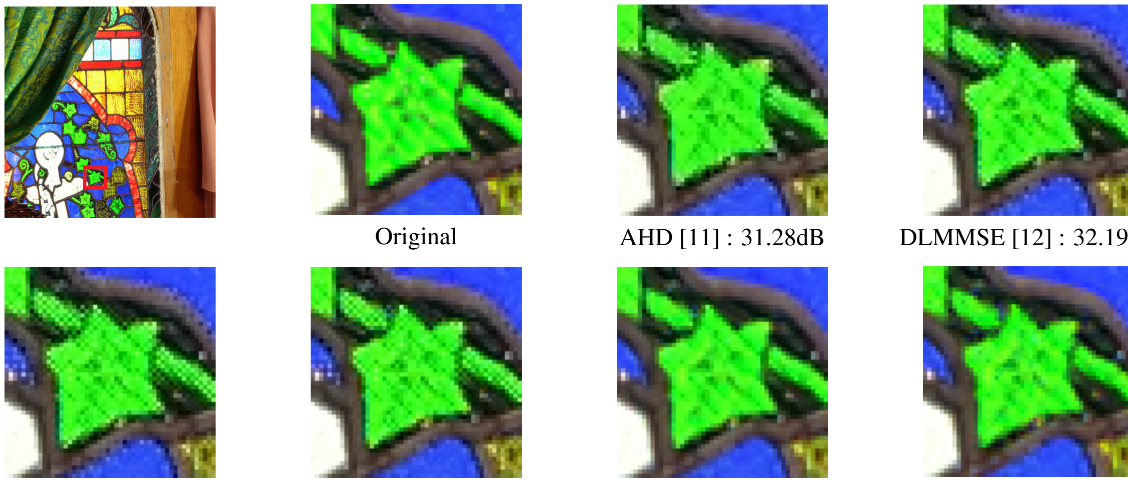
Table 2. The average PSNRs and CPSNRs of Kodak 12 images, where the bold font represents the best performance.

method	PSNR			CPSNR
	R	G	B	
AHD [11]	38.81	40.84	38.42	39.22
DLMMSE [12]	41.17	43.94	40.51	41.62
GBTF [8]	**41.71**	**44.85**	**41.01**	**42.21**
LPA [13]	41.66	44.46	41.00	42.12
LDI-NAT [14]	38.30	40.49	37.94	38.77
Proposed	39.74	42.21	38.90	40.05

Table 3. The average PSNRs and CPSNRs of IMAX and Kodak 30 images, where the bold font represents the best performance.

method	PSNR			CPSNR
	R	G	B	
AHD [11]	35.32	38.52	34.66	35.78
DLMMSE [12]	36.89	40.37	36.02	37.33
GBTF [8]	36.77	39.89	36.03	37.22
LPA [13]	37.28	40.51	36.38	37.68
LDI-NAT [14]	37.09	40.05	35.81	37.23
Proposed	**37.55**	**40.89**	**36.79**	**37.92**

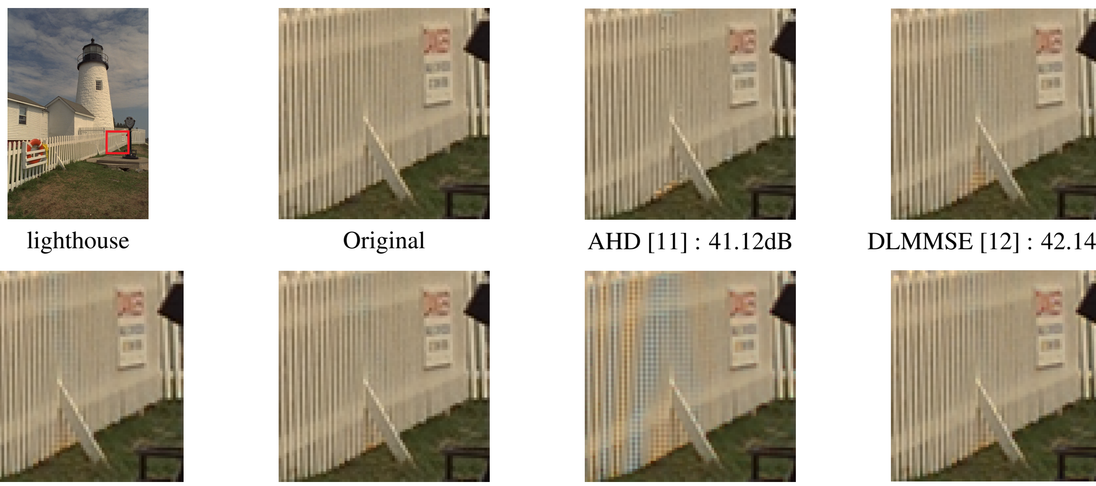
The average PSNRs and CPSNRs of the IMAX 18 images are shown in Table 1. The average CPSNR of the proposed algorithm on IMAX dataset outperforms the state-of-the-art algorithms. The average PSNRs and CPSNRs of the Kodak 12 images are shown in Table 2. The average CPSNR of the proposed algorithm on the Kodak dataset is lower than the GBTF, LPA and DLMMSE algorithms. It is remarkable that several algorithms only work well for one dataset, but do not for another dataset. For example, the LDI-NAT algorithm only works well for the IMAX dataset, in contrast, the GBTF algorithm only works well for the Kodak dataset. Then, we evaluate the average PSNRs and CPSNRs of the whole IMAX and Kodak datasets as shown in Table 3. The proposed algorithm outperforms all the state-of-the-art algorithms in terms of the total average PSNRs and CPSNR. Fig. 3 and Fig. 4 re-



