American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402
© Global Society of Scientific Research and Researchers
http://asrjetsjournal.org/

Saliency Detection Gradient Preservation for Bayer Image Color Reconstruction

TANG Chun-ming[a]*, Hu Li-li[b]

[a]School of Artificial Intelligence Institute, Tianjin Polytechnic University, Tianjin, 300387, China
[b]School of Electronics and Information Engineering, Tianjin Polytechnic University, Tianjin, 300387, China
[a]Email: tangchunming@tjpu.edu.cn
[b]Email: 2982478813@qq.com

Abstract

Image color reconstruction is a necessary process to recover high quality full color images from Bayer images. In view of the existence of image texture and edge blurring in color reconstruction algorithms, a four-direction joint gradient weighted residual interpolation algorithm is proposed, which uses four-direction weights obtained from RGB pixel gradients and residual gradients in Bayer images, linearly combined with the color difference estimation to effectively obtain the full G image. Aiming at the color cast phenomenon of the image after color interpolation, a saliency detection gradient-preserving color correction algorithm is proposed based on the RGB image captured under natural light. Firstly, the saliency detection method is used to segment the interpolated image and the RGB image into two regions, then carrying out the region correspondence for gradient-preserving color correction, and finally the weighted fusion method is used to obtain the final color reconstructed image. The experimental results show that the reconstructed image texture and edges are clearer and the colors are closer to RGB images.

Keywords: Bayer image; Color correction; Four-direction weights; Residual interpolation; Saliency detection.

1. Introduction

Considering the high cost and large space of using three sensors, most digital cameras use a single sensor to capture image data by covering a surface with a Color Filter Array (CFA). The most commonly used CFA is the Bayer array [1]. In this array format, each pixel position samples only one of the RGB colors.

----

* Corresponding author.

The other two missing color values need to be obtained through an interpolation process. This process is called color interpolation or demosaicking [2]. In recent years, in order to improve the demosaicking performance, researchers have proposed a number of demosaicking algorithms [3-7]. Based on the directional linear minimum mean square-error estimation (DLMMSE) framework, Zhang and Wu [3] proposed an adaptive algorithm to improve the problem. They assumed that the primary difference signals between G and R or G and B channels are low-pass. In both horizontal and vertical directions, the missing G pixel values are adaptively estimated by DLMMSE. Since it only considers the color difference values in the horizontal and vertical directions, the interpolated image has very serious zipper artifacts. Pekkucuksen and Altunbasak [4] proposed a gradient-based threshold free (GBTF) algorithm, which uses the color difference gradient and color difference estimation in four directions to obtain the G channel values, but it does not fully consider the edge characteristics of the image, so that the edge of the interpolated image is blurred and the noise is more. Kiku and his colleagues [5] proposed a demosaicking algorithm using residual interpolation (RI), which replaces the color difference interpolation with residual interpolation based on [4], although the residual color is smoother than the color difference processing, it can generate more accurate preliminary estimates and get better interpolation effect, but it also does not take into account the chrominance mutation area of the image edge, which is easy to cause the edge of the interpolated image to be blurred. They later improved the RI algorithm and proposed the minimum Laplacian residual interpolation algorithm (MLRI) by combining the RI algorithm with the minimum Laplace energy of the residual [6]. Although this method mitigates the pseudo-color effect of the interpolated image, the edges of the image are still unclear. The adaptive residual interpolation (ARI) algorithm [7] adaptively combined two RI-based algorithms in [5, 6] and selected the appropriate number of iterations at each pixel to improve the existing RI- based algorithms. However, the weakness of this algorithm is that the algorithm has high computational complexity. Due to the change of the light source in the scene, the Bayer images captured by the camera tend to produce color shifts. For example, the image taken under the fluorescent lamp is greenish, and the image taken under the tungsten lamp is yellowish. Therefore, the image after demosaicking needs to be color corrected to restore the true color. The simplest color correction method is the global method proposed by Reinhard and his colleagues [8], which uses the mean and standard deviation of each channel in the Lab color space to complete color correction. Although the method is efficient, simple mean and standard deviation matching tends to make the resulting image texture poor and produce color distortion. Yao and his colleagues [9] proposed an example-based color correction method that preserves the gradient details of the image to be corrected by constructing a Laplacian pyramid and corrects the color of the image by using an patch matching method. However, it does not take into account the luminance channel of the image, resulting in over-exposure of the resulting corrected image and color distortion. Xie and his colleagues [10] proposed a high-fidelity color correction method for overall tone and structure refinement. They first decompose the image into a tone layer and a structure layer, then maintain the structure layer, optimize the tone layer, and finally the structure layer and the new tone layer are combined to obtain a refined corrected image. Although this method maintains the structure of the image, it is a refinement of the tone layer that is taken after the histogram matching of the image, and it is easy to introduce wrong colors. Grogan and his colleagues [11] first used the Gaussian mixture model to model the color distribution of the image, and then achieved the color correction of the image by minimizing the L2 distance between the color distributions. The image obtained by this method has high detail retention and clear texture, but it has a high computational complexity. This paper proposes a four-direction joint gradient weighted residual interpolation algorithm. First, for the G channel, the MLRI method is used to obtain the initial estimated values of the color difference in the horizontal and vertical directions, and then the gradation is linearly weighted with the four directions obtained from the original Bayer image plane and the residual plane to obtain the combined color difference. The color difference value is added to the original R/B channel value to obtain the full G plane. Second, the MLRI method is applied to the recovery of R/B channel values. Aiming at the color cast phenomenon of the image after color interpolation, we propose a saliency detection gradient-preserving color correction algorithm. First, we construct a saliency map of the color interpolation image and the RGB image to softly segment the image into foreground and background, and then calculate the weighted global mean and standard deviation of the image, where the saliency map is used as a weight function, Then add the gradient retention factor based on [8] to correct the two regions separately, and finally using the weighted fusion method to obtain the final result image.

2. The Proposed Algorithm

The G plane is usually reconstructed first because it contains twice as many samples as the R or B planes. Thus, the G plane possesses most of the spatial information of the image to be demosaicked and has great influence on the perceptual quality of the image. In this paper, the G component is first estimated by a four-direction joint gradient weighted residual interpolation algorithm, and then the R and B components are estimated by the method in MLRI [6].

2.1. G channel color interpolation

We here only focus on interpolating the G color pixel values at the given R color pixel positions, and the missing G color pixel values at the given B color pixel positions are estimated in the same manner. We first use the method in MLRI [6] to get the horizontal and vertical color difference values of the R color pixel positions $\Delta_{g,r}^{H}(i,j)$ and $\Delta_{g,r}^{V}(i,j)$. Then they are weighted and averaged in the four directions of east (E), west (W), north (N) and south (S), and the combined color difference $\tilde{\Delta}_{g,r}(i,j)$ is obtained as in (1):

$$\tilde{\Delta}_{g,r}(i,j)=\left\{\begin{matrix}\omega_{i,j}^{N}*f_1*\Delta_{g,r}^{V}(i\text{-}3:i,j)+\omega_{i,j}^{S}*f_2*\Delta_{g,r}^{V}(i:i+3,j)+\\ \omega_{i,j}^{E}*f_1^{T}*\Delta_{g,r}^{H}(i,j\text{-}3:j)+\omega_{i,j}^{W}*f_2^{T}*\Delta_{g,r}^{H}(i,j:j+3)\end{matrix}\right\}/\,\omega_{i,j}^{T}$$

$$\omega_{i,j}^{T}=\omega_{i,j}^{N}+\omega_{i,j}^{S}+\omega_{i,j}^{E}+\omega_{i,j}^{W} \tag{1}$$

Where f_1 and f_2 are 1x4 Gaussian filter with a standard deviation of 1, According to [6], take f_1 = [0.01, 0.08, 0.35, 0.56], f_2 = [0.56, 0.35, 0.08, 0.01]. $\omega_{i,j}^{N}$, $\omega_{i,j}^{S}$, $\omega_{i,j}^{E}$ and $\omega_{i,j}^{W}$ are the weights of the four directions of N, S, E and W, respectively. The specific values are defined as in (2):

$$\omega_{i,j}^{N} = 1/(\sum_{a=-1}^{1}\sum_{b=-1}^{1}\nabla^{V}(i-1+a,j+b))^{2}, \quad \omega_{i,j}^{S} = 1/(\sum_{a=-1}^{1}\sum_{b=-1}^{1}\nabla^{V}(i+1+a,j+b))^{2}$$

$$\omega_{i,j}^{S} = 1/(\sum_{a=-1}^{1}\sum_{b=-1}^{1}\nabla^{V}(i+a,j\text{-}1+b))^{2}, \quad \omega_{i,j}^{W} = 1/(\sum_{a=-1}^{1}\sum_{b=-1}^{1}\nabla^{H}(i+a,j+1+b))^{2} \qquad (2)$$