Original | AHD [11] : 31.28dB | DLMMSE [12] : 32.19dB

GBTF [8] : 32.09dB | LPA [13] : 32.33dB | LDI-NAT [14] : 33.22dB | Proposed : 33.26dB

Fig. 3. Visual comparison and local CPSNRs for the star region in the IMAX dataset.



lighthouse | Original | AHD [11] : 41.12dB | DLMMSE [12] : 42.14dB

GBTF [8] : 42.81dB | LPA [13] : 42.97dB | LDI-NAT [14] : 35.38dB | Proposed : 41.57dB

Fig. 4. Visual comparison and local CPSNRs for the fence region of lighthouse in the Kodak dataset.

spectively show the visual comparison of the star region in the IMAX dataset and the fence region of the lighthouse in the Kodak dataset. From these visual comparisons, we can find that the proposed algorithm can sharply interpolate the images without color artifacts for both the IMAX and Kodak datasets.

5. CONCLUSION

In this paper, we proposed the residual interpolation for color image demosaicking. The proposed residual interpolation can be used as an alternative to the widely used color different interpolation. We also proposed a novel demosaicking algorithm by incorporating the residual interpolation into the GBTF algorithm, which is one of the current state-of-the-art demosaicking algorithms. Experimental results demonstrate that our proposed demosaicking algorithm using the residual interpolation can give state-of-the-art performance for the 30 images of Kodak and IMAX datasets. The source code of our proposed algorithm can be downloaded from http://www.ok.ctrl.titech.ac.jp/res/DM/RI.html.

6. REFERENCES

[1] R. Lukac, *Single-sensor imaging: methods and applications for digital cameras*, CRC Press, 2008.

[2] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, “Demosaicking:color filter array interpolation,” *IEEE Signal Processing Magazine*, vol. 22, pp. 44–54, 2005.

[3] X. Li, B. Gunturk, and L. Zhang, “Image demosaicing: a systematic survey,” *Proc. of SPIE*, vol. 6822, pp. 68221J–68221J–15, 2008.

[4] B. Bayer, “Color imaging array,” *U.S. Patent 3971065*, 1976.

[5] J.F. Hamilton Jr and J.E. Adams Jr, “Adaptive color plan interpolation in single sensor color electronic camera,” *U.S. Patent 5629734*, 1976.

[6] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, “Color plane interpolation using alternating projections,” *IEEE Trans. on Image Processing*, vol. 11, no. 9, pp. 997–1013, 2002.

[7] K. He, J. Sun, and X. Tang, “Guided image filtering,” *Proc. of the 11th European Conf. on Computer Vision (ECCV)*, vol. 6311, pp. 1–14, 2010.

[8] I. Pekkucuksen and Y. Altunbasak, “Gradient based threshold free color filter array interpolation,” *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, pp. 137–140, 2010.

[9] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, “Joint bilateral upsampling,” *ACM Trans. on Graphics*, vol. 26(3), no. 96, 2007.

[10] Y. Monno, M. Tanaka, and M. Okutomi, “Multispectral demosaicking using guided filter,” *Proc. of SPIE*, vol. 8299, pp. 82990O–1–82990O–7, 2012.

[11] K. Hirakawa and T. W. Parks, “Adaptive homogeneity-directed demosaicking algorithm,” *IEEE Trans. on Image Processing*, vol. 14, no. 3, pp. 360–369, 2005.

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

[13] D. Paliy, V. Katkovnik, R. Bilcu, S. Alenius, and K. Egiazarian, “Spatially adaptive color filter array interpolation for noiseless and noisy data,” *Int. Journal of Imaging Systems and Technology*, vol. 17, no. 3, pp. 105–122, 2007.

[14] L. Zhang, X. Wu, A. Buades, and X. Li, “Color demosaicking by local directional interpolation and nonlocal adaptive thresholding,” *Journal of Electronic Imaging*, vol. 20, no. 2, pp. 023016–023016, 2011.