Where ∇^{H} and ∇^{V} represent the joint gradients in the horizontal and vertical directions, respectively, which are calculated as in (3):

$$\nabla^{H}(i,j) = \left|Z_{i,j-1} - Z_{i,j+1}\right| + \left|Z_{i,j-1}^{H} - Z_{i,j+1}^{H}\right| + \left|\Delta Z_{i,j-1}^{H} - \Delta Z_{i,j+1}^{H}\right|$$

$$\nabla^{V}(i,j) = \left|Z_{i-1,j} - Z_{i+1,j}\right| + \left|Z_{i-1,j}^{V} - Z_{i+1,j}^{V}\right| + \left|\Delta Z_{i-1,j}^{V} - \Delta Z_{i+1,j}^{V}\right| \qquad (3)$$

Different from the color difference gradient obtained by the correlation between color channels in [6], the joint gradient we proposed uses the correlation within the same color channel, which is obtained from three planes: 1) the original Bayer pattern plane (*Z*); 2) the horizontally and vertically interpolated Bayer planes (Z^{H}, Z^{V}); 3) the horizontal and vertical residual planes (ΔZ^{H}, ΔZ^{V}), which can be obtained by the method in MLRI [6]. The directionally interpolated Bayer plane is given as in (4):

$$Z^{H} = Z * f, \; Z^{V} = Z * f^{T} \qquad (4)$$

Where f = [1/4, 1/2, -1/2, 1/2, 1/4] is the interpolation filter [12].

Finally, we obtain the G pixel values at the R pixel positions by adding the given R pixel values as in (5):

$$G_{R}^{'}(i,j) = R(i,j) + \tilde{\Delta}_{g,r}(i,j) \qquad (5)$$

2.2. G channel refinement

After the G channel is reconstructed, in order to further suppress visible color interpolation artifacts and enhance the edge definition of the image, we refine it based on the color difference plane as in (6):

$$\begin{aligned} G(i,j) = {} & R(i,j) + \alpha(G_{R}^{'}(i,j) - R(i,j)) \\ & + (1-\alpha)\frac{\omega^{N}(i,j)\breve{G}^{N}(i,j) + \omega^{S}(i,j)\breve{G}^{S}(i,j) + \omega^{W}(i,j)\breve{G}^{W}(i,j) + \omega^{E}(i,j)\breve{G}^{E}(i,j)}{\omega^{N}(i,j) + \omega^{S}(i,j) + \omega^{W}(i,j) + \omega^{E}(i,j)} \end{aligned} \qquad (6)$$

Where α is the weight parameter, assumed to be α = 0.9 based on empirical evidence. The parameters $\breve{G}^{N}(i,j)$, $\breve{G}^{S}(i,j)$, $\breve{G}^{W}(i,j)$ and $\breve{G}^{E}(i,j)$ represent the color differences in the N, S, E, and W directions, respectively. These parameters are calculated as in (7):

$$\breve{G}^N(i,j) = G(i-1,j) - 0.5\times(R(i,j)+R(i-2,j))$$

$$\breve{G}^S(i,j) = G(i+1,j) - 0.5\times(R(i,j)+R(i+2,j))$$

$$\breve{G}^W(i,j) = G(i,j-1) - 0.5\times(R(i,j)+R(i,j-2))$$

$$\breve{G}^E(i,j) = G(i,j+1) - 0.5\times(R(i,j)+R(i,j+2)) \tag{7}$$

In order to ensure an accurate final estimate, we need different weight parameters, so the weights in the four directions $\omega^N(i,j)$, $\omega^S(i,j)$, $\omega^W(i,j)$ and $\omega^E(i,j)$ can be defined as in (8):

$$\omega^X(i,j) = \frac{1}{\nabla^X(i,j)},\ \ X \in \{\mathrm{N,S,W,E}\} \tag{8}$$

The color differences obtained using the reconstructed G pixel are determined as $\nabla^N(i,j)$, $\nabla^S(i,j)$, $\nabla^W(i,j)$ and $\nabla^E(i,j)$, which are calculated as in (9):

$$\nabla^N(i,j) = \left|G_R^{'}(i,j) - \mathrm{G}(i-1,j)\right| + \left|R(i,j) - R(i-2,j)\right|$$

$$\nabla^S(i,j) = \left|G_R^{'}(i,j) - \mathrm{G}(i+1,j)\right| + \left|R(i,j) - R(i+2,j)\right|$$

$$\nabla^W(i,j) = \left|G_R^{'}(i,j) - \mathrm{G}(i,j-1)\right| + \left|R(i,j) - R(i,j-2)\right|$$

$$\nabla^E(i,j) = \left|G_R^{'}(i,j) - \mathrm{G}(i,j+1)\right| + \left|R(i,j) - R(i,j+2)\right| \tag{9}$$

2.3. Color correction

Bayer images taken with a camera under fluorescent light are generally greenish. To improve this situation, we propose a saliency detection gradient-preserving color correction algorithm. We first convert the image I_d obtained by the color interpolation process and the RGB image I_{RGB} captured under natural light into the CIELab color space, and Then the saliency detection methods in [13] are used to calculate the saliency maps W_d and W_{RGB} of the color interpolation image I_d and the RGB image I_{RGB}, respectively (see Figure 4(b) (c)). The saliency map segment the image into foreground and background. Here, the pixels with high saliency value, which attracts the visual attention, are called foreground. On the contrary, the pixels with low saliency value are called background.

$\mu_{ds} = \sum_{i\in\Omega_d} \omega_d(i)v_d(i) \Big/ \sum_{i\in\Omega_d} \omega_d(i)$ and $\sigma_{ds} = \sqrt{\sum_{i\in\Omega_d} \omega_d(i)(v_d(i)-\mu_{ds})^2 \Big/ \sum_{i\in\Omega_d} \omega_d(i)}$ are the mean and standard deviation of the foreground of the I_d , respectively. Where Ω_d is the region of I_d; i is an arbitrary pixel in Ω_d; $\omega_d(i)$ is the saliency value of I; and $v_d(i)$ is the pixel value of i.

The weighted mean and standard deviation of the background of I_d are

$\mu_{dn} = \sum_{i\in\Omega_d} (1-\omega_d(i))v_d(i) \Big/ \sum_{i\in\Omega_d} (1-\omega_d(i))$ and $\sigma_{dn} = \sqrt{\sum_{i\in\Omega_d} (1-\omega_d(i))(v_d(i)-\mu_{dn})^2 \Big/ \sum_{i\in\Omega_d} (1-\omega_d(i))}$, respectively. The weighted means μ_{rs} , μ_{rn} and standard deviations σ_{rs} , σ_{rn} of I_{RGB} is computed in the same manner.

When color correction is performed separately for the foreground and background of I_d using [8], we introduce gradient factor λ_1 and λ_2 to preserve the details of the image while correcting the color of the image. The corrections are performed as in (10):

$$v_{os}(i) = (v_d(i)-\mu_{ds})(\lambda_1 \frac{\sigma_{rs}}{\sigma_{ds}} + \lambda_2) + \mu_{rs}$$

$$v_{on}(i) = (v_d(i)-\mu_{dn})(\lambda_1 \frac{\sigma_{rn}}{\sigma_{dn}} + \lambda_2) + \mu_{rn} \tag{10}$$

Where λ_1 and λ_2 satisfy $\lambda_1 + \lambda_2 = 1$. v_{os} is the output of foreground corrected, v_{on} is the output of background corrected.

Finally, the color corrected output image I_o is obtained by the weighted fusion method (see Figure 4(e)), which is computed as in (11):

$$I_o(i) = \omega_d(i)v_{os}(i) + (1-\omega_d(i))v_{on}(i) \tag{11}$$

3. Experimental Results and Discussion

3.1. Comparison and analysis of color interpolation results

To evaluate the performance of the proposed algorithm, we tested it on 18 McMaster datasets images with a resolution of 500×500, as listed in Figure 1. We chose the McMaster datasets because the images have lower spectral correlation and are similar to nature images captured by color sensors. The proposed algorithm was compared with GBTF [4], ESF [14], MSG [15], RI [5], ECC [16] and MLRI [6].

Figure1: 18 McMaster test images.

To simulate the proposed algorithm, we conducted experiments using MATLAB with an Intel Core i5-7200U and 2.5GHz CPU processer. To avoid the boundary effects, every calculation excluded the border of 10 pixels around the image. In the experiment, the peak signal-to-noise ratio (PSNR) and the composite PSNR (CPSNR) were used as indicators to evaluate the objective performance of the demosaicking algorithms.

Table1: The PSNR (dB) of G components for the McMaster images.

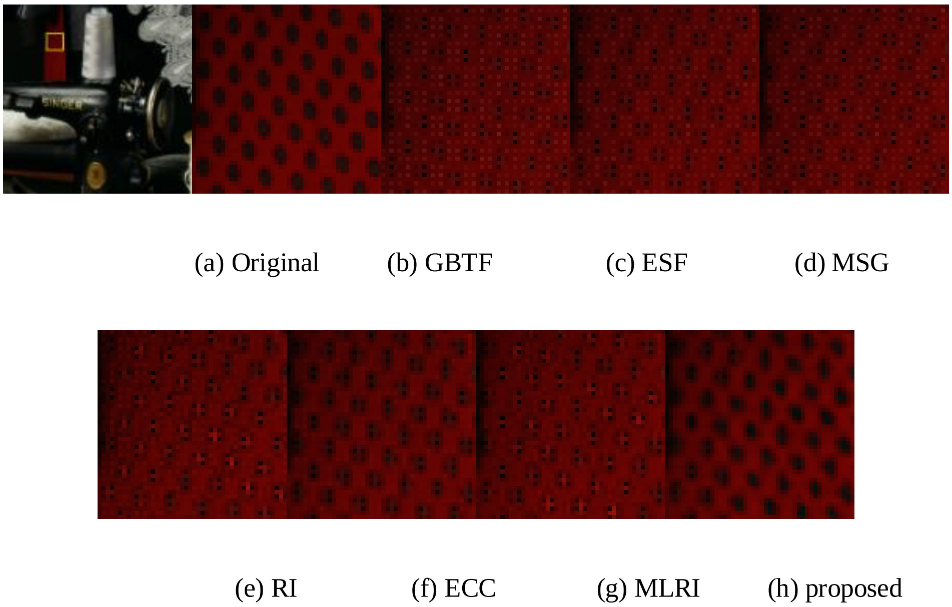
Image.	GBTF	ESF	MSG	RI	ECC	MLRI	Proposed
1	29.99	28.47	30.21	32.38	32.37	32.79	**32.81**
2	37.20	36.20	37.30	39.42	39.44	39.63	**39.69**
3	35.29	34.48	35.72	36.73	36.75	36.91	**37.10**
4	37.28	36.71	38.87	42.10	42.14	41.84	**43.21**
5	34.22	33.17	34.52	37.85	37.86	38.36	**38.64**
6	36.39	34.75	36.81	42.22	42.16	42.60	**43.09**
7	41.44	41.44	**41.77**	38.77	38.77	38.73	38.53
8	41.36	41.08	**41.64**	41.40	41.37	41.25	41.48
9	37.92	36.98	38.39	41.63	41.62	41.92	**42.53**
10	39.13	37.77	39.49	42.10	42.07	42.64	**42.97**
11	39.33	38.21	39.61	42.00	42.03	42.27	**42.52**
12	39.48	38.14	39.64	42.23	42.24	42.38	**42.50**
13	41.89	41.22	42.08	45.13	45.10	45.33	**45.86**
14	40.66	39.96	40.77	43.06	43.05	43.28	**43.70**
15	40.02	39.26	40.15	42.81	42.67	42.91	**43.23**
16	32.13	30.50	32.15	35.26	35.16	35.54	**35.61**
17	31.85	30.57	32.28	37.41	37.38	37.97	**38.19**
18	36.56	35.54	36.36	**37.75**	37.68	37.71	**37.75**
average	37.34	36.36	37.65	40.01	39.99	40.23	**40.52**

Table 1 shows the PSNR of different algorithms to G components, since G components are the peak sensitivity of HVS. Table 2 shows the CPSNR of different algorithms. As shown in Table 1, the proposed algorithm performs better than the other six algorithms on 16 out of 18 images. The average PSNR of G components in the proposed algorithm is higher than MLRI and RI by 0.29 dB and 0.51 dB, respectively. Moreover, as for CPSNR in Table 2, the proposed algorithm performs better than the other six algorithms on 14 out of 18 images. The average CPSNR of the proposed algorithm is higher than MLRI and ECC by 0.37 dB and 0.50 dB, respectively. This demonstrates that the proposed algorithm is effective for image demosaicking.

Table2: The CPSNR (dB) of McMaster images.

Image.	GBTF	ESF	MSG	RI	ECC	MLRI	Proposed
1	26.89	26.07	27.21	28.90	29.22	29.41	**29.61**
2	33.62	33.08	33.82	34.97	35.17	35.35	**35.40**
3	32.69	32.32	33.05	33.67	33.42	34.05	**34.13**
4	34.74	34.67	35.74	37.90	36.45	38.00	**38.86**
5	30.92	30.31	31.29	33.89	34.43	34.43	**34.83**
6	33.21	32.09	33.76	38.36	38.47	38.83	**39.50**
7	38.97	38.81	39.16	36.95	37.00	**37.04**	36.52
8	37.50	37.32	37.67	37.09	38.66	37.30	**38.79**
9	34.31	33.95	34.86	35.97	37.15	36.84	**37.70**
10	36.25	35.57	36.69	38.20	38.54	39.12	**39.37**
11	37.04	36.43	37.43	39.48	39.91	40.21	**40.37**
12	36.65	35.91	36.92	39.61	39.78	**39.84**	39.83
13	38.65	38.29	38.97	40.31	**41.12**	40.66	40.96
14	37.10	36.67	37.25	38.95	38.98	39.11	**39.37**
15	37.09	36.71	37.30	38.32	39.32	39.25	**39.55**
16	30.08	28.97	30.24	35.21	35.12	35.42	**35.66**
17	29.02	28.35	29.51	32.35	33.37	33.19	**34.00**
18	34.04	33.46	34.13	**36.57**	35.95	36.41	36.54
average	34.38	33.83	34.72	36.48	36.78	36.91	**37.28**

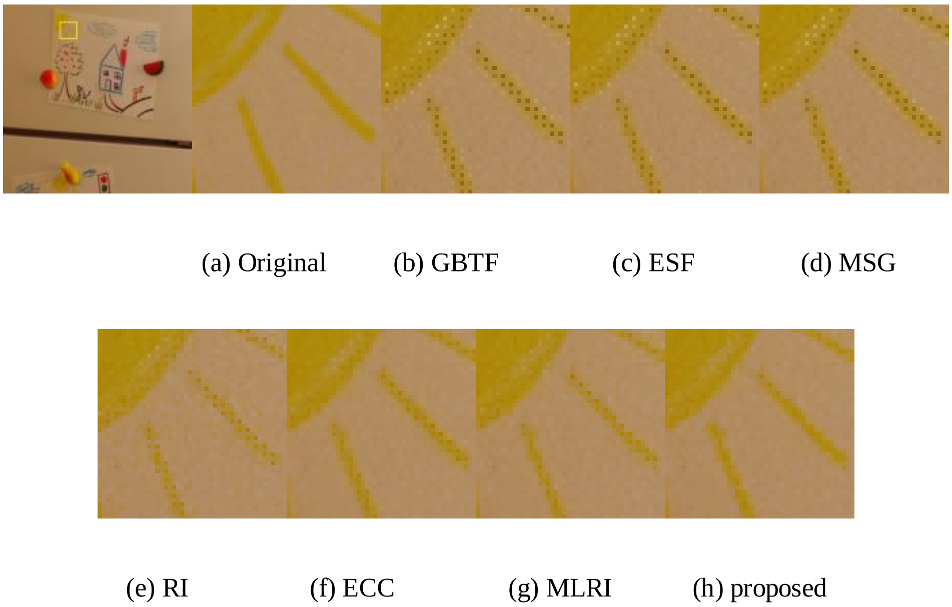
Objective measures are not reliable enough to judge the performance of the proposed method sometimes. Thus, we choose the McMaster number 8 and number 13 images to show visual comparison of the interpolated images. Figure 2 shows the visual comparison of demosaicking results on the McMaster number 8 image. Figure 3 shows the visual comparison of demosaicking results on the McMaster number 13 image. We can find that the proposed algorithm produces precise image color information in details. In Figure 2, we can see that the image obtained by the proposed algorithm has good edge recovery and clear texture. In Figure 3, the proposed algorithm effectively suppresses the zipper artifacts, and the obtained image noise is less.



(a) Original (b) GBTF (c) ESF (d) MSG

(e) RI (f) ECC (g) MLRI (h) proposed

Figure 2: Visual comparison of the demosaicking results on the McMaster number 8 image.



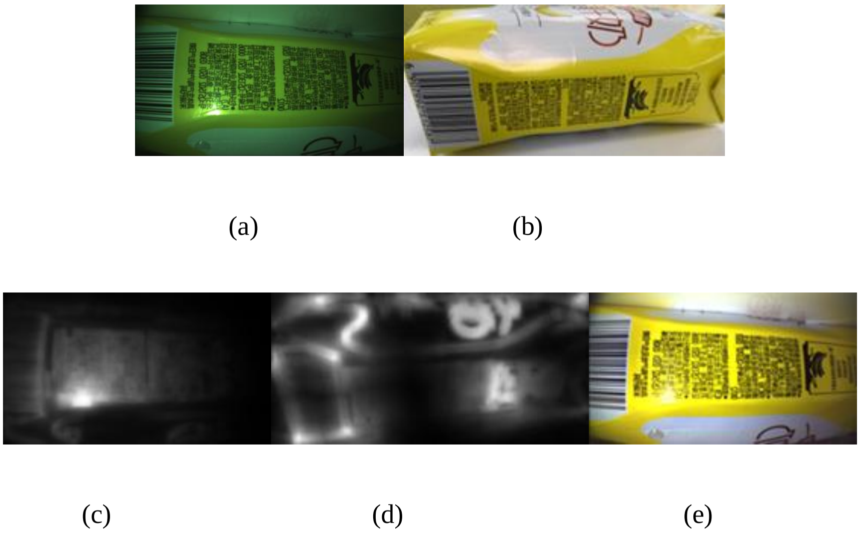
(a) Original (b) GBTF (c) ESF (d) MSG

(e) RI (f) ECC (g) MLRI (h) proposed

Figure3: Visual comparison of the demosaicking results on the McMaster number 13 image.

3.2. Comparison and analysis of color correction results

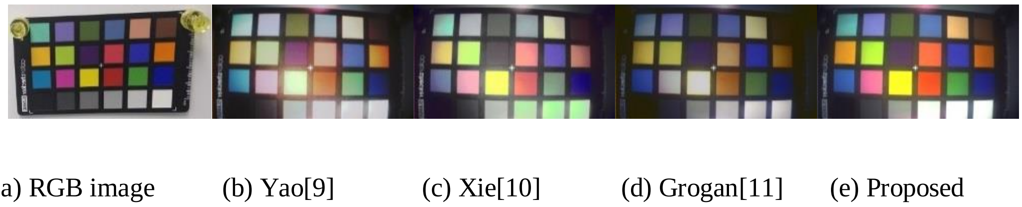
To evaluate the performance of the proposed color correction algorithm, we use the image processed by the demosaicking algorithm taken under the fluorescent lamp as the test image, and the RGB image taken under natural light as the standard image. The proposed algorithm is compared with [9,10,11], Figure 5 shows the visual comparison of the color correction results, in Figure 4(e) λ_1=0.7，λ_2=0.3. In Figure 5, the color of the image obtained by the proposed algorithm is more similar to the RGB image. We also calculated the PSNR values for the six colors, as shown in Figure 6. It can be seen that the PSNR values of the image obtained by the proposed algorithm are the highest in black, yellow and blue except for green and red. Compared with the other three algorithms, the algorithm has better color rendition abilities.



(a) (b)

(c) (d) (e)

Figure4: The color correction pipeline. (a) The color interpolation image I_d; (b) RGB image I_{RGB}; (c) The saliency map of I_d; (d) The saliency map of I_{RGB}; (e) The color corrected output image I_o.



(a) RGB image (b) Yao[9] (c) Xie[10] (d) Grogan[11] (e) Proposed

Figure 5: Visual comparisons of the color correction results.

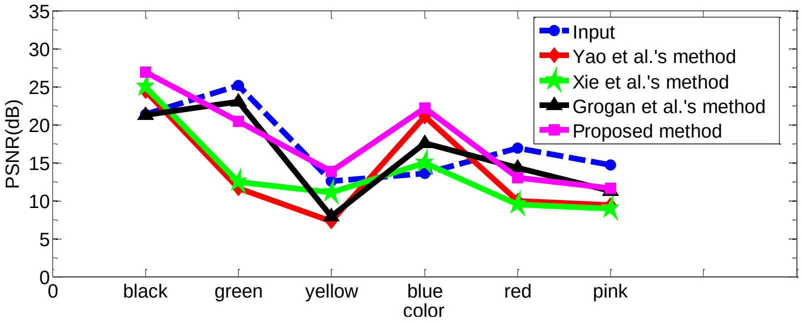


Figure 6: Comparison of PSNR values of six colors after color reconstruction.

4. Conclusions

In this paper, we propose an effective four-direction joint gradient weighted residual interpolation algorithm and a saliency detection gradient-preserving color correction algorithm. The proposed demosaicking algorithm uses the RGB pixel gradients and residual gradients of the original Bayer images as the weights of the four directions to generate the full G image. The interpolation of R/B values is the same as MLRI. The proposed color correction algorithm segments the demosaicked image and the standard image into two regions by using the saliency detection method, and adds a gradient preserving factor during the region correspondence color correction process, which avoids the loss of the detail information of the image to be corrected. As shown in the experimental results, the proposed algorithm has different degrees of improvement on the test image results subjectively and objectively compared to existing algorithms.

5. Recommendations

This paper analyzes the existing problems of Bayer image color reconstruction algorithms, and based on calculating the color difference estimations, we use a novel method to effectively reduce the influence of false edge information in irregular edges. It provides better color fidelity. For further work, we will continue to focus on the improved algorithm with better visual effect on more types of images.

References

[1] B.E. Bayer. “Color Imaging Array,” U.S. Patent 3 971 065, July. 20, 1976.

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

[3] L. Zhang and X. Wu. “Color demosaicking via directional linear minimum mean square-error estimation,” IEEE Trans. Image Process., vol. 14, pp. 2167-2178, Dec. 2005.

[4] I. Pekkucuksen and Y. Altunbasak. “Gradient based threshold free color filter array interpolation,” in Proc. IEEE Int. Conf. Image Process. (ICIP), 2010, pp. 137-140.

[5] D. Kiku, Y. Monno, M. Tanaka and M. Okutomi. “Residual interpolation for color image demosaicking,” in Proc. IEEE Int. Conf. Image Process. (ICIP), 2013, pp. 2304-2308.

[6] D. Kiku, Y. Monno, M. Tanaka and M. Okutomi. “Beyond Color Difference: Residual Interpolation for Color Image Demosaicking,” IEEE Trans. Image Process., vol. 25, pp. 1288-1300, Mar. 2016.

[7] D. Kiku, Y. Monno, M. Tanaka and M. Okutomi. “Adaptive Residual Interpolation for Color and Multispectral Image Demosaicking,” Sensors, vol. 17, pp. 2787-2807, Dec. 2017.

[8] E. Reinhard, M. Ashikhmin, B. Gooch and P. Shirley. “Color Transfer between Images,” IEEE Computer Graphics and Applications, vol. 21, pp. 34-41, Oct. 2001.

[9] C.H. Yao, C.Y. Chang and S.Y. Chien. “Example-based video color transfer,” IEEE International Conference on Multimedia and Expo(ICME), 2016, pp.1-6.

[10] Z. Xie, S. Ding, B. Sheng, L. Ma. “Integrated tone and structure refinement for high-fidelity colour

transfer," Iet Image Processing, vol. 11, pp. 1281-1290, Oct. 2017.

[11] M. Grogan and R. Dahyot. "L2 Divergence for robust colour transfer," Computer Vision and Image Understanding, vol. 181, pp. 39-49, Feb. 2019.

[12] J.F. Hamilton and J.E. Adams. "Adaptive color plan interpolation in single sensor color electronic camera," U.S. Patent 5 629 734, May. 13, 1997.

[13] S. Goferman, L. Zelnik-Manor and A. Tal. "Context-Aware Saliency Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, pp. 1915-1926, Oct. 2012.

[14] I. Pekkucuksen and Y. Altunbasak. "Edge Strength Filter Based Color Filter Array Interpolation," IEEE Trans. Image Process., vol. 21, pp. 393-397, Jan. 2012.

[15] I. Pekkucuksen and Y. Altunbasak. "Multiscale Gradients-Based Color Filter Array Interpolation," IEEE Trans. Image Process., vol. 22, pp. 157-165, Jan. 2013.

[16] S.P. Jaiswal, O.C. Au, V. Jakhetiya, Y. Yuan and Y. Hai. "Exploitation of inter-color correlation for color image demosaicking," in Proc. IEEE Int. Conf. Image Process. (ICIP), 2014, pp. 1812-1816